Zbornik 19. mednarodne multikonference

INFORMACIJSKA DRUŽBA - IS 2016

Zvezek A

Proceedings of the 19th International Multiconference

INFORMATION SOCIETY - IS 2016

Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredili / Edited by
Matjaž Gams, Mitja Luštrek, Rok Piltaver

http://is.ijs.si

12. oktober 2016 / 12 October 2016
Ljubljana, Slovenia

Zbornik 19. mednarodne multikonference

INFORMACIJSKA DRUŽBA – IS 2016

Zvezek A

Proceedings of the 19th International Multiconference

INFORMATION SOCIETY – IS 2016

Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredili / Edited by

Matjaž Gams, Mitja Luštrek, Rok Piltaver

12. oktober 2016 / 12 October 2016
Ljubljana, Slovenia

Uredniki:

Matjaž Gams
Odsek za inteligentne sisteme
Institut »Jožef Stefan«, Ljubljana

Mitja Luštrek
Odsek za inteligentne sisteme
Institut »Jožef Stefan«, Ljubljana

Rok Piltaver
Odsek za inteligentne sisteme
Institut »Jožef Stefan«, Ljubljana

Založnik: Institut »Jožef Stefan«, Ljubljana
Priprava zbornika: Mitja Lasič, Vesna Lasič, Lana Zemljak
Oblikovanje naslovnice: Vesna Lasič

Dostop do e-publikacije:
http://library.ijs.si/Stacks/Proceedings/InformationSociety

Ljubljana, oktober 2016

CIP - Kataložni zapis o publikaciji
Narodna in univerzitetna knjižnica, Ljubljana

004.8(082)(0.034.2)

SLOVENSKA konferenca o umetni inteligenci (2016 ; Ljubljana)
Slovenska konferenca o umetni inteligenci [Elektronski vir] : zbornik 19. mednarodne multikonference Informacijska družba - IS 2016, 12. oktober 2016, [Ljubljana, Slovenija] : zvezek A = Slovenian Conference on Artificial Intelligence : proceedings of the 19th International Multiconference Information Society - IS 2016, 12 October 2016, Ljubljana, Slovenia : volume A / uredili, edited by Matjaž Gams, Mitja Luštrek, Rok Piltaver. - El. zbornik. - Ljubljana : Institut Jožef Stefan, 2016

Način dostopa (URL):
http://library.ijs.si/Stacks/Proceedings/InformationSociety/2016/IS2016_Volume_A%20-%20SKUI.pdf

ISBN 978-961-264-097-2 (pdf)
1. Gl. stv. nasl. 2. Vzp. stv. nasl. 3. Gams, Matjaž, računalništvo 4. Mednarodna multikonferenca Informacijska družba (19 ; 2016 ; Ljubljana)
1537202883

PREDGOVOR MULTIKONFERENCI INFORMACIJSKA DRUŽBA 2016

Multikonferenca Informacijska družba (http://is.ijs.si) je z devetnajsto zaporedno prireditvijo osrednji srednjeevropski dogodek na področju informacijske družbe, računalništva in informatike. Letošnja prireditev je ponovno na več lokacijah, osrednji dogodki pa so na Institutu »Jožef Stefan«.

Informacijska družba, znanje in umetna inteligenca so spet na razpotju tako same zase kot glede vpliva na človeški razvoj. Se bo eksponentna rast elektronike po Moorovem zakonu nadaljevala ali stagnirala? Bo umetna inteligenca nadaljevala svoj neverjetni razvoj in premagovala ljudi na čedalje več področjih in s tem omogočila razcvet civilizacije, ali pa bo eksponentna rast prebivalstva zlasti v Afriki povzročila zadušitev rasti? Čedalje več pokazateljev kaže v oba ekstrema – da prehajamo v naslednje civilizacijsko obdobje, hkrati pa so planetarni konflikti sodobne družbe čedalje težje obvladljivi.

Letos smo v multikonferenco povezali dvanajst odličnih neodvisnih konferenc. Predstavljenih bo okoli 200 predstavitev, povzetkov in referatov v okviru samostojnih konferenc in delavnic. Prireditev bodo spremljale okrogle mize in razprave ter posebni dogodki, kot je svečana podelitev nagrad. Izbrani prispevki bodo izšli tudi v posebni številki revije Informatica, ki se ponaša z 39-letno tradicijo odlične znanstvene revije. Naslednje leto bo torej konferenca praznovala 20 let in revija 40 let, kar je za področje informacijske družbe častitljiv dosežek.

Multikonferenco Informacijska družba 2016 sestavljajo naslednje samostojne konference:

- 25-letnica prve internetne povezave v Sloveniji
- Slovenska konferenca o umetni inteligenci
- Kognitivna znanost
- Izkopavanje znanja in podatkovna skladišča
- Sodelovanje, programska oprema in storitve v informacijski družbi
- Vzgoja in izobraževanje v informacijski družbi
- Delavnica »EM-zdravje«
- Delavnica »E-heritage«
- Tretja študentska računalniška konferenca
- Računalništvo in informatika: včeraj za jutri
- Interakcija človek-računalnik v informacijski družbi
- Uporabno teoretično računalništvo (MATCOS 2016).

Soorganizatorji in podporniki konference so različne raziskovalne institucije in združenja, med njimi tudi ACM Slovenija, SLAIS, DKZ in druga slovenska nacionalna akademija, Inženirska akademija Slovenije (IAS). V imenu organizatorjev konference se zahvaljujemo združenjem in inštitucijam, še posebej pa udeležencem za njihove dragocene prispevke in priložnost, da z nami delijo svoje izkušnje o informacijski družbi. Zahvaljujemo se tudi recenzentom za njihovo pomoč pri recenziranju.

V 2016 bomo četrtič podelili nagrado za življenjske dosežke v čast Donalda Michija in Alana Turinga. Nagrado Michie-Turing za izjemen življenjski prispevek k razvoju in promociji informacijske družbe bo prejel prof. dr. Tomaž Pisanski. Priznanje za dosežek leta bo pripadlo prof. dr. Blažu Zupanu. Že šestič podeljujemo nagradi »informacijska limona« in »informacijska jagoda« za najbolj (ne)uspešne poteze v zvezi z informacijsko družbo. Limono je dobilo ponovno padanje Slovenije na lestvicah informacijske družbe, jagodo pa informacijska podpora Pediatrične klinike. Čestitke nagrajencem!

Bojan Orel, predsednik programskega odbora
Matjaž Gams, predsednik organizacijskega odbora

FOREWORD - INFORMATION SOCIETY 2016

In its 19th year, the Information Society Multiconference (http://is.ijs.si) remains one of the leading conferences in Central Europe devoted to information society, computer science and informatics. In 2016 it is organized at various locations, with the main events at the Jožef Stefan Institute.

The pace of progress of information society, knowledge and artificial intelligence is speeding up, but it seems we are again at a turning point. Will the progress of electronics continue according to the Moore's law or will it start stagnating? Will AI continue to outperform humans at more and more activities and in this way enable the predicted unseen human progress, or will the growth of human population in particular in Africa cause global decline? Both extremes seem more and more likely – fantastic human progress and planetary decline caused by humans destroying our environment and each other.

The Multiconference is running in parallel sessions with 200 presentations of scientific papers at twelve conferences, round tables, workshops and award ceremonies. Selected papers will be published in the Informatica journal, which has 39 years of tradition of excellent research publication. Next year, the conference will celebrate 20 years and the journal 40 years – a remarkable achievement.

The Information Society 2016 Multiconference consists of the following conferences:

- 25th Anniversary of First Internet Connection in Slovenia
- Slovenian Conference on Artificial Intelligence
- Cognitive Science
- Data Mining and Data Warehouses
- Collaboration, Software and Services in Information Society
- Education in Information Society
- Workshop Electronic and Mobile Health
- Workshop »E-heritage«
- 3st Student Computer Science Research Conference
- Computer Science and Informatics: Yesterday for Tomorrow
- Human-Computer Interaction in Information Society
- Middle-European Conference on Applied Theoretical Computer Science (Matcos 2016)

The Multiconference is co-organized and supported by several major research institutions and societies, among them ACM Slovenia, i.e. the Slovenian chapter of the ACM, SLAIS, DKZ and the second national engineering academy, the Slovenian Engineering Academy. In the name of the conference organizers we thank all the societies and institutions, and particularly all the participants for their valuable contribution and their interest in this event, and the reviewers for their thorough reviews.

For the fourth year, the award for life-long outstanding contributions will be delivered in memory of Donald Michie and Alan Turing. The Michie-Turing award will be given to Prof. Tomaž Pisanski for his life-long outstanding contribution to the development and promotion of information society in our country. In addition, an award for current achievements will be given to Prof. Blaž Zupan. The information lemon goes to another fall in the Slovenian international ratings on information society, while the information strawberry is awarded for the information system at the Pediatric Clinic. Congratulations!

Bojan Orel, Programme Committee Chair
Matjaž Gams, Organizing Committee Chair

KONFERENČNI ODBORI
CONFERENCE COMMITTEES

International Programme Committee

Vladimir Bajic, South Africa
Heiner Benking, Germany
Se Woo Cheon, South Korea
Howie Firth, UK
Olga Fomichova, Russia
Vladimir Fomichov, Russia
Vesna Hljuz Dobric, Croatia
Alfred Inselberg, Israel
Jay Liebowitz, USA
Huan Liu, Singapore
Henz Martin, Germany
Marcin Paprzycki, USA
Karl Pribram, USA
Claude Sammut, Australia
Jiri Wiedermann, Czech Republic
Xindong Wu, USA
Yiming Ye, USA
Ning Zhong, USA
Wray Buntine, Australia
Bezalel Gavish, USA
Gal A. Kaminka, Israel
Mike Bain, Australia
Michela Milano, Italy
Derong Liu, Chicago, USA
Toby Walsh, Australia

Organizing Committee

Matjaž Gams, chair
Mitja Luštrek
Lana Zemljak
Vesna Koricki
Mitja Lasič
Robert Blatnik
Aleš Tavčar
Blaž Mahnič
Jure Šorn
Mario Konecki

Programme Committee

Bojan Orel, chair
Nikolaj Zimic, co-chair
Franc Solina, co-chair
Viljan Mahnič, co-chair
Cene Bavec, co-chair
Tomaž Kalin, co-chair
Jozsef Györkös, co-chair
Tadej Bajd
Jaroslav Berce
Mojca Bernik
Marko Bohanec
Ivan Bratko
Andrej Brodnik
Dušan Caf
Saša Divjak
Tomaž Erjavec
Bogdan Filipič
Andrej Gams
Matjaž Gams
Marko Grobelnik
Nikola Guid
Marjan Heričko
Borka Jerman Blažič Džonova
Gorazd Kandus
Urban Kordeš
Marjan Krisper
Andrej Kuščer
Jadran Lenarčič
Borut Likar
Janez Malačič
Olga Markič
Dunja Mladenič
Franc Novak
Vladislav Rajkovič Grega
Repovš
Ivan Rozman
Niko Schlamberger
Stanko Strmčnik
Jurij Šilc
Jurij Tasič
Denis Trček
Andrej Ule
Tanja Urbančič
Boštjan Vilfan
Baldomir Zajc
Blaž Zupan
Boris Žemva
Leon Žlajpah

KAZALO / TABLE OF CONTENTS

Zbornik 19. mednarodne multikonference

INFORMACIJSKA DRUŽBA – IS 2016

Zvezek A

Proceedings of the 19th International Multiconference

INFORMATION SOCIETY – IS 2016

Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredili / Edited by

Matjaž Gams, Mitja Luštrek, Rok Piltaver

12. oktober 2016 / 12 October 2016
Ljubljana, Slovenia

PREDGOVOR

Slovenska konferenca o umetni inteligenci je naslednica konference Inteligentni sistemi, ki je sestavni del multikonference Informacijska družba že od njenega začetka leta 1997. Za novo ime smo se odločili zaradi tesnejšega sodelovanje s Slovenskim društvom za umetno inteligenco. Pomemben cilj tako konference kot tudi društva je namreč povezovanje slovenskih raziskovalcev s tega področja, čeprav na konferenci niso nič manj dobrodošli tudi prispevki iz drugih držav. Žal letos nismo prejeli dovolj kvalitetnih prispevkov iz tujine, smo pa pritegnili nadpovprečen delež prispevkov iz organizacij izven Instituta »Jožef Stefan«, posebej s Fakultete za računalništvo in informatiko, ki ima skupaj z Institutom vodilno vlogo pri raziskavah umetne inteligence v Sloveniji, kar štejemo za uspeh.

Tema konference ostajajo umetna inteligenca, inteligentni sistemi in inteligentne storitve informacijske družbe. Obravnavamo metode umetne inteligence, njene aplikacije v praksi, vpliv na družbo in gospodarstvo, njeno prihodnost ter priložnosti in nevarnosti, ki jih vnaša v informacijsko družbo. Ker konferenca stremi h kakovostnim mednarodno zanimivim prispevkom, je večina izmed 18 sprejetih v angleščini; a ker gre vendar za slovensko konferenco, ki želi spodbuditi dialog in sodelovanje med domačimi raziskovalci, sprejemamo tudi slovenske. Vsi prispevki so bili recenzirani in popravljeni v skladu z navodili recenzentov. Menimo, da je letošnja kakovost prispevkov visoka in upamo na še višjo prihodnje leto, ko bo konferenca praznovala 20. obletnico.

FOREWORD

Slovenian Conference on Artificial Intelligence is the successor of the Intelligent Systems conference, which has been a part of the Information Society multiconference since its establishment in 1997. The new name was adopted because of a closer collaboration with the Slovenian Artificial Intelligence Society. The reason for this collaboration is that the goal of both the conference and the society is bringing together Slovenian artificial intelligence researchers, although international papers are just as welcome at the conference. Unfortunately we have not received enough such papers this year, but we are proud to have attracted an unusually large number of papers from outside the Jožef Stefan Institute, particularly from the Faculty of Computer and Information Science, which shares the leading role in artificial intelligence research in Slovenia with the Institute.

The topics of the conference remain artificial intelligence, intelligent systems and intelligent services in the information society. The conference addresses artificial intelligence methods, their practical application, the impact of the artificial intelligence on the society and industry, its future, and the opportunities and threats it presents to the information society. Since the conference aims at international relevance, the majority of the 18 accepted papers are in English; however, being a Slovenian conference aiming to open a dialogue and collaboration between national researchers, papers in Slovenian are also accepted. All the papers were reviewed and revised according to the reviewers' comments. The quality of this year's papers is high, and we hope for an even higher one next year when the conference will celebrate its 20th anniversary.

Matjaž Gams, Mitja Luštrek, Rok Piltaver

PROGRAMSKI ODBOR / PROGRAMME COMMITTEE

Mitja Luštrek, IJS (co-chair)

Matjaž Gams, IJS (co-chair)

Rok Piltaver, IJS (co-chair)

Marko Bohanec, IJS

Tomaž Banovec, Statistični zavod

Cene Bavec, UP, FM

Jaro Berce, UL, FDV

Marko Bonač, ARNES

Ivan Bratko, UL, FRI in IJS

Dušan Caf, FIŠ

Bojan Cestnik, Temida

Aleš Dobnikar, CVI

Bogdan Filipič, IJS

Nikola Guid, FERI

Borka Jerman Blažič, IJS

Tomaž Kalin, DANTE

Marjan Krisper, FRI

Marjan Mernik, FERI

Vladislav Rajkovič, FOV

Ivo Rozman, FERI

Nikolaj Schlamberger, Informatika

Tomaž Seljak, IZUM

Miha Smolnikar, IJS

Peter Stanovnik, IER

Damjan Strnad, FERI

Peter Tancig, RZ

Pavle Trdan, Lek

Iztok Valenčič, GZ

Vasja Vehovar, FDV

Martin Žnidaršič, IJS

Usage of SIGMO, a Decision Support System for the Assessment of Genetically Modified Organisms in Food and Feed

Biljana Mileva Boshkoska
Faculty of information studies in Novo mesto, Ljubljanska cesta 32, Novo mesto, and Jožef Stefan Institute Jamova 39, Ljubljana
biljana.mileva@ijs.si

Marko Bohanec
Jožef Stefan Institute Jamova 39, Ljubljana, and University of Nova Gorica
marko.bohanec@ijs.si

Theo W. Prins
RIKILT Wageningen University & Research 6700 AE Wageningen, Netherlands
theo.prins@wur.nl

Esther J. Kok
RIKILT Wageningen University & Research 6700 AE Wageningen, Netherlands
esther.kok@wur.nl

ABSTRACT

SIGMO is a web-based decision support system for assessing the probability of existence of a genetically modified organisms in food and feed products imported in the European Union. The system has been developed in the frame of the EU FP7 project Decathlon. In this paper, we describe SIGMO, provide examples of how to use it for training and evaluation purposes, and report the statistics of its actual use.

Keywords

decision support system, qualitative multi-criteria modeling, genetically modified organisms, SIGMO

1. INTRODUCTION

The import of genetically modified organisms (GMOs) in the European Union (EU) has begun in 1994, and since then EU has developed different monitoring mechanisms to assess the risk of entering any GMO in its market. The goal of the risk assessment is to ensure that imported food or feed is safe for human and animal health and the environment. Hence, the EU treats all genetically modified crops (GMO crops) as "a modified food". All GMO crops that are intended to be admitted to the EU, are extensively evaluated by the European Food Safety Authority (EFSA) [1], which reports its findings to the European Commission (EC). The EC then proposes whether to grant or refuse the authorisation for the GMO [2]. In the EU, currently (September 2016) 68 GMOs are authorised in cotton (10 GMOs), maize (36), oil seed rape (6), soybean (15), and sugar beet (1) [3]. Despite the large number of authorized GMOs, the import of non-authorized GMOs is still being detected in EU, in particular from the US, Canada, Argentina, China and Brazil [4]. Whenever a product that contains GMO is imported, this information needs to be provided on the product label. There is a threshold of 0.9% for the adventitious and technically unavoidable presence of authorized GMOs in non-GMO batches that do not require labelling [5, 6].

Regardless of the strict measures, it still happens that products containing GMOs without labeling are imported into the EU. These products can reach the market and then be further on inspected. In cases of finding unauthorized GMOs in products, these will be withdrawn from the market, leading to significant financial losses for the traders. To assist the traders and producers of complex products, whose ingredients may contain GMOs, a decision support system called SIGMO (System for Identification of GMOs) has been developed within the European Framework 7 project 'Decathlon' (http://www.decathlon-project.eu/). In this paper, we describe SIGMO, illustrate its application for training and evaluation through several examples, and report statistics of using the system since it has been made available on-line.

2. SIGMO

SIGMO [6] is a web-based decision support system (DSS) that has been designed to provide help to producers, traders and importers with the aims to:

- reduce the number of necessary GMO analyses;
- better cope with the complexity of GMO market without requiring its users to have extensive knowledge on the world-wide production;
- comply with EU GMO regulations in a cost-effective way.

SIGMO is composed of:

- a data base providing data about GMO crop species produced and approved in counties worldwide;
- a multi-attribute model for the assessment of GMO presence in food/feed products; and
- an on-line user interface available at www.decathlon.ijs.si/gmo/.

2.1 SIGMO database

The SIGMO database consists of tree tables that hold data on countries, the list of possible status types that a GMO-containing product may obtain regarding the GM presence, such as "High likelihood", "Medium likelihood", and "Low likelihood"; and a list of all currently listed GMOs species. These tree tables are connected with three relations. The first relation, which defines the area of a particular crop planted in a certain country, determines if a certain GMO/country pair belongs to a region of increased GMO production. The second relation holds information regarding each particular GMO event. For example, maize may have several gene modifications, each one having a different Event name. Such names would be "MON810" encoded as "MONØØ81Ø-6". Finally, the third one defines a many-to-many relation between Events and Countries, associated with a particular status type. At the current implementation, the database holds over 300 GMOs.

2.2 SIGMO multi-attribute decision model

The central part of SIGMO is a multi-attribute decision model that provides an assessment of the potential presence of GMOs in imported feed or food products. The model has been developed using the methodology DEX (Decision Expert) [7]. DEX belongs to the group of qualitative multi-criteria decision making methods. In DEX, a hierarchical structure containing qualitative attributes is built which represents a decomposition of the decision problem into smaller, less complex and possibly easier to solve sub-problems. There are two types of attributes in DEX: basic and aggregated. The former are the directly measurable attributes, which are used for describing the decision options and represent input data to the model. The latter are obtained by aggregating the basic and/or other aggregated attributes. They represent the evaluations of the options. The hierarchical structure represents the dependencies among attributes such that the higher-level attributes depend on their immediate descendants in the tree. This dependency is defined by a utility function by the expert.

Attribute	Description
GM_Presence	Assessment of GM presence in food/feed products
TraceabilityData	GM presence due to traceability data
Products	GM presence due to product characteristics
ProductGMPresence	Assessment of GM presence in the product
CropGMPresence	Assessment of GM presence related to the crop
GeoGMPresence	Assessment of GM presence related to the geographical origin of the product
EU	Does the product originate in an EU country?
GM_Region	Does the product originate in a region of large GMO production?
ProductComplexity	Product type
Countries	Likelihood of GM presence due to the properties of countries and regions of origin
NumberCountries	Number of countries involved in storage
CountryGMPresence	Likelihood of GM presence with respect to involved countries
CoexistenceMeasures	Are coexistence measures in place in countries?
Transportation	Likelihood of GM presence due to transportation
PrepackagedProduct	Is product prepackaged?
Logistics	Likelihood of GM presence due to logistics
LogComplexity	GM presence due to logistics complexity
NumberInteractions	Number of interactions in the supply chain
NumberCompanies	Number of companies involved in logistics
LogStorage	Likelihood of GM presence due to storage used
Harbour	Has the product been shipped through harbour(s)?
Silo	Has the product been stored in silos?
SystemsUsed	GM presence due to used traceability systems
TraceabilitySystemInPlace	Is a traceability system in place?
IP_GMO	Are IP systems for GMO being used?
IP_Other	Are other IP systems being used?
AnalCtrl_Systems	Are there systems used that include analytical control?
PrivateContracts	Are there any private contracts?
AnalyticalData	GM presence due to analytical data
AnalyticalResults	GM presence according to analytical results
AnalyticalResultsAvailable	Are analytical results available?
ApprovedGMOsIdentified	Approved GMOs identified
UnapprovedGMOsIdentified	Unapproved GMOs identified
Methods	Risk due to applied methods
ProcessingLevel	Processing level
AppropriateSampling	Have appropriate sampling methods been used?
AppropriateMethods	Have appropriate analysis methods been used?
Reliability	Reliability of applied methods
ReliabilityForApprovedGMO	Reliability of applied methods to detect approved GMOs
RelevantGMCropsIncluded	Did the analysis include all the relevant GMO crops?
AllIngredientsIncluded	Are all ingredients, listed on the product label, included in the analysis?
OmnipresentGMOsIncluded	Are omnipresent GMO varieties included in the analysis?
NumberScreenElem	Number of screening elements
ReliabilityForUnapprovedGMO	Reliability of applied methods to detect unapproved GMOs
NumberScreenElem	Number of screening elements
AppropriateDataAnalysis	Application of appropriate data analysis methods
AppliedQualitySystem	Can we trust the applied methods and analytical lab?
ValidatedMethods	Application of validated analytical methods
AccreditedLab	Is analytical laboratory acredited?

Figure 1. Hierarchical structure of the SIGMO multi-attribute model

The SIGMO model, as given in Figure 1, is implemented in DEXi software package [8]. To assess the likelihood of GMO presence in a product, the model is at the highest level decomposed into two main subtrees:

- *TraceabilityData* – data that accompanies each product and describes its geographical origin and path in the production/supply chain [9], and
- *AnayticalData* – provide information about the approved and/or unapproved GMOs already detected in the product, and about the methods used to analyze the products for GMO content [10, 11].

Usually analytical data are rarely available with a product, but if they are available, they are potentially more relevant for assessing the possible presence of (unauthorized) GMOs than traceability data and should thus take precedence.

A more detailed description of attributes and utility functions of the model are given in [6].

2.3 SIGMO user interface

When using SIGMO (http://decathlon.ijs.si/gmo/), the user is presented with an interactive input form, such as the one presented in **Figure 2**, in which the user enters data about the product to be assessed. The system provides guidance through drop-down data-entry lists and info buttons on the right hand side of each data-entry field.

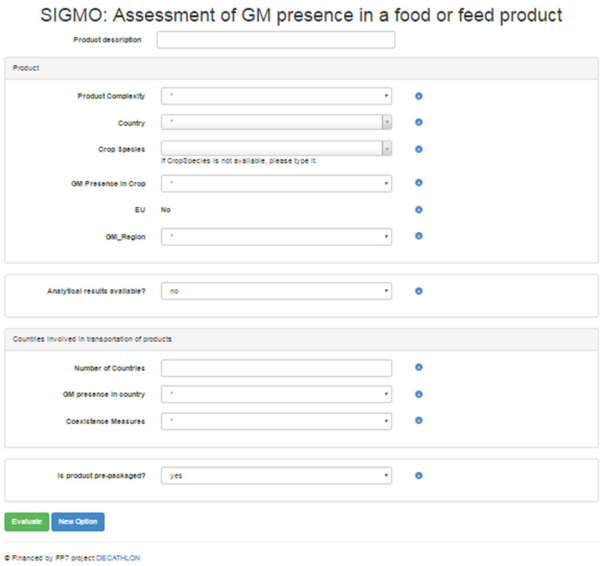


Figure 2. SIGMO user interface

Additionally, some parts of the input form are optional and open up only when necessary. Such examples are:

- only when *Product Complexity* is "complex", more than one ingredient can be entered;
- when *Analytical results available* is "yes", further detail about analytical results are requested;

- for products that are not pre-packaged, further data for transportation is requested;

After the data has been entered, pressing the button "Evaluate" shows the output page that displays the results of assessment. These results can be further explored by "drilling down" the evaluation tree.

SIGMO also provides a print page, which is suitable for saving and printing the evaluation results, and a help page describing the meaning of input data fields.

The system's architecture was designed by employing concepts that will allow seamless scalability. In that manner, the system is depends on MongoBD and MySql. The application is Django web application that runs on Apache through mod_wgsi. As such the application can be easily scaled in order to accommodate substantial traffic increase.

3. USAGE OF THE SIGMO DSS

Here we present several examples of how to use the SIGMO DSS in order to answer important GMO related questions about imported feed and food products.

3.1 Example 1

In the first example we answer the following question: given the origin of a certain product that consists of only one ingredient (for example popcorn), can one find all crops that may contain genetic modifications? To answer this question we consider a product that is made of crops, and that originates from Indonesia. Running SIGMO and checking the *Crop Species* field, we get the list of all crops that are produced in Indonesia and which potentially may contain unauthorized gene modifications in EU. The list contains *Maize, Soybean* and *Sugarcane*.

3.2 Example 2

Another relevant question when importing products from outside the EU is whether the country of origin has large fields that are populated with certain GMOs thus providing higher possibility for importing them in EU. For example, using the SIGMO model, one may be interested whether or not *Cream Style Corn* canned product (see **Error! Reference source not found.**) that contains *maize* and is produced in *Thailand,* is produced in a region of large GMO production. By selecting the appropriate values for *Country* and *Crop Species*, SIGMO provides the result *no*.

Figure 3. Cream Style Corn product from Thailand. Available from https://www.alibaba.com/product-detail/Cream-Style-Corn_131818000.html

3.3 Example 3

In this example we consider products that are made of several ingredients, each of which may have different probability of GMO presence. For example, one may be tempted to buy pasta made form *maize* and *rice* that are produced in Philippines. To assess the probability of presence of GMOs, we use SIGMO to evaluate both ingredients separately regarding the possibility of GMOs. As a final result, SIGMO aggregates the outputs of the evaluations into one recommended answer. For the provided example, SIGMO evaluates the possibility of unauthorized genes in the selected products as high. However, SIGMO also considers other attributes. Considering the fact that the product is packed, the possibility of commingling with other *maize* or *rice* is none. Therefore, the system evaluates the risk of GMOs in the product as *medium* to *low*.

3.4 Example 4

Consider that one wants to find out the GMO presence in complex products consisting from *rice* and *papaya* produced in China. The product is *packed*, and comes with a documentation from which the user may see that the product provides *Analytical results* from a laboratory. The results show that no *Approved* or *Unapproved GMOs* were identified by the laboratory. Hence a new drop-down menu is shown that asks questions regarding the laboratory results. The user answers as follows: "yes" for *GM presence in country* , "moderate" for *Processing Level*, "yes" for *Appropriate Sampling*, "yes" for *All Ingredients Included*, "yes" for *Omnipresent GMOs Included*, "many" for *Number of Screening Elements*, "yes" for *Appropriate Data Analysis*, "yes" for *Validated Methods*, "yes" for Accredited Lab. The next question is *Is product pre-packed* to which the user says "no". Hence, the user is asked a new set of question about the product transportation. The answers are "few" for *Number of interactions*, "no" for *Harbour*, and "no" for *Silo*.

Given these data, SIGMO provides information on *a-priori* GM presence in the product which is "high" for rice and "med" for papaya. Still, SIGMO's final assessment of GMO likelihood is *v_low*. The rationale is that the product has been inspected by a laboratory that did not find any authorized or unauthorized GMOs, and all additional questions regarding the laboratory indicate that the product has been properly inspected. Although the product has not been packaged, it also has neither been stored in silos nor had other interactions with other products at harbors. Thus, the final recommendation from SIGMO is that there is a very low probability that the product contains GMOs.

3.5 Usage of SIGMO

So far, SIGMO has been presented twice during the Decathlon project to project participants, traders, producers and importers: once in Lisbon, Portugal in 2015, and once in Shanghai, China in 2016. To measure the impact and usage of the system we have anonymously collected the country of accession of the SIGMO's web page. A histogram of different accesses to SIGMO in the

period of 15.1. – 15.7.2016 is given in Figure 4. From the histogram we can see that although the system has been presented only to participants from involved project countries (http://www.decathlon-project.eu/article/consortium), SIGMO has been used worldwide. Additionally, we present the monthly distributions of web-accesses of SIGMO in the period from December 2015 to September 2016 in Figure 5. The histogram shows increases of web-access from less than 100 in December, 2015 to more than 800 in July, 2016.

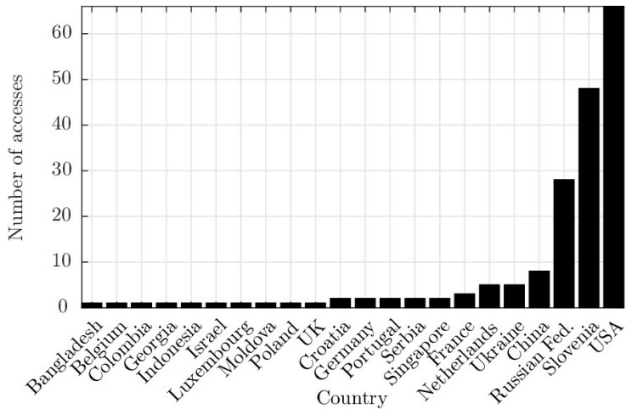


Figure 4. Different accesses points to SIGMO in the period of January, 2015 to July, 2016

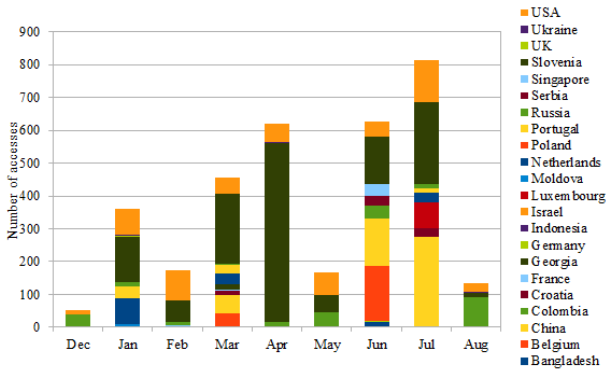


Figure 5. Distribution of SIGMO web-site visits per month in the period from December, 2015 to August 2016

4. CONCLUSION

The main goal of SIGMO is to help traders, producers and importers to assess the probability of existence of (un)authorized GMOs in the feed and food products. Here we presented four examples of how to use the system to achieve its goal. In addition we presented the number of accesses to the system in a six month period. Although the system has been presented only in Portugal and in China, it has been accessed from 22 countries. The presented examples can be used in future also for demonstration purposes to students studying for example decision sciences, in particular the DEX methodology.

5. ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement FP7-KBBE-2013-7-613908-Decathlon (http://www.decathlon-project.eu).

6. REFERENCES

[1] European Food Safety Authority. 2016. http://www.efsa.europa.eu/.

[2] European Commission Fact Sheet. 2015. Questions and Answers on EU's policies on GMOs Press release Database," http://europa.eu/rapid/press-release_MEMO-15-4778_en.htm.

[3] European Commission. 2016. Genetically Modified Organisams. EU register of authorised GMOs. http://ec.europa.eu/food/dyna/gm_register/index_en.cfm.

[4] European Commission. 2016. RASFF - Food and Feed Safety Alerts. http://ec.europa.eu/food/safety/rasff/index_en.htm.

[5] European Commission. 2003. Regulation (EC) No 1829/2003 of the European parliament and of the council of 22 September 2003 on genetically modified food and feed. *Official Journal of the European Union, L268,* 1 - 23.

[6] M. Bohanec, B. M. Boshkoska, T. W. Prins and E. J. Kok. 2016. SIGMO: A decision support System for Identification of genetically modified food or feed products. *Food Control,* 168–177.

[7] M. Bohanec, V. Rajkovič, I. Bratko, B. Zupan and M. Žnidaršič. 2013. DEX methodology: Three decades of qualitative multi-attribute modelling. *Informatica,* 49 - 54.

[8] M. Bohanec. 2015. DEXi: Program for multi-attribute decision making, User's manual, version 5.00. Ljubljana: Jožef Stefan Institute. IJS Report DP-11897. Ljubljana.

[9] European Commission. 2013. Regulation No. 1830/2003 concerning the traceability and labelling of genetically modified organisms and the traceability of food and feed products produced from genetically modified organisms and amending Directive 2001/18/EC.

[10] A. Holst Jensen, 2009. Testing for genetically modified organisms (GMOs): Past,present and future perspectives. *Biotechnology Advances,* 27, 71 - 1082.

[11] A. J.Arulandhu, J. P. V. Dijk, D. Dobnik, A. Holst-Jensen, J. Shi and J. Zel. 2016. Critical review: DNA enrichment approaches to identify unauthorised genetically modified organisms (GMOs). *Analytical and Bioanalytical Chemistry*.

Parallel Draws from the Polya-Gamma Distribution for Faster Bayesian Multinomial and Count Model Inference

Rok Češnovar
University of Ljubljana
Faculty of Computer and Information Science
Večna pot 113
Ljubljana, Slovenia
rok.cesnovar@fri.uni-lj.si

Erik Štrumbelj
University of Ljubljana
Faculty of Computer and Information Science
Večna pot 113
Ljubljana, Slovenia
erik.strumbelj@fri.uni-lj.si

ABSTRACT

We propose an algorithm for drawing random variates from the two-parameter Polya-Gamma distribution PG(h, z) in parallel on GPU, when h is integer. This distribution plays an important role in Gibbs-sampling schemes for inference from Bayesian categorical and count models. We demonstrate speedups of the order of 100 on mid-range and of the order of 10 on low-end GPU and how these speedups translate to similar speedups when the sampler is used in Gibbs sampling for binomial regression.

Keywords

OpenCL, random variables, sampling, Bayesian inference

1. INTRODUCTION

Bayesian methods play an important role in modern statistics and machine learning due to their robustness and conceptual simplicity. The main drawback of Bayesian methods is that it is often analytically and computationally difficult to infer from the posterior distribution. Instead, we typically resort to structural approximation or, when we are interested in the full posterior density, computationally-intensive sampling-based approximation, in particular, Markov Chain Monte Carlo (MCMC).

For example, take the binomial regression setting with n samples and k input variables. Let $y_i \in \{0, 1\}$ be the dependent variable, $x_i \in R^k$ the input variables, and n_i the number of attempts for the $i-$th sample, $i = 1..n$. Even for a standard and relatively simple model, such as the Bayesian binomial regression model

$$\begin{aligned} y_i &\sim \text{Binomial}\left(n_i, \frac{1}{1+e^{-\beta^T x_i}}\right) \\ \beta &\sim \text{N}_k(\mu_0, \Sigma_0) \end{aligned} \tag{1}$$

neither the joint posterior nor the full conditional posterior distributions can be derived analytically. Note that Bayesian logistic regression is a special case of Eq. (1) with all $n_i = 1$. Typically, Laplace approximation [1, p. 213] or the Metropolis algorithm [1, p. 541] are used to infer from this model. Note that β are the k model coefficients, and μ_0, Σ_0 the prior mean vector and prior covariance matrix, respectively.

Recently, Polson et al. [9] proposed a data-augmentation scheme that leads to a simple Gibbs sampler for the posterior of the Bayesian Binomial regression model Eq. (1)

$$\begin{aligned} w_i|\beta &\sim \text{PG}(n_i, \beta^T x_i) \\ \beta|w &\sim \text{N}_k(\mu_w, \Sigma_w) \end{aligned} \tag{2}$$

where $\mu_w = \Sigma_w(X^T\kappa + \Sigma_0^{-1}\mu_0)$, $\Sigma_w = (X^T W X + \Sigma_0^{-1}\mu_0)^{-1}$, $\kappa = y - \frac{n_i}{2}$, W is the diagonal matrix of w, and PG(h, z) is the Polya-Gamma distribution, which is the focus of this paper and will be discussed in more detail in Section 2.

This data-augmentation scheme in Eq. (2) is more efficient than Metropolis samplers and other data-approaches (see [9] for details) and, which is even more important, easily extends to other categorical and count models, such as Multinomial Logistic regression and Negative Binomial regression.

However, the Polya-Gamma distribution is not easy to sample from and can be a performance bottleneck, in particular when h is large. In this paper we propose an algorithm for sampling from PG(h, z) in parallel on GPU, when h is integer, which is sufficiently general for categorical and count models. We demonstrate its utility as a stand-alone random variable generator and when used as part of a Gibbs sampler for Bayesian inference.

2. PG DISTRIBUTION AND SAMPLING

In this section we briefly review the properties of the PG distribution relevant to our work. The Polya-Gamma distribution PG(h, z) is a two-parameter continuous distribution ($h > 0$, $z \in R$). The first important property of the PG distribution is that if $X \sim \text{PG}(h_x, z)$ and $Y \sim \text{PG}(h_y, z)$ then $X + Y$ is distributed PG($h_x + h_y, z$). Therefore, for integer h, a random variate from PG(h, z) can be generated using h random variates from PG($1, z$).

The above property allows us to focus on sampling from PG($1, z$). Our parallel algorithm is based on the algorithm from [9], which is based on sampling from the (exponentially tilted) Jacobi distribution J*(1,z) [4] and the relationship

that if $X \sim$ J*(1,z/2) then $Y = X/4 \sim \text{PG}(1, z)$. Therefore, our sampling problem reduces to sampling from J*(1,z).

The density $f(x)$ of a J*(1,z) distributed random variable contains an infinite sum and can only be approximated (the same holds for the density of PG(1,z), of course), which makes sampling difficult. However, it has an alternating series representation

$$f(x) = \sum_{i=0}^{\infty} (-1)^i cosh(z) \exp\left\{-\frac{z^2 x}{2}\right\} a_i(x),$$

$$a_i(x) = \begin{cases} \pi(i+1/2)(\frac{2}{\pi x})^{3/2} \exp\left\{-\frac{2(i+1/2)^2}{x}\right\} & 0 < x < t \\ \pi(i+1/2) \exp\left\{-\frac{2(i+1/2)^2\pi^2}{2} x\right\} & x > t, \end{cases}$$

where the optimal choice of $t = 0.64$ [4].

The partial sums of the series $S_i(x) = \sum_{j=0}^{i} (-1)^j a_j(x)$ satisfy

$$S_0(x) > S_2(x) > \cdots > f(x) > \cdots S_3(x) > S_1(x),$$

which facilitates a modified rejection algorithm for sampling from $f(x)$. The problem that $f(x)$ can not be evaluated is circumvented by iteratively calculating the partial sums and stopping early using the above property of $S_i(x)$. Given an upper hull (envelope) function $g(x) > f(x)$, we can sample from $f(x)$ by sampling from $g(x)$ and accepting each samples x_0 with probability $f(x_0)/q(x_0)$, which can be done by sampling $U \sim U(0, g(x_0))$ and checking $U < f(x_0)$. Due to the property of the partial sums, we know that all odd terms are less than $f(x)$ - therefore, if $U < S_i(x_0)$ for odd i, we can deduce $U < f(x)$ and accept the sample. Similarly, if $U > S_i(x_0)$ for even i, we can deduce $U > f(x)$ and reject the sample.

The first partial sum $S_0(x)$ is a natural (and efficient [9]) choice for the upper hull function $g(x)$ and it can be sampled from a mixture of a truncated inverse-Gaussian distribution and a truncated exponential distribution

$$X \sim \begin{cases} \text{IG}(|z|^{-1}, 1)\mathbb{I}_{(0,t]} & \text{with prob. } p \text{ see [9]} \\ \text{Exp}(-z^2/2 + \pi^2/8)\mathbb{I}_{(t,\infty]} & \text{with prob.} 1 - p. \end{cases}$$

Left-truncated exponential variates can be generated trivially by translating and scaling a Exp(1) variate. Inverse-Gaussian variates are sampled using the method of [5], which is based on Exp(1) and standard normal $\text{N}(0, 1)$ variates. For further details on the above algorithm for sampling from J*(1,z) and therefore $\text{PG}(1, z)$ see [9]. Note that other, faster approximate approaches for sampling from PG have been developed [11], but are unstable for larger h.

3. PARALLEL SAMPLING ON GPU

The basic case of sampling from the distribution is drawing r samples from $\text{PG}(h, z)$. In this paper we focus on a more generalized problem of performing l simultaneous samplings, each with different r_i, h_i, and z_i. The total number of samples to be drawn is $R = \Sigma_{i=1}^{l} r_i$.

When parallelizing, the first step is to split the problem into smaller tasks that require minimal amount of communication. Each task is then assigned to a single thread. When parallelizing for the GPU, we are dealing with the trade-off between the demands for a very large number of threads and the demand that the threads have a large enough workload so that the cost of creating threads does not outweigh the benefits of parallel execution. We also want to have similar workloads in the threads, because uneven workloads are huge performance drawbacks in GPU programming.

The naive approach would be to split the problem, so that each sampling i from $\text{PG}(h_i, z_i)$ is a task. This way each task is independent, therefore requiring no communication. However, we could have the problem of uneven workloads, since threads that would perform draws with large h_i would have a substantially higher workload than threads with small h_i. Furthermore, with small R, the low number of threads would not fully utilize GPUs performance.

To deal with these drawbacks we define a task to be a single draw from $\text{PG}(1, z_i)$. The final draws $\text{PG}(h_i, z_i)$ are calculated on the CPU by summing h_i samples from $\text{PG}(1, z_i)$, as explained in Section 2. This way tasks are more homogeneous and the workload of threads more even, as the computational complexity of $\text{PG}(1, z_i)$ is practically independent of z_i. The number of tasks in this parallelization is $R_{\text{task}} = \Sigma_{i=1}^{l} r_i h_i$. Note that $R_{\text{task}} > R$ if there are $h_i > 1$, so we are able to better utilize the GPU when R is small.

The homogeneity of tasks also gives us the ability to easily group them in larger but still homogeneous workloads. Each thread thus performs batches of B draws from $\text{PG}(1, z_i)$ instead of single draws. This ability is useful when searching for the optimal point in the aforementioned trade-off. For example, when R_{task} is much greater than the number of execution units C on the GPU. If each thread would execute a single draw, the benefits of parallelization would be small as R_{task}-C threads would be assigned to the queue. The cost of creating tasks would therefore outweigh these benefits. In cases when $C > R_{\text{task}}$, we could define $B = 1$, thus utilizing all the execution units. With the ability to easily fine-tune B we can get the best performance from GPUs of different vendors and architectures. Note that parallelization within the execution of single draws from $PG(1, z_i)$ is not feasible as these parallel tasks would require a lot of communication, outweighing the benefits of parallel execution.

3.1 Pseudo-Random number generation and sampling from probability distributions

The algorithm for drawing from $\text{PG}(1, z)$ requires random variates from distributions $U(0, 1)$, $N(0, 1)$, and $Exp(1)$. We implemented the XORShift128 [7] uniform random number generator, as this generator is one of the fastest on the GPU and has satisfactory statistical properties [3, 6]. Unit uniform variates $U(0, 1)$ are generated trivially, by dividing the generated number with the maximum possible uniform random number. Standard normal variates $N(0, 1)$ are generated using $U(0, 1)$ variates and the Box-Mueller transform [2]. Exponential variates $Exp(1)$ are generated using the $U(0, 1)$ variates and the inverse transform method.

3.2 Data transfers

The input data required for sampling from $PG(1, z)$ are: the array of all R_{task} z values and the seeds for the random

number generator in each thread. Values z are floating-point numbers while random seeds consist of 4 unsigned integer values. The output data of the calculation on the GPU are the R_{task} real-valued samples from $PG(1, z)$. The input data size is therefore $F \times R_{\text{task}} + 4 \times 4 \times \left\lceil \frac{R_{\text{task}}}{B} \right\rceil$ bytes, while the output data size is $F \times R_{\text{task}}$ bytes, where F is 8 for double precision or 4 for single precision. If the input and output data size is larger than the size of global memory, execution is split in smaller steps which has a negative effect on overall speedup. For a mid-range notebook GPU with 4GB of global memory, the execution has to be split for $R_{\text{task}} > 2^{27}$ in the worst case of $B = 1$ and $F = 8$.

3.3 Thread organization

Both modern and older GPU consist of a large number of small and simple execution units, grouped together into a smaller number of groups. These groups of units are termed Streaming Multiprocessors (SM) in Nvidia and Computation Units (CU) in AMD/ATI graphics processing units. Execution units in a single CU/SM share more advanced logic for scheduling, caches, registers, etc. How threads get assigned to these executions units, SM or CU depends on the thread organization. GPU threads are organized in workgroups or blocks of the same size. Each block is assigned to a single SM/CU and the threads inside the block are executed in the SM/CU execution units.

Threads in the same block therefore share the registers and caches. Local values for threads are stored in registers. If all the registers are used, the local data for non-active threads is cached or even stored in global memory, depending on the architecture and the amount of data used. If we want to achieve optimal performance we want to avoid global memory access. Beside memory access, parallelization on the execution unit level and SM/CU level is the other key factor when determining thread block size. If we organize threads in a large number of small blocks we can underutilize the execution units in a single SM/CU. On the other hand if we organize threads in fewer large blocks, some SM/CU could be underutilized while threads in other SM/CU would queue.

In our case, we can not define a constant thread-block size that would be optimal or even near-optimal for all possible values R_{task}, B and all architectures. For the experiments done in Section 4, we fine tuned the thread-block size together with the batch size B empirically. We reduced the search space for these two parameters based on detailed knowledge of the architectures used.

3.4 Implementation

We implemented the algorithm as part of a R package. Input data checking and output data allocation was implemented in R, while the rest of the algorithm was implemented in C++ and OpenCL 1.2. OpenCL was used over CUDA for the purpose of being GPU-vendor agnostic. OpenCL 1.2 was used over 2.x in order to be compatible with more GPUs.

4. EMPIRICAL VALIDATION

We performed two experiments. The first experiment is a direct comparison of our parallel GPU algorithm and existing CPU implementations for sampling from the PG distribution from the `BayesLogit` package [9]. The second experiment is an indirect comparison of the sampling algorithms through the use in Bayesian inference.

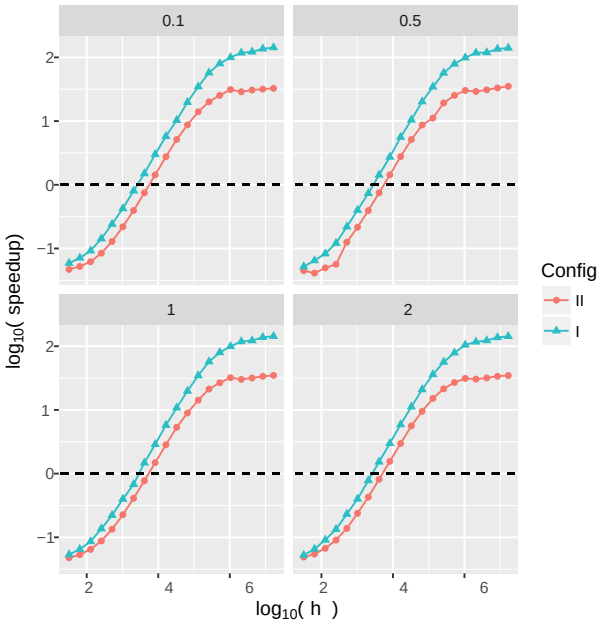


Figure 1: A plot of speedups (BayesLogit algorithm time divided by GPU algorithm time) against size of h. A log-log scale is used to simplify interpretation. Dashed line indicate for which n both algorithms take the same amount of time.

In the following two subsections, we describe the experiments and report results on two different hardware configurations **I** (CPU: Intel Core i7-4790 @ 3.60GHz, GPU: AMD Radeon HD 7970, RAM: 8GB DDR3 @ 1333 MHz) and **II** (CPU: Intel Core i7-5600U @ 2.60GHz, GPU: Intel HD Graphics 5500, RAM: 12 GB DDR-3 @ 1333MHz). Note that thread-block (64) and batch (20) sizes were tuned for best (on average) performance on hardware configuration **I** and used for both hardware configurations. All measurements were made using the `microbenchmark` package [8], taking the median value of 1000 repetitions.

4.1 Generating PG random variates

We measured the running times of generating a single PG random variate ($r = 1$) for increasing $h = 2, 4, 8, \ldots$ across different $z = \{0.1, 0.5, 1.0, 2.0\}$. The number of samples r is set to 1. Varying both r and h is redundant, as increasing either one increases the key performance variable R_{task}.

Results are summarized in Figure 1 and show that for large-enough h speedups of over 100 and approximately 30 can be achieved on configurations I and II, respectively. Results also show that the value of z does not have a substantial influence on running times.

For small h, running times of the GPU algorithm can be up to 10 times slower than the BayesLogit implementation and R_{task} of over 10000 is required to achieve at least some speedup. This is due to the overhead of building and trans-

Table 1: Estimated running times and speed-ups for a single iteration of Gibbs sampling on the NBAplayers data set. Results are for each hardware configuration (I and II) separately and broken down into three parts: sampling from Polya-Gamma, sampling from multivariate normal, and matrix operations. Standard deviations are included, because times vary from iteration to iteration.

	Part	BayesLogit [µs]	GPU [µs]	Speedup
I	PG	230423 ± 4745	5217 ± 120	44 ± 1
	N_k	50 ± 2	50 ± 1	-
	matrix	3212 ± 133	3107 ± 83	-
	total	233685 ± 4790	8435 ± 160	27 ± 0
II	PG	127482 ± 2525	6793 ± 112	18 ± 0
	N_k	53 ± 1	54 ± 1	-
	matrix	3969 ± 9	3996 ± 42	-
	total	131504 ± 2522	10844 ± 101	12 ± 0

ferring the OpenCL kernel and the transfers of input and output data. The cost of the former is larger but is also one-time and thus negligible for large h.

4.2 Binomial regression

We illustrate the benefits of GPU-accelerated draws from PG to Bayesian inference. We use the binomial regression model from Eq. (2) to model the relationship between basketball players' features and their career three-point shooting percentages. The model was implemented in R, using core R matrix operations and `MASS` package for generating multivariate normal variates [10]. Zero μ_0 and unit diagonal Σ_0 were used as prior values.

The data set contains 744 NBA basketball players and, for each player, his total three points attempted and made, weight, height, minutes per game, and indicator variables whether the player was a center or a forward. Therefore, a total of 744 samples and 6 input features, including the constant term (intercept). The average number of three-point shots attempted (n_i in the model, h_i in sampling) is 322.6 (ranging from 1 to 4270), leading to $R_{\text{task}} = 239988$.

Results are summarized in Table 1. Between-iteration variability is relatively small, which is consistent with values of z not having a strong influence on running times (the $h_i = n_i$ are fixed for all iterations). The speedup for draws from PG (44 and 18 for configurations I and II, respectively) is also consistent with results from the first experiment for $R_{\text{task}} = 239988$. The example also illustrates how most of the speedup in PG draws translates to total speedup, because the PG part represents the majority of the total time. This is true when n and number of input features k are small relative to R_{task}, which is common in the practice of binomial, multinomial, and count regressions.

5. CONCLUSIONS

In this paper we have parallelized the process of generating Polya-Gamma random variates on GPU. Speedup were of the order of 100 on a mid-range non-specialized GPU, which already makes it very useful both as a standalone sampler and as part of a Gibbs sampling scheme for Bayesian multinomial and count models. Higher speedups can be expected on high-end specialized GPU. Furthermore, the measurements in our paper regarding the size of h required to achieve speedup and maximum speedup are pessimistic, because they include, for each sampling, the overhead for transferring data and code to the GPU, which would only have to be done once.

Additional speedup could be achieved by parallelizing the matrix operations and generating multivariate random variates, for which efficient GPU implementations exist. Combining these into fully GPU-parallelized multinomial and count models is delegated to future work. Another direction for future work is automatically tuning the two parallelization parameters (thread-block size, batch size) to the architecture of the GPU and R_{task}. This can be done either offline, learning from performance on known configurations, or online, gradually adapting the parameters based on performance, which is particularly suited to Gibbs sampling, as several iterations are made with the same values of h_i.

6. ACKNOWLEDGMENTS

This work was supported by the Slovenian Research Agency (ARRS Applied Project L1-7542).

7. REFERENCES

[1] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, 2006.

[2] G. E. P. Box and M. E. Muller. A note on the generation of random normal deviates. *Ann. Math. Statist.*, 29(2):610–611, 06 1958.

[3] V. Demchik and N. Kolomoyets. QCDGPU: Open-Source Package for Multi-GPU Monte Carlo Lattice Simulations. *Computer Science*, 1(1):13–21, 2014.

[4] L. Devroye. On exact simulation algorithms for some distributions related to Jacobi theta functions. *Statistics & Probability Letters*, 79(21):2251–2259, 2009.

[5] D. Luc. Non-uniform random variate generation. *NY: Springer*, 1986.

[6] M. Manssen, M. Weigel, and A. K. Hartmann. Random number generators for massively parallel simulations on GPU. *The European Physical Journal Special Topics*, 210(1):53–71, 2012.

[7] G. Marsaglia. Xorshift RNGs. *Journal of Statistical Software*, 8(1):1–6, 2003.

[8] O. Mersmann. *microbenchmark: Accurate Timing Functions*, 2015. R package version 1.4-2.1.

[9] N. G. Polson, J. G. Scott, and J. Windle. Bayesian inference for logistic models using Pólya–Gamma latent variables. *Journal of the American statistical Association*, 108(504):1339–1349, 2013.

[10] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, New York, fourth edition, 2002. ISBN 0-387-95457-0.

[11] J. Windle, N. G. Polson, and J. G. Scott. Sampling polya-gamma random variates: alternate and approximate techniques. *arXiv preprint arXiv:1405.0506*, 2014.

Monitoring and Management of Physical, Mental and Environmental Stress at Work

Božidara Cvetković, Martin Gjoreski, Martin Frešer
Jožef Stefan Institute
Department of Intelligent Systems
Jamova cesta 39, Ljubljana, Slovenia
boza.cvetkovic@ijs.si

Michał Kosiedowski
Poznań Supercomputing and Networking Center
ul. Jana Pawla II 10, Poznań, Poland
kat@man.poznan.pl

Mitja Luštek
Jožef Stefan Institute
Department of Intelligent Systems
Jamova cesta 39, Ljubljana, Slovenia
mitja.lustrek@ijs.si

ABSTRACT
In this paper, we present the design of the Fit4Work system for monitoring and management of physical, mental and environmental stress at work. In addition, preliminary results of the monitoring modules are presented. The goal was to design a low-cost system that can use commercial sensors to monitor several aspects of the users' lifestyle and to provide recommendations for improving it. The system is designed for older workers who are subject to sedentary stressful work in an office environment. The preliminary results show that the system can sufficiently detect sedentary, stressful and unhealthy environment and provide information in the form of recommendation to the user.

Categories and Subject Descriptors
D.3.3 [**Human-centered computing**]: Ubiquitous and mobile computing – *ambient intelligence, mobile computing, ubiquitous computing*

Keywords
Physical activity monitoring, mental stress detection, environment quality management, wearable sensors, ambient sensors

1. INTRODUCTION
The pace of life has been increasing with each day in the developed world, resulting in a lot of health risk factors. Most of the active population spends large amount of time at work in closed environments under productivity pressure, which contributes to different aspects of stress, such as unhealthy amount of sedentary activity, high mental stress and in many cases low quality of the environment. If such lifestyle is maintained over longer period of time and even becomes a regular lifestyle of a person it can severely contribute to development of other behavioral changes and chronical or even deadly diseases.

Physical stress or physical inactivity is the fourth leading cause of death worldwide. It contributes to increased mental stress, development of cardiovascular diseases, diabetes, obesity and other unhealthy behavioral changes. Exposure to *mental stress* at the workplace is not necessarily a negative thing since right amount of stress is needed for better productivity, however, prolonged exposition to stress can result in chronical stress which is a trigger for slower body recovery, other mental illnesses and vulnerability to infections due to decreased immune system. *Environmental stress* is related to overstaying in the environment with low quality conditions. The term Sick Building Syndrome (SBS) is used for buildings that cause the occupants various health and comfort problems, and is most commonly linked with air quality. It has been shown that in addition to health related problems, the bad environmental conditions can cause decreased productivity. Inappropriate temperature decreases the productivity by 10%, humidity by 5% and air pollutants by 6-9%, which can result in a higher level of mental stress due to productivity pressure [1][2][3]. The objective of the Fit4Work system [4] is to detect sedentary, stressful and unhealthy environment and provide information in the form of recommendation to the user.

The smartphone industry is constantly developing and with each new product the difference between the computing power of a computer and a smartphone is being blurred. Today's average smartphone is equipped with sensors for motion monitoring (e.g., accelerometers, gyroscopes, GPS, etc.), ambiance monitoring (e.g., microphones, light, etc.) and with each next generation this list is extended. If we take into account that by the end of 2017, over a third of the world population is projected to own a smartphone (2.6 billion users) it makes it the most appropriate device for mobile sensing and computing. Furthermore, with increased availability of commercial sensors and even dedicated devices that can connect with a smartphone and sense environmental parameters or physiological parameters of a person, the possibility of applications has become almost infinite.

Monitoring of persons' physical activity is not new and is in fact very popular in terms of number of smartphone applications, dedicated devices and even smartwatch applications already available on the market. Smartphone only applications for activity monitoring use smartphone accelerometer to estimate the burned calories and amount of movement based on the number of steps the user takes over a day. Dedicated devices such as Microsoft Band 2 [6] use simplified activity recognition and machine-learning to improve the estimated calorie burn. Apple Watch [7] goes one step further and contains hourly reminders if the person is sedentary and provides feedback and motivation to reach the daily goal. However, if the user wants to be monitored accurately it is required to input the current activity which is being performed. The applications and devices available on the market are satisfactory for most of the active users who want to track their activities such as running or just to get an insight into their lifestyle. If the application/device also provides a feedback and a motivation the price gets too high, thus being too expensive for an average person.

Monitoring mental stress using commercial and unobtrusive devices is relatively new and challenging research topic. Healey and Picard [8] were first to show that the stress can be detected using physiological sensors which required intrusive wires and electrodes. Hovsepian et al. [9] proposed cStress which

continuously monitors stress level using an ECG sensor, and were looking forward to use smartwatches in the future. With the advancement of the technological devices equipped with physiological sensors, such as Empatica [10] and Microsoft Band 2 [6], these obtrusive methods could finally be implemented for everyday use. Various studies exist, where researchers combined signal processing and machine-learning to implement stress detection. The studies were utilizing different sources of signals, obtrusive sensors such as camera, microphone or unobtrusive wearables [11] such as wrist device which is acceptable for most of the people.

The monitoring and control of indoor environment parameters is a popular topic in smart–building research where they mainly focus on the trade-off between the occupants' comfort in terms of environment quality, and energy consumption [12]. In this research it is prerequisite to have an automated building and fully equipped rooms with HVAC systems which can be a substantial investment and mainly indicates that most of the people living in older buildings will not be able to use such system. To develop a system which does not need very expensive equipment one must develop several virtual sensors for detection/estimation of parameters that cannot be sensed directly and develop prediction models for environmental parameters. A correlation between the environmental values and the occupancy state or window state has been reported in previous research [13] which proves that it can be modeled. Estimation of the number of occupants was successfully achieved with using the data of single or multiple environmental parameters [14]. We did not come across any relevant methods for window state detection. The prediction of the dynamics of the parameters is usually done with the predictive control technique [15] which requires the user to describe the environment in detail (e.g., material of walls, windows, etc.) which is not very user friendly.

The Fit4Work system will utilize sensors from an accelerometer equipped smartphone, Microsoft Band 2 and a commercial weather station NetAtmo [16] to monitor and detect increased physical, mental and environmental stress at workplace and provide recommendations through the dedicated smartphone application which will assist in management of the stresses. To be able to successfully achieve this goal we designed a system which utilizes multiple machine-learning models to monitor physical activity, mental stress of the user and predict the state of the workplace and predict future dynamics of the environmental parameters.

The paper presents the design of the final Fit4Work system for monitoring and management of physical, mental and environmental stress, and refers to preliminary results of the modules which were obtained independently of other modules.

2. FIT4WORK SYSTEM OVERWIEW

The Fit4Work system for monitoring and management of physical, mental and environmental stress is composed of three data analysis and recommendation components and is presented in Figure 1. The physical activity monitoring utilizes data from the smartphone accelerometer and Microsoft Band 2 to recognize the current activity of the user and to estimate the energy expenditure while performing the recognized activity. The mental stress monitoring utilizes data from the Microsoft Band 2 and physical activity as recognized by the physical activities module to detect the level of mental stress. The quality of the environment module utilizes data from the commercial weather station NetAtmo to detect whether the quality of the environment has decreased. Each of the data analysis components has its own recommendation component which utilizes results from the data analysis to generate recommendation. The generated recommendations are sent back to the users' smartphone.

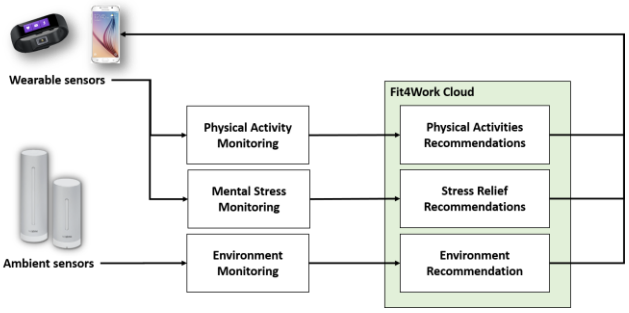


Figure 1. Fit4Work system for monitoring and management of physical, mental and environmental stress overview.

2.1 Physical Activities

The goal of the physical activity module is to recognize current state of the user is terms of activity and expended energy. Additionally, the output of the physical activity module is aggregated and used for generating recommendations which will guide the user to successfully achieve daily and weekly activity goals.

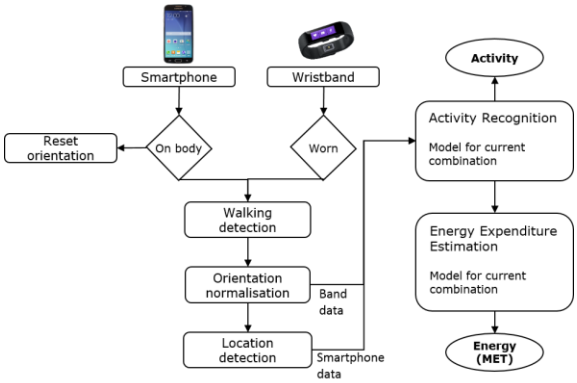


Figure 2. Workflow of the physical activity monitoring data analysis component

The physical activities method is composed of six tasks and is presented in **Figure 2**. The devices we use are the accelerometer equipped smartphone and the Microsoft Band 2 wristband capable of measuring acceleration and physiological signals such as heart rate, galvanic skin response, skin temperature and R-R interval. First, we check whether each of the two devices is present on the user's body. Second, if the smartphone or wristband is on the body, we wait for a walking period. Walking is detected with a machine-learning model in a location- and orientation-independent manner. The walking signal is a prerequisite for normalizing the orientation of the smartphone and the wristband and for recognizing the location of the smartphone. Third, when a 10-second period of walking is detected, we normalize the orientation and detect the location of the smartphone utilising location machine-learning model. If the smartphone ceases to be on the body because the user has taken it out of the pocket or bag, the information on the orientation and location is no longer valid

and we have to wait until the next period of walking. Fourth, depending on the devices which are on the body and the location in which the smartphone is, we invoke the appropriate classification model for activity recognition (AR) and regression model for estimating user's energy expenditure (EEE). For more information the reader is referred to [17].

In this paper we present preliminary results for AR (lying, walking, running, standing, sitting, cycling, mixed) and EEE with the wristband only if the wristband is present and smartphone only if wristband is not present. The results are presented in Table 1. The experiments were performed on the data obtained with Empatica in scenario presented in [17]. The results of AR is presented in terms of accuracy and the EEE in terms of mean absolute error (MAE). The results of the EEE are compared against the commercial device SenseWear [19], one of the most accurate consumer devices for EEE.

Table 1. Preliminary results of the activity recognition and estimation of energy expenditure.

		Smartphone			
	Wristband	Trousers	Torso	Bags	SenseWear
AR [%]	88.4	87.9	78.9	79.5	/
EEE [MET]	0.87	0.87	0.92	1.15	1.0

Future work includes use of both devices if both are present as done in our previous work [17].

2.2 Mental Stress

The stress monitoring module continuously monitors user's stress and provides overview of stressful events to the user. Additionally, the output of the stress-monitoring module is used by a stress-relief module for suggesting appropriate exercises at appropriate time.

For the stress-monitoring module we developed a machine learning method which is applied on data collected via sensor-equipped device worn on the wrist, Microsoft Band 2. The method for detection of mental stress is presented in Figure 3 and consists of three machine-learning components (a base stress detector, an activity recognizer and a context-base stress detector).

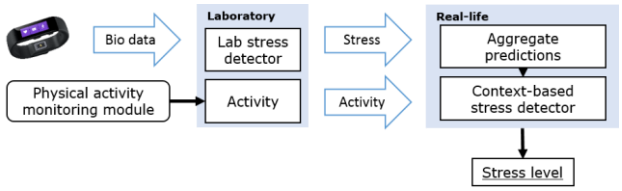


Figure 3. Method for mental stress detection.

The laboratory stress detector is a machine-learning classifier trained on laboratory data to distinguish stressful vs. non-stressful events in 4-minute data windows with a 2-minute overlap. As input it uses features computed from the physiological signals (blood volume pulse, heart rate, R-R intervals, skin temperature and electrodermal activity), and it outputs a prediction for possible stress event. In addition, the activity is retrieved from the physical activity monitoring module and provides a context information for the context-based stress detector. The context-based stress detector aggregates the predictions of the laboratory stress detector, uses context information and provides a prediction every 20 minutes. The interval of 20 minutes was chosen empirically. We decided to use the context-based classifier to distinguish between true stress and the many situations which induce a similar physiological arousal (e.g., exercise, eating, hot weather, etc.). By introducing the context-based classifier we can provide more information about the real-life circumstances and the user, thus to improve the detection performance.

The preliminary experiment of the module was done using the Empatica wristband and the results are presented in Table 1. For more details the reader is referred to [21].

Table 2. Confusion matrix for leave one user out evaluation of the mental stress detection module.

	No Stress	**Stress**
No Stress	790	23
Stress	51	63

Future work includes investigating the relation between labelled stress level, recognized stress level and cortisol levels and personalization of the models.

2.3 Quality of the Environment

Monitoring of the quality of the environment detects worsening of measured parameters and recommends appropriate actions which will return to or keep the parameters in optimal range. The module for monitoring and management of the quality of the environment is composed of three components - a sensing component, an ontology with a reasoner, and a simulator, as presented in Figure 4.

The sensing component is composed of hardware sensors and virtual sensors. The hardware sensors are the real sensors which measure the environmental parameters such as temperature, humidity, CO_2, noise, etc., while virtual sensors use machine – learning models on the raw parameter values to estimate room-specific properties that cannot be sensed directly such as estimating the number of occupants and detecting the state of the devices. The outputs of the sensing component are fed into the ontology, from which the reasoner infers actions that can improve the state of the environment, based on the current state and present devices. The list of actions returned by the ontology is fed into the simulator. The simulator is composed of prediction models and the quality rating module (Q–rating) which predicts the environmental values for all combination of actions and evaluates the resulting parameter states (values range between -1 and 1). The action resulting in the best state is finally recommended by the system. The reader is referred to [22] for more details.

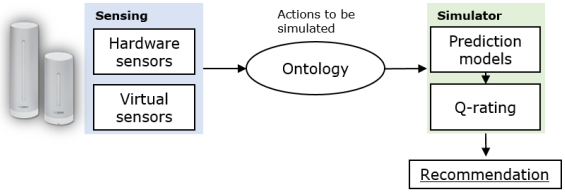


Figure 4. Method for monitoring and management of the quality of the environment.

The preliminary results on the winter data are shown in Table 3. First two rows present the results of the virtual sensors, one for detecting the window state and the second for estimating number of occupants, last three rows present the results of the prediction models. The result for the classification is presented in terms of

accuracy and results for the regression in terms of mean absolute error (MAE) and root mean squared error (RMSE).

The future work includes development of the virtual sensors for other devices, such as humidifier, air conditioning etc. and improvement of the prediction models by using additionally collected data.

Table 3. Results of the machine-learning models used for virtual sensors (window state and number of occupants) and for prediction of environmental parameters.

	ACC		MAE		RMSE	
	E	V	E	V	E	V
Window state [%]	91	81	✗	✗	✗	✗
No. of occupants	✗	✗	0.6	0.5	1.2	0.8
Predict T [℃]	✗	✗	0.4	0.5	0.5	0.6
Predict H [%]	✗	✗	0.6	0.3	0.9	0.5
Predict CO_2 [ppm]	✗	✗	55	45	104	61

3. CONCLUSION

We present a work-in-progress system for monitoring and management of physical, mental and environmental stress which utilizes machine-learning models on the data from commercial devices (accelerometer equipped smartphone, Microsoft Band 2 and NetAtmo weather station) to detect the state of the user and recommend appropriate actions to improve users physical, mental and environmental state.

The goal of the Fit4Work project is to develop a low cost system which can be used in any environment (without the need to renovate or automate the environment). The preliminary results of each module show that selected devices in combination with intelligent algorithms can sufficiently detect sedentary, stressful and unhealthy environment and provide information in the form of recommendations to the user.

In the future, we plan to integrate the three modules into a single system in form of a smartphone application and develop first recommendation prototypes. We will upgrade the machine-learning models for recognition of the activities and estimation of energy expenditure to utilize both devices (the smartphone and the wristband) as seen in our previous work, personalize the stress detection method and upgrade its machine-learning models to utilize other contextual information and enhance the environment quality monitoring with additional machine-learning models.

4. ACKNOWLEDGMENTS

This research was conducted within the Fit4work project co-financed through the AAL Programme.

5. REFERENCES

[1] Kohl, H. W., Craig, C. L., Lambert, E. V., Inoue, S., Alkandari, J. R., Leetongin, G., Kahlmeier, S. 2012. The pandemic of physical inactivity: global action for public health. *The Lancet*, vol. 380, no. 9838, pp. 294–305, 2012.

[2] WHO Mental Stress, http://www.who.int/occupational_health/topics/stressatwp/

[3] Lan, L., Wargocki, P., Lian. Z. 2012. Optimal thermal environment improves performance of office work. REHVA European HVAC Journal (January 2012), 12–17

[4] Fit4Work, http://www.fit4work-aal.eu/

[5] Smartphone statistics, https://www.statista.com/topics/840/smartphones/

[6] Microsoft Band 2, https://www.microsoft.com/

[7] Apple Watch, http://www.apple.com/watch/

[8] Healey J. A., Picard R. W. 2005. Detecting Stress During Real-World Driving Tasks Using Physiological Sensors. IEEE Trans. Intell. Transp. Syst., vol. 6, no. 2, pp. 156–166.

[9] Hovsepian, K., Absi, M., Kamarck, T., Nakajima, M. 2015. cStress : Towards a Gold Standard for Continuous Stress Assessment in the Mobile Environment. In the Proceedings of the 2015 ACM UbiComp, pp. 493-504, 2015.

[10] Empatica, https://www.empatica.com/

[11] Muaremi, A., Bexheti, A., Gravenhorst, F., Arnrich, B., Tröster, G. 2014. Monitoring the Impact of Stress on the Sleep Patterns of Pilgrims using Wearable Sensors. IEEE-EMBS Int. Conf. Biomed. Heal. Informatics, pp. 3–6.

[12] Kumar, P., Martani, C., Morawska, L., Norford, L., Choudhary, R., Bell, M., Leach, M. 2016. Indoor air quality and energy management through real-time sensing in commercial buildings. Energy and Buildings 111, 145–153.

[13] Konuah, A. B. 2014. Occupant window opening behaviour: the relative importance of temperature and carbon dioxide in university office buildings. Ph.D. Dissertation. University of Sheffield.

[14] Candanedo, L. M., Feldheim, V. 2016. Accurate occupancy detection of an office room from light, temperature, humidity and CO 2 measurements using statistical learning models. Energy and Buildings 112, 28–39.

[15] Kolokotsa, D., Pouliezos, A., Stavrakakis, G., Lazos. C. 2009. Predictive control techniques for energy and indoor environmental quality management in buildings. Building and Environment 44, 9, 1850–1863.

[16] NetAtmo, https://www.netatmo.com/

[17] Cvetković, B., Milić R., and Luštrek, M. 2016. Estimating Energy Expenditure With Multiple Models Using Different Wearable Sensors, in IEEE Journal of Biomedical and Health Informatics, vol. 20, no. 4, pp. 1081-1087.

[18] Cvetković, B., Janko, V., Luštrek, M. 2015. Demo abstract: Activity recognition and human energy expenditure estimation with a smartphone, PerCom 2015, pp. 193-195.

[19] SenseWear, https://templehealthcare.wordpress.com/the-sensewear-armband/

[20] Palshikar, G.K. 2009. Simple Algorithms for Peak Detection in Time-Series". Ii the Proc. 1st Int. Conf. Advanced Data Analysis, Business Analytics and Intelligence, June.

[21] Gjoreski, M., Gjoreski, H., Luštrek, M., Gams. M. 2016. Continuous stress detection using a wrist device: in laboratory and real life. In Proceedings of the 2016 ACM UbiComp '16

[22] Frešer, M., Cvetković, B., Gradišek, A., and Luštrek, M. 2016. Anticipatory system for T--H--C dynamics in room with real and virtual sensors. In Proceedings of the 2016 ACM UbiComp '16.

Bayesian binary and ordinal regression with structured uncertainty in the inputs

Aleksandar Dimitriev
University of Ljubljana
Faculty of Computer and Information Science
Večna pot 113, Ljubljana, Slovenia
ad7414@student.uni-lj.si

Erik Štrumbelj
University of Ljubljana
Faculty of Computer and Information Science
Večna pot 113, Ljubljana, Slovenia
erik.strumbelj@fri.uni-lj.si

ABSTRACT

We propose a novel approach to binary and ordinal prediction with structured uncertainty in the input variables. It is based on efficiently approximating the prediction model conditional on the inputs and then marginalizing the conditional model over the input space using Monte Carlo approximation. For efficiency, the well-known Laplace approximation is used for the binary case and we derive a similar approximation for the ordinal case. Empirical evaluation on sports data shows that the proposed approach substantially improves forecasting accuracy and highlights the severity of the problem of uncertainty in the input variables in sports.

Keywords

Laplace approximation, sampling, sports, inference

1. INTRODUCTION

This work is motivated by a problem from sports forecasting [9] - predicting future outcomes from past performances. To illustrate the problem, imagine we are interested in predicting the outcome of a basketball game between two teams. Typically, we would first compile a set of past games with outcomes and relevant performance-related variables, which are typically count variables, such as shots made/missed, rebounds, etc... Then we aggregate these variables across past performances and possibly transform them into variables that are known to be good descriptors of team quality and predictors of future performance, for example, shooting percentage (total shots made divided by total shots attempted). Finally, we would use some binary regression model to model the relationship between the inputs and the outcomes.

What has so far not been taken into account in related work is that past performance variables are noisy and that the noise is transferred to any input variables we derive from them. Fitting a model and failing to account for this uncertainty will result in being overconfident in the parameters of the model and subsequently the forecasts. We hypothesize that forecasting accuracy can be improved by first modelling the uncertainty in the inputs and second taking this uncertainty into account in the model. In this paper we focus on developing an approach for the the second task, while addressing the first in a limited case.

In its essence, the problem is similar to modelling with measurement error, where input variables X (in our case, aggregated counts) are measured with error. This has received a lot of attention [2, 3]. Measurement error models can roughly be split into two groups: functional or structural modelling [2]. Functional models make no or very few assumptions about the distribution of X, while in structured models, a model is placed on X, typically a parametric one.

The difference between our case and a typical measurement error settings is that sports teams recur, so training samples are not independent or identically distributed, but we have multiple measurements of the same input variable. This allows us to use a structural approach as described in section 3.1. However, we also need a model that can handle uncertain inputs and this is the focus of the paper. In Section 2, we propose a Bayesian approach to model the relationship between the noisy inputs and the outputs and incorporate uncertainty associated with the inputs in the form of prior information. The empirical evaluation, data, and results are described in Section 3. With Section 4 we conclude the paper and offer directions for further work.

2. METHODS

The standard prediction setting is the following: we are given a dataset $X \in \mathbb{R}^{n \times m}$, which consists of n observations and m input variables, and a target variable $y \in \mathbb{R}^n$. We then train a model that can produce predictions y^* for a given new x^* by using the training data X. We adopt the Bayesian approach $p(\beta|X, y) \propto p(X, y|\beta)p(\beta)$, where β are the parameters of the model. For the prediction setting, we obtain the probability by marginalizing over the model parameters:

$$p(y^*|x^*, X, y) = \int p(y^*|\beta, x^*)p(\beta|X, y)\, d\beta.$$

In the above, the data are treated as constant. A more general problem arises in our setting, where we instead treat the inputs X and x^* as random variables, with densities $p(X|w)$ and $p(x^*|w^*)$, where w and w^* are known constants. In this case, to train a model and obtain its posterior distribution $p(\beta|w, y)$ we must marginalize over the input space of pos-

sible training data sets X:

$$p(\beta|w,y) = \int p(\beta, X|w,y)\, dX = \int p(\beta|X,y)p(X|w)\, dX.$$

Similarly, to obtain a prediction for y^* we must not only marginalize over the parameters β, but also over the distribution of the test sample x^* as follows:

$$p(y^*|w,w^*,y) = \int_{x^*}\int_{\beta} p(y^*|x^*,\beta)p(x^*|w^*)p(\beta|w,y)\, dx^*\, d\beta \quad (1)$$

This approach is general, since it can be applied with any model that produces a posterior probability distribution over its parameters $p(\beta|X)$ and a distribution over its predictions $p(y^*|x^*,X)$ for a given test sample x^*. What remains for the model to be fully specified is defining the distributions that generate the training and test data sets. The integral in Eq. (1) will generally be intractable even for the simplest of Bayesian models and is typically approximated using Monte Carlo approximation $E[y^*|w^*,w,y] \approx \frac{1}{N}\sum_{i=1}^{N} y^*_{(i)}$, where $y^*_{(i)}$ is a random sample from the posterior predictive distribution in Eq. (1) and can be obtained by sequentially sampling $X_{(i)}$ from $p(X|w)$, $\beta_{(i)}$ from $p(\beta|X=X_{(i)},y)$, $x^*_{(i)}$ from $p(x^*|w^*)$, and finally, $y^*_{(i)}$ from $p(y^*|x^*=x^*_{(i)},\beta=\beta_{(i)})$.

The densities $p(X|w)$, $p(x^*|w^*)$ represent our structural measurement error model and are in most practical cases easy to sample from efficiently. The densities $p(\beta|X=X_{(i)},y)$ and $p(y^*|x^*=x^*_{(i)},\beta=\beta_{(i)})$ are the posterior and posterior predictive for the selected prediction model, conditional on the inputs being fixed, and we have to be able to efficiently sample from them. This implies that we need an efficient prediction model or, in the case of Bayesian models, which are typically computationally intensive, an efficient structural approximation to $p(\beta|X=X_{(i)},y)$.

2.1 Approximate logistic regression

Bayesian logistic regression is a discriminative model that for a given example (x,y), $y\in\{0,1\}^n$, yields a probabilistic prediction: $P(y=1|x)=\sigma(\beta^T x)$, where $\sigma(x)=1/(1+e^{-x})$ is the logistic function and β is the parameter vector of the model. The typically-used prior is $\beta\sim\mathcal{N}(\mu_0,\Sigma_0)$.

A closed-form expression does not exist for neither its posterior nor its predictive distribution. For efficiency, we use the well-known Laplace approximation (see [1], pages 213-215). The approximation to the posterior is the following Gaussian $p(\beta|X,y)\sim\mathcal{N}(q_{map},(\Sigma_0^{-1}+X^TRX)^{-1})$, where R is a diagonal matrix with entries $r_{ii}=\sigma(q_{map}^T x_i)(1-\sigma(q_{map}^T x_i))$ and q_{map} is the posterior maximum, which can be obtained with gradient methods.

2.2 Approximate proportional-odds model

We now derive a Laplace approximation to the proportional-odds model (ordinal logistic regression), which is the most commonly used model for the ordinal setting [6]. Let n and m again be the be the number of samples, and input variables, respectively and k the number of (ordered) categories. The model is based on the assumption that the odds of all binary decisions between categories are proportional to each other or, equivalently, that the k–1 logit surfaces are parallel:

$$\text{logit}(P(Y\le j|x)) = \log\left(\frac{P(Y\le j|x)}{P(Y>j|x)}\right) = \beta x+\alpha_j,$$

for $j\in\{1,..,k-1\}$, where β and α_j are parameters. For convenience, let $\alpha_0=-\infty$ and $\alpha_k=+\infty$. The outcome probabilities can then be written as

$$\begin{aligned}P(Y=j|x) &= P(Y\le j|x) - P(Y\le j-1|x)\\ &= \sigma(\beta x+\alpha_j)-\sigma(\beta x+\alpha_{j-1}), \text{for } j\in\{1,..,k\},\end{aligned}$$

where σ is again the inverse logit function.

The proportional odds model and its generalizations (see [7]) have received very little attention in the Bayesian setting. We now derive the Laplace approximation to the posterior of this model. First, we place priors on the parameters. To ensure in-order intercepts, we introduce parameters d_j, $j\in\{1,..,k-1\}$ and a stick-breaking re-parametrization of the $k-1$ parameters a_j with $a_j=\sum_{i=1}^{j}=d_i$. We place flat priors on the parameters $p(d_i)\propto 1$ and all d_i are restricted to be positive, except for d_1. As in the binary case, we place normal priors on the coefficients $\beta_1,...,\beta_m\sim N(0,\sigma_\beta)$. The model's likelihood is

$$p(\beta,d|\mathcal{D}) \propto \prod_{i=1}^{n}\prod_{j=1}^{k}(R_{i,j}-R_{i,j-1})^{(y_i=j)}\prod_{i=1}^{m} e^{\frac{(\beta_i)^2}{2\sigma_\beta^2}},$$

where $R_{i,j}=\sigma(\beta x_i+\alpha_j)$, and the log-likelihood

$$L = C-\frac{1}{2\sigma_\beta^2}(\beta\circ\beta)+\sum_{i=1}^{n}\sum_{j=1}^{k} I(y_i=j)\log(R_{i,j}-R_{i,j-1}),$$

where I is the indicator function and C is a constant. The gradient of the log-likelihood cannot be written as succinctly as is the case with logistic regression. Instead, we start with the derivative for some parameter θ:

$$\frac{\partial}{\partial\theta}L = -\frac{1}{2\sigma_\beta^2}\frac{\partial}{\partial\theta}(\beta\circ\beta)+\sum_{i=1}^{n}\sum_{j=1}^{k}\frac{\partial}{\partial\theta}L_{i,j},$$

where $\frac{\partial}{\partial\theta}L_{i,j}=0$ if $y_i\neq j$ and

$$\begin{aligned}\frac{\partial}{\partial\theta}L_{i,j} &= \frac{1}{R_{i,j}-R_{i,j-1}}\left(\frac{\partial}{\partial\theta}R_{i,j}-\frac{\partial}{\partial\theta}R_{i,j-1}\right)\\ &= \frac{1}{R_{i,j}-R_{i,j-1}}\Big[R_{i,j}(1-R_{i,j})\frac{\partial}{\partial\theta}(\beta x_i+\alpha_j)\\ &\qquad - R_{i,j-1}(1-R_{i,j-1})\frac{\partial}{\partial\theta}(\beta x_i+\alpha_{j-1})\Big],\end{aligned}$$

otherwise. The derivation of the Hessian is omitted. It had little effect on the accuracy and running times - all results reported in the empirical evaluation were obtained using a numerical approximation to the Hessian.

3. EMPIRICAL EVALUATION

We empirically evaluated our approach on (a) forecasting basketball game outcomes, which always have a winner and can be treated as a binary regression task, and (b) football match outcomes, where draws are also possible, making it an ordinal regression task.

In both cases, the count data are first preprocessed to obtain uncertainty in the input variables, which is described in Section 3.1. As a baseline for comparison, binary logistic regression and ordered logistic regression are included, using mean counts. We also include the baseline with noisy test cases.

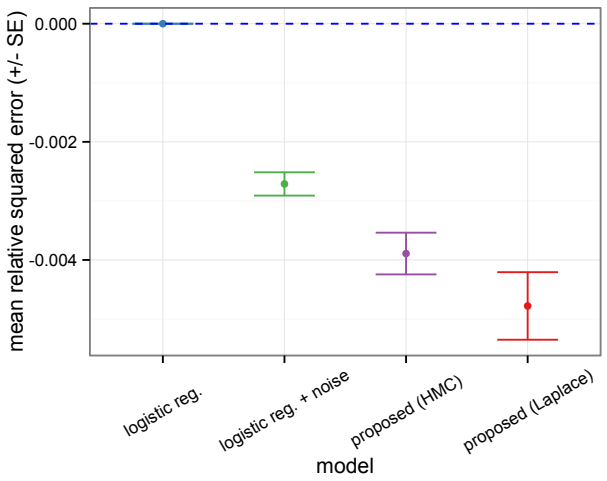


Figure 1: Estimated prediction errors across all train-test season pairs in the NBA basketball data set. All errors are relative to the baseline for comparison (logistic regression).

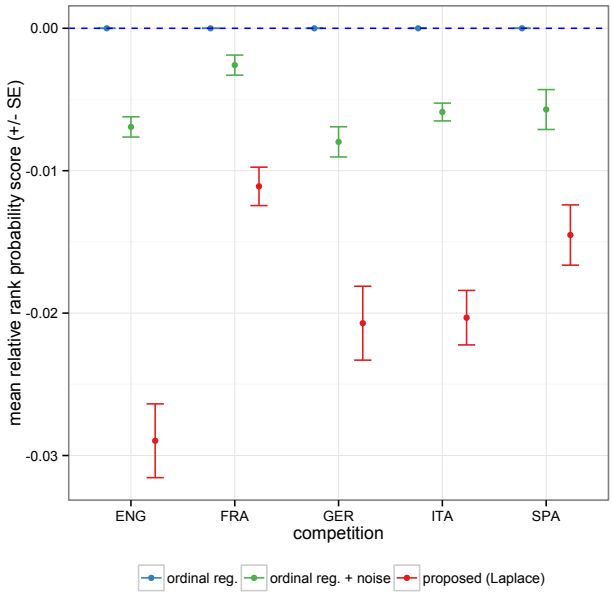


Figure 2: Estimated prediction errors for each football league and across all train-test season pairs in that league. All errors are relative to the baseline for comparison (ordial regression)

This is obtained by treating the test input variables as random, using the same structural model for the inputs as for the proposed model, and approximating the expected prediction of the baseline models with Monte Carlo sampling. For the binary case we also include the proposed model without marginalization - jointly modelling β and X, including the uncertainty in the form of an informative prior on X. We implement this model in the probabilistic programming language and tool for Bayesian inference Stan [8].

Our empirical evaluation procedure is a straightforward measurement of out-of-sample forecasting accuracy, while respecting the time line. We use train-test season pairs, training on one season and forecasting on the next. Only data available prior to a match are used in forecasting it. We measure forecasting accuracy with mean squared error (MSE) in the binary case and the rank probability score (RPS) in the ordinal case [4].

3.1 Preprocessing count data

Input variables that are used as predictors in sports are typically count variables or ratios of count variables, in particular ratios of the form $\frac{A}{A+B}$ where A and B are sums of count variables. We will assume that the count variables follow independent Poisson distributions and are time-homogeneous. While a more sophisticated model for measurement error could be used, we focus on illustrating the benefits of taking into account the uncertainty in the inputs.

A natural choice for the prior distribution of the rate parameter λ is the Gamma distribution $\lambda \sim \text{Gamma}(a_0, b_0)$, which is conjugate. Therefore, for each count variable with mean $\bar{\lambda}_i$ over n_i games, the posterior is again Gamma $\lambda_i|\bar{\lambda}_i, n_i \sim \text{Gamma}(a_0 + \lambda_i n_i, b + n_i)$, where we select weakly informative priors $a_0 = b_0 = 0.001$. A sum of Gamma distributed random variables with the same scale is again Gamma distributed with same scale. Furthermore, if A and B are Gamma distributed random variables with the same scale $A \sim \text{Gamma}(\alpha, \theta)$ and $B \sim \text{Gamma}(\beta, \theta)$, then $X = \frac{A}{A+B}$ is distributed $X \sim Beta(\alpha, \beta)$.. Therefore, for a ratio variable derived from Poisson posterior rates of the form $R = \frac{\sum_i \lambda_{A,i}}{\sum \lambda_{A,i} + \sum \lambda_{B,i}}$ is Beta distributed $R|\bar{\lambda}_{A,i}, \bar{\lambda}_{B,i} \sim \text{Beta}(\sum \bar{\lambda}_{A,i}, \sum \bar{\lambda}_{B,i})$. The scale parameters for the θ are always the same, since the second parameter in the Gamma distribution corresponds to the number of games played, and it is the same for all counts at a particular point in time.

3.2 Basketball data

The basketball data used in our experiments consist of all the regular season and play-off games in the past 13 seasons (12 train-test season pairs) of the National Basketball Association (NBA) from 2001/02 to 2013/14. The data were obtained from `http://www.basketball-reference.com/`. The count variables included in the data are counts of two-point shots made and missed, three-point shots made and missed, turnovers, offensive rebounds and defensive rebounds. We use these counts indirectly by transforming them into 8 ratios, described in [10], which are known to be good predictors of basketball match outcomes. Note that these ratios are based on the well-known four factors effective field goal percentage, rebounding percentage, turnover percentage, and free throw percentage [5].

3.3 Football data

Our football data set consists of 5 complete seasons (2010/11 - 2014/15) of each of the top 5 European national club competitions: English Premier League, French Ligue 1, German Bundesliga, Italian Serie A, and Spanish La Primera. That is, 4 train-test season pairs for each of the 5 leagues for a total of 20. In addition to the outcome, we include,

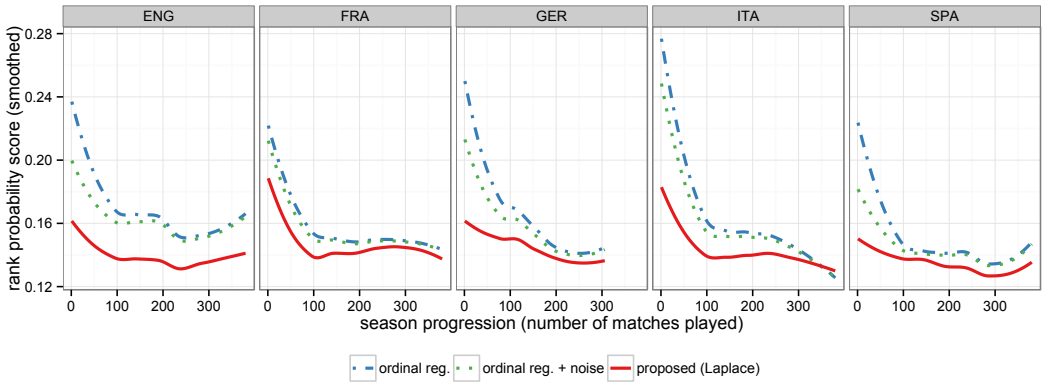


Figure 3: Prediction errors over the course of a season, averaged across all seasons and for each football competition separately.

for each match and each of the two teams that participated in the match, the number of goals scored, shots and shots on target, corners, fouls committed, and yellow and red cards received. The data were obtained from `http://football-data.co.uk/data.php`.

3.4 Basketball results

The structural approximation variant of the proposed model outperforms all other models (see Figure 1). These differences are not only statistically discernible, but also practically relevant (see [10] and references therein). Although the HMC-based approximation yields relatively good predictions, it is discernibly worse than the structural approximation. This can, at least in part, be explained by the inferior accuracy of the HMC-based approximation due to slow mixing (effective sample sizes for β were, on average, $\sim 20\%$ of the total number of iterations and at 1000 iterations, the estimated MCMC sampling error was, on average, approximately 10% of posterior standard deviation).

3.5 Football results

Football results are similar to basketball results, however, the proposed model outperforms the other models even more convincingly across all football competitions (see Figure 2). Compared to NBA basketball, football seasons are much shorter (in terms of matches per team) and there is more uncertainty in the inputs derived from match statistics. As anticipated, the proposed model excels at the beginning of each season and the differences between models' prediction errors decrease as the season progresses and input variables become more certain (see Figure 3). Similar results were observed for basketball, but are omitted for brevity. Note that the HMC variant of the proposed ordinal model was not included in the comparison on football data, because the computation times make it infeasible for practical use.

4. CONCLUSION

Our hypothesis that ignoring the uncertainty in the input variables in sports will lead to less accurate predictions was correct. The proposed model substantially outperformed typically-used models. As expected, the improvement is most substantial at the beginning of the season where uncertainty is highest. This makes the proposed approach a valuable contribution to the sports forecasting toolbox, but it can also be applied in other similar domains. In this paper we focused on prediction with structured uncertainty in the inputs and put less emphasis on the estimation of the uncertainty. Future work could entail developing a model that allows the Poisson rates to vary over time and takes into account the possible correlations between them.

5. REFERENCES

[1] C. M. Bishop. *Pattern recognition and machine learning*. springer, 2006.

[2] R. J. Carroll, D. Ruppert, L. A. Stefanski, and C. M. Crainiceanu. *Measurement error in nonlinear models: a modern perspective*. CRC press, 2006.

[3] W. A. Fuller. *Measurement error models*, volume 305. John Wiley & Sons, 2009.

[4] T. Gneiting and A. E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

[5] J. Kubatko, D. Oliver, K. Pelton, and D. T. Rosenbaum. A starting point for analyzing basketball statistics. *Journal of Quantitative Analysis in Sports*, 3(3), 2007.

[6] P. McCullagh. Regression models for ordinal data. *Journal of the royal statistical society. Series B (Methodological)*, pages 109–142, 1980.

[7] B. Peterson and F. E. Harrell Jr. Partial proportional odds models for ordinal response variables. *Applied statistics*, pages 205–217, 1990.

[8] Stan developement team. *Stan Modeling Language Users Guide and Reference Manual, Version 2.8.0*, 2015.

[9] H. O. Stekler, D. Sendor, and R. Verlander. Issues in sports forecasting. *International Journal of Forecasting*, 26(3):606–621, 2010.

[10] E. Štrumbelj and P. Vračar. Simulating a basketball match with a homogeneous Markov model and forecasting the outcome. *International Journal of Forecasting*, 28(2):532–542, 2012.

Multiobjective discovery of driving strategies using the SCANeR Studio

Erik Dovgan
Artificial Intelligence Laboratory
Faculty of Computer and Information Science
University of Ljubljana
Večna pot 113, SI-1000 Ljubljana, Slovenia
Department of Intelligent Systems
Jožef Stefan Institute
Jamova cesta 39, SI-1000 Ljubljana, Slovenia
erik.dovgan@fri.uni-lj.si

ABSTRACT
This paper presents the SCANeR Studio simulation environment, which is used to design future vehicles by several worldwide companies. In addition, this environment can be used to evaluate autonomous vehicle driving algorithms, although it has several limitations. This paper describes the main environment limitations that are relevant for the evaluation of the Multiobjective optimization algorithm for discovering driving strategies (MODS). Finally, a set of actions for mitigating the SCANeR shortages is given.

Categories and Subject Descriptors
J.7 [**Computer Applications**]: Computers in other systems

Keywords
driving strategies, multiobjective optimization, SCANeR Studio

1. INTRODUCTION
Autonomous vehicle driving is recently being investigated by many automotive and other companies, e.g., Ford [6], Mercedes-Benz [7], Toyota [12], BMW [5], Audi [8], and Google [13]. Google's project of autonomous vehicles is led by Sebastian Thrun, former director of the Stanford Artificial Intelligence Laboratory and co-inventor of Google Street View, who wan the DARPA autonomous car competition. Several driver assistance systems are already installed in modern vehicles, such as line assist (see, e.g., Volkswagen [15], Audi [1], and Toyota [14]). In addition, fully autonomous vehicles start driving in urban environments. For example, 100 self-driving Volvo cars will drive on public roads around the city of Gothenburg by 2017 [16]. Till June 2015, the Google autonomous vehicles have driven over 1 million miles encountering 200,000 stop signs, 600,000 traffic lights, and 180 million other vehicles.

The development of an autonomous driving solution usually focuses on the determination of the vehicle surroundings, e.g., other vehicles, obstacles, and pedestrians, in order to increase safety and avoid collisions. However, such a strategy may miss to achieve other objectives that are also important. An objective that has to be taken into account is the reduction of the environmental pollution. This objective is especially important since the awareness of the need to protect our environment is increasing. When driving a vehicle, the pollution reduction is proportional to the reduction of the fuel consumption. An interesting side effect of this is a reduction in traveling cost, although this is countered by an increase in traveling time. It is clear that heavily reducing fuel consumption is not optimal because it leads to unacceptably long traveling time. Therefore, several objectives need to be taken into account simultaneously when constructing a driving strategy.

To obtain autonomous vehicle driving, real-time driving strategies have to be applied by the autonomous vehicle driving algorithms. These strategies have to select the best control action at each step with respect to the current vehicle and route state, e.g., position on the route, position of neighbor vehicles, velocity of neighbor vehicles etc. The best control action can be selected by implementing an algorithm that optimizes various objectives, such as the traveling time, the fuel consumption, the distance to other vehicles for collision avoidance etc. Since several objectives have to be simultaneously taken into account, it is preferable to use the multiobjective approach when developing the optimization algorithm, since it is in general the case that multiobjective algorithms exhibit advantages over single-objective algorithms (since they enable, e.g., better exploration of the multiobjective search space).

To use the autonomous vehicle driving algorithms in practice, the proper evaluation of these algorithms is of key importance. Since untested algorithms cannot be simply deployed in autonomous vehicles and tested in real environment, e.g., city roads, the evaluation has to be performed in the near-real-life driving simulation environment. Typically, the best existing driving simulation environments are used by the worldwide automotive companies. This paper examines the SCANeR Studio simulation environment [9], which is used by several worldwide companies such as Renault, PSA Peugeot Citroën and VOLVO Trucks [10].

The goal of the ongoing research is to enhance the existing Multiobjective optimization algorithm for discovering driving strategies (MODS) [2, 3, 4] to take into account real-life driving situations and to evaluate it with the SCANeR Studio. To this end, MODS will have to be significantly adapted due to limitations of the SCANeR Studio.

The paper is further organized as follows. Section 2 describes the existing Multiobjective optimization algorithm for dis-

covering driving strategies. The SCANeR Studion and its limitations are presented in Section 3. Section 4 presents the ongoing and future enhancements of the MODS algorithm. Finally, Section 5 concludes the paper with the summary of work.

2. MULTIOBJECTIVE OPTIMIZATION ALGORITHM FOR DISCOVERING DRIVING STRATEGIES

Multiobjective optimization algorithm for discovering driving strategies (MODS) [2, 3, 4] is a two-level multiobjective optimization algorithm that finds a set of nondominated driving strategies with respect to two conflicting objectives: traveling time and fuel consumption. The lower-level algorithm is based on a deterministic breadth-first search, driving prediction and nondominated sorting, and searches for nondominated driving strategies. The search is performed by simulating vehicle driving by route steps. It starts with a single driving strategy with no control actions. If for a route step the driving strategy does not define the control action, the driving strategy is cloned for each possible control action and the obtained driving strategies are stored in the set of driving strategies. When the control action is defined, it is used to predict the vehicle driving for a predefined number of prediction steps. Due to the cloning, the number of driving strategies grows exponentially. To maintain a constant number of driving strategies at each route step, fast nondominated sorting is used to select the most promising driving strategies.

The upper-level algorithm is an evolutionary algorithm that optimizes the input parameters for the lower-level algorithm. This algorithm applies evolutionary mechanisms, i.e., selection, crossover and mutation, to the sets of input-parameter values through generations and maximizes the hypervolume covered by the driving strategies found by the lower-level algorithm.

MODS was implemented in two variants. The original MODS algorithm minimizes the traveling time and the fuel consumption [2, 3], while the enhanced version, called MOCDS, optimizes also the driving comfort [4].

Currently, the MODS algorithm is integrated and was evaluated in a vehicle driving simulator that includes data from real-world routes and a black-box vehicle simulator. The results show that MODS finds better driving strategies than existing algorithms for discovering driving strategies. However, MODS has some limitations. For example, it does not take into account neighbor/other vehicles on the route, it does not produce human-like driving strategies, and it does not find (good) driving strategies in real time. To overcome these shortages, the ongoing research consists of design and implementation of the enhanced MODS algorithm, and its deployment in the near-real-life environment, i.e., the SCANeR Studio simulation environment.

3. SCANeR STUDIO SIMULATION ENVIRONMENT

The SCANeR Studio [9] is a modular driving simulation environment, developed by the OKTAL company [11]. It simulates the vehicle driving behavior and the behavior of various entities in the environment. To this end, it consists of several modules such as:

- Complex dynamic vehicle models simulating the behavior of every component of a real vehicle
- Simple models of autonomous traffic vehicles simulating the behavior of other vehicles on the road
- Models for simulating the behavior of pedestrians

The behavior of these models can be predefined, controlled online with the scripting language or managed through APIs using custom algorithms. In addition, complex vehicle models can be controlled by the driver through the acquisition modules that can be connected to keyboards, gaming steering wheels etc. Vehicle driving is presented to the driver using visual and sound modules, and the module that controls dynamic platforms (if present). Moreover, the models involved in the simulation can be controlled/executed with either SCANeR user interface or programs that control the simulation, i.e., the supervisors.

The SCANeR Studio enables to create a complete driving simulation environment through the usage of five dedicated modes of the graphic interface:

- Terrain mode: Road network creator allowing the rapid creation of realistic road networks that are useable directly in the simulation
- Vehicle mode: Tool for fine-tuning and study of dynamic vehicle models
- Scenario mode: Driving simulator scenario editing tool
- Simulation mode: Simulation supervision tool
- Analysis mode: Detailed graphical analysis tool

Although The SCANeR Studio is a powerful tool for evaluating autonomous vehicle driving algorithms, it introduces some limitations for the development of these algorithms. When taking into account the MODS algorithm, the SCANeR Studio limitations presented Section 3.1 are the most critical ones.

3.1 Limitations of the SCANeR Studio

Limited implementation of complex dynamic vehicle models

The SCANeR Studio contains several types of complex dynamic vehicle models such as cars, trucks and buses. The instances of these models can be controlled only with throttle, braking and clutch pedals, gearbox, steering wheel etc. On the other hand, if an autonomous vehicle driving algorithm controls the vehicle by setting vehicle's velocity, such algorithm cannot be evaluated with complex vehicle models.

The SCANeR Studio contains also a large set of various simple vehicle models. These models define some vehicle parameters such as length, weight etc., but do not simulate engine behavior in details. For example, they do not

simulate fuel consumption or engine limitations. Since no engine limitation is simulated, these simple models enable to instantaneously change the vehicle velocity without limitations, which results in unrealistic driving behavior. The advantage of these algorithms is that they can be controlled by autonomous vehicle driving algorithms that set vehicle's velocity.

Limited offline simulation
The SCANeR Studio enables to execute the simulation in an offline mode that schedules the execution of the SCANeR and other modules, involved in the simulation, with respect to their running frequency, and then executes them with respect to the schedule without any sleep time. This enables to execute the simulation as fast as possible, which is specially suitable when the autonomous vehicle driving algorithms control the vehicle behavior and no user interaction is required.

The offline simulation can, however, be used only when user executes such a simulation thought the SCANeR user interface. On the other hand, when the simulation has to be executed by a supervisor program, the offline simulation cannot be used. This represents a significant limitation for optimization programs that need to test various parameter settings of an autonomous vehicle driving algorithm, where each setting has to be evaluated with the execution of the same simulation scenario.

Driving prediction is inapplicable
Several autonomous vehicle driving algorithms use the driving prediction technique to construct the driving strategy and select the best control action at each step. The usage of driving prediction requires to periodically (ideally at each step) stop the driving simulation, start the driving prediction from the current vehicle and route state possibly multiple times if various control actions have to be tested, and then apply the obtained information during prediction in order to continue with the driving simulation. To execute such a procedure, the main simulator has to simulate the driving, while additional simulators have to be periodically executed to predict the driving. Such a configuration is not supported by the SCANeR Studio, since SCANeR requires to execute only one instance of SCANeR modules simultaneously. Otherwise, conflicts can appear in the communication between modules if several instances of the same module are active simultaneously.

Synchronization issues
The modules of the SCANeR Studio are implemented and executed as independent processes that communicate between them through network. Since they are executed as independent processes, their actual execution depends on the hardware and OS specifications, e.g., the number of processor cores that can simultaneously execute various processes. Consequently, the sequence of messages sent between modules is not deterministic, which is also due to the network latency. For example, messages from vehicle model containing the current vehicle velocity may sometimes be delayed with respect to messages for autonomous vehicle driving algorithm containing control actions, which may result in delayed reaction of the autonomous vehicle driving algorithm. The main issue of such behavior is the fact that the simulation results are not deterministic, i.e., the same scenario with the same SCANeR modules does not always produce the same driving behavior and consequently the same traveling time, fuel consumption etc. Therefore, an instance of the autonomous vehicle driving algorithm has to be evaluated several times in order to statistically evaluate its quality.

The synchronization issues are, however, present only in the online simulation mode. On the other hand, the offline simulation mode does not have these issues, since it schedules the execution of the SCANeR modules in advance and then executes them with respect to the predefined, deterministic schedule. Consequently, the sequence of messages sent between modules is also deterministic.

3.2 Adaptation of MODS to the SCANeR Studio

The limitations of the SCANeR Studio significantly influences the enhancement of the MODS algorithm. Since the MODS algorithm controls the vehicle by setting vehicle's velocity, it requires to use SCANeR's simple vehicle models. Due to the fact that these models do not simulate engine behavior in details, such engine behavior will have to be additionally implemented. This will require the implementation of, for example, wheel friction, aerodynamic, slope friction, inertial, maximum torque, specific fuel consumption, and maximum engine moving functions, in order to limit the vehicle inertial force and calculate the fuel consumption.

The MODS algorithm will require the usage of offline simulation in order to execute the multiobjective optimization algorithm. Ideally, the multiobjective optimization algorithm should be implemented as a supervisor program. Due to the fact that a supervisor algorithm cannot control the offline simulation, the simulation will have to be controlled with HTTP POST calls. Although controlling SCANeR with POST calls is not optimal, this is the only mode that can be currently applied for the offline simulation.

Driving prediction is one of the key mechanisms of the MODS algorithm. Since this prediction is inapplicable in the SCANeR Studio, the MODS algorithm will have to be significantly redesigned. To compensate the missing driving prediction, a simple prediction approach will have to be added to the MODS algorithm, e.g., a set of rules that predict the future collision time.

MODS also requires exact evaluation of the driving strategies. Since this can be achieved only by the offline simulation mode, only this mode will be used. Consequently, the driving strategies obtained with the online simulation mode will have to be re-evaluated with the offline simulation mode before the comparison with the MODS driving strategies.

4. ENHANCEMENTS OF THE MODS ALGORITHM

The MODS algorithm is going to be significantly enhanced in order to take into account real-life driving situations and to be evaluated with the SCANeR Studio. Currently, MODS is being adapted in order to control the vehicle's velocity. In addition, the enhanced algorithm uses only the current

and the next velocity limits as hypercube dimensions of the driving strategies [3]. The other attributes, such as the route inclination, are added to the formulas that change the target velocity. In the future, the hypercube dimensions will be redefined by taking into account the results of the analysis of the human driving strategies.

To adapt to the SCANeR Studio limitations, the prediction of the lower-level MODS is not used anymore. Nevertheless, a simple prediction will added to the MODS algorithm if needed. The lower-level MODS will be also enhanced to simulate the car following, lane changing, and vehicle overtaking behavior.

The upper-level MODS is going to be adapted to take into account the enhanced definition of the hypercubes. The new parameters of the driving strategies, which will be stored in hypercubes, will include the vehicle velocity, acceleration, deceleration, duration of acceleration, duration of deceleration, etc. In addition, the upper-level MODS will also be enhanced to search for parameter values for the car following, lane changing, and vehicle overtaking behavior.

5. CONCLUSIONS

This paper presented the SCANeR Studio simulation environment and its limitations that are relevant for the enhancement of the Multiobjective optimization algorithm for discovering driving strategies (MODS) and its deployment in this Studio. Four main limitations were described: limited implementation of complex dynamic vehicle models, limited offline simulation, inapplicability of driving prediction, and synchronization issues. The analysis of MODS and SCANeR Studio shows that MODS will need several adaptations before the evaluation with the SCANeR Studio, e.g., the implementation of engine behavior and the algorithm redesign to remove the driving prediction from the algorithm and adapt it accordingly.

6. ACKNOWLEDGMENTS

The work presented in this paper was in part funded by the NERVteh, raziskave in razvoj, d.o.o., which also provided the SCANeR Studio simulation environment. In addition, this work was also in part funded by the Slovenian Research Agency under research project Z2-7581.

7. REFERENCES

[1] Audi. Stay in lane, 2014. Available online: http://www.audi.co.uk/new-cars/a3/a3-sportback/driver-assistants/audi-active-lane-assist.html.

[2] E. Dovgan, M. Javorski, T. Tušar, M. Gams, and B. Filipič. Comparing a multiobjective optimization algorithm for discovering driving strategies with humans. *Expert Systems with Applications*, 40:2687–2695, 2013.

[3] E. Dovgan, M. Javorski, T. Tušar, M. Gams, and B. Filipič. Discovering driving strategies with a multiobjective optimization algorithm. *Applied Soft Computing*, 16:50–62, 2014.

[4] E. Dovgan, T. Tušar, M. Javorski, and B. Filipič. Discovering comfortable driving strategies using simulation-based multiobjective optimization. *Informatica*, 36:319–326, 2012.

[5] S. Elmer. BMW targets 2020 for self-driving cars, 2013. Available online: http://www.autoguide.com/auto-news/2013/02/bmw-targets-2020-for-self-driving-cars.html.

[6] K. Fitchard. Ford is ready for the autonomous car. Are drivers?, 2012. Available online: http://gigaom.com/2012/04/09/ford-is-ready-for-the-autonomous-car-are-drivers/.

[7] N. Ingraham. Mercedes-Benz shows off self-driving car technology in its new $100,000 S-Class, 2013. Available online: http://www.theverge.com/2013/5/18/4341656/mercedes-benz-shows-off-self-driving-car-technology.

[8] D. Johnson. Audi predicts self-driving cars by 2020, 2013. Available online: http://www.leftlanenews.com/audi-predicts-self-driving-cars-by-2020.html.

[9] Oktal. Automotive simulators, 2016. Available online: http://www.oktal.fr/en/automotive/range-of-simulators/software.

[10] Oktal. Automotive simulators references, 2016. Available online: http://www.oktal.fr/en/automotive/references-automotive-simulator.

[11] Oktal. OKTAL, 2016. Available online: http://www.oktal.fr/.

[12] R. Read. Toyota will roll out autonomous cars by the mid-2010s, 2013. Available online: http://www.thecarconnection.com/news/1087636_toyota-will-roll-out-autonomous-cars-by-the-mid-2010s.

[13] R. J. Rosen. Google's self-driving cars: 300,000 miles logged, not a single accident under computer control, 2012. Available online: http://www.theatlantic.com/technology/archive/2012/08/googles-selfdriving-cars-300-000-miles-logged-not-a-single-accident-under-computer-control/260926/.

[14] Toyota. Lane Keeping Assist, 2014. Available online: http://www.toyota-global.com/innovation/safety_technology/safety_technology/technology_file/active/lka.html.

[15] Volkswagen. Lane assist, 2014. Available online: http://www.volkswagen.co.uk/technology/proximity-sensing/lane-assist.

[16] Volvo. Volvo Car Group initiates world unique Swedish pilot project with self-driving cars on public roads, 2013. Available online: https://www.media.volvocars.com/global/en-gb/media/pressreleases/136182/volvo-car-group-initiates-world-unique-swedish-pilot-project-with-self-driving-cars-on-public-roads.

Ali nam metode strojnega učenja lahko pomagajo pri načrtovanju novih visokoentropijskih zlitin?

Anton Gradišek
Institut »Jožef Stefan«
Jamova 39
1000 Ljubljana, Slovenija
anton.gradisek@ijs.si

Andraž Kocjan
Institut »Jožef Stefan«
Jamova 39
1000 Ljubljana, Slovenija
andraz.kocjan@ijs.si

Miha Mlakar
Institut »Jožef Stefan«
Jamova 39
1000 Ljubljana, Slovenija
miha.mlakar@ijs.si

POVZETEK

Visokoentropijske zlitine (angleško: high entropy alloys, HEA) so zanimiva skupina kovinskih materialov, ki so v zadnjem času pritegnile pozornost znanstvenikov zaradi lastnosti, zanimivih za uporabo v industriji. HEA so sestavljene iz petih ali več elementov, ki so med seboj naključno pomešani. Kljub mešanju pa nered ne pokvari globalne strukture, ki ostane pravilen kristal. Zaradi kompleksne zgradbe HEA težko obravnavamo s klasičnimi računskimi metodami znanosti materialov. V tem prispevku raziščemo, če lahko uporabimo metode strojnega učenja na zbirki že odkritih HEA, da bi lahko napovedovali strukturo sistemov z novimi kemijskimi sestavami. Ugotovili smo, da binarni klasifikator dobro določi posamezne strukture, s točnostjo tudi nad 90 %, medtem ko bi za natančnejše delovanje klasifikatorja za več razredov potrebovali večjo količino podatkov.

Ključne besede

Visokoentropijske zlitine, strojno učenje, kristalna struktura

1. UVOD

Visokoentropijske zlitine (HEA) so relativno nova skupina kovinskih materialov. Prvič so jih v znanstveni literaturi sistematično opisali leta 2004, bolj intenzivne raziskave na tem področju pa so se začele v zadnjem desetletju. HEA so zlitine najmanj 5 kovinskih elementov, pri tem, da je vsak od glavnih elementov zastopan vsaj med 5 in 35 atomskimi odstotki (lahko so prisotni tudi drugi elementi v manjših količinah).[1] Ti elementi imajo nizko entalpijo mešanja, zaradi česar v Gibbsovi enačbi proste energije ($\Delta G = \Delta H - T\Delta S$, kjer je G Gibbsova energija, H entalpija, T temperatura in S entropija) pride do prevlade entropijskega člena, kjer je entropija definirana z Bolzmannovim zakonom o konfiguracijski entropiji. Slednja je najvišja ob ekvimolarnem deležu vseh elementov in je tedaj odvisna le še od števila dodanih elementov N, $\Delta S = R \ln N$, kjer je R splošna plinska konstanta. Poenostavljeno bi lahko dejali, da entropijski člen v enačbi (ki predstavlja neurejenost sistema) v primeru HEA prevlada nad entalpijskim členom, ki bi sicer pripeljal do segregacije posameznih elementov oziroma do tvorbe intermetalnih faz z le dvema do tremi elementi in običajno kompleksnejšo strukturo, kot jo imajo HEA zlitine sicer. Če namreč pogledamo fazne diagrame za zlitine iz dveh ali treh elementov, vidimo, da v njih nastopa množica različnih faz in struktur, kar pomeni, da imajo take zlitine običajno kompleksno mikrostrukturo. Posledično so običajno krhke in brez posebne praktične uporabe. Po drugi strani pa ravno entropijski člen v HEA povzroči, da se atomi različnih elementov uredijo v relativno majhno število enostavnih struktur. Pri tem je potrebno poudariti, da v HEA običajno nastopajo elementi, katerih atomi imajo podobne atomske radije – zato pri zamenjavah ne nastopijo problemi zaradi različnih velikosti.

Zaradi specifične strukture imajo HEA vrsto obetavnih fizikalnih in kemijskih lastnosti, zaradi katerih so pritegnili zanimanje inženirjev za uporabo v industriji. Gre za odlične mehanske lastnosti (denimo visoka natezna trdnost), odpornost na korozijo, nizko obrabo, obstojnost pri visokih temperaturah itd. Poleg tega je sinteza HEA relativno enostavna (običajno se pripravi talino ustrezne mešanice elementov in jo ohladi) in ne potrebuje naprednih metod, kot je denimo rotacijsko kaljenje, ki se uporablja za pripravo sistemov, ki so sicer obstojni v temperaturnem območju, daleč od sobne temperature.[1]

Specifična struktura in sestava HEA predstavlja določene probleme pri usmerjenem načrtovanju strukture z metodami, ki so sicer dobro poznane za enostavnejše kovinske sisteme. Pri enostavnih sistemih namreč lahko definiramo osnovno celico kristala, s translacijami te celice pa nato lahko rekonstruiramo celoten kristal. Za matematičen opis zato uporabimo periodične funkcije, kar omogoči računsko določanje različnih lastnosti. V HEA tega pristopa ne moremo uporabiti. Čeprav je struktura enostavna (najpogosteje ploskovno ali prostorsko centrirana kocka), zaradi naključne razporeditve elementov ne moremo govoriti o pravi periodičnosti. Zato so zaželene nove metode, s katerimi bi lahko dobili več informacij o HEA, ki bi nas pripeljale do bolj načrtovanega načina sinteze novih sistemov.[2]

V pričujočem prispevku pogledamo, če namesto eksaktnih fizikalnih izračunov za napovedovanje strukture HEA lahko uporabimo bazo že znanih sistemov (v člankih [3,4] jih je opisanih okrog 300) v povezavi z algoritmi strojnega učenja. Ti algoritmi nam lahko odkrijejo povezave med posameznimi parametri, ki jih sicer morda ne bi opazili, saj se pokažejo šele v visoko dimenzionalnem prostoru. Podobni pristopi se že uporabljajo na področjih, kot je medicina, kjer denimo uporabljajo metode strojnega učenja za iskanje povezave med posameznimi geni in vrstami raka, ki bi ga ti geni lahko povzročali.[5,6] Tu predstavimo prvo enostavno študijo, kjer nas zanima, kako dobro lahko s pomočjo algoritma napovemo strukturo HEA z znano sestavo.

2. PODATKI IN METODA

Za bazo podatkov smo uporabili seznam HEA iz članka avtorjev Toda-Caraballo in Rivera-Diaz-del-Castillo [6], kjer je opisanih 323 znanih HEA. Za vsakega od sistemov so navedeni naslednji podatki:

- Družina oz. seznam elementov, ki v HEA nastopajo, denimo AgAuCuPdSi.

- Kemijska sestava, ki pove tudi razmerje med elementi. Pri HEA CoCrFeNiCu so elementi zastopani v enakih atomskih razmerjih, pri $Ti_{0,5}Co_{1,5}CrFeNi_{1,5}Mo_{0,5}$ pa ne.
- Struktura. Najpogostejše strukture so BMG (bulk metallic glass – kovinsko steklo), FCC (face-centred cubic – ploskovno centrirana kocka), BCC (body-centred cubic – telesno centrirana kocka) ter kombinacije FCC in BCC.
- δ% - razlika v atomskih radijih
- ΔH_{mix} – mešalna entalpija (v kJ/mol)
- $\Delta\chi$% - razlika v elektronegativnosti
- Ω, parameter, definiran kot $\Omega=T_m \Delta S_{mix}/| \Delta H_{mix} |$, kjer je ΔS_{mix} entropija mešanja in T_m temperatura tališča
- μ, parameter, definiran kot $\mu= T_m/T_{SC}$, kjer je T_{SC} temperatura spinodalne dekompozicije
- VEC – koncentracija valenčnih elektronov (ang.= valence electron concentration)
- e/a – število valenčnih elektronov na število atomov v osnovni celici
- $s_{m\%}$ – neujemanje medatomskih razdalj
- K_m% - neujemanje parametrov mreže

Večino zgornjih parametrov lahko izračunamo iz atomske sestave in parametrov za posamezne atome.[6] Nas je zanimalo, če lahko na podlagi opisanih parametrov določimo fazno sestavo ter njihovo kristalno strukturo. Število HEA s posamezno strukturo je predstavljeno v Tabeli 1.

Tabela 1. Število HEA s posamezno strukturo v bazi podatkov

Struktura	N
BMG	76
FCC+BCC	74
BCC	31
FCC	31
Ostalo*	111

* V razredu Ostalo se nahajajo vse preostale strukture, ki so zastopane z manj kot 30 primeri. Gre za intermetalike (IM), heksagonalno celico (HCP), Lavesovo fazo (L), kombinacije zgoraj navedenih in drugo. Skupaj imamo 42 možnih struktur, vendar pa je večina le-teh zastopana z le po enim ali dvema primeroma.

Napovedovanje strukture HEA lahko obravnavamo kot klasifikacijski problem, ki se ga lotimo z algoritmi strojnega učenja. Pri tem strukturo obravnavamo kot razred, devet parametrov (δ%, ΔH_{mix}, $\Delta\chi$%, Ω, μ, VEC, e/a, s_m% in K_m%) pa kot atribute. Preizkusili smo več algoritmov, ki so vgrajeni v program Weka.[7] Vsakič smo klasifikacijsko točnost preverili z metodo navzkrižne validacije, pri kateri začetno množico podatkov razdelimo na N podmnožic, nato zgradimo skupen klasifikacijski model na N-1 podmnožicah in ga testiramo na preostali. To ponovimo N-krat.

3. REZULTATI

Klasifikacijskega problema se lahko lotimo na več načinov. Idealno bi bilo izdelati klasifikator, ki bi bil sposoben napovedati katerokoli od struktur, ki nastopajo v naši bazi podatkov. Ker pa imamo v več kot polovici od 42 razredov po le en ali dva primera, se taka klasifikacijska metoda ne bo dobro obnesla. Če imamo premalo učnih podatkov, se algoritem namreč dobro nauči prepoznavati le obstoječe primere, ko dodamo nove primere, pa odpove.

Klasifikacijski model zato najprej poskušamo zgraditi na petih razredih, torej na štirih najpogostejših strukturah in razredu Ostalo. Kot klasifikacijski algoritem se v tem primeru najbolje obnese Random Forest (naključni gozd), ki ima 62 % klasifikacijsko točnost.

Bistveno boljše rezultate dobimo, če zgradimo binarne klasifikatorje. V en razred postavimo primere z določeno strukturo, drug razred pa predstavljajo vsi ostali primeri. Tudi tokrat se najbolje obnese metoda Random Forest. Klasifikacijske točnosti za binarne klasifikatorje s to metodo so predstavljene v Tabeli 2.

Tabela 22. Klasifikacijske točnosti binarnega klasifikatorja Izbrana struktura vs. Ostale strukture z metodo Random Forest

Struktura	Točnost (%)
BMG	91
FCC+BCC	87
BCC	77
FCC	92

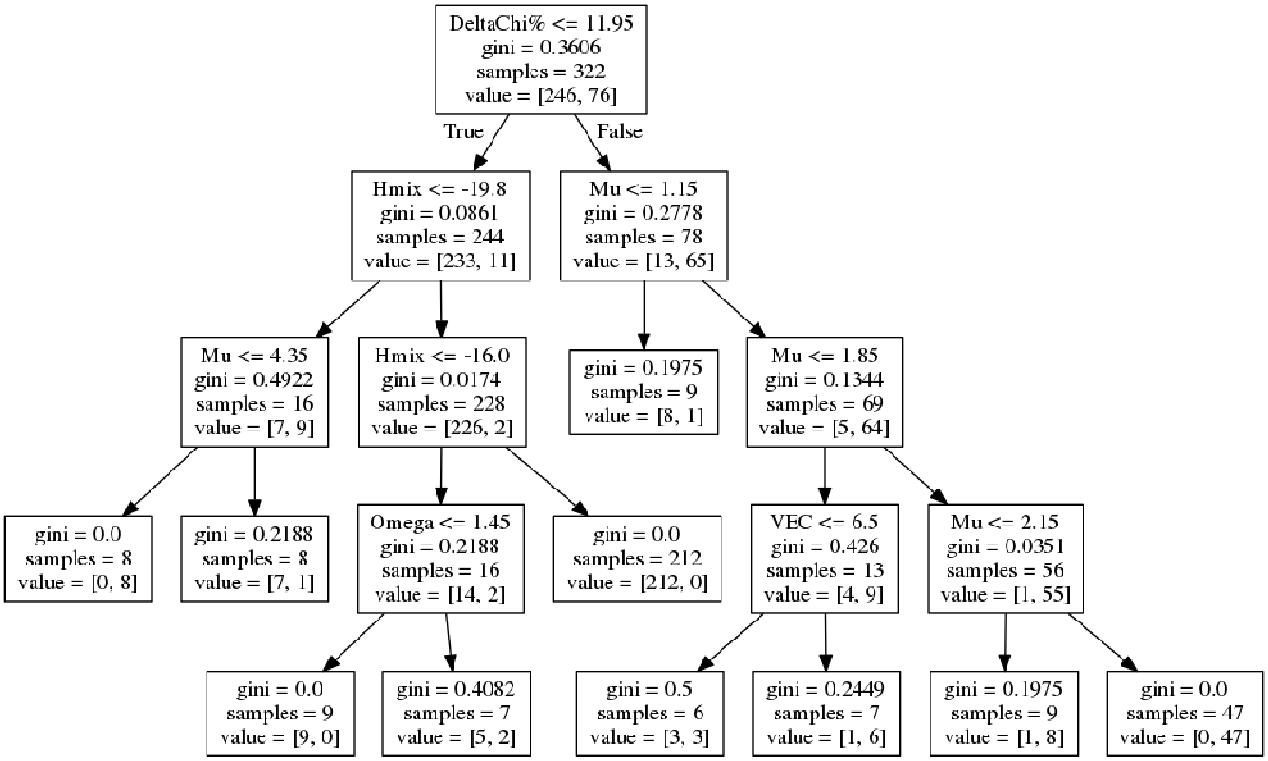
Vidimo, da so klasifikacijske točnosti binarnega klasifikatorja veliko višje od tistega, ki deluje na petih razredih. V primeru večkratnega klasifikatorja imamo namreč v vsaki od množic majhno število elementov, pri binarnem klasifikatorju pa nekaj od teh množic združimo in tako povečamo število elementov.

Metoda Random Forest deluje tako, da na naključnih podmnožicah učne množice zgradi odločitvena drevesa, nato pa s pomočjo teh dreves klasificira nov primer – končni razred je tisti, ki ga napove največ dreves. Čeprav je ta metoda najbolj uspešna pri klasifikaciji, ni intuitivno razumljiva za uporabnika. Podatke lahko namesto tega predstavimo s posameznim odločitvenim drevesom (klasifikacijski algoritem J48), kjer v vsakem listu drevesa naredimo odločitev glede na določen parameter. Primer takega odločitvenega drevesa za binarni klasifikator BMG vs. Ostale strukture je prikazan na Sliki 1. Pod spremenljivko Value prva vrednost označuje število ostalih struktur, druga pa število BMG. Klasifikacijska točnost tega drevesa je 89 %. Vidimo, da je najpomembnejši parameter δ%, sledita mu ΔH_{mix} in μ.

4. DISKUSIJA

Kot pokaže naša enostavna študija, je uporaba metod strojnega učenja lahko koristna pri napovedovanju strukture visokoentropijskih zlitin. V znanosti materialov je še posebej zanimivo iskanje prehodov med posameznimi strukturami. Naš pristop nam omogoča izdelavo enostavnega simulatorja strukture. Če nas denimo zanima prehod med BMG in BCC, ko spreminjamo razmerje dveh od petih elementov v zlitini, nam tak

simulator lahko pove, okrog kakšne sestave lahko pričakujemo prehod med strukturama. Na ta način lahko k sintezi novih sistemov pristopimo bolj sistematično in ne le na podlagi poskušanja. Naše modele lahko izboljšamo z večjo količino podatkov v učni množici, preučiti pa velja tudi, ali lahko uvedemo še dodatne atribute, ki bi v model vnesli nove informacije.



Slika 1: Odločitveno drevo za binarni klasifikator BMG vs. Ostale strukture.

5. REFERENCE

[1] Tsai, M. H., Yeh, J. W., High-Entropy Alloys: A Critical Review, Materials Research Letters, 2014, 2, 107-123.

[2] Murty, B.S., Yeh, J.W., Ranganathan, S. High-Entropy Alloys, Butterworth-Heinemann, Elsevier, 2014, p. 57.

[3] Poletti, M. G., Battezzati, L. Electronic and thermodynamic criteria for the occurrence of high entropy alloys in metallic systems, Acta Materialia, 2014, 75, 297-306.

[4] Toda-Caraballo, I, Rivera-Diaz-del-Castillo, P. E. J., A criterion for the formation of high entropy alloys based on lattice distortion, Internetallics, 2016, 71, 76-87.

[5] Wang Y. et al., Gene selection from microarray data for cancer classification—a machine learning approach, Computational Biology and Chemistry, 2005, 29, 37-46

[6] Guyon I. et al., Gene Selection for Cancer Classification using Support Vector Machines, Machine Learning, 2002, 46, 389-422.

[7] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten I. H. The WEKA Data Mining Software: An Update; SIGKDD Explorations, 2009, 11, 1.

Markov chain model for energy-efficient context recognition

Vito Janko
Jožef Stefan Institute
Jožef Stefan International Postgraduate School
Jamova 39, 1000 Ljubljana, Slovenia
vito.janko@ijs.si

Mitja Luštrek
Jožef Stefan Institute
Jožef Stefan International Postgraduate School
Jamova 39, 1000 Ljubljana, Slovenia
mitja.lustrek@ijs.si

ABSTRACT
Continuous sensing of the user and subsequent context recognition using wearable sensors is a popular area of research. One of the big problems of such automatic context recognition is the battery life of the sensing device. To maximize the energy efficiency, the context recognition system should adapt its settings to the current situation. Choosing the appropriate setting for each situation usually requires either a lot of expert knowledge or extensive experimentation. We propose a method that simulates all possible combinations of contexts and settings using a Markov chain model, automating and speeding up the whole process. We show on a small example that the simulation is accurate, and that it allows us to quickly select best trade-off between the energy efficiency and context-recognition accuracy.

Categories and Subject Descriptors
G.3 [**Probability and statistics**]: Markov processes; H.2.8 [**Database Applications**]: Data mining

Keywords
Context recognition; continuous sensing; energy efficiency; Markov chain; acitivity recognition.

1. INTRODUCTION
Widespread accessibility of wearable sensing devices allows for many possibilities for tracking the users who wear them. Possible applications range from measuring their exercise patterns and checking on their health, to giving them location-specific recommendations. The recognition of user's context using sensors is a popular and mature area of research [2]. For example, using activity recognition, we can recognize "walking, "running", "resting" and similar activities from accelerometer data. This task was made easier and more practical with the increased use of smartphones, which have many sensors built in and are often carried. Sensing with multiple sensors, possibly at once, opens additional options for context recognition: detecting one's location, ambient sound level, or some higher level activities such as "shopping", "traveling" or "working".

A major limitation of such continuous sensing and context recognition is its heavy toll on the sensing device's battery life. This is especially relevant for smartphones, which have a very limited battery that must be shared between many applications, but the same limitation applies to basically any wearable device. This issue is often neglected when discussing the design of context-recognition systems, however it is an essential component if such systems are to be used in practice.

Some solutions deal with this issue by optimizing the sampling rate or sensor duty cycle for the particular recognition task [1]. This alone however, might be suboptimal. We might for example want to have the GPS active at a high sampling rate while the user is driving, but at a very low rate when he is working in the office. This calls for dynamic changes in the settings for both the sensors and for the subsequent processing. Many adaptive approaches already exist [3][4][5][6].

A problem with these and similar solutions is that they have complex pipelines and/or require many parameters specific to the particular recognition problem. If we were to recognize different activities using different sensors we would have to adapt these parameters, requiring either a lot of experimentation or expert knowledge. It would therefore be useful to have a method that can be provided with a sensor-rich dataset, and it would be able to tell which sensors or which sensor setting to use in each context.

A relatively simple solution of the above problem can be found in the work of Yan, et al. [6]. They select the sensor and attribute settings based on the last classified activity. For each activity they find the setting that best recognizes it, by testing all the settings on their dataset, and then use it when that activity is detected. However, since the selection is made for each activity in isolation, the effect on the whole system can be unpredictable. To illustrate: an accelerometer is very good at recognizing walking and resting, while a GPS is very good at recognizing driving. However, if we have only the accelerometer active while walking, driving will never be detected and the sensor switch will never occur. To take such interactions into account, we would have to run an experiment that has a specific setting for each activity and switches between them in runtime. Since there

are many (activity, setting) combinations, this process might be prohibitively time consuming, possibly taking months on large datasets with many possible settings.

We propose a Markov chain model to simulate runtime settings switching and predict what would happen if we used a particular setting for a particular context. Using this model requires only a small amount of actual experimentation. Since the experimentation is a lot more time consuming than the simulation, many more combinations can be tried out and a better combination is therefore expected to be found with the same resources. In this paper we explain the proposed Markov chain and try it on a simple activity recognition dataset.

2. METHOD DESCRIPTION

Suppose we have a sensing system that works as follows: the user is in one of predefined contexts (for example, the context can be the current activity) and each context is associated with some setting. Settings can be which sensors are in use, or what sampling frequency or duty-cycle policy is active, or which feature set is used for classification, and so on. Using the current setting, we watch for a possible context change, and if it happens, we switch the sensor setting to the one assigned to the new context. For example: if we are sitting we need a lower sensor frequency, compared to when we are walking. To determine the optimal parameters for such a system, we might want to try every combination of assignments of reasonable settings to possible contexts. If we have c contexts and s settings, we would have s^c different combinations. Our goal is to make only s experiments and simulate the rest, gaining a drastic increase in efficiency.

We begin by selecting all reasonable settings. For each of them, we make an experiment where the classification model is trained and tested with this particular setting. For each experiment, we calculate and remember the confusion matrix. Additionally we need the transition probability from one state to the next one; that can easily be inferred from the dataset. Finally we must make energy consumption estimation for each setting. Energy consumptions for most sensors at different configurations are known or can be estimated with simple measurements.

To simulate an experiment where the settings are dynamically switched depending on the current activity, we create a Markov chain. This model has a state for each possible (context, setting) combination. The Markov state (c,s) represents that we are currently in context c, with the system having setting s (which depends on which context the system believes we are currently in).

Next we have to calculate the transition probability from one Markov state to another. They can be calculated from the transition probabilities of contexts and data from the previous computed confusion matrices. Intuitively: we get in a state (c,s) if the context really changes to c and if the system classifies this instance into one of the contexts that have assigned setting s.

$$S(c_1, s_1) \to S(c_2, s_2) = T(c_1, c_2) \sum_{c \in C(s_2)} C_{s_1}(c|c_1)$$

$S(c_1, s1)$ - the Markov state with context c and setting s.
$T(a, b)$ - the probability in the dataset that the next context will be b given that the current one is a.
$C(s)$ - the set of all contexts that use setting s.
$C_s(c_1|c_2)$ - the probability that the classifier that works with setting s will classify an instance to c_1, if the true context is c_2.

Having all transition probabilities, we can use the basic Markov chain calculus to calculate the steady state of the Markov chain. This gives us the amount of time the system will be in each of the states. Since we know how much time any setting is active and how much energy this setting consumes per time unit, the energy consumption of the whole system can be estimated. Additionally, since confusion matrices give us the accuracy for each state, we can calculate the accuracy of the whole system.

$$\text{Energy estimation} = \sum_{m \in M} t(m) e(s(m))$$

M - the set of all states in the Markov chain.
$t(m)$ - the predicted proportion of time spent in state m.
$e(s)$ - the energy requirement of a setting s in a given time unit.
$s(m)$ - the setting corresponding to state m.

$$\text{Accuracy estimation} = \sum_{m \in M} t(m) acc(c(m), s(m))$$

$acc(c, s)$ - the accuracy of the classifier that works with setting s, if the true context is c.
$c(m)$ - the context corresponding to state m.

It should be noted that many other metrics can be determined from such a model. Example: the accuracy for a particular activity or the latency of activity change detection. They can be used instead of the accuracy when evaluating the performance.

Every simulated experiment represents a possible trade-off between the accuracy and energy consumption. Simulating all the combinations, the Pareto optimal trade-offs can than be presented to the application designer, which - knowing the energy and accuracy requirements of the application - can then choose the ideal settings for it.

3. EXPERIMENTAL SETUP

We will demonstrate our method on a simple example. We have a dataset of accelerometer data generated by a smartphone, and we want to classify basic activities (Table 1) from it. We can do this by using two different settings: a high frequency sampling (50 Hz) and a low frequency sampling (2 Hz). Two extremes were chosen for simplicity. In practice we would perhaps also try other frequencies (10 Hz, 20 Hz etc...) and duty cycles. We assign one of the two frequencies

to each activity, generating 16 different combinations. All 16 combinations will be simulated by doing only 2 actual experiments. In this case we might expect that the best combination would be using the low frequency for the activity rest, and the high frequency for the other activities, but the trade-offs are not so clear in general.

The confusion matrices are in Table 2 and Table 3. States of the generated Markov chain are in Figure 1. Note that such a chain is in principle fully connected, including the connections of each state to itself.

s	w	r	c
62.7	21.1	10.4	5.8

Table 1: Proportion (in %) of each activity in the dataset. The values are in %. s - rest, w - walking, r - running, c - cycling

	s	w	r	c
s	97.6	1.4	0.0	1.0
w	5.6	86.2	5.5	2.7
r	0.7	10.0	89.1	0.2
c	22.9	16.4	0.0	60.7

Table 2: The confusion matrix using the high frequency. The values are in %.

	s	w	r	c
s	97.2	2.3	0.1	0.4
w	9.5	82.4	7.2	0.9
r	3.9	28.4	67.7	0.0
c	46.8	32.6	1.5	19.1

Table 3: The confusion matrix using low frequency. The values are in %. We can observe that the accuracy for rest does not change much compared to the high frequency case, while the accuracy loss when cycling is quite drastic.

We also did a simpler simulation in the spirit of Yan, et. al [6], where every activity was considered in isolation. In this case, the accuracy of the system was computed simply by computing the accuracy for each activity given its setting and then weighted by this activity's proportion in the dataset.

4. RESULTS

The method was evaluated in the following way. First all the simulation trade-offs were computed (both Markov chain simulation and the simple one). Then we ran the actual experiments we were simulating, switching classifiers and sampling frequencies during the runtime. All three sets of results were plotted in Figure 2.

The trade-offs in Figure 2 are marked with letters that correspond to the activities where the low frequency was used. The case where only rest was used with the low frequency (marked as 's') could be considered the best trade-off between the energy gained compared to the accuracy lost.

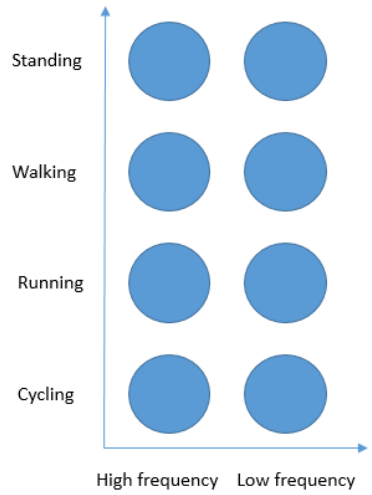


Figure 1: Markov chain states for our example. Vertical axis signifies the true activity, while the horizontal signifies the setting, which depends on the last classified activity.

We also plotted the Pareto front that shows the sensible solutions. We see that the Markov chain simulation points very closely correspond to the non-simulated ones. The simple simulation captures the general trend of the Pareto front, but makes substantial mistakes in predicting the actual values. If the interaction between sensors and activities were more complex, we expect the error to be be even greater. The error is numerically evaluated in Table 4.

	acc.	energy
Markov	2.03	0.35
Simple	12.82	1.83

Table 4: The average prediction mistake, made on energy consumption and classification error for the simple and Markov chain model. The values are given in % with respect to the maximal value for the corresponding axis.

We also explored what happens if the underlying activity distribution in the dataset changes. This can be easily simulated by modifying the transition probabilities of the Markov chains. It turns out that while the values themselves change drastically, the overall shape of the Pareto front remains similar. This means that the best simulated trade-off likely remains the best with most other activity distributions. Such simulation is also handy to see if the energy requirements exceed the application limits if some condition changes. Note that no additional experiments with the actual data were needed to generate this information, which is an additional benefit of our method.

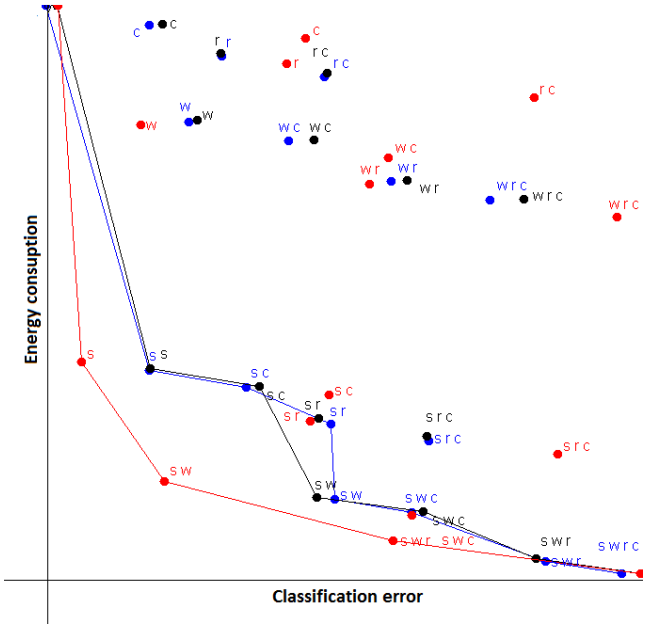


Figure 2: Black points are real trade-offs, blue points are simulated with Markov chains, red points are simulated with the simple model. The Pareto fronts are drawn using corresponding colors. The lower left corner represents the point with the lowest error and the lowest energy consumption. We can see that Markov chain simulation corresponds very closely to the values of real experiments.

5. CONCLUSION AND FUTURE WORK

The simulations display a very high decree of fidelity to the actual experiments and are very fast. Our method can thus effectively tackle the important but difficult task of selecting system settings that give us a good compromise between the accuracy and energy efficiency.

Future work on the topic will include testing the proposed method on a more complex dataset that contains more sensors and activities to see if the results still have the same fidelity. Another improvement will be to explore options for searching the settings combination space more efficiently. Since this is essentially a multi-objective optimization problem, many approaches from that area can be then used to further increase the effectiveness of our method.

6. REFERENCES

[1] Aftab Khan, Nils Hammerla, Sebastian Mellor, and Thomas Plötz. 2016. Optimising sampling rates for accelerometer-based human activity recognition. *Pattern Recognition Letters* (2016).

[2] Seon-Woo Lee and Kenji Mase. 2002. Activity and location recognition using wearable sensors. *IEEE pervasive computing* 1, 3 (2002), 24–32.

[3] Hong Lu, Jun Yang, Zhigang Liu, Nicholas D Lane, Tanzeem Choudhury, and Andrew T Campbell. 2010. The Jigsaw continuous sensing engine for mobile phone applications. In *Proceedings of the 8th ACM conference on embedded networked sensor systems*. ACM, 71–84.

[4] Jeongyeup Paek, Joongheon Kim, and Ramesh Govindan. 2010. Energy-efficient rate-adaptive GPS-based positioning for smartphones. In *Proceedings of the 8th international conference on Mobile systems, applications, and services*. ACM, 299–314.

[5] Yi Wang, Jialiu Lin, Murali Annavaram, Quinn A Jacobson, Jason Hong, Bhaskar Krishnamachari, and Norman Sadeh. 2009. A framework of energy efficient mobile sensing for automatic user state recognition. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*. ACM, 179–192.

[6] Zhixian Yan, Vigneshwaran Subbaraju, Dipanjan Chakraborty, Archan Misra, and Karl Aberer. 2012. Energy-efficient continuous activity recognition on mobile phones: An activity-adaptive approach. In *Wearable Computers (ISWC), 2012 16th International Symposium on*. Ieee, 17–24.

Reliability estimation of individual predictions for incremental models

Anže Javornik
univ. dipl. inž. rač. in inf.
Dragomelj 150
1230 Domžale
+386 41 550 515
anze.javornik@pointar.si

Igor Kononenko
Ph. D., Professor
Faculty of Computer and Inf. Sc.
Večna pot 113,1000 Ljubljana
+386 1 479 8230
xaigor@fri.uni-lj.si

Darko Pevec
PhD.
Faculty of Computer and Inf. Sc.
Večna pot 113,1000 Ljubljana
+386 1 479 8230
darko.pevec@gmail.com

ABSTRACT

Reliability estimate of an individual prediction can represent a very important information, especially if a wrong decision can have serious consequences. Same as classical prediction models, classical methods for reliability estimation are based on known examples – the learning set. Such methods cannot be used with incremental models, since this knowledge is not available. In this article we present three known methods of reliability estimation and propose their incremental versions. We tested incremental and classical methods using four incremental models on eight data streams with and without drift. Obtained results were statistically analyzed and compared.

Categories and Subject Descriptors

I.2.m [**Artificial intelligence**]: Miscellaneous

General Terms

Algorithms, Reliability.

Keywords

Incremental, Machine learning, Classification, Reliability estimate

1. INTRODUCTION

Task of machine learning is to construct a model, which is used for forecasting. The learning process uses known examples to adapt the prediction model in such a way that it is able to solve problems better. The result of learning process is knowledge which can be a set of remembered examples, an algorithm or a set of rules which describe the problem domain [1]. Such knowledge is called a model, which can be used for understanding the problem better or for predicting new examples.

The process of forecasting is closely linked to the concept of reliability. In situations when wrong prediction can have serious consequences, we would like to know how accurate the prediction is. However, for an unmarked example we cannot provide an exact accuracy but can only estimate it.

Classical prediction models and methods of reliability estimation are based on examples from a problem domain and so their knowledge is static. Often the problem domain changes over time. Consider the sensor readings whose sensitivity is decreasing. Since the knowledge is static and based on the data that was gathered from the sensor when it was still in a good operating form, the accuracy of the model and reliability estimate will drop. To battle such issues an incremental approach is needed.

2. INCREMENTAL LEARNING

Classical learning requires that enough data is available to construct knowledge. Incremental learning has no such requirements. In the process of incremental learning each new marked example is used to refine the current model.

2.1 Incremental Naive Bayes Classifier

Naive Bayes classifier makes prediction with an assumption that different attribute values with given class are conditionally independent. The classification formula is based on Bayes rule:

$$P(r_k|\,V) = P(r_k)\prod_{i=1}^{a}\frac{P(r_k \mid v_i)}{P(r_k)}$$

The task of learning process is to use the learning set and approximate probabilities on the right hand-side of equation. The knowledge of naive Bayes classifier is a table of approximations of prior probabilities of classes $P(r_k)$ and a table of conditional probabilities $P(r_k \mid v_i)$ of classes r_k with value v_i for attribute A_i.

In incremental learning the algorithm needs to refine prior probabilities and conditional probabilities. In [2] the adjustment formula for prior probabilities is defined as

$$P'(r_k) = \begin{cases} \frac{s}{s+1}P(r_k) + \frac{1}{1+s} & ;\ A_0 = r_k \\ \frac{s}{s+1}P(r_k) & ;\ A_0 \neq r_k \end{cases}$$

where

$$s = |D| + |T|$$

and for conditional probabilities

$$P(r_k, v_i) = \begin{cases} \frac{m}{1+m}P(r_k, v_i) + \frac{1}{1+m} & ;\ A_0 = r_k\ in\ A_i = v_i \\ \frac{m}{1+m}P(r_k, v_i) & ;\ A_0 = r_k\ in\ A_i \neq v_i \\ P(r_k, v_i) & ;\ A_0 \neq r_k \end{cases}$$

where

$$m = |A_i| + count(r_j)$$

where $|A_i|$ represents the number of values of attribute A_i and $count(r_j)$ represents the number of examples whose class is r_k.

2.2 Incremental Decision Trees

Decision tree is composed of internal nodes that correspond to attributes, branches that correspond to subsets of attribute values and leaf's that correspond to classes. A path from root to a leaf corresponds to one decision rule. The task of the learning process is to build a tree from the learning set in such a way, that for each current level it finds a most important attribute – one who splits the example set best – and use it as the root of current subtree.

Incremental algorithms are refining the decision tree for each new example. The goal of the algorithm for incremental learning of decision tree is to construct the same decision tree as non-incremental algorithm on the same set of examples. An example of incremental learning algorithm is ID5R, which for each new example finds most important attribute A_i and lifts it towards the root of the tree and refines the tree [3].

2.3 Incremental Bagging and Boosting

Traditional algorithms for supervised learning give their result according to the individual basic prediction model. Aggregated learning algorithms combine predictions of many such base models where each of them is learnt in traditional way. Bagging and boosting are well known aggregate algorithms for which it was shown that they provide better results compared to individual models [4].

Bagging is a method in which M different learning sets are generated using randomly selected examples from original learning set which can be repeated. Each generated learning set contains each example K-times, where

$$P(K = k) = \binom{N}{k}\left(\frac{1}{N}\right)^k\left(1 - \frac{1}{N}\right)^{N-k}$$

which is a binominal distribution. With $N \rightarrow \infty$ distribution of K approximates to Poisson(1) distribution.

This feature is utilized by incremental bagging. For each new example for each individual base model M the algorithm chooses this example $K \sim Poisson(1)$ times and refines the model M.

Boosting is a more complex process of base model set generation. Each base model $M_1, \ldots, M_n$ is learnt with weighted learning set. The weight is defined by the classification error of previous base prediction models.

Incremental learning for boosting uses Poisson sampling to approximate aggravating. The algorithm works as the algorithm for bagging with a difference that when a base model wrongly classifies an example the Poisson distribution parameter, bound to the example, is incremented for the next base model, else it is decreased [5].

3. RELIABILITY ESTIMATION

Since we do not know the actual result of an example we can only speculate about the prediction accuracy. The speculation is based on previously seen examples. In most cases we focus only on space near the current example – that is closest neighbors.

With incremental learning the space of known examples becomes unlimited which poses a problem since we do not have unlimited space nor the time to find the closest neighbors. In our work we used two different strategies to deal with space of known examples: sliding window and space adjustment.

The sliding window is a simple strategy. In this strategy we remember the new examples and forget old examples, always keeping a specific number in memory. Downside of this strategy is that it forgets previously seen examples and is thus sensitive to example distribution.

The space adjustment strategy which we developed in our work differs from the sliding window in a sense that it does not forget previously seen examples. Instead of tracking the space of specific examples (x_i, C_i) we track the space of examples' description (x_i', C_i'). Let

$$E = \{(x_1, C_1), \ldots, (x_n, C_n)\}$$

be an example space where each example x is defined with a class label C and a set of attributes

$$x = \{a_1, \ldots, a_n\}$$

A description of example space E is defined as (x_E', C_E') where

$$x_E' = \{a_1', \ldots, a_n', C'\}$$
$$a_i' = f(x_1.a_i, \ldots, x_n.a_i)$$
$$C_E' = f(x_1.C_i, \ldots, x_n.C_i)$$
$$f(a, b) = a + \lambda * (b - a)$$

Let

$$\mathbb{P} = \{(x_1', C_1'), \ldots, (x_k', C_k')\}$$

be a set of descriptions where x_a' is a description and C_a' the class of description. For each new example (x, C) we need to decide either we join this example into existing description or start building a new description with it. We based this decision on ratio between the closest and the furthest description with the same class label. If the ratio is smaller than some theta value we join it with the closest description of the same label, else we start building a new description. In case of building a new description, we also implemented the logic to reduce space size. If the number of total remembered descriptions exceeded the allowed amount, then in the whole space we find the closest two descriptions with the same class label and join them into one. Using the mentioned method on first hundred examples of SEA generator [6] the space contained only fifty-nine descriptions. On the left side of Figure 1 we see how the space would look like if we would remember every example and on the right side is the space using the space adjustment strategy.

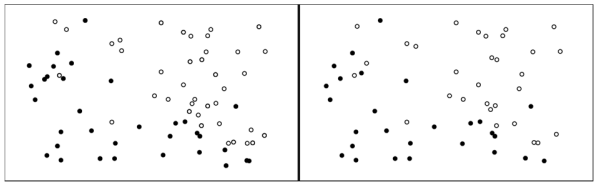


Figure 1. Distribution of examples in internal space

In our work we took three reliability scores (CNK, LCV, BAGV) from [5] and modified them by adding the incremental nature.

3.1 Incremental Local Error Score

Classical local error score is defined as follows: let K be the models' prediction of an unmarked example $(x, -)$ and $N = \{(x_1, C_1), \ldots, (x_k, C_k)\}$ the set of closest neighbors, where C_i is correct class label of the i-th example. The reliability estimate (CNK) is then the average distance between the class label of the closest neighbors and the models prediction

$$CNK = \frac{\sum_{i=1}^{k} f(C_i, K)}{k}$$

Incremental version (iCNK) for the example space management uses previously described strategy, the space adjustment. Instead of

using actual examples iCNK operates on closest descriptions $N = \{(x'_1, C'_1), \ldots, (x'_k, C'_k)\}$, and is defined as

$$iCNK = \frac{\sum_{i=1}^{k} f(C'_i, K)}{k}$$

where C'_i is the class label of the i-th description (Algorithm 1).

Algorithm 1. iCNK

Definitions:	N – max. number of descriptions
	τ – theta
	λ – adjustment rate
	$\mathbb{P}$ – description set
	k – num. of relevant closest neighbors
Input:	example with prediction (x, K)
Output:	reliability estimate

1. find set of k closest neighbors $\mathbb{P}_k$
2. calculate score $iCNK(\lambda, N) = \frac{\sum_{i=1}^{k} f(C_i, K)}{k}$
3. refine $\mathbb{P}$ with (x, K)
4. if $|\mathbb{P}| > N$
5. find descriptors $P_i, P_j \in \mathbb{P}$, for which $\min_{i,j} f(P_i, P_j)$ and $P_i.C = P_j.C$
6. in $\mathbb{P}$ replace P_i, P_j with $P_i + P_j$
7. end if
8. return score

3.2 Local Cross-Validation Score

The main idea behind LCV score is that it is independent from the models' prediction. From the set of closest neighbors $N = \{(x_1, C_1), \ldots, (x_k, C_k)\}$ it builds k models using cross validation "leave one out" and uses the model to predict the example left out. The score is then defined as

$$LCV = \sum_{i=1}^{k} \frac{f(C_i, K_i)}{k}$$

Incremental version of the algorithm (iLCV) uses the adjustment space strategy for managing examples and is thus operating on $N = \{(x'_1, C'_1), \ldots, (x'_k, C'_k)\}$ (see Algorithm 2)

Algorithm 2. iLCV

Definitions:	N – max. number of descriptions
	τ – theta
	λ – adjustment rate
	$\mathbb{P}$ – description set
	k – num. of relevant closest neighbors
	$\mathcal{M}$ – base incremental model
Input:	example with prediction (x, K)
Output:	reliability estimate

1. find set of k closest neighbors $\mathbb{P}_k$
2. for (x_i, C_i) in $\mathbb{P}_k$
3. construct model $M_i = new(\mathcal{M})$ on $\mathbb{P}_k \setminus (x_i, C_i)$
4. use model M_i and get prediction K_i for x_i
5. calculate distance $f(C_i, K_i)$
6. end for
7. calculate score $iLCV = \sum_{i=1}^{k} \frac{f(C_i, K_i)}{k}$
8. refine $\mathbb{P}$ with (x, K)
9. if $|\mathbb{P}^K| > N$
10. find descriptors $P_i, P_j \in \mathbb{P}$, for which $\min_{i,j} f(P_i, P_j)$ and $P_i.C = P_j.C$
11. in $\mathbb{P}$ replace P_i, P_j with $P_i + P_j$
12. end if
13. return score

3.3 Incremental Variance of Bagging Model

The aggregated prediction model, which is composed of m prediction models, provides its prediction as the average of individual predictions:

$$K = \sum_{i=1}^{m} \frac{K_i}{m}$$

The reliability estimate BAGV is defined as the variance of prediction distributions:

$$BAGV = \frac{1}{m} \sum_{i=1}^{m} (K_i - K)^2$$

The incremental version for space management uses a version of "sliding window" strategy. We define iBAGV on space $\mathcal{M} = \{M_1, \ldots, M_n, \ldots, M_{n+\max(1,\eta*n)}\}$, where M_i is an incremental model, η the rate of forgetting and n a predefined number of models. The position (order) of a model M in $\mathcal{M}$ is also important. In a sense we have converted a set of aggregated models in BAGV algorithm to a list and extended it to include additional $n + \max(1, \eta * n)$models. The models under indexes 1 to n are then used for calculating the score, while models under indexes n + 1 to the end are used for incremental learning. When it is time to implement the forgetting feature, we take first $m * \eta$ models from start of the list, re-initialize them and push them to the end of the list, forcing previous models towards start (see Algorithm 3).

Algorithm 3. iBAGV

Definitions:	$\mathcal{M}$ – prediction model list
	κ – adjustment rate
	η – forgetting rate
	n –number of models
	bagInit – is bag already initialized
	count – counter for applying forgetting
Input:	example with prediction (x, K)
Output:	reliability estimate

1. define relevant model set $\mathcal{M}^u = \{M_i\}$, where $M_i \in \mathcal{M}$ and $i \leq n$
2. for (M_i) in $\mathcal{M}^u$
3. calculate prediction K_i with M_i for x
4. end for
5. calculate score $iBAGV = \frac{1}{m}\sum_{i=1}^{m}(K_i - K)^2$
6. define learning set $\mathcal{M}^{learn} = \begin{cases} \mathcal{M}^u, & not\ bagInit \\ \mathcal{M}/\mathcal{M}^u, & bagInit \end{cases}$
7. randomly choose $\mathcal{M}^\kappa = \left|\mathcal{M}^{learn}\right| * (1 + \kappa)$ models with repeat from $\mathcal{M}^{learn}$
8. use (x, K) to learn models in $\mathcal{M}^\kappa$
9. increase count by 1
10. if $count \geq n$
11. replace first $n * \eta$ models from $\mathcal{M}$ with new and push them into the end of list $\mathcal{M}$
12. set $count$ = 0 and bagInit = True
13. end if
14. return score

4. TESTS AND RESULTS

We tested the developed algorithms and their classical versions on 4 incremental models (Naive Bayes, Hoeffding Tree, OzaBag and OzaBoost) from [7] using 8 generators from [7] with and without drift. We test the calculated scores against actual error by calculating the Person's correlation coefficient which we substantiated with statistical significance.

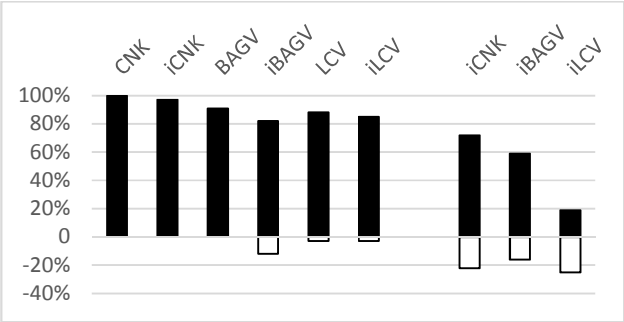


Figure 2. Results on generators without drift

The results of tests on generators without drift are presented in Figure 2. On the left side filled columns represent the percentage of tests where the reliability score had a positive correlation coefficient and was statistically significant and empty columns represent the percentage of tests where the score had a negative correlation coefficient and was statistically significant. The right side shows the comparison between incremental and classical versions of reliability scores. The filled column represents the percentage of tests where both scores had a statistically significant correlation coefficient and incremental version had a higher value of it, while the empty columns show where higher value of correlation coefficient was held by the classical version.

Results show that incremental scores and classical scores had about the same percentage of statistically significant positive correlation. The estimate iBAGV also shows a good percentage of statistically significant negative correlation. However, estimates iCNK and iBAGV in most tests provided a higher correlation coefficient. The estimate iLCV performed about as well as its classical version.

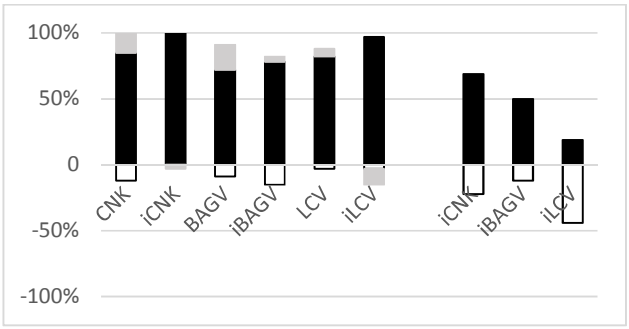


Figure 3. Results on generators with drift

Results on generators with drift are shown in Figure 3. Additional gray columns represent the change compared to generators without drift. We can notice a good drop in statistically significant positive correlations in classical estimates CNK and BAGV and the increase in statistically significant negative correlations in the incremental estimate iLCV. Incremental scores show a much lower drop in percentage of statistically significant positive correlations. The same as in tests on generators without drift, incremental scores iCNK and iBAGV in most tests, where both correlation coefficients are statistically significant, provide higher correlation coefficients than their classical versions. Also we can notice that unlike in test with no drift, all incremental scores have a higher percentage of statistically significant positive correlation.

5. REFERENCES

[1] I. Kononenko, M. Kukar. *Machine Learning and Data Mining*. Horwood (2007).

[2] S. Ren, Y. Lian, X. Zou, Incremental Naive Bayesian Learning Algorithm based on Classification Contribution Degree, *Journal of computers*, vol. 9, no. 8, August 2014: 1967 - 1974.

[3] Utgoff, P. E. (1989). Incremental induction of decision trees. *Machine Learning,* 4, 161-186.

[4] N. Oza, S. Russell, Online bagging and boosting, In *Artificial Intelligence and Statistics* 2001, 105–112, Morgan Kaufmann (2001).

[5] Z. Bosnić, Ocenjevanje zanesljivosti posameznih napovedi z analizo občutljivosti regresijskih modelov, doktorska disertacija, Fakulteta za računalništvo in informatiko, Ljubljana (2007).

[6] W. Nick Street, YongSeog Kim, A streaming ensemble algorithm (SEA) for large-scale classification, *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, 377-382 (2001).

[7] Albert Bifet, Geoff Holmes, Richard Kirkby, Bernhard Pfahringer (2010); MOA: Massive Online Analysis; *Journal of Machine Learning Research* 11: 1601-160.

Dve nadgradnji algoritma vzvratnega razširjanja

Bojan Ploj
School Centre Ptuj
Volkmerjeva cesta 19, 2250 Ptuj
Slovenia, Europe
+386 2 787 1 800
bojan.ploj@scptuj.si

Martin Terbuc
School Centre Ptuj
Volkmerjeva cesta 19, 2250 Ptuj
Slovenia, Europe
+386 2 787 1 812
martin.terbuc@scptuj.si

ABSTRACT
In this article two new algorithms for learning Feed-Forward Artificial Neural Network are presented. In the introduction, a brief description of the development of the existing algorithms and their flaws are shown. The second part describes the first new algorithm - Bipropagation. Basic idea is given first, followed by a detailed description of the algorithm. In the third part yet another new algorithm is given, called Border Pairs Method. Again is first given a basic idea and then follows a detailed description of the algorithm. In the fourth part the results and findings of experimental work are presented. In the conclusion, it is found that two described algorithms are fast and reliable - the second one is also constructive.

Categories and Subject Descriptors
• **Computing methodologies~Artificial intelligence** • *Computing methodologies~Supervised learning* • *Computing methodologies~Supervised learning by classification* • **Computer systems organization~Neural networks** • *Theory of computation~Design and analysis of algorithms* • *Theory of computation~Machine learning theory*

General Terms
Algorithms, Design, Reliability, Experimentation, Theory,

Keywords
Artificial Neural Network, Machine Learning, Supervised Learning, Bipropagation, Border Pairs Method, Multi-layer Perceptron

1. UVOD
Prvi modeli umetnih nevronskih mrež (ANN) so nastali pred več kot sedemdesetimi leti (Warren McCulloch in Walter Pitts) ter od takrat naprej niso doživeli korenitih sprememb. Izdelava novega modela je razmeroma preprosta, saj lahko nevrone tako rekoč poljubno povezujemo med seboj. Težji del naloge je najti algoritem, ki bo omogočal uspešno in učinkovito učenje modela. Eden prvih ANN modelov je bil Perceptron, njegov učni algoritem – Hebbovo pravilo [1] so odkrili šele čez nekaj let in je omogočal le reševanje preprostih (linearnih) problemov. Modeli in njihovi učni algoritmi so od takrat naprej postoma napredovali in so zmogli reševanje vse bolj zahtevnih problemov. Ta napredek se je dogajal v nekakšnih valovih, pri čemer so se izmenjevala obdobja počasnega in hitrega razvoja, podobno kot letni časi. Zatišna obdobja so bila tako izrazita, da so dobila celo svoje ime – zima umetne inteligence [2]. Razlog za to naj bi bil psihološke narave, kajti prebojna odkritja so privedla do vzhičenja raziskovalcev, ki so jim nato sledila obdobja razočaranja. Drugi val napredka je prinesel algoritem vzvratnega razširjanja [3] (angleško backpropagation), ki omogoča učenje večslojnih nevronskih mrež s povezavami naprej (FFANN) na osnovi gradientnega zmanjševanja. Najbolj znana med FFANN je večslojni perceptron (MLP), ki lahko rešuje tudi najbolj zahtevne probleme, če ima za to na voljo dovolj veliko število nevronov in slojev [4]. V praksi se MLP izkaže kot razmeroma zmogljiv inteligentni sistem, njegov učni algoritem pa ima žal številne pomanjkljivosti, zaradi česar se učenje pogosto konča neuspešno. Kadar so učni podatki nekoliko obsežnejši, se učni čas zelo podaljša, hkrati pa se za nameček še zmanjša uspešnost učenja.

Dodatna težava je še dejstvo, da ne poznamo optimalne oblike FFANN (število slojev in nevronov v posameznem sloju), kar je dokaj pomembno za dober učni rezultat. Premajhna FFANN namreč ne zmore iz podatkov posrkati vsega znanja, prevelika pa se nauči prekomerno. In ne samo to, prava sestava FFANN še vedno ni garancija za uspešno učenje. Med učenjem namreč zmanjšujemo učno napako na osnovi gradienta, zato se lahko zgodi, da zaidemo v lokalni minimum in tam obtičimo. Uteži – prosti parametri FFANN, ki jih med učenjem optimiziramo, so pri algoritmu vzvratnega razširjanja sprva izbrani naključno, kar pomeni, da je uspeh učenja odvisen tudi od sreče. Vse našteto nas je privedlo do iskanja alternativnih učnih algoritmov, med katerimi smo odkrili dva zanimiva.

2. BIPROPAGATION
Prvi novi algoritem se imenuje bipropagation [5, 6] in je manjša nadgradnja algoritma vzvratnega razširjanja. Osnovna ideja izboljšave je v tem, da začetne vrednosti uteži in želene vrednosti notranjih slojev ne prepuščamo več naključju kot pri vzvratnem razširjanju, ampak jih izračunamo oziroma določimo. Kako to naredimo bomo prikazali kasneje na primeru fotografij. Namenoma smo uporabili izraz »notranji sloj« in ne več »skriti sloj«, kot je to običaj pri vzvratnem razširjanju. Če namreč poznamo želene vrednosti notranjih slojev, se lahko lotimo učenja vsakega sloja posebej. Tako razbijemo zahteven problem na več manjših, ki jih dobro obvladamo, saj posamični sloji nimajo lokalnih minimumov. Edini še nerešeni problem je torej določanje smiselnih želenih vrednosti notranjih slojev. Z intuicijo najdemo rešitev ali dve tudi za ta problem. Najlažje pojasnimo to na primeru fotografij obrazov, kjer z MLP ugotavljamo spol fotografirane osebe. MLP ima toliko vhodov, kolikor je slikovnih točk (pikslov) in en binarni izhod, kjer velja moški = 0, ženska = 1. Zaradi lažjega razumevanja bodo vsi notranji sloji imeli enako število nevronov, kot je število vhodov oziroma pikslov. Sicer so ta števila lahko brez zapletov tudi drugačna. Prvi notranji sloj ima nalogo, da vse piksle osvetli za en odtenek, če je na fotografiji moški, oziroma potemni, če je ženska. Tako dobimo želene vrednosti prvega notranjega sloja, ki so zelo podobne vhodnim

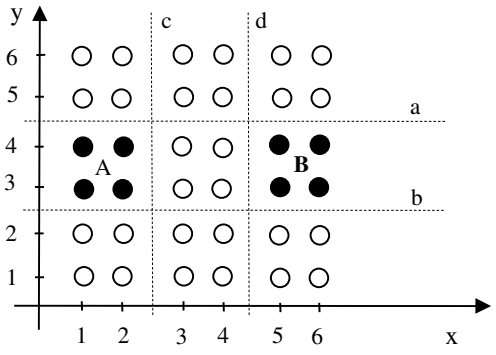
vrednostim. Ostane nam le še izbira uteži prvega sloja. Če bi slika v prvem notranjem sloju ostala enaka vhodni, bi za uteži ustrezala enotska matrika (nevroni imajo odsekoma linearno prenosno funkcijo). Ker je slika dejansko zelo podobna vhodni, je enotska matrika dobro izhodišče za izračun uteži in zato za dobro naučenost zadostuje že malo število učnih iteracij. Prav tako je zaradi male spremembe uteži med učenjem prvega sloja mala tudi verjetnost, da med učenjem naletimo na lokalni minimum. V višjih notranjih slojih se zgodba ponavlja, fotografija postane glede na spol še svetlejša oziroma temnejša in v zadnjem notranjem sloju postane čisto svetla ali temna. Izhodni sloj ima le en nevron, ki ima trivialno zadolžitev: če dobi na vhod množico samih ničel, da na izhodu ničlo, če pa dobi same enice, pa da enico. Ko zaključimo učenje posameznih slojev, lahko z metodo vzvratnega razširjanja po želji opravimo še fino uglaševanje. Posplošitev tega algoritma na razvrščanje med več razredi je trivialna. Če bi na primer razvrščali desetiške števke, ne bi temnili cele slike, ampak samo eno desetino. Za enico bi temnili prvo desetino pikslov, za dvojko drugo in tako naprej.

Glede hitrosti učenja je bipropagation v prednosti zaradi dveh razlogov. (1) Uteži so že pred učenjem podobne pravim. (2) Ker se učijo posamezni sloji, ni možno, da bi spremembe uteži v predhodnih slojih motile učenje v trenutnem sloju. Kljub vsem prednostim pa algoritem bipropagation ostaja iterativen, saj še vedno optimizira uteži z računanjem gradienta v številnih malih korakih. Demo program za algoritem Bipropagation je objavljen na spletu [10].

3. METODA MEJNIH PAROV

3.1 Analiza naučenega MLP

Drugi algoritem se imenuje metoda mejnih parov (Border Pairs Method, BPM) [6, 7]. Njegovo izhodišče je popolnoma drugačno kot pri algoritmu bipropagation. Najprej smo MLP naučili do te mere, da je čisto vse učne vzorce razvrstil pravilno, nato pa smo opazovali lastnosti notranjih slojev. Za učne podatke smo uporabili črno-belo fotografijo brez sivih odtenkov – slika 1. Vhodna podatka sta pikslova vrstica Y in stolpec X (dve celi števili od 1 do 6) izhodni podatek pa barva piksla (0 = bela, 1= črna). Med opazovanjem notranjih slojev smo dajali na vhod MLP



Slika 1. Razmejitev pikslov v črno-beli fotografiji

a, b, c in d so mejne črte; A in B sta področji

tudi realna števila (prav tako na intervalu od 1 do 6) in tako opazovali tudi navidezni prostor med piksli.

Ugotovitve so bile naslednje:

(1) Vsi nevroni, navkljub sigmoidni prenosni funkciji, delujejo v zasičenju. Razen v zelo ozkem obmejnem pasu imajo vedno izhodno vrednost enako nič ali ena.

(2) Nevroni v prvem sloju lokalno razmejujejo področja bele in črne barve.

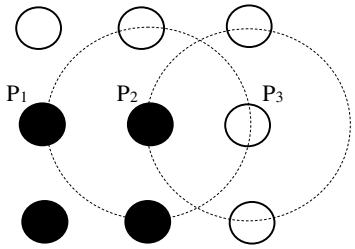
(3) Nevroni drugega sloja opravljajo logično operacijo IN tistih nevronov iz prvega sloja, ki omejujejo isto črno področje.

(4) Nevron tretjega sloja opravlja logično operacijo ALI nevronov iz drugega sloja.

3.2 Sinteza MLP

Za fotografijo na sliki 1 torej velja naslednje:

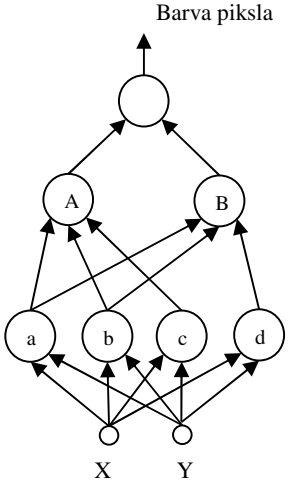
- 4 nevroni v prvem sloju (za vsako mejno črto a, b , c in d po eden)
- 2 nevrona za logično operacijo IN v drugem sloju (za vsako področje A in B po eden)
- 1 izhodni nevron za logično operacijo ALI



Slika 2. Iskanje mejnih parov

P_2 in P_3 sta mejni par, ker je njun presek krožnic prazen.

Zgradba MLP je s tem že določena (glej sliko 3.), nerešen nam ostane le še drugi del naloge – učenje oziroma iskanje optimalnih vrednosti uteži nevronov. V drugem in tretjem sloju se opravljata logični operaciji IN in ALI. Iskanje optimalnih vrednosti uteži zanju je trivialno. Torej smo z drugim in tretjim slojem že opravili



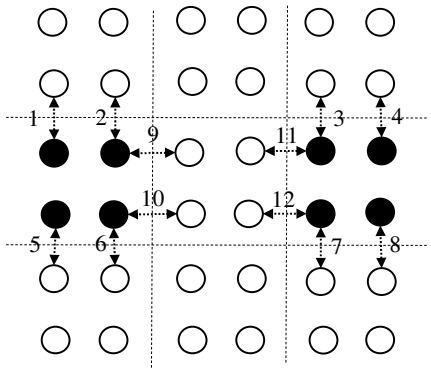
Slika 3. Zgradba optimalnega MLP

Nekatere sinapse v drugem sloju so izločene.

v celoti in nam ostane le še učenje nevronov v prvem sloju, ki določajo položaj posameznih lokalnih mejnih črt.

3.2.1 Določanje mejnih črt

Na sliki 1 so narisane štiri mejne črte (črtkane črte a, b, c in d) v dvodimenzionalnem prostoru (dimenziji X in Y). Te mejne črte so položene tako, da razdelijo celoten prostor na homogena področja. Področji A in B vsebujeta le črne piksle, preostala področja pa samo bele. Pri risanju mejnih črt izhajamo iz predpostavke, da mejne črte potekajo med črnim in belim piksli, ki so si blizu. V ta namen bližnji črno-bel par poimenujemo »mejni par«, pri čemer besedo »bližnji« definiramo s presekom krožnic. Krožnici narišemo tako, da je en piksel v središču kroga, drugi pa na krožnici. Če v preseku obeh krožnic ni tretje točke, potem rečemo, da sta si točki blizu, oziroma da mejita. Piksla P_2 in P_3 na sliki 2. torej tvorita mejni par. Če bi narisali krožnici za P_1 in P_3, bi ugotovili, da je P_2 v preseku njunih krožnic ter zato P_1 in P_3 nista mejni par. Slika 4. prikazuje vse mejne pare za našo črno-belo sliko. Skupaj je na sliki 12 mejnih parov, ki jih sedaj želimo ločiti s premicami. Gremo po vrsti in najprej ločimo par 1.



Slika 4. Mejni pari

Dvoglave puščice ponazarjajo mejne pare, ki jih sečejo mejne premice

Iz slike je razvidno, da lahko ista premica loči tudi pare 2, 3 in 4. Tako ubijemo 4 muhe na en mah in dobimo prvo vodoravno premico. Druga vodoravna premica loči pare 5, 6, 7 in 8. Z dvema navpičnima premicama ločimo še ostale štiri pare in ugotovimo, da so skupno potrebne štiri premice. Med postavljanjem premice učimo en nevron z udeleženci mejnih parov. Tako dobimo 4 nevrone v prvem sloju in s tem je znana (skoraj) minimalna zgradba celotnega MLP. Za računanje razdalje med točkami (polmer krogov) in za ugotavljanje ali par meji, uporabljamo zgolj evklidsko razdaljo, ki je določljiva v prostoru s poljubnim številom dimenzij. Tako lahko rešujemo probleme v poljubno dimenzionalnem prostoru. Krožnice v tridimenzionalnem prostoru nadomestijo sfere, v še več dimenzionalnem prostoru pa hipersfere. V redkih primerih se zgodi, da že ločimo vse mejne pare, a ostane kakšno področje še vedno nehomogeno. V tem primeru se lotimo iskanja dodatnih mejnih parov, a tokrat le še v nehomogenem področju. Tako smo med učenjem uporabljali le 20 od 36 pikslov. Učili smo le 4 nevrone od skupno 7 in še te 4 vsakega posamično. Posebej pomembno pri tem pa je, da ta algoritem v vsakem primeru najde rešitev. V primeru, ko so učni podatki več kot dvodimenzionalni, si vhodnega prostora ne moremo narisati in se zato postopek malo spremeni. Vse sloje določamo po enakem postopku, kot smo ga uporabili pri prvem sloju v dvodimenzionalnem problemu. Pri tem so izhodni podatki enega sloja vhodni podatki naslednjega. Vsak sloj ima enako ali manjše število nevronov od predhodnega. Ko ostane le še en nevron, je postopek končan.

Razen za učenje MLP so mejni pari primerni tudi za razšumljanje podatkov in ugotavljanje zahtevnosti učnih podatkov že pred pričetkom učenja. Razšumljanje z mejnimi pari poteka tako, da udeležence mejnih parov nekoliko premaknemo v takšni smeri, da jih približamo najbližjemu podatku istega razreda, ki ni udeležen v mejnem paru. Ostalih podatkov sploh ni potrebno korigirati, saj na učenje vplivajo samo mejni pari.

Zahtevnost učnih podatkov ugotavljam že pred pričetkom učenja iz deleža podatkov, ki so udeleženi v mejnih parih. Kadar so učni podatki nezahtevni, jih je le nekaj % udeleženih v mejnih parih. V nasprotnem primeru sta dve možnosti: velik šum v podatkih ali pa podatki skrivajo v sebi zapletena pravila. Lahko pa je tudi kombinacija obojega. Podatki v obravnavanem primeru so ekvidistančni, a eksperimenti so pokazali, da algoritem deluje enako dobro tudi pri poljubnem razporedu podatkov.

4. EKSPERIMENTALNO DELO

Prava vrednost vsakega izdelka se seveda pokaže med njegovo uporabo, zato smo oba algoritma testirali z različnimi učnimi podatki. Zaradi lažje primerljivosti z drugimi algoritmi in z drugimi inteligentnimi sistemi smo uporabili uveljavljene učne podatke. Začeli smo z logično funkcijo ekskluzivni ALI (XOR), ki je ena od najmanjših učnih množic, a je vseeno zanimiva zaradi izrazitega lokalnega minimuma. Oba algoritma sta se izkazala izvrstno. Vzvratno razširjanje je potrebovalo za isto nalogo petindvajsetkrat več iteracij kot algoritem bipropagation. Rezultati temeljijo na povprečju množice poskusov. BPM je brez težav našla dve mejni črti in izvedla eno logično operacijo IN v enem koraku. Ponavljanje poskusa ni potrebno, ker tu ni naključnih začetnih vrednosti, ki bi vnesle dejavnik sreče v poskus. Naslednja učna množica so bile Perunike (Iris) [8]. To so štiri dimenzionalni podatki, ki vsebujejo tri razrede in 150 učnih vzorcev. BPM je že pred pričetkom učenja izločila 146 od 150 učnih vzorcev in nalogo rešila z enim samim nevronom. Najzahtevnejši in najobsežnejši test je bil izveden z zbirko podatkov MNIST, ki obsega slike 70 tisoč rokopisnih števk v ločljivosti 28*28 slikovnih točk (784 dimenzionalni problem), v 8 bitni barvni ločljivosti (256 odtenkov sive barve) [9]. Metoda vzvratnega razširjanja tukaj popolnoma odpove, bipropagation pa najde rešitev v zgolj nekaj 10 iteracijah. BPM je bila preizkušena le z delnim naborom podatkov MNIST zaradi omejitev strojne opreme. Rezultat dosežen z delnim naborom MNIST je bil podoben rezultatom, ki jih drugi algoritmi dosegajo s celotnim naborom učnih vzorcev. Pri nobenem od obeh obravnavanih algoritmov nismo zaznali nobenih omejitev glede uporabnosti. Oba sta primerna za razvrščanje poljubnih podatkov.

5. ZAKLJUČEK

Uvodoma smo se seznanili s težavami, ki jih srečamo med učenjem nevronskih mrež tipa FFANN oziroma MLP: počasnost, nezanesljivost, nenatančnost in nekonstruktivnost. V ta namen sta nastala dva nova algoritma: »bipropagation« in »metoda mejnih parov« (BPM). Izkazalo se je, da oba nova algoritma odpravljata mnoge pomanjkljivosti vzvratnega razširjanja in ne prinašata nobenih novih. Tukaj izpostavljamo konstruktivnost in ne iterativnost algoritma BPM. Algoritem BPM je bil do sedaj

objavljen le v različici za razvrščanje, v razvoju pa je že tudi različica za regresijo. Zelo koristna za uveljavljanje obeh novih algoritmov bi bila izdelava programske opreme zanju. Bodisi njuna vključitev v uveljavljena programska orodja na področju strojnega učenja, ali pa izdelava njunih knjižnic v uveljavljenih programskih jezikih.

6. REFERENCE

[1] Hebb, Donald Olding, The Organization of Behaviour: A Neuropsychological Theory, 1949.

[2] AI winter, Wikipedija, https://www.wikiwand.com/en/AI_winter, 15. september 2016.

[3] Davida E. Rumelharta, Geoffreya E. Hintona in Ronalda J. Williamsa, Learning representations by back-propagating errors, Nature, October 1986.

[4] Irie, Miyake; Capabilities of three-layered perceptrons, IEEE International Conference on Neural Networks, 1988.

[5] B. Ploj, Bipropagation - nov način učenja večslojnega perceptrona (MLP), Proceedings of the Eighteenth International Electrotechical and Computer Science Conference ERK 2009, Slovenian section IEEE, pp 199-202, 2009.

[6] Bojan Ploj: Advances in Machine Learning Research, 3rd chapter, Nova Science Publishers, 2014, New York.

[7] Bojan Ploj, Milan Zorman, Peter Kokol: Border Pairs Method – Constructive MLP Learning Classification Algorithm, Adaptive and Intelligent Systems,Second International Conference, ICAIS 2011, Klagenfurt, Austria, September 6-8, 2011. Proceedings

[8] Iris data set, http://en.wikipedia.org/wiki/Iris_flower_data_set, 12. 1. 2013

[9] Yann LeCun, Corinna Cortes: yan.lecun.com/exdb/mnist, MNIST , Handwritten digit database

[10] Bojan Ploj: Bipropagation Demo App, https://www.mathworks.com/matlabcentral/fileexchange/55389-bipropagation, 10. 9. 2016.

Ebook Recommendations Based on Stylometric Features

Lovro Žitnik and Marko Robnik-Šikonja
University of Ljubljana, Faculty of Computer and Information Science
Večna pot 113, 1000 Ljubljana, Slovenia
lovro@zitnik.com, marko.robnik@fri.uni-lj.si

ABSTRACT

We present a recommender system for Slovene ebooks using stylometric features and machine learning. Recommender systems have become a standard component of modern web platforms and a popular way for users to find novel items or services. With an increasing number of available Slovene ebooks and growing popularity among users, we were motivated to build a tool that helps them find their next book. As a data set we use books in the ePub format, anonymized list of users from an e-library system and a set of transactions between books and users. The textual content of books is tokenized, POS-tagged and lemmatized. Numerical stylometric features of books are extracted and evaluated by unsupervised feature ranking methods SPEC and Laplacian score. The reduced feature set describing books and users is used in clustering and classification tasks, including one-class. Several variants of recommender algorithms are presented, based on both content and collaborative filtering. The automatic and user evaluations show that the content-based approach with stylometric features is feasible, however, the collaborative filtering gives better overall performance.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

Keywords

stylometric features, ebooks, recommender systems, machine learning, Slovene language, clustering, classification

1. INTRODUCTION

Recommender systems (RS) suggest an ordered list of items to users by predicting their relevance. In our case, an algorithm suggests a personalized list of recommended ebooks. The motivation behind building such a tool were an increased number of Slovene ebooks, growing number of users that demand easier decision-making and the fact that there was no available RS for Slovene ebooks.

Stylometry is defined as a statistical analysis of variations in a literary style for writers or genres. When the preferred literary style of a user can be determined, books of the same style could make usable recommendations.

The aim of our work is to test the hypothesis that stylometric features, derived only from the text, suffice for an operational content-based filtering (CB) recommender system. This approach could be used to solve *the cold-start problem*, where recommendations are required for a new user, and other situations with inadequate data for a collaborative filtering (CF). Using the CB we could also recommend books similar to only one designated book, which greatly reduces required input of new users.

For the CB we use only the content of ebooks to analyze the writing style of the authors, and do not use other metadata (e.g. information about authors, literary style, keywords etc). In pre-processing we use natural language processing techniques and apply statistical methods to obtain 92 numerical stylometric features for each book. The examples of raw features include average number of syllables in a sentence, the percentage of words in direct speech, the percentage of common and proper nouns, average concreteness of the words etc. Additionally, we compute several well-known readability metrics. We evaluate the importance of features using unsupervised measures Laplacian score and SPEC.

We perform a dimensionality reduction and clustering on the stylometric features of books to reduce noise and produce an input data set for five CB algorithms. The list of users and transactions are processed in the same way to produce two data sets as input for two CF algorithms. The first data set uses only profile data, whereas book rankings are added to the second.

In related work Fleischman et al [5] use natural language processing (NLP) techniques for recommendations, but do not use stylometry. Vaz et al [12] use writing style and negative examples to prototype a working recommender system.

The paper is based on [16] and consists of six Sections. In Section 2 we present stylometric features. Section 3 describes data sets used to train the algorithms described in Section 4. Section 5 contains the evaluation and comparison of developed approaches. In Section 6 we summarize the conclusions and present guidelines for further work.

2. STYLOMETRIC FEATURES

Stylometric features (SF) are numerical descriptions of a book's writing style. They are derived from the content of a book. In our approach we tokenize books into chapters, paragraphs and sentences. Stopping at the sentence level is necessary to preserve the context of each separate word for a more precise classification. We perform the lemmatization

and POS-tagging, where each word or punctuation is classified as an adjective, verb, adverb, comma etc. Publicly available Slovene document corpus ccGigafida [4] is used for learning. We tested several taggers for speed and accuracy and choose Brill tagger [1].

From the pre-processed data we derive different SF. Simple features include percentage of sentences with different number of commas, number of punctuation marks, average number of syllables per word or sentence etc. More advanced features include overall percentage of the quoted words or number of sentences with direct speech. A special care was taken to preserve pronouns during the analysis. They are usually removed from the analysis via stop-words lists, because they are not relevant for analysis based on content. Pennebaker [10] advocates that pronouns play a vital role in determining writing style. Our analysis confirms this, as the feature "Percentage of pronouns"is ranked third by the Laplacian score [6] and fifth by the SPEC algorithm [14] in a list of total 92 stylometric features. The scores of all SF are available in [16].

We compute several readability metrics derived for English language as described by Mailloux et al [8]. While their transfer to Slovene language might be questionable, they were used previously [15] and our tests in Table 1 show that they are strongly correlated to the text comprehensibility. The average word concreteness and average age of word acquisition are also used features, based on [7] and [2].

Table 1. Average readability metrics for different book categories. Columns report the Gunning fog index (GFI), Flesch reading-ease score (FRES) and Flesh-Kincaid grade level (FKGL).

category	sample size	GFI	FRES	FKGL
E-books for Kids	81	7,26	39,76	10,47
E-books for Teens	25	8,41	33,63	11,68
Romance	69	7,91	35,68	11,15
Poetry and Drama	161	8,22	34,21	11,80
Natural Science, Technology, Math	5	12,04	10,88	15,93
Humanities and Social Sciences	96	13,17	10,13	16,63

3. DATABASE

To form stylometric data sets we used 1047 digitalized ebooks in ePub format. We obtained anonymized user profiles of 10,813 users of a Slovene e-library system and 2,151,834 transactions between users and books. We extracted and consolidated the data into a unified MongoDB database. The process is depicted in Fig. 1 and results in three data sets.

As a result of the feature extraction process, we obtain the data set with 92 numerical stylometric features for each book.

The "user profile" (UP) consists of a gender, year of birth, education and current status of each user. To impute missing values for some users, we use model-based imputation based on naive Bayes classifier.

The biggest challenge was to extract positive and negative book ratings from the transactions. In recommender systems usually explicit user ratings of items are available, where each user rates some items with a score (e.g., from

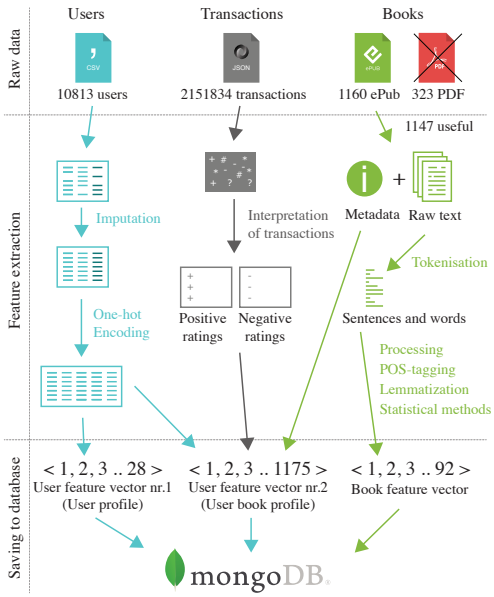


Figure 1. The diagram of feature extraction process.

1 to 5). In our case the ratings are implicit and have to be derived from users' interaction with the e-library web application. The transaction "Confirm rental" was used as the best approximation for positive rating of the book. The best candidate for negative rating was the transaction "View of the book page" occurring only once, without any further interactions with the same book.

The ratings for all books were added to the database for all the users for the purpose of CF. The attributes for all the books contain the value 1 for the positive rating, value -1 for the negative rating, and value 0 if no rating was extracted from transactions. This data set is called "User book profile" (UBP).

For each of the three data sets (books stylometric features, user profile and user book profile) we computed several derivative data sets with dimensionality reduction techniques, such as PCA and t-SNE. The DBSCAN and k-means algorithms were used for clustering, where the number of centers k was determined by optimization of Silhouette score. The visualization of one such end result is available in Fig. 2.

4. RECOMMENDER SYSTEM

Using the data described in the previous Section seven different recommendation algorithms were developed: five content-based filtering (CB) algorithms and two collaborative filtering (CF) algorithms.

We assume that the ratings extracted from the implicit user transactions are of two significantly different quality levels, i.e., positive ratings as more reliable than negative ratings. We describe the tested algorithms using different approaches and different subsets of data below.

4.1 Content-based filtering

The **average book algorithm (CNTR)** calculates the the average book (i.e. the centroid) from the positively rated books of a user and uses nearest neighbor techniques proposed by Sarwar et al [11] to find recommended books.

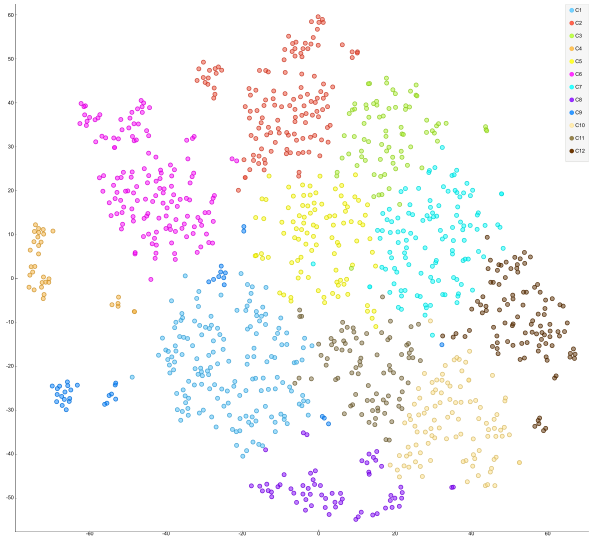


Figure 2. Clustering of books with k-means (k=12, Silhouette optimized) in 10-dimensional space, additionally reduced to 2-dimensional space with t-SNE.

The **nearest books algorithm (NEAR)** loops through the list of positively rated books, adding the next nearest book to the list of recommended books.

The **binary classification algorithm (BIN)** is the only CF algorithm that uses the negative ratings. The positive and negative ratings of books for a user are used as a training data for probabilistic binary classifier, which is used for classification of new books. The returned probability scores are used to sort the list.

The **one-class SVM algorithm (SVM)** is a modified version of Support Vector Machine, using learning from only one class. We discard the negatively rated books and only learn from positively rated books. We use the resulting classifier on a new unrated set of books. This approach is described by Manevitz et al [9] and used by Yu et al [13], where it is reported to have a poor performance.

The **PU algorithm (PU)** uses two sets of books from a particular user, the positively rated books and the unrated books. From the set of unrated books it tries to single out the books into the negatively rated books set. In this way the problem is transformed into a binary classification problem. we use the algorithm proposed by Elkan et al [3].

4.2 Collaborative filtering

CF algorithms do not use stylometric features. Instead they rely on the history of users' transactions. We form two classifiers.

The **user profile only algorithm (CFUPO)** first clusters users based only on their user profile, then it finds the nearest user in the same cluster and recommends the books of that user, removing the books already rated positively to avoid duplication.

The **user book profile algorithm (CFUBP)** differs from the CFUPO only by using the user book profile.

5. EVALUATION

We used two types of evaluation: **offline testing** and **user preference** testing.

With **offline testing**, we evaluated the prediction accuracy of algorithms by using the existing data. 20% of positively rated books were hidden and the other 80% is used for learning (i.e. input to recommendation algorithms). By measuring how many books from the hidden set the recommendation algorithms managed to retrieve we evaluate their precision and recall.

The distribution of positively rated books has a strong long-tail characteristics (see Fig. 3). To avoid imbalanced distribution, we divided users into 10 groups (G1 - G10), depending on a number of positively rated books (2, 3, 4, 5, 7, 10, 18, 28, 50, or 100) and tested each group separately.

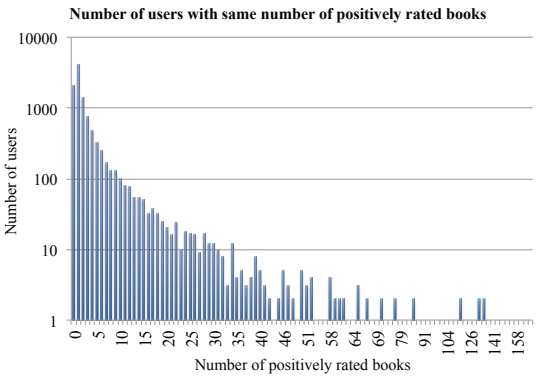


Figure 3. Number of users (logarithmic scale) in sets with same number of positively rated books.

As a baseline, we included a random book recommendation (RAND). All seven algorithms were tested with several different settings (different type and level of dimensionality reduction, metrics, classifiers etc.). The precision scores for the best settings of each algorithm are presented in Fig. 4 and 5. For details, see [16].

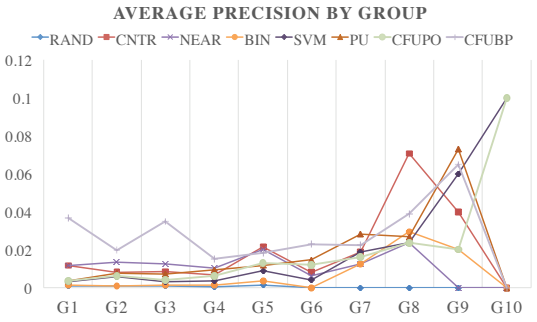


Figure 4. Average precision by group.

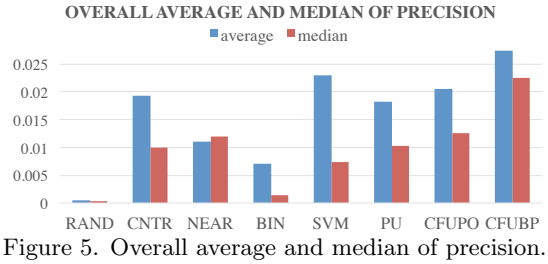


Figure 5. Overall average and median of precision.

The results show that all the algorithms outperformed the baseline random algorithm. CFUPB, which uses user book profile, scores best, followed surprisingly by SVM, which contradicts low expectations expressed in [13]. We assume that the worst score for BIN is caused by lack of data for training, whereas other CB algorithms score surprisingly well compared to CF.

In evaluation of **user preferences** we assigned 9 users a task to select 10 books they like and 5 they do not like. The age, birth year, education and current status of the tested persons were also collected. WE supplied this input to all 8 recommendation algorithms and assembled a list of recommended books for each user. A visual user interface to rank algorithms from 1 to 8 was provided. The first three recommendations were marked with gold, silver and bronze medals. The average rank and its standard deviation are presented in Fig. 6.

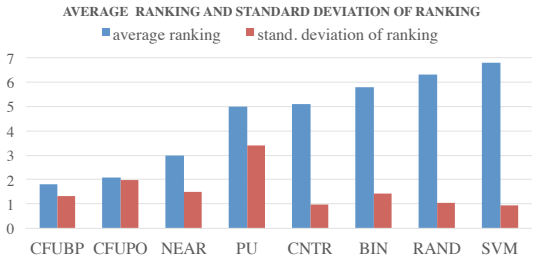


Figure 6. The algorithm ranking by users.

The results show that users prefer CF recommendations, which took first two places, before all CB algorithms. Surprisingly, the random algorithm ranked second to last. This could be attributed either to the users desire for novel books or to a noise, since users reported that ranking lower ranked suggestion was difficult.

6. CONCLUSIONS

We propose the content-filtering recommender system for books using only stylometric features. We propose several content-based filtering algorithms using only numerical stylometric features extracted from book content. We benchmark them against two collaborative filtering algorithms and random recommendations.

The results of both automatic evaluation and user preferences indicate that recommender systems using only stylometric features are feasible, but they are inferior in performance compared to collaborative filtering recommendations.

We assume that a significant improvement of content-filtering algorithms could be obtained by gathering explicit ratings of books. Incorporating the word-space similarity as described in [5] could also prove beneficial. For a real-world book recommender system we recommend development of a hybrid algorithm that combines both content-based and collaborative filtering approaches. This could solve the cold start problem.

7. REFERENCES

[1] E. Brill. A simple rule-based part of speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, ANLC'92, pages 152–155, 1992.

[2] M. Brysbaert, A. B. Warriner, and V. Kuperman. Concreteness ratings for 40 thousand generally known English word lemmas. *Behavior research methods*, 46(3):904–911, 2014.

[3] C. Elkan and K. Noto. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 213–220, 2008.

[4] T. Erjavec and N. Logar Berginc. Referenčni korpusi slovenskega jezika (cc)Gigafida in (cc)KRES. In T. Erjavec and J. Žganec Gros, editors, *Zbornik Osme konference Jezikovne tehnologije*, pages 57–62, Ljubljana, 2012.

[5] M. Fleischman and E. Hovy. Recommendations without user preferences: A natural language processing approach. In *Proceedings of the 8th International Conference on Intelligent User Interfaces*, IUI '03, pages 242–244, 2003.

[6] X. He, D. Cai, and P. Niyogi. Laplacian score for feature selection. In Y. Weiss, B. Schölkopf, and J. C. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 507–514. 2006.

[7] V. Kuperman, H. Stadthagen-Gonzalez, and M. Brysbaert. Age-of-acquisition ratings for 30,000 English words. *Behavior Research Methods*, 44(4):978–990, 2012.

[8] S. L. Mailloux, M. E. Johnson, D. G. Fisher, and T. J. Pettibone. How reliable is computerized assessment of readability? *Computers in nursing*, 13:221–221, 1995.

[9] L. M. Manevitz and M. Yousef. One-class SVMs for document classification. *J. Mach. Learn. Res.*, 2:139–154, Mar. 2002.

[10] J. W. Pennebaker. *The Secret Life of Pronouns: What Our Words Say About Us*. Bloomsbury USA, 2011.

[11] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Application of dimensionality reduction in recommender system-a case study. Technical report, DTIC Document, 2000.

[12] P. C. Vaz, D. Martins de Matos, and B. Martins. Stylometric relevance-feedback towards a hybrid book recommendation algorithm. In *Proceedings of the Fifth ACM Workshop on Research Advances in Large Digital Book Repositories and Complementary Media*, BooksOnline '12, pages 13–16, 2012.

[13] H. Yu, W. Zuo, and T. Peng. A new pu learning algorithm for text classification. In *Mexican International Conference on Artificial Intelligence*, pages 824–832, 2005.

[14] Z. Zhao and H. Liu. Spectral feature selection for supervised and unsupervised learning. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 1151–1157, 2007.

[15] A. Zwitter Vitez. Ugotavljanje avtorstva besedil: primer "trenirkarjev". In T. Erjavec and J. Žganec Gros, editors, *Zbornik Osme konference Jezikovne tehnologije*. Ljubljana, 2014.

[16] L. Žitnik. Recommender system for slovene electronic books. University of Ljubljana, Faculty of Computer and Information Science, 2016. Diploma thesis.

Model selection on the JSI grid: Metis use-case

Jernej Zupančič
"Jožef Stefan" Institut
Jamova cesta 39
Ljubljana, Slovenia
jernej.zupancic@ijs.com

Damjan Kužnar
"Jožef Stefan" Institut
Jamova cesta 39
Ljubljana, Slovenia
damjan.kuznar@ijs.com

Matjaž Gams
"Jožef Stefan" Institut
Jamova cesta 39
Ljubljana, Slovenia
matjaz.gams@ijs.com

ABSTRACT

The paper presents model selection using a random search method on the Jožef Stefan Institute (JSI) grid infrastructure. Best practices for model selection on the grid are presented through the use-case, where the dataset from the Metis project was used. The model selection task was to find the best learning algorithm and its parameters according to the area under the curve performance metric. Although the gradient boosting algorithm performed best in the experiment, logistic regression was chosen for the implementation in the Metis prediction module due to the high model transparency and good performance.

1. INTRODUCTION

The objective of a learning task $\mathcal{T} = (\mathcal{L}, \mathcal{D})$ is to find an algorithm $\mathcal{A}$ and the algorithm parameters λ so that the function $f = \mathcal{A}_\lambda(\mathcal{D})$, which is the result of the use of training algorithm on a dataset, minimizes some expected loss $\mathcal{L}(x; f)$ over i.i.d. samples x from a ground truth distribution $\mathcal{G}_x$. Algorithm $\mathcal{A}$ is a functional that maps a dataset $\mathcal{D}$, which is a finite set of samples from $\mathcal{G}_x$, to f. Algorithm $\mathcal{A}$ usually performs an optimization of some loss metric with respect to parameters θ that correspond to the internal representation of f by $\mathcal{A}$ (e.g. weight values in a neural network). Additionally, the algorithm $\mathcal{A}$ often has adjustable parameters λ called *hyper-parameters* (e.g. number and sizes of hidden layer in a neural network). Those hyper-parameters often determine the performance of the algorithm and their setting is crucial for obtaining an algorithm that effectively learns a function f on the training dataset $\mathcal{D}$ and performs well on new and previously unseen samples from $\mathcal{G}_x$. Therefore, an algorithm and its settings that minimize generalization error

$$E_{x\sim\mathcal{G}_x}\left[\mathcal{L}(x; \mathcal{A}_\lambda(\mathcal{D}))\right]$$

are required. The problem of finding the best model and its parameters with respect to the generalization error is called *model selection*, while the problem of finding only the best parameters for a chosen algorithm is called *hyper-parameter tuning*. This paper presents methods (and an application of one such method) to address the model selection problem:

$$\mathcal{A}^*, \lambda^*_{\mathcal{A}} = \text{argmin}_{\mathcal{A}\in A, \lambda\in\Lambda} E_{x\sim\mathcal{G}_x}\left[\mathcal{L}(x; \mathcal{A}_\lambda(\mathcal{D}))\right]. \quad (1)$$

The optimization problem 1 is hard due to several reasons:

1. The problem is usually non-convex.
2. Search space is complicated and hierarchical.
3. Gradients are usually not available.

Additionally, in real life scenarios one does not have the access to the ground truth distribution $\mathcal{G}_x$, therefore, the value that we wish to optimize, i.e. expectation over the unknown distribution, is impossible to evaluate. In order to mitigate the problem of $\mathcal{G}_x$s unavailability, cross-validation is often used. In cross-validation the expectation is replaced with a mean loss over validation set $\mathcal{D}_{\text{valid}}$. Cross-validation is unbiased as long as $\mathcal{D}_{\text{valid}}$ is independent of the data used by $\mathcal{A}_\lambda$ [4]. Therefore, the problem 1 is approximated by the problem

$$\mathcal{A}^*, \lambda^*_{\mathcal{A}} \approx \text{argmin}_{\mathcal{A}\in A, \lambda\in\Lambda} \text{mean}_{x\in\mathcal{D}_{\text{valid}}} \mathcal{L}\left(x; \mathcal{A}_\lambda(\mathcal{D})\right). \quad (2)$$

Since the problem 2 typically involves an evaluation of several $(\mathcal{A}, \lambda)$ pairs, which in turn usually require an inner optimization run, the problem is computationally intensive. The use of a cluster of computers is therefore required in order to obtain satisfiable solutions in a reasonable time frame. JSI grid[1] was used in the presented experiments.

The rest of the paper is structured as follows. In section 2 related work in model selection and hyper-parameter tuning is presented, in section 3 the experimental setup is described, in section 4 the results from the experiment are presented, and section 5 concludes the paper.

2. RELATED WORK

Model selection and hyper-parameter tuning problems are usually addressed by *manual search* and *grid search* or a combination of the two (e.g. [10, 6]). Although several model and hyper-parameter optimization methods have been introduced in the recent years such as *sequential model based optimization* [7, 3] or *efficient global optimization algorithm* [8, 1] researchers and practitioners still prefer a combination of manual and grid search. This combination does have some nice properties:

1. It enables the practitioner to build the intuition about the performance of the models and their parameters.
2. It enables the practitioner to focus on models and settings that are most promising.
3. It is effective in lower dimensions.

[1] https://www.ijs.si/ijsw/nsc/Uporaba%20gru%C4%8De%20IJS

4. Grid search is simple to implement and embarrassingly parallelizable.

However, it has the disadvantage of not being reproducible and not being effective in the cases of large search spaces and higher dimensions.

State-of-the-art approaches for model selection and hyper-parameter tuning are based on sequential model based optimization approaches and have already been implemented in several software libraries such as AutoWEKA[2], AutoSklearn[3], Spearmint[4], Hyperopt[5], etc. The problem with the sequential model based optimization methods is that they are not simple to parallelize and are difficult to implement. In the parallel implementation the authors usually choose a trade-off between effective modelling of the problem and the time speed-up enabled by the use of the cluster [3].

Although not state-of-the-art any more the *random search* is another method for model selection and parameter tuning which performs very well on high dimensional problems [2]. Additionally, random search is simple to implement and parallelize, while being much more efficient than a grid search.

3. EXPERIMENTAL SETUP

Random search method for model selection was implemented for a use in a multi-node setting. Each worker ran independently and it followed the following steps:

1. Set-up an environment on a computational node.
2. Sample from available algorithms and corresponding parameters.
3. Evaluate the performance of a chosen model.
4. Report the performance of the model to the central node.
5. Repeat from step 2.

Learning algorithms available in established Python 3.5 libraries Scikit-learn [11] (logistic regression, random forest, naive bayes, nearest neighbours, support vector classifier) and XGBoost [5] (tree boosting) were selected as possible algorithms. MongoDB[6] was used to store the results. The following parameter options (according to the algorithm implementation) were available:

- *LogisticRegression:*
  - `penalty` $\in \{'l_1', 'l_2'\}$
  - `C` $\in \{0.001, 0.01, 0.1, 1, 10, 50, 100\}$
- *RandomForestClassifier:*
  - `n_estimators` $\in \{120, 300, 500, 800, 1200\}$
  - `max_depth` $\in \{5, 8, 15, 25, 30, None\}$
  - `min_samples_split` $\in \{1, 2, 4, 10, 15, 100\}$
  - `min_samples_leaf` $\in \{1, 2, 5, 10\}$
  - `max_features` $\in \{'log2', 'sqrt', None\}$
- *KNeighborsClassifier:*
  - `n_neighbors` $\in \{'l_1', 'l_2'\}$
  - `p` $\in \{0.001, 0.01, 0.1, 1, 10, 50, 100\}$
- *SVC:*
  - `C` $\in \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$
  - `gamma` $\in \{'auto', 0.01, 0.05, 0.1, 0.2\}$
  - `class_weight` $\in \{'balanced', None\}$
- *XGBClassifier:*
  - `learning_rate` $\in \{0.01, 0.025, 0.05, 0.1\}$
  - `n_estimators` $\in \{120, 300, 500, 800, 1200\}$
  - `gamma` $\in \{0.05, 0.3, 0.7, 1.\}$
  - `max_depth` $\in \{3, 6, 12\}$
  - `min_child_weight` $\in \{1, 3, 7\}$
  - `subsample` $\in \{0.6, 0.8, 1.\}$
  - `colsample_bytree` $\in \{0.6, 0.8, 1.\}$
  - `reg_lambda` $\in \{0.01, 0.025, 0.05, 0.075, 0.1, 1.\}$
  - `reg_alpha` $\in \{0, 0.1, 0.25, 0.5, 0.75, 1.\}$

The search space includes 235 538 possible combinations. Taking into account the time required to build the model in the case of random forest or tree boosting (about 1 hour as observed from the log files) a full grid search was not feasible.

3.1 The data

The data about student performance analyzed in the Metis project was used to provide a use-case. The data includes the same features as presented in [9], however, the information from more students was added. In the use-case the information of 70 155 unique students at four subjects (Mathematics, Slovene, English and Chemistry) was used, which translates to 1 736 979 training instances. Each instance represents a state of the pupil at the end of a month (e.g. October) at one subject (e.g. Mathematics). This should result in 10 instances for one pupil for one subject per a school year, however, since the data for December was not available only 9 instances per school year were available. A total of 32 features was constructed taking into account the grades in current and previous grading periods and simple statistics derived from the grades (min, max, mean, deviation, difference), the number of excused and not excused absences, time remaining to the end of the grading period, and the number of times a pupil was missing from the examination. The goal was to predict whether a pupil will have a negative grade at the end of the grading period. The dataset included 114 414 of positive instances (i.e. student had a negative grade at the end of the grading period) and 1 622 565 of negative instances (i.e. student had a positive grade at the end of the grading period). Due to sensitive personal information the data is not publicly available.

[2]http://www.cs.ubc.ca/labs/beta/Projects/autoweka/
[3]http://automl.github.io/auto-sklearn/stable/
[4]https://github.com/HIPS/Spearmint
[5]http://hyperopt.github.io/hyperopt/
[6]https://www.mongodb.com/

3.2 Model evaluation

We have used an area under the curve (AUC), standard metric for the evaluation of binary classification models, with few adjustments. To evaluate a model, instances were first divided by month. The 5-fold cross validation was used on the instances of the same month to get a cross validation probability prediction for every instance. Using the predicted probability for every instance AUC metric was calculated for every month, which was the result of the model evaluation.

3.3 The grid

JSI computing cluster, which is a part of SLING - Slovenian initiative for national grid, is accessed through the membership at `nsc.ijs.si` virtual organization. The cluster enables the submission of batch jobs using the Advanced Resource Connector (ARC)[7] middleware and has 1 984 CPUs available for computational tasks. Every researcher also has the right to be a member of the `gen.vo.sling.si` virtual organization used for submitting jobs to common Arnes cluster, however, this cluster is usually occupied and the times for the jobs to start and eventually finish are much longer than using the JSI cluster.

4. RESULTS

4.1 Metis use-case results

The experiment was run on 20 computational nodes in the grid for 24 hours. We have obtained 112 model evaluations. The reason for such a low number of successful model evaluations was the failure of computing nodes on the grid, which will be explained in more details in section 4.2. As described in 3.2 evaluation values were obtained for every month during the school year. In order to compare between them a mean of those values was calculated for each model evaluation score. It was observed (Figure 1) that XGBClassifier was performing best no matter the parameter values used (mean score of 0.939 AUC), followed by RandomForestClassifier (mean score of 0.937 AUC) and LogisticRegression (mean score of 0.935 AUC). All results are presented in Figure 2. GaussianNB model was evaluated only once, since it does not require the setting of parameters, while SVC model was also evaluated only once, however, that was due to the fact that all other attempts to evaluate the SVC model on the given data resulted in error.

Although the XGBClassifier was performing best, we have chosen LogisticRegression for the use in the Metis prediction module, since it is a transparent model and it performs similar to the best performing XGBClassifier (when parameters are set to the values found in the search that perform best in case of LogisticRegression - `penalty` $=' l_1'$, `C` $= 1$).

We have also ranked the models according to the Pareto front dominance (Figure 3). Although the first front does present some valuable insight (only XGBClassifier and RandomForestClassifier appear on the first front and are therefore non-dominated) the information from other second and other fronts does not provide additional information, since the fronts strive for diversity.

[7]http://www.nordugrid.org/arc/about-arc.html

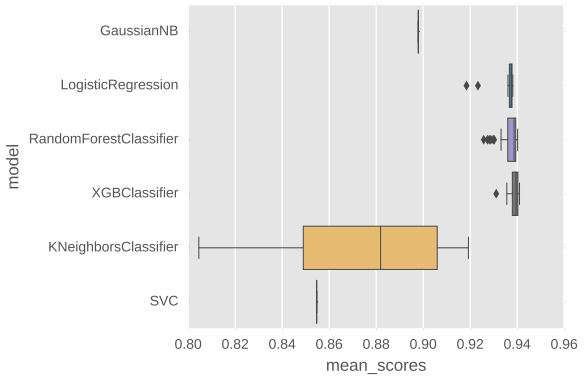


Figure 1: A box-plot of scores grouped by a model.

4.2 The grid experience

Although the availability of the grid is a big bonus for the JSI researchers, it still has some issues that have to be taken into account when writing software that is to be run on the grid.

The connection between nodes in a grid is not available (at least through Python and SSH). This makes the deployment of software designed for multiple nodes not possible.

When an error occurs in a job the whole job is cancelled. Even when multiple processes are run on one node, it the program in the process results in an error, the whole job is terminated.

Only batch jobs are possible. While some problems are appropriate for a batch job grid usage type, others are not. Especially the data mining problems, where the practitioner tries several approaches and requires instant feedback in order to proceed with the next step.

The following best practices are proposed when working with the JSI grid:

Parallelize your code so that each worker can work on independent task. It would be best if no communication between workers is required, the only communication would be the reporting of results to your local database, which is accessible to the grid inside the JSI. This will enable you to launch as many workers as you wish.

Use only one CPU per worker/job. In the case of error you will loose only one computation node.

Use local testing environment for testing the scripts you are about to submit as a job to the grid. Best, use container technologies (e.g. Docker[8]) to start a fresh Linux environment (as used by the grid), transfer all the required scripts and data to the environment and perform a test (e.g. in a case of data mining task a test could be done with smaller amount of data).

[8]https://www.docker.com/

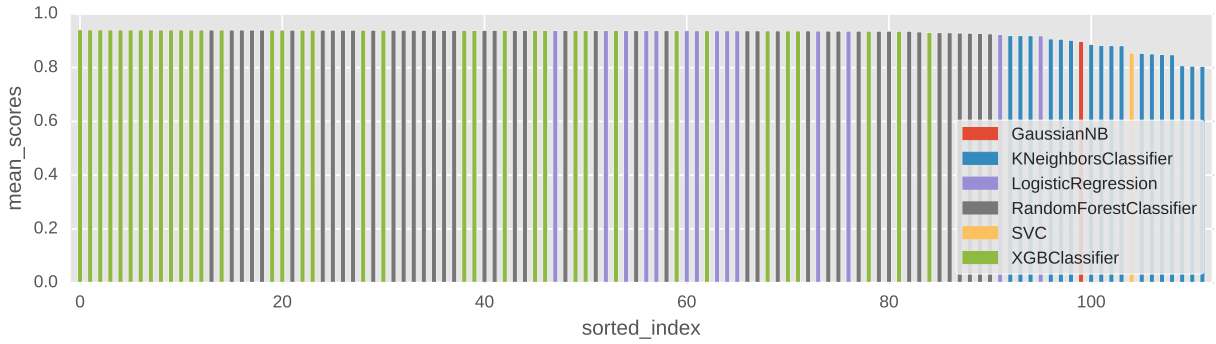


Figure 2: Mean scores over all months for all evaluations.

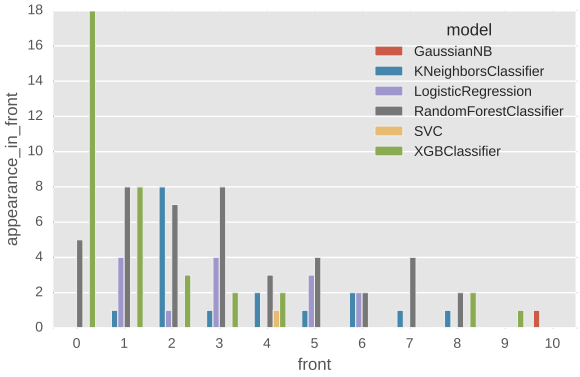


Figure 3: Models ranked by Pareto dominance.

Use scripts for submitting the jobs and obtaining the logs.

5. CONCLUSION

The paper presents the model selection and hyper-parameter tuning problems. Since the problem is computationally demanding JSI grid infrastructure was used in combination with a simple random search method for optimization in complex search space. Methods were applied to the Metis use-case, where the goal was to predict whether a pupil will have a negative grade at the end of the grading period. Using random search on the grid we have found a well performing white-box model that will be applied in production.

The experience in working with the JSI grid infrastructure was also described. It was found that connectivity issues between nodes, resource issues on the nodes themselves, and batch job submissions prevent the user from fully exploiting the grid resources. However, the grid can still be exploited by following the proposed best practices. By increasing the awareness of the grid availability among the JSI researchers we expect that several issues that are now present will be removed in the future. This will enable JSI researchers to use the grid to its full potential and give the JSI advantage in national and international research projects.

All scripts and data for this paper are available at `https://repo.ijs.si/jernejzupancic/IS2016/tree/master`.

6. REFERENCES

[1] T. Bartz-Beielstein and S. Markon. Tuning search algorithms for real-world applications: A regression tree based approach. In *Evolutionary Computation, 2004. CEC2004. Congress on*, volume 1, pages 1111–1118. IEEE, 2004.

[2] J. Bergstra and Y. Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb):281–305, 2012.

[3] J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems*, pages 2546–2554, 2011.

[4] C. M. Bishop. *Neural networks for pattern recognition*. Oxford university press, 1995.

[5] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. *arXiv preprint arXiv:1603.02754*, 2016.

[6] G. Hinton. A practical guide to training restricted boltzmann machines. *Momentum*, 9(1):926, 2010.

[7] F. Hutter. *Automated configuration of algorithms for solving hard computational problems*. PhD thesis, University of British Columbia, 2009.

[8] D. R. Jones, M. Schonlau, and W. J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492, 1998.

[9] D. Kuznar and M. Gams. Metis: system for early detection and prevention of student failure. In *6thInternational Workshop on Combinations of Intelligent Methods and Applications (CIMA 2016)*, page 39, 2016.

[10] H. Larochelle, D. Erhan, A. Courville, J. Bergstra, and Y. Bengio. An empirical evaluation of deep architectures on problems with many factors of variation. In *Proceedings of the 24th international conference on Machine learning*, pages 473–480. ACM, 2007.

[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Sinteza slovenskega govora na mobilni platformi Android

Tomaž Šef
Institut "Jožef Stefan"
Jamova cesta 39
1000 Ljubljana
+386 1 477 34 19
tomaz.sef@ijs.si

POVZETEK

V članku predstavljamo mobilno aplikacijo in e-storitev, ki omogoča sintezo slovenskega govora na mobilni platformi Android. Podpora se vgradi v sam operacijski sistem mobilne naprave, kar zagotavlja povezljivost s poljubnim programom nameščenim na tej napravi. Opisana rešitev je nepogrešljiva za slepe in slabovidne ter osebe z motnjami branja, ker jim omogoča rabo različnih orodij za delo z mobilnimi napravami, učenje in samopomoč. Koristna je tudi za vse druge uporabnike mobilne platforme Android, saj je govor velikokrat najbolj primerna in naravna oz. včasih celo edina varna oblika posredovanja informacij in drugih vsebin.

Ključne besede

Govorni bralnik besedil, sinteza slovenskega govora, mobilna platforma Android.

1. UVOD

Razvili smo mobilno aplikacijo in e-storitev, ki omogoča sintezo slovenskega govora na mobilni platformi Android. Sistem je zasnovan v obliki strežnika v oblaku in pripadajočih aplikacij. Podpira industrijski standard Android Speech API, kar zagotavlja vključitev sintetizatorja govora v sam operacijski sistem in posledično njegovo povezljivost s poljubnimi programi na tej razširjeni mobilni platformi. Po kvaliteti nekoliko slabši sintetizator slovenskega govora se nahaja na sami mobilni napravi (off-line različica) in omogoča branje tudi ob izklopljeni ali nedosegljivi podatkovni povezavi. Bolj kakovosten in naraven govor je dosegljiv preko e-storitve v oblaku (on-line različica), ki za svoje delovanje potrebuje podatkovno povezavo (3G ali Wifi) z oddaljenim govornim strežnikom.

Opisana rešitev je nepogrešljiva za slepe in slabovidne ter osebe z motnjami branja, saj jim omogoča rabo različnih orodij za delo z mobilnimi napravami, učenje in samopomoč. Najbolj razširjena programska oprema je že prilagojena njihovim specifičnim potrebam. Uporabna pa je tudi za vse druge uporabnike mobilne platforme Android, saj je govor velikokrat najbolj primerna in naravna oz. včasih celo edina varna oblika posredovanja informacij in drugih vsebin (npr. branje elektronske pošte med vožnjo).

Trenutno obstajata dve različici programa oz. e-storitve:

- eBralec KSS (https://play.google.com/store/apps/details?id=si.ijs.dis.ebralec) je namenjen slepim in slabovidnim, osebam z motnjami branja ter zaposlenim v javni upravi,
- DysLex [1] (https://play.google.com/store/apps/details?id=si.ijs.dyslex) je brezplačna storitev, ki je splošno dostopna in vsem na razpolago.

Obe verziji vključujeta »off-line« sintetizator slovenskega govora, ki deluje tudi takrat, ko podatkovna povezava ni na razpolago. Obe verziji prav tako podpirata »on-line« sintezo, ki zahteva podatkovno povezavo, s to razliko, da so novo razviti glasovi, ki zagotavljajo občutno višjo stopnjo naravnosti rezultirajočega sintetičnega govora, dosegljivi le preko eBralca On-Line.

2. E-STORITEV

E-storitev omogoča vgradnjo poljubnega Microsoft SAPI kompatibilnega (slovenskega) sintetizatorja govora v operacijski sistem mobilnih naprav. E-storitev je sestavljena iz strežnika in mobilne aplikacije.

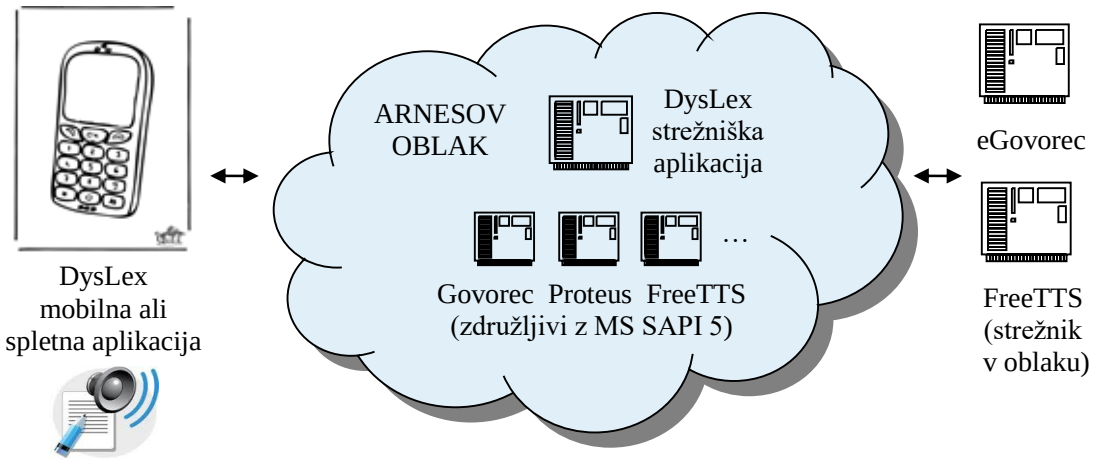
Strežnik na eni strani omogoča povezavo s sintetizatorjem govora, na drugi strani pa z mobilno napravo. Sintetizator govora je lahko naložen bodisi na istem strežniku kot sama strežniška aplikacija bodisi se poveže z nekim oddaljenim strežnikom z nameščenim sintetizatorjem govora (npr. s strežnikom eGovorec). Sintetizator govora lahko razvijalci sproti posodabljajo, ne da bi uporabniki za to sploh vedeli oz. jim za uporabo najnovejše različice ni potrebno narediti čisto nič (ni nobene potrebe po prenosu sintetizatorja govora, saj se ta posodobi na samem strežniku).

Mobilna aplikacija na platformi Android omogoča vgradnjo glasov vseh v strežnik prijavljenih sintetizatorjev govora v operacijski sistem mobilne naprave. V nastavitvenem meniju mobilne naprave se pri govornem bralniku avtomatsko pojavijo še slovenski glasovi.

Arhitektura obeh razvitih e-storitev je podobna Tako DysLex kot eBralec KSS znata samodejno preklapljati med 3G in Wifi povezavo, pri čemer lahko rabo mobilne podatkovne povezave (3G) tudi onemogočimo. Vgrajeni senzor prenosa podatkov na mobilni napravi zagotavlja spremljanje in nadzor nad porabo podatkovnega prenosa. Ob odsotnosti kakršnekoli podatkovne povezave sistema avtomatsko preklopita na (po kvaliteti sicer slabšo) »off-line« sintezo govora. Ko je povezava z govornim strežnikom ponovno mogoča, pa se sinteza govora začne zopet izvajati v boljši kvaliteti.

Arhitektura razvite e-storitve DysLex s strežniško aplikacijo v Arnesovem oblaku in pripadajočo mobilno aplikacijo (odjemalcem) na mobilni napravi je prikazana na sliki 1 (http://dis.ijs.si/dyslex/). Sintetizatorji govora se nahajajo bodisi v Arnesovem oblaku (MS SAPI 5 združljivi) bodisi na samostojnem strežniku (e-Govorec).

Pri e-storitvi eBralec KSS se strežniška aplikacija in novo razviti sintetizator govora eBralec nahajata izključno na strežniku Knjižnice slepih in slabovidnih. eBralec KSS zna dodatno upravljati z vgrajeno zaščito novega sintetizatorja govora, nastavlja se lahko tudi različne parametre »Besednika« in »Analizatorja«. Androidna aplikacija eBralec KSS omogoča še vpis testnega besedila, rabo odložišča ter posreduje (govorna) navodila o rabi programa (slika 2).



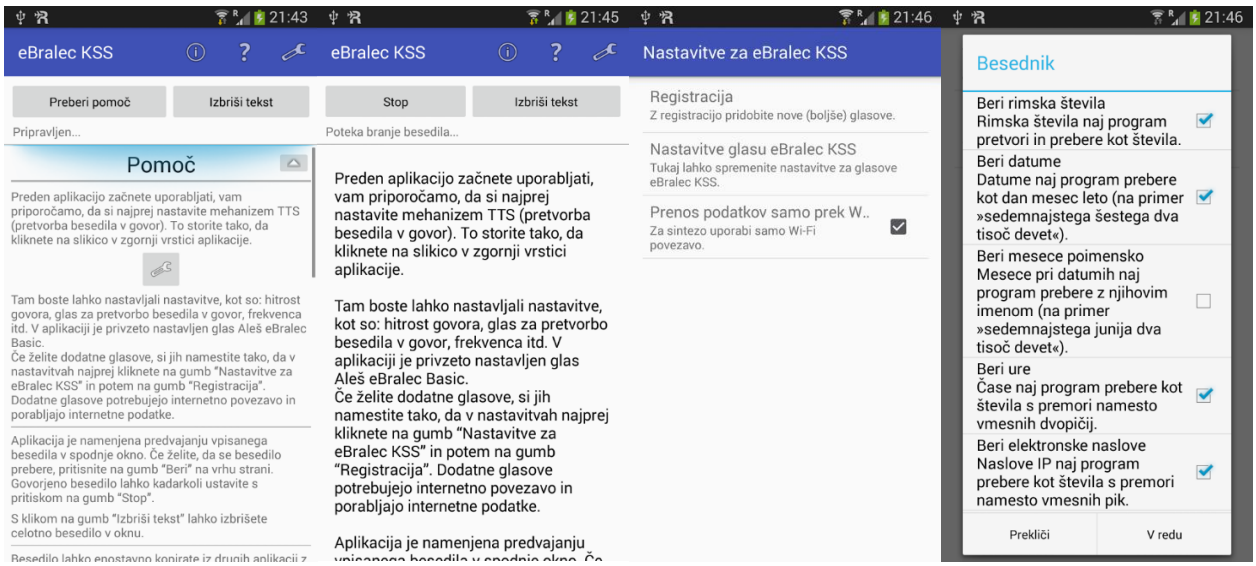
Slika 1. Arhitektura e-storitve DysLex.

E-storitev se zelo preprosto nadgrajuje in posodabljala. Rešitev je horizontalno skalabilna, s čimer je omogočeno optimalno prilagajanje strojnih virov dejanskim potrebam oz. rasti števila uporabnikov in vsebin; omogočena je hitra vzporedna obdelava na večprocesorskih računalnikih, podprta je možnost namestitve na grozd računalnikov (v oblaku). Sintetizatorje govora lahko razvijalci posodabljajo sproti, ne da bi uporabniki za to sploh vedeli. Slednji zgolj opažajo, da je govor čedalje bolj razumljiv in naraven ter da se v sistemu pojavljajo novi glasovi v slovenskem jeziku.

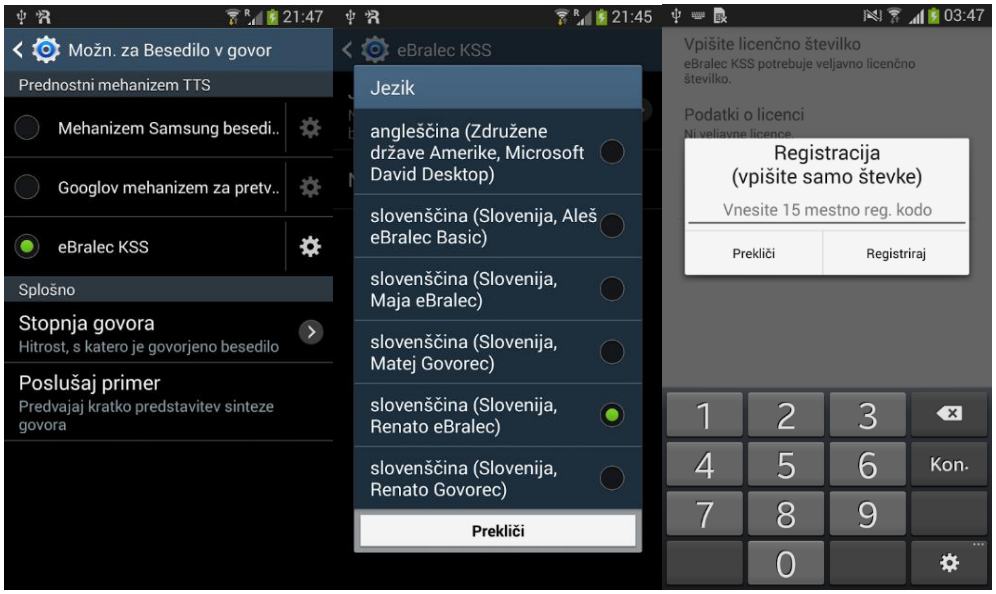
Vgradnja govornega bralnika v sam operacijski sistem mobilne naprave (s podporo standardu Android Speech API, slika 3) zagotavlja največjo uporabniško dostopnost in najboljšo uporabniško izkušnjo. Avtomatsko začnejo delovati vse z operacijskim sistemom podprte funkcionalnosti govornega bralnika mobilne naprave tudi v slovenskem jeziku. Takšne funkcionalnosti so npr.:

- branje izbranega besedila v poljubni aplikaciji (za lektoriranje lastnega dela dislektikov [2], branje knjig, izgovorjava posameznih besed, branje elektronskih izročkov, dostopanje na internet),
- branje zaslona mobilne naprave,
- avtomatsko branje samodejnih popravkov in velikih začetnic,
- branje menijev v načinu za slepe in slabovidne oz. drugače prizadete (funkcija TalkBack).

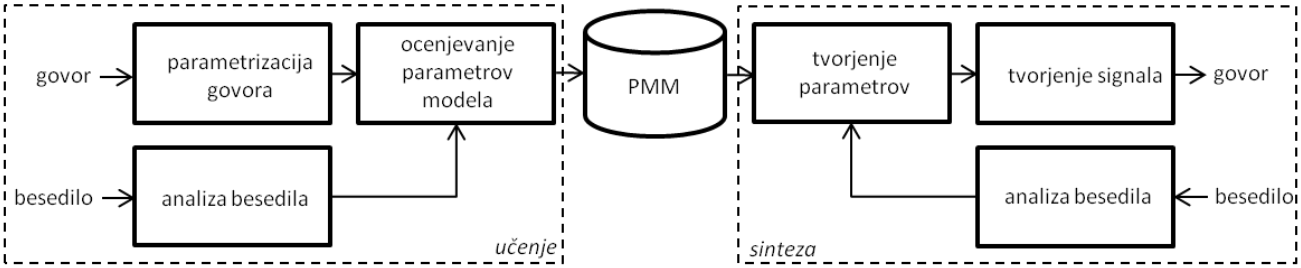
Pri uporabi sintetizatorjev govora je namreč zelo pomembno, da je podprto čim več tipov elektronskega gradiva (npr. tudi pdf datoteke in internetni viri) ter čim več že obstoječe programske opreme. **Slovenski glasovi** sedaj delujejo v vseh aplikacijah, ki uporabljajo govorni bralnik operacijskega sistema Android; vključno z vsemi aplikacijami za dislektike ter slepe in slabovidne.



Slika 2. Androidna aplikacija eBralec KSS



Slika 3. Vgradnja eBralca v izbirnike operacijskega Sistema Android



Slika 4. Shema sistema eBralec za tvorjenje govora z uporabo PMM [4]

Strežnik e-storitve DysLex je dostopen tudi programerjem, ki bi želeli podporo sinteze (slovenskega) govora vnesti neposredno v svoje programe. Odjemalec, ki je bodisi spletna stran bodisi aplikacija, pošlje želeno besedilo na strežnik HTTP. Strežnik pretvori besedilo v govor, ki ga v obliki zvočnega zapisa pošlje nazaj odjemalcu v predvajanje. Strežnik se nahaja na IP naslovu 141.ablak.arnes.si in posluša na vrata 80. **Splošne funkcije:**

Funkcija ***info***

- Namenjena je splošni poizvedbi o glasovih, ki so nameščeni na strežniku DysLex. Odjemalec praviloma najprej kliče to funkcijo, da lahko (po potrebi) v svojem uporabniškem vmesniku prikaže spisek vseh razpoložljivih glasov.
- Zahteva tipa GET (GET Request)
  Primer: *http://141.ablak.arnes.si/?f=info*
- Odgovor tipa *application/json*
  Pomembni parametri odgovora so: ***id, name, lang, country in age***
- Primer odgovora v JSON formatu:
  {"id":"GovRenato",
  "name":"Renato Govorec",
  "lang":"slv",
  "gender":"Male",
  "age":"Adult",
  "country":"SI",
  "localVoice":true,
  "remoteServerName":"",
  "remoteServerIP":"localhost"}

Funkcija *speak*

- Namenjena je pretvorbi besedila v govor.
- Parametri:
  f=speak – strežniku pove, da hoče pretvorbo besedila v zvok
  t="<besedilo>" – besedilo, ki bo pretvorjeno v zvok
  v="<ime glasu>" – ime glasu za sintezo govora
  s="<hitrost>" – hitrost govora
  o=*"<mpeg oz. wav>"* – kodiranje
- Primer sinteze:
  http://141.ablak.arnes.si/?f=speak&t=Primer&v=Renato%20Govorec&s=0&o=mpeg

Funkcija ***heartbeat***

- Namenjena je za preverjanje delovanja strežnika.
- Odgovor s statusom **200 OK** in besedilo **Server is working!**
- Primer zahteve: http://141.ablak.arnes.si/?f=heartbeat

Spletna stran projekta DysLex s podrobnejšimi informacijami se nahaja na naslovu http://dis.ijs.si/dyslex/.

3. SINTETIZATOR GOVORA

Umetno generirani govor mora zveneti naravno in biti prijeten za poslušanje [3]. Pomembne so tudi nastavitve za hitrost branja in jakost zvoka ter možnost uporabe različnih glasov.

eBralec (http://ebralec.si) je najnovejši sintetizator slovenskega govora [4]. Prvenstveno je namenjen slepim in slabovidnim uporabnikom ter osebam z motnjami branja. V primerjavi s preostalimi (predhodnimi) sintetizatorji govora za slovenski jezik omogoča občutno višjo stopnjo naravnosti sintetičnega govora. Aplikacija eBralec je plod sodelovanja Instituta »Jožef Stefan« ter podjetij Alpineon in Amebis. Vsebuje moški (eBralec Renato) in ženski (eBralec Maja) glas. Temelji na obsežnih govornih korpusih (tabela 1) in sintezi govora s prikritimi markovovimi modeli (PMM; slika 4). Na platformi Android je dosegljiv preko aplikacije eBralec KSS v »on-line« načinu delovanja.

Velikost besednega korpusa	7.145.345 povedi 77 milijonov besed
Obseg govorne zbirke	***4.000 povedi*** 46.785 besed 6 ur 3 min posnetkov za ženski glas 5 ur 33 min posnetkov za moški glas
Število različnih difonov v zbirki	1.883
Število različnih trifonov v zbirki (št. kombinacij v korpusu)	21.369 (24.702)

Tabela 1. Podatki o govorni zbirki eBralca [4]

Govorec je starejši sintetizator govora, ki temelji na difonski sintezi. Prva različica je nastala v letu 1998, kasneje pa so se posamezni moduli nadgrajevali. Difonska sinteza govora zveni precej manj naravno od korpusne [5, 6]. Vebuje dva moška glasova: *Renato Govorec* in *Matej Govorec*. Na platformi Android sta oba glasova dosegljiva tako preko aplikacije eBralec KSS kot preko aplikacije DysLex v »on-line« načinu delovanja. Oba glasova sta priložena tudi novemu sintetizatorju govora eBralec.

E-govorec (http://dis.ijs.si/e-govorec/) ponudnikom najraz-ličnejših e-vsebin omogoča dinamično podajanje informacij v govorni obliki ter v domačem slovenskem jeziku. Po kvaliteti oz. naravnosti govora je enak Govorcu. Gre za brezplačno storitev, ki jo lahko uporablja kdorkoli. Do njega dostopa tudi aplikacija DysLex.

eBralec off-line je kompakten sintetizator slovenskega govora, ki teče neposredno na mobilni napravi. Temelji na difonski sintezi. Pri njegovem razvoju je bil poudarek predvsem na odzivnosti ter majhni potrebi po računski moči in pomnilniku. Posledica je občutno nižja kvaliteta oz. naravnost govora kot pri eBralcu. Njegova prednost je, da za svoje delovanje na mobilnih napravah ne potrebuje podatkovne povezave. Vsebuje moški glas *Aleš*. Na platformi Android ga v »off-line« načinu uporabljata tako aplikacija eBralec KSS kot DysLex.

4. ZAKLJUČEK

Predstavljeni e-storitvi in mobilni aplikaciji omogočata podporo sintezi slovenskega govora v operacijskem sistemu Android, kar pomeni, da lahko umetno generirani govor poslušamo v vseh aplikacijah (npr. PDF reader, brskalnik), ki sintezo govora podpirajo in so naložene na teh napravah. Za operacijska sistema iOS in Windows Phone takšna vgradnja žal še ni možna, ker proizvajalca operacijskega sistema te funkcionalnosti zaenkrat še ne omogočata.

Na mobilnih napravah z Androidom lahko dislektiki ter slepi in slabovidni sedaj uporabljajo vse (učne) pripomočke, izdelane za globalni trg ali že standardno vgrajene v napravo.

Naravnost in razumljivost novega sintetizatorja slovenskega govora eBralec (http://ebralec.si/) sta primerljivi s sintetizatorji govora za druge jezike. Poslušanje takšnega govora ni več naporno, zato je sintetizator primeren za najširši krog uporabnikov.

5. ZAHVALA

Posebna zahvala gre Dejanu Kostadinovskemu in Janiju Bizjaku, ki sta sprogramirala strežniško in mobilno aplikacijo ter zagotovila podporo standardu Android Speech API.

Operacijo je delno sofinancirala Evropska unija iz Evropskega sklada za regionalni razvoj ter Ministrstvo za izobraževanje, znanost in šport. Operacija se je izvajala v okviru Operativnega programa krepitve regionalnih razvojnih potencialov za obdobje 2007-2013, razvojne prioritete: Gospodarsko razvojna infrastruktura; prednostne usmeritve Informacijska družba.

Razvoj eBralca je bil delno financiran v okviru projekta Knjižnica slepih in slabovidnih Minke Skaberne. Operacijo je delno financirala Evropska unija iz Evropskega socialnega sklada. Operacija se je izvajala v okviru Operativnega programa razvoja človeških virov, razvojne prioritete "Enake možnosti in spodbujanje socialne vključenosti", prednostne usmeritve "Dvig zaposlenosti ranljivih družbenih skupin na področju kulture in podpora njihovi socialni vključenosti".

6. LITERATURA IN VIRI

[1] Šef, T., 2015. Univerzalni govorni e-bralnik za slovenski jezik kot osebni učni pripomoček za ljudi z disleksijo in različnimi vrstami motnje vida, *Vzgoja in izobraževanje v informacijski družbi - VIVID 2015 : zbornik referatov,* Fakulteta za organizacijske vede, str. 510-519, 2015.

[2] *Bravo, društvo za pomoč otrokom in mladostnikom s specifičnimi učnimi težavami.* (http://www.drustvo-bravo.si).

[3] Taylor, P., 2009. *Text-to-Speech Synthesis*, Cambridge University Press.

[4] Žganec Gros, J., Vesnicer, B.,Rozman, S., Holozan, P., Šef, T., 2016. Sintetizator govora za slovenščino eBralec, *Zbornik konference Jezikovne tehnologije in digitalna humanistika*, Filozofska fakulteta, Univerza v Ljubljani, 2016.

[5] Šef, T., Gams, M., 2003. Speaker (GOVOREC): a complete Slovenian text-to speech system. *International journal of Speech Technology*, Kluwer Academic Publishers, št. 6, str. 277-287, 2003.

[6] Šef, T. 2001. *Analiza besedila v postopku sinteze slovenskega govora*. Doktorska disertacija, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2001.

Showing the Knee of a 4-D Pareto Front Approximation via Different Visualization Methods

Tea Tušar
Department of Intelligent Systems
Jožef Stefan Institute
Ljubljana, Slovenia
tea.tusar@ijs.si

Bogdan Filipič
Department of Intelligent Systems
Jožef Stefan Institute
Ljubljana, Slovenia
bogdan.filipic@ijs.si

ABSTRACT

When decision makers select one or more trade-off solutions to a multiobjective optimization problem, they are mostly interested in solutions residing at knees—regions of the Pareto front where a small improvement in one objective leads to a large deterioration in at least one other objective. It is therefore important to be able to detect such regions, preferably through visualization. This paper presents visualizations of Pareto front approximations of a multiobjective problem with knees. More specifically, we show how a sampled Pareto front of a four-objective problem with a single knee looks like when visualized using seven different methods (scatter plot matrix, bubble chart, parallel coordinates, radial coordinate visualization, level diagrams, hyper-radial visualization and prosections). We can observe that while the first four methods cannot visualize the knee, the remaining three are able to do so and are therefore more suitable for use in visualization of Pareto front approximations.

1. INTRODUCTION

When inspecting a set of solutions to a multiobjective optimization problem, decision makers usually prefer solutions lying at knees—regions of the Pareto front where a small improvement in one objective leads to a large deterioration in at least one other objective [2]. Visualizing knee regions in a clear and concise way is therefore a crucial requirement for supporting the decision making process. Finding a good representation of the Pareto front and its knees is rather straightforward in the case of two or three objectives (see Figures 1 and 2 for two examples)[1], but very challenging when the number of objectives is four or more [8].

This paper presents visualizations of the single-knee instance of the DEB4DK problem [2, 7] using seven different methods, namely the scatter plot matrix, bubble chart, parallel coordinates [5], radial coordinate visualization [4], level diagrams [1], hyper-radial visualization [3] and prosections [8]. All these methods have been previously used to visualize Pareto front approximations and have been analyzed with respect to some desired properties for visualization methods [8].

Section 2 provides the formal definition of the DEB4DK problem, while the visualizations with different methods are shown in Section 3. Some concluding remarks are presented in Section 4.

[1]Minimization in all objectives is assumed throughout this paper.

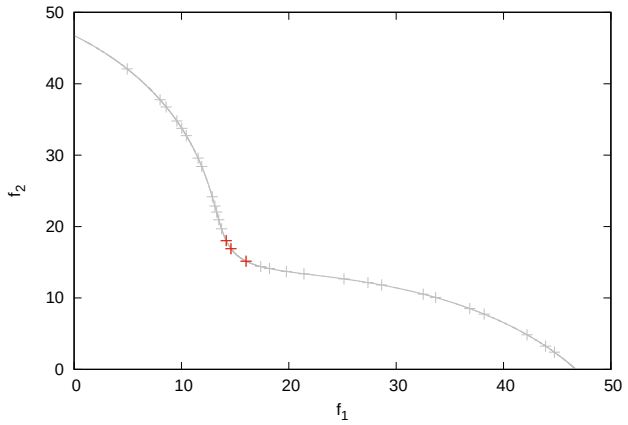


Figure 1: The DEB2DK problem with a single knee (solutions near the knee are shown in red).

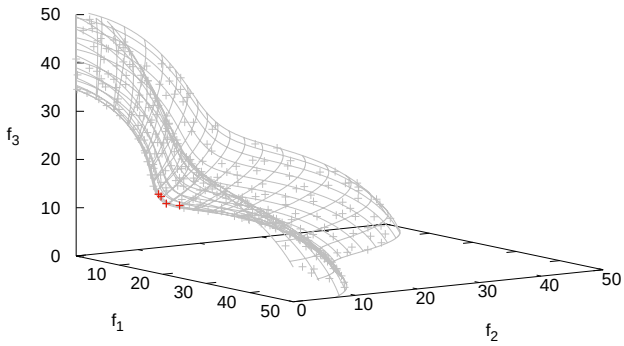


Figure 2: The DEB3DK problem with a single knee (solutions near the knee are shown in red).

2. THE DEB4DK PROBLEM

Branke et al. introduced a family of optimization problems with knees that are based on the DTLZ benchmark problems and scalable to any number of objectives [2]. While only the instances with two and three objectives were presented originally, they were scaled to four and five objectives in [7].

Let us recall the formal definition of the DEB4DK problem

with four objectives:

$$f_1(\boldsymbol{x}) = g(\boldsymbol{x})r(\boldsymbol{x})\sin\left(\frac{\pi}{2}x_1\right)\sin\left(\frac{\pi}{2}x_2\right)\sin\left(\frac{\pi}{2}x_3\right)$$
$$f_2(\boldsymbol{x}) = g(\boldsymbol{x})r(\boldsymbol{x})\sin\left(\frac{\pi}{2}x_1\right)\sin\left(\frac{\pi}{2}x_2\right)\cos\left(\frac{\pi}{2}x_3\right)$$
$$f_3(\boldsymbol{x}) = g(\boldsymbol{x})r(\boldsymbol{x})\sin\left(\frac{\pi}{2}x_1\right)\cos\left(\frac{\pi}{2}x_2\right)$$
$$f_4(\boldsymbol{x}) = g(\boldsymbol{x})r(\boldsymbol{x})\cos\left(\frac{\pi}{2}x_1\right)$$
$$g(\boldsymbol{x}) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i$$
$$r(\boldsymbol{x}) = \frac{r_1(x_1) + r_2(x_2) + r_3(x_3)}{3}$$
$$r_i(x_i) = 5 + 10(x_i - 0.5)^2 + \frac{3\cos(2K\pi x_i)}{K}$$
$$\boldsymbol{x} \in [0,1]^n.$$

Here, n denotes the dimensionality of the decision space, while parameter K is used to control the number of knees on the Pareto front. A DEBmDK optimization problem with m objectives has K^{m-1} knees. This means that the four-objective DEB4DK problem can be instantiated to have $1, 8, 27, \ldots$ knees (for $K = 1, 2, 3, \ldots$ respectively). In the rest of the paper, the value $K = 1$ yielding the Pareto front with a single knee will be used.

3. RESULTS OF DIFFERENT VISUALIZATION METHODS

This section shows how different methods visualize the single knee of the DEB4DK problem with $K = 1$.

The set of solutions approximating the Pareto front (also called approximation set) was obtained by sampling the front of the DEB4DK problem with 3000 points. Among them, eight solutions closest to the knee were chosen as the solutions "at the knee" to be emphasized in the visualizations (analogous to the red knee solutions emphasized in Figures 1 and 2). The underlying idea is that a visualization method should be capable of showing the knee solutions in a way that makes the knee recognizable to a decision maker.

Since some methods have difficulties when visualizing a large number of solutions, only the first 300 solutions were used in such cases. Out of the eight knee solutions contained in the original sample, five were retained in this smaller set.

3.1 Scatter Plot Matrix

Let us first view the results of the scatter plot matrix—a matrix of all possible 2-D projections of a multidimensional set of solutions. Figure 3 presents the scatter plot matrix for the smaller approximation set containing 300 solutions. Because of the projection to a 2-D space, most of the information is lost and the solutions at the knee cannot be differentiated from the rest of the solutions. Even an interactive exploration cannot be of much help in such a case.

3.2 Bubble chart

A bubble chart is a scatter plot in which three objectives are shown on axes while the fourth objective is represented with point size. Figure 4 contains the bubble chart for the

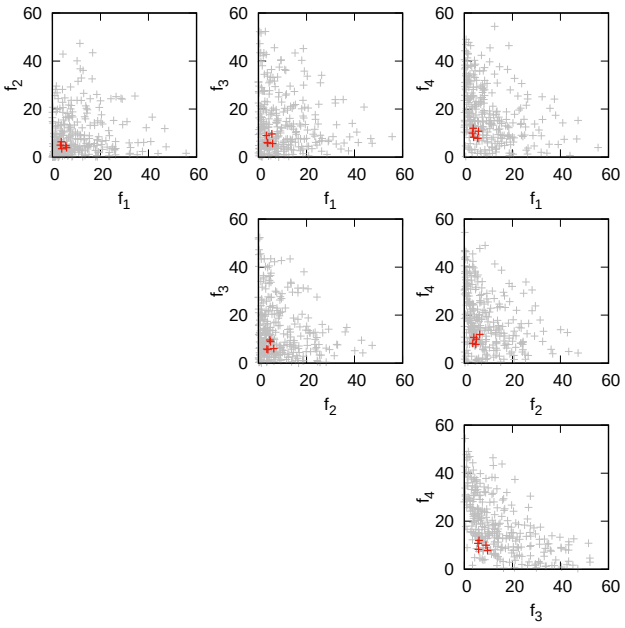


Figure 3: Scatter plot matrix of 300 Pareto-optimal solutions of the DEB4DK problem with a single knee (solutions near the knee are shown in red).

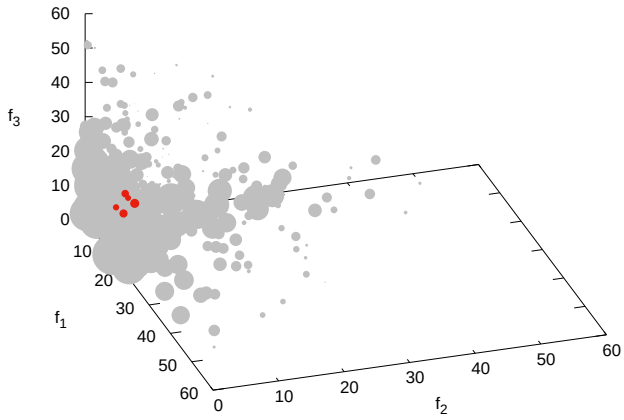


Figure 4: Bubble chart showing 300 Pareto-optimal solutions of the DEB4DK problem with a single knee (solutions near the knee are shown in red).

approximation set with 300 solutions. Similarly as before, this plot shows that the red knee solutions have low values in all four objectives, but we cannot perceive the presence of a knee.

3.3 Parallel Coordinates

Parallel coordinates [5] is a very popular method for visualizing multidimensional data that represents objectives as vertical parallel lines. Each solution is shown as a poly-line intersecting each objective (i.e. coordinate) at its value. In the case of our smaller approximation set (see Figure 5), the knee solutions do not stand out from the rest and therefore do not give a good idea regarding the shape of the approx-

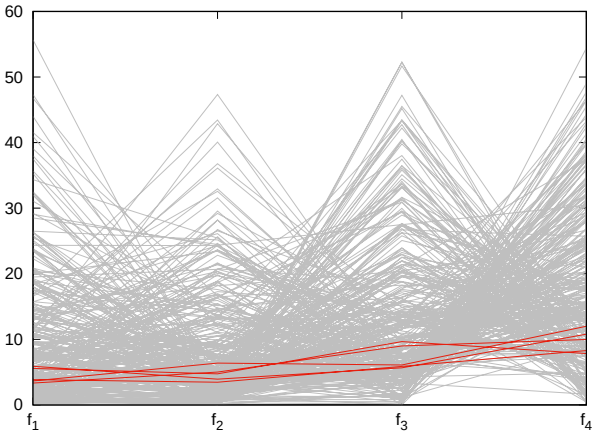


Figure 5: Parallel coordinates of 300 Pareto-optimal solutions of the DEB4DK problem with a single knee (solutions near the knee are shown in red).

imation set. Moreover, the method is very sensitive to the number of displayed solutions. When this number is small, it can be very informative and is able to show dominance relations between solutions. When, on the other hand, hundreds or thousands of solutions need to be visualized, the cluttered result causes loss of most of the information.

3.4 Radial Coordinate Visualization

The radial coordinate visualization (also called RadViz [4]) places objectives on the circumference of a unit circle and the solutions inside the circle so that the distance of a solution to each of the objectives is proportional to its value in that objective. For example, solutions with greater values in f_1 than in any other objective are placed closer to f_1 than the other objectives. Figure 6 shows the radial coordinate visualization of the entire approximation set with 3000 solutions. Again, the knee solutions cannot be differentiated from the rest in this plot.

3.5 Level Diagrams

Level diagrams [1] plot each solution against one of its objectives and the (Euclidean) distance to the ideal point. In case of four objectives, four such diagrams are produced. Figure 7 presents the level diagrams for the smaller approximation set and we can clearly see the knee in all of them.

3.6 Hyper-Radial Visualization

In hyper-radial visualization [3] the solutions are plotted against their distance to the ideal point (their *hyper-radius*) separately for two subsets of objectives. In our case, the x axis represents the hyper-radius for objectives f_1 and f_2, while the y axis represents the hyper-radius for objectives f_3 and f_4. As shown in Figure 8, the eight knee solutions from the larger approximation set are clearly recognizable in this visualization.

A note on the level diagrams and the hyper-radial visualization: while both methods are able to show the knee in this case, this is mostly due to the fact that the knee has the shortest distance to the ideal point. In case of multiple knees, for example, the knees that are not very close to

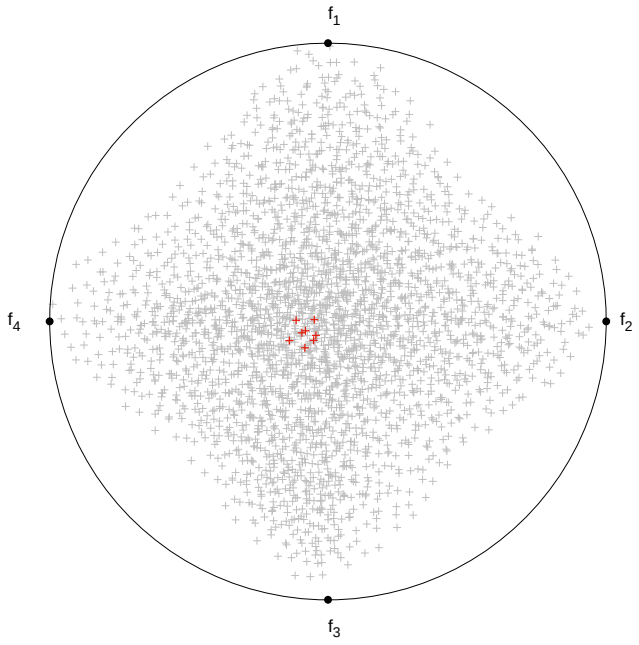


Figure 6: Radial coordinate visualization of 3000 Pareto-optimal solutions of the DEB4DK problem with a single knee (solutions near the knee are shown in red).

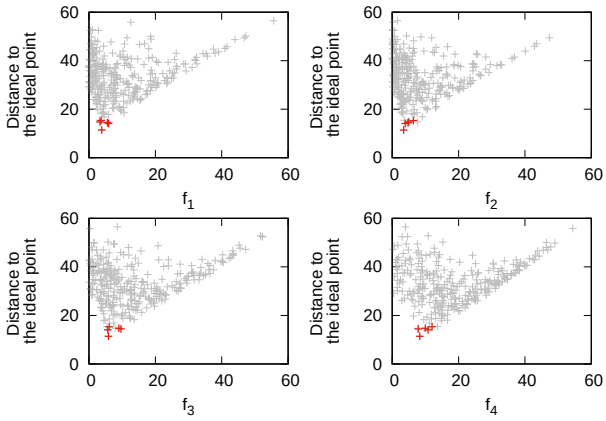


Figure 7: Level diagrams for 300 Pareto-optimal solutions of the DEB4DK problem with a single knee (solutions near the knee are shown in red).

the ideal point would probably be undetectable with these visualization methods. Showing this is left for future work.

3.7 Prosections

Prosections [6, 8] are projections of only a section of the objective space at a time. For example, a section of width d is selected at angle φ on the plane $f_1 f_2$. All solutions that fall in this section are projected onto the line going through the ideal point at angle φ while all other solutions are discarded. By showing a 3-D scatter plot with the x axis representing the projected line and the other two axes

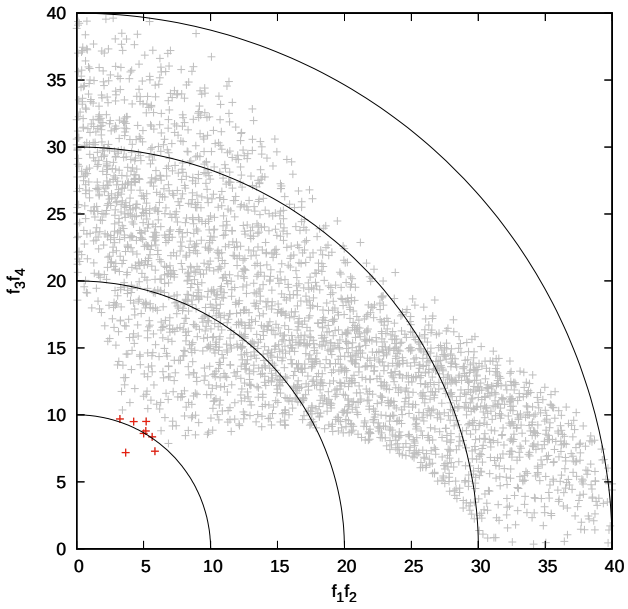


Figure 8: Hyper-radial visualization of 3000 Pareto-optimal solutions of the DEB4DK problem with a single knee (solutions near the knee are shown in red).

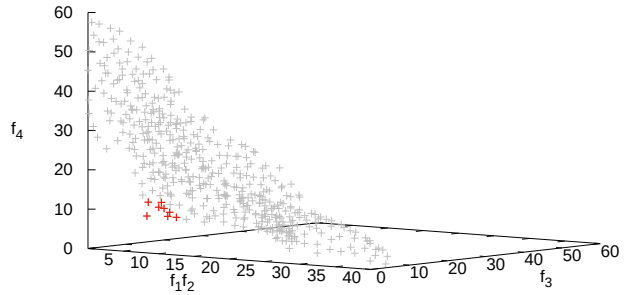


Figure 9: Prosection $4\mathrm{D}(\mathbf{0}, f_1 f_2, 45°, 2)$ showing a part of 3000 Pareto-optimal solutions of the DEB4DK problem with a single knee (solutions near the knee are shown in red).

representing objectives f_3 and f_4, we get a 3-D representation of this section of the 4-D objective space that can be denoted as $4\mathrm{D}(\mathbf{0}, f_1 f_2, \varphi, d)$. Figure 9 shows the prosection $4\mathrm{D}(\mathbf{0}, f_1 f_2, 45°, 2)$, that is, a prosection on the plane $f_1 f_2$ using the angle $\varphi = 45°$ and section width $d = 2$. The knee points are clearly visible in this plot. In theory, prosections should be able to visualize any number of knees regardless of their position (using multiple plots—each visualizing a different section of the space), but experiments to demonstrate this are still needed.

The advantages and disadvantages of prosections originate from the fact that only a section of the space is visualized at a time. While this means that a visualization of high accuracy that mostly preserves the dominance relations among solutions is possible, it also requires to plot multiple visualizations (using various planes and angles) to get a complete view of an approximation set. Another advantage of this method is that it can easily handle large sets of solutions because only a part of the set is visualized in a single plot.

4. CONCLUSIONS

This paper presented how seven visualization methods previously used to visualize Pareto front approximations cope with visualization of knees. Only three methods (level diagrams, hyper-radial visualization and prosections) were able to show the single knee of the four-objective DEB4DK problem. Additional work is needed to see how these visualization methods perform on a problem with more knees.

5. ACKNOWLEDGMENTS

This work is part of a project that has received funding from the *European Union's Horizon 2020 research and innovation program* under grant agreement No. 692286. This work was partially funded by the Slovenian Research Agency under research program P2-0209.

6. REFERENCES

[1] X. Blasco, J. M. Herrero, J. Sanchis, and M. Martínez. A new graphical visualization of n-dimensional Pareto front for decision-making in multiobjective optimization. *Information Sciences*, 178(20):3908–3924, 2008.

[2] J. Branke, K. Deb, H. Dierolf, and M. Oswald. Finding knees in multi-objective optimization. In *Proceedings of the 8th International Conference on Parallel Problem Solving from Nature, PPSN VIII, Birmingham, UK*, volume 3242 of *Lecture Notes in Computer Science*, pages 722–731. Springer, 2004.

[3] P.-W. Chiu and C. Bloebaum. Hyper-radial visualization (HRV) method with range-based preferences for multi-objective decision making. *Structural and Multidisciplinary Optimization*, 40(1–6):97–115, 2010.

[4] P. E. Hoffman, G. G. Grinstein, K. Marx, I. Grosse, and E. Stanley. DNA visual and analytic data mining. In *Proceedings of the IEEE Conference on Visualization*, pages 437–441, Los Alamitos, CA, USA, 1997. IEEE.

[5] A. Inselberg. *Parallel Coordinates: Visual Multidimensional Geometry and its Applications*. Springer, New York, NY, USA, 2009.

[6] T. Tušar and B. Filipič. Visualizing 4D approximation sets of multiobjective optimizers with prosections. In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO '11, Dublin, Ireland*, pages 737–744. ACM, 2011.

[7] T. Tušar and B. Filipič. Scaling and visualizing multiobjective optimization test problems with knees. In *Proceedings of the 15th International Multiconference Information Society, IS 2012, Ljubljana, Slovenia*, volume A, pages 155–158, Ljubljana, Slovenia, 2012. Jožef Stefan Institute.

[8] T. Tušar and B. Filipič. Visualization of Pareto front approximations in evolutionary multiobjective optimization: A critical review and the prosection method. *IEEE Transactions on Evolutionary Computation*, 19(2):225–245, 2015.

Machine learning method for stress detection with an EEG device

Martin Gjoreski, Mitja Luštrek, Matjaž Gams
Department of Intelligent Systems, Jožef Stefan Institute
Jožef Stefan International Postgraduate School
Jamova cesta 39, Ljubljana, Slovenia
{martin.gjoreski, mitja.lustrek, matjaz.gams}@ijs.si

ABSTRACT
We present a study for stress detection using a one-channel commercial EEG device. For this purpose, a stress-inducing experiments were performed. 21 subjects were monitored while performing mental-arithmetic tasks under time and evaluation pressure. The data was used to develop a machine learning method for stress detection which consists of: a segmentation phase, a filtering phase, a feature extraction phase and a model-learning phase. The initial evaluation results are promising and showcase that person-specific models can perform for a 19 percentage points better than the majority class.

Categories and Subject Descriptors
J.3 [**Computer Applications**]: *Health*

Keywords
Stress detection; EEG device; machine learning; health.

1. INTRODUCTION
Stress is a process triggered by a demanding physical and/or psychological event [6]. It is not necessarily a negative process, but when present continuously it can result in chronical stress, which has negative health consequences such as raised blood pressure, bad sleep, increased vulnerability to infections, slower body recovery and decreased mental performance [7]. Regarding the economic costs of stress, the European Commission projected the costs of work-related stress to €25 billion a year. This is because work-related stress leads to increased number of injuries, increased absenteeism and decreased productivity [3]. Therefore, a stress-detection system would be useful for self-management of mental (and consequently physical) health of workers [5], students and others in the stressful environment of today's world.

To develop a stress-detection method, we must better understand the stress process. When humans undergo a demanding physical and/or psychological event (e.g., meeting, exam, etc.), the body is faced with a large physiological stressor which invokes a response of the sympathetic nervous system to meet the increased metabolic demands [1]. The sympathetic nervous system speeds up certain processes within the body ("fight-or-flight" response) [2]: it raises the heart rate, raises sweating rate, increases blood pressure, etc. All of these changes are controlled by the brain, thus it is expected for the brain's activity to be changed in a stressful situation. In addition, it has been reported that there are neuro-chemical changes due to stress response [10].

Electroencephalography (EEG) is a method for monitoring the electrical activity of the brain. It reflects the upper cognitive functions and mental or psychological states. In general, increased activity in the higher frequency bands is correlated with arousal, thinking or emotional activity. For example, the midrange beta frequency band (16–20 Hz) is considered to be correlated with emotional "intensity" that may come from anxiety [20]. In contrast, a lower spectral activity, e.g., activity in the delta frequency band (0.1Hz to 3Hz) is correlated with deep sleep.

The recent technological advancements have made cheap commercial EEG devices generally available. Depending on the quality of the device, they may capture the contrasting sleep-thinking working modes of the brain or possibly even stress/no-stress situations. In this study, we explore the use of a single channel EEG device for stress detection in office-like conditions.

2. RELATED WORK
Even though there are numerous studies on stress using psychology instruments or hormones analysis, the number of stress studies using electroencephalogram (EEG) for stress detection is quite low. This indicates the complexity of the problem [19]. Furthermore, the use of a commercial one-channel device makes the problem of stress detection even more challenging.

Peng et al. analyzed methods for identifying chronic stress by a medical 3-chnnel EEG device. They used statistical analysis to distinguish between relaxed state and chronical stress. They have concluded that there are different hemispheric activities between left and right hemispheres for the no-stress and stress subjects. Similar findings were presented by Giannakakis et al. [19] who were detecting stress in subjects while watching videos using a 32 channel EEG device. They have asserted that EEG response to positive valence stimuli is left frontal activity, while negative valence stimuli cause an increase to right activity. However, these changes cannot be captured with a single channel EEG device.

Recently, Saeed et al. [21] presented a study for stress detection using a one channel EEG device. They tried to distinguish between high and low stress based on a 3 minute data labelled using a Perceived Stress Scale questionnaire. This questionnaire gives information about stress levels accumulated over longer period of time since it contains questions such as: "In the last month how many times …".

In our study, stress is induced using a standardized method, and it is assessed using State-Trait Anxiety Inventory (STAI) questionnaire, which provides momentary information. Thus, we

are trying to detect different stress levels over a short period of time (e.g., 25 minutes).

In the literature there are other approaches for stress detection using wearable devices which capture physiological signals such as heart rate, sweating rate, R-R intervals, skin temperature and similar [22][23][24]. However, these approaches are out of the scope of this study.

3. DATA

A standardized stress-inducing method was used for collecting the experimental data. A web application was developed in collaboration with psychologists, which implements a variation of the stress-inducing method presented by Dedovic et al. [4]. The main stressor is solving a mental arithmetic task under time and evaluation pressure. In short, a series of randomly generated equations were presented to subjects, who provide answers verbally. The time given per equation was dynamically changing. For each two consecutive correct answers the time was shortened by 10%, and for each two consecutive wrong answers the time was increased by 10%. Each session consisted of three series of equations with increasing difficulty: easy, medium and hard. Each series of equations lasted for five minutes. For motivation, a reward was promised to the top three participants. After each stage, the participant was shown false ranking score, positioning him/her in the top five, and this way motivating him/her to try harder in the next stage and try to win the award. The application is available on-line: http://dis.ijs.si/thestest/.

During the experiments, the subjects were monitored with a NeuroSky's MindSet as presented in Figure 1. It is a wearable EEG that provides raw EEG data with a 512Hz sampling frequency by an electrode placed on left side of the forehead (at the FP1 location). In addition, the device provides computed 1Hz data for 7 frequency sub-bands, information for meditation intensity, information for attention intensity and information for detected blinks.

Figure 1. Experimental setup. Subject 1 performing the stress inducing experiment.

Four Short STAI-Y anxiety questionnaires [8] were filled by each participant: before the experiment (1), and after the easy (2), medium (3) and hard session (4). The STAI questionnaire reflects the degree to which the subjects have a feeling of unease, worry, tension, and stress. Its value range is from 0 to 24, with 24 being the highest level. The mean STAI score is presented in Table 1. We performed statistical analysis using repeated measures ANOVA [9]. The resulting p-value was 0.0014, confirming that there is a statistically significant difference in the answers for the intensity of stress. The answers of the STAI questionnaire were used for subject-specific labelling of the data.

Table 1. Mean STAI score for the experimental data.

Type	Before	Easy	Medium	Hard
STAI score	10.95	13.33	14.05	13.81

For each subject, the period before answering the STAI questionnaire in which they achieved the lowest score is labelled as low stress, and for each +3 STAI points (the statistical tests showed that difference of 2.38 is enough), the stress label is increased by one, thus we get low stress (lowest STAI score), medium stress (lowest STAI score +3) and high stress (lowest STAI score +6). In the final experiments the medium and high stress were merged because only two subjects achieved a high level of stress, so we had two degrees of stress low and high. The overall duration of the low stress data was 840 minutes and the duration of the high stress events was 724 minutes.

4. METHOD

The proposed method is a machine-learning pipeline presented in Figure 2. During the experiments, the data was streamed to a smartphone via Bluetooth, then transferred to a computer where the rest of the processing was performed. The method contains several phases: filtering, segmentation, feature extraction and model learning. In the next subsections, each stage is described in details.

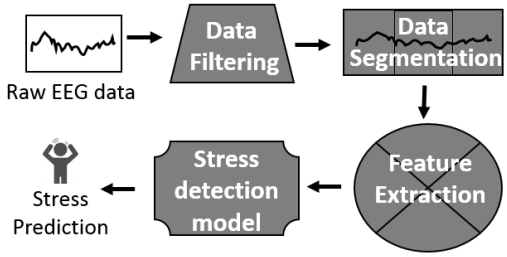


Figure 2. The proposed method for stress detection from EEG.

4.1 Filtering and Segmentation

After a visual inspection of the data, it was obvious that the data contains noisy segments which needed filtering. For detecting outliers (noise) in the data we used Tukey's method of leveraging the interquartile range [11]. This method is independent of distributional assumptions and it ignores the mean and standard deviation, thus it is resistant to extreme values in the range. After the detection, each outlier is replaced with the average value of the two neighboring samples. Figure 3 presents an example data of one subject. On the x-axis is the time in seconds and on the y-axis is the amplitude of the high beta band (20-30 Hz). The bottom graph is the raw data as provided by the device and the top graph is the same data after filtering.

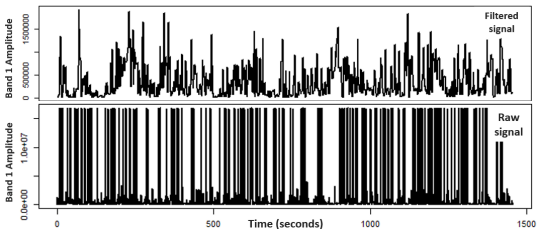


Figure 3. Data of one experimental session for Subject 3. A raw signal (bottom) and the same signal after filtering (top).

4.2 Feature extraction

In the feature extraction phase, a sliding-window technique was used. From each window, numerical features were calculated using statistical functions (mean, variability, quartiles, quartile deviation and power) and regression analysis (intercept and slope of a fitted regression line). The features were computed from each data type provided by the device (raw EEG data downsampled to 4Hz because the high frequency does not influence significantly the computed features, frequency bands' data, meditation data, attention data and blinks data). In addition, blinks per second, significant blinks per seconds and power of significant blinks was computed from the blinks data. A blink was considered significant if its value was over some threshold (80), which was chosen experimentally.

4.3 Model learning

For the model learning we used the machine learning algorithms as implemented in the WEKA machine learning toolkit [25]. We experimented with variety of ML algorithms:

- **Majority classifier** – always predicts the majority label. This is a baseline model.
- **J48** – algorithms for building a decision tree (DT) [12].
- **Naïve Bayes classifier** – algorithm for building simple probabilistic classifiers based on Bayes' theorem with strong independence assumptions between the features [13].
- **KNN** – simple algorithm which provides a prediction based on k-nearest instances (from the training set) to the test instance 0.
- **SVM** – algorithm for building a classifier where the classification function is a hyperplane in the feature space [15].
- **Bagging** – ensemble algorithm which learns base models on subsets of train data with the purpose of reducing variance and avoiding overfitting [16]. The algorithm for learning base models was J48.
- **Boosting** – ensemble algorithm which learns models on subsets of the train data and "boosts" the weights of misrecognized instances allowing for the models in the ensemble to focus on the misclassified instances [17]. The algorithm for learning base models was J48.
- **Random Forest** – ensemble method which learns base decision trees by subsetting the feature set [18].

5. EXPERIMENTAL RESULTS

We performed experiments with a varying size of the sliding window used in the feature extraction phase (size of the feature-extraction window). We experimented with a window size of 0.5, 1, 2 and 4 minutes. For evaluation we used 10-fold cross-validation technique. The results are presented in Table 2. Each row represents the accuracy achieved by the specific classifier and each column represent the size of the feature-extraction window.

The results show that the feature-extraction window size does not influence significantly the performance of any of the classifiers. The worst performing classifier is Naïve Bayes. This is due to the correlation of the features in the dataset. A high correlation is expected since all of the signals from which the features are extracted come from one source, the raw EEG data. Naïve Byes is well known for performing poorly on correlated datasets due to its "naïve" assumption that features are not correlated. On the other hand, the best performing classifier is Random Forest. This is due to its noise robustness. The highest achieved accuracy is 71% which is for 19 percentage points better the majority classifier.

Table 2. Accuracy (%) for each of the ML methods for a varying size of the feature extraction window.

ML Classifier	Feature-extraction window (minutes) 0.5	1	2	4
Majority	52	52	52	52
Naïve Bayes	55	53	53	54
Boosting	57	57	60	55
J48	60	63	57	56
KNN (3)	62	61	64	66
SMO	64	64	64	63
Bagging	66	65	66	63
Random Forest	70	71	71	66

6. CONCLUSION AND DISCUSSION

We presented a study for stress detection with a one-channel commercial EEG device. For this purpose, a stress-inducing experiments were performed allowing us to gather EEG data for 21-subjects. This data can be utilized for studying the brain activity under stress, induced using time and evaluation pressure.

In addition, we presented a machine learning method for stress detection including segmentation technique, filtering technique and a feature extraction technique. The initial evaluation results are promising, but still there is a room for improvements.

An inter-study comparison with the related work is not directly applicable mainly due to the use of different devices, different datasets, different number of subjects etc. As an example, Vanitha et al. [26] presented a system for real time stress detection based on 14-channel EEG device. The accuracy of the proposed machine learning models varied form 70-90%. This clearly suggests that better results can be achieved with EEG devices that provide data from more channels.

The evaluation technique used in the experiments is a 10-fold cross-validation which is person-depended evaluation approach. We also experimented with a person-independed evaluation technique (leave-one-subject-out cross-validation) and the performance of the models (in terms of accuracy) was near the majority classifier. There may be several reasons for this:

- **Subject specificity**. The EEG data may be subject-specific because each person has a specific reaction to such stressor. The specific reaction may not allow to build general ML model trained on one set of subjects and tested on another set of subjects.
- **Noise**. The commercial EEG device may not be powerful enough to capture some of the subtleties which would allow building general ML model. Each recorded session may be different due to persistent noise in the data.
- **Advanced features**. The extracted features in this study are based on statistical and regression analysis. In future, more advanced features can be extracted to improve the models' performance and generalization.

7. REFERENCES

[1] A. Angeli, M.minetto, A. Dovio, P. Paccoti, "The overtraining syndrome in athletes: A stress-related disorder", Journal of Endocrinological Investigation, 2004, pp 603-612.

[2] W. B. Cannon. The Wisdom of the Body. 1932.

[3] Calculating the cost of work-related stress and psychosocial risks European Risk Observatory Literature Review [Online]. Available: https://osha.europa.eu/en/tools-and publications/publications/literature_reviews/calculating-the-cost-of-work-related-stress-and-psychosocial-risks, [Accessed 15.10.2014].

[4] K. Dedovic, R. Renwick, N. K. Mahani, and V. Engert, "The Montreal Imaging Stress Task : using functional imaging to investigate the ...," vol. 30, no. 5, pp. 319–325, 2005.

[5] Fit4Work Project, http://www.fit4work-aal.eu/

[6] H. G. Ice, G. D. James. Measuring Stress in Humans: A practical Guide for the field," Cambridge university press. 2007.

[7] S.C. Segerstrom, G.E. Miller. Psychological stress and the human immune system: a meta-analytic study of 30 years of inquiry. Psychological Bulletin 130, 2004, 601

[8] C. D. Spielberger. State-Trait Anxiety Inventory Anxiety. vol. 19. p. 2009, 1987.

[9] T. Eftimov, P. Korošec, B. Koroušic Seljak. Disadvantages of Statistical Comparison Of Stochastic Optimization Algorithms. Proceedings of the Bioinspired Optimizaiton Methods and their Applications, BIOMA 2016.

[10] McEwen, Bruce S., and John H. Morrison. "The brain on stress: vulnerability and plasticity of the prefrontal cortex over the life course." Neuron 79.1 (2013): 16-29.

[11] J.W. Tukey. "Exploratory data analysis." Addison-Wesely, 1977.

[12] J.R. Quinlan, "Improved use of continuous attributes in c4.5," J. Artif. Intell. Res., Vol. 4, pp. 77–90, 1996.

[13] R. Stuart, N. Peter, "Artificial Intelligence: A Modern Approach," 2nd Ed., Prentice Hall: New York, USA, 2003.

[14] D. Aha, D. Kibler, "Instance-based learning algorithms," Machine Learning Vol. 6, pp. 37–66, 1991.

[15] N. Cristianini, J. Shawe-Taylor, "An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods," Cambridge University Press: Cambridge, 2000.

[16] L. Breiman, "Technical Report No. 421," 1994.

[17] K. Michael, "Thoughts on Hypothesis Boosting," Unpublished manuscript, December 1988.

[18] H. Tin Kam, "Random Decision Forests," International Conference on Document Analysis and Recognition, Montreal, QC, Canada, pp. 278–282, 1995.

[19] G. Giannakakis, D. Grigoriadis, M. Tsiknakis, "Detection of stress/anxiety state from EEG features during video watching," IEEE Eng Med Biol Soc. pp. 6034-7, 2015.

[20] P.M. Lehrer, R.L. Woolfolk, W.E. Sime. "Principles and practice of stress management," Guilford Press, 2007.

[21] S. Saeed, S. M. Anwar, M. Majid, A. Bhatti, "Psychological Stress Measurement Using Low Cost Single Channel EEG Headset," International Symposium on Signal Processing and Information Technology, 2015.

[22] M. Gjoreski. "Continuous Stress Monitoring Using a Wrist Device and a Smartphone," Master thesis, Jozef Stefan International Postgraduate School, Slovenia, 2016.

[23] J. A. Healey, R. W. Picard, "Detecting Stress During Real-World Driving Tasks Using Physiological Sensors," IEEE Transactions on Intelligent Transportation Systems, 2005.

[24] K. Hovsepian, M. Absi, T. Kamarck, and M. Nakajima, "cStress : Towards a Gold Standard for Continuous Stress Assessment in the Mobile Environment," ACM Conf. on Ubiquitous Computing, 2015.

[25] M. Hall, et al. "The WEKA Data Mining Software: An Update," SIGKDD Explorations, Vol 11, Issue 1, 2009.

[26] V. Vanitha, P. Krishnan, "Real time stress detection system based on EEG signals." Biomedical Research, pp. 271-275, 2016.

DESIGN OF MOBILE SYSTEMS THAT ENABLE AN ACTIVE AND MEANINGFUL LIFE FOR DEMENTIA PATIENTS

David A. Fabjan
Ambient Intelligence Research Group
Department of Intelligent Systems
Jožef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia
Tel: +386 1 477 3865; fax: +386 1 4251038
E-mail: david.fabjan@ijs.si

ABSTRACT

The paper presents an outlook on two expert mobile health systems, proposed for the benefits of prolonged and meaningful living for people in the early stages of Dementia.

The research group has recently proposed such prototype systems to be built and hopefully used commercially. Both systems help people with mild cognitive impairments serving as memory aid in one case, and as a cognitive training tool in the second case, which are both presented in this exploratory paper.

The main goal of proposed prototypes is to enable and maintain an acceptable level of autonomy that would allow affected people to live an active life, despite the limitations imposed by the condition.

Keywords
Mobile systems, Dementia, cognitive impairments, ambient sensors, context aware systems, prediction, wearables, care giving.

1 INTRODUCTION

Dementia is one of the most frequent neurological disorders affecting the elderly population. It is estimated that in 2015, there were 46.8 million people living with dementia worldwide, with one new case every 3 seconds [1]. A disproportionally large number of them – 10.5 million – live in Europe, since it has much older population on average than other continents.

This spectrum of diseases with common name Dementia is characterized by a decline in one or more of the seven cognitive domains (learning, memory, language, executive function, attention, and perceptual-motor) from the previous (normal, functioning) level of functionality. Such decline is severe enough to interfere with normal daily activities and leads to personal dependence on external help even when performing basic tasks such as hygiene and daily diet.

Cognitive impairment may be determined by multiple neuropathological processes including neurodegenerative diseases, by far the most common being Alzheimer disease, and vascular disease. In fact, most frequently, cognitive impairment is the result of more than one disease process (degenerative and vascular) [1]. Cognitive impairment represents a progressive process that gets worse over time and has no foreseen completely effective cure.

The main goal for people affected by Dementia is the ability to maintain an acceptable level of autonomy that would allow them to live a quality life at home, despite the limitations imposed by the condition. Similarly, family caregivers and care professionals of nursing homes and day care facilities are in need of technological tools to help the people with Dementia develop a coping strategy concerning their illness.

The quality of life is associated with health, social relationships and ability of using resources such as shops, transportation and housing [3]. Those people suffering from dementia score poorly on all these criteria; they are frequently ill, have difficulties in maintaining social relationships, and are unable to utilise the resources at their disposal. To improve the quality of life, people with dementia need to maintain most of their personal independence, engage in social interaction, and must keep themselves occupied [4].

These factors are interrelated and there is no consensus about their relative importance, although some literature suggests that independence in the activities of daily living is the most important one [5]. To address these important factors, the research group in collaboration with the international team of medical and social experts on Dementia and specialists in “serious games” and human-computer interaction, proposed two intelligent mobile systems.

2 SMART COACH

First system is re-hashing an old idea of using short reminder notes (post-it, stickers), strategically placed to

remind us of important or frequent tasks. Generally, external memory aids are the most frequent strategy for dealing with the memory loss caused by Dementia [6]. Reminder notes are usually placed in various locations around the home – those for meal preparation in the kitchen, for personal hygiene in the bathroom, etc.

However, the reminder notes should also depend on what the person is currently doing: while putting clothes in the washing machine - on how to operate it, on how to use shampoo; when going to the kitchen in the morning - about making coffee, about drinking water, about taking medicine, etc. If one wanted to have a paper note for every occasion, though, they would become too numerous to be of use and it would become very confusing for the individual, particularly when trying to follow certain timeline of tasks.

The system intends to develop intelligent reminder notes that are context-sensitive. To identify the activity of the user, we have to use various types of ambient sensors to detect what the person is doing. In addition, we will be able to utilize a pre-programmed schedule of the person's normal and procedural daily activities (eating, personal hygiene, exercises) as well as learn individual's personal habits.

This combined can give us a very good idea about what exactly the person is doing and what the person wants or needs to accomplish at a given moment. This way we will be able to display just the right reminder with the correct steps to follow at the appropriate time.

2.1 TECHNOLOGY AND METHODOLOGY

The farfetched technology for the system would be smart glasses or even smart contact lenses that would sense and analyse the user's environment, interpret context and intent, and provide the appropriate reminders or procedural instructions directly in front of the eyes [7]. From proven technology, available in the market, the following options were selected:

Display: Tablets mounted on strategic locations in the user's home and smart glasses, almost indistinguishable from regular glasses;

Sensing: Camera and microphone in the tablet, we use a camera and perform vision and sound analysis, intended to detect the presence of the user, basic activities such as handling dishes, and environmental events; 3D sensing technology similar to Microsoft Kinect; a wristband with multiple sensors for user's activities recognition; RFID reader that can detect tags attached to various important objects in the user's home; localisation system using Bluetooth Low Energy beacons to detect the exact user's location.

The interpretation and reasoning from the data acquired will be done in three stages. The first stage will use machine learning, sensor fusion and heuristics to infer information such as the user's location (in the kitchen, on the sofa) and activity (cooking, resting). Most methods at this stage will run on the sensing devices (computer vision and sound analysis on the tablet) to reduce the communication overhead.

The second stage will create a global awareness of the user's situation involving all the sensing devices. Machine learning and sensor fusion are used to recognize only parts of the user's situation. Next step will involve decision-support methods such as proprietary hierarchical qualitative decision models and expert defined rules to completely determine the user's situation and the most probable intent. Algorithms for these methods will run remotely in the cloud.

The third stage will use symbolic reasoning based on an ontology and rules to select the most appropriate reminders and instructions based on the user's situation evaluated and inferred in the second stage. Methods in this stage will be also implemented in the cloud, returning the output to the display devices.

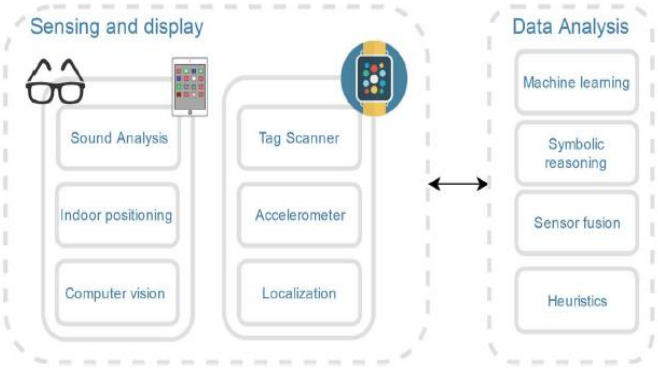


Figure 1. The Architecture of the smart coach system

2.2 EXPECTED IMPACT

Each person with dementia living at home is cared for by two or three carers on average. With the growing number of dementia cases, this "dependency ratio" is difficult to sustain, resulting in a high demand for care solutions, and representing a growing burden in terms of economic, social and human costs.

The main impact of the expert mobile system is to prolong the independence and living at home for people with mild and moderate dementia. We want to extend such living for at least two years before constant care and living in nursing homes is needed.

Staying in home environments has the smallest socio-economic cost, which is important due to demographic changes where the population is getting older.

3 PLAN AND EXECUTE DAILY ACTIVITIES

The second system proposed has the objective of cognitive training [8] to exercise the remaining cognitive abilities of people with mild cognitive impairment and moderate forms of Dementia to help them remain independent as long as possible.

While cognitive training cannot fully reverse the damage caused by the disease, it can slow it down considerably and even improve the abilities in specific areas. Research shows [3] that cognitive training is particularly effective when it is personalized, taking into account the person's specific situation and goals – this type of cognitive training is sometimes called cognitive rehabilitation. The training can address various cognitive functions: memory, attention, executive function and others. While there are several applications available for cognitive training of memory and attention, the executive function has been neglected, which is exactly what this system is aiming to address.

Executive function is the ability to execute tasks that involve planning and decision making. It helps us deal with the procedural tasks necessary to perform required set of activities in daily life that are not well rehearsed, such as shopping, preparing meals and visiting a doctor.

The core of the system developed in this project will be a serious game that will simulate such tasks, including various mishaps that can occur (forgetting a wallet while going shopping, the shop being out of an ingredient for the planned meal, taking a wrong bus when going to the doctor). This will enable the users to train and prepare themselves for these tasks in the safety of their own homes on a tablet or on a TV without the stress brought about by mishaps in the real world.

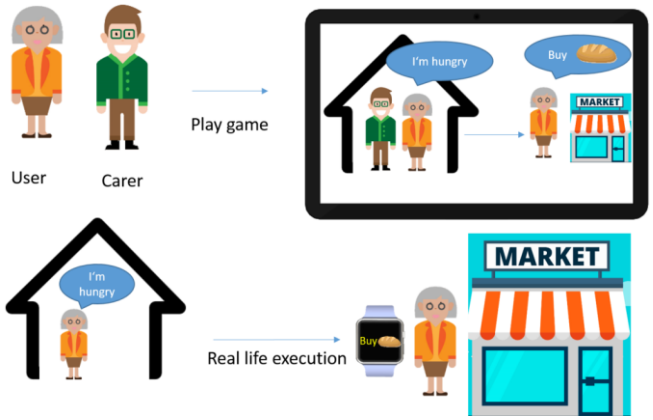


Figure 2. The concept of the serious game system

People with Dementia will have an effective way to evaluate which exercises and e-learning modules allow them to perform their activities better; family carers and care professionals will have a library of learning modules and games that will simplify their job in training the users to live well with dementia; the families of the person with dementia will be able to understand those activities in which their relatives find more difficulties to help them with the coping strategy.

3.1 TECHNOLOGY AND METHODOLOGY

The three main components of the prototype are a serious game for cognitive training of executive function, a mobile device companion to track the progress of the execution of the trained activities in real life, e-learning methods to train the users (people with Dementia and their carers) in the use of the system, and a back-end to connect all the components.

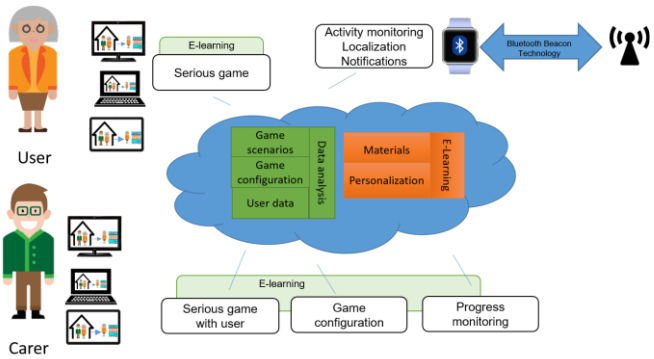


Figure 3. The Architecture of the 2nd prototype system

Back-end of the system supports the previous three components with: serious game scenarios defined during the gathering of user requirements; serious game configuration input by the care giver; user medical and psychological profile, preferences and the history of the use of the system; a module for advanced data analysis for the serious game executions in terms of personalisation, planning and analysis of the progress.

E-learning materials are using an existing technology OpenEDX. The OpenEDX will be extended with algorithms to render the learning path of the users dynamic and to consider their preferences, physiological values and behaviour.

3.2 EXPECTED IMPACT

Although a technological intervention cannot replace live social interaction, the system is expected to have a positive personal impact on the affected person's interaction with friends and care givers.

The system should increase a quality of life and ease the burden of family and care givers. The care givers will be able to interact with the users remotely trough the cognitive-

training game in a playful manner, re-acting the scenarios in home environments.

4 INVOLVEMENT OF THE END USERS

The end-users involvement and systems adaptation to their needs is crucial for the success of the proposed systems. The target groups involve people with mild to moderate dementia, formal carers such as medical, pharmaceutical and outpatient services and care givers from dementia institutions, and private subjects, and informal care givers.

We will investigate the requirements of the target groups, primary, secondary and tertiary end users. User requirements will be collected separately for each of the implied groups, as the motivations, needs and attitude towards the proposed technology will vary.

The testing follows three principal objectives including: exploratory studies regarding user-requirements (both primary users and formal and informal care givers); input and feedback in the development process, including the review of testing of prototypes in pilot studies; and finally the retrieval of input for the further development and potential commercialization.

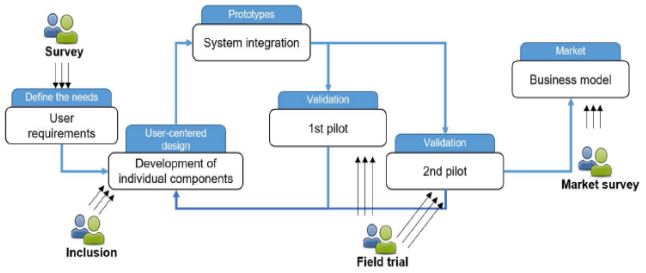


Figure 4. The involvement of the end-users

Testing is an inclusive, user-centred approach. The involvement of end-users shall be iterative through multiple cycles of design, testing and adaptation of a prototype.

5 CONCLUSION

In this exploratory paper we provide an insight into the parts of past year's research and work by the ambient intelligence group. These ideas and results are possible when combining technology and methods with initiatives and in-depth knowledge by the experts form health and social domains.

Mobile pervasive systems and devices have seen tremendous advances in recent decades. Ambient intelligence is becoming more and more pervasive with every new generation of sensors with the added modalities and advances in miniaturisation processes. Equipped with an array of sensors and powerful processing hardware that can support sophisticated machine learning algorithms, devices such as smartphones, can build predictive models of the context. Based on these models, actionable decisions that impact the future state of the system and/or its environment can be made. [9]

6 ACKNOWLEDGMENT

This work would not be possible without the insights and thoughtful discussions with Dr. Mitja Luštrek, Head of the Ambient Intelligence Research Group and Božidara Cvetković, Department of Intelligence Systems at JSI. Both have contributed with excellent knowledge on recent advances in the domain of ambient intelligence and possibilities of mobile computing. I would like to thank Martin Frešer, the young researcher in the group, for helping to finalize the prototypes' design.

Special thanks go to all the external experts for all their invaluable advice and their contribution. I especially wish to express my gratitude for contribution of Slovene experts: prof. Zvezdan Pirtošek, Head of Neurology Division at University Medical Centre Ljubljana, and Štefanija Lukič Zlobec, the President of the Slovenian Association for Help with Dementia – Spominčica, Alzheimer Slovenia.

7 REFERENCES

[1] Alzheimer's disease International. *World Alzheimer Report 2015: The Global Impact of Dementia.* (2015)

[2] Sonnen JA, Larson EB, Crane PK, et al. *Pathological correlates of dementia in a longitudinal, population-based sample of aging*. Ann Neurol. 2007 Oct. 62(4):406-13

[3] Keating N, Gaudet N. *Quality of life of persons with dementia*. The Journal of Nutrition, Health & Aging 16(5). (2012)

[4] Moyle W, Fetherstonhaugh D, Greben M, Beattie E. *Influencers on quality of life as reported by people living with dementia in long-term care: A descriptive exploratory approach.* BMC Geriatrics 15. (2015)

[5] Andersen CK, Wittrup-Jensen KU, Lolk A, Andersen K, Kragh-Sørensen P. *Ability to perform activities of daily living is the main factor affecting quality of life in patients with dementia. Health and Quality of Life Outcomes 2*(52). (2004)

[6] Beard RL, Knauss J, Moyer D. *Managing disability and enjoying life: How we reframe dementia through personal narratives*. Journal of Aging Studies 23(4). (2009)

[7] Web page https://www.wareable.com/wearable-tech/best-smart-contact-lens (September, 2016)

[8] 5 Bennett DA, Schneider JA, Arvanitakis Z, et al. *Neuropathology of older persons without cognitive impairment from two community-based studies*. Neurology. 2006 Jun 27. 66(12):1837-44

[9] Pejovic V. *Roadmap to Anticipatory Mobile Computing*, Elektrotehniški vestnik 81(5): 247–250, 2014

SeaAbs: developing a low cost seafloor observatory

Aleksandar Tosić
University of Primorska, FAMNIT
Glagoljaška 8
Koper, Slovenia
aleksandar.tosic@upr.si

Matjaž Kljun
University of Primorska, FAMNIT
Glagoljaška 8
Koper, Slovenia
matjaz.kljun@upr.si

ABSTRACT
Emphasis on ocean science in the past two decades has led to an increased demand for maritime observation and data collection. Such data is commonly collected by specially designed maritime observatories which are used by various stakeholders (e.g. civil society, NGO, government agencies, policy makers, researchers) for various purposes (e.g. environmental protection, development of sea infrastructure, bolstering tourism, climate change prediction). Most seafloor explorations have so far been completed either using remotely operated underwater vehicles (ROV's) or static observatories placed on the sea floor. The latter are arguably better suited for observing the dynamics of the oceans for longer periods of time. A lot of effort has also been invested into developing of on-line communication capabilities ranging from optic cables to underwater acoustic wireless communication systems. However, such systems demand large monetary investment, which is often an obstacle especially for above mentioned civil society, NGOs and researchers. In pursuit to make seafloor observation affordable for a larger society and to progress the ocean science we have built a seafloor observatory as an open-source platform aiming to increase and speed-up sea-floor exploration, data gathering and data sharing.

Categories and Subject Descriptors
C.2.3 [**Computer systems organization**]: Embedded and cyber-physical systems—*Sensors and actuators*

General Terms
Seafloor Observatory Science, Seafloor observatory

Keywords
Seafloor Observatory, Maritime Data, Sea Exploration, Seafloor Observatory Science

1. INTRODUCTION
Oceans are still one of the least explored ecosystems despite the fact that they greatly influence our climate, and consequently environment and life on our planet. There has been a growing trend in the past decades of moving from exploratory research with underwater vehicles to in-situ observations [5]. Oceans are dynamic and best observed from a static point of view. Dynamic observation can affect the measurements, which introduces uncertainty as if the measurements are results of the dynamics of the observatory itself [9, 7, 8, 11].

Two of the key challenges of seafloor observatories are constant communication capabilities and power requirements. The most widely used approach today is to have a network of cables put in place [7, 3] in order to connect observatories to internet and power them. This makes such solution relatively expensive for general use [4]. Another approach to this problem is a buoy based design [2], which is usually making use of satellite transmission. However, such solutions can limit the possible placement location greatly and there is still a power supply in question. The later is also true for all acoustic based communication systems.

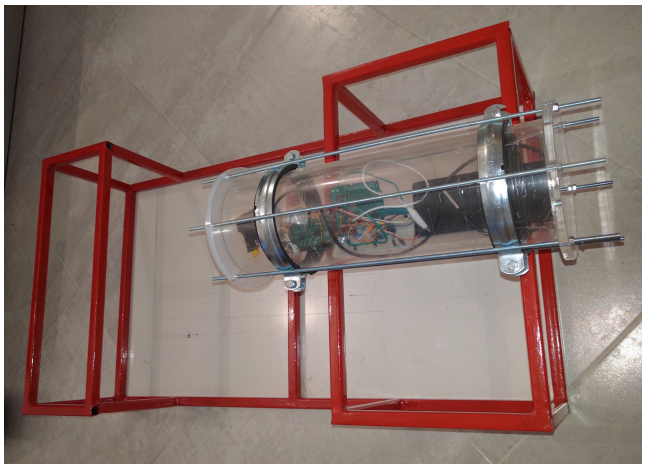


Figure 1. SeaAbs: a low cost seafloor observatory before materials are mounted on

To our knowledge, with the exception of bouy Vida operated by Marine Biology Station Piran (described in e.g. [13]), there are no other initiatives directed towards collecting ocean data with the scientific objective of long-term monitoring of environmental processes in Slovenian waters. This is the case despite an increasing worldwide demand for such time-series data sets. Such measurements do not necessarily need constant power supply and networking capabilities, which we took into consideration while building our system. In this paper we describe our off-line approach suitable for obtaining time-series data sets where analysis can be done subsequently. To achieve this, we designed a prototype that was built using available of-the-shelf components.

2. MOTIVATION

Our motivation comes from the need to understand how different materials introduced to marine environments affect its wildlife. One such example are man-made structures introduced to the underwater forming artificial reefs. These are placed usually on the featureless bottom seafloor to promote wildlife or prevent erosion. Artificial reefs are most commonly created by scuttling ships, oil rigs, or are purposely built using construction debris, tires, concrete or plastic (e.g. PVC). These materials are likely to cause pollution by releasing substances and (micro)particles not naturally found in sea environment and often become part of the sea-life food-chain [6].

As inferred above, material interference with live (marine) ecosystems must be done with great care. For this purpose we built an observatory that is designed to monitor how different materials are absorbed into marine environment. The purpose of our observatory is to identify appropriate materials for artificial reef constriction and provide guidelines as to which materials are better absorbed by marine environment than others. In addition we will make all collected data freely available for others to carry out further research. In this paper we describe hardware, system architecture and software used in pursuit of these goals.

3. HARDWARE AND SYSTEM ARCHITECTURE

The system design can be seen in Figure 2. The base of the observatory measures 120 cm by 80 cm and serves as a weight that keeps the observatory in place on the sea floor. The frame of the observatory is made of steel (marked with blue color on Figure 2) where the left part holds different materials for which environment absorption is being measured. On the right part of the frame we have mounted a watertight custom built containment unit made of transparent PMMA thermoplastic in a form of a tube that holds all the electronics of the observatory. The tube can withstand pressure of 3.5 bar (equivalent to pressure at the depth of 25 meters). This pressure capability is sufficient for planned observations. However, tubes withstanding higher pressure can be mounted as well. The tube is locked on the frame with two clamps and can be easily removed for data reading and battery changing.

The electronics' system architecture was designed with adaptability, affordability and extensibility in mind, and it consists of three main modules, namely: the power supply module, the power regulator module, and the data logging module (see Figure 3). The modules do not require intermodule communication.

One of the main challenges of supporting off-line long-term observations was to keep the battery powered for as long as possible. Our power supply consists of an array of Lithium cells forming a power bank. The power bank outputs a regulated $5V$ at $2A$. The total capacity of the power bank is estimated at $50000mAh$. The high capacity is needed to achieve the desired operating time considering the expected temperature at the seabed (which can drop to 5 degrees Celsius). The temperature is expected to have a great impact on the battery capacity [1] as can be seen in Figure 4.

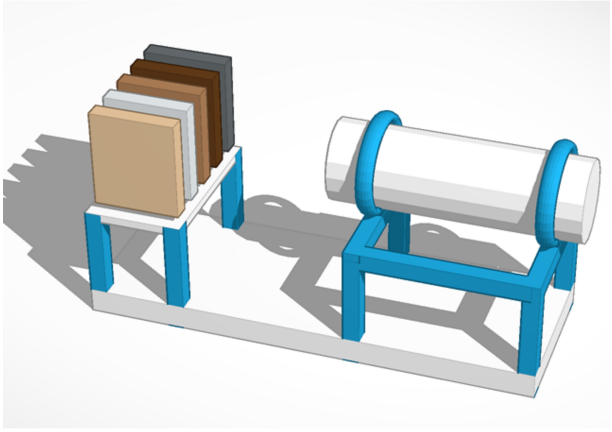


Figure 2. SeaAbs seafloor observatory: on the right is the tube holding all necessary hardware including sensors and camera, while on the left are placed different materials

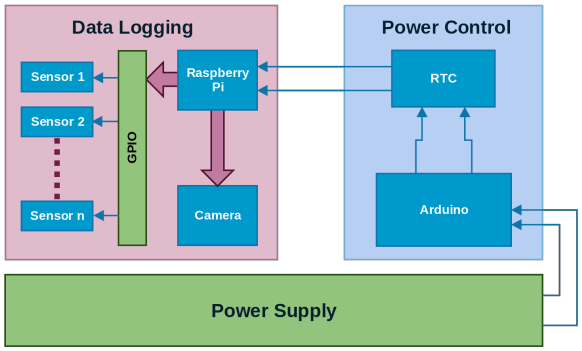


Figure 3. System Architecture

To save the energy the system is turned off most of the time and boots up automatically on pre-configured intervals. Battery life can be increased by prolonging the wake-up intervals, hence it is very important to calculate the time interval depending on the duration of the experiment. However, as explained above, the power bank and SD card can be replaced at regular intervals by removing the tube from the frame and taking it out of the water, whilst keeping the observatory underwater. This enables us to conduct uninterrupted observations for longer.

The Power Control module consists of an Arduino and an RTC (Real-Time Clock) shield with a dedicated $3V$ battery for maintaining the RTC – hence it does not drain the power from the power bank while in sleep mode. Once the RTC triggers a wake-up signal, the Arduino turns on the Data Logging module by connecting it to the main Power Supply.

Once the Data Logging module is active, the Raspberry Pi boots up and immediately reads the values from the sensors and writes all the data to an SD card. After a successful write the module shuts down by it self, ready to be turned on again at the next interval. Reading sensors can also be achieved with the Arduino board. However, in our case we

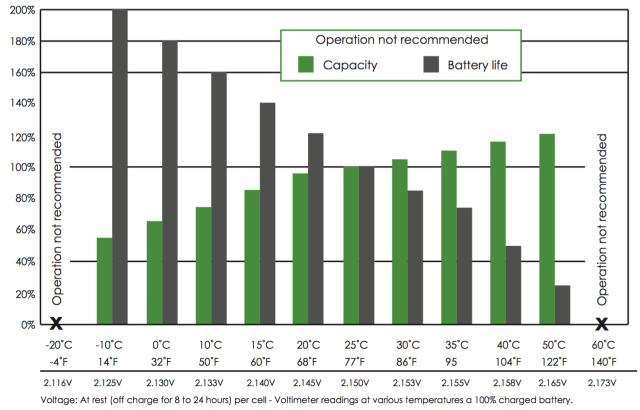


Figure 4. Battery capacity & battery life compared at different temperatures [1]

needed a good camera module to take photos of the materials observed in front of the tube, which Arduino lacks. Using Raspberry PI also expanded the GPIO pins available and provided several ways to build upon the existing platform. An abundance of modules and shields with plug & play capabilities are available for both platforms. Adding new sensors and monitoring capabilities is therefore very simple.

4. SOFTWARE

For the described scenario to work, we use a stock operating system Raspbian that comes pre-installed on Raspberry Pi. It is a Linux distribution, which allowed us to take advantage of its runlevels. These are the states of the system, indicating if the system in is the process of booting, rebooting, single-user mode, shutting down, or in normal multiuser modes. When runlevel changes, *init* runs so called *rc scripts* that take care of appropriate tasks for that runlevel. After all the normal services start at the end of the process of switching to multiuser mode, the *init* executes */etc/rc.local* to start custom services.

Our custom service is written in Python and follows the following procedure, based on the sensors connected to GPIO and the camera with the flash we use to take time-series of photos:

- Reads the temeprature from the Waterproof DS18B20 Digital temperature sensor
- Reads the the luminosity (Lux) from the TSL2561 Digital Luminosity/Lux/Light Sensor
- Turns on the LED flesh made out of an array of ultra bright LED's.
- Snaps a photo with of the illuminated observed materials (we have placed the lens of the camera on the front tube cover facing the observed materials in order to avoid glare)
- Turns off the LED flash
- Writes a new line to the measurements file (on SD card) with a timestamp and stores the image with the same timestamp.
- Initiates a graceful shutdown

Besides this simple Python script run in rc.local on the Raspberry Pi, we have written a simple script for Arduino PCB that regulates power supply to the Raspberry Pi. The purpose of the script is to ensure that the system does not drain unnecessary power after the shut down by cutting off the power supply to Raspberry Pi after two minutes. The purpose of the script is also to forcefully shut down the Raspberry if the shutdown process of the Raspberry Pi hangs during the shutdown process.

All the code for both Raspberry Pi and Arduino is freely available in our Git repository `http://git.redmine.pint.upr.si/absorbcija-tujkov-v-morskem-okolju`.

5. CONCLUSIONS

By using mostly off-the-shelf open-source software and hardware we designed and build a seafloor observatory system that is robust and reliable enough for obtaining time-series data sets for subsequent analysis after the system is retrieved from the seafloor (usually just the tube). Compared to other seafloor observatories, our SeaAbs costs just a fraction of their price (less than 1800€ or $2000). This makes it affordable for wider public and the research community enabling large scale marine environment data collation. Effective data collection tools are indispensable for maritime research filed, such as: marine biology, sea ecology, physical oceanography, and conservation biology[3].

We have currently built two observatories and are testing them before deploying them into North Adriatic sea. Both observatories hold different materials for observing their absorption into marine ecosystem. We are focusing on materials that are abundantly available, comparatively inexpensive, and not too invasive in terms of releasing substances and (micro)particles into the sea. In the near future we are planing to deploy both observatories in two different locations observing 5 different materials on each. Current system lacks real-time data processing capabilities which we plan to add in future versions. Another limitation of our observatory is also the fact that all components are placed underwater, hence can only be reached by scuba divers (required when power bank and memory card need replacement). In future versions, we plan to look into ways of automating the retrial of the tank holding electronics.

Irrespective of aforementioned limitations, the project has a long-term scope. Human interventions and operations are causing global and regional environmental and climate changes that may dramatically modify the structures and functions of marine (eco)system. The altered resource potentials and ecosystem services in turn affect the livelihoods of the population depended on these ecosystems. Slovenia is a maritime country economically dependent on the state of the oceans. Therefore it needs to participate in active monitoring of the oceans through maritime data collection which may contribute to regional as well as global societal, economic, political, and cultural changes. Our prototype,

as an inexpensive open source alternative to more expensive seafloor observatories, is a step in this direction.

Furthermore, data that will be collected by our prototype (and similar systems) offers an excellent starting-point for different simulations and models of various marine ecosystem aspects - one such example could be to use a combination of computer vision techniques [12] (to transform the captured pictures and/or video feed into numeric features) and machine learning algorithms [10] (to build a mathematical model from the data) to model the affluence of various fish species to artificial reefs and thus improve our understanding of artificial reef dynamics in large marine and freshwater systems.

Acknowledgment

The initial stage of the project was funded by AdFutura through "Path to Practical Knowledge" programme, which was carried out with the support of the Ministry of Education, Science and Sport, and the European Social Fund.

6. REFERENCES

[1] Temperature effects on battery performance & life. `www.heliant.it/images/FV/ev_temperature_effects.pdf`, 2015.

[2] M. Chaffey, L. Bird, J. Erickson, J. Graybeal, A. Hamilton, K. Headley, M. Kelley, L. McBride, E. Mellinger, T. Meese, T. O'Reilly, W. Paul, M. Risi, and W. Radochonski. MBARI's buoy based seafloor observatory design. In *Oceans '04 MTS/IEEE Techno-Ocean '04 (IEEE Cat. No.04CH37600)*, volume 4, pages 1975–1984. IEEE, 2004.

[3] P. Favali and L. Beranzoli. Seafloor observatory science: A review. *Annals of Geophysics*, 49(2-3):515–567, 2006.

[4] P. Favali, L. Beranzoli, and A. De Santis. *Seafloor observatories: A new vision of the Earth from the abyss*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[5] S. Favali, P., Person, R., Barnes, C.R., Kaneda, Y., Delaney, J.R., Hsu. Seafloor Observatory Science. In *Proceeding of the 10th IEEE International Conference on Advanced Learning Technologies ICALT010*, 2010.

[6] L. Jacobson. Artificial Reefs, Sunken Ships, and Military Toxins. `http://www.toxipedia.org/display/toxipedia/Artificial+Reefs,+Sunken+Ships,+and+Military+Toxins`, 2011.

[7] T. Lado Insua, K. Moran, I. Kulin, S. Farrington, J. Newman, M. Riedel, G. Iturrino, W. Masterson, C. Furman, A. Klaus, M. Storms, J. Attryde, C. Hetmaniak, and D. Huey. The Successful Delpoyment of a New Sub-seafloor Observatory. In *American Geophysical Union Fall Meeting 2013*, 2013.

[8] T. Lado-Insua, K. Moran, I. Kulin, S. Farrington, and J. B. Newman. SCIMPI: a new borehole observatory. *Scientific Drilling*, 16(16):57–61, nov 2013.

[9] T. Lado Insua, K. Moran, I. Kulin, S. Farrington, M. Riedel, M. Scherwath, M. Heesemann, B. Pirenne, G. Iturrino, W. Masterson, and C. Furman. One year of data of SCIMPI borehole measurements. In *American Geophysical Union Fall Meeting*, 2014.

[10] T. M. Mitchell. *Machine Learning*. McGraw-Hill Education, New York, NY, USA, 1st edition, 1997.

[11] K. Moran, S. Farrington, E. Massion, C. Paull, R. Stephen, A. Tréhu, and W. Ussler. SCIMPI: A new seafloor observatory system. In *OCEANS 2006*, 2006.

[12] M. Nixon and A. Aguado. *Feature Extraction & Image Processing for Computer Vision, Third Edition*. Academic Press, Oxford, UK, 3rd edition, 2012.

[13] D. Turk, J. Book, and W. McGillis. pCO2 and CO2 exchange during high bora winds in the Northern Adriatic. *Journal of Marine Systems*, 117-118:65–71, may 2013.

Primerjava razdalj pri klasifikaciji in regresiji

Tomaž Stepišnik Perdih
Fakulteta za matematiko in fiziko, Jadranska ulica 19, in Institut Jožef Stefan, Jamova cesta 39, Ljubljana, Slovenija
tomaz.stepisnik@ijs.si

Panče Panov
Institut Jožef Stefan
Jamova cesta 39
Ljubljana, Slovenija
pance.panov@ijs.si

Andrej Bauer
Fakulteta za matematiko in fiziko, Jadranska ulica 19,
Ljubljana, Slovenija
andrej.bauer@fmf.uni-lj.si

Sašo Džeroski
Institut Jožef Stefan
Jamova cesta 39
Ljubljana, Slovenija
saso.dzeroski@ijs.si

POVZETEK

V članku je narejena primerjava uspešnosti algoritma k najbližjih sosedov ob uporabi različnih razdalj med vhodnimi podatki. Primerjana je uspešnost pri klasifikaciji in regresiji s standardno obliko vhodnih podatkov (n-terice z numeričnimi ali nominalnimi komponentami).

Natančnost za posamezne razdalje je bila dobljena z 10-delnim prečnim preverjanjem. S Friedmanovim testom je bila testirana hipoteza, da je algoritem z vsemi razdaljami enako natančen. V primeru zavrnitve hipoteze so bili pari razdalj primerjani še z Nemenyijevim testom.

1. UVOD

Valentin Gjorgjioski v svojem doktoratu [4] naredi obširno raziskavo o vplivu različnih razdalj na natančnost napovednega modeliranja. Posveti se predvsem razdaljam na strukturiranih podatkih (časovnih vrstah ter množicah), nekoliko manj pa razdaljam na najpogostejši strukturi: n-terici. V tem članku bodo primerjani različni načini agregacije razdalj na komponentah n-terice v skupno razdaljo.

Najprej bodo v drugem razdelku opisane razdalje, ki so bile primerjane. V tretjem razdelku je opisan algoritem k najbližjih sosedov in postopek različnih mer uspešnosti algoritma. Opisan je tudi način primerjave razdalj ter podatkovne množice, na katerih so bile primerjane. V četrtem razdelku so predstavljeni rezultati testiranja.

2. OPIS RAZDALJ

Vhodni podatki so bili podatkovne množice n-teric z numeričnimi ali nominalnimi komponentami. Razdalja med dvema n-tericama je bila dobljena tako, da so bile najprej izračunane razdalje med istoležečimi komponentami, nato pa te razdalje po postopku urejenostno uteženе agregacije (ang. *ordered weighted aggregation*)[6] združene v skupno razdaljo. Opisati moramo torej računanje razdalje med numeričnimi in nominalnimi komponentami ter postopek agregacije.

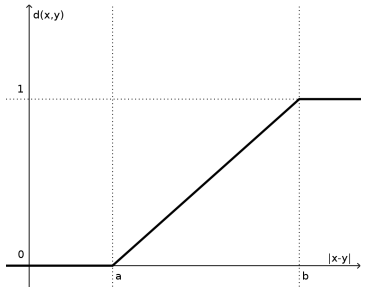
Za razdalje bomo zahtevali, da so simetrične (tj. $d(x,y) = d(y,x)$) in da slikajo na interval $[0,1]$. Za nominalne komponente uporabimo kar razdaljo, ki je 0, če sta podatka enaka, in 1 kadar nista.

$$d(x,y) = \begin{cases} 0, & \text{če } x = y \\ 1, & \text{sicer} \end{cases} \tag{1}$$

Pri numeričnih podatkih uporabimo absolutno vrednost razlike, vendar moramo poskrbeti še za omejenost na interval $[0,1]$. To naredimo tako, da razliko pod določeno mejo zanemarimo in preslikamo v razdaljo 0, od neke meje naprej pa obravnavamo kot maksimalno in preslikamo v razdaljo 1. Med tema dvema točkama pa linearno interpoliramo.

$$d(x,y) = \min(1, \max(0, \frac{|x-y|-a}{b-a})) \tag{2}$$

Vrednost parametra a je bila pri testiranju vedno 0, vrednost b pa odvisna od učne množice, kot bo opisano kasneje. Za lažjo predstavo je odvisnost razdalje od absolutne vrednosti razlike prikazana na sliki 1.



Slika 1. Numerična razdalja v odvisnosti od absolutne vrednosti razlike.

Za razdaljo med n-tericama najprej izračunamo n razdalj med istoležečimi komponentami z zgoraj definiranima razdaljama. Za skupno razdaljo potem izračunamo uteženo potenčno sredino razdalj med komponentami. Denimo, da imamo n-terici $x_1, \ldots, x_n$ in $y_1, \ldots, y_n$, pozitiven parameter p ter nenegativne uteži $o_1, \ldots, o_n$. Razdaljo med njima dobimo po naslednjih korakih.

1. Če je vsota uteži $O = \sum o_i$ enaka 0, je tudi skupna razdalja enaka 0. Sicer računamo naprej.

2. Izračunamo razdalje d_i med komponentami.

3. Razdalje med komponentami uredimo po velikosti. Dobimo $u_1, \ldots, u_n$ od najmanjše do največje.

4. Vrnemo vrednost

$$\left(\frac{1}{O}\sum_{i=1}^{n} o_i u_i^p\right)^{\frac{1}{p}}. \quad (3)$$

Utežem $o_1, \ldots, o_n$ pravimo urejenostne, saj utežujejo po načelu OWA [6].

Testirali smo 10 različnih razdalj, ki so se razlikovale večinoma v urejenostnih utežeh. Skice uporabljenih uteževanj so predstavljene na sliki 3. Vse razdalje razen evklidske so imele parameter p enak 1, pri evklidski pa je bil 2.

3. METODOLOGIJA

Pri poskusih smo vhodnim podatkom napovedovali njihove izhode s pomočjo učne množice. Delali smo tako klasifikacijo (nominali izhodi) kot regresijo (numerični izhodi). Za napovedovanje je bil uporabljen algoritem k najbližjih sosedov (kNN), znotraj katerega so bile uporabljene razdalje, opisane v prejšnjem razdelku.

Algoritem k najbližjih sosedov, opisan v algoritmu 1, danemu vhodnemu podatku poišče k najbližjih vhodov v učni množici in napove centroid njihovih izhodov. Pri klasifikaciji za izračun centroida naredimo uteženo glasovanje: vsak izmed k najbližjih vhodov glasuje za svoj izhod, utež pa je obratna vrednost njegove oddaljenosti od vhodnega podatka. Centroid je izhod z največ glasovi. Pri regresiji pa izračunamo uteženo povprečje izhodov, uteži pa so ponovno obratne vrednosti oddaljenosti. Če je oddaljenost katerega soseda 0, je njegova utež neskončna. V tem primeru vse neskončne uteži zamenjamo z 1, preostale pa z 0, da o centroidu odločajo enakomerno sosedi z oddaljenostjo 0.

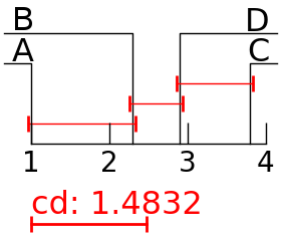
Algoritem 1 k najbližjih sosedov

Vhod: Učna množica T, razdalja na vhodih d, vhod x.
Izhod: Napoved za izhod od x.
1: Izberi k glede na T in d.
2: Izračunaj razdaljo d med x in vhodi v T.
3: Poišči izhode k-tih najbližjih vhodov.
4: **vrni:** Centroid teh izhodov.

Primerjali smo uspšenost napovedovanja z različnimi razdaljami v algoritmu kNN. Uspešnost smo merili z 10-delnim prečnim preverjanjem. Za vsako od desetih učnih množic je bil posebej določen parameter b pri numerični razdalji (enačba 2) in sicer kot razpon vseh števil na tej komponenti pri podatkih v učni množici. Potem je bil glede na učno množico izbran tudi parameter k. Z metodo "izloči enega"(ang. *leave-one-out*) dobimo napovedi za podatke v učni množici za k med 1 in 50, ter izberemo tisti k, pri katerem je napaka najmanjša (oz. natančnost največja).

Uporabimo mere uspešnosti iz programskega orodja WEKA[5]. Za klasifikacijo uporabimo utežena povprečja mer natančnost (ang. *precision*), priklic (ang. *recall*) in *f1* po razredih, za regresijo pa mere MAE (ang. *mean absolute error*), RMSE (ang. *root mean squared error*), RAE (ang. *relative absolute error*) ter RRSE (ang. *root relative squared error*). Napovedi naredimo za vsako mero natančnosti posebej, saj se lahko pri različnih merah izbrani parametri k razlikujejo.

Napovedi naredimo za več različnih podatkovnih množic, primerjava razdalj pa poteka kot je predlagano v [3]. Za vsako mero uspešnosti s popravljenim Friedmanovim testom testiramo hipotezo, da je algoritem k najbližjih sosedov z vsemi razdaljami enako uspešen. Če je hipoteza zavrnjena, z Nemenyjevim testom naredimo še primerjave posameznih parov razdalj. Vsi testi so bili narejeni s stopnjo zaupanja 0,05.



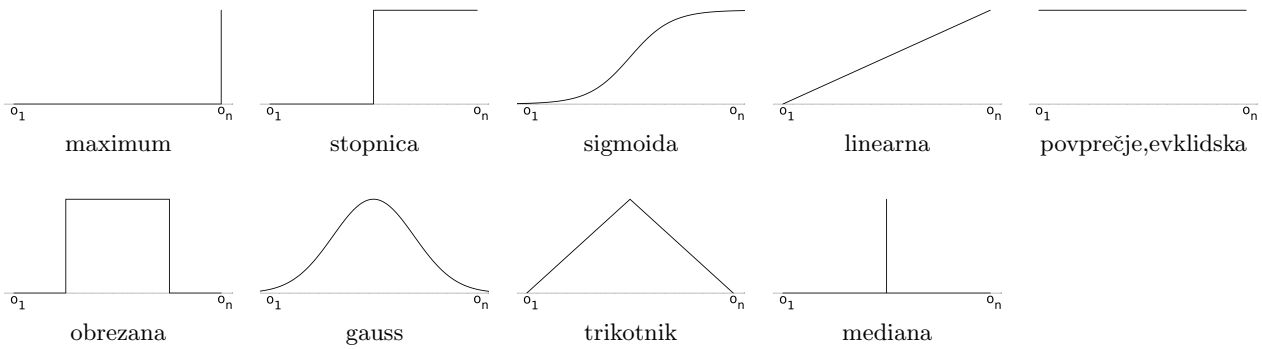
Slika 2. Primer rezultatov Nemenyijevih testov.

Rezultate predstavimo z grafom, primer katerega je na sliki 2. Na skali je označen povprečen rang razdalje, dosežen na različnih podatkovnih množicah. Črta pod skalo označuje, za koliko se morata povprečna ranga dveh razdalj razlikovati, da je razlika med njima statistično pomembna. Črte na grafu pa povezujejo razdalje, med katerimi ni statistično pomembnih razlik. V tem primeru lahko torej sklepamo, da je razdalja A boljša od razdalj D in C, ter da je B boljša od C. Med ostalimi pari pa ni pomembnih razlik.

Za regresijo smo uporabili 10 podatkovnih množic, katerih lastnosti so prikazane v tabeli 1. Klasifikacija je bila narejena v dveh delih. V prvem delu je bilo uporabljenih 10 podatkovnih množic z nizkimi dimenzijami vhodov (podobno kot pri regresiji), predstavljenih v tabeli 2. V drugem delu pa smo uporabili podatkovne množice s precej višjimi dimenzijami vhodov. Bilo jih je prav tako 10, opisane pa so v tabeli 3. Podatkovne množice z nizkodimenzionalnimi vhodi so bile pridobljene s spletne strani [2], množice z visokodimenzionalnimi vhodi pa s strani [1]. Izbira množic je bila pretežno naključna, z nekoliko ozira na čim večjo raznolikost

Tabela 1. Podatkovne množice za regresijo

podat. množica	primerov	atributov
quake	2178	3
pbc	418	18
housing	506	13
meta	528	21
strike	625	6
abalone	4177	8
autoMpg	398	7
hungarian	294	13
cpu	209	7
pharynx	195	11



Slika 3. Skice urejenostnih uteži.

v številu primerov, atributov, pri klasifikaciji tudi razredov.

Tabela 2. Podatkovne množice za klasifikacijo.

podat. množica	primerov	atributov	razredov
iris	150	4	3
diabetes	768	8	2
credit-a	690	15	2
car	1728	6	4
primary-tumor	339	17	22
autos	205	25	7
mfeat-morpho	2000	6	10
vote	435	16	2
tic-tac-toe	958	9	2
credit-g	1000	20	2

Tabela 3. Podatkovne množice visokih dimenzij za klasifikacijo.

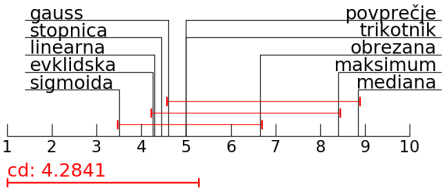
podat. množica	primerov	atributov	razredov
amlPrognosis	54	12625	2
bladderCancer	40	5724	3
breastCancer	24	12625	2
childhoodAll	110	8280	2
cmlTreatment	28	12625	2
dlbcl	77	7070	2
levkemija	72	5147	2
mll	72	12533	3
prostata	102	12533	2
srbct	83	2308	4

4. REZULTATI

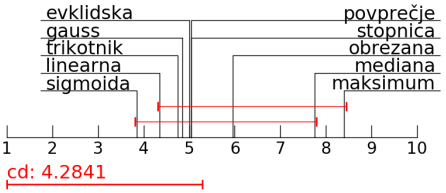
4.1 Regresija

Pri regresijskih podatkih je Friedmanov test hipotezo ovrgel za vse štiri mere natančnosti. To pomeni, da med različnimi razdaljami obstajajo razlike v uspešnosti. Te so opisane z rezultati Nemenyjevih testov, predstavljenih na sliki 4.

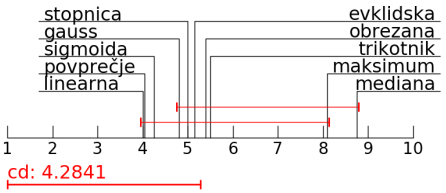
Pri meri MAE je sigmoida boljša od mediane in maksimuma, poleg tega pa so od mediane boljše tudi stopnica, linearna in evklidska. Pri RMSE je sigmoida boljša od maksimuma, pri RRSE pa linearna od mediane. Pri RAE so od mediane boljše linearna, sigmoida in povprečje. Če gledamo samo povprečne range lahko vidimo, da nobena razdalja nima precej večjega uspeha od ostalih, da pa sta ranga mediane in maximuma večinoma precej slabša od rangov ostalih razdalj.



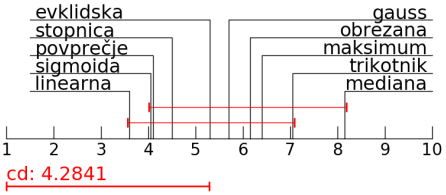
(a) MAE



(b) RMSE

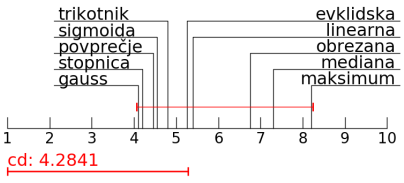


(c) RAE

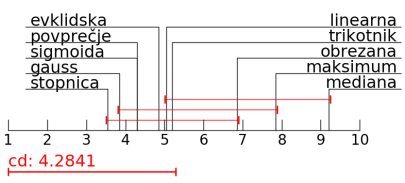


(d) RRSE

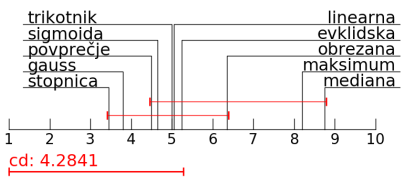
Slika 4. Diagrami povprečnih rangov z rezultati Nemenyijevih testov za kNN z različnimi razdaljami na regresijskih množicah.



(a) natančnost

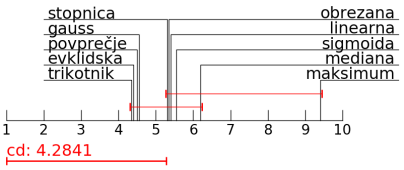


(b) priklic

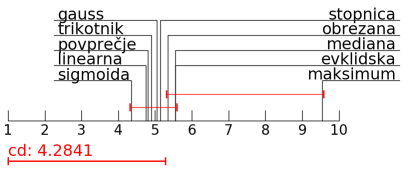


(c) f1

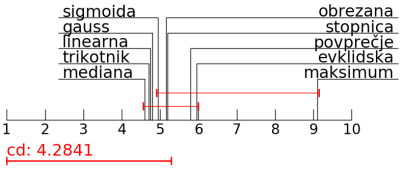
Slika 5. Diagrami povprečnih rangov z rezultati Nemenyijevih testov za kNN z različnimi razdaljami na klasifikacijskih množicah nizke dimenzije.



(a) natančnost



(b) priklic



(c) f1

Slika 6. Diagrami povprečnih rangov z rezultati Nemenyijevih testov za kNN z različnimi razdaljami na klasifikacijskih množicah visoke dimenzije.

4.2 Klasifikacija

Tudi tukaj je Friedmanov test hipotezo zavrgel za vse mere. Razlike v uspešnosti opisujejo rezultati Nemenyijevih testov na sliki 5.

Testi pri meri natančnost niso pokazali nobenih statistično pomembnih razlik. Pri meri priklic je stopnica boljša od maksimuma in mediane, od mediane pa so boljše tudi gauss, sigmoida, povprečje in evklidska. Pri meri f1 sta gauss in stopnica boljša od maksimuma in mediane. Na zadnjih treh mestih po povprečnih rangih so vedno obrezana, maksimum in mediana.

4.3 Klasifikacija podatkov visoke dimenzije

Friedmanov test znova hipotezo zavrne pri vseh merah. Razlike v uspešnosti opišejo rezultati Nemenyijevih testov na sliki 6.

Rezultati so precej drugačni kot pri podatkih nizke dimenzije. Pri vseh treh merah je sedaj le maksimum statistično slabši od prvih nekaj razdalj. Če pogledamo povprečne range, vidimo da je res daleč za ostalimi razdaljami, ki so tukaj bolj na kupu kot v prejšnjih primerih. Zanimivo je, da se mediana odreže toliko bolje (pri meri f1 celo najbolje), saj ravno tako kot maksimum za skupno razdaljo vzame le eno od več tisočih komponent.

5. ZAKLJUČEK

Primerjana je bila uspešnost algoritma k najbližjih sosedov ob uporabi različnih razdalj med podatki, v primerih ko so vhodni podatki n-terice z numeričnimi ali nominalnimi komponentami, ciljna spremenljivka pa prav tako numerična (regresija) ali nominalna (klasifikacija). Uporabljenih je bilo več mer uspešnosti, merjenih z 10-delnim prečnim preverjanjem, primerjava pa je bila narejena s Friedmanovimi in Nemenyijevimi testi.

Testi so pokazali večjo uspešnost razdalj, ki bolj utežijo velike razdalje med komponentami (z izjemo maksimuma, ki je ekstremen primer teh razdalj). V večini primerov ni bilo ene razdalje, ki bi bila bistveno uspešnejša od ostalih, pogosto pa sta mediana in maksimum odstopala v negativno smer.

Delo je predvsem podlaga za nadaljnje raziskovanje. Primerjava razdalj se lahko naredi tudi v primerih, ko so vhodni ali izhodni podatki strukturirani (zaporedja, množice, ...). Program, narejen za testiranje, omogoča algoritmu k najbližjih sosedov, da glede na trening množico podobno kot parameter k izbere tudi razdaljo, ki jo bo uporabljal za napovedovanje, izmed množice možnih. Več možnih razdalj načeloma poveča možnosti za dobro napovedovanje, vendar poveča časovno zahtevnost izbiranja razdalje. Zanimivo bi bilo na primer testirati, kateri nabori razdalj dobro delujejo skupaj.

6. VIRI

[1] Bioinformatics laboratory. http://www.biolab.si/supp/bi-cancer/projections/. Obiskano: 2016-06-24.

[2] UCI Machine Learning Repository. http://archive.ics.uci.edu/ml/. Obiskano: 2016-06-24.

[3] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[4] V. Gjorgjioski. *Distance-Based Learning from Structured Data*. PhD thesis, Mednarodna podiplomska šola Jožefa Stefana, Ljubljana, 2014.

[5] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA Data Mining Software: An Update. *SIGKDD Explor. Newsl.*, 11(1):10–18, Nov. 2009.

[6] R. R. Yager and J. Kacprzyk, editors. *The Ordered Weighted Averaging Operators*. Springer, 1997.

Avtomatska prepoznava teniških udarcev iz senzorskih podatkov

Miha Mlakar
Institut »Jožef Stefan«
Jamova cesta 39
1000 Ljubljana
miha.mlakar@ijs.si

POVZETEK

V tem članku smo prikazali postopek izdelave sistema za avtomatsko prepoznavo teniških udarcev iz senzorskih podatkov. Prikazali smo, kako je potekal zajem, kakšno je potrebno pred-procesiranje podatkov, izračun potrebnih atributov in nato gradnja modelov. Modele smo ovrednotili na dva različna načina in prikazali, da dobljeni model avtomatske prepoznave udarcev deluje zelo dobro.

Pridobljeni sistem je mišljen kot začetni del celovitega sistema za analizo teniške igre iz senzorskih podatkov. Prepoznavanje tipov udarcev je osnova iz katere bodo nato izhajale prihodnje analize.

Splošni izrazi

Algoritmi, meritve, testiranje, poskusi.

Ključne besede

Senzorske meritve, tenis, pospeškomer, teniški udarci.

1. UVOD

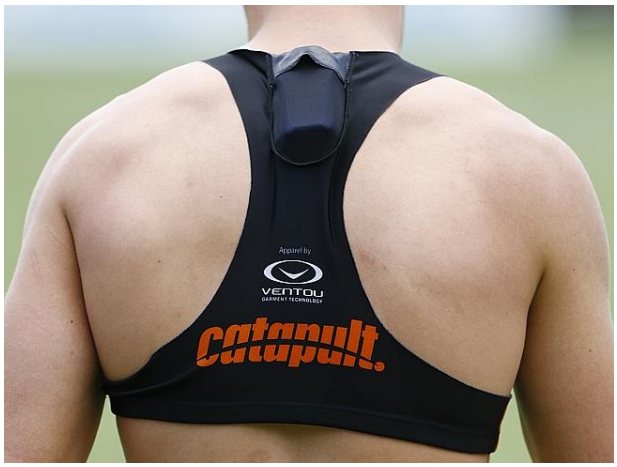
Uporaba oblačilnih senzorjev, ki jih športniki nosijo med športno vadbo je trenutno v velikem porastu. Različni senzorji in pristopi se uporabljajo v večini športov. Naprave, ki uporabljajo različne senzorje za izvajanje analiz in meritev v tenisu lahko razdelimo na tri skupine. Prva skupina so senzorji, ki se jih nosi na roki, oziroma so vgrajeni v teniški lopar. Ti senzorji nam prinašajo informacije o udarcih, vendar je ta informacija brez konteksta in je zato težko uporabna za izboljševanje teniške igre. Druga skupina so senzorji, ki se nosijo na hrbtu in poleg pospeškomerov in žiroskopov vsebujejo GPS sprejemnik za zaznavanje premikov igralca. Vendar pa se GPS izkaže za relativno nenatančnega, tako da so napake velike od 3-5 metrov, kar je glede na velikost teniškega igrišča preveč, da bi bile lahko meritve uporabne za določanje položaja na igrišču. Pole tega ti sistemi ne nudijo nobenih teniško prilagojenih meritev in analiz. Tretja skupina pa so sistemi, ki temeljijo na video analizi. Pri teh sistemih je potrebno postaviti 4-8 kamer okoli igrišča in potrebno je kupiti pripadajočo programsko opremo. Zaradi tega so ti sistemi izredno dragi in so vezani na igrišče in ne na igralca, tako da, če igralec igra na nekem drugem igrišču, mu analize niso na razpolago.

Kljub tem različnim možnostim pa na trgu trenutno ni senzorja, ki bi prepoznaval tako gibanje kot tenisko specifične metrike. Zato smo vzeli obstoječi senzor S5 podjetja Catapult Sports in ga uporabili za naše namene. Iz senzorja smo pridobili surove podatke, ki smo jih nato analizirali in izdelali avtomatski sistem za prepoznavanje teniških udarcev. Prepoznava udarcev nam bo služila kot izhodišče za vse nadaljnje analize teniške igre, zato je kvalitetna prepoznava udarcev zelo pomembna.

2. PRIDOBITEV PODATKOV

Za testiranje smo posneli šest različnih tekem s petimi različnimi igralci. Ker nas je zanimala avtomatska prepoznava udarcev v realnem okolju, smo vse posnetke posneli med tekmami, ne med treningi v predvidljivem okolju.

Igralci so imeli na sebi napravo S5 podjetja Catapult Sports, ki so ga nosili na hrbtu med lopaticami v zato namenjeni oprijeti majici (slika 1)



Slika1: Lokacija senzorja, ki smo ga uporabljali za meritve. Vir: sporttechie.com

Naprava S5 je vseboval naslednje senzorje:

- Tridimenzionalni pospeškomter (osveževanje s frekvenco 100HZ)
- Tridimenzionalni žiroskop (osveževanje s frekvenco 100HZ)
- Tridimenzionalni magnetometer (osveževanje s frekvenco 100HZ)
- GPS senzor, ki je vračal pozicijo v obliki zemljepisna širina in dolžina (osveževanje s frekvenco 10HZ)

Na ta način smo posneli za več kot 6 ur posnetkov. Ker so bili podatki pridobljeni 100 krat na sekundo smo tako pridobili 2.172.363 podatkovnih vnosov. V tem času smo zabeležili 1.373 udarcev. Vsak udarec smo označili kot eno izmed naslednjih možnosti:

- Servis
- Forehand

- Backhand

3. PRED-PROCESIRANJE

Na podlagi izrisanih signalov za pospeške in hitrosti obračanja smo najprej poskusili ugotoviti kaj so ključne značilke za diferenciacijo udarcev. Na sliki 2 lahko vidimo tipično sled pospeškomera in žiroskopa pri backhand udarcu.

Slika 2: Vrednosti pospeškomera in žiroskopa skozi čas pri backhand udarcu označenemu z navpično črno črto.

Ko smo sled žiroskopa pri backhandu primerjali s sledjo pri forhendu smo ugotovili, da se ključno razlikujeta po vrednostih kotnih hitrosti izmerjenih po osi 1 in 3. Na sliki 3 je prikazana razlika.

Slika 3: Razlika v vrednostih kotnih hitrosti za os 1 in 3 za žiroskop pri backhand in forehand udarcu.

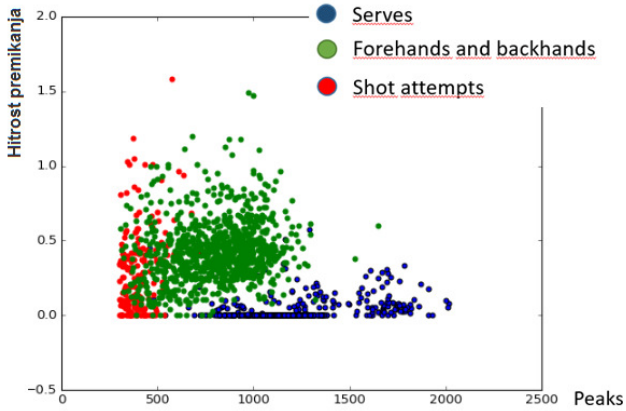
Izračunana razlika med vrednostma kotnih hitrosti za osi 1 in 3 žiroskopa pri udarcu nam je služila kot začetno izhodišče pri avtomatskem določanju tipa udarcev.

Postopek pridobivanja spremenljivke, ki nam pomaga pri odkrivanju ali gre pri neki kombinaciji senzorskih vrednosti za udarec sledi sledečim korakom:

- izračunamo novo spremenljivko *Spin*, ki nam predstavlja absolutno vsoto kotnih hitrosti pridobljenih z žiroskopom
- spremenljivko *Spin* potenciramo, da damo poudarek večjim vrednostim
- apliciramo Butterworth pasovni filter [1] na spremenljivko *Spin*
- negativne vrednosti popravimo na 0
- omejimo minimalno razdaljo med dvema vrhovoma (razdaljo smo eksperimentalno določili na 1,3 sekunde)

Na ta način dobimo spremenljivko, ki nam označuje potencialne udarce in jo poimenujemo *Peaks*. Ta spremenljivka seveda ni dovolj za prepoznavo ali se je nek udarec zgodil v nekem trenutku ali ne, nam pa pomaga pri identifikaciji. V gradnjo modela je potrebno vključiti še dodatne spremenljivke, kot so povprečne vrednosti in standardni odkloni posameznih parametrov ter vsote in razlike pospeškov in kotnih hitrosti po različnih oseh.

Kot primer je na sliki 4 prikazano, kako nam spremenljivki *Peaks* (prikazane so le vrednosti večje od 400) in hitrost premikanja pridobljena iz GPS podatkov razdelita prostor, kar nam pomaga pri identifikaciji udarcev.



Slika 4: Razdelitev prostora potencialnih udarcev glede na spremenljivki *Peaks* in hitrost premikanja.

4. TESTIRANJE

Testiranje smo razdelili na dva dela. Najprej smo testirali, kako točno lahko napovemo ali in kdaj se je zgodil udarec. V drugem delu pa smo poskusili napovedati tudi tip udarca – forehand, backhand ali servis.

Za gradnjo modela za napovedovanje udarcev smo v obeh primerih uporabili metodologijo naključnih gozdov (RF) [2]. Vzeli smo implementacijo v programskem jeziku Python implementirano v paketu scikit-learn. Metoda RF je vsebovala 10

naključnih dreves, ostalih parametrov pa nismo spreminjali, tako da so imeli privzete vrednosti.

Model smo testirali na dva načina. Pri prvem načinu smo za preverjanje točnosti modela uporabili metodo desetkratnega prečnega preverjanja [3], ki je za deljenje podatkov znotraj prečnega preverjanja uporabljala postopek uravnoteženega naključne delitve (Stratified shuffle split) [4]. Ta postopek zagotavlja, da so pri vsakem prečnem preverjanju razmerja med razredi v testni in učni množici enaka. To nam zagotovi pravilno povprečje rezultata čez celotno prečno preverjanje.

Pri drugem načinu smo testirali točnost modela po metodi navzkrižnega preverjanja s po enim izpuščenim igralcem, ki deluje po principu, da se model nauči na podatkih, ki ne vsebuje vrednosti izpuščenega igralca. Te vrednosti pa se nato uporabijo za testiranje točnosti modela. Ta postopek se ponovi tolikokrat, kolikor imamo različnih igralcev. Tak pristop nam omogoča vpogled v univerzalnost modela, saj nam njegovo dobro delovanje pove, da bo model uspešno napovedoval udarce tudi za igralce, ki jih nismo vključili v testiranje.

5. REZULTATI

5.1 Metoda prečnega preverjanja

Z metodo prečnega preverjanja smo klasificirali točnost napovedovanja forehand, backhand in servis udarcev.

Rezultati klasifikacije za forehand udarec:

- Natančnost 95.3%
- Priklic 91.4%

Rezultati klasifikacije za backhand udarec:

- Natančnost 94.3%
- Priklic 90.2%

Rezultati klasifikacije za servis udarec:

- Natančnost 99.1%
- Priklic 99.3%

Skupni povprečni rezultati preko vseh udarcev:

- Natančnost 96.2%
- Priklic 93.6%

5.2 Navzkrižno preverjanje s po enim izpuščenim igralcem

Z metodo navzkrižnega preverjanja s po enim izpuščenim igralcem smo dobili rezultate za natančnost in priklic ločeno za vsakega igralca.

Rezultati klasifikacije za forehand udarec:

- Natančnost: 91.5% [94%, 91%, 84%, 96%, 92%]
- Priklic: 90.5% [91%, 88%, 86%, 91%, 96%]

Rezultati klasifikacije za backhand udarec:

- Natančnost: 93.6% [98%, 98%, 84%, 91%, 95%]
- Priklic: 90.6% [96%, 78%, 85%, 96%, 95%]

Rezultati klasifikacije za servis udarec:

- Natančnost: 99.8% [100%, 100%, 100%, 99%, 100%]
- Priklic: 98.2% [98%, 98%, 100%, 99%, 95%]

Skupni povprečni rezultati preko vseh udarcev (in igralcev):

- Natančnost 95.0%
- Priklic 93.1%

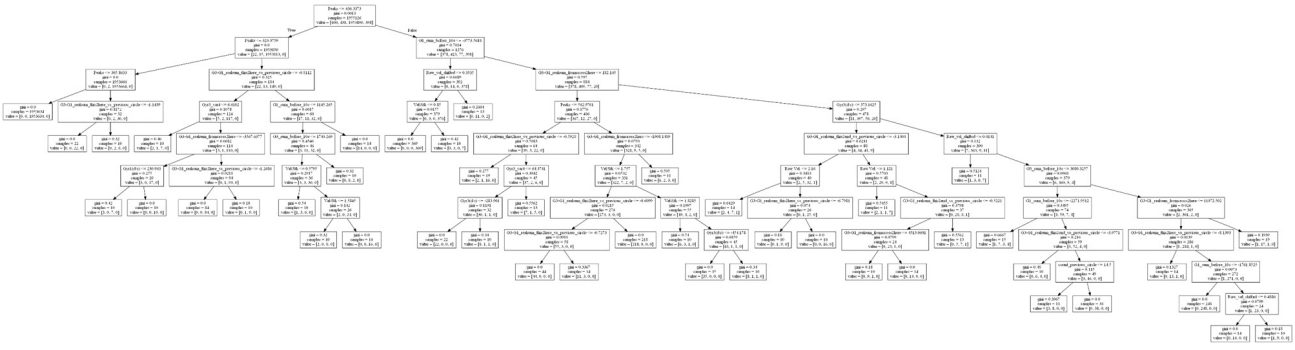
Kot lahko vidimo iz rezultatov sta natančnost in priklic pridobljena z metodama prečnega preverjanja in navzkrižnega preverjanja s po enim izpuščenim igralcem podobna. To pomeni, da je model relativno neodvisen od tipa igralca, oziroma od tehnike izvedbe udarcev in da model, ki smo ga naučili na ostalih udarcih dobro napoveduje tudi za izbranega igralca.

Glavni vir napak so posebnosti, ki se dogodijo med samo igro, kot je recimo udarec senzorja z loparjem, posebni gibi, ki jih ne moremo klasificirati med izbrane tri udarce in pa posebni primeri hitrega obračanja igralca okrog svoje osi, kar rezultira v podobnih vrednostih senzorjev, kot jih imamo pri udarcih.

Ker nam model zgrajen z RF ne omogoča prikaza kompleksnost smo zgradili odločitveno drevo za detekcijo tipov udarcev (slika 5). Ker je drevo preveč kompleksno za podroben prikaz, prikazujemo le shemo drevesa, da prikažemo strukturo in kompleksnost dobljenega modela. Kot lahko vidimo je drevo razvejano in uporablja veliko različnih atributov v vozliščih.

6. ZAKLJUČEK

V članku smo pokazali postopek za gradnjo modela za



avtomatsko detekcijo teniških udarcev iz realni tekem in ne le treningov v kontroliranem okolju. Vsak igralec je na hrbtu nosil posebno napravo s senzorji iz katerih smo nato s pred-procesiranjem in izdelavo atributov zgradili model, ki nam je služil za prepoznavo udarcev. Natančnost in priklic modela sta bila nad 93%, kar je zadovoljiv rezultat. Z dodatnim pred-procesiranjem podatkov, ki bi odstranil posebnosti in pogladil vrednosti senzorjev, pa bi natančnost in priklic lahko še povečali.

Pridobljeni model je dovolj dober, da nam bo lahko v prihodnosti služil kot izhodišče za nadaljnje raziskave in analize v tenisu.

7. LITERATURA

[1] Selesnick, Ivan W., and C. Sidney Burrus. "Generalized digital Butterworth filter design." IEEE Transactions on Signal Processing 46.6 (1998): 1688-1694.

[2] Liaw, Andy, and Matthew Wiener. "Classification and regression by randomForest." R news 2.3 (2002): 18-22.

[3] Kohavi, Ron. "A study of cross-validation and bootstrap for accuracy estimation and model selection." Ijcai. Vol. 14. No. 2. 1995.

[4] Balmin, Andrey, et al. "Stratified sampling using adaptive parallel data processing." U.S. Patent Application No. 14/141,635.

Indeks avtorjev / Author index

Konferenca / Conference

Uredili / Edited by

Slovenska konferenca o umetni inteligenci / Slovenian Conference on Artificial Intelligence

Matjaž Gams, Mitja Luštrek, Rok Piltaver