

# PP-RAI'2019

Polskie Porozumienie na rzecz  
Rozwoju Sztucznej Inteligencji

16-18.10.2019

Wrocław, Poland

**Conference  
Proceedings**

Wrocław University of Science and Technology  
Department of Systems and Computer Networks  
wybrzeże Stanisława Wyspiańskiego 27  
50-370 Wrocław, Poland

This proceedings was supported by the statutory funds of the Department of Systems and Computer Networks, Faculty of Electronics, Wrocław University of Science and Technology.

Editorial layout and cover design Paweł Ksieniewicz and Paweł Zybilewski

© Copyright by Department of Systems and Computer Networks, Faculty of Electronics, Wrocław University of Science and Technology  
Wrocław, 2019

Department of Systems and Computer Networks  
Faculty of Electronics  
Wrocław University of Science and Technology  
wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland

**ISBN 978-83-943803-2-8**



ISBN 978-83-943803-2-8



A standard linear barcode representation of the ISBN number, positioned below the ISBN text.

9 788394 380328

## Preface

PP-RAI (*Polskie Porozumienie na Rzecz Rozwoju Sztucznej Inteligencji*) has been established in 2018 in Poznań, Poland (<https://pp-rai.cs.put.poznan.pl/>), where the main organization scientific organizations:

- PSSI · Polskie Stowarzyszenie Sztucznej Inteligencji
- PTSN · Polskie Towarzystwo Sieci Neuronowych
- PL SIGML · Polska Grupa Systemów Uczęcych się
- IEEE SMC · Polski Oddział IEEE SMC (Polish Chapter of the IEEE Systems, Man, and Cybernetics Society)
- IEEE CIS · Polski Oddział IEEE Computational Intelligence Society

signed the declaration about cooperation to boost the AI development in Poland.  
Also the following organizations support PP-RAI:

- Organizations supporting the Congress:
- IEEE Robotics and Automation Society Polish Section
- Network Science Society (Polish Chapter)
- Towarzystwo Przetwarzania Obrazów
- Societas Humboldtiana Polonorum
- Polski Węzeł International Neuroinformatics Coordination Facility

The main objective of the PP-RAI 2019 are:

1. PP-RAI aims to bring together researches and to provide a national forum for the sharing, exchange, presentation and discussion of original research results in different areas of artificial intelligence.
2. Creation of a platform for discussion of important Polish AI/ML research environments and the following disciplines.
3. Discussion on the new forms of organization and cooperation of research teams, reference to the growth of interest in AI and pointing the strategic directions of research and applications, including interdisciplinary areas.
4. Indication of the need for changes in cooperation of the scientific community with business partners and state research institutions.

During the PP-RAI 2019 4 plenary talk have been delivered, 5 discussion panels have been organizer and more than 100 papers have been presented during 4 scientific sessions.

# Organization

## Chairs

Robert Burduk  
Pawel Ksieniewicz

Wroclaw University of Science and Technology  
Wroclaw University of Science and Technology

## Track chairs

### **AAI (Advances in AI — another AI's directions)**

Przemyslaw Kazienko      Wroclaw University of Science and Technology  
Jacek Koronacki      Polish Academy of Sciences

### **Young.AI (session for young researchers)**

Piotr Brodka      Wroclaw University of Science and Technology  
Tomasz Kajdanowicz      Wroclaw University of Science and Technology  
Paweł Ksieniewicz      Wroclaw University of Science and Technology

### **CV (computer vision)**

Leszek Chmielewski      Warsaw University of Life Sciences - SGGW  
Marek Gorgon      AGH University of Science and Technology  
Konrad Wojciechowski      Silesian University of Technology

### **RAS ((R)obotics and (A)utonomous (S)ystems)**

Bogdan Kwolek      AGH University of Science and Technology  
Piotr Skrzypczynski      Poznan University of Technology  
Cezary Zieliński      Warsaw University of Technology

### **NLP+ASR+CAI (natural language processing, automatic speech recognition, and conversational AI)**

Maciej Piasecki      Wroclaw University of Science and Technology  
**KE (knowledge engineering)**

Dariusz Krol      Wrocław University of Science and Technology  
Agnieszka Lawrynowicz      Poznań University of Technology  
Grzegorz J. Nalepa      AGH University of Science and Technology

### **NI (neuroinformatics)**

Włodzisław Duch      Nicolaus Copernicus University

### **ML (machine learning)**

Ireneusz Czarnowski      Gdynia Maritime University  
Szymon Jaroszewicz      Polish Acadamy of Sciences

### **PS+O (problem solving and optimization)**

Jarosław Arabas      Warsaw University of Technology  
Krzysztof Krawiec      Poznan University of Technology  
Jacek Mańdziuk      Warsaw University of Technology

### **UAI (uncertainty in artificial intelligence)**

Leszek Rutkowski      Częstochowa University of Technology

## **Program Commitee**

|                       |  |
|-----------------------|--|
| Ireneusz Czarnowski   | Gdynia Maritime University                   |
| Włodzisław Duch       | Nicholas Copernicus University               |
| Krzysztof Dembczyński | Poznan University of Technology              |
| Janusz Kacprzyk       | Polish Academy of Sciences                   |
| Jacek Koronacki       | Polish Academy of Sciences                   |
| Jacek Mańdziuk        | Warsaw University of Technology              |
| Grzegorz J. Nalepa    | AGH University of Science and Technology     |
| Leszek Rutkowski      | Czestochowa University of Technology         |
| Jerzy Stefanowski     | Poznan University of Technology              |
| Dominik Ślęzak        | University of Warsaw                         |
| Michał Woźniak        | Wroclaw University of Science and Technology |

## **Organization committee**

|                   |
|-------------------|
| Robert Burduk     |
| Paweł Ksieniewicz |
| Barbara Bobowska  |
| Dariusz Jankowski |
| Jakub Klikowski   |
| Mariusz Koziol    |
| Mariusz Topolski  |
| Jakub Zgraja      |
| Paweł Zyblewski   |
| Michał Źak        |

## Table of Contents

|  |    |
|--|----|
| <b>1 AAI (Advances in AI) .....</b>  | 1  |
| Semipresentable Posets and Fuzzy Sets .....  | 2  |
| <i>P. Gladki, W. Borczyk, M. Rostanski</i>   |    |
| On Practical Aspects of MTE .....  | 6  |
| <i>W. Filipowicz</i>   |    |
| Deep Learning Approach to Rare CNV Detection .....   | 10 |
| <i>L. Neumann, R. Nowak, T. Gamin, W. Kusmirek</i>   |    |
| Multilayer Quantum-Classical Neural Network .....  | 11 |
| <i>P. Gawron, A. Krawiec</i>   |    |
| Advanced Model Supporting Football Team Building .....   | 15 |
| <i>B. Cwiklinski, A. Gielczyk, M. Choras</i>   |    |
| Application of Quantum Annealing to Combinatorial Optimization Problems .....  | 19 |
| <i>P. Gawron</i>   |    |
| <b>2 Young.AI .....</b>  | 20 |
| Aggregate and individual approaches for building neural models for mid-term electric energy demand forecasting ..... | 21 |
| <i>P. Pelka</i>  |    |
| Classification of Tree Species from Limited Dataset of Bark Images Using Convolutional Neural Networks .....         | 25 |
| <i>W. Czarnecki, P. Gawron</i>   |    |
| Decision tree integration algorithm using static regions of competence and geometric representation .....            | 29 |
| <i>J. Biedrzycki</i>   |    |
| Text based meme classification .....   | 33 |
| <i>P. Bielak, M. Bieronski, M. Jozwiak</i>   |    |
| Real-Time Polish Traffic Sign Recognition .....  | 37 |
| <i>K. Kania, M. Kosturek</i>   |    |
| Mixed-curvature Embedding of Human Diseases Network .....  | 41 |
| <i>M. Falkiewicz</i>   |    |

|   |    |
|---|----|
| Identification of Players Ranking in E-Sport: CS:GO Study Case .....  | 45 |
| <i>K. Urbaniak</i>  |    |
| Deep learning in EEG: Detection of error-related negativity in Eriksen<br>flanker task .....                | 49 |
| <i>K. Kotowski, K. Stapor</i>   |    |
| Ensemble data preprocessing based methods for imbalanced data<br>stream classification .....                | 53 |
| <i>J. Klikowski</i>   |    |
| Drifted Data Stream Classification using Oversampled Dynamic<br>Ensemble Selection .....                    | 54 |
| <i>P. Zyblewski</i>   |    |
| Generation of context-free grammars for Grammar Inference methods ...                                       | 55 |
| <i>L. Culer, O. Unold, A. Kaczmarek</i>   |    |
| Assessment of Electric City Buses in the Tendering Process: MCDA<br>Case Study.....                         | 59 |
| <i>A. Baczkiewicz</i>   |    |
| Effect of various normalization techniques on the TOPSIS method .....                                       | 63 |
| <i>K. Palczewski</i>  |    |
| RAndom Neural Networks (RANNs): a new general classifier inspired<br>by Random Forest .....                 | 68 |
| <i>P. Piasecki, T. Gorecki</i>  |    |
| Efficient Algorithm for Set-Valued Prediction in Multi-Class Classification                                 | 73 |
| <i>T. Mortier, M. Wydmuch, K. Dembczynski, E. Hullermeier, W.<br/>Waegeman</i>                              |    |
| Multi-classifier system based on center of mass classifier.....   | 77 |
| <i>S. Wojciechowski</i>   |    |
| Concept of Research into Cognitive Load in Human-Computer<br>Interaction Using Biometric Techniques .....   | 78 |
| <i>P. Z. Muke, B. Trawinski</i>   |    |
| Investigating initialization method in the process of facial landmarks<br>detection in thermal images ..... | 84 |
| <i>A. Smolinski, P. Forczmanski</i>   |    |
| Single particle diffusion classification by deep learning .....   | 88 |
| <i>P. Kowalek, H. Loch-Olszewska, J. Szwabinski</i>   |    |
| Data imputation methods in classification task .....  | 92 |
| <i>M. Gaciarz, M. Topolski</i>  |    |

|   |            |
|---|------------|
| Exploration of performance measurement methods for selected unsupervised machine learning algorithms .....        | 99         |
| <i>F. Guzy</i>  |            |
| A Cluster-Based Approach for AIS Data Analysis and Vessel Trajectory Reconstruction .....                         | 103        |
| <i>M. Mieczynska and I. Czarnowski</i>  |            |
| Aspect-based Sentiment Analysis Summarization using Rhetorical Analysis and Complex Networks .....                | 107        |
| <i>L. Augustyniak, T. Kajdanowicz, P. Kazienko</i>  |            |
| Towards More Efficient and Secure Federated Learning Methods .....  | 112        |
| <i>M. Piwowarczyk, B. Trawinski</i>   |            |
| <b>3 CV (computer vision) .....</b>   | <b>116</b> |
| Automatic identification of vitreomacular pathologies based on optical coherence tomography scans .....           | 117        |
| <i>A. Stankiewicz, T. Marciniak, A. Dabrowski, M. Stopa, E. Marciniak</i>   |            |
| Band selection with Higher Order Multivariate Cumulants for small target detection in hyperspectral images .....  | 121        |
| <i>P. Glomb, K. Domino, M. Romaszewski, M. Cholewa</i>  |            |
| Unsupervised deep learning approach to hyperspectral anomaly detection  | 128        |
| <i>B. Grabowski, P. Glomb, K. Domino, M. Romaszewski, M. Ostaszewski</i>  |            |
| Break the curse of small datasets in computer vision tasks with transfer learning methods .....                   | 133        |
| <i>J. Jaworek-Korjakowska, A. Brodzicki, D. Kucharski, M. Piekarski, M. Gorgon</i>                                |            |
| Semisupervised Segmentation using Autoencoder - Comparison of Convergence for Manual and Random Assignments ..... | 137        |
| <i>P. Mazurek, D. Oszutowska-Mazurek, O. Knap</i>   |            |
| Application of GGD Based Preprocessing for Region Based Binarization of Degraded Document Images .....            | 141        |
| <i>R. Krupinski, P. Lech, H. Michalak, K. Okarma</i>  |            |
| Graph CNN with Filter Transformations for Structure Detection and Identification .....                            | 145        |
| <i>A. Tomczyk</i>   |            |
| Handcrafted features for CNN - is it worth it? .....  | 146        |
| <i>M. Stefanczyk, D. Seredyński, M. Wegierek</i>  |            |

|  |            |
|--|------------|
| Towards color visual cryptography with completely random shares .....  | 150        |
| <i>L. J. Chmielewski, G. Gawdzik, A. Orlowski</i>  |            |
| Neural Network-based Compressed Image Improvement .....  | 156        |
| <i>P. Najgebauer, R. Scherer</i>   |            |
| <b>4 RAS (Robotics and Autonomous Systems) .....</b>   | <b>160</b> |
| Low Effort Cross-Modal Learning for 3-D LiDAR Data Segmentation<br>in SLAM .....   | 161        |
| <i>K. Cwian, T. Nowak, M. Nowicki, P. Skrzypczynski</i>  |            |
| Machine learning approach to constrained path planning for intelligent<br>articulated buses .....                            | 165        |
| <i>P. Kicki, T. Gawron, M. M. Michalek</i>   |            |
| You Only Look Once Around: Learnable Object Detection for<br>Bioinspired Visual Localization .....                           | 169        |
| <i>M. Rostkowska</i>   |            |
| Usability of Reinforcement Learning Methods in the Task of<br>Manipulation of Deformable Linear Objects .....                | 173        |
| <i>M. Bednarek, K. Walas</i>   |            |
| Improving Person Re-identification by Segmentation-Based Detection<br>Bounding Box Filtering .....                           | 177        |
| <i>D. Pieczynski, M. Kraft, M. Fularz</i>  |            |
| From the Edge to the Datacenter: Evaluating the Throughput and<br>Power Efficiency of Deep Learning Hardware Platforms ..... | 178        |
| <i>M. Kraft, D. Pieczynski, M. Fularz</i>  |            |
| Kinematic Structures Detection and Estimation with Neural Network<br>and Black-box Optimization .....                        | 179        |
| <i>D. Belter</i>   |            |
| <b>5 NLP+ASR+CAI (natural language processing,<br/>automatic speech recognition, and conversational<br/>AI) .....</b>        | <b>183</b> |
| Parliamentary election's predictions using social media content .....  | 184        |
| <i>A. Sobkowicz, M. Kozlowski</i>  |            |
| Neural style transfer for non-native speech recognition .....  | 189        |
| <i>K. Radzikowski, R. Nowak, O. Yoshie</i>   |            |
| Sub-word units in Polish text generation .....   | 198        |
| <i>E. Zawadzka-Gosk, K. Wolk</i>   |            |

|   |     |
|---|-----|
| Comparison of topic modelling algorithms in text clustering problem ....  | 202 |
| <i>T. Walkowiak, M. Gniewkowski</i>   |     |
| Does the syntax matter for composing semantic representation of medical products therapeutic indications? .....       | 206 |
| <i>W. Jaworski</i>  |     |
| Integrating Polish Language Tools and Resources in Spacy .....  | 210 |
| <i>R. Tuora, L. Kobylinski</i>  |     |
| Bias-variance tradeoff problem in dialogue agents: a nearly infinite size state space model postulate .....           | 215 |
| <i>D. Oklesinski</i>  |     |
| Fine-Grained Named Entity Recognition for Polish using Deep Learning .  | 219 |
| <i>M. Marcinczuk</i>  |     |
| Information retrieval from biomedical textstrongly depends on parameters: application tothe bioCADDIE benchmark ..... | 223 |
| <i>A. Cieslewicz, J. Dutkiewicz, C. Jedrzejek</i>   |     |
| <b>6 KE (knowledge engineering)</b> .....   | 231 |
| Disambiguation of experts in the Chinese knowledge base .....   | 232 |
| <i>R. Nowak, W. Franus</i>  |     |
| Mining Cardinality Restrictions in OWL .....  | 234 |
| <i>J. Potoniec</i>  |     |
| Towards smart enterprises: supporting the business processes using artificial intelligence .....                      | 238 |
| <i>M. Hernes</i>  |     |
| Automatic Translation of Ontology Competency Questions into SPARQL-OWL Queries .....                                  | 242 |
| <i>D. Wisniewski, A. Lawrynowicz</i>  |     |
| Brief Overview of Research Directions in Artificial Intelligence Methods for Business Process Management .....        | 243 |
| <i>K. Kluza, P. Wisniewski, W. T. Adrian, A. Ligeza, M. Adrian, B. Stachura-Terlecka, K. Jobczyk</i>                  |     |
| Semantic Information Extraction and Knowledge Graph Analysis .....  | 247 |
| <i>W. T. Adrian, M. Manna, G. Amendola, R. Penaloza</i>   |     |
| Context-Based Inference in Technical Diagnostics .....  | 251 |
| <i>A. Timofiejczuk</i>  |     |
| Towards data-event-driven approach in ADVISOR project .....   | 253 |
| <i>D. Krol</i>  |     |

|  |     |
|--|-----|
| <b>7 NI (neuroinformatics) . . . . .</b>   | 254 |
| Kernel Current Source Density (kCSD) as an example of applied<br>Machine Learning . . . . .                      | 255 |
| <i>J. M. Dzik, M. Bejtka, C. Chintaluri, D. Wojcik</i>   |     |
| Computational investigation of biochemical foundations of learning and<br>memory . . . . .                       | 259 |
| <i>Z. T. Slawinski, J. Jedrzejewska-Szmek, D. Wojcik</i>   |     |
| <b>8 ML (machine learning) . . . . .</b>   | 264 |
| Boolean Biclustering Review and Perspectives . . . . .   | 265 |
| <i>M. Michalak</i>   |     |
| New Methods of Generating Random Parameters in Feedforward<br>Neural Networks with Random Hidden Nodes . . . . . | 269 |
| <i>G. Dudek</i>  |     |
| Solving Inconsistencies of the Perfect Clustering Concept . . . . .  | 273 |
| <i>M. A. Kłopotek, R. A. Kłopotek</i>  |     |
| On the Shape of $k$ -means Clusters and Their Motion Consistency . . . . .                                       | 277 |
| <i>M. A. Kłopotek, S. T. Wierzchon, R. A. Kłopotek</i>   |     |
| Analytical Forms of Normalized and Combinatorial Laplacians of Grid<br>Graphs . . . . .                          | 281 |
| <i>M. A. Kłopotek, S. Wierzchon, R. A. Kłopotek</i>  |     |
| Return of Investment in Machine Learning: Crossing the Chasm<br>between Academia and Business . . . . .          | 285 |
| <i>J. Mizgajski, A. Szymczak, P. Zelasko, M. Morzy, L. Augustyniak,<br/>        P. Szymanski</i>                 |     |
| SOUP-Bagging: a new approach for multi-class imbalanced data<br>classification . . . . .                         | 292 |
| <i>M. Lango, J. Stefanowski</i>  |     |
| Streaming approach to Big Data analysis . . . . .  | 296 |
| <i>P. Duda, L. Rutkowski</i>   |     |
| Various Aspects of Data Distribution Monitoring Using the Restricted<br>Boltzmann Machine . . . . .              | 299 |
| <i>M. Jaworski, L. Rutkowski</i>   |     |
| Improving Evolutionary Instance Selection with Clustering and Ensembles  | 302 |
| <i>M. Kordos, M. Blachnik</i>  |     |

|   |            |
|---|------------|
| Evaluation of Musical Data Representation for Music Information Retrieval.....                                | 306        |
| <i>M. Klec, K. Marasek, K. Szklanny</i>   |            |
| SAFAIR: Secure and Fair AI Systems for Citizens .....   | 310        |
| <i>M. Choras, M. Pawlicki, R. Kozik</i>   |            |
| MDFS – a statistical filter for multivariate interactions .....   | 311        |
| <i>K. Mnich, W. Rudnicki, R. Piliszek</i>   |            |
| Prediction of Drug-induced Liver Injury using different integration techniques .....                          | 315        |
| <i>W. Lesinski, A. Kitlas-Golinska, K. Mnich, W. Rudnicki</i>   |            |
| Weighted Context-free Grammar Induction-a preliminary report .....  | 319        |
| <i>O. Unold, M. Gabor</i>   |            |
| Robust Machine Learning protocol with estimation of biases .....  | 323        |
| <i>W. Rudnicki, K. Mnich, R. Piliszek, A. Polewko-Klim, W. Lesinski, B. Sapinski</i>                          |            |
| Recent Advances in Cross-Domain Sentiment Analysis of Polish Texts ...  | 327        |
| <i>A. Janz, J. Kocon</i>  |            |
| Bayes optimal prediction for NDCG@k in extreme multi-label classification                                     | 332        |
| <i>K. Jasinska, K. Dembczynski</i>  |            |
| Preliminary tests of a real-valued Anticipatory Classifier System .....                                       | 336        |
| <i>N. Kozlowski, O. Unold</i>   |            |
| <b>9 PS+O (problem solving and optimization) .....</b>  | <b>341</b> |
| Influence of Traffic Type on Traffic Prediction Quality in Dynamic Optical Networks with Service Chains ..... | 342        |
| <i>D. Szostak, K. Walkowiak</i>   |            |
| Social Impact Assessment and Multicriteria Optimization of AI Tools for Online Knowledge Provision .....      | 346        |
| <i>A. M. J. Skulimowski</i>   |            |
| Dynamic signature verification using AI methods .....   | 352        |
| <i>M. Zalasinski, K. Cpalka</i>   |            |
| The use of new space properties of binary vectors in the set partitioning problem .....                       | 356        |
| <i>Z. Pliszka, O. Unold</i>   |            |
| Ain't Nobody Got Time for Coding: Structure-Aware Program Synthesis from Natural Language .....               | 360        |
| <i>J. Bednarek, K. Piaskowski, K. Krawiec</i>   |            |

|  |            |
|--|------------|
| Population-based Algorithms for Selecting Parameters and Structures<br>of Various Crisp and Fuzzy Systems .....                            | 364        |
| <i>K. Lapa, K. Cpalka</i>  |            |
| On the development of the ASDM method .....  | 368        |
| <i>P. Wawrzynski, P. Zawistowski, L. Lepak</i>   |            |
| Differential Evolution Strategy: a differential evolution version of the<br>Covariance Matrix Adaptation Evolution Strategy .....          | 372        |
| <i>J. Arabas, D. Jagodzinski</i>   |            |
| Optimization of ultra-thin magnetron sputtered aluminum films with<br>the use of AI models .....   | 376        |
| <i>E. Warchulski, R. Mroczynski, J. Arabas</i>   |            |
| Generalized Self-Adapting Particle Swarm Optimization algorithm with<br>model-based optimization enhancements .....                        | 380        |
| <i>M. Zaborski, M. Okulewicz, J. Mandziuk</i>  |            |
| Exploring Constraint Programming. Approaching a Practical<br>Optimization Problem .....  | 384        |
| <i>W. T. Adrian, M. Slazynski, A. Ligeza, M. Manna, N. Leone, M.<br/>Adrian, K. Jobczyk, K. Kluza, B. Stachura-Terlecka, P. Wisniewski</i> |            |
| Employing supervised learning algorithms in the task of dynamic<br>spectrally-spatially flexible optical networks optimization .....       | 388        |
| <i>P. Ksieniewicz, M. Klinkowski, K. Walkowiak</i>   |            |
| <b>Author Index .....</b>  | <b>392</b> |

## AAI (Advances in AI)

# Semipresentable posets and fuzzy sets.\*

Wojciech Borczyk<sup>1,3</sup>, Paweł Gładki<sup>1,2,3</sup>, and Maciej Rostański<sup>1,4</sup>

<sup>1</sup> Incuvo SA, ul. Ligocka 103, 40-568 Katowice, Poland

<sup>2</sup> AGH University of Science and Technology, Department of Computer Science, al.  
Mickiewicza 30, 30-059 Kraków, Poland

<sup>3</sup> University of Silesia, Institute of Computer Science, ul. Bedzińska 39, 41-200  
Sosnowiec, Poland,  
[wojciech.borczyk@us.edu.pl](mailto:wojciech.borczyk@us.edu.pl)

Institute of Mathematics, ul. Bankowa 14, 40-007 Katowice, Poland  
[pawel.gladki@us.edu.pl](mailto:pawel.gladki@us.edu.pl)

<sup>4</sup> WSB University, Faculty of Computer Science, ul. Cieplaka 1c, 41-300 Dąbrowa  
Górnica, Poland  
[mrostanski@wsb.edu.pl](mailto:mrostanski@wsb.edu.pl)

**Abstract.** In this paper we investigate how fuzzy sets give rise to a certain notion of fuzziness in the realm of semipresentable posets which are, in turn, objects defined on certain partially ordered sets. Semipresentable posets are building blocks for presentable groups and which are somewhat equivalent to hypergroups, that is algebras with multivalued addition resembling groups, frequently used in fuzzy set theory.

## 1 Introduction.

Hypergroups were introduced in 1934 by Marty [3] and have been intensively researched since then. They have proven applicable in many areas of science, including artificial intelligence and probability theory – for survey of applications see [1]. Rough set theory was introduced by Pawlak in [4] and then evaluated towards AI and data analysis [5].

Fuzzy logic provides possible solution for dealing with approximate reasoning, when imprecision and uncertainty is a factor, as shown in various applications in [6]. Those also include artificial intelligence based data analysis.

We have been analyzing various methods of data analysis in the context of research on computer system that analyzes user related data in order to improve some of its key performance indicators by automatically adjusting parameters. The system is a computer game in which users attack other users' castles and multitude of data is provided describing some of user behavior, including usage frequency and attack frequency. One of the directions of research is to research applying fuzzy logic and using certain hyperstructures in order to implement

---

\* This research was in part supported by the Polish Programme GAMEINN, agreement no. PIOR.01.02.00-00-0130/16 from March 9th, 2019, competition 3/1.2/2016/POIR organized by the National Centre for Research and Development and co-funded by the European Fund for Sustainable Development

approximate reasoning in context of provided data and data sources. This might lead to interesting results in terms of classification by using fuzzy classification instead of strict classification. Further applying of fuzzy logic on those datasets could lead to better channeling of user flow between groups and in the result improve key performance indicators.

We have been experimenting with replacing hyperstructures as underlying engine for fuzzy computations with semipresentable posets. These are relatively recently developed tools, invented primarily for simplifying category theoretical approach to hyperalgebras. In this paper we show that semipresentable posets are well suited for generalizing the notion of a fuzzy set.

## 2 Semipresentable posets.

Let  $(A, \leq)$  be a poset. For a subset  $S \subseteq A$  we shall write  $\bigvee S$  for the supremum of  $S$  (if it exists), and  $\bigwedge S$  for the infimum of  $S$  (if it exists). If  $S = \{s_1, \dots, s_n\}$ , we shall write  $s_1 \vee \dots \vee s_n$  to denote  $\bigvee S$  and  $s_1 \wedge \dots \wedge s_n$  to denote  $\bigwedge S$ .

**Definition 1.** *The triple  $(A, \mathcal{S}_A, \leq)$ , where  $\mathcal{S}_A \subset A$ , is called a semipresentable poset if*

- i.  $\mathcal{S}_A$  is coinitial subset of  $A$  of cardinality equal to the coinitiality of  $A$ ,
  - ii. every nonempty subset  $S \subseteq \mathcal{S}_A$  has a supremum,
  - iii.  $x = \bigvee \{s \in \mathcal{S}_A \mid s \leq x\}$ , for every  $x \in A$ ,
  - iv.  $\wp_A \in \mathcal{S}_A$ .
  - v. it is pointed, i.e. it contains a distinguished element  $\wp_A$  called basepoint,
- The unique coinitial subset  $\mathcal{S}_A$  of a semipresentable set  $A$  will be called the set of supercompacts. We shall denote  $\mathcal{S}_x = \{s \in \mathcal{S}_A \mid s \leq x\}$ .

*Example 1.* Let  $X \neq \emptyset$  be a nonempty set, let  $A = 2^X \setminus \{\emptyset\}$  be the set of its nonempty subsets.  $(A, \subseteq)$  is clearly a poset, and the triple  $(A, \mathcal{S}_A, \subseteq)$  is a semipresentable poset if one chooses  $\mathcal{S}_A$  to be the set of all singletons of  $X$ , and  $\wp_A = \{x_0\} \in A$ ,  $x_0 \in X$ , a distinguished singleton set. Singletons form a coinitial set  $\mathcal{S}_A$  of cardinality equal to that of  $A$ , and every subset  $Y$  of  $X$  is a union of its elements, and hence a supremum of supercompacts below  $Y$ .

*Example 2.* Let  $X \neq \emptyset$  be a nonempty set. A fuzzy subset of  $X$  is a function  $\xi : X \rightarrow [0, 1]$  and the empty fuzzy subset is the function  $0 : X \rightarrow [0, 1]$ . Denote by  $\mathcal{F}(X)$  the family of all fuzzy subsets of  $X$ . Let  $A = \mathcal{F}(X) \setminus \{0\}$ .  $(A, \leq)$  is a poset with  $\leq$  defined by  $\xi \leq \eta$  if and only if  $\forall x \in X (\xi(x) \leq \eta(x))$ . For  $x \in X$  and  $a \in [0, 1]$  denote by  $\delta_x^a \in \mathcal{F}(X)$  the fuzzy set  $\delta_x^a(y) = \begin{cases} a, & \text{if } x = y, \\ 0, & \text{if } x \neq y, \end{cases} y \in X$ .

Let  $\mathcal{S}_A = \{\delta_x^a \mid x \in X, a \in (0, 1] \cap \mathbb{Q}\}$ . The triple  $(A, \mathcal{S}_A, \leq)$  is a semipresentable poset if one arbitrarily chooses  $\wp_A = \delta_{x_0}^1$ , for some  $x_0 \in X$ : by the completeness of  $\mathbb{R}$ , every subset of  $\mathcal{S}_A$  has a supremum in  $A$ , and, by the density of  $\mathbb{Q}$  in  $\mathbb{R}$ , every nonzero function  $\xi : X \rightarrow [0, 1]$  is a supremum of some  $\delta_x^a$ 's from  $\mathcal{S}_A$ . The only elements less or equal  $\delta_x^a$ ,  $x \in X$  and  $a \in (0, 1]$ , are  $\delta_x^b$  with  $0 < b \leq a$ , so that any coinitial subset of  $A$  must contain all  $\delta_x^b$ , where  $x \in X$  and  $b \in (0, 1]$  are from some dense subset of  $\mathbb{R}$ : as  $\mathbb{Q}$  is a dense subset of  $\mathbb{R}$  of least cardinality, the cardinality of  $\mathcal{S}_A$  is equal to the coinitiality of  $A$ .

**Definition 2.** For  $(A, \mathcal{S}_A, \leq_A)$ ,  $(B, \mathcal{S}_B, \leq_B)$  a map  $A \xrightarrow{f} B$  is a morphism if  
 i. it is pointed, i.e.  $f(\wp_A) = \wp_B$ , ii. it is strict, i.e.  $f(\mathcal{S}_A) \subseteq \mathcal{S}_B$ ,  
 iii. it is continuous, i.e.  $f(\bigvee Y) = \bigvee f(Y)$ , for every  $Y \subset A$ .

The category of semipresentable posets will be denoted by **sPos**.

*Example 3.* Let  $X, Z \neq \emptyset$ ,  $A = 2^X \setminus \{\emptyset\}$ ,  $B = 2^Z \setminus \{\emptyset\}$ ,  $\wp_A = \{x_0\}$ ,  $\wp_B = \{z_0\}$ , for  $x_0 \in X$ ,  $z_0 \in Z$ . Let  $\phi : X \rightarrow Z$  be such that  $\phi(x_0) = z_0$ . Then  $A \xrightarrow{f} B$  defined by  $f(Y) = \bigcup\{\phi(y) \mid y \in Y\}$ ,  $Y \in A$ , is a morphism.

*Example 4.* Let  $X, Z \neq \emptyset$ ,  $A = \mathcal{F}(X) \setminus \{0\}$ ,  $B = \mathcal{F}(Z) \setminus \{0\}$ ,  $\wp_A = \delta_{x_0}^1$ ,  $\wp_B = \delta_{z_0}^1$ , for  $x_0 \in X$ ,  $z_0 \in Z$ . Let  $\phi : X \rightarrow Z$  be such that  $\phi(x_0) = z_0$ . Define  $\phi^* : \mathcal{S}_A \rightarrow \mathcal{S}_B$  by  $\phi^*(\delta_x^a) = \delta_{\phi(x)}^a$ . Then  $A \xrightarrow{f} B$ ,  $f(\bigvee\{\delta_x^a \mid \delta_x^a \in S\}) = \bigvee\{\phi^*(\delta_x^a) \mid \delta_x^a \in S\}$ ,  $S \subset \mathcal{S}_A$ , is a morphism.

### 3 Fuzzy sets and pointed fuzzy sets

If  $\eta : Y \rightarrow [0, 1]$  is another fuzzy set, a morphism is defined, e.g. in Goguen [2], to be a function  $f : X \rightarrow Y$  such that  $\forall x \in X (\xi(x) \leq \eta \circ f(x))$ . The category of fuzzy sets shall be denoted by **Fuzz**. We define a partial ordering  $\leq$  on the set  $\mathcal{F}(X)$  by  $\xi \leq \eta \Leftrightarrow \forall x \in X (\xi(x) \leq \eta(x))$ . Following Example 1, we would like to investigate the connection between fuzzy sets and presentable posets. Firstly, fuzzy sets indeed define presentable posets in a rather natural way, as described in Proposition 1. This relationship between fuzzy sets and presentable posets is almost functorial: to be more specific, if we focus on the category **Fuzz\*** of pointed fuzzy sets, that is pairs  $(\xi, x_0)$  consisting of a fuzzy set  $\xi \in \mathcal{F}(X)$  and an element  $x_0 \in X$ , with morphisms  $(\xi, x_0) \xrightarrow{f} (\eta, y_0)$  being the usual morphisms in **Fuzz**, but with the extra requirement that  $f(x_0) = y_0$ , then we have well behaved functors between **sPos** and **Fuzz\***, as shown in Proposition 2.

**Proposition 1.** Let  $X \neq \emptyset$ ,  $\xi \in \mathcal{F}(X)$ . Denote  $2^\xi = \left\{ \bigvee\{\delta_x^{\xi(x)} \mid x \in S\} \mid S \subset X \right\}$ .  $(2^\xi, \leq)$  is a presentable poset with  $\wp_{2^\xi} = \delta_x^{\xi(x)}$ ,  $x \in X$ ,  $\mathcal{S}_{2^\xi} = \{\delta_x^{\xi(x)} \mid x \in X\}$ .

*Proof.* This is, more or less, obvious:  $(2^\xi, \leq)$  is clearly a poset which becomes pointed once we arbitrarily choose  $\delta_x^{\xi(x)}$ ,  $x \in X$ , and  $\mathcal{S}_{2^\xi}$  is readily the unique coinitial subset of  $2^\xi$  of cardinality equal to the coinitiality of  $2^\xi$ : that  $\mathcal{S}_{2^\xi}$  is coinitial is apparent, and the only elements less or equal  $\delta_x^{\xi(x)}$  are  $\delta_x^{\xi(x)}$  themselves, thus forcing  $\mathcal{S}_{2^\xi}$  to be contained in any coinitial subset of  $2^\xi$ . That every nonempty subset of  $\mathcal{S}_{2^\xi}$  has a supremum in  $2^\xi$  follows from the definition, and that  $\eta = \bigvee \mathcal{S}_\eta$  is apparent. Finally, that every  $\delta_x^{\xi(x)}$ ,  $x \in X$ , is compact can be easily seen: if  $\delta_x^{\xi(x)} \leq \bigvee Y$ , for some  $Y \subset 2^\xi$ , then  $\xi(x) \leq \bigvee\{\eta(x) \mid \eta \in Y\}$ . But for  $\eta \in Y$ ,  $\eta(x)$  is either 0 or  $\xi(x)$ , so just take one that is equal to  $\xi(x)$ .  $\square$

**Proposition 2.**  $S : \mathbf{sPos} \rightarrow \mathbf{Fuzz*}$  and  $T : \mathbf{Fuzz*} \rightarrow \mathbf{sPos}$  defined by  $S(A) = (\chi_{\mathcal{S}_A}, \wp_A)$ ,  $S(f)(s) = f(s)$ , for  $A \xrightarrow{f} B$ , are covariant functors, where  $\chi_{\mathcal{S}_A} \in$

$$\mathcal{F}(\mathcal{S}_A), \chi_{\mathcal{S}_A}(s) = 1, \text{ and } T(\xi, x_0) = 2^\xi, \wp_{2^\xi} = \delta_{x_0}^{\xi(x_0)} \text{ for } \xi \in \mathcal{F}(X), \\ T(f) \left( \bigvee \{\delta_x^{\xi(x)} \mid x \in S\} \right) = \bigvee \{\delta_y^{\eta(y)} \mid y \in f(S)\}, \xi \xrightarrow{f} \eta, \xi \in \mathcal{F}(X), \eta \in \mathcal{F}(Y).$$

*Proof.* That  $S(A)$  defines a function  $\mathcal{S}_A \rightarrow [0, 1]$  is apparent. Likewise, if  $\wp_A \in \mathcal{S}_A$  is the point in  $A$ , then  $(S(A), \wp_A)$  becomes a pointed fuzzy set. If  $A \xrightarrow{f} B$  is a morphism of presentable posets, then, in particular,  $f$  carries  $\mathcal{S}_A$  to  $\mathcal{S}_B$ . Moreover,  $\chi_{\mathcal{S}_A}(s) = 1 = \chi_{\mathcal{S}_B}(f(s))$ , so that  $f \upharpoonright_{\mathcal{S}_A}$ , indeed, is a well defined morphism of fuzzy sets. As  $f(\wp_A) = \wp_B$ , it is a morphism of pointed fuzzy sets.

Likewise, that  $T(\xi)$  defines a presentable poset and  $\delta_{x_0}^{\xi(x_0)}$  can be chosen as the point is Proposition 1. If  $(\xi, x_0) \xrightarrow{f} (\eta, y_0)$  is a morphism of pointed fuzzy sets  $\xi \in \mathcal{F}(X), \eta \in \mathcal{F}(Y), x_0 \in X, y_0 \in Y$ , then  $T(f) : 2^\xi \rightarrow 2^\eta$  is a well defined function that clearly carries  $\mathcal{S}_{2^\xi}$  to  $\mathcal{S}_{2^\eta}$ , i.e. is strict. It is also continuous:

$$\begin{aligned} T(f) \left( \bigvee \left\{ \bigvee \{\delta_x^{\xi(x)} \mid x \in S\} \mid S \in \mathcal{T} \right\} \right) &= T(f) \left( \bigvee \left\{ \delta_x^{\xi(x)} \mid x \in \bigcup_{S \in \mathcal{T}} S \right\} \right) \\ &= \bigvee \left\{ \delta_y^{\eta(y)} \mid y \in \bigcup_{S \in \mathcal{T}} f(S) \right\} = \bigvee \left\{ \bigvee \{\delta_y^{\eta(y)} \mid y \in f(S)\} \mid S \in \mathcal{T} \right\} \\ &= \bigvee \left\{ T(f) \left( \bigvee \{\delta_x^{\xi(x)} \mid x \in S\} \right) \mid S \in \mathcal{T} \right\}, \end{aligned}$$

for  $\mathcal{T} \subset 2^{\mathcal{S}_A}$ . Since  $f(x_0) = y_0$ , also  $T(f)(\wp_{2^\xi}) = T(f)(\delta_{x_0}^{\xi(x_0)}) = \delta_{y_0}^{\eta(y_0)} = \wp_{2^\eta}$ .  $\square$

**Proposition 3.** Let  $X \neq \emptyset$ . The set  $A = \mathcal{F}(X) \setminus \{0\}$  ordered by  $\leq$  is a presentable poset with  $\wp_A = \delta_x^a, x \in X, a \in (0, 1], \mathcal{S}_A = \{\delta_x^a \mid x \in X, a \in (0, 1]\}$ .

*Proof.*  $(A, \leq)$  is a poset which becomes pointed once we choose  $\delta_x^a, x \in X, a \in (0, 1]$  and  $\mathcal{S}_A$  is the unique minimal coinitial subset of  $A$ : the only elements less or equal  $\delta_x^a$ , for  $x \in X$  and  $a \in (0, 1]$ , are  $\delta_x^b$  with  $0 < b \leq a$ , so that  $\mathcal{S}_A$  is contained in any coinitial subset of  $A$ . That every nonempty subset of  $\mathcal{S}_A$  has a supremum in  $A$  is clear, and that  $\eta = \bigvee \mathcal{S}_\eta$  is apparent, for  $\eta \in A$ . Every  $\delta_x^a, x \in X, a \in (0, 1]$  is compact: if  $\delta_x^a \leq \bigvee Y$ , for  $Y \subset A$ , then  $a \leq \bigvee \{\eta(x) \mid \eta \in Y\}$ . But for  $\eta \in Y$ ,  $\eta(x)$  is 0 or  $\xi(x)$ , so take the one equal to  $\xi(x)$ .  $\square$

## References

- [1] P. Corsini and V. Leoreanu. *Applications of hyperstructure theory*, volume 5 of *Advances in Mathematics*. Kluwer Academic Publishers, Dordrecht, 2003.
- [2] J. Goguen. Concept representation in natural and artificial languages. *Int. J. Man-Mach. Stud.*, 6:513–561, 1974.
- [3] F. Marty. Sur une généralisation de la notion de groupe. In *Congrès des Mathématiciens Scandinaves Tenu à Stockholm*, pages 45–49. Stockholm, 1934.
- [4] Z. Pawlak. Rough sets. *Int. J. Comput. Inform. Sci.*, 11:341–356, 1982.
- [5] Z. Pawlak and A. Skowron. Rudiments of rough sets. *Inform. Sci.*, 177:3–27, 2007.
- [6] R. Yager and L. Zadeh. *An Introduction to Fuzzy Logic Applications in Intelligent Systems*. Springer Science Business Media, 2012.

# On Practical Aspects of MTE

Włodzimierz Filipowicz<sup>1[0000-0002-9713-8780]</sup>

<sup>1</sup> Gdynia Maritime University, Gdynia, Poland  
w.filipowicz@wpit. umg.edu.pl

**Abstract.** Uncertainty, which is related to random and systematic deflections, is present in all measurements. In metrology and nautical science many of their aspects are fuzzy. Fuzzy sets proved to be suitable platform enabling modelling imprecise and uncertain data such as nautical observations fuzziness can be used to include the knowledge into a mathematical model. Mathematical Theory of Evidence is of primary importance when dealing with uncertainty. Mathematical Theory of Evidence (MTE) exploits belief and plausibility measures. It operates on belief assignments that are evidence models. MTE delivers combination scheme enabling association of various quality data obtained from different sources. Combination scheme enables various hypotheses support calculations, associated data feature higher informative context. MTE enables evaluation of uncertainty propagation.

**Keywords:** Belief Functions, Uncertainty, Empirical Data.

## 1 Introduction

In many practical cases truth of propositions can be proven based on related evidence. These take place in criminology where evidence is meant as eyewitnesses statements. Hypothesis refers to pointing the culprit. Who is the culprit – this question is supposed to be answered based on interrogations results. Thinking about upgrading mathematical model one can assume that set of offenders is confined to limited numbers of elements. Power set of such collection is treated as a frame of discernment. Examination of a single witness can be perceived and modelled as matrix containing pairs of power set constituent and a measure assigned to it by particular testimonial.

In navigation and metrology one has distorted observation and seeks a degree that it represents the true value. Latest is crucial in order to confirm ship being situated at given location [2]. To reason on the truth of representing the real observation one has great amount of knowledge and experience. Those are to be included into upgraded mathematical model. Piece of evidence embraces pair: uncertain observation (for example taken bearing or distance to a landmark) and probability that it is a true value.

In practice more interesting is the case where many eyewitnesses' testimonials are available or elsewhere many different observations are given. Various quality factors are assumed regarding each of them. Above mentioned enable definition of so called belief assignment that can be perceived as described by Formula (1). Index  $i$  indicate multiple pieces of evidence referring z-th proposition.

$$\begin{aligned} m(e_i) &= \left[ (z_x, m(z_x)_i), (\Theta, m(\Theta)_i) \right] \\ m(\Theta)_i &= g(\Theta, s_i) \end{aligned} \quad (1)$$

where:

- $z_x$  proposition pointing at culprit(s)  $x$ , stating that location  $x$  represents: true observation, true prediction, fixed position (two dimensional approach), etc.;
- $m(z_x)_i$  supporting mass of the proposition imbedded within  $i$ -th piece of evidence;
- $\Theta, m(\Theta)$  frame of discernment and mass attributed to it, latest represents uncertainty;
- $s_i$  subjective evaluation of  $i$ -th piece of evidence.



**Fig. 1.** Establishing relationship between evidence and related hypotheses is important practical issue

Thus more important proposition that should be considered is “what is the truth attributed to particular hypothesis given all measurements (testimonials) at hand”.

Belief assignments are subject to combination in order to increase their informative context [3]. Idea of conjunctive combination of two structures is specified by Formula (2).

$$m_C(Z_{ci}) = \sum_{Z_{1j} \cap Z_{2k} = Z_{ci}} m_{1j} \cdot m_{2k} \quad (2)$$

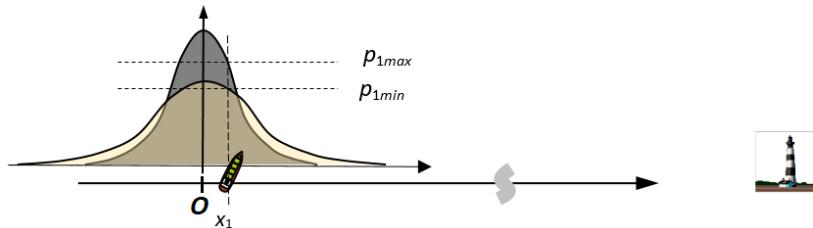
where:

- $Z_{ij}$  set that is  $j$ -th element of a power set of considered frame of discernment related to  $i$ -th observation
- $m_{ij}$  mass of confidence attributed to  $Z_{ij}$ . Amount of support for  $j$ -th hypothesis included in  $i$ -th piece of evidence

Results of combination may require conversion since some mass could be attributed to empty set. Such assignment means conflicting situation that should be avoided. Once occurred, mainly during conjunctive association, must be eliminated. It is normalization that leads to belief structures, assignments without unwanted inconsistency cases. Association is valid for functions defined by Formula (1).

## 2 Uncertainty Model for Measurements

Measurements also called as observations made with any aid is randomly deflected and can be treated as instances of a random variables, governed by some kind of density distribution. The Gaussian bell function is often used, although discrepancies in the parameters of such distributions frequently occur. Seafarers know much about the unavoidable random nature of measurements.



**Fig. 1.** Aleatory uncertainty related to distance taken to a landmark

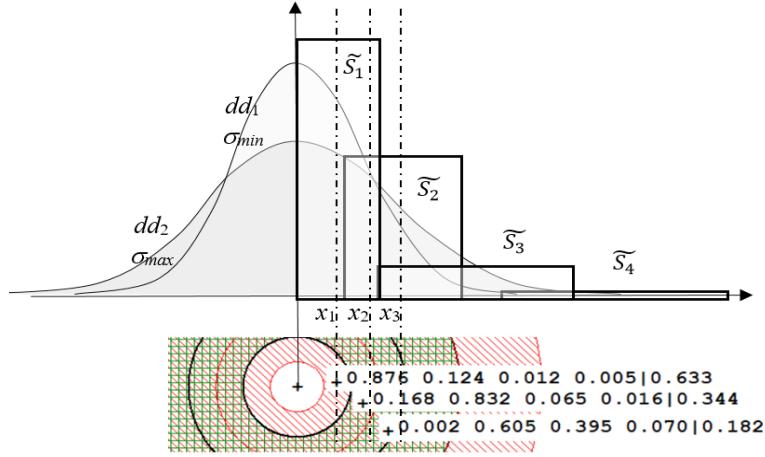
It was stated that probability mass should be available once MTE is intended as platform for upgrading mathematical model. Necessary conversion from density distribution can be made based upon fuzzy sets. Hereafter it is assumed that randomness is governed by histograms or Gaussian distributions. There are wide variety of dispersion estimations available. The latest is valid even for same aid and observer. Two dissemination diagrams referring to taken distance are shown in Figure 2, narrow one seems optimistic and the second one appears to be pessimistic. For presented case distance measured is assumed to be located at point  $O$ .

In view of above mentioned data the truth of various correlated statements might be evaluated. Propositions should be assessed taking into account interval  $[C_1, C_2]$ . Where  $C_i$  is a function of  $p_{min}$  and  $p_{max}$ , since given diagrams represent a probability density functions.

### 2.1 From Density Distribution to Probability Values

The Gaussian bell function is often used in order to represent density of locations to be considered as true measurements. Discrepancies in estimated parameters of such distributions frequently occur. Two of the density dispersions, one of which is named  $dd_1$  with a standard deviation  $\sigma_{min}$ , and the second one  $dd_2$  with deflection  $\sigma_{max}$ , are presented in Figure 3. Given various diagrams leads to overlapping ranges while selecting confidence intervals. Having value of  $\sigma_{min}$ , related value of  $|\sigma_{max} - \sigma_{min}|$  can be attributed to uncertainty of the observation. The difference is width of an overlap that can be perceived as a doubtfulness level.

Given the above mentioned data, one can seek for the truth of the question "what is the truth that the true observation is represented by point  $x^*$ " [1, 2]. For each  $x_i$  out of considered frame of discernment value of appropriate support is to be obtained. Concept of fuzzy probability sets can be useful to establish adequate relation.



**Fig. 1.** Adjacent confidence intervals with fuzzy limits and example membership grades for selected points.

Four fuzzy cumulated probability sets ( $\widetilde{S}_i$ ) were introduced in the vicinity of obtained position. Possibility of various values of measurement belonging to each of fuzzy probability sets are returned by sigmoidal membership functions. Example sets of figures refereeing to plotted in Figure 3 points are presented in included exploded insertion. Each of the sets contains four figures meaning particular point inclusion grades within each of probability sets. The latest number refers to probability that given point, located at right hand side of measurement, represents the true observation.

Belief function refers to established fuzzy probability sets and involved uncertainty. Fuzzy sets may be associated with cumulated probability calculated for specified confidence intervals while theoretical density distribution is considered. They can be related to bins when empirical distribution, histograms are involved.

## References

1. Filipowicz, W.: Imprecise data handling with MTE. In: LNAI 11683 N.T. Nguyen et al. (eds.), pp. 579–588. Springer (2019).
2. Filipowicz, W.: Mathematical Theory of Evidence in Navigation. In: Cuzzolin F. (ed.) Belief Functions: Theory and Applications, Third International Conference, BELIEF 2014 Oxford, UK, pp. 199 – 208. Springer International Publishing Switzerland (2014).
3. Yen, J.: Generalizing the Dempster – Shafer theory to fuzzy sets. IEEE Transactions on Systems, Man and Cybernetics 20/3 (1990).

# Deep learning approach to rare CNV detection

Lukasz Neumann<sup>1</sup>, Tomasz Gambin<sup>1</sup>, Wiktor Kuśmirek<sup>1</sup>, and Robert Nowak<sup>1</sup>

Institute of Computer Science, Warsaw University of Technology  
[lukasz.neumann@pw.edu.pl](mailto:lukasz.neumann@pw.edu.pl)

**Keywords:** Copy-number Variation · CNN · Imbalanced Training.

Copy-number variations (CNVs) in human genome can be a cause of various health issues. Automating the classification of rare CNVs could potentially improve the diagnostic process, as well as allow for a wider case studies on the available genomic data. Detection of CNVs is still challenging, despite numerous different attempts to solve the problem [3]. One strategy is using whole exome sequencing (WES) data, which we explore.

In this study we propose a machine learning approach to the problem of rare CNVs classification. Our method is based on read depth analysis. We use 1-D convolutional layers to fully utilize spatial information about neighbouring regions. Moreover, we use Importance Sampling to further improve model's performance. Influence of normalization techniques on the quality of trained model is discussed.

We show that proper clustering of samples [1] and preprocessing within clusters has a positive effect on model's performance. The proposed method has been tested using dataset from '1000 genomes' project [2]. We evaluate model using two prediction scenarios - per-window aggregated predictions and single region prediction. Despite high imbalance in the datasets our solution outperforms state-of-the-art methods in 'common' and 'all' CNV detection problems and is competitive in 'rare' CNV detection, with regards to the F1 and Average Precision scores.

## References

1. Kuśmirek, W., Szmurlo, A., Wiewiórka, M., Nowak, R., Gambin, T.: Comparison of knn and k-means optimization methods of reference set selection for improved cnv callers performance. *BMC Bioinformatics* **20** (12 2019). <https://doi.org/10.1186/s12859-019-2889-z>
2. Siva, N.: 1000 genomes project (2008)
3. Zhao, M., Wang, Q., Wang, Q., Jia, P., Zhao, Z.: Computational tools for copy number variation (CNV) detection using next-generation sequencing data: features and perspectives. *BMC Bioinformatics* **14**(11), S1 (Sep 2013). <https://doi.org/10.1186/1471-2105-14-S11-S1>

# Multilayer quantum-classical neural network\*

Piotr Gawron<sup>1</sup>[0000–0001–7476–9160] and Aleksandra  
Krawiec<sup>1</sup>[0000–0001–8390–6569]

Institute of Theoretical and Applied Informatics, Polish Academy of Sciences  
Bałycka 5, 44-100 Gliwice, Poland  
[{gawron, akrawiec}@itit.pl](mailto:{gawron, akrawiec}@itit.pl)

**Abstract.** We propose a two-layer quantum-classical classifier employing two-qubit sub-classifiers. We train and validate our approach using the famous Wine dataset having twelve features for each sample. We divide those features into three subsets and classify them using three sub-classifiers. The results obtained from sub-classifiers are combined and classified further by the fourth classifier in the second layer. We show that the obtained classification results validate usability of our approach.

**Keywords:** Classification · quantum machine learning · quantum neural networks.

## 1 Introduction

Quantum machine learning is a new field of research laying at the intersection of quantum computation and machine learning [3]. The idea of quantum computation was proposed at the beginning of the 1980s [7]. The first generation of quantum computers has already been built and nowadays they are available not only for specialists, but also for general public [4]. Quantum computers that exist today are limited in two fundamental ways – they have relatively small number of qubits and they are noisy [10]. Hence, they cannot be applied to execute such complicated quantum algorithms as for example Shor’s algorithm [13]. Modern research on quantum computation is currently focused on finding applications of existing and future Near-term Noisy Intermediate-Scale Quantum computers which can execute only shallow quantum circuits before quantum information processed by a computer becomes too noisy to be useful.

Typical scenario for a quantum program executed on a quantum computer is divided into the following computation stages. 1. A quantum computer is reset and quantum state  $|0\rangle$  is prepared. 2. A data-loading circuit  $U_1(x)$  transforming the initial state into quantum state  $|x\rangle$ , where  $x$  represents the input data, is executed. 3. A quantum program  $U_p(\theta)$  parametrized by parameters  $\theta$  is executed. 4. A measurement  $P$  is performed on the state of the quantum computer and measurement’s result is returned to the classical computer. During the computation process, from the reset up to the measurement, the quantum state of the

---

\* National Science Centre (NCN, Poland) under Grant No. 2016/22/E/ST6/00062.

computer should remain as coherent as possible. Unfortunately, each application of a quantum gate causes disturbance and the state of quantum computer decoheres [10]. Therefore it is desirable to design quantum programs that are composed of only few unitary gates i.e. use only shallow quantum circuits.

One of the examples of shallow quantum circuits are so-called variational circuits [8] which are often employed for machine learning tasks [2]. For example in [12] the authors propose a simple quantum binary classification procedure which is able to classify four-dimensional feature vectors. Due to the small number of features used for the classification, the  $U_1(x)$  procedure is relatively shallow. The proposed classifier is used to classify a subset of the famous Iris [1] dataset.

In this report we adapt the procedure designed to classify four-dimensional feature vectors into a procedure which is able to classify up to twelve-dimensional feature vectors. Our approach is based on parallel and sequential stacking of the classification procedure. We can effectively create a shallow-depth multi-layer quantum-classical neural network. We verify our approach on a subset of the well-known Wine dataset [6].

This report is organized as follows. In Section 2 we recall the construction of single-layer quantum neural network. Section 3 focuses on presenting our multi-layer approach. In Section 4 we describe our experiment and finally the conclusions are drawn in Section 5.

## 2 Single-layer quantum classifier

*Quantum gates* The classifier proposed in [2,12] had been inspired by [11] and consists of two quantum gates  $U_1(x)$  and  $U_p(\theta)$ , where  $x = (x_0, x_1, x_2, x_3) \in \mathbb{R}^4$  and  $\theta \in \mathbb{R}^{2 \times 3 \times nl}$ . The number  $nl$  denotes the number of quantum layers in the classifier. The gate  $U_1(x)$  (proposed in [9]) is presented in the form of program written in OpenQASM 2.0 [5] programming language in Fig. 1. The action of this gate on a two-qubit state is given by  $x_0 |0\rangle + x_1 |1\rangle + x_2 |2\rangle + x_3 |3\rangle = U_1(x) |0\rangle$ .

```
qreg r[2];
      Ry(-β1/2) r[1];
      CX r[0],r[1];
      Ry(β1/2) r[1];
```

|  |   |   |           |
|--|---|---|-----------|
| $R_y(-\beta_1/2)$ r[1];<br>$CX$ r[0],r[1];<br>$R_y(\beta_1/2)$ r[1]; | $X$ r[0];<br>$CX$ r[0],r[1];<br>$R_y(-\beta_0/2)$ r[1]; | $CX$ r[0],r[1];<br>$R_y(\beta_0/2)$ r[1]; | $X$ r[0]; |
|--|---|---|-----------|

**Fig. 1.** The representation of  $U_1(x)$  program in OpenQASM language. The following parameters are used in the program:  $\beta_0 = 2 \arcsin \sqrt{x_1^2} / \sqrt{x_0^2 + x_1^2}$ ,  $\beta_1 = 2 \arcsin \sqrt{x_3^2} / \sqrt{x_2^2 + x_3^2}$  and  $\beta_2 = 2 \arcsin \sqrt{x_2^2} / \sqrt{x_0^2 + x_1^2 + x_2^2 + x_3^2}$ .

The task of the program gate  $U_p(\theta)$  is to scramble the information stored in the quantum register. The gate  $U_p(\theta)$  is presented in Fig. 2 in the form of OpenQASM program.

$$\begin{aligned}
& U(\theta_{0,0}^{(i)}, \theta_{0,1}^{(i)}, \theta_{0,2}^{(i)}) \text{ r}[0]; \\
& U(\theta_{1,0}^{(i)}, \theta_{1,1}^{(i)}, \theta_{1,2}^{(i)}) \text{ r}[1]; \\
& CX \text{ r}[0], \text{r}[1];
\end{aligned}$$

**Fig. 2.** A single quantum layer of variational part of the quantum program. The gate  $U_p(\theta)$  consists of  $nl$  consecutive layers indexed by  $i$ .

*Classification function* The soft classification function  $f : \mathbb{R}^4 \rightarrow [-1, 1]$  is expressed as a sum of an expected value of the observable  $(\sigma_z \otimes \mathbb{1})$  in the state  $|\psi\rangle = U_p(\theta)U_1(x)|0\rangle$  and a real bias parameter  $b$ , that is  $f(x; \theta, b) = \langle (\sigma_z \otimes \mathbb{1}) \rangle_{|\psi\rangle} + b$ , where the expected value reads  $\langle (\sigma_z \otimes \mathbb{1}) \rangle_{|\psi\rangle} = \langle \psi | (\sigma_z \otimes \mathbb{1}) | \psi \rangle$ .

*Data preprocessing* The following preprocessing steps are performed on the data. First, the data is divided by the maximal value in the dataset and then for each sample  $x_{\text{orig}}$  the feature values are normalized to unity i.e.  $x = x_{\text{orig}} / \|x_{\text{orig}}\|_2$ .

*Training and validation* For data  $X$ , the sequences of true labels  $Y$  and predicted scores  $\hat{Y}' = (f(x; \theta, b))_{x \in X}$ , the loss and accuracy functions are defined as  $\text{loss}(Y, \hat{Y}') = \sqrt{\sum_{y_i \in Y, \hat{y}'_i \in \hat{Y}'} (y_i - \hat{y}'_i)^2 / |Y|}$ , where  $\hat{Y}' = (f(x; \theta, b))_{x \in X}$  and  $\text{accuracy}(Y, \hat{Y}) = \left( \sum_{y_i \in Y, \hat{y}_i \in \hat{Y}} \mathbb{1}_{y_i \neq \hat{y}_i} \right) / |Y|$ , where  $\hat{Y} = (\text{sgn}(f(x; \theta, b)))_{x \in X}$ .

The training process is performed by fitting parameters  $\theta$  and  $b$  of the function  $f(x; \theta, b)$  to the training data  $X \times Y$ , where  $X$  is a set containing preprocessed samples and  $Y$  contains the class labels  $\{-1, 1\}$ .

The training process is done in batches using Nesterov Momentum Optimizer in order to minimize the loss function with respect to parameters  $\theta$  and  $b$ . The size of batch is fixed to five and the learning rate is set to 0.005. The initial values of  $\theta$  are sampled from normal distribution with zero mean and standard deviation equal to 0.01. The initial value of bias  $b$  is set to zero.

### 3 Multilayer quantum-classical classifier

In order to use the aforementioned classifier to classify a 12-feature samples we create a two-layer cascade of identical classifiers. The features one to four are fed to the first classifier of the first layer, five to eight to the second classifier and nine to twelve to the third one. The values  $y^{[i]} = f(x^{[i]}; \theta^{[i]}, b^{[i]})$  for  $i \in \{1, 2, 3\}$ , obtained from classifiers of the first layer, are transferred to the second-layer classifier after transformation to  $(y^{[i]} + 1)/2$ . The fourth input to the second-layer classifier is set as zero. The value of  $\text{sgn}(f(x^{[4]}; \theta^{[4]}, b^{[4]}))$ , produced by the second-layer classifier, is the classification result.

### 4 Experiment and results

To verify our approach we use a subset of the Wine dataset [6]. The dataset contains 130 samples having 12 features. We select only two first classes in order to

use a binary classifier. The data is randomly divided into training and validation sets in ratio 75%–25%.

The model is implemented in PennyLane Python library [2]. The quantum devices are simulated and ideal. We train our classifier for 100 epochs. The best classification accuracy 84.8% on the validation set was obtained in 52-nd epoch.

## 5 Conclusions

We have presented an example of cascade of quantum two-qubit classifiers creating a quantum-classical neural network. This approach shows the potential of small quantum devices for performing classification of datasets that cannot be encoded directly on a quantum state of such small devices. While classification accuracy might not be satisfactory, we believe that building a larger network of small quantum classifiers should lead to better accuracy.

## References

1. Anderson, E.: The species problem in iris. *Annals of the Missouri Botanical Garden* **23**(3), 457–509 (1936)
2. Bergholm, V., Izaac, J., Schuld, M., Gogolin, C., Killoran, N.: PennyLane: Automatic differentiation of hybrid quantum-classical computations. arXiv preprint arXiv:1811.04968 (2018)
3. Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N., Lloyd, S.: Quantum machine learning. *Nature* **549**(7671), 195 (2017)
4. Castelvecchi, D.: Ibm’s quantum cloud computer goes commercial. *Nature News* **543**(7644), 159 (2017)
5. Cross, A.W., Bishop, L.S., Smolin, J.A., Gambetta, J.M.: Open quantum assembly language. arXiv preprint arXiv:1707.03429 (2017)
6. Dua, D., Graff, C.: UCI Machine Learning Repository (2019), <http://archive.ics.uci.edu/ml>
7. Feynman, R.P.: Quantum mechanical computers. *Optics news* **11**(2), 11–20 (1985)
8. McClean, J.R., Romero, J., Babbush, R., Aspuru-Guzik, A.: The theory of variational hybrid quantum-classical algorithms. *New Journal of Physics* **18**(2), 023023 (2016)
9. Mottonen, M., Vartiainen, J.J., Bergholm, V., Salomaa, M.M.: Transformation of quantum states using uniformly controlled rotations. *Quantum Information and Computation* **5**(6), 467–473 (2005)
10. Preskill, J.: Quantum computing in the nisq era and beyond. *Quantum* **2**, 79 (2018)
11. Schuld, M., Bocharov, A., Svore, K., Wiebe, N.: Circuit-centric quantum classifiers. arXiv preprint arXiv:1804.00633 (2018)
12. Schuld, M., Killoran, N.: Variational quantum classifier, [https://github.com/XanaduAI/pennylane/blob/v0.4.0/examples/Q3b\\_variational-classifier-iris.py](https://github.com/XanaduAI/pennylane/blob/v0.4.0/examples/Q3b_variational-classifier-iris.py)
13. Shor, P.W.: Algorithms for quantum computation: Discrete logarithms and factoring. In: Proceedings 35th annual symposium on foundations of computer science. pp. 124–134. Ieee (1994)

# Advanced model supporting football team building

Bartosz Ćwikliński<sup>1</sup>, Agata Gielczyk<sup>1[0000–0002–5630–7461]</sup>, and Michał Chorąś<sup>1</sup>

UTP University of Science and Technology  
Faculty of Telecommunications, Computer Science  
and Electrical Engineering, Bydgoszcz, Poland  
[agata.gielczyk@utp.edu.pl](mailto:agata.gielczyk@utp.edu.pl)

**Abstract.** Advanced data analysis is now extensively used in team sports to measure the performance of both individual players and teams. Coaches and staff members (also data analysts) track statistics during games in order to make quick decisions. Metrics and data analysis can also be used for football (soccer) team building (e.g. between seasons or in transfer windows), which is the focus of this paper. A model is proposed which can help to choose best-fitting players within a given budget. Apart from technical and physical factors, the use of psychological aspects is suggested.

**Keywords:** Data analysis · Data science · Sports analytics · Machine learning

## 1 Introduction

Recently the popularity of artificial intelligence (AI) has raised rapidly and now it is implemented widely in many domains of our everyday life. Artificial intelligence gives an opportunity to use human-like thinking in computer systems. One of the most popular implementations of AI methods is prediction - using AI methods it is possible to predict the weather or currency rates. Lately, some interesting publications concerning the use of AI in sports have been proposed. Most of them use machine learning in order to predict the results of matches in general [1] or basketball games in NBA [4]. In [3] the authors presented the prediction of high potential archers based on their physical parameters, while in [5] injury risk of players across a single National Rugby League season is predicted. Also in [2] a learning-based model to forecast sports injuries was introduced, using data from various information systems.

In this article, we present a model that combines football data (e.g. scores, goals) with AI methods. The possible advantages of using this kind of model to predict the choice of the most necessary and appropriate players for the team in the team-building phase (or in transfers) are discussed. In Section 2 some commercial and scientific solutions that have been already proposed are presented. In Section 3 the proposed model is described, with its main idea, metrics, and advantages. Finally, in the last section, conclusions and future work are provided.

## 2 Motivation

Yogi Berra, the baseball legend, said '*Baseball is 90% mental. The other half is physical.*' when he was asked about data science in sports. Since then, people's awareness of data and predictions has changed significantly. Thanks to technological progress, experts are able to check and analyze various information. An interesting example can be observed in season 2002 of the baseball team Oakland Athletics. Due to financial reasons, general manager Billy Beane and assistant Peter Brand decided to use technological progress to strengthen the team. Using computer analysis, they selected players who should join the team to build the strongest lineup using a low budget. The key to making the decision was the ratio of the quality of individual skills to the market price. After completing the team, tactics were established that allowed to fully use the potential of the players. The result was winning the American League West Division and a record victory in 20 consecutive games. Of course, there are also critics of data science in sports claiming that deciding the outcome of games or seasons (e.g. great past NBA players like Charles Barkley) is pure talent. Still, the authors believe that data analytics, and the presented model in particular, can help in bringing the right players with the right talent, skills and mental characteristics to the team.

Therefore this paper presents a way to create and verify a model that supports football (soccer) team-building with the support of AI methods and models.

## 3 Proposed method

### 3.1 Model description

The general parameters of a player in football can be: technical, physical and psychological. The details of the utilized parameters are presented in Fig. 1.

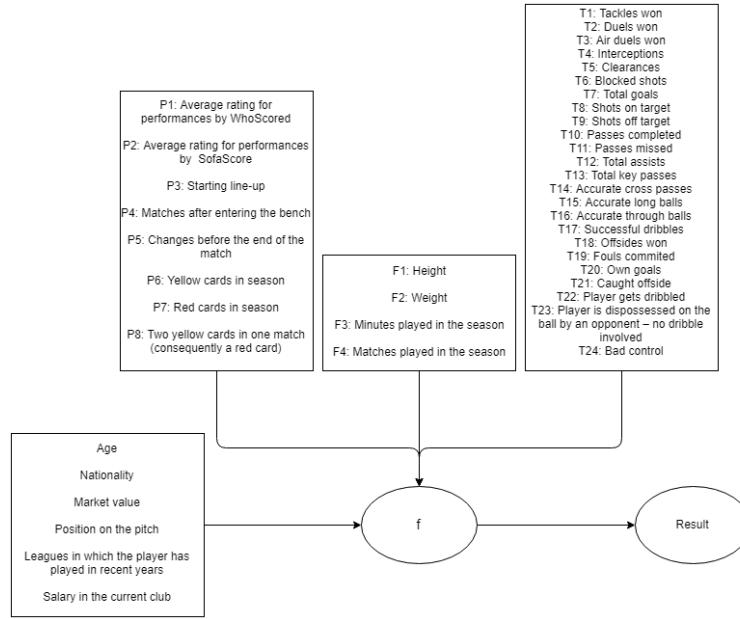
Fig. 1 presents a model that estimates the player's impact on the team. Before starting, it is essential to check the player's basic criteria, e.g. market value, age or nationality. All aspect parameters are non-negative. A scored goal is more valuable than an assisted one, thus, we use weights to imitate the actual translation of each metric into the impact on the team's game.

The technical aspect defines the skills associated with playing football. The basic metrics are: goals scored, number of assists, the accuracy of passes, and the effectiveness of tackles. These statistics are easily available and only reflect the general image of the player. If any behaviour of a player is taken into account during matches, both when he is at the ball and without contact with the ball, the data received will be very detailed. Various correlations of acquired metrics enable the creation of a very accurate player profile.

The physical aspect means the player's body structure and general abilities, not only related to football. Height, weight, muscles, or the jumping height, speed and acceleration are parameters that affect the player's skills. Each position in a team's formation has a set of recommended features that a player fulfilling a

given function should have, e.g. the defender should be strong and tall, and the winger fast and agile.

The psychological aspect is the most complex of these factors. It consists of parameters that cannot be studied and which affect the use of other aspects. A player who is in a poor psychological condition is not able to use his full potential and his abilities and physical conditions do not bring full benefits. Adequately, a footballer who is not affected by any mental discomfort is able to use all his abilities and further develop. In the case of football, the psychological aspect consists of relations with other team players, coaching staff, the style of play in a given league, the country in which the player lives and purely private circumstances. The value of the psychological parameters should be in the range of  $< 0.5, 1.5 >$ . A lower value indicates a poor mental situation of the player, while a high value means mental comfort allowing further development.



**Fig. 1.** The overall pipeline of the model

To predict the success of a football transfer, we use the formula expressed with Eq 1, where  $T_x$  is the technical aspect,  $F_y$  is the physical aspect and  $P_z$  is the psychological aspect. The prediction is performed only if the initial data is suitable.

$$f = \left( \sum_{x=1}^X T_x + \sum_{y=1}^Y F_y \right) \cdot \sum_{z=1}^Z P_z \quad (1)$$

### 3.2 Model verification

There are many websites providing statistics of clubs and players for free. The WhoScored website (<https://www.whoscored.com/>) provides many match statistics, while Transfermarkt (<https://www.transfermarkt.com/>) focuses on transfers. Google also presents a lot of football-related data, including a percentage analysis of the teams' chances of winning a match.

The model is currently being validated by analyzing the most important football transfers carried out in the 2018/2019 and 2019/2020 seasons. We will use the collected data from the previous five seasons to get real results. Given the changes that may occur in a player over the years, we will adapt the weights.

## 4 Conclusions

Artificial intelligence has been introduced into numerous elements of our daily lives recently. This paper proposes to use data science for football transfer prediction and team building efficiency. The interesting approach is to use both real statistics and physical metrics, as well as the mental and psychological aspects. The presented model is a preliminary work that will be validated in the next phases of this project.

## References

1. Bunker, R.P., Thabtah, F.: A machine learning framework for sport result prediction. *Applied computing and informatics* (2017)
2. Liu, G., Sun, H., Bai, W., Li, H., Ren, Z., Zhang, Z., Yu, L.: A learning-based system for predicting sport injuries. In: *MATEC Web of Conferences*. vol. 189, p. 10008. EDP Sciences (2018)
3. Musa, R.M., Majeed, A.P.A., Taha, Z., Chang, S.W., Nasir, A.F.A., Abdullah, M.R.: A machine learning approach of predicting high potential archers by means of physical fitness indicators. *PloS one* **14**(1), e0209638 (2019)
4. Thabtah, F., Zhang, L., Abdelhamid, N.: Nba game result prediction using feature analysis and machine learning. *Annals of Data Science* **6**(1), 103–116 (2019)
5. Welch, M.C., Cummins, C., Thornton, H., King, D., Murphy, A.: Training load prior to injury in professional rugby league players: Analysing injury risk with machine learning. *ISBS Proceedings Archive* **36**(1), 330 (2018)

# Application of quantum annealing to combinatorial optimization problems

Piotr Gawron[0000-0001-7476-9160]

Institute of Theoretical and Applied Informatics, Polish Academy of Sciences  
Batycka 5, 44-100 Gliwice, Poland  
[gawron@iiit.is.pl](mailto:gawron@iiit.is.pl)

Quantum annealing [2] is one of the models of quantum computation. Although currently existing quantum annealers are unable to perform universal quantum computation they are able to provide solutions for a very important problem of finding low energy states of classical binary Hamiltonians. Finding the minimal value of Hamiltonian problem is important since it is well known to be NP-complete. It is very unlikely that quantum annealers will ever be able to find minimal values for generic classical Hamiltonian problems efficiently but it is postulated that quantum annealers can provide good approximate solutions quickly or an exact solutions for a limited class of Hamiltonians.

Quantum annealers can be applied to (approximately) solve such problems as: travelling salesman problem, constraints satisfaction problems, graph colouring sampling from Gibbs distribution, non-negative matrix factorization and simulation of dynamic Hamiltonian systems.

For over twenty years a Canadian company—D-Wave Systems Inc. develops quantum annealers. The most recent publicly accessible quantum annealer is based on so called Chimera architecture [3] having up to 2048 qubits and 5600 couplers. Recently the company announced that next year a new architecture named Pegasus [1] having larger number of qubits and denser connectivity will be available. In this context the architecture denotes the topology of possible interactions between the qubits.

While the academic dispute is not settled on the subject of whether quantum annealers will be ever able to solve optimization problems faster and better than classical machines, it is difficult to deny the fact, that quantum annealers are a new and possibly revolutionary computational tool.

## References

1. Dattani, N., Szalay, S., Chancellor, N.: Pegasus: The second connectivity graph for large-scale quantum annealing hardware. arXiv preprint arXiv:1901.07636 (2019)
2. Kadowaki, T., Nishimori, H.: Quantum annealing in the transverse ising model. Physical Review E **58**(5), 5355 (1998)
3. King, A.D., Carrasquilla, J., Raymond, J., Ozfidan, I., Andriyash, E., Berkley, A., Reis, M., Lanting, T., Harris, R., Altomare, F., et al.: Observation of topological phenomena in a programmable lattice of 1,800 qubits. Nature **560**(7719), 456 (2018)



# Aggregate and individual approaches for building neural models for mid-term electric energy demand forecasting

Paweł Pelka<sup>1</sup>[0000–0002–2609–811X]

Electrical Engineering Faculty, Czestochowa University of Technology, Czestochowa,  
Poland  
p.pelka@el.pcz.czest.pl

**Abstract.** Medium-term electric energy demand forecasting is becoming a key tool for power system operation, energy management and maintenance scheduling. This paper describes aggregate and individual approaches to forecasting monthly electricity demand based on multilayer perceptron model which approximates a relationship between historical and future demand patterns. Energy demand time series exhibit long-run trend, non-stationarity, random noise and seasonal fluctuations. To simplify the forecasting problem the monthly demand time series is represented by patterns of yearly periods, which unify data and filter out a trend. An output variable is encoded using coding variables describing the process. As an illustration, the proposed neural network models were applied to monthly energy demand forecasting for 35 European countries.

**Keywords:** Medium-term load forecasting · Multilayer perceptron · Pattern-based forecasting.

## 1 Introduction

Forecasting of power system load is a key activity built into the processes of the system current control and its operation planning in a longer horizon. Accurate predictions are necessary to operate the system. Current demand has to be covered by production at any time, because electricity cannot be stored in larger quantities. The production and transmission costs depends on the accuracy of forecasts, so it plays a large role in reliability of the electricity supplies to recipients [1].

There are two general categories, in which the methods of medium-term prediction of power systems loads can be divided [2]. First group is named autonomous modeling approach, where information about primarily historical loads and sometimes weather conditions are applied as input variables to predict electrical power loads. Models from this category are more often used for stable economies, without sudden changes affecting the electricity demand. Second group is called the conditional modeling, which idea consist of long-term planning, forecasting of energy policy and it is more focused on the economic analysis.

The presented neural models belong to the first group. We consider two approaches to learning the model. In the individual approach we train the model on each dataset separately while in the aggregate approach we train the model once on all combined datasets. To unify data and filter out a trend we represent time series using patterns of yearly cycles.

## 2 Neural forecasting model

Monthly electricity load time series exhibit yearly cycles which are transformed into input patterns. An input pattern  $\mathbf{x}_i = [x_{i,1} x_{i,2} \dots x_{i,n}]^T$  of length  $n = 12$  is a vector of predictors representing  $n$  timepoints preceding the forecasted point, i.e. the time series sequence covering a seasonal cycle [1]  $X_i = \{E_{i-n+1}, E_{i-n+2}, \dots, E_i\}$ . The vector  $\mathbf{x}_i$  is a normalized version of the demand vector  $[E_{i-n+1} E_{i-n+2} \dots E_i]^T$ . Its components are calculated as follows [3], [4]:

$$x_{i,t} = \frac{E_{i-n+t} - \bar{E}_i}{D_i} \quad (1)$$

where  $t = 1, 2, \dots, n$ ,  $\bar{E}_i$  is the mean value of the sequence  $X_i$ , and  $D_i = \sqrt{\sum_{j=1}^n (E_{i-n+j} - \bar{E}_i)^2}$  is a measure of its dispersion.

The normalized x-vectors for different demand sequences have all mean value equal to zero, the unity length, and the same variance. Thus, the input data are unified. They carry information about shapes of yearly cycles [1].

The forecasted variable is  $E_{i+\tau}$ , i.e. electricity demand at month  $i + \tau$ , where  $\tau \geq 1$  is a forecast horizon. This variable is also encoded to unify data filtering the trend out. The encoded demand is:

$$y_{i,\tau} = \frac{E_{i+\tau} - \bar{E}_*}{D_*} \quad (2)$$

In this equation coding variables  $\bar{E}_*$  and  $D_*$  should be determined for the seasonal cycle covering the timepoint  $i + \tau$ . But this future cycle is unobtainable in the moment of forecasting (timepoint  $i$ ). Thus, the coding variables cannot be determined from it. We use in their place coding variables determined for the known preceding seasonal cycle  $X_i$ , i.e.  $\bar{E}_* = \bar{E}_i$ ,  $D_* = D_i$ .

Having transformed input and output data the training set is composed. It includes pairs of x-patterns and corresponding encoded output variables  $y$ :  $\Phi = \{(\mathbf{x}_i, y_{i,\tau}) | \mathbf{x}_i \in \mathbb{R}^n, y_{i,\tau} \in \mathbb{R}, l = 1, 2, \dots, N\}$ . The x-pattern size determines a number of NN inputs, 12 in our case. The number of hidden neurons is a variable, adjusted to the complexity of the target function which maps  $\mathbf{x}$  onto  $y$ . When the forecast horizon is  $\tau$ , the neural model has one output,  $y$ . This variant of the forecasting model is marked by A1 in the simulation study section. But other variant is also considered, marked by A2, where the network forecasts all seasonal cycle for the next year. In this case it has  $n = 12$  outputs for  $\tau = 1, 2, \dots, 12$ , and the training set is  $\Psi = \{(\mathbf{x}_i, \mathbf{y}_i) | \mathbf{x}_i \in \mathbb{R}^n, \mathbf{y}_i \in \mathbb{R}^n, l = 1, 2, \dots, N\}$ , where  $\mathbf{y}_i = [y_{i,1} y_{i,2} \dots y_{i,n}]$ . Variants A1 and A2 are used for twelve months ahead

forecasts. In experimental part of the work we test the NNs also in one month ahead forecasting (variant B). In this case the training set is  $\Phi$ , where  $\tau = 1$  and x-pattern represents the sequence of twelve months directly preceding the forecasted month [1], [3].

The forecasting NN model generates the y-patterns or their components. To obtain the forecasts of the monthly demand we use transformed equation (2):

$$\hat{E}_{i+\tau} = \hat{y}_{i,\tau} D_* + \bar{E}_* \quad (3)$$

We use two approaches for building the NN models. In the individual approach, Ind, we build the NN model individually for each dataset (containing monthly loads of a country). So, for  $m$  countries we construct  $m$  models. Each of them is trained on the time series of loads of a certain country. In the aggregate approach, Agg, the datasets for all  $m$  countries are combined and we construct only one NN model for all countries.

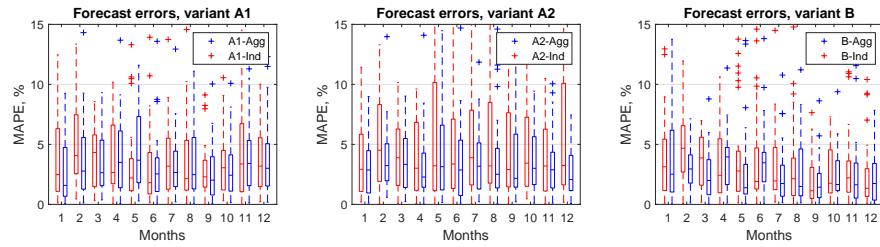
In all cases the NN has a single hidden layer with sigmoidal neurons. It learns using Levenberg–Marquardt algorithm with Bayesian regularization, which minimizes a combination of squared errors and the weights. This prevent overfitting [1]. The model hyperparameters, i.e. the number of neurons, were selected on the historical data.

### 3 Simulation Study

In this section, the proposed neural models are evaluated on real-word data including monthly electricity demand for 35 European countries. The data are taken from the publicly available ENTSO-E repository ([www.entsoe.eu](http://www.entsoe.eu)). They cover in most cases the time period from 1998 to 2014. A goal is to construct the forecasting models for 2014 using historical data.

Fig. 1 shows the boxplots of the test errors (mean absolute percentage errors, MAPE) for each month over all 35 countries. The mean and median of the forecast errors are shown in Table 1. In this table also are shown:  $IQR$  - interquartile range as a measure of error dispersion, Best cases - in how many cases the given approach (Agg or Ind) was better than the opponent one, and #neurons - average number of hidden neurons selected on the data from 2013. In all variants, Agg approach is significantly better than the Ind one. For A1 variant it is also better than the comparative models using classical methods: ARIMA (5.65) and exponential smoothing, ETS (5.05). For A2 and B variants, errors generated by the comparative models were similar to Agg approach.

In the aggregate approach NN needs between 2.00 and 3.08 hidden neurons, while in the individual approach it needs more neurons, from 3.11 to 5.22. So, Agg approach leads to the more compact NN architecture. The total learning time for Agg was over ten times shorter than for Ind.

**Fig. 1.** Errors for aggregate and individual approaches.**Table 1.** Forecast errors and optimal number of neurons for A1, A2 and B variants.

| Model           | A1-Ind | A1-Agg | A2-Ind | A2-Agg | B-Ind | B-Agg |
|-----------------|--------|--------|--------|--------|-------|-------|
| $MAPE_{mean}$   | 5.56   | 4.69   | 7.81   | 4.84   | 4.96  | 3.39  |
| $MAPE_{median}$ | 3.55   | 3.52   | 4.05   | 3.08   | 2.87  | 2.74  |
| $IQR$           | 2.84   | 2.66   | 4.54   | 2.48   | 2.30  | 1.59  |
| Best cases, %   | 20     | 80     | 37.14  | 62.86  | 31.43 | 68.57 |
| #neurons        | 5.22   | 2.66   | 3.11   | 2.00   | 5.15  | 3.08  |

## 4 Conclusion

In this work the neural network models for pattern-based forecasting monthly electricity demand was examined. The models work on patterns representing normalized yearly seasonal cycles of the demand time series. Input patterns express shapes of the yearly cycles after filtering out a trend and unifying a variance. Also the output data are unified using coding variables determined on the historical data. The pattern approach simplify the forecasting problem so the model does not have to capture the complex nature of the process. This leads to model simplification (only several hidden nodes) and faster learning.

The aggregate approach of NN learning generated more accurate forecasts than individual approach. It leads also to the more compact NN architecture and several times shorter learning time.

## References

1. Pelka, P., Dudek, G.: Pattern-Based Forecasting Monthly Electricity Demand Using Multilayer Perceptron. Proc. 18th Int. Conf. Artificial Intelligence and Soft Computing ICAISC 2019. LNCS **11508**, Springer, 663–672, (2019)
2. Ghiassi, M., Zimbra, D.K., Saidane, H.: Medium term system load forecasting with a dynamic artificial neural network model. Electric Power Systems Research **76**, 302–316 (2006)
3. Pelka, P., Dudek, G.: Neuro-fuzzy system for medium-term electric energy demand forecasting. Proc. 38th Int. Conf. Information Systems Architecture and Technology ISAT 2017. AISC **655**, Springer, 38–47, (2018)
4. Dudek, G., Pelka, P.: Medium-term electric energy demand forecasting using Nadaraya-Watson estimator. IEEE 18th Int. Conf. Electric Power Engineering EPE 2017, 1–6 (2017)

# Classification of Tree Species from Limited Dataset of Bark Images Using Convolutional Neural Networks

Wojciech Czarnecki<sup>1,2[0000-0002-2641-9542]</sup> and  
Piotr Gawron<sup>1[0000-0001-7476-9160]</sup>

<sup>1</sup> Institute of Theoretical and Applied Informatics, Polish Academy of Sciences,  
ul. Baltycka 5, 44-100 Gliwice, Poland  
[{wczarnecki, gawron}@iiit.is.pl](mailto:{wczarnecki,gawron}@iiit.is.pl)

<sup>2</sup> University of Economics in Katowice, ul. 1 Maja 50, 40-287 Katowice, Poland

**Abstract.** Identification of tree species using bark images is a difficult problem that can be useful in many forestry tasks. While researchers are developing large datasets for tree species classification, we tried to achieve satisfactory classification results using a small dataset. In this work, we present our experimental results obtained from application of deep neural networks to this problem. More specifically, we obtain an average accuracy of 99.17% in tree species recognition task.

**Keywords:** deep learning · convolutional neural networks · trees classification.

## 1 Introduction

Automatic classification of tree species from their images is a task that can find many applications in dendrology and forestry. When faced with the problem of recognizing tree species, many people use attributes such as the appearance of their leaves, needles or fruits [3]. The bark of a tree is usually present on the tree despite seasonal changes. Therefore images of bark can be acquired all-year long and fed to a classification system. However, the task of bark images classification is not easy, because some species have only very subtle differences in their bark structure. The task of tree species recognition using bark images is difficult even for human experts [3]. For example, a forestranger achieved a classification accuracy of 77.8%, whereas a biologist had 56.6% on the Austrian Federal Forests (AFF) dataset [3]. Recent advances in deep learning have shown that the results of neural networks in visual recognition tasks are able to outperform humans' results [4]. In our experiments we check if tree bark image classification can be performed efficiently even using a small training dataset.

This paper is organized as follows. In Section 2, we discuss existing methods and datasets used to classify tree species. In Section 3, we present the dataset we use, the net architecture and details of the training protocol. Section 4 presents the results of our experiments. Finally, Section 5 concludes this paper.

## 2 Related work

Automatic identification of tree species based on bark images is usually formulated as texture recognition problem. In [1, 8] the authors employ Local Binary Patterns (LBP) for classification. The authors of [3] applied one-vs-all SVM classifier to the same task and obtained the accuracy of 69.7%. With the advent of deep learning approaches results of images classification improved [6]. In work [2] which applied deep learning the authors achieved the accuracy of 97.81%, but the authors had to prepare a large dataset called BarkNet 1.0 containing over 23000 images of the bark.

## 3 Experiments

**Dataset.** In our experiments we used a publicly available dataset Trunk12 [9]. It contains 393 images of bark of 12 different trees species found in Slovenia. Each class consists of 30–45 images with a resolution of  $3000 \times 4000$  pixels. All images were acquired using Nikon COOLPIX S3000 camera, in the same conditions: the distance from trees was fixed and the lighting conditions were similar.

**Architecture.** Our tool of choice was PyTorch 1.1.0 [7] and its implementation of the ResNet-18 network. In one of experiments we used the network weights that were pre-trained on ImageNet dataset as presented in [5].

**Training details.** The input images were resized to  $224 \times 224$  pixels and normalized. The dataset was divided into training, validation and test set. During each experiment, 6 random images of each class were selected as a validation set, 6 of each class as a test set, and a remaining images were assigned to the training set. Then, due to the small number of images in the training set, we augmented the data by applying vertical flips, horizontal flips and colour jittering.

The validation set was used in stopping criterion. The training stopped after 50 epochs since last improvement of accuracy on the validation set was observed. The final network is the one having best accuracy obtained on the validation set. Initial learning rate—a hyper-parameter—was dropped by a factor of 10 after every 20 epochs without improvement of accuracy. In the experiments the accuracy was evaluated over the test set. The reported accuracy is an average over 10 experiment runs following the repeated random sub-sampling validation protocol. Two configurations of the experiments were chosen: with and without transfer learning. In the case of transfer learning application the network weights pre-trained on ImageNet dataset were used.

We performed a grid search for the following optimizers and hyper-parameters:

- optimizer: *adagrad*, *adam*, *SGD*;
- mini-batch size: 16, 32, 64;
- initial learning rate: 0.0001, 0.001, 0.01;
- weight decay: 0, 0.0001, 0.001, 0.01.

## 4 Results

### 4.1 Randomly initialized weights

Without data augmentation we obtained 86.25% accuracy with the *adam* optimizer, mini-batch size of 64, initial learning rate of 0.001 and weight decay of 0.01. After adding data augmentation the accuracy increased to 91.67% with the same hyper-parameters. Figure 1 shows confusion matrix for this set of experiments.

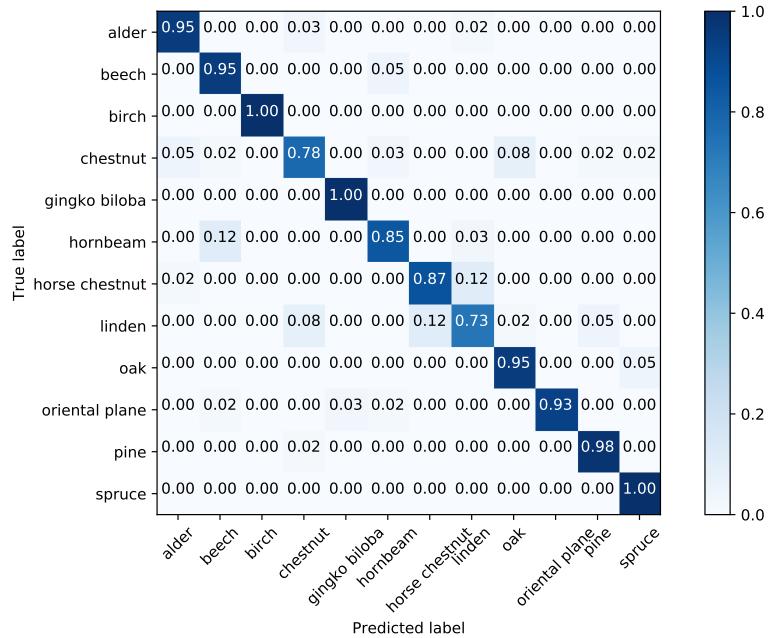


Fig. 1: Normalized confusion matrix averaged over 10 experiment runs following the repeated random sub-sampling validation protocol.

### 4.2 Pre-trained network

Finally, we achieved accuracy of 99.17% using transfer learning. Mini-batch size was set to 64. The *adam* optimizer again gave the best results. For a network with pre-trained weights a lower initial learning rate intuitively should work better what is supported by the results of our experiments, we obtained best results for learning rate of 0.0001. The experiments suggested that regularization in form of weight decay was not necessary. Data augmentation was applied.

## 5 Conclusion

In this paper we showed that it is possible to achieve satisfactory results in task of identification of tree species from bark images using Convolutional Neural Networks even with a small dataset. Despite the complexity of the problem, achieving 99.17% accuracy, we outperformed human experts' results.

Nevertheless, our results aren't as meaningful as these obtained on BarkNet 1.0 [2]. The network trained on a small dataset is more likely to get adapted to specific data acquisition conditions.

## Acknowledgement

The authors would like to thank Bartosz Grabowski and Wojciech Masarczyk for help in understanding many issues in the field of machine learning. Special thanks to Maciej Psych Smykowski and Wojciech Zarzycki for suggesting the problem.

## References

1. Boudra, S., Yahiaoui, I., Behloul, A.: A comparison of multi-scale local binary pattern variants for bark image retrieval. ACIVS 2015: Advanced Concepts for Intelligent Vision Systems pp. 764–775 (Oct 2015)
2. Carpentier, M., Giguere, P., Gaudreault, J.: Tree species identification from bark images using convolutional neural networks. 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) pp. 1075–1081 (Oct 2018)
3. Fiel, S., Sablatnig, R.: Automated identification of tree species from images of the bark, leaves or needles. Proceedings of the 16th Computer Vision Winter Workshop pp. 67–74 (2011)
4. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. 2015 IEEE International Conference on Computer Vision (ICCV) pp. 1026–1034 (Dec 2015)
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 770–778 (Jun 2016)
6. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. NIPS'12 Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1 pp. 1097–1105 (Dec 2012)
7. Paszke, A., Gross, S., Chintala, S., Chanan, G.: Pytorch 1.1.0. <https://pytorch.org/> (Apr 2019), accessed on 2019-09-05
8. Sulc, M.: Tree identification from images (May 2014)
9. Švab, M.: Computer-vision-based tree trunk recognition. Bsc Thesis, (Mentor: doc. dr. Matej Kristan), Fakulteta za računalništvo in informatiko, Univerza v Ljubljani (2014)

# Decision tree integration algorithm using static regions of competence and geometric representation

Jędrzej Biedrzycki [0000–0002–4924–1759]

Wrocław University of Science and Technology Wybrzeże Wyspiańskiego 27, 50-370  
Wrocław, Poland [jedrzej.biedrzycki@pwr.edu.pl](mailto:jedrzej.biedrzycki@pwr.edu.pl)

**Abstract.** Ensemble of classifiers provides possible improvement in classification quality. One of the most common techniques of integration of multiple classifiers is majority voting. In this paper an algorithm based on majority voting is proposed. Static areas of competence and weighting functions based on geometrical representation of decision tree are used.

**Keywords:** multiple classifier system · ensemble of classifiers · decision tree · majority voting

## 1 Introduction

Classification quality is very often improved by using committee of classifiers, also known as Multiple Classifier System (MCS) or Ensemble of Classifiers (EoC) ([5, 8]). A pool of classifiers is created and their responses are composed into integrated model. The reasons for the use of a classifier ensemble include, for example, the fact that single classifiers are often unstable (small changes in input data may result in creation of very different decision boundaries).

Due to differences between quality of classification of base models over different areas of classification space the areas of competence were introduced([3]). These can be dynamic (varying with validation results) and static splits of feature space([4]).

The majority voting (MV) is a combining method that works at the abstract level – output as a single label from base models is taken into consideration. The MV algorithm is defined as:

$$\Psi_{MV}(x) = \arg \max_{\omega} \sum_{k=1}^K I(\Psi_k(x), \omega), \quad (1)$$

where  $I(\cdot)$  is the indicator function with the value 1 in case of correct classification of the object described by the feature vector  $x$ , i.e. when  $\Psi_k(x) = \omega$  and 0 otherwise. Weighted MV can be considered, where in equation 1 each operand is weighted with custom function.

## 2 Related work

Geometric approach to classification problem by Voronoi cells utilization was recently examined by Polianskii and Pokorny ([6]). This approach was tested using SVM, nearest neighbor and random forest classifiers.

Local confidence was examined in order to provide better classification compared to validation over entire classification space([4]). Combining complementary characteristics of the base models outperforms individual classifiers and several other integration methods.

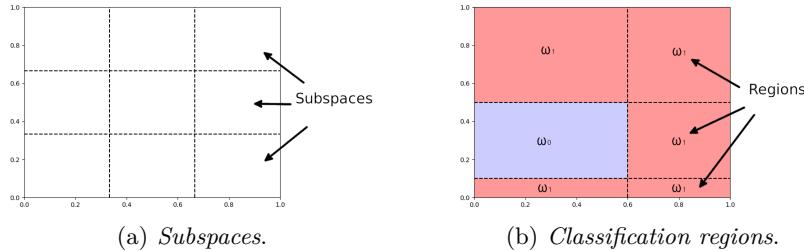
For over a decade now a new approach to developing integrated classifiers has been studied – using geometrical representation [7]. Based on operations in geometrical space generated by real-valued features this procedure has proven itself to be effective in comparison to others, commonly used integration techniques such as majority voting. Relevant improvement in classification by applying different functionals was obtained. The examples consist of weighted mean and median ([2]) or harmonic mean ([1]).

## 3 Proposed method

Geometrically decision tree can be considered as a finite set of cubes of points with same label. They are non-overlapping and compose the entire classification space. They will be further referred to as regions.

The whole dataset (training and testing subsets combined) generates a cubic space, that can be divided into smaller ones. Those cubical sets (further referred to as subspaces) are of the same shape as the original dataset but of different size, since cube is divided into same amount of parts along every dimension (feature axis).

Definitions of region and subspace are depicted using two-dimensional space in 1.



**Fig. 1.** Graphical explanation of *subspace* and *classification region*.

For each subspace candidate is resolved as label of classification region that spans the midpoint of considered subspace. For every such candidate weight is

derived based on the area (volume) of classification region. Finally all intermediate results are aggregated by summing their weights and the class with the largest weight is assigned to the subspace.

To keep the notation consistent the classifier that maps classification region into a label will be denoted as  $\Upsilon(A) \equiv \forall_{x \in A} \Psi(x)$ , provided  $\forall_{x_1, x_2 \in A} \Psi(x_1) = \Psi(x_2)$ .

Let  $K$  ( $k \in \{1, 2, \dots, K\}$ ) different decision trees  $\Psi_1, \Psi_2, \dots, \Psi_K$  be used to solve the classification task,  $S_j$  –  $j$ -th subspace,  $R_i^k$  –  $i$ -th classification region of  $k$ -th classifier with label  $\omega = \Upsilon(R_i^k)$  ( $\omega \in \Omega$ ) and  $x$  – classified object.  $k$ -th decision tree is represented as  $R^k$ . Let us denote by  $M$  number of partitions of classification space into subspaces along one dimension. Then for  $n$ -dimensional problem  $M^n$  subspaces will be considered.

If we define  $\delta_S(S_j, x)$  as 1 if  $S_j$  spans  $x$  and 0 otherwise,  $\delta_R(R_i^k, S_j)$  as 1 if midpoint of  $S_j$  ( $mid(S_j)$ ) lies within  $R_i^k$  and 0 otherwise, i.e.

$$\delta_S(S_j, x) = \begin{cases} 1 & \text{if } x \in S_j \\ 0 & \text{if } x \notin S_j \end{cases}$$

$$\delta_R(R_i^k, S_j) = \begin{cases} 1 & \text{if } mid(S_j) \in R_i^k \\ 0 & \text{if } mid(S_j) \notin R_i^k \end{cases}$$

and  $f_\omega(R_i^k)$  as a weighting function that depends on region only. We can describe the proposed algorithm as:

$$\Psi_T(x) = \arg \max_{\omega} \sum_{k=1}^K \sum_{j=1}^{M^n} \sum_{i=1}^{|R^k|} \delta_S(S_j, x) \delta_R(R_i^k, S_j) f_\omega(R_i^k) \quad (2)$$

**Lemma.** (Weighted) Majority Voting is a special case of presented algorithm for infinitely dense space division.

*Proof.* First let us notice, that for any training point  $x$  there are only one  $S_x$  and  $R_s^k$  that span  $x$ .

Special case of the proposed algorithm is when division into subspaces becomes infinitely dense. This means, that the size of every subspaces becomes infinitely small and shrinks to single point:

$$\lim_{|S_x| \rightarrow 0} mid(S_x) = x \quad (3)$$

Combining equation 3 and 2 we can obtain:

$$\lim_{M \rightarrow \infty} \Psi_T(x) = \operatorname{argmax}_{\omega} \sum_{k=1}^K f_\omega(R_x^k), \quad (4)$$

where  $R_x$  denotes decision tree region, that spans  $x$ , which proves the lemma.  $\square$

In special case  $f_\omega(R_x^k) = I(\Psi_k(x), \omega)$ , where  $I$  is defined as in equation 1, majority voting is obtained.

## 4 Discussion

This paper proposes an algorithm for combining multiple decision tree models using static division of classification space into regions of competence and geometric representation of decision trees. Further research includes comparison between proposed algorithm and majority voting as well as other common integration techniques. Formal proof for convergence of this method to majority voting was presented. The method produces another decision tree of custom depth (dependent on division density), which makes it easy to reason about. This can have practical applications as an intermediate step of machine learning process, where cardinality reduction is needed. The complexity of integrated model can be fine-tuned by changing the division density. The only requirement for weighting function is to depend on classification model only, what provides a wide range of possible applications and optimizations. Aside of possible better quality of classification the requirement of odd number of models for majority voting can be omitted by using proper weighting.

### Acknowledgement

This work was supported in part by the National Science Centre, Poland under the grant no. 2017/25/B/ST6/01750.

## References

1. Robert Burduk. Integration base classifiers in geometry space by harmonic mean. In Leszek Rutkowski, Rafał Scherer, Marcin Korytkowski, Witold Pedrycz, Ryszard Tadeusiewicz, and Jacek M. Zurada, editors, *Artificial Intelligence and Soft Computing*, pages 585–592, Cham, 2018. Springer International Publishing.
2. Robert Burduk and Jędrzej Biedrzycki. Integration and selection of linear svm classifiers in geometric space. *Journal of Universal Computer Science*, 25(6):718–730, jun 2019.
3. Luca Didaci, Giorgio Giacinto, Fabio Roli, and Gian Luca Marcialis. A study on the performances of dynamic classifier selection based on local accuracy estimation. *Pattern Recognition*, 38(11):2188 – 2191, 2005.
4. E. Kim and J. Ko. Dynamic classifier integration method. In Nikunj C. Oza, Robi Polikar, Josef Kittler, and Fabio Roli, editors, *Multiple Classifier Systems*, pages 97–107, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.
5. Ludmila I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, New York, NY, USA, 2004.
6. Vladislav Polianskii and Florian T. Pokorny. Voronoi boundary classification: A high-dimensional geometric approach via weighted Monte Carlo integration. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5162–5170, Long Beach, California, USA, 09–15 Jun 2019. PMLR.
7. Oriol Pujol and David Masip. Geometry-based ensembles: Toward a structural characterization of the classification boundary. *IEEE transactions on pattern analysis and machine intelligence*, 31:1140–6, 07 2009.
8. Michal Wozniak, Manuel Graña, and Emilio Corchado. A survey of multiple classifier systems as hybrid systems. *Information Fusion*, 16:3–17, 03 2014.

# Text based meme classification

Piotr Bielak<sup>1</sup>, Michał Bieroński<sup>1</sup>, and Michał Jóźwiak<sup>1</sup>

Department of Computational Intelligence, Wrocław University of Science and Technology  
piotr.bielak@pwr.edu.pl

**Abstract.** Memes, especially those consisting of an image template with some text located on it, became a quite popular medium for sharing opinions and emotions. We propose a deep neural network model for prediction of a meme image template using the textual information only. We test multiple architectures and model hyperparameters to evaluate it against Polish and English newly constructed real-world datasets and report the promising results.

**Keywords:** memes · language features · template prediction · machine learning

## 1 Introduction

Nowadays, the Internet is full of so called **memes**. The definition depends mostly on the context, but in general memes are some humorous content, in various forms, which spreads in the Internet. Memes have become an influential medium to express people's opinions and emotions. The main idea is to convey a semantic information in a more amusing for the viewer way and easily share with other people. In this work, we focus on a particular type of memes – **image macros**, which are basically pictures with some text located on them. The pictures used for memes are usually not random – there exists a set of commonly used, popular image templates ("classic templates"). Usually, the creators of memes select the templates that should be relevant to described events, situations but also conveying specific emotions that the viewer can easily recognize just by looking on the image. It might be very useful to support the process of meme creation with automatic tools, e.g. when we don't know which template should be used for our text. In this paper we aim at creating a classification model, which would select an appropriate image template (from an existing finite set) for an inputted text (usually very short). We test it with different values of hyper-parameters, including the content of meme templates set and the model's architecture.

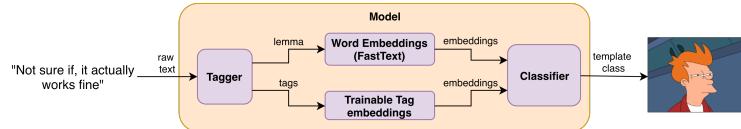
## 2 Related Work

To the best of our knowledge, there was no model proposed, which could predict the image template of a meme using text only. Nevertheless, there are currently three papers [1,4,3] published, that consider memes in the form of image macros. In [1], the authors tried to create a model for estimation of meme embeddings, that would incorporate different aspects of the meme itself. Given a set of meme templates they created an algorithm for template matching, which iteratively removes the captions and

other visual overlay elements from the picture and then tries to assign it to one of the templates (sparse matching). The authors evaluate those representations by meme clustering and show related clustering metrics. The paper [4] proposes a encoder-decoder model for automated captioning of memes. The process is opposite to our model. On the input the model takes a meme template image and a series of additional keywords. Those information are used to generate a caption (text) for the meme. The authors in [3] proposed a complex similarity-based approach for the task of *meme retrieval*. The proposed method introduces a multimodal similarity measure utilising both visual and textual information (and also some metadata like descriptive tags) to query the memes according to user-defined criteria.

### 3 Our solution

We propose a model for the prediction of meme image templates based on just the text. Let's denote a meme  $M$  as a sequence of words  $w_i$  and its true image template class  $c$ :  $M = (w_1, w_2, \dots, w_N, c)$ , where the total length  $N$  is rather small, usually  $N \leq 15$ . Our pipeline (Figure 1) starts with a tagging process - each word  $w_i$  is processed by an appropriate tagger (**WCRFT** [5] for Polish and **Spacy** [2] for English memes), producing 2-tuples  $(l_i, t_i)$ , where  $l_i$  is the **word lemma** and  $t_i$  is the **part-of-speech tag** for the given word. There are 33 distinct part-of-speech tags for Polish and 15 for English. Every word lemma is converted into an appropriate numerical representation, which can be further processed (word embedding vector). We evaluated both pretrained vectors (FastText taken from Clarin<sup>1</sup> for Polish) as well as trained by us from scratch on our meme corpus (FastText and Word2vec). For the part-of-speech tags we used a trainable embedding layer to encode them. Eventually, the POS tags embedding vectors are concatenated with the corresponding word embeddings, to make the final classifier input vectors. When it comes to the actual classifier models, we proposed and tested the following set: SVM, LSTM and Bidirectional LSTM. All of them make direct use of word embeddings. However they differ in the way the embeddings are prepared. SVM performs template classification based on a mean vector, obtained by averaging embedding vectors for every word in meme text. LSTM, on the other hand, simply accepts the sequence of word embeddings. Additionally, we considered dropping stop words from the meme text, which should serve denoising purposes.



**Fig. 1.** Block diagram of the proposed model

<sup>1</sup> Clarin main page

## 4 Experiments

We conducted experiments to see how hyperparameters influence the quality of our model, separately for each language (Polish and English). Every hyperparameter was changed independently, with the default values defined as follows:

- classifier: **Bidi LSTM with POS tags** (both languages),
- number of hidden units in LSTM: **128** (both languages),
- word embeddings: Clarin (Polish) and trained (English) FastText 100-dim,
- stop words: removed (Polish) and not removed (English).

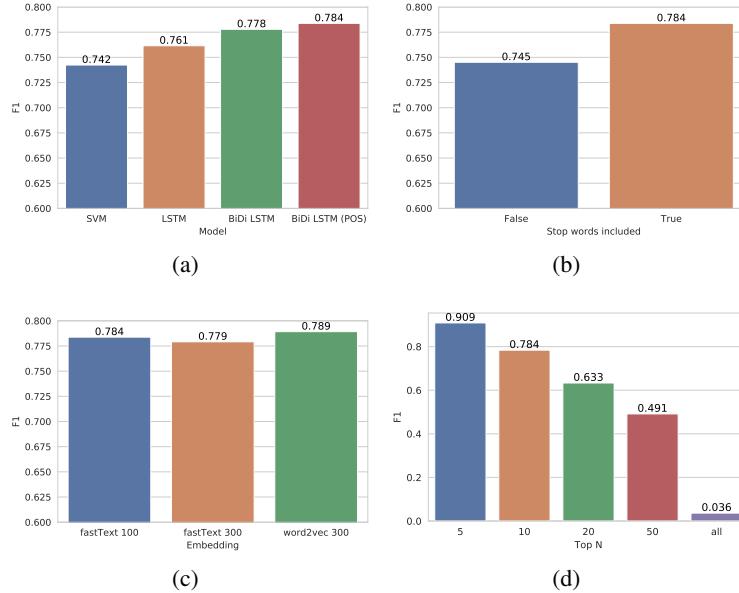
We show the results of best hyperparameters search for English only (Section 4.1). Table 1 depicts the classification results for the best hyperparameter values found (for both languages). The datasets used in the experiments were obtained from two big meme web portals, separately for Polish memes (`fabrykamemow.pl`; about 60 000 instances) and English memes (`memegenerator.net`; approximately 700 000 instances). In both cases only top  $N$  most popular image templates (classes) were modelled which did not reduce the size of data.

### 4.1 Experimental results for English

Figure 2(a) shows superiority of LSTM based methods over the baseline method – SVM classifier with mean word embedding vector. This indicates that word ordering carries important information. The bidirectional version of the LSTM model tends to improve the score. What is more, our intuition that parts-of-speech may be the useful has been confirmed. Inclusion of these into our recurrent model allowed to increase the classification quality. At first, while building our processing pipeline we decided to remove stop words for denoising purposes. However, further analysis shown that meme text after this step was sometimes shortened to barely two words. This indicated that including stop words might perform in better quality what was confirmed in experiments and is shown in Figure 2(b). Another experiment shown that all used word embeddings resulted in similar prediction quality what can be observed on Figure 2(c). For English language model, one more experiment has been performed to see how many classes (meme templates) can be predicted with satisfactory quality (Figure 2(d)). Higher number of classes, as expected, resulted in decreased score compared to the  $N = 5$  scenario. However, for  $N = 20$ , prediction quality was still satisfactory enough.

**Table 1.** Results summary for best models

| Lang | Hyperparameters   | Accuracy | Precision | Recall | F1    |
|------|---|----------|-----------|--------|-------|
| PL   | classifier: <b>Bidirectional LSTM with POS tags</b><br>number of hidden units in LSTM: <b>128</b><br>word embeddings: <b>Clarin FastText 300-dim</b><br>stop words: <b>removed</b>      | 0.827    | 0.812     | 0.770  | 0.783 |
| EN   | classifier: <b>Bidirectional LSTM with POS tags</b><br>number of hidden units in LSTM: <b>128</b><br>word embeddings: <b>trained FastText 100-dim</b><br>stop words: <b>not removed</b> | 0.921    | 0.906     | 0.913  | 0.909 |



**Fig. 2.** Experimental results for English memes. F1 scores for different: (a) prediction models, (b) influence of stop words inclusion, (c) word embeddings, (d) no. of  $N$  most popular templates.

## 5 Conclusions

We created a model for template prediction of memes based only the meme text. We acquired data from one of the biggest Polish memes websites and from a large database of English ones, and tested our model against it. Multiple model architectures and hyperparameters were evaluated and the results aggregated. Although the meme texts are really short and the proposed model is quite simple, it performs very well.

## References

1. Dubey, A., Moro, E., Cebrian, M., Rahwan, I.: Memesequencer: Sparse matching for embedding image macros. In: Proceedings of the 2018 World Wide Web Conference. pp. 1225–1235 (2018)
2. Honnibal, M., Montani, I.: spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing (2017), to appear
3. Milo, T., Somech, A., Youngmann, B.: Simmeme: A search engine for internet memes. In: 35th IEEE International Conference on Data Engineering, ICDE 2019, Macao, China, April 8–11, 2019. pp. 974–985 (2019)
4. Peirson, V., Abel, L., Tolunay, E.M.: Dank learning: Generating memes using deep neural networks. arXiv preprint arXiv:1806.04510 (2018)
5. Radziszewski, A.: A tiered CRF tagger for Polish. In: Intelligent Tools for Building a Scientific Information Platform: Advanced Architectures and Solutions, p. to appear. Springer Verlag (2013)

# Real-Time Polish Traffic Sign Recognition

Kacper Kania<sup>[0000–0003–4177–1349]</sup> and Michał Kosturek<sup>[0000–0002–2918–0688]</sup>

Wrocław University of Science and Technology,  
27 Wybrzeże Wyspiańskiego, Poland  
[kacp.kania@gmail.com](mailto:kacp.kania@gmail.com)  
[kosturek.michal@gmail.com](mailto:kosturek.michal@gmail.com)

**Abstract.** Automatic recognition of objects on images captured by car-mounted cameras is one of the main areas of interest in the automotive industry. Traffic sign recognition is one of such tasks and it is addressed in this paper. This work covers our solution for the Polish road signs recognition using two-stage recognition pipeline built with state-of-the-art deep learning methods. We also present used data and show its processing steps, which allowed for creating the first Polish road signs dataset. Conducted experiments and the final evaluation of our solution show that the pipeline can be installed on a mountable vehicle device for an accurate, fast traffic sign recognition.<sup>1</sup>

**Keywords:** Traffic sign recognition · Deep learning · Computer vision

## 1 Introduction

In the automotive industry there is a demand for image analysis algorithms designed for car-mounted cameras. This includes inter alia lane, pedestrian or obstacle detection and traffic sign recognition. The latter is the main subject of this study. The task was divided into two subproblems: sign detection and classification to a subset of Polish signs, specified in section 2. However, there are no publicly available datasets containing images of Polish signs. Thus, we collected databases of images of traffic signs from other countries, integrated them and sanitised to create a dataset of signs that match the appearance of Polish signs. We further investigate a two-stage processing pipeline for automatic detection and classification of these signs that works in real-time. We conclude that our approach allows for high performance traffic signs recognition that can be used in embedded systems. Our main contributions are: – An efficient pipeline for detection and classification of Polish traffic signs – An integration of publicly available datasets of foreign traffic signs that is suitable for recognition of polish traffic signs. To our knowledge, this is the first dataset of this kind. The result dataset contains signs that are the most crucial for road safety<sup>2</sup>

---

<sup>1</sup> The code is publicly available at:

<https://github.com/vanitas-vanitatum/traffic-sign-recognition>

<sup>2</sup> This includes signs with symbols: A-1, A-2, A-7, A-11a, A-14, A-15, A-15, A-17, A-18b, A-29, A-30, B-1, B-2, B-20, B-25, B-33, B-34, B-35, B-36, C-1, C-3, C-5, C-6, C-7, C-9, C-10, C-12, D-1, D-2, D-3, D-4a, D-6.

## 2 Datasets

As mentioned in section 1, there is no publicly available dataset for Polish traffic signs classification or detection. Nevertheless there are some well-known collections of labelled traffic signs images, both for classification and detection, from different countries. We argue, that for the detection task it is not crucial to use a dataset representing signs from the country considered. Therefore for the detector we used a combination of three publicly available datasets: LISA [5] - American, TsignDet<sup>3</sup> and Tsinghua [12] - Chinese. Overall, the dataset contained 25,000 images with 50,000 labelled signs.

In order to be able to train a classifier suitable for Polish road signs, we prepared a processing pipeline, which allowed us to utilise open datasets originating from different countries. We used the following datasets: GTSRB [1] - German, BelgiumTSC [9] - Belgian and TsignDet<sup>4</sup> - Chinese. All images were converted to greyscale so the final pipeline infers its decision using solely shape and brightness information. We manually mapped all classes from these datasets to Polish signs types. We also introduced mapping with transformations, which allowed us to address classes imbalance and to extend the dataset. This mapping is based on simple observations, such as that, for example, the sign ‘turn left before sign’ (C-3), when flipped left-to-right can be used as a ‘turn right before sign’ (C-1). We also introduced an artificial ‘not-a-sign’ class. Images for this class were generated by sampling fragments of images from the detection dataset. We used pixel value variance calculated over  $48 \times 48$  windows as a probability distribution of sampling a fragment at a specific location. We ensured that no fragment covering a sign was extracted. Ultimately, the dataset contained 130,000 images from 32 classes, including the ‘not-a-sign’ class.

## 3 Proposed method

In this work, we propose a processing pipeline for automatic Polish traffic sign recognition. The method consists of two main processing steps. Firstly, an input image is converted to greyscale to ensure that next steps base their decision merely on shape and brightness features. It is then passed to the sign detector. It returns coordinates of all found traffic signs in the image. Coordinates are used to extract boxes. The boxes are served as an input to the classification network. The network returns whether an introduced part of the input image is a sign or not and one of 31 classes<sup>2</sup>. Since the classifier is an output of the softmax function, we take the highest value of that output.

We do not restrict the use of any particular detection and classification method. Our final pipeline consisted of two neural networks selected empirically following results of experiments in section 4. The detector was built upon

---

<sup>3</sup> <http://www.nlpr.ia.ac.cn/pal/trafficdata/detection.html>

<sup>4</sup> <http://www.nlpr.ia.ac.cn/pal/trafficdata/recognition.html>

Single-Shot Detector architecture [3]. It was further optimised by reducing convolutional kernel sizes and replacing standard convolution with depthwise convolution. We replaced *Online Hard-Example Mining* strategy used during learning of SSD and applied a novel focal loss [2] instead.

The classifier follows ShuffleNetV2 architecture guidelines introduced in [4]. The network outputs 32 values transformed by softmax function.

## 4 Experiments

### 4.1 Metrics

Accuracy of the detectors was measured using recall and precision in 5-way fashion. We manually gathered consecutive frames from LISA dataset to form groups of images and sampled random images from Tsinghua dataset. We augment all images using random scaling and translation so each situation has at least five images. Finally, we use detector for a group of images and average results on images in a single group. The same procedure is used for classifiers. We average softmax outputs of each classification and take an index of the highest activation. To measure performance of classifiers, we use a standard classification accuracy. We also measured the number of frames per second that our model can process.

### 4.2 Models

We compared four deep learning detectors: YOLOv2 [6], YOLOv3 [7], EAST [11] and SSDLite [3]. The detectors were trained using Chinese detection datasets. Each model was modified to predict only rectangular detection, thus reducing total number of parameters to learn and facilitating learning convergence. The result are shown in the table 1. Using both recall and precision, the SSDLite model works comparably well to YOLOv2 and YOLOv3 but performs much faster, making it usable for the real-time system.

To select the best classifier, we chose three models that demonstrate high classification accuracy, while maintaining real-time processing speed: Wide Residual Network [10], MobileNetV2 [8] and ShuffleNetV2 [4]. Models were trained using the dataset, introduced in the section 2. Obtained results are shown in table 2. We distinguished classification to one of 32 classes, and whether an image contains a sign or not. We selected ShuffleNetV2 for the final pipeline for its accuracy and fast processing speed. Each model could easily distinguish whether an image represent a sign or not, thus improving general detection accuracy.

All time measurements were taken on a machine with nVidia GTX 740m, Intel® Core™ i7-4702MQ processor and 8 GB RAM to show that the pipeline can run in real-time on a budget device.

## 5 Conclusions

In this work, we introduced the first Polish traffic sign dataset formed from publicly available data. We presented a traffic sign recognition pipeline based on

Table 1: Recall, precision and speed performance of detectors.

| Model       | Recall      | Precision   | FPS          |
|-------------|-------------|-------------|--------------|
| YoloV2 [6]  | <b>0.83</b> | 0.99        | 1.19         |
| YoloV3 [7]  | 0.57        | <b>1.00</b> | 1.37         |
| EAST [11]   | 0.62        | 0.88        | 2.06         |
| SSDLite [3] | 0.64        | 0.99        | <b>11.36</b> |

Table 2: Accuracy and speed performance of classifiers.

| Model            | 32 classes  | binary      | FPS          |
|------------------|-------------|-------------|--------------|
| MobileNetV2 [8]  | 0.84        | 0.95        | 52.77        |
| WRN-16-4 [10]    | 0.95        | <b>0.99</b> | <b>96.68</b> |
| ShuffleNetV2 [4] | <b>0.97</b> | <b>0.99</b> | 85.57        |

deep learning, that achieved high recognition accuracy while maintaining real-time processing speed. Obtained results showed that the pipeline can provide accurate information about currently operative rules while driving. Future research will focus on a comparison of the presented models using statistical tests.

## References

- Houben, S., Stallkamp, J., Salmen, J., Schlippling, M., Igel, C.: Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark. In: International Joint Conference on Neural Networks. No. 1288 (2013)
- Lin, T., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. CoRR **abs/1708.02002** (2017), <http://arxiv.org/abs/1708.02002>
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C., Berg, A.C.: SSD: single shot multibox detector. CoRR **abs/1512.02325** (2015), <http://arxiv.org/abs/1512.02325>
- Ma, N., Zhang, X., Zheng, H., Sun, J.: Shufflenet V2: practical guidelines for efficient CNN architecture design. CoRR **abs/1807.11164** (2018), <http://arxiv.org/abs/1807.11164>
- Mogelmose, A., Trivedi, M.M., Moeslund, T.B.: Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey. IEEE Transactions on Intelligent Transportation Systems **13**(4), 1484–1497 (Dec 2012). <https://doi.org/10.1109/TITS.2012.2209421>
- Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. CoRR **abs/1612.08242** (2016), <http://arxiv.org/abs/1612.08242>
- Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. CoRR **abs/1804.02767** (2018), <http://arxiv.org/abs/1804.02767>
- Sandler, M., Howard, A.G., Zhu, M., Zhmoginov, A., Chen, L.: Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. CoRR **abs/1801.04381** (2018), <http://arxiv.org/abs/1801.04381>
- Timofte, R., Zimmermann, K., van Gool, L.: Multi-view traffic sign detection, recognition, and 3d localisation. In: Ninth IEEE Computer Society Workshop on Application of Computer Vision. pp. 1–8. Snowbird, Utah, USA (December 2009)
- Zagoruyko, S., Komodakis, N.: Wide residual networks. CoRR **abs/1605.07146** (2016), <http://arxiv.org/abs/1605.07146>
- Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., Liang, J.: EAST: an efficient and accurate scene text detector. CoRR **abs/1704.03155** (2017), <http://arxiv.org/abs/1704.03155>
- Zhu, Z., Liang, D., Zhang, S., Huang, X., Li, B., Hu, S.: Traffic-sign detection and classification in the wild. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2110–2118 (June 2016). <https://doi.org/10.1109/CVPR.2016.232>

# Mixed-curvature Embedding of Human Diseases Network

Maciej Falkiewicz[0000–0002–3851–1995]

Wroclaw University of Science and Technology, Wroclaw, Poland  
[maciej.falkiewicz@pwr.edu.pl](mailto:maciej.falkiewicz@pwr.edu.pl)

**Abstract.** Improvement of the results and reduction of computational costs have recently been considered as the main challenges of machine learning for relational data. It has been previously shown that it can be achieved by reducing the dimensionality of problems. One of the most exquisite directions of this domain is mixed-curvature embedding that incorporates non-Euclidean spaces and benefits from their vast capacity. We examine this new approach on a real-world data-set of human diseases where node representations are evaluated by means of graph reconstruction and link prediction tasks. The results show clear benefits from the use of hyperbolic space.

**Keywords:** Graph Embedding · Mixed-curvature Product Manifolds · Representation Learning.

## 1 Introduction

Among numerous approaches incorporating advances in machine learning to the field of graph embedding in recent years, there is one direction that has attracted particular interest, namely using non-Euclidean vector spaces. Most of the effort was put to exploit hyperbolic space resulting in matured methods with good quality of embedding. However, they perform best with specific structures, that is hierarchies, and are practically unable to encapsulate cyclical high-order relation. The very recent paper by Albert Gu et al. [3] addresses this issue. The authors propose a simple, yet brilliant idea to benefit from all 3 model spaces of Riemannian geometry in the form of a mixed-curvature product manifold. In this paper we report preliminary results on Mixed-Curvature Embedding [3] using human diseases [1] network and evaluate it on the task of *graph reconstruction* and *link prediction*.

## 2 Related work

The research in the direction of graph embedding has gained most of the community attention around the year 2016 with the famous node2vec [2] paper. The taxonomy of methods identifies the following graph representation techniques[8]: *Matrix factorization-based embedding*, *Deep Learning graph embedding* and *Random walkbased Skip-gram model* [5].

### 3 Mixed-curvature graph embedding

It has been identified that for particular graphs, it is worthwhile to make use of the remaining model spaces of Riemannian geometry of constant curvature [6], that is *hyperbolic* and *spherical space*.

|             | Euclidean space                         | spherical space  | hyperbolic space<br>(hyperboloid model)  |
|-------------|---|--|--|
| Curvature   | 0                                       | $K$  | $-K$   |
| Definition  | $\mathbb{E}^n = \{x \in \mathbb{R}^n\}$ | $\mathbb{S}_K^n = \{x \in \mathbb{R}^{n+1} : \ x\  = K^{-1/2}\}$ | $\mathbb{H}_K^n = \{x \in \mathbb{R}^{n+1} : \langle x, x \rangle_{\mathbb{H}} = -K^{-1/2} \wedge x_0 > 0\}$ |
| Dot product | $\langle x, y \rangle = x \cdot y$      | $\langle x, y \rangle_{\mathbb{S}} = x \cdot y$                  | $\langle x, y \rangle_{\mathbb{H}} = -x_0 y_0 + \sum_{i=1}^n x_i y_i$  |
| Distance    | $d_{\mathbb{E}}(x, y) = \ x - y\ $      | $d_{\mathbb{S}}(x, y) = \arccos(\langle x, y \rangle)$           | $d_{\mathbb{H}}(x, y) = \text{arccosh}(-\langle x, y \rangle_{\mathbb{H}})$                                  |

Table 1: Comparison of model spaces of Riemannian geometry of constant curvature [4].

In order to benefit from all of the model space, the so-called *product manifolds* are used [4]. The embedding space is constructed then as:

$$\mathcal{P} = \mathbb{S}_{s^1}^{s_1} \times \dots \times \mathbb{S}_{s^k}^{s_k} \times \mathbb{H}_{h^1}^{h_1} \times \dots \times \mathbb{H}_{h^l}^{h_l} \times \mathbb{E}^e, \quad (1)$$

where  $s_i$  and  $s^i$  stand respectively for the dimension and curvature of each of the  $k$  spherical components, similarly for  $l$  hyperbolic components. The distance between two vectors  $p, q \in \mathcal{P}$  is simply

$$d_{\mathcal{P}}(x, y) = \sum_{i=1}^{k+l+1} d_{\mathcal{P}_i}(x, y). \quad (2)$$

### 4 Experiments

Two evaluation approaches are proposed. *Graph reconstruction* originates from the classical studies on embedding for metric spaces. Average *Distortion* ( $D_{avg}$ ) is defined as:

$$D_{avg}(\phi) = \frac{1}{\binom{|V|}{2}} \sum_{u, v \in V: u \neq v} \frac{|d_{\mathcal{P}}(\phi(u), \phi(v)) - d_G(u, v)|}{d_G(u, v)}, \quad (3)$$

where  $d_{\mathcal{P}}(\cdot, \cdot)$  and  $d_G(\cdot, \cdot)$  are the distances in the embedding space and original graph  $G = (V, E)$  respectively.  $\phi(u)$  denotes the vector representation of node  $u$ .

*Mean Average Precision (MAP)* is defined as:

$$\text{MAP}(\phi) = \frac{1}{|V|} \sum_{u \in V} \frac{1}{\deg(u)} \sum_{v \in N(u)} \frac{|N(u) \cap R_u(v)|}{|R_u(v)|}, \quad (4)$$

where  $\deg(u)$  is the degree of node  $u$ ,  $N(u)$  is the set of  $u$ 's neighbors and  $R_u(v)$  is the smallest set of vertices surrounding representation of  $u$  that contains  $v$ , formally:

$$d_P(\phi(u), \phi(w)) \leq d_P(\phi(u), \phi(v)) \iff w \in R_u(v). \quad (5)$$

Another essential evaluation is performed by means of *link (edge) prediction* task. As proposed in [2] logistic regression on top of four binary operators – Average, Hadamard, Weighted-L1, Weighted-L2.

The used data-set of human diseases [1] consists out of 516 vertices connected by 1188 edges, the density of 0.00894107, and exhibits 4080 triangles.

The Mixed-curvature embedding is learned with a matrix factorization objective. The loss function takes the following form:

$$\mathcal{L}(\phi) = \sum_{u \in V} \sum_{v \in V : v \neq u} \left| \left( \frac{d_P(\phi(u), \phi(v))}{d_G(u, v)} \right)^2 - 1 \right|. \quad (6)$$

The space structure implies the use of Riemannian Stochastic Gradient Descent [7].

In Mixed-curvature embedding, an arbitrary number of hyperbolic and spherical components can be applied, each having a different or same size. The number of possible combinations is immense. We set a restriction that each geometry is represented by a single factor, and the dimensions come from the set  $\{0, 2, 5, 10\}$ . The summarized results for graph reconstruction can be seen in Fig. 1.

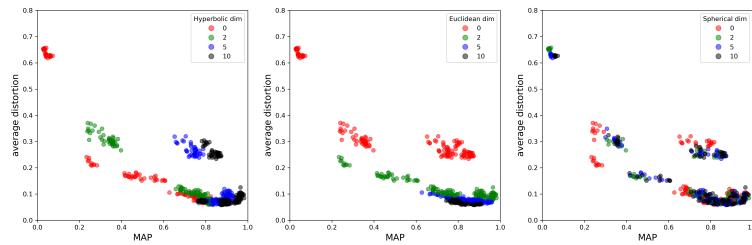


Fig. 1: Graph reconstruction results (MAP vs average distortion).

In fig. 2 it can be seen that even simple product manifolds perform surprisingly well. The Euclidean component is decisive for the quality of the representation. However, hyperbolic one can enhance results. Spherical geometry fails to capture the graph topology of human diseases network.

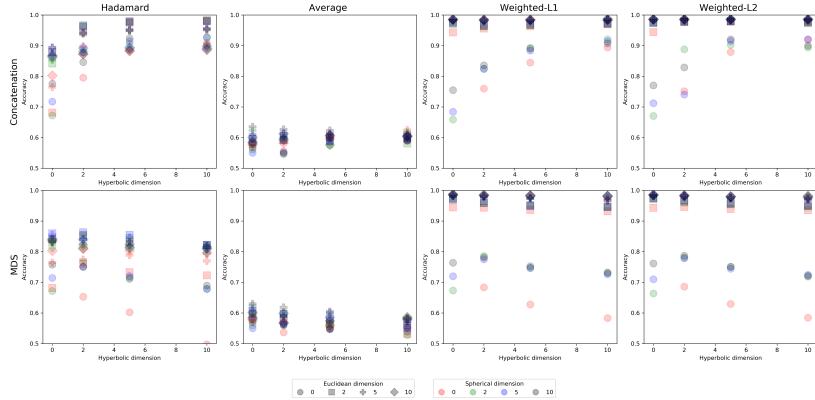


Fig. 2: Link prediction results (accuracy for hyperbolic component vs spherical-Euclidean signature).

## 5 Conclusions

A mixed-curvature Riemannian product manifold embedding was examined in the paper. Graph reconstruction and link prediction results within the disease network show promising direction of research in compound embedding.

## References

1. Goh, K.I., Cusick, M.E., Valle, D., Childs, B., Vidal, M., Barabási, A.L.: The human disease network. *Proceedings of the National Academy of Sciences* **104**(21), 8685–8690 (2007)
2. Grover, A., Leskovec, J.: node2vec: Scalable Feature Learning for Networks arXiv:1607.00653 (Jul 2016)
3. Gu, A., Sala, F., Gunel, B., Ré, C.: Learning mixed-curvature representations in product spaces. In: ICLR (2019)
4. Lee, J.M.: Introduction to smooth manifolds, Graduate Texts in Mathematics, vol. 218. Springer-Verlag, New York (2003)
5. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed Representations of Words and Phrases and their Compositionality arXiv:1310.4546 (Oct 2013)
6. Sarkar, R.: Low distortion Delaunay embedding of trees in hyperbolic plane. In: Graph drawing, Lecture Notes in Comput. Sci., vol. 7034, pp. 355–366. Springer, Heidelberg (2012)
7. Wilson, B., Leimeister, M.: Gradient descent in hyperbolic space pp. 1–10 (2018)
8. Zhang, D., Yin, J., Zhu, X., Zhang, C.: Network Representation Learning: A Survey arXiv:1801.05852 (Dec 2017)

# Identification of Players Ranking in E-Sport: CS:GO Study Case

Karol Urbaniak<sup>1</sup>[0000–0002–5373–9760]

<sup>1</sup> Department of Artificial Intelligence method and Applied Mathematics  
in the Faculty of Computer Science and Information Technology,  
West Pomeranian University of Technology, Szczecin 71-210, Poland;  
(e-mail: karol-urbaniak@zut.edu.pl)

**Abstract.** Nowadays, more and more people from all over the world are keen on growing fascination with e-sport. In practice, e-sport is a type of sport in which players compete using computer games. The competitions in such games like FIFA, Dota2, the League of Legends and Counter-Strike are prestigious tournaments with a global reach and a budget of millions of dollars. The reliable player ranking is a critical issue in both classic and e-sport. For example, the "Golden Ball" is the most valuable prize for an individual football player in the whole football history. What's more, the entire players' world wants to know who the best player is. The position of each player in the ranking depends on the assessment of his skills and predispositions. In this paper, we make studies on identification of players ranking obtained using the COMET method on the example of the popular game Counter-Strike: Global Offensive.

**Keywords:** E-sport · Ranking · COMET method.

## 1 Introduction

E-sport is a form of sport in which the players' activities focus on computer games. Competition between players takes place both in the form of a recreation and professional field called "pro gaming." Professional players usually belong to different e-sport organizations and represent their teams competing in omnifarious tournaments, events, and international championship. In recent years, e-sport has become one of the fastest-growing forms of new media driven by the growing origins of games broadcasting technologies [1]. In 2019, 453.8 million people will watch e-sport worldwide, which means an increase of about 15% compared to 2018. It will consist of 201 million regular and 252 million occasional viewers. In the current economic situation, global revenues from e-sport may reach USD 1.8 billion by 2022, and optimistic USD 3.2 billion. Hamari in [1] claims that with the development of e-sport, classic sport is becoming more

and more a computer-based form of media and information technology. Therefore, e-sport is a very interesting subject of research in the field of information technology.

The accurate player ranking is a crucial issue in both classic and e-sport. Each player's position in the ranking is strictly determined by their abilities, predispositions, and talent in the field of represented discipline. In this paper, we identify the model to generate a ranking of players in the popular e-sport game, i.e., Counter-Strike: Global Offensive (CS: GO), using the Characteristic Objects METhod (COMET). The obtained ranking will be compared to Rating 2.0, which is the most popular for CS: GO game. Rating 2.0 was proposed by Half-Life Television (HLTV). It is a news website which covers professional CS: GO news, tournaments, statistics, and rankings. The identified model creates a ranking, which is more natural to interpret. Each player assessment has three additional parameters.

The concept of this work is preliminary and is intended to show only partial results of the study. This study case facilitates application of the COMET in the new field of application. The COMET is a novel method of identifying a multi-criteria expert decision model to solve decision problems based on a rule set, using elements of the theory of fuzzy sets [3,4]. This technique is a modern approach, completely free of the rank reversal paradox. The identified model considers multiple criteria and is self-contained of chosen alternatives in the criteria domain. It was most commonly used in the field of sustainable transport [5,6].

The rest of the paper is organized as follows: Section II introduces the results of the study and the discussion about the differences in both rankings. In section III, we present the summary and conclusions.

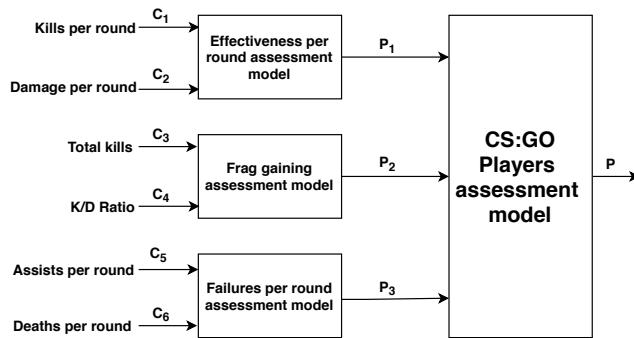
## 2 Results and discussions

Many parameters influence the player's performance, including the assessment of his skills and predispositions. For instance, with player's age, the drop-off in reaction time makes it hard for them to compete and harder to aim the head of moving target. Hight percentage of headshots reflects the shooting skills and is a kind of prestige. Therefore, the following six criteria have been selected (Based on the information from the HLTV website):

- $C_1$  - Kills per round, the average number of kills scored by the player during one round;
- $C_2$  - Damage per round, average damage inflicted by a player during one round;
- $C_3$  - Total kills, the total number of kills gained by the player;
- $C_4$  - K/D Ratio, the number of kills divided by number of deaths;
- $C_5$  - Assists per round, the average number of assists gained by the player during one round;
- $C_6$  - Deaths per round, the average number of deaths of a player during one round;

Especially important are the  $C_1$  and  $C_4$  criteria. They inform us that the chance to eliminate the player is smaller than the possibility that he will kill the enemies. In this study case, the considered problem is simplified to a structure which is presented in Figure 1. In that way, we have to identify three related models, where each one requires a lot smaller number of queries to the expert. The final decision model consists of three following models, where for each one nine characteristic objects and 36 pairwise comparisons are needed:

- $P_1$  - Effectiveness per round assessment model with two inputs;
- $P_2$  - Frag gaining assessment model with two inputs;
- $P_3$  - Failures per round assessment model with two inputs;



**Fig. 1.** The hierarchical structure of the players ranking assessment problem.

The sample data for the top 10 players is shown in Table 1. The final decision assessment model identified '*Simple*' as the best player at all when the worst rating was given to '*Hatz*'. Analyzing the results of the three related models, we can conclude that the highest score in the first model ( $P_1$ ) was obtained by '*Simple*' again, and is equal to 0.8825. In the ( $P_2$ ) and  $P_3$  models, the best outcome was acquired by '*Jame*' with the value 0.8423 as  $P_2$  and 0.7750 as  $P_3$ . The interesting fact is, that '*ZywOo*', who placed the second position, even if he hadn't the best score in any of the three models, was still better than Jame. '*ZywOo*' received much better result in the first model and had comparable score to '*Jame*' in the second model. Furthermore, '*huNter*' with the fourth result was close to beat '*Jame*' and take over his position. In comparison with '*Jame*', '*huNter*' had much higher assessment in  $P_1$ , getting average results at the rest of the models. It follows from this that the most critical models are  $P_1$  and  $P_2$ .

$\rho$  Spearman's rank correlation coefficient between the  $P_1$ ,  $P_2$ , and  $P_3$  model and reference ranking is equal respectively to 0.7818, 0.7091, and 0.0061. The correlation between the first two models is moderately strong, and in the case of the third model, there is no correlation. However,  $\rho$  Spearman's coefficient between the final model and reference ranking is equal to 0.9636, which means that both rankings are strongly correlated, and the proposed structure of the assessment model well defines the investigated relationships.

**Table 1.** The performance table of the alternatives and selected criteria.

| Pos. | Name   | $C_1$ | $C_2$ | $C_3$ | $C_4$ | $C_5$ | $C_6$ | $P_1$  | $P_2$  | $P_3$  | $P$    |
|------|--------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|
| 1    | s1mple | 0.88  | 86.6  | 0.168 | 1.50  | 0.09  | 0.59  | 0.8825 | 0.6849 | 0.5125 | 0.7437 |
| 2    | ZywOo  | 0.83  | 85.3  | 1.000 | 1.40  | 0.12  | 0.59  | 0.6788 | 0.7857 | 0.5625 | 0.7414 |
| 3    | Jame   | 0.78  | 79.3  | 0.755 | 1.51  | 0.09  | 0.52  | 0.4163 | 0.8423 | 0.7750 | 0.6432 |
| 4    | Jamppi | 0.83  | 83.1  | 0.507 | 1.30  | 0.10  | 0.64  | 0.6513 | 0.5553 | 0.3500 | 0.5941 |
| 5    | huNter | 0.80  | 88.2  | 0.981 | 1.22  | 0.15  | 0.66  | 0.6025 | 0.5753 | 0.3375 | 0.5959 |
| 6    | vsm    | 0.80  | 86.6  | 0.343 | 1.22  | 0.13  | 0.65  | 0.5825 | 0.4158 | 0.3500 | 0.4556 |
| 7    | meyern | 0.82  | 83.8  | 0.080 | 1.28  | 0.12  | 0.64  | 0.6225 | 0.4271 | 0.3750 | 0.4732 |
| 8    | Kaze   | 0.78  | 80.7  | 0.089 | 1.32  | 0.10  | 0.60  | 0.4338 | 0.4723 | 0.5000 | 0.4129 |
| 9    | Hatz   | 0.76  | 81.8  | 0.190 | 1.28  | 0.15  | 0.60  | 0.3725 | 0.4546 | 0.5625 | 0.3617 |
| 10   | Sico   | 0.76  | 78.4  | 0.137 | 1.36  | 0.13  | 0.56  | 0.3300 | 0.5271 | 0.6875 | 0.3759 |

### 3 Conclusions

The main contribution of the paper is a proposal of the CS: GO players assessment model with three related evaluation sub-models. For verification purposes, the obtained decision model was compared to the existing ranking created by HLTV called Rating 2.0. The results of incorrectly classified players are quite close to each other. Rating 2.0 provides ranking data only to two decimal places, and with an equalized skill level of each player, there is an uncertainty of the results. Perhaps the uncertain data processing will be a good form of solving this problem using interval arithmetic or fuzzy logic. The future work directions should concentrate on the improvement of effectiveness and further empirical investigation for CS: GO, but also in other e-sport games.

**Acknowledgments:** The work was supported by the National Science Centre, Decision No. 2018/29/B/HS4/02725

### References

1. Hamari, J., Sjöblom, M. (2017). What is eSports and why do people watch it?. *Internet research*, 27(2), 211-232.
2. Faizi, S., Rashid, T., Sałabun, W., Zafar, S., Watróbski, J. (2018). Decision making with uncertainty using hesitant fuzzy sets. *International Journal of Fuzzy Systems*, 20(1), 93-103.
3. Sałabun, W. (2015). The Characteristic Objects Method: A New Distance-based Approach to Multicriteria Decision-making Problems. *Journal of Multi-Criteria Decision Analysis*, 22(1-2), 37-50.
4. Sałabun, W., Piegat, A. (2017). Comparative analysis of MCDM methods for the assessment of mortality in patients with acute coronary syndrome. *Artificial Intelligence Review*, 48(4), 557-571.
5. Sałabun W., Palczewski, K., Watróbski, J. (2019). Multicriteria Approach to Sustainable Transport Evaluation under Incomplete Knowledge: Electric Bikes Case Study. *Sustainability*, 11(12), 3314.
6. Sałabun W., Karczmarczyk, A. (2018). Using the comet method in the sustainable city transport problem: an empirical study of the electric powered cars. *Procedia computer science*, 126, 2248-2260.

# Deep learning in EEG: Detection of error-related negativity in Eriksen flanker task

Krzysztof Kotowski<sup>1</sup> and Katarzyna Stapor<sup>1</sup>

<sup>1</sup> Institute of Informatics, Silesian University of Technology, Gliwice, Poland

**Abstract.** The complexity and high dimensionality of electroencephalographic (EEG) brain signal make it difficult to approach standard machine learning techniques. Deep learning methods, especially artificial neural networks inspired by the structure of the brain itself seem to be a better approach. In this paper, the simple EEGNet architecture is shown to perform at least as good as the state-of-the-art traditional machine learning methods in the detection of erroneous responses based on EEG recordings in the famous Eriksen flanker task.

**Keywords:** Deep learning, EEG, Event-related potentials, Error-related negativity

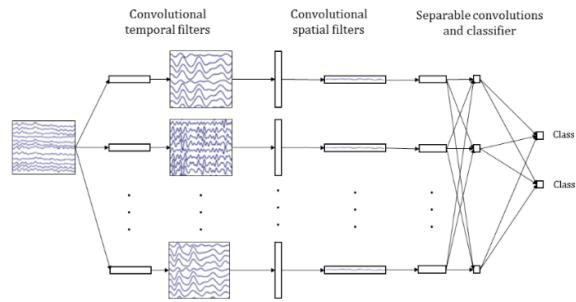
## 1 Introduction

Neurons in the brain are a source of the electric activity that is the biological basis of cognitive processes. Discovering how this activity transforms into our thoughts is the main goal of nowadays neuroscience. The advantages in this domain may help treat psychological and neurological disorders, boost our cognitive skills, and speed up communication between people and computers. However, the complexity of the human brain makes it extremely hard to decode. Thousands of tedious experiments supported by the technological development of measuring devices and computer science provided only basic principles of neuroscience. The deep artificial neural networks (inspired by the brain itself) seem to be the next significant step towards exploring massive amounts of data produced each second by our biological neural networks.

### 1.1 Deep Learning in EEG

Deep learning is inherently connected with artificial neural networks. In the past few years, the different types of architectures have abounded, especially in the domain of computer vision. Traditional fully-connected artificial neural networks, Restricted Boltzmann Machines (RBMs) and Deep Belief Networks (DBNs) have been replaced with more specialized models and layers. The most popular architecture in the EEG domain is, similarly as in the computer vision domain, the convolutional neural network (CNN). It is used in more than 40% of articles about deep learning for EEG and is the most frequently adopted architecture [1][2]. The design of CNN allows encoding spatial and temporal relationships among the EEG data in the form of a set of trainable convolutional filters stacked in multiple layers of the network. Only the last two or three layers constitute a traditional fully-connected (dense) classifier. Addi-

tionally, multiple supporting layers have been designed to prevent overfitting, speed up the training and decrease the number of parameters. The most popular ones are pooling, dropout and batch normalization layers. All of them have been used in state-of-the-art deep architectures dedicated to EEG classification, namely EEGNet [3] presented in Fig. 1Error: Reference source not found, and DeepConvNet from [4]. As reported by their authors, the use of Dropout and BatchNormalization significantly increased accuracy. Interestingly, an additional significant increase in accuracy in both models was achieved by replacing the very common (70% of the articles in EEG domain [1]) Rectified Linear Units (ReLU) activations with Exponential Linear Units (ELU).



**Fig. 1.** Overall visualization of the EEGNet architecture (figure inspired by [3]).

## 2 Methods

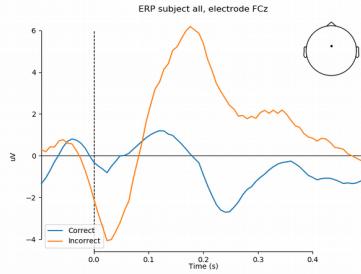
### 2.1 Eriksen Flanker Task

In our experiment, we were interested in classifying errors and correct responses in a modified Eriksen flanker task [5]. This cognitive task is designed to test among other inhibition (suppressing the response suggested by context) and selective attention. There were 80 participants: 39 men and 41 women (4 of them were excluded due to low quality of signal), each presented with 384 trials. The data were registered with BrainVision Recorder software at 2500 Hz sampling with actiCHamp (Brainproducts GmbH, Gilching, Germany) amplifier, 64 active electrodes (with active shielding on) placed according to 10–10 system, Cz as a reference electrode, and passive EOG electrodes (a bipolar montage with one electrode below and one in the corner of the eye, allowing recording of both horizontal and vertical eye movements).

### 2.2 Preprocessing

All signals were initially bandpass filtered in the range from 1 Hz to 1245 Hz using the default two-way least-squares FIR filter. Individual bad channels detected during

manual analysis were interpolated with a spherical spline algorithm. EEG was re-referenced to the common average reference (CAR). To speed up the computations and decrease the dimensionality of the training data the EEG was downsampled from 2500 Hz to 125 Hz sampling rate. Eye blinks filtering was applied where possible by removing ICA independent components highly correlated with EOG. The resulting ERP is presented in Fig. 3.



**Fig. 3.** Grand average over all the participants for incorrect and correct answers, after all the preprocessing procedures. Electrode Fz.

### 2.3 EEGNet Classifier

The EEGNet was selected for implementation because of its superior performance on the ERP datasets [3] and easily accessible implementation on Github ([github.com/vlawhern/arl-eegmodels](https://github.com/vlawhern/arl-eegmodels)). As recommended by the authors of the EEGNet architecture, the input to the network is represented by a simple 2D signal matrix with 14 channels as consecutive rows and 75 samples as consecutive columns.

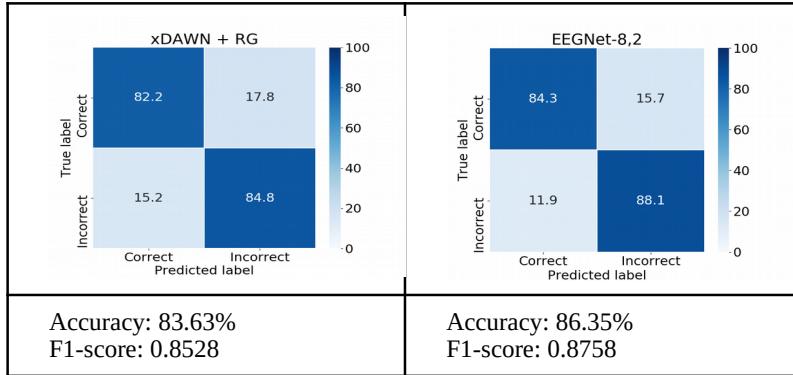
The training was performed in a cross-subject manner. 15% randomly selected correct and 15% randomly selected incorrect samples from each subject were left as the test set. Other 70% correct and 70% incorrect answers from each subject were added to the training set. The problem of unbalanced classes was solved here by simply adding additional samples augmented with Gaussian noise. The EEGNet-8,2 network was trained for 50 epochs with batch size 16. The training time was around 1 hour on a mediocre PC. The training loss achieved 0.2914 with the corresponding training F1-score of 0.9333. Validation loss and F1-score for the best model were 0.3137 and 0.9230.

## 3 Results and discussion

Exactly like in [3], the results of the network were compared with the state-of-the-art traditional machine learning approach employing a combination of xDAWN spatial filtering, Riemannian geometry, channel subset selection, and  $L_1$  feature regularization (xDAWN+RG) which achieved the best result in Kaggle BCI competition in 2015. The comparison of results for xDAWN+RG and EEGNet on the testing dataset is presented in a form of confusion matrices in Table 1. They suggest that deep learn-

ing allows to easily achieve results of the best traditional machine learning methods and may be applied to classification of erroneous responses and other types of ERPs.

**Table 1.** Confusion matrices and overall scores for xDAWN+RG method (on the left) and EEGNet (on the right) on the testing set



## Acknowledgements

Research supported by statutory funds of the Institute of Informatics (BKM-2019).

## References

1. A. Craik, Y. He, and J. L. Contreras-Vidal, “Deep learning for electroencephalogram (EEG) classification tasks: a review,” *Journal of Neural Engineering*, vol. 16, no. 3, p. 031001, Jun. 2019.
2. Y. Roy, H. Banville, I. Albuquerque, A. Gramfort, T. H. Falk, and J. Faubert, “Deep learning-based electroencephalography analysis: a systematic review,” *Journal of Neural Engineering*, May 2019.
3. V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, “EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces,” *Journal of Neural Engineering*, vol. 15, no. 5, p. 056013, Oct. 2018.
4. R. T. Schirrmeister et al., “Deep learning with convolutional neural networks for EEG decoding and visualization: Convolutional Neural Networks in EEG Analysis,” *Human Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, Nov. 2017.
5. B. A. Eriksen and C. W. Eriksen, “Effects of noise letters upon the identification of a target letter in a nonsearch task”, *Perception & Psychophysics* (1974) 16: 143. <https://doi.org/10.3758/BF03203267>.

# Ensemble data preprocessing based methods for imbalanced data stream classification

Jakub Klikowski

Department of Systems and Computer Networks,  
Wrocław University of Science and Technology,  
Wrocław, Poland  
[jakub.klikowski@pwr.edu.pl](mailto:jakub.klikowski@pwr.edu.pl)

**Abstract.** Imbalanced data and data streams are becoming a popular topic of research. Combining these two phenomena is a much more difficult problem. In this paper is proposed a new method that uses pre-processing techniques to create a classifier that deals with the problem of imbalanced data streams. The main contributions of this work is proposition of method which employ data sampling techniques to balance class distributions. Experimental evaluation of the proposed method and their comparison with state-of-art methods. The quality of the proposed algorithm was evaluated on the basis of computer experiments, using 26 real data streams and 60 generated data streams. The evaluation procedure was conducted in test-then-train manner. The metrics chosen were F-score, Gmean and AUC score. Non-parametrical statistical tests were performed namely the Friedman Test as well as a Nemenyi's Post-Hoc Procedure. The results obtained from statistical tests indicate the high quality of the classification of the presented method. It is worth mentioning, that in the proposed method, high imbalance ratio does not negatively impact on performance. In most of the compared method and data combinations, the algorithm presented in this article is statistically better than its state of art competitors.

**Keywords:** imbalanced data · data stream classification · data preprocessing.

## Acknowledgement

This work was supported by the Polish National Science Centre under the grant No. 2017/27/B/ST6/01325 as well as by the statutory funds of the Department of Systems and Computer Networks, Faculty of Electronics, Wrocław University of Science and Technology.

# Drifted Data Stream Classification using Oversampled Dynamic Ensemble Selection

Paweł Zybłiewski<sup>[0000-0002-4224-6709]</sup>

Department of Systems and Computer Networks,  
Faculty of Electronics, Wrocław University of Science and Technology,  
Wrocław, Poland  
[pawel.zyblewski@pwr.edu.pl](mailto:pawel.zyblewski@pwr.edu.pl)

**Abstract.** This work aims to connect two research trends important for real-life decision tasks, i.e., imbalanced data analysis and non-stationary data stream classification. We propose a novel framework using stratified bagging, dynamic ensemble selection (DES) and data preprocessing techniques for the classification of imbalanced data streams. During stratified bagging, each bootstrap is created by sampling with replacement both minority and majority classes separately in a way that preserves the number of instances in the original data chunk. The proposed approach has been evaluated based on the computer experiments carried out on 112 artificially generated data streams with various characteristics. During the experiments, the *test-then-train* evaluation procedure was used. We consider two DES methods (KNORA-E and KNORA-U) used on both bagging and base models level as well as two variations of preprocessing techniques based on the SMOTE algorithm (SVM-SMOTE and Borderline2-SMOTE). Experimentation results and statistical tests proved that the DES coupled with data preprocessing can outperform the approaches that do not combine both of these concepts.

**Keywords:** Dynamic ensemble selection · Imbalanced data · Data stream · Data preprocessing · Concept drift

## Acknowledgments

This work was supported by the Polish National Science Centre under the grant No. 2017/27/B/ST6/01325 as well as by the statutory funds of the Department of Systems and Computer Networks, Faculty of Electronics, Wrocław University of Science and Technology.

# Generation of context-free grammars for Grammar Inference methods<sup>\*</sup>

Olgierd Unold<sup>1</sup>[0000–0003–4722–176X], Łukasz Culer<sup>2</sup>[0000–0002–8614–5512], and  
Agnieszka Kaczmarek<sup>3</sup>

<sup>1</sup> Department of Computer Engineering  
Wrocław University of Science and Technology  
Wybrzeże Wyspiańskiego 50-370 Wrocław, Poland  
[olgierd.unold@pwr.edu.pl](mailto:olgierd.unold@pwr.edu.pl)

<sup>2</sup> Department of Computer Engineering  
Wrocław University of Science and Technology  
Wybrzeże Wyspiańskiego 50-370 Wrocław, Poland  
[lukasz.culer@pwr.edu.pl](mailto:lukasz.culer@pwr.edu.pl)

<sup>3</sup> Department of Computer Engineering  
Wrocław University of Science and Technology  
Wybrzeże Wyspiańskiego 50-370 Wrocław, Poland  
[agnieszka.kaczmarek@pwr.edu.pl](mailto:agnieszka.kaczmarek@pwr.edu.pl)

**Abstract.** There are numerous grammar inference methods, that use sets of both positive and negative examples as an algorithm input. The origin of these examples could be very diverse - from real-life data to manually crafted data. Both categories have their advantages and disadvantages. We present an alternative approach: the application of an automated grammar generator.

**Keywords:** Grammatical Inference · context-free grammars · grammar generator

## 1 Introduction

Real-life data, as learning sets, promise the greatest performance in industrial applications if grammars are inferred properly. However, due to an imperfection of measurement equipment, some examples could include errors. Moreover, some of the phenomena, that are expressed through those data cannot be covered with formal language theory methods.

As opposed to real-life data there are sets for manually crafted grammars. Despite many advantages, such as possessing full knowledge about them or the certainty that the examples are error-free, they also create some issues - creating a grammar of given complexity with positive and negative learning sets is a difficult and time-consuming task.

---

<sup>\*</sup> This abstract was created based on the detailed article that was submitted to Fundamenta Informaticae journal and is currently in review.

To provide a solution for those difficulties, in this article we present a complete and unique approach for automatic generation of coherent grammars. It allows specifying required grammar complexity (using the defined measure) or a number of specific rules. Integrated additional modules allow also to create positive and negative learning sets for a given grammar. All of the mentioned constituents make the output tailored to specific research needs, without additional effort (grammar is generated by a computer, the user has only to define its needs) and time (few seconds of generation process) spent on creating a grammar and sets manually. We decided to rely on context-free languages due to the richness of theoretical background, which we utilized to reliably justify the theory base of our approach, making it a solid base for grammar inference related research.

The algorithm starts with a set of given parameters, which constitute the base to generate an artificial context-free grammar. This grammar, converted to Chomsky Normal Form, is used to create positive and negative example sets. All this data is collected to create a complete kit for testing.

## 2 Grammar generator

### 2.1 Input

The core part of the algorithm is the grammar generator. It takes given parameters as an input: the exact number of parenthesis rules with non-terminal symbol ( $|R_P^+|$ ), the exact number of parenthesis rules without non-terminal symbol ( $|R_P^-|$ ), the exact number of branch rules ( $|R_B|$ ), exact number of iterative rules ( $|R_I|$ ), the maximum number of terminal symbols ( $S_T$ ) and the maximum number of non-terminal symbols ( $S_{NT}$ ).

Alternatively, all those parameters could be replaced with a simple indicator of grammar complexity - the sum of all grammar rules ( $|R|$ ). The rule types were selected based on paper [3].

### 2.2 Algorithm

The procedure starts with adding all parenthesis rules without a non-terminal symbol. During creation, the algorithm randomly chooses whether to create a new symbol (preserving the parameter requirements). Then, all other rules are created randomly using existing symbols or creating new ones. The creation of a non-terminal symbol is allowed only if it will be attached to the left-hand side of the new rule and the previously added non-terminal symbol would be applied to the right-hand side at least once. This approach ensures that all symbols are productive. The procedure also has to verify, that all the left-hand side symbols of parenthesis rules without non-terminal symbols are connected. The last step is a conversion of the recently added non-terminal symbol into a start symbol. This conversion makes all productive symbols achievable, which results in a consistent grammar.

### 2.3 Mathematical analysis

The mathematical analysis of the created method was performed based on its principles and properties of types of rules, that were utilized. This analysis resulted in a set of dependencies (Eq. 1), that input properties have to fulfill to create a consistent grammar.

$$\left\{ \begin{array}{l} \frac{|R_P^-|}{S_T^2} \leq S_{NT} \\ \frac{|R_P^-|}{S_T^2} \leq |R_P^+| + |R_I| + 2|R_B| + 1 \\ \sqrt{\frac{|R_P^+|}{S_T^2}} \leq S_{NT} \\ \sqrt{\frac{|R_I|}{2S_T}} \leq S_{NT} \\ \sqrt[3]{|R_B|} \leq S_{NT} \end{array} \right. \quad \text{where } \left\{ \begin{array}{l} S_{NT}, S_T, |R_P^-| \in \mathbb{N}_+ \\ |R_P^+|, |R_I|, |R_B| \in \mathbb{N}_0 \end{array} \right. \quad (1)$$

### 3 Test study

An example grammar generator run is printed below for given parameters -  $|R_P^-| = 2$ ,  $|R_P^+| = 0$ ,  $|R_B| = 1$ ,  $|R_I| = 2$ ,  $S_{NT} = 4$  and  $S_T = 3$ . The symbols and rules added in a certain step are marked in bold. The final grammar obtained in the last step is presented in Fig. 1. The final grammar was obtained using the latest version of the tool available at webpage [4].

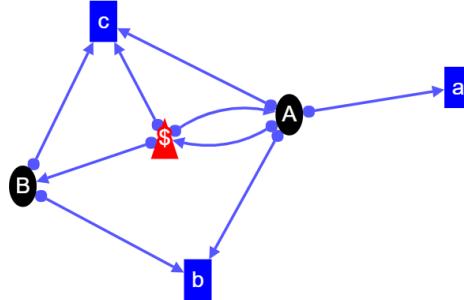
**Table 1.** Example grammar generator run.

| Step 1   |               | Step 2   |               | Step 3   |               | Step 4 |               | Step 5 |               | Step 6 |         |
|----------|---------------|----------|---------------|----------|---------------|--------|---------------|--------|---------------|--------|---------|
| S        | R             | S        | R             | S        | R             | S      | R             | S      | R             | S      | R       |
| <b>A</b> | <b>A → ab</b> | A        | A → ab        | A        | A → ab        | A      | A → ab        | A      | A → ab        | A      | A → ab  |
| a        |               | <b>B</b> | <b>B → bc</b> | B        | B → bc        | B      | B → bc        | B      | B → bc        | B      | B → bc  |
| b        |               | a        |               | <b>C</b> | <b>C → AA</b> | C      | C → AA        | C      | C → AA        | \$     | \$ → AA |
|          |               | b        |               | a        |               | a      | <b>A → Cc</b> | a      | A → Cc        | a      | A → \$c |
|          |               | <b>c</b> |               | b        |               | b      |               | b      | <b>C → Bc</b> | b      | \$ → Bc |
|          |               |          |               | c        |               | c      |               | c      |               | c      |         |

### 4 Positive and negative set creation

One of the implemented positive set generator algorithms was introduced based on the paper [1]. Set creation begins with grammar conversion to a linear grammar. Then, a graph is created based on it. Positive examples are created using given paths that consist of 1-, 2- and 3-element combinations of rules.

The second one uses the external tool known as GenRGenS described in detail in [2].



**Fig. 1.** The generated grammar visualization.

The negative set is created iteratively - a random string built using terminal symbols is created and entered into the CYK algorithm [5]. If the algorithm does not parse the string, it does not belong to the language, so it is a negative example. This procedure is repeated until the demanded number of examples is created.

## 5 Future work

The future work will focus on introducing new grammar attributes that describe their behaviour in terms of example generation and structure. Consequently, that would lead to a new parameter creation for the generation process, allowing the generated grammar's specific attributes to be easier to control and customize.

## 6 Acknowledgements

The research was supported by National Science Centre Poland (NCN), project registration no. 2016/21/B/ST6/02158.

## References

1. Mayer, M., Hamza, J.: Optimal test sets for context-free languages. arXiv preprint arXiv:1611.06703 (2016)
2. Ponty, Y., Termier, M., Denise, A.: Genrgens: Software for generating random genomic sequences and structures. Bioinformatics **22**(12), 1534–1535 (June 2006)
3. Sakakibara, Y.: Learning context-free grammars using tabular representations. Pattern Recognition **38**(9), 1372–1383 (2005)
4. Unold, O., Kaczmarek, A., Culer, L.: Language generator. <http://lukasz.culer.staff.iiar.pwr.edu.pl/gencreator.php>, accessed: 2019-09-09
5. Younger, D.H.: Recognition and parsing of context-free languages in time  $n^3$ . Information and control **10**(2), 189–208 (1967)

# Assessment of Electric City Buses in the Tendering Process: MCDA Case Study

Aleksandra Bączkiewicz<sup>1</sup>[0000-0003-4249-8364]

<sup>1</sup> Department of Artificial Intelligence method and Applied Mathematics in the Faculty of Computer Science and Information Technology, West Pomeranian University of Technology, Szczecin 71-210, Poland;  
(e-mail: aleksandra\_baczkiewicz@zut.edu.pl)

**Abstract.** This paper presents two methods to facilitate a proper decision on the choice of the most advantageous alternative among the available ones: TOPSIS and COMET. The operation of these methods was presented on the example of the selection of the most beneficial offer for the purchase of zero-emission electric buses in the tender announced by the Szczecin City Hall. The provisions of the Public Procurement Act oblige the contracting authority to conduct tenders to select the most advantageous offer. The contracting authority decides a tender based on criteria indicated in the contract documents (siwz document). The most advantageous tender shall be the one presenting the most advantageous balance between price and other criteria relating to the subject-matter of the public contract.

**Keywords:** city transport · decision-making · multi-criteria decision analysis · TOPSIS · COMET method.

## 1 Introduction

The development of low-carbon and zero-carbon transport is one of the priorities of EU environmental policy. The need to develop alternative fuel infrastructure in transport is clearly highlighted in the European Commission's White Paper of 28<sup>th</sup> March 2011 entitled "*Roadmap to a Single European Transport Area - Towards a competitive and resource efficient transport system*". The document calls for a reduction in the dependence of transport on oil. In addition, transport is expected to reduce greenhouse gas emissions by 60% by 2050 compared to 1990. In 2017 The Economic Committee of the Council of Ministers, at the request of the Prime Minister, presented recommendations under the name of the "*Clean Air*" Programme. Measures to improve air quality were also included in government strategic documents - *the Strategy for Responsible Development until 2020 (with a perspective until 2030)* and *the Electromobility Development Plan "Energy for the Future"*.

*The Electromobility Act* provides for and imposes an obligation on public transport organisers and operators to ensure the share of zero emission buses in the fleet of vehicles in use, amounting to respectively: 5% - from

1<sup>st</sup> January 2021; 10% - from 1<sup>st</sup> January 2023; 20% - from 1<sup>st</sup> January 2025; by local government units referred to in Article 36(1) (i.e. local government units, excluding communes and poviats whose number of inhabitants does not exceed 50 000).

The market share of electric buses has featured steady growth in recent years [1]. The Szczecin City Hall also undertakes comprehensive actions to improve the functioning of public transport and reduce low emissions in the city area. Within the framework of these activities, the city purchased, by the way of public procurement, 11 zero-emission electric buses.

**TOPSIS** (*the Technique for Order of Preference by Similarity to Ideal Solution*) is a commonly used multi-criteria decision-making method based on finding an alternative that is as close to the ideal solution as possible and as far away from the ideal solution as possible. A detailed description of the stages and applications of this method has been described in the article [2].

**COMET** (*the Characteristic Objects METhod*) is a method of identification of multi-criteria expert decision model based on the idea of characteristic objects, which are points in the space of the state of the problem. It is used in solving decision problems. Its detailed description is presented in the articles [3,4]. In the COMET method, the criteria are grouped and the preferential values of all characteristic objects are determined on the basis of the tournament method and the indifference principle. Constructed model is used to calculate preference values of the alternatives, making it a multicriteria model that is free of rank reversal [3].

## 2 Results and discussions

This paper presents a model of decision-making concerning zero-emission urban transport. The task was to choose the optimal bus model from 6 models with the necessity to take into account 6 criteria (the tender announcement published on 12<sup>th</sup> September 2018). The following criteria are specified:

- $C_1$  - maintenance and servicing susceptibility, expressed in points, 1 point for outer sheathing made by means of gluing or riveting, 3 points for outer sheathing made with screws, without the need for welding and gluing;
- $C_2$  - type of suspension, expressed in points, 0 points for independent suspension and 3 points for dependent suspension;
- $C_3$  - type of windscreen, expressed in points, 0 points for single-part glass and 3 points for a glass pane divided vertically in the middle;
- $C_4$  - price, expressed in Polish zloty (PLN);
- $C_5$  - energy consumption expressed in kilowatt hours per kilometres (kWh/km);
- $C_6$  - air-conditioning - quantity of refrigerant expressed in kilograms (kg);

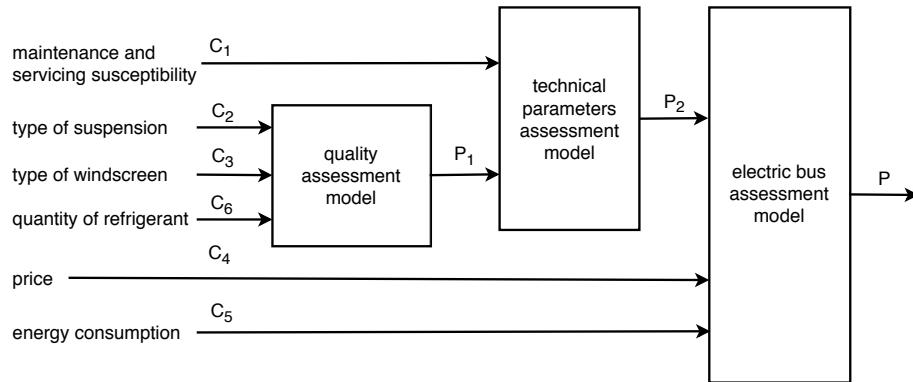
$C_1-C_3$  were indicated as profit criteria and  $C_4-C_6$  as cost criteria.  $C_4$  and  $C_5$  were indicated as the most important criteria in the tender.

**Table 1.** Selected criteria  $C_1-C_6$  and their characteristic values {low, medium, high}.

| $C_i$ Name                                     | Unit   | Low       | Medium    | High      |
|--|--------|-----------|-----------|-----------|
| $C_1$ maintenance and servicing susceptibility | points | 1         | -         | 3         |
| $C_2$ type of suspension                       | points | 0         | -         | 3         |
| $C_3$ type of windscreen                       | points | 0         | -         | 3         |
| $C_4$ price                                    | PLN    | 2 018 430 | 3 330 802 | 4 782 147 |
| $C_5$ energy consumption                       | kWh/km | 0.8       | 1.01      | 1.25      |
| $C_6$ quantity of refrigerant                  | kg     | -         | -         | 3         |

Creating a hierarchical structure of the decision-making problem in COMET method allows to reduce the number of pairwise comparisons to the expert [5]. Table 1. presents the characteristic values and Fig. 1. shows the hierarchical structure of the problem. The complete performance data are presented in Table 2. and Table 3. respectively for COMET and TOPSIS method.

In both methods the final decision assessment model indicated *EBN 11* as the most advantageous alternative and *eCitaro* as the most unfavorable alternative.

**Fig. 1.** The hierarchical structure of the electric bus assessment problem in COMET.**Table 2.** The performance table of the alternatives in COMET.

| Pos. Name                    | Profit |       |       | Cost      |       |       | $P_1$ | $P_2$ | $P$    |
|------------------------------|--------|-------|-------|-----------|-------|-------|-------|-------|--------|
|                              | $C_1$  | $C_2$ | $C_3$ | $C_4$     | $C_5$ | $C_6$ |       |       |        |
| 1 SOR EBN 11                 | 1      | 0     | 0     | 2 018 430 | 1.08  | 3     | 0     | 0     | 0.7649 |
| 2 BYD K9 (eBUS-12)           | 1      | 0     | 0     | 2 410 667 | 1.08  | 3     | 0     | 0     | 0.6581 |
| 3 Ursus City Smile 12        | 3      | 3     | 3     | 3 480 900 | 1.03  | 3     | 1     | 1     | 0.5285 |
| 4 Volvo 7900 Electric        | 1      | 0     | 0     | 2 803 170 | 1.25  | 3     | 0     | 0     | 0.5007 |
| 5 Solaris Urbino 12 Electric | 3      | 3     | 3     | 4 489 500 | 0.84  | 3     | 1     | 1     | 0.3443 |
| 6 Mercedes-Benz eCitaro      | 3      | 0     | 3     | 4 782 147 | 0.8   | 3     | 0.5   | 0.83  | 0.2614 |

**Table 3.** The performance table of the alternatives in TOPSIS method.

| Pos. Name                    | Profit |       |       | Cost      |       |       | $C_i$   |
|------------------------------|--------|-------|-------|-----------|-------|-------|---------|
|                              | $C_1$  | $C_2$ | $C_3$ | $C_4$     | $C_5$ | $C_6$ |         |
| 1 SOR EBN 11                 | 1      | 0     | 0     | 2 018 430 | 1.08  | 3     | 0.79425 |
| 2 BYD K9 (eBUS-12)           | 1      | 0     | 0     | 2 410 667 | 1.08  | 3     | 0.74426 |
| 3 Volvo 7900 Electric        | 1      | 0     | 0     | 2 803 170 | 1.25  | 3     | 0.63152 |
| 4 Ursus City Smile 12        | 3      | 3     | 3     | 3 480 900 | 1.03  | 3     | 0.49631 |
| 5 Solaris Urbino 12 Electric | 3      | 3     | 3     | 4 489 500 | 0.84  | 3     | 0.25912 |
| 6 Mercedes-Benz eCitaro      | 3      | 0     | 3     | 4 782 147 | 0.8   | 3     | 0.21116 |
| maximum value                | 3      | 3     | 3     | 4 782 147 | 1.25  | 3     |         |
| criteria weights             | 0.05   | 0.05  | 0.05  | 0.6       | 0.2   | 0.05  |         |

In the ranking obtained by the COMET method in the case of *City Smile 12* and *Volvo 7900 Electric* alternatives, the importance of the expert's preference for quality criteria of the assessed models is greater than in the case of the TOPSIS ranking. Other alternative positions in both rankings are the same and they are closely linked to the price - the most important criterion in the tender, like in the TOPSIS ranking.  $\rho$  Spearman's rank correlation coefficient between rankings obtained for both methods is equal to 0.9429 and p-value is equal to 0.0048. This indicates a strong correlation between the both rankings.

### 3 Conclusions

The organization of public tenders always arouses high emotions, accompanied by large amounts of public money spent. The paper presents the possibility of using two MCDA methods to determine the offers ranking. Despite the very high correlation, the two alternatives have different positions in both rankings. It is planned to extend the research to other MCDA methods and to check their suitability for the selection of tender offers.

### References

1. Mahmoud M., Garnett R., Ferguson M., Kanaroglou P.: Electric buses: A review of alternative powertrains. *Renewable and Sustainable Energy Reviews* **62**, 673–684 (2016).
2. Sałabun W.: The mean error estimation of TOPSIS method using a fuzzy reference models. *Journal of Theoretical and Applied Computer Science* **7**(3), 40–50 (2013).
3. Sałabun W.: The Characteristic Objects Method: A New Distance-based Approach to Multicriteria Decision-making Problems. *Journal of multi-criteria decision analysis* **22**(1-2), 37–50 (2014).
4. Wałtrowski J., Sałabun W.: The Characteristic Objects Method: A New Intelligent Decision Support Tool for Sustainable Manufacturing. *Smart Innovation, Systems and Technologies*. **52**, 349–359 (2016).
5. Sałabun W., Karczmarczyk A.: Using the COMET Method in the Sustainable City Transport Problem: an Empirical Study of the Electric Powered Cars. *Procedia Computer Science*. **126**, 2248–2260 (2018).

# Effect of various normalization techniques on the TOPSIS method

Krzysztof Palczewski<sup>1</sup>[0000–0001–5035–1923]

Department of Artificial Intelligence Methods and Applied Mathematics,  
Faculty of Computer Science and Information Technology,  
West Pomeranian University of Technology in Szczecin ul. Żołnierska 49, 71-210  
Szczecin, Poland

**Abstract.** Normalization is a vital part of many multi-criteria decision-analysis methods (MCDA). In this paper, the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS), as one of the most popular MCDA methods, is investigated. The influence of five main normalization techniques, as well as the effects of the case without normalization on the final results of TOPSIS method is examined, based on empirical data.

**Keywords:** Normalization · TOPSIS · Multi-criteria decision-analys.

## 1 Introduction

Multi-criteria decision analysis (MCDA) methods have been successfully used in many ranking, assessment or selection problems, aiding and improving the decision-making process in applications in fields such as business, supply chain management, engineering, logistics, health or sustainable development, among others [1–3]. In the majority of scenarios, MCDA challenges are often characterized by many, usually contradictory, criteria. Furthermore, in these cases, decision matrix consists of the data not suitable for further proceeding, because they could be of different orders of magnitude or have incompatible measurements units. Hence, the important part of many MCDA methods is the normalization step, in which the decision matrix (table of the alternatives) is transformed by one of the normalization techniques. TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) developed by Hwang and Yoon [4] is one of the most popular, extensively known and widely implemented MCDA method. By its simplicity, computational efficiency and comprehensive mathematical concept, it has been thoroughly adopted in many real-life applications. This study compares five main normalization methods. The aim of this paper is to investigate the effects of normalization techniques on the final results of the TOPSIS method. Study case of classical TOPSIS method implemented on the empirical set of alternatives is conducted, in order to present the results of the normalization effects.

## 2 Normalization methods

Five main normalization methods are presented below. It is important to note that profit type criteria and cost type criteria should be normalized differently.

**Method 1 - Minimum-Maximum** This method uses the greatest and the least values in the considered set. The formulas are described for profit type (1) and cost type criteria (2) respectively as follows:

$$r_{ij} = \frac{x_{ij} - \min_j(x_{ij})}{\max_j(x_{ij}) - \min_j(x_{ij})} \quad (1)$$

$$r_{ij} = \frac{\max_j(x_{ij}) - x_{ij}}{\max_j(x_{ij}) - \min_j(x_{ij})} \quad (2)$$

**Method 2 - Maximum** This method uses the only the greatest value in the considered set. The formulas are described for profit type (3) and cost type criteria (4) respectively as follows:

$$r_{ij} = \frac{x_{ij}}{\max_j(x_{ij})} \quad (3)$$

$$r_{ij} = 1 - \frac{x_{ij}}{\max_j(x_{ij})} \quad (4)$$

**Method 3 - Sum** This method uses the sum of all values in the considered set. The formulas are described for profit type (5) and cost type criteria (6) respectively as follows:

$$r_{ij} = \frac{x_{ij}}{\sum_{i=1}^m(x_{ij})} \quad (5)$$

$$r_{ij} = \frac{\frac{1}{x_{ij}}}{\sum_{i=1}^m(x_{ij})} \quad (6)$$

**Method 4 - Square root of sum** This method uses the square root of the sum of all elements. Although it may look similar to the Method 3 by using the sum of all elements, its results are quite different. All the previous 3 methods are scaled to the range of zero to one, but this method does not. The formulas are described for profit type (7) and cost type (8) respectively as follows:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m(x_{ij})}} \quad (7)$$

$$r_{ij} = 1 - \frac{x_{ij}}{\sqrt{\sum_{i=1}^m(x_{ij}^2)}} \quad (8)$$

**Method 5 - Logarithmic** This method of normalization uses the natural logarithm. Values of considered set are assumed to be positive. The formulas are described for profit type (9) and cost type (10) [5] respectively as follows:

$$r_{ij} = \frac{\ln(x_{ij})}{\ln(\prod_{i=1}^m x_{ij})} \quad (9)$$

$$r_{ij} = \frac{1 - \frac{\ln(x_{ij})}{\ln(\prod_{i=1}^m x_{ij})}}{m - 1} \quad (10)$$

### 3 The TOPSIS method

As the most popular MCDA method, TOPSIS is broadly used and well-known, not only in the field of the decision-making process but also in other areas of computer science, economics or business. The detailed explanation and in-depth analysis of the steps of this method will not be stated in this paper. However, the method described can be found in [6–8].

### 4 Results and discussion

In this study, five normalization procedures, described in section 2, are applied to the TOPSIS method. For this study case, empirical data with ten alternatives ( $A_0 - A_9$ ) and five criteria ( $C_1 - C_5$ ) is used. For simplicity, all criteria have equal weights. In the Table 1 decision matrix is represented. The steps of TOPSIS method are conducted for each method of normalization independently, using normalized decision matrices. The Table 2 depicts the calculated final preferences for each alternative in regard to case without normalization and cases examined with five normalization methods respectively. Based on the final preferences obtained through TOPSIS method, rankings were created for each case, presented in Table 3.

**Table 1.** Decision matrix of empirical data

| $A_i$ | $C_1$ | $C_2$ | $C_3$ | $C_4$    | $C_5$ |
|-------|-------|-------|-------|----------|-------|
| $A_0$ | 2000  | 30    | 1     | 40000    | 1     |
| $A_1$ | 5000  | 50    | 2     | 10000000 | 2     |
| $A_2$ | 3000  | 45    | 3     | 20000000 | 3     |
| $A_3$ | 2000  | 30    | 4     | 300000   | 20    |
| $A_4$ | 6000  | 44    | 5     | 2000000  | 5     |
| $A_5$ | 2000  | 55    | 6     | 60000    | 6     |
| $A_6$ | 4000  | 45    | 7     | 5000000  | 7     |
| $A_7$ | 3000  | 65    | 3     | 4000000  | 400   |
| $A_8$ | 1000  | 1000  | 7     | 3000000  | 1000  |
| $A_9$ | 1000  | 1000  | 1     | 400000   | 1000  |

**Table 2.** Comparison of the preferences by various methods of normalization

| $A_i$ | No norm. | Max-Min  | Max      | Sum      | Square Root | Logarithmic |
|-------|----------|----------|----------|----------|-------------|-------------|
| $A_0$ | 0.000051 | 0.084959 | 0.075931 | 0.043437 | 0.06381     | 0.028719    |
| $A_1$ | 0.498998 | 0.358273 | 0.34005  | 0.294848 | 0.342155    | 0.244661    |
| $A_2$ | 0.999835 | 0.403877 | 0.409    | 0.433794 | 0.459293    | 0.33883     |
| $A_3$ | 0.013026 | 0.216024 | 0.197505 | 0.102884 | 0.148326    | 0.489441    |
| $A_4$ | 0.098197 | 0.415089 | 0.377809 | 0.226282 | 0.313727    | 0.442667    |
| $A_5$ | 0.001003 | 0.311096 | 0.286522 | 0.153087 | 0.216527    | 0.46926     |
| $A_6$ | 0.248497 | 0.421344 | 0.389462 | 0.244605 | 0.323157    | 0.505888    |
| $A_7$ | 0.198397 | 0.293272 | 0.28578  | 0.253333 | 0.265105    | 0.688667    |
| $A_8$ | 0.148297 | 0.569602 | 0.5797   | 0.591251 | 0.550382    | 0.917959    |
| $A_9$ | 0.018036 | 0.450993 | 0.473922 | 0.539319 | 0.478447    | 0.587068    |

**Table 3.** Comparison of the final rankings by various methods of normalization

| $A_i$ | No norm. | Max-Min | Max | Sum | Square Root | Logarithmic |
|-------|----------|---------|-----|-----|-------------|-------------|
| $A_0$ | 1        | 1       | 1   | 1   | 1           | 1           |
| $A_1$ | 6        | 4       | 4   | 4   | 4           | 2           |
| $A_2$ | 4        | 8       | 8   | 6   | 6           | 3           |
| $A_3$ | 10       | 6       | 6   | 5   | 8           | 5           |
| $A_4$ | 5        | 2       | 2   | 7   | 5           | 6           |
| $A_5$ | 9        | 3       | 5   | 8   | 7           | 4           |
| $A_6$ | 8        | 5       | 7   | 2   | 2           | 7           |
| $A_7$ | 7        | 7       | 3   | 3   | 3           | 10          |
| $A_8$ | 2        | 10      | 10  | 10  | 10          | 8           |
| $A_9$ | 3        | 9       | 9   | 9   | 9           | 9           |

The final ranking of the alternatives differs considering various normalization techniques, what is presented in Table 2 and is visible in Table 3. The highest-ranked alternative  $A_0$  was given the highest preference in all cases, even including the case without any normalization. However, in any other alternative such scenario is not duplicated. Although some similarities occur between methods of normalization, in a majority of cases, alternatives are ranked differently by various normalization techniques. Even though some normalization methods exhibit similar properties or even their mathematical concepts are analogous, not a single pair of cases of normalization produced exactly identical results. However, it should be noted that the similarity of rankings between the cases with normalization and without normalization is smaller than those between normalization only.

## 5 Conclusions

Normalization step is a crucial part of many multi-criteria decision-analysis methods. In this study, the Technique for Order of Preference to Ideal Solution (TOPSIS) was used. Ten alternatives and five criteria were taken into consideration. Five normalization procedures were implemented. Additionally, TOPSIS

without normalization was examined too. The obtained results were compared, and the influence of distinct normalization methods was investigated and described. The final conclusions concerning the TOPSIS method show that they do have a considerable impact on the rankings, as the differences between normalization methods were significant, producing various outcomes. Future studies should be focused on analyzing normalization effects on different methods.

## References

1. Opricovic, S., Tzeng, G. H. (2004). Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS. European journal of operational research, 156(2), 445-455.
2. Watróbski, J., Sałabun, W. (2016, April). Green supplier selection framework based on multi-criteria decision-analysis approach. In International Conference on Sustainable Design and Manufacturing (pp. 361-371). Springer, Cham.
3. Sałabun, W. (2015). The Characteristic Objects Method: A New Distance-based Approach to Multicriteria Decision-making Problems. Journal of Multi-Criteria Decision Analysis, 22(1-2), 37-50.
4. Hwang, C. L., Masud, A. S. M. (2012). Multiple objective decision making—methods and applications: a state-of-the-art survey (Vol. 164). Springer Science Business Media.
5. Vafaei, N., Ribeiro, R. A., Camarinha-Matos, L. M. (2016, April). Normalization techniques for multi-criteria decision making: analytical hierarchy process case study. In doctoral conference on computing, electrical and industrial systems (pp. 261-269). Springer, Cham.
6. Lai, Y. J., Liu, T. Y., Hwang, C. L. (1994). Topsis for MODM. European journal of operational research, 76(3), 486-500.
7. Opricovic, S., Tzeng, G. H. (2004). Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS. European journal of operational research, 156(2), 445-455.
8. Sałabun, W. (2013). The mean error estimation of TOPSIS method using a fuzzy reference models. Journal of Theoretical and Applied Computer Science, 7(3), 40-50.

# RAndom Neural Networks (RANNs): a new general classifier inspired by Random Forest

Górecki Tomasz<sup>1</sup>[0000–0002–9969–5257] and Paweł Piasecki<sup>2</sup>[0000–0002–4206–4550]

<sup>1</sup> Faculty of Mathematics and Computer Science, Adam Mickiewicz University,  
Poland

[tomasz.gorecki@amu.edu.pl](mailto:tomasz.gorecki@amu.edu.pl)

<sup>2</sup> Faculty of Mathematics and Computer Science, Adam Mickiewicz University,

Poland

[pawel.piasecki@amu.edu.pl](mailto:pawel.piasecki@amu.edu.pl)

**Abstract.** Our study aims to propose a powerful extension of neural network classifier - Random Neural Networks. The basic idea is to enrich the ensemble of neural networks by the techniques originated from Random Forest. This paper makes the initial commitment to the investigation of how neural networks can benefit from injecting randomness in the training process. We expect that in the future Random Neural Networks can be broadly developed, as — comparing to Random Forest — there are a lot of base classifiers' variants. We would like to propose a simple and general-purpose architecture, that will not involve much effort from the user to get satisfactory results.

**Keywords:** Neural networks · Random forest · Classification

## 1 Introduction

In recent years, an enormous increase of interest in machine learning has been observed. In the literature we can find a lot of new learning algorithms for classification and for regression as well, to list only a few: XGBoost [3], Light GMB [10], CNN-LSTM recurrent neural networks [17], Neural Oblivious Decision Ensembles [13].

One of the groups that we consider particularly interesting is ensemble methods. Ensemble methods construct a set of classifiers and then classify new data points by taking a (weighted) vote of their predictions [5]. The idea of combining results coming from different "experts" to get an overall, boosted decision is rooted in our culture [15] and, besides having a strong theoretical background, it is very intuitive.

It is a widespread fact, that bagging improves the robustness of the base classifier, whereas random feature selection enhances the accuracy in domains described by many input features [14]. It seems reasonable, that neural networks can benefit from injecting randomness in the learning process.

Ensembles of neural networks have already been described (for example in [7]), but, to our best knowledge, none proposed applying more random steps (random feature selection, random layer size as we propose) yet.

### 1.1 Neural networks

Since 1943, when McCulloch and Pitts [12] proposed the first neural network architecture, an enormous number of models utilizing it have been proposed. Nowadays, we can distinguish 3 leading types of neural networks. The first one, Deep Feed Forward (DFF) Neural Networks, has been proposed by Ivakhenko and Lapa [9] and as the simplest type of neural networks, provide a basis for further development. DFF networks consist of multiple layers of computational units (neurons), interconnected in a feed-forward manner. The second type of neural networks - Convolutional Neural Networks (CNNs) were first described by Fukushima [6], and - as indicated by its name - employ a mathematical operation called convolution. CNNs are neural networks that use convolution, instead of general matrix multiplication in at least one layer. The third group, Recurrent Neural Networks (RNNs) [16], let connections between neurons form a directed graph along a temporal sequence, which allows us to manage temporal dynamic behaviour. Long Short-Term Memory [8] Networks (LSTMs) is a specific kind of RNNs, dealing with the exploding and vanishing gradient problems encountered commonly when training RNNs. LSTMs solved many previously unlearnable deep learning tasks, as they allow to control state and memory of a single neuron, and in consequence to deal with time dependences in data more efficiently.

### 1.2 Random Forest

Random Forest is an ensemble learning algorithm proposed by Breiman [2]. The basic idea of the algorithm is to build many small decision-trees splitting at each node on the best feature selected from a random features' subset. A single decision-tree is a weak learner, but taking the majority vote (or averaging) of many trees we can obtain a strong learner. One of the main improvements of Random Forest is, that it avoids overfitting, which was decision trees' habit. Also, comparing to some boosting algorithms, Random Forest is more robust with respect to noise [2]. One thing that is characteristic to Random Forest (and in general models utilizing bootstrap aggregating) is using out-of-bag (OOB) error which is the mean prediction error on each training sample, computed only by trees that did not have this sample in their training set (bootstrap sample).

Random Forest is considered to be one of the most efficient general-purpose classification and regression method [1]. With only a few parameters to tune, its popularity seems to be well deserved.

## 2 Random Neural Networks

For an ensemble classifier, it is necessary that base classifiers are accurate and diverse [7]. As DFF Neural Networks are a powerful algorithm themselves, we claim that diversifying by injecting randomness possibly could make them competitive to the state-of-the-art solutions.

The process of training DFF Networks contains randomness itself, as initial weights are set randomly. Apart from it, in Random Network classifier, we

propose to add 3 sources of randomness: random sample selection (by bootstrapping), random feature selection and random base networks' architecture. All of them have reasonable theoretical justification. Bootstrapping is considered to lower the variance and to help avoid overfitting, while random feature selection limits the estimation bias due to multicollinearity. RANNs' architecture increases base classifiers' diversity and is realized by sampling (for the moment from a 3-element, arbitrarily chosen set) the size of layers in the network, before construction of each base network in Random Network ensemble. For now, we use simple, only two-layer networks.

Summarizing, RANNs is a bagged classifier combining a collection of  $N$  classification networks. Each network  $n$  is trained on a different bootstrap sample  $S_n$  (instances are drawn randomly with replacement from the original training set) and using  $m$  features selected at random out of  $M$  features in the original training set. Before each network is trained, the number of neurons in each of the two layers is sampled uniformly from an arbitrarily chosen set (described in the last section). Finally, predicted classes are computed by majority voting. We present pseudocode of Random Neural Networks in Algorithm 1.

---

**Algorithm 1** Random Neural Networks

---

**Input:** A training set  $\mathbf{S} := (x_1, y_1), \dots, (x_n, y_n)$ , features  $\mathbf{F}$  and number of networks in TODO networks ensemble  $\mathbf{B}$ .

```

1: function RANDOMNEURALNETWORKS( $\mathbf{S}, \mathbf{F}, \mathbf{B}$ )
2:    $H \leftarrow \emptyset$ 
3:   for  $i \in 1, \dots, \mathbf{B}$  do
4:      $S^{(i)} \leftarrow$  A bootstrap sample from  $\mathbf{S}$ 
5:      $h_i \leftarrow$  RANDOMIZEDNETENSLEARN( $S^{(i)}, \mathbf{F}$ )
6:      $H \leftarrow H \cup h_i$ 
7:   return  $H$ 
8: function RANDOMIZEDNETENSLEARN( $\mathbf{S}, \mathbf{F}$ )
9:    $f \leftarrow$  subset of  $\mathbf{F}$ 
10:  return The learned network on features  $f$ 
```

---

### 3 Results and conclusion

We performed the main experiment on two open-source datasets MNIST [11] and Fashion-MNIST [18]. Both of them have identical consist of a training set of 60000 examples and a test set of 10000 examples. Each example is a 28x28 greyscale image, assigned to one out of 10 classes. We perform basic preprocessing - flattening each example to  $1 \times 784$  vector and division by 255, to normalize values into the range  $[0, 1]$ .

We present results of classification in Figure 1. As a base classifier, we have experimentally chosen a simple neural network with two hidden layers of sizes, which are sampled (separately for each network) from a 3-element set

$\{2m, m, 0.5m\}$ , where  $m = 250$  is the number of randomly drawn input features. In between layers, we drop out 20% of units. In two hidden layers, we use ReLU activation function, while on the output layer we use Softmax function. Networks are trained using a backpropagation algorithm, with categorical cross-entropy as loss function and using Adam optimizer. Moreover, we set up the number of epochs to 30 and batch size to 128. Finally, we use an ensemble of 200 such networks.

In Table 3, we present a comparison of Random Neural Networks with other general classifiers that originated from [19]. By "general" we mean all-purposes classifiers, as a comparison with tailor-made classifiers (e.g. MCDNN [4] achieving 0.9980 accuracy on MNIST) is beyond the scope of this study. Comparing to other general classifiers, Random Neural Networks presents the highest performance on both datasets - prediction accuracy on test sets are equal (respectively): 0.985 and 0.901. During our experiments, we have noted, that already ensembles of 100 networks gives stable results, that do not change much after adding next networks

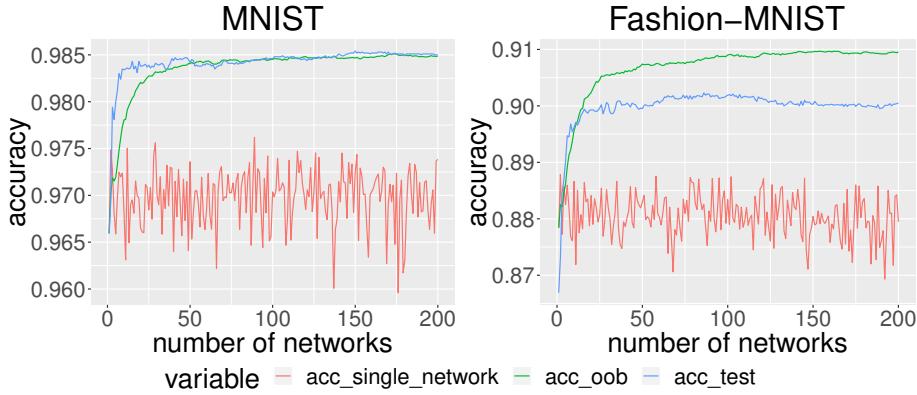
**Table 1.** Comparison of prediction accuracy of Random Neural Networks with other general classifiers.

| Classifier                 | Prediction error rate |               |
|----------------------------|-----------------------|---------------|
|                            | MNIST                 | Fashion-MNIST |
| Random Neural Networks     | 0.985                 | 0.901         |
| SVC                        | 0.978                 | 0.896         |
| MLPClassifier              | 0.972                 | 0.874         |
| Random Forest              | 0.971                 | 0.876         |
| GradientBoostingClassifier | 0.969                 | 0.879         |
| XgBoost                    | 0.958                 | 0.898         |
| KNeighborsClassifier       | 0.955                 | 0.850         |
| LogisticRegression         | 0.917                 | 0.840         |

Initial results show, that Random Neural Networks outperform other general classifiers. Nevertheless, there is a need to evaluate them on other datasets from various domains, as well as to find the best general-purpose architecture by tuning hyperparameters of basic classifiers. Hopefully, in future works, the performance may be boosted by implementing convolutional layers to the architecture.

## References

1. Biau, G., Scornet, E.: A random forest guided tour. *TEST* **25**(2), 197–227 (2016)
2. Breiman, L.: Random forests. *Machine Learning* **45**(1), 5–32 (2001)
3. Chen, T., Guestrin, C.: XGBoost: A scalable tree boosting system. In: KDD’16. pp. 785–794. ACM, New York, NY, USA (2016)
4. Ciregan, D., Meier, U., Schmidhuber, J.: Multi-column deep neural networks for image classification. In: 2012 IEEE CVPR. pp. 3642–3649 (June 2012)



**Fig. 1.** Test (blue color), OOB (green color) and single's networks (red color) accuracy for Random Neural Networks on MNIST and Fashion-MNIST datasets.

5. Dietterich, T.G.: Ensemble methods in machine learning. In: MCS 2010. pp. 1–15. Springer-Verlag, London, UK, UK (2000)
6. Fukushima, K.: Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics **36**, 193–202 (1980)
7. Hansen, L., Salamon, P.: Neural network ensembles. Pattern Analysis and Machine Intelligence, IEEE Transactions on **12**, 993 – 1001 (11 1990)
8. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
9. Ivakhnenko, A.G., Lapa, V.G.: Cybernetic predicting devices (1965)
10. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W.J., Ma, W., Ye, Q., Liu, T.M.: Lightgbm: A highly efficient gradient boosting decision tree. In: NIPS (2017)
11. LeCun, Y., Cortes, C.: MNIST handwritten digit database (2010), <http://yann.lecun.com/exdb/mnist/>
12. McCulloch, W.S., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics **5**(4), 115–133 (1943)
13. Popov, S., Morozov, S., Babenko, A.: Neural oblivious decision ensembles for deep learning on tabular data (2019)
14. Prinzie, A., Van den Poel, D.: Random multiclass classification: Generalizing random forests to random mnl and random nb. vol. 4653, pp. 349–358 (08 2007)
15. Re, M., Valentini, G.: Ensemble methods: A review, pp. 563–594 (01 2012)
16. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning Representations by Back-propagating Errors. Nature **323**(6088), 533–536 (1986)
17. Sainath, T.N., Vinyals, O., Senior, A., Sak, H.: Convolutional, long short-term memory, fully connected deep neural networks. In: ICASSP 2016. pp. 4580–4584 (2015)
18. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms (2017)
19. Zalando-Research: Comparison of classifiers on MNIST and fashion-MNIST datasets. <http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/>, accessed: 2019-09-20

# Efficient Algorithm for Set-Valued Prediction in Multi-Class Classification

Thomas Mortier<sup>1</sup>, Marek Wydmuch<sup>2</sup>, Krzysztof Dembczyński<sup>2</sup>, Eyke Hüllermeier<sup>3</sup>, and Willem Waegeman<sup>2</sup>

<sup>1</sup> Department of Data Analysis and Mathematical Modelling, Ghent University, Belgium [{thomasf.mortier, willem.waegeman}@ugent.be](mailto:{thomasf.mortier, willem.waegeman}@ugent.be)

<sup>2</sup> Institute of Computing Science, Poznań University of Technology, Poland [{mwydmuch, kdembczynski}@cs.put.poznan.pl](mailto:{mwydmuch, kdembczynski}@cs.put.poznan.pl)

<sup>3</sup> Intelligent Systems and Machine Learning, Universität Paderborn, Germany [eyke@upb.de](mailto:eyke@upb.de)

**Abstract.** In cases of uncertainty, a multi-class classifier preferably returns a set of candidate classes instead of predicting a single class label with little guarantee. More precisely, the classifier should strive for an optimal balance between the correctness (the true class is among the candidates) and the precision (the candidates are not too many) of its prediction. We formalize this problem within a general decision-theoretic framework that unifies most of the existing work in this area. In this framework, uncertainty is quantified in terms of conditional class probabilities, and the quality of a predicted set is measured in terms of a utility function. We address the problem of finding the Bayes-optimal prediction, i.e., the subset of class labels with the highest expected utility. For this problem, which is computationally challenging as there are exponentially (in the number of classes) many predictions to choose from, we propose an efficient algorithm that can be applied to a broad family of utility scores.

**Keywords:** Multi-class Classification · Set-Valued Prediction

## 1 Introduction

In probabilistic multi-class classification, one often encounters situations in which the classifier is uncertain about the class label for a given instance. In such cases, instead of predicting a single class, it might be beneficial to return a set of classes as a prediction, with the idea that the correct class should at least be contained in that set. For example, in medical diagnosis, when not being sure enough about the true disease of a patient, it is better to return a set of candidate diseases. The set that is sufficiently small compared to the total number of diagnoses can be a great help for a medical doctor, because only the remaining candidate diseases need further investigation.

Formally, we assume training examples  $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$  are drawn from a distribution  $P(\mathbf{x}, y)$  on  $\mathcal{X} \times \mathcal{Y}$ , with  $\mathcal{X}$  some instance space (e.g., images, documents,

etc.) and  $\mathcal{Y} = \{c_1, \dots, c_K\}$  an output space consisting of  $K$  classes. In the context of set-valued prediction, we will consider a prediction  $\hat{Y}$  from the power set of  $\mathcal{Y}$ , i.e., predictions are (non-empty) subsets of  $\mathcal{Y}$ , or more formally,  $\hat{Y} \in 2^{\mathcal{Y}} \setminus \{\emptyset\}$ . The quality of the prediction  $\hat{Y}$  can be expressed by means of a set-based utility score  $u(c, \hat{Y})$ , where  $c$  corresponds to the ground-truth class and  $\hat{Y}$  is the predicted set.

Multi-class classifiers that return set-valued predictions have been considered by several authors under different names [1,2,3,4,5]. In this paper, we introduce Bayes-optimal algorithms for maximizing set-based utility scores  $u(c, \hat{Y})$ . To this end, we will consider a decision-theoretic framework, where we estimate a probabilistic model, followed by an inference procedure at prediction time. In a probabilistic multi-class classification framework, we estimate the conditional probability distribution  $P(\cdot | \mathbf{x})$  over  $\mathcal{Y}$ :

$$\forall c \in \mathcal{Y} : 0 \leq P(c | \mathbf{x}) \leq 1, \quad \sum_{c \in \mathcal{Y}} P(c | \mathbf{x}) = 1. \quad (1)$$

This distribution can be estimated using several types of probabilistic methods. At prediction time, the goal is to find the Bayes-optimal solution  $\hat{Y}_u^*$  by expected utility maximization:

$$\hat{Y}_u^*(\mathbf{x}) = \arg \max_{\hat{Y} \in 2^{\mathcal{Y}} \setminus \{\emptyset\}} \mathbb{E}_{P(c | \mathbf{x})}[u(c, \hat{Y})] = \arg \max_{\hat{Y} \in 2^{\mathcal{Y}} \setminus \{\emptyset\}} \sum_{c \in \mathcal{Y}} u(c, \hat{Y}) P(c | \mathbf{x}). \quad (2)$$

In the remaining we will use the shorthand notation  $U(\hat{Y}, P, u)$  for the expected utility  $\sum_{c \in \mathcal{Y}} u(c, \hat{Y}) P(c | \mathbf{x})$ . The above optimization problem is non-trivial, as a brute-force search requires checking all subsets of  $\mathcal{Y}$ , resulting in an exponential time complexity. However, we will be able to find the Bayes-optimal prediction more efficiently.

## 2 Utility scores for set-valued prediction

The inference algorithms that we introduce can be applied to a general family of set-based utility functions  $u : \mathcal{Y} \times 2^{\mathcal{Y}} \setminus \{\emptyset\} \rightarrow [0, 1]$  given as follows:

$$u(c, \hat{Y}) = \begin{cases} 0 & \text{if } c \notin \hat{Y}, \\ g(|\hat{Y}|) & \text{if } c \in \hat{Y}, \end{cases} \quad (3)$$

where  $|\hat{Y}|$  denotes the cardinality of the predicted set  $\hat{Y}$ . This family is parametrized by a sequence  $(g(1), \dots, g(K)) \in [0, 1]^K$  that should obey the following properties:

1.  $g(1) = 1$ , i.e., the utility  $u(c, \hat{Y})$  should be maximal when the classifier returns the true class label as a singleton set;
2.  $g(s)$  should be non-increasing, i.e., the utility  $u(c, \hat{Y})$  should be higher if the true class is contained in a smaller set of predicted classes;
3.  $g(s) \geq 1/s$ , i.e., the utility  $u(c, \hat{Y})$  of predicting a set containing the true and  $s - 1$  additional classes should not be lower than the expected utility of randomly guessing one of these  $s$  classes. This requirement formalizes the idea of risk-aversion: in the face of uncertainty, abstaining should be preferred to random guessing (see e.g. [4]).

### 3 The main result

In this section, we present our theoretical results along with an efficient algorithm that is based on these results. The formulation in (2) seems to suggest that all subsets of  $\mathcal{Y}$  need to be analyzed to find the Bayes-optimal solution, but our first result shows that this is not the case.

**Theorem 1.** *The exact solution of (2) can be computed by analyzing only  $K$  subsets of  $\mathcal{Y}$ .*

*Proof.* With  $P(\hat{Y} | \mathbf{x}) = \sum_{c \in \hat{Y}} P(c | \mathbf{x})$ , the expected utility can be written as

$$\begin{aligned} U(\hat{Y}, P, u) &= \sum_{c \in \mathcal{Y}} u(c, \hat{Y}) P(c | \mathbf{x}) = \sum_{c \in \hat{Y}} u(c, \hat{Y}) P(c | \mathbf{x}) + \sum_{c' \notin \hat{Y}} u(c', \hat{Y}) P(c' | \mathbf{x}) \\ &= \sum_{c \in \hat{Y}} g(|\hat{Y}|) P(c | \mathbf{x}) = g(|\hat{Y}|) P(\hat{Y} | \mathbf{x}), \end{aligned} \quad (4)$$

where the last summation in the second equality cancels out since  $u(c', \hat{Y}) = 0$ . Let us decompose (2) into an inner and an outer maximization step. The inner maximization step then becomes

$$\hat{Y}_u^{*s} = \arg \max_{|\hat{Y}|=s} g(s) P(\hat{Y} | \mathbf{x}) = \arg \max_{|\hat{Y}|=s} P(\hat{Y} | \mathbf{x}), \quad (5)$$

for  $s = \{1, \dots, K\}$ . This step can be done very efficiently, by sorting the conditional class probabilities, as for a given  $s$ , only the subset with highest probability needs to be considered. The outer maximization simply consists of computing  $\hat{Y}_u^*(\mathbf{x}) = \arg \max_{\hat{Y} \in \{\hat{Y}_u^{*1}, \dots, \hat{Y}_u^{*K}\}} g(|\hat{Y}|) P(\hat{Y} | \mathbf{x})$ , which only requires the evaluation of  $K$  sets.  $\square$

So, one only needs to evaluate  $\hat{Y}_u^{*1}, \dots, \hat{Y}_u^{*K}$  to find the Bayes-optimal solution, which already limits the search to  $K$  subsets. We can do even better. By restricting  $g$ , we can assure that the sequence  $U(\hat{Y}_u^{*1}, P, u), \dots, U(\hat{Y}_u^{*K}, P, u)$  is unimodal. The restriction required for  $g$  is  $(1/x)$ -convexity, i.e., convexity after a  $(1/x)$  transformation.

**Definition 1.** *A sequence  $g(1), g(2), \dots, g(K)$  is  $(1/x)$ -convex if*

$$\frac{1}{g(s+1)} \leq \frac{1/g(s) + 1/g(s+2)}{2} \quad \text{for all } s \in \{1, \dots, K-2\}. \quad (6)$$

**Theorem 2.** *Let  $g(1), g(2), \dots, g(K)$  be a  $(1/x)$ -convex sequence and for a given  $s \in \{1, \dots, K-1\}$  let  $U(\hat{Y}_u^{*s}, P, u) > U(\hat{Y}_u^{*s+1}, P, u)$ . Then  $U(\hat{Y}_u^{*s}, P, u) > U(\hat{Y}_u^{*s+i}, P, u)$  for all  $i \in \{1, \dots, K-s\}$ .*

Assuming that during inference one can query the conditional class distribution  $P$  for a given  $\mathbf{x}$ , we can combine Theorems 1 and 2 to get an efficient inference procedure presented in Algorithm 1.

**Algorithm 1 input:**  $g(\cdot)$ ,  $\mathbf{x}$ ,  $\mathcal{Y} = \{c_1, \dots, c_K\}$ ,  $P$ 


---

```

1:  $\hat{Y} \leftarrow \emptyset$ ,  $p_{\hat{Y}} \leftarrow 0$ ,  $U^* \leftarrow 0$   $\triangleright$  Initialize the best solution, its probability and utility
2:  $\mathcal{Q} \leftarrow \emptyset$   $\triangleright$  Initialize a priority queue that sorts decreasingly the classes by  $P(c|\mathbf{x})$ 
3: for  $c \in \mathcal{Y}$  do  $\triangleright$  For all classes
4:    $p_c \leftarrow P(c|\mathbf{x})$ ,  $\mathcal{Q}.\text{add}((c, p_c))$   $\triangleright$  Query the distribution  $P$  to get  $P(c|\mathbf{x})$ 
5:  $\mathcal{Q}.\text{sort}()$   $\triangleright$  Sort the list decreasingly according  $P(c|\mathbf{x})$ 
6: while  $\mathcal{Q} \neq \emptyset$  do  $\triangleright$  Loop until the list of sorted classes is empty
7:    $(c, p_c) \leftarrow \mathcal{Q}.\text{pop}()$   $\triangleright$  Pop the first element from  $\mathcal{Q}$ 
8:    $\hat{Y} \leftarrow \hat{Y} \cup \{c\}$ ,  $p_{\hat{Y}} \leftarrow p_{\hat{Y}} + p_c$   $\triangleright$  Update the current solution and its probability
9:    $U_{\hat{Y}} \leftarrow p_c \times g(\hat{Y})$   $\triangleright$  Compute  $U(\hat{Y}, P, u)$  according to Eq. (4)
10:  if  $U^* \leq U_{\hat{Y}}$  then  $\triangleright$  If the current utility is greater than the best solution so far
11:     $\hat{Y}_u^* \leftarrow \hat{Y}$ ,  $U^* \leftarrow U_{\hat{Y}}$   $\triangleright$  Replace the current best solution
12:  else  $\triangleright$  If there is no improvement
13:    break  $\triangleright$  break the while loop according to Theorem 2
14: return  $\hat{Y}_u^*$   $\triangleright$  Return the set of classes with the highest utility

```

---

We start by obtaining conditional class probabilities  $P(c|\mathbf{x})$  and sorting them in decreasing order. Then, the algorithm computes  $U(\hat{Y}_u^{*1}, P), \dots, U(\hat{Y}_u^{*s}, P)$  in a sequential way. The class with  $s$ -highest conditional class probability is added to the predicted set obtained in the previous steps, till the stopping criterion of Theorem 2 is satisfied. It is easy to notice that Algorithm 1 finds the Bayes-optimal solution  $\hat{Y}_u^*$  in quasilinear time in the number of classes.

## 4 Conclusion

We introduced a decision-theoretic framework for a general family of set-based utility functions and developed Bayes-optimal inference algorithm that exploit specific assumptions to improve runtime efficiency.

## References

1. Juan José Del Coz, Jorge Díez, and Antonio Bahamonde. Learning nondeterministic classifiers. *The Journal of Machine Learning Research*, 10:2273–2293, 2009.
2. Giorgio Corani and Marco Zaffalon. Learning reliable classifiers from small or incomplete data sets: The naive credal classifier 2. *Journal of Machine Learning Research*, 9:581–621, 2008.
3. Giorgio Corani and Marco Zaffalon. Lazy naive credal classifier. In *Proceedings of the 1st ACM SIGKDD Workshop on Knowledge Discovery from Uncertain Data*, U '09, pages 30–37, New York, NY, USA, 2009. ACM.
4. Marco Zaffalon, Corani Giorgio, and Denis Deratani Mauá. Evaluating credal classifiers by utility-discounted predictive accuracy. *Int. J. Approx. Reasoning*, 53:1282–1301, 2012.
5. G. Yang, S. Destercke, and M. Masson. The costs of indeterminacy: How to determine them? *IEEE Transactions on Cybernetics*, 47:4316–4327, 2017.

# Multi-classifier system based on center of mass classifier

Szymon Wojciechowski<sup>[0000–0002–8437–5592]</sup>

Department of Systems and Computer Networks,  
Wrocław University of Science and Technology,  
Wrocław, Poland  
[szymon.wojciechowski@pwr.edu.pl](mailto:szymon.wojciechowski@pwr.edu.pl)

**Keywords:** Machine Learning · Classification · Multi-classifier · Center of Mass Classifier

This paper is introducing new algorithm for solving classification problem - center of mass classifier. The idea is based on data distribution and its representation, which also comes with some design limitations. To overcome them, classifier can be used in a multi-classifier system, which will be a subject of research in this paper.

The foundation for the algorithm is a geometrical center of mass calculated in two-dimensional subspace of original problem instance. The corresponding angles between each point of learning set and center of mass are used to create their distribution histogram, which later on provides classification results.

The center of mass classifier is assuming that samples are placed in two-dimensional space, which makes it impossible to apply on most of the known datasets. However, by creating homogeneous ensemble of center of mass classifiers following subspace division pool generation schema, it is possible to extend this method for most of the datasets. This paper is considering a random selection of subspaces and majority voting classifiers integration.

The experiments were conducted on well-known datasets available online in *KEEL* repository. Those were selected to include only binary problems without nominal features. The goal of the experiments was to verify if using center of mass classifier as a base classifier will provide better results the other reference algorithms, which are *Gaussian Naive Bayes*, *K-Nearest Neighbours*, *Support Vector Machines*.

In three datasets: *heart*, *monk-2* and *spambase* it was observed, that center of mass classifier have provided better results, especially in first case where accuracy improvement was significant. In *wisconsin* dataset center of mass classifier also outperformed other methods, however Wilcoxon signed-rank test shown that there is results correlation with K-nearest neighbors classifier.

This work is supported by the statutory funds of the Department of Systems and Computer Networks, Faculty of Electronics, Wrocław University of Science and Technology.

# **Concept of Research into Cognitive Load in Human-Computer Interaction Using Biometric Techniques**

Patient Zihisire Muke [0000-0001-7860-5067], Bogdan Trawiński [0000-0002-2956-6388]

Wrocław University of Science and Technology, Department of Information Systems, Wrocław, Poland  
`{patient.zihisire, bogdan.trawinski}@pwr.edu.pl`

**Abstract.** This short paper presents the concept of research that will be carried out to prepare a doctoral dissertation at the Wrocław University of Science and Technology. Interdisciplinary study combining the fields of psychology and computer science is planned. The problems of forming and controlling cognitive load in human-computer interfaces will be addressed. The methods for measuring cognitive load using multimodal biosensor techniques will be devised. The machine learning models for prediction of cognitive load when using web and mobile applications will be constructed. The proposed methods and models will be verified experimentally by usability testing enhanced with biometric devices.

**Keywords:** cognitive load, user experience, human computer interfaces, web applications, mobile applications, machine learning

## **1 Introduction**

The human cognitive system is under enormous pressure today with the augmentation of user interface complexity in interactive systems. Cognitive Load Theory (CLT) has a very crucial role in Human-Computer Interaction (HCI). Users generally, while performing tasks on interactive systems such as websites or app have one goal in mind, to get the tasks done in the less period of time possible. By the time it takes an extravagant amount of cognitive load to execute the tasks, the brain starts to slowing down, provoking the user to feel depressed. Because the interface is overwhelming their brain with information, the only solution to escape this is by restarting the system. That will leave users confused and exasperated. It is very clear that a lot is still required to be done in this area. To overcome the above stated problems, the objective is to improve user experience by minimizing the level of cognitive load when the user manipulates or uses interactive systems (websites or mobile applications).

Since the HCI systems are being utilized to perform critical tasks in different domains and are becoming more ubiquitous, the necessity to measure the cognitive load generated by a HCI system designed for a given goal, is becoming more and more important before using them [Zagermann et al. 2016]. The current literature in the field of HCI indicates that several researchers have developed instruments using traditional as well as biosensor methods with machine learning techniques for cognitive load measurement in the HCI context. Therefore, the research carried out during the doctoral

dissertation will propose prediction models to enhance usability testing of interactive systems.

## 2 Related Works

John Sweller an educational psychologist was the first who described the concept cognitive load [Sweller 1988] , [Sweller 1994]. He relied on a memory employed model which outlined that long-term memory can be established when visual and auditory information is processed (or reproduced) at a larger scale than other daily observations. Sweller felt that the factors that make learning so difficult or distracting us from the information we are trying to pay attention to, increase the cognitive load of the person during treatment of that information by the memory. Because of high cognitive load, it is more difficult to focus on practice and memorize stimuli, and that make learning less efficient.

One way to determine the problems of a website is to measure the level of cognitive load of the user. Cognitive load theory (CLT) is a pedagogical theory that outlines the occurrence of best way to learn when instructional materials are designed in the way to correspond to the human cognitive function. Human cognition comprises a mixture of working memory, which has limited capacity of information processing time, and long-term memory, which offers unlimited storage capacity for high automaticity and cognitive schemes [Hadie et al. 2016].

In accordance with [Zu et al. 2017], objective methods such as eye movement measures and electroencephalography (EEG) have been used to measure the total cognitive load. Very few research studies, if any, have been completed to measure the three kinds of load (intrinsic, extraneous and germane) separately with physiological methods in a continuous manner. In their study, they have shown several eye-tracking based parameters which are related to the three kinds of load by having explicit manipulation of the three loads independently.

In addition, [Zagermann et al. 2016] encourage the use of eye tracking measurements (voluntary eye-movements like fixations and saccades and involuntary eye-movements like pupil dilation and blinking) to investigate users' cognitive load while interacting with a system.

In the same context, [Marquart et al. 2015] summarize the results of recently conducted studies about the relation between eye measurement parameters and driver mental workload. The study reports that various eye activity measures including blinks, fixations, and saccades were previously researched and confirmed as useful estimates of a driver's mental workload.

Besides that, [Kumar et al. 2016] propose a physiological methodology to measure cognitive load using EEG power spectrum. Physiological measurement using EEG tool for cognitive load would provide an objective measure of mental activities as the EEG gives indication of mental resources spent in a task through spectral power of the signals collected from the scalp of the user during task performance.

[Cengizhan et al. 2011] review psychophysiological measures applied in Human Computer Interaction (HCI) focusing on studies related to human cognitive states. The study mentions that there are numerous types of psychophysiological measures in the

literature such as electroencephalogram (EEG), heart rate variability (HRV) and electrodermal activity (EDA).

[Anderson et al. 2011] describe extracting cognitive load measures from EEG data, and show how those measures are used to quantitatively evaluate the effectiveness of visualizations.

Furthermore, [Tracy 2006] and [Albers 2011] examine three simple methods for measuring cognitive load, namely Sternberg memory task, tapping task, and the NASA TLX. They also mention that techniques measuring physical responses such as EEG and pulse rates are accurate, but are also expensive and require special equipment and training.

[Krejtz et al. 2018] with the use of an eye tracker, test two metrics related to the task difficulty, namely the change in pupil diameter with respect to inter- or intra-trial baseline, and the rate and magnitude of microsaccades.

### 3 Problem formulation and research tasks

Cognitive load has been studied for above three decades. Especially extensive investigations into this phenomenon using multimodal biosensor techniques have been conducted for the last five years. The analyses have been conducted using eye tracking methods [Wang et al. 2014], [Vogels et al. 2018], [Demberg 2013], [Liu et al. 2016], electroencephalography (EEG) [Friedman et al. 2019], [Antonenko et al. 2010], respiration and heart rate [Nakamura et al. 2018], [Charles and Nixon 2019], as well as facial expression analysis [Hussain et al. 2014], [Ahmed 2018].

The main goal of the study is to find answers to the following research questions:

- How to explore cognitive load in the area of human-computer interactions?
- How to measure cognitive load in the area of human-computer interactions?
- How to assess the impact of cognitive load on the usability of web and mobile applications?
- How to form and control cognitive load in human-computer interfaces?
- How to design web and mobile applications to achieve an acceptable level of cognitive load?
- How to evaluate individual design patterns and other components of web and mobile applications in terms of cognitive load?
- How to build machine learning models to predict cognitive load when using user interfaces in web and mobile applications?

In order to tackle the aforementioned problems several following research tasks should be completed:

- Literature study into the cognitive load in the context of human-computer interactions and user experience (UX) of web and mobile applications.
- Conducting comparative analysis of classic UX research methods with methods enhanced with biometric sensors.
- Conducting comparative analysis of individual biosensors, how they contribute into better detection and understanding of UX problems.

- Collecting and categorizing tasks to complete by the participants during experiments: brain teasers, puzzles, riddles, IQ tests, data entry controls, navigation controls, search engines, etc.
- Collecting and assessing metrics for evaluation of UX of user interfaces in web and mobile applications considering the impact of cognitive workload.
- Working out and selection of metrics to measure cognitive load using multimodal biosensor techniques.
- Working out methods for measuring cognitive load using multimodal biosensor techniques.
- Working out methods for forming and controlling cognitive workload in web and mobile applications.
- Constructing machine learning models for prediction of cognitive load when using web and mobile applications.
- Working out plans of experiments to evaluate proposed metrics and methods.
- Conducting evaluation experiments using multimodal biometric methods.
- Analysing experimental results using statistical tests of significance.
- Formulating the recommendations for user interface designers which reduce/minimize cognitive workload.

The research will be conducted using the iMotions Platform which integrates and synchronizes signals from several biosensors (see Figures. 1 and 2).



**Fig. 1.** Schema of iMotions platform integrating and synchronizing biosensors



**Fig. 2.** Research stand to conduct UX experiments with iMotions Platform

The iMotions Platform provides integration and synchronization of above 50 biosensors such as eye tracking, facial expression analysis, electrodermal activity / galvanic skin response, EEG, EMG, and ECG hardware. It supports the researchers by presenting various stimuli such as images, videos, websites, games, mobile applications, software, and VR environments. It visualizes recordings both in real-time during experiments and replays the sessions on demand. It enables the researchers to annotate the data live and after its collection as well as to export raw data, results, and metrics into different file formats [iMotions 2017], [iMotions 2018], [iMotions 2019].

## 4 Conclusions

The concept of research into cognitive load in the field of human-computer interaction is presented in this short paper. The application of the following biometric methods: eye tracking, facial expression analysis, galvanic skin response, and EEG will allow for deeper insight into this phenomenon. A set of research problems was defined including forming and controlling cognitive load in human-computer interfaces. The methods to measure cognitive load with multimodal biosensor techniques will be also devised. The machine learning models for prediction of cognitive load when using web and mobile applications will be built. The experimental usability testing enhanced with biometric devices is planned to evaluate the proposed methods and models.

## References

- [Ahmed 2018] Ahmed, L.: Knowing how you are feeling depends on what's on my mind: Cognitive load and expression categorization. *Emotion* 18(2), 190-201 (2018). DOI: <http://dx.doi.org/10.1037/emo0000312>
- [Albers 2011] Albers, M.: Tapping as a Measure of Cognitive Load and Website Usability. *Proceedings of the 29th ACM International Conference on Design of Communication, SIGDOC '11*, pp. 25-32 (2011). DOI: <https://doi.org/10.1145/2038476.2038481>
- [Antonenko et al. 2010] Antonenko, P., Paas, F., Grabner, R., van Gog, T.: Educational Psychology Review 22(4), 425–438 (2010). DOI: <https://doi.org/10.1007/s10648-010-9130-y>
- [Cengizhan et al. 2011] Cengizhan, A., Göktürk, M.: Psychophysiological Measures of Human Cognitive States Applied in Human Computer Interaction. *Procedia Computer Science* 3, 1361–1367 (2011). DOI: <https://doi.org/10.1016/j.procs.2011.01.016>
- [Charles and Nixon 2019] Charles, R.L., Nixon, J.: Measuring mental workload using physiological measures: A systematic review. *Applied Ergonomics* 74, 221-232 (2019). DOI: <https://doi.org/10.1016/j.apergo.2018.08.028>
- [Demberg 2013] Demberg, V.: Pupilometry: the Index of Cognitive Activity in a dual-task study. *Proceedings of the 35th Annual Meeting of the Cognitive Science Society CogSci-13* (2013). Retrieved from <https://escholarship.org/uc/item/4vf5w6bn>
- [Friedman et al. 2019] Friedman, N., Fekete, T., Gal, K., Shriki, O.: EEG-Based Prediction of Cognitive Load in Intelligence Tests. *Frontiers in Human Neuroscience* 13, p. 191 (2019). DOI: <https://doi.org/10.3389/fnhum.2019.00191>

- [Hadie et al. 2016] Hadie, S. N. H., Yusoff, M. S. B.: Assessing the validity of the cognitive load scale in a problem-based learning setting. *Journal of Taibah University Medical Sciences* 11(3), 194–202 (2016). DOI: <https://doi.org/10.1016/j.jtumed.2016.04.001>
- [Hussain et al. 2014] Hussain, S., Calvo, R., Chen, F.: Automatic Cognitive Load Detection from Face, Physiology, Task Performance and Fusion During Affective Interference. *Interacting with Computers* 26(3), V256–268 (2014). DOI: <https://doi.org/10.1093/iwc/iwt032>
- [iMotions 2017] Human Behavior. Pocket Guide. iMotions (2017). Downloaded from <https://imotions.com/> on June 20, 2019
- [iMotions 2018] iMotions Biometric Research Platform. Software Modules. iMotions (2018). Downloaded from <https://imotions.com/> on June 20, 2019
- [iMotions 2019] UX and Usability Research. Increase Insights with Biosensors. iMotions (2019). Downloaded from <https://imotions.com/> on June 20, 2019
- [Krejtz et al. 2018] Krejtz, K., Duchowski, A.T., Niedzielska, A., Biele, C., Krejtz, I.: Eye tracking cognitive load using pupil diameter and microsaccades with fixed gaze. *PLoS ONE* 13(9): e0203629 (2018). DOI: <https://doi.org/10.1371/journal.pone.0203629>
- [Kumar et al. 2016] Kumar, N., & Kumar, J.: Measurement of Cognitive Load in HCI Systems Using EEG Power Spectrum: An Experimental Study. *Procedia - Procedia Computer Science* 84, 70–78 (2016). DOI: <https://doi.org/10.1016/j.procs.2016.04.068>
- [Liu et al. 2016] Xin Liu, Tong Chen, Guoqiang Xie, and Guangyuan Liu: Contact-Free Cognitive Load Recognition Based on Eye Movement. *Journal of Electrical and Computer Engineering*, vol. 2016, Article ID 1601879 (2016). DOI: <https://doi.org/10.1155/2016/1601879>
- [Marquart et al. 2015] Marquart, G., Cabrall, C., Winter, J. De.: Review of eye-related measures of drivers' mental workload. *Procedia Manufacturing* 3, 2854–2861 (2015). DOI: <https://doi.org/10.1016/j.promfg.2015.07.783>
- [Nakamura et al. 2018] Nakamura, N.H., Fukunaga, M., Oku, Y.: Respiratory modulation of cognitive performance during the retrieval process. *PLoS ONE* 13(9), e0204021 (2018). DOI: <https://doi.org/10.1371/journal.pone.0204021>
- [Sweller 1988] Sweller, J.: Cognitive load during problem solving. Effects on learning. *Cognitive Science* 12, 257–285 (1988). DOI: [10.1207/s15516709cog1202\\_4](https://doi.org/10.1207/s15516709cog1202_4)
- [Sweller 1994] Sweller, J.: Cognitive load theory, learning difficulty, and instructional design. *Learng and Instruction* 4(4), 293–312 (1994)
- [Tracy 2006] Tracy, J.P., Albers, M.J.: Measuring Cognitive Load to Test the Usability of Web Sites. In *Usability and Information Design*, pp. 256–260 (2006)
- [Vogels et al. 2018] Vogels, J., Demberg, V., Kray, J.: The Index of Cognitive Activity as a Measure of Cognitive Processing Load in Dual Task Settings. *Frontiers in Psychology* 9, p. 2276 (2018). DOI: <https://doi.org/10.3389/fpsyg.2018.02276>
- [Wang et al. 2014] Wang, Q., Yang, S., Liu, M., Cao, Z., Ma, Q.: An eye-tracking study of website complexity from cognitive load perspective. *Decision Support Systems* 62, 1–10 (2014). DOI: <https://doi.org/10.1016/j.dss.2014.02.007>
- [Zagermann et al. 2016] Zagermann, J., Pfeil, U., Reiterer, H.: Measuring Cognitive Load using Eye Tracking Technology in Visual Computing. Proceedings of the Sixth Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization, BELIV '16, pp. 78–85, ACM Press (2016). <https://dx.doi.org/10.1145/2993901.2993908>
- [Zu et al. 2017] Zu, T., Hutson, J., Loschky, L.C., Rebello, N.S.: Use of Eye-Tracking Technology to Investigate Cognitive Load Theory. 2017 Physics Education Research Conference, pp. 472–475 (2017). DOI: <https://doi.org/10.1119/perc.2017.pr.113>

# Investigating initialization method in the process of facial landmarks detection in thermal images

Anton Smoliński and Paweł Forczmański

West Pomeranian University of Technology, Szczecin,  
Faculty of Computer Science and Information Technology,  
Żołnierska Str. 52, 71–210 Szczecin, Poland  
`{ansmolinski,pforczmansi}@wi.zut.edu.pl`

**Abstract.** In the paper we present a problem of face detection and alignment in thermal imagery. We focus on the initialization stage as a key element of further processing. The experiments involved two face detectors and manual face cropping and were performed on our own benchmark database consisting of images from visible-band and thermal cameras. The analysis of the results shows some directions in the further research, mainly associated with precise pose-invariant face detection.

## 1 Introduction

Human face is one of the most evident biometric features, easy to capture, discern and identify. There are still situations when typical biometric approaches are not enough, e.g. when environmental conditions are not fully controlled or there is a need of increased security level. Such problems could be solved by thermal imaging [1], since images registered by infrared or thermal sensors are independent from a proper illumination of the subject.

In the paper we focus on feature extractors and predictors aimed at thermal facial portrait alignment, making possible to develop algorithms aimed at emotion/state estimation.

Although the problems of human face detection and recognition in visible light have been investigated many times, the problems of detecting and recognizing faces in thermal spectrum are infrequent [1].

The typical algorithm of facial landmarks localization, no matter if it works in visible or thermal band, consists of two stages: face detection and localization (called initialization) and facial landmarks prediction (called face alignment). In most cases, face localization is performed using general purpose detector, e.g Viola-Jones detector based on Haar-like features, Histogram of Oriented Gradients or Local Binary Patterns [8]. There are also some other successful approaches, like HOG+SVM [4] or CNN-bases ones [7]. The other stage employs mostly cascades of regressors, active shape (appearance) models or also deep-learning algorithms.

In this research we apply two algorithms of face detection, namely VJ-based approach with Haar-like features [8], and HOG+SVM detector [4]. At the stage

of face alignment we employ an algorithm that performs fast face alignment and achieves accuracy superior or comparable to state-of-the-art methods on standard datasets [3]. It is based on an ensemble of randomized regression trees used to detect 194 landmarks on face from a single image that performs shape invariant feature selection while minimizing the same loss function during training time. We used an implementation that can be found in Deep Learning Library (dlib), that uses 68 facial landmarks. Our focus is put on the initialization stage and its influence on the accuracy of face alignment. Although there are a few works aimed at precise facial landmark detection in thermal images (e.g. [5]), they are mainly focused on the alignment itself, not taking into consideration initialization stage.

## 2 Experiments

### 2.1 Experimental setup

We evaluated the accuracy of face alignment on our own WIZUT database [2] containing portraits captured in both visible and thermal spectra. Collected faces include full frontal and rotated portraits, some of them include glasses. The thermal images are normalized in terms of temperature range. The properties of data are given in Tab. 1.

Table 1: Benchmark dataset characteristics

|                            |               |
|----------------------------|---------------|
| No. image pairs/subjects   | 535 /101      |
| Image width x height [pix] | 320 × 240     |
| Min. face size [pix]       | 119 × 124     |
| Max. face size [pix]       | 182 × 178     |
| Rotation angle [°]         | {0, ±20, ±45} |

The experiments were aimed at evaluating the face localization methods taking into consideration facial landmark prediction. In each experiment we tested 3 approaches of initialization (face localization): a cascading classifier based on Haar-like features (Haar-Thr) trained on the subset of thermal facial images, a detector based on Histogram of Oriented Gradients joint with SVM trained on visible-band facial portraits (HOG-Vis) and a manual marking of faces in the images (Manual). It should be noted that the HOG-SVM detector worked on visible-band images and the detection results were transferred on thermal images. The first classifier was implemented using Open Computer Vision library (OpenCV), while the second one using Deep Learning Library (dlib).

### 2.2 Accuracy Evaluation

We assumed the landmarks found by the predictor in visible band images to be the ground-truth. We used a visible-band predictor included in dlib package.

For thermal images we used our own-trained landmark predictor. We calculated a mean distance between the localized landmarks in thermal images and the ground truth landmarks divided by the inter-ocular distance (the distance between the outer eye corners). Exemplary results of face alignment for visible band and thermal spectrum are presented in Fig. 1. The quantitative results for all three methods of face localization and three groups of face rotations (frontal faces, faces rotated by 20° and faces rotated by 45°) are presented in Tab. 2.

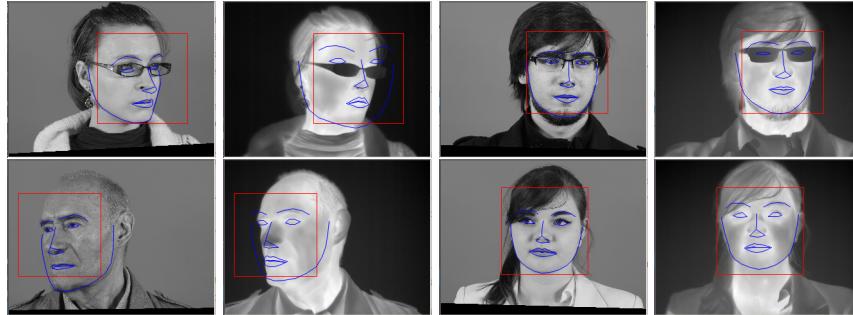


Fig. 1: Exemplary cropped and aligned images.

Table 2: Model accuracy - mean errors normalized by inter-ocular distance

| Initialization | Rotation 0° |                |                | Rotation ±20° |                |                | Rotation ±45° |                |                |
|----------------|-------------|----------------|----------------|---------------|----------------|----------------|---------------|----------------|----------------|
|                | Mean error  | Mean min. err. | Mean max. err. | Mean error    | Mean min. err. | Mean max. err. | Mean error    | Mean min. err. | Mean max. err. |
| Haar-Thr       | 40.75       | 19.55          | 61.83          | 42.50         | 18.61          | 64.02          | 64.79         | 21.27          | 93.07          |
| HOG-Vis        | 10.68       | 2.42           | 22.82          | 10.53         | 2.32           | 22.43          | 12.34         | 2.42           | 26.66          |
| Manual         | 10.39       | 3.57           | 19.86          | 10.11         | 2.48           | 21.10          | 16.78         | 3.22           | 29.97          |

The mean errors were calculated over all 68 landmarks and all testing images. The analysis of the results unveils that HOG-based face detector transferred on thermal images gives similar results to the manual initialization. On the other hand, Haar-based detector fails in many cases. It is caused by a very tight frame around the face returned by this detector. In such case, the predictor often can not find features which are outside it. It can be seen that the alignment accuracy depends on the rotation angle. For faces with no or slight rotation, the accuracy is higher. For larger rotation angles, the errors increase.

In order to check, if the face alignment depends on the certain facial features, we analysed the results in groups of points, belonging to the jaw, eyebrows, nose, eyes and mouth, respectively. The mean errors are presented in Tab. 3. In this case, the HOG-based initialization was used. As it can be seen, the central parts

Table 3: Mean errors for particular face fragments (normalized)

| Jaw       |            | Eyebrows |       | Nose  |          | Eye   |       | Mouth |      |
|-----------|------------|----------|-------|-------|----------|-------|-------|-------|------|
| left side | right side | left     | right | nose  | nostrils | left  | right | mouth | lips |
| 15.06     | 15.01      | 19.87    | 18.98 | 12.99 | 9.85     | 16.51 | 15.33 | 9.26  | 9.04 |
|           |            | 15.04    |       | 19.43 |          | 11.42 |       | 15.92 |      |

of face, namely mouth and nose can be aligned with the lowest error, while peripheral parts (jaw and eyebrows) - with the highest, respectively. The main problem with an alignment of eyes lies in the presence of glasses, which cover eyes area in the thermal imaging.

### 3 Summary

The results of the experiments show that presented classifiers/predictors can be applied to face alignment in thermal band images. The comparison with other methods working on images taken in visible band shows (e.g. DAN [6]) that the accuracy of currently used facial landmarks detectors, based on cascaded regressors, is quite acceptable, yet it could be improved. The future works will be focused on applying other facial alignment methods to this task.

### References

1. Ghiass, R.S., Arandjelovic, O., Bendada, H., Maldague, X.: Infrared face recognition: a literature review, International Joint Conference on Neural Networks (cs.CV) arXiv:1306.1603 [cs.CV] (2013)
2. Jasiński, P.; Forczmański, P.: Combined Imaging System for Taking Facial Portraits in Visible and Thermal Spectra, Image Processing and Communications Challenges 7, AISC vol. 389 pp. 63–71 (2016)
3. Kazemi, V., Sullivan, J.: One Millisecond Face Alignment with an Ensemble of Regression Trees, IEEE Conf. on Computer Vision and Pattern Recognition (2014)
4. King, D.E.: Max-Margin Object Detection, Computer Vision and Pattern Recognition (cs.CV), arXiv:1502.00046 (2015)
5. Kopaczka, M., Acar, K., Merhof, D.: Robust Facial Landmark Detection and Face Tracking in Thermal Infrared Images using Active Appearance Models. Internat. Conf. on Computer Vision Theory and Applications, pp. 150-158 (2016)
6. Kowalski, M., Naruniec, J., Trzciński, T.: Deep Alignment Network: A Convolutional Neural Network for Robust Face Alignment. 2017 IEEE Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW) (2017)
7. Krizhevsky, A., Sutskever, I., Hinton, G.: Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems 25, pp. 1106–1114 (2012)
8. Viola, P., Jones, M.J.: Robust real-time face detection. Int. J. Comput. Vision 57(2), pp. 137–154 (2004)

# Single particle diffusion classification by deep learning

Patrycja Kowalek<sup>[0000–0002–0743–5125]</sup>, Hanna Loch-Olszewska, and Janusz Szwabiński<sup>[0000–0002–6992–3634]</sup>

Faculty of Pure and Applied Mathematics, Hugo Steinhaus Center,  
Wrocław University of Science and Technology

**Abstract.** Single Particle trajectories measured in microscopy experiments contain important information about dynamic processes undergoing in a range of materials including living cells and tissues. The ability to correctly classify the trajectories motion is the key point to understand the dynamics of single molecules in a living cell, its organization and function. In this paper, we present novel classification method based on convolutional neural networks to identify four types of diffusion. The main conclusion is that deep learning method, trained on artificial trajectories, provides good results for particle classification problem.

**Keywords:** single particle tracking · diffusion modes · deep learning.

## 1 Introduction

Single Particle Tracking (SPT) is a popular method for analyzing dynamic processes within living cells and intracellular particles organization [1]. Individual trajectories are obtained by tracking and recording a single particle with a high-speed camera. To follow the molecule, scientists tag particle with a fluorescent dye which produces light and consequently they obtain the image with the particle position. These recordings can be used to reconstruct the trajectories of the particles [1].

The cooperation of scientists from multiple fields like molecular biology, mathematics, biochemistry and biophysics provides new techniques of diffusion modelling used in SPT. The basic types of motion which are observed in SPT experiments are: normal diffusion [2], directed motion [3, 4], anomalous diffusion [5] and confined diffusion [6].

Over the past few years, several alternative machine learning approaches have been proposed to identify the type of motion of a single particle [6–8]. In general, the most common methods are based on mean square displacement (MSD) curves [4]. The biggest disadvantage of such attempts is the disability to extract unbiased features from short length trajectories therefore the produced classifiers could not be so accurate.

To deal with the limitation of MSD, we decided to analyze SPT using deep learning approach, which allows us to extract features from raw data. In that paper, we used convolutional neural networks (CNN) [9] to determine the particle movement type.

## 2 Method

The advantage of deep learning is that there is no need for any feature selection or extraction carried out by a human expert. The CNN can extract and transform the features due to its architecture of multiple layers. Detection of data is set during a series of convolutions and pooling operations.

## 3 Diffusion modes and synthetic data

All supervised methods need a set of data to train the model. We simulated 5000 two-dimensional trajectories for four types of each motion: normal, directed, anomalous and confined.

In case of real SPT data, we can expect some noises because the trajectories extraction is the combination of several methods (which are often biased), as well as it requires human preprocessing, and apparatus can cause an errors (electronic noise). We decided to add noise to the simulated trajectories to make them more authentic. The noise also allows to avoid overfitting in model.

## 4 Model learning

We decided to used *mcfly* [10], a deep learning library for time series processing. The simulations and learning process were carried out on a cluster of 24 CPUs (2.6 GHz each) with the total memory of 50 GB.

The set of synthetic data was split randomly into two subsets containing training and testing data with the proportion 70% and 30% respectively. Both subsets had equal proportions of the movement types. The input data (2D trajectory) had a fixed length.

To find the best model hyper-parameters, we used random search method on a smaller set of data. We looked for the proper hyper-parameters of (i) the learning rate, (ii) the regularization rate, (iii) the number of convolutional layers, (iv) the number of filters in each convolutional layer and (vi) the number of hidden nodes (in dense layers). Once the best parameters were found we trained the model again on the full data set.

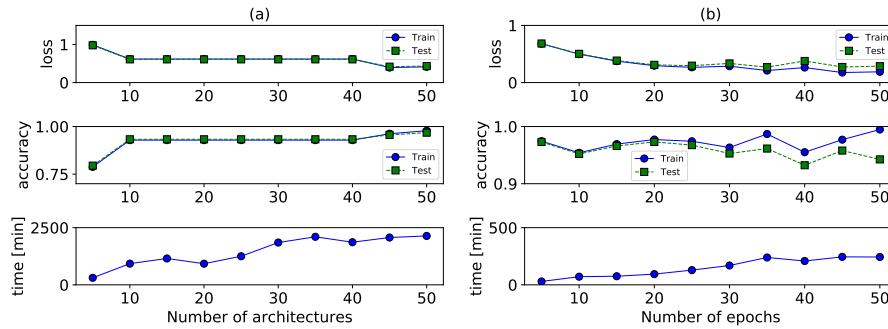
During searching for hyper-parameters, we established two constant parameters: number of epoch and searched architectures. Generally, many epochs are needed to achieve a combination of the weights in the network, which improves accuracy. Intuitively the bigger number of the searched models we have, the more accurate we are. This dependence is visible since more architectures cover a matrix of hyper-parameters. Increasing the values of these parameters has one big flaw which is the time of the calculations.

To prove the right choice of fixed parameters: epoch and architectures searched, we checked their impact on the loss and accuracy of the final model, the results are present in Fig. 1. The analysis shows the optimal number of architectures (left column) which was performed for 10 epochs. The decision for choosing the number of searches is relatively subjective as this is the compromise between the

accuracy, the loss, and the execution time. The differences in loss and accuracy for 10, 40, and 50 architectures are small but the time needed for searching is significantly bigger.

The number of epochs similarly affects the model loss and accuracy parameters. The variation between the accuracy of the train and the test sets starting from 30 epochs can be an indication of overfitting. In both analyses, we can see almost monotonic growth of the execution time. Given results convinced us to perform the best model for 20 architectures and 30 epochs.

Once we had the best parameters, we train our model and the execution time was 3 days 5 hours and 50 minutes.



**Fig. 1.** Impact of (a) the number of architectures in the random search and (b) the number of epochs in training the final model on the loss, accuracy and execution time.

## 5 Results

Once the CNN model is built, we were able to run the actual classification. Results are presented in table 1. It can be clearly seen that the classification turned out to be excellent. The model can easily detect direct and anomalous motions, although it performs slightly worse for the confined motion in terms of precision and for the normal diffusion in terms of recall. The overall accuracy of the model is 97%.

## 6 Conclusions

This paper shows the potential of Deep Learning methods in SPT classification problem. Machine learning techniques can recognize diffusion modes in terms of four types of motions. The problem of identifying the type of movement is a key point to describe the mechanical properties of the molecule.

The biggest advantage of deep learning approach is that it works with raw data and does not require human-described features. Presented method also

|               | precision | recall | support |
|---------------|-----------|--------|---------|
| anomalous     | 1.00      | 0.99   | 1500    |
| confined      | 0.91      | 1.00   | 1500    |
| directed      | 1.00      | 1.00   | 1500    |
| normal        | 0.99      | 0.90   | 1500    |
| average/total | 0.98      | 0.97   | 6000    |
| accuracy      | 97.30%    |        |         |

**Table 1.** A brief summary of the CNN classifier performance. All results are rounded to two decimal digits.

tends to be better than methods which use MSD curves [11]. CNN classifier gives the possibility to extract information from short trajectories which are commonly noticed in SPT experiments.

The research was performed mainly to prepare ourselves for further analysis based on real data. The challenge which we have to cope with is the fixed length of the input data. In the future, we want to develop other deep learning classifiers of SPT data and find the best approach to solve that problem.

**Acknowledgments:** P. Kowalek, H. Loch-Olszewska and J. Szwabiński were supported by NCN-DFG Beethoven Grant No. 2016/23/G/ST1/04083. Computations were performed on the BEM cluster in the Wrocław Center for Networking and Supercomputing (WCSS).

## References

1. C. Manzo and M. F. Garcia-Parajo, Reports on Progress in Physics 78, 124601 (2015).
2. S. B. Alves, G. F. O. Jr., L. C. Oliveira, T. P. de Silansa, M. Chevrollier, M. Ori, and H. L. S. Cavalcante, Physica A 447, 392 (2016).
3. G. Ruan, A. Agrawal, A. I. Marcus, and S. Nie, J. Am. Chem. Soc. 2007, 129, 47, 14759-14766 (2007), 10.1021/ja074936k.
4. X. Michalet, Physical Review E 82, 041914 (2010).
5. G. R. Kneller, Journal of Chemical Physics 141, 041105 (2014).
6. N. Monnier, S.-M. Guo, M. Mori, J. He, P. Lnrt, and M. Bathe, Biophysical Journal 103, 616 (2012).
7. P. Dosset, P. Rassam, L. Fernandez, C. Espenel, E. Rubinstein, E. Margeat, and P.-E. Milhiet, BMC Bioinformatics 17, 197 (2016).
8. T. Wagner, A. Kroll, C. R. Haramagatti, H.-G. Lipinski, and M. Wiemann, PLoS ONE 12(1), e0170165 (2017).
9. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, Proceedings of the IEEE 86, 2278 (1998).
10. D. van Kuppevelt, C. Meijer, V. van Hees, and M. Kuzak, "mcfly: time series classification made easy," (2017).
11. P. Kowalek, H. Loch-Olszewska, and J. Szwabiński, Physical Review E 100, 032410 (2019)

# Data imputation methods in classification task

Gaciarz Maciej<sup>1</sup> and Topolski Mariusz<sup>1</sup>

Wroclaw University Of Technology, Stanisawa Wyspianskiego 27 50-370 Wroclaw,  
Poland

**Abstract.** Missing data is frequent issue when it comes to classification task. Missing data has to be somehow dealt with when the data set will be used to train classifier. This paper tries to tackle the problem of missing data by introducing new method of data imputation that uses Random Subspace Method. Paper then compares the effectiveness of this method against older methods that use average, mean and median method. Results are presented within graphs. Graphs confirm that new method is more effective when dealing with empty data than other methods.

**Keywords:** Data imputation · Classification task · Machine learning.

## 1 Introduction

One of the biggest challenges in data quality is the presence of gaps in data. There are many sources of missing data such as patient's death, malfunction of measuring equipment or mishandling of the samples. As data quality is the key factor when it comes to machine learning the quality of the set will largely decide on the accuracy of algorithms [3]. To avoid having to discard data, we must come up with solutions that deal with these gaps, while being careful not to introduce bias into the sets. Most of the time the attributes in the dataset are dependent on each other, so that some techniques may be used to find these connections and, based on them, determine the missing values [2]. The term imputation implies replacing the empty value by another one that is plausible in the dataset terms, and not introduce bias while doing so [4]. All of the imputation work is done before feeding the dataset into a learning algorithm, thus the imputation is useful throughout many situations.

## 2 Purpose and scope of work

The ultimate goal of this paper was to create a method that could replace missing data in a dataset used for classification. The missing data has been replaced using linear regression. To process the data, a script was created that implemented various primitive data imputation methods used now, and the new linear regression method. Then, using metrics of widely used machine learning library, the comparison of new and old methods had been done and the results of those comparisons were summarized in the results chapter.

### 3 Datasets

This paper was based on four datasets. Naming classes for datasets had been assigned. Iris and Breast cancer are considered small sets. Yeast and Car evaluation are considered big. This naming convention will come handy later, when the performance of each imputation process is measured. Iris is the most popular dataset frequently used in machine learning study. It consists of four variables which are the length and the width of flower petals of three different kinds of flowers these kinds of flowers make up for classes of the dataset. The second dataset is Yeast. Yeast attributes are results of certain tests performed on yeast. The class of the set corresponds to the localization site of a protein in yeast. There is a possibility of ten localizations, thus the set has ten classes with names implying the localization. The third dataset is Car evaluation. Car evaluation consists of car attributes such as buying price, number of doors and a few technical characteristics. Overall, there are six attributes and four classes. The result classes are: unacceptable, acceptable, good or very good. The last dataset is Breast cancer prognostic dataset. It contains thirty-four attributes, out of which attribute number 1 is the id of a patient and the rest are certain characteristics of breast cancer occurring in a patient. There are two labels. The first label is a Boolean value predicting if cancer is recurring or not, the second label is time: time of recurrence, if the cancer is recurrent, or time for a patient to become disease-free if the cancer is non-recurrent. The datasets were additionally prepared before experiments. Preparation consisted of removing certain columns of values that were deemed not useful for classification tasks (as the work revolves around preparing the dataset to classification task). All datasets had label columns removed, as those should be present without gaps in a dataset. Yeast dataset had been absolved of the first column, which contained a non-relevant accession number for the SWISS-PROT database. Breast cancer had the first column removed too, as it contained patients id.

The Car evaluation dataset has been reworked to contain integer numbers instead of text values in all columns. In figure 5 we can see the Car evaluation dataset before and after preparation.

All of the datasets were obtained from the UCI machine learning repository [1]

### 4 Workflow

This section will thoroughly explain the workflow of experiments, and explain how the process of gathering the results looked like.

At the launch of the script original datasets had been loaded. After that, the preparation of the datasets occurred. When the sets were prepared, 1% of missing values was introduced into datasets. Essentially, we replaced 1% of data with nan values. Nan value is a not-defined or not-representative value. In programming, it is frequently used to show the absence of a value in a certain place. After the removal of data, four imputation methods were applied to datasets, and three

metrics were measured. These metrics show the difference between the original dataset and the one that has just had its empty values replaced using imputation. After the procedure was completed, we saved the metrics results to a separate array.

The process was repeated 100 times for every possible percentage value between 1% and 35%. This means the script had run 100 iterations per 1%, 100 iterations per 2% up until 100 iterations had been running on 35% of missing data. Every time the missing data percentage increased by 1%, the metric results were averaged, so that we get one value per metric. Then the metrics were saved in memory, alongside the current missing data percentage.

When 100th iteration on the dataset with 35% of missing values was completed, the script saved the results from memory into a text file (Fig. 6). The saved results were then imported into a Microsoft Excel file to visualize the results on the graphs

## 5 Methods

In this part, we will take a look at the imputation patterns used in this work. Four imputation methods were used: Mean, median, most frequent and linear regression. The first three are already implemented in the Scikit-learn package, in a module known as "SimpleImputer" and thus will later be referenced as simple imputer mean, simple imputer median and simple imputer most frequent. The last approach has been invented and implemented by the author of this document. For the sake of this paper, the last technique has been called a linear regression method. Simple imputer mean, median and most frequent are considered the most primitive options of imputation, while linear regression is the most advanced.

## 6 Metrics

For the sake of paper three metrics were used: R2 score, Root mean square error and mean square error.

R2 (R squared) is a statistical measure, that represents the proportion of the variance for a dependent variable, that is explained by an independent variable, or variables, in the regression. It is also known as the coefficient of determination. It shows how close the data are to the fitted regression line. R2 score is a value between 1 and 0, where 1 is the best possible score, and 0 is the worst. Root mean square error is a value between 1 and 0. The less the value of RSME, the better the performance of the method is

Root mean square error (often called RMSE) is a measure of the differences between the imputed dataset and the original dataset. The error shows a difference between a prediction and an actual observation. This measure shows how well the regression model performed.

Mean absolute error is another measure of the differences between original and imputed datasets. It measures the average magnitude of errors in a set of predictions. MAE shows the number of errors in a given measurement.

Both MAE and RSME express average model prediction error, but taking the square root in case of RSME has implications. Since errors are squared before they are averaged, RMSE gives high weight to large errors. In this case, RMSE increases with the variance of the frequency distribution of error magnitudes, while MAE remains steady.

## 7 Results

This part contains the results of the tests for separate datasets. The results of the tests were gathered and presented in graphs to show how well they performed. Figure 20 to 31 present averaged results of every approach as a function of the percentage of missing values (from 1% to 35% with a jump of 1%). Results were aggregated per metric type. Every dataset has three graphs - one graph per one metric. All metrics are values between 1 and 0. In the case of R2 score, the closer to 1, the better. For the rest of the metrics (RMSE and MAE) the closer to 0 they are, the better.

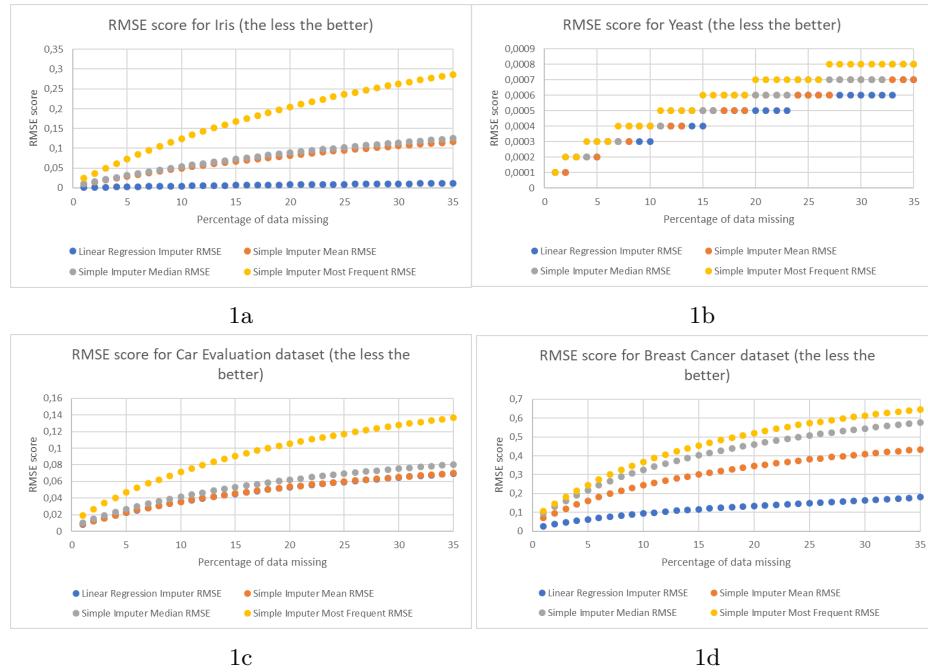


Fig. 2: Results of root mean square error score for the Iris (1a), Yeast (1b), Car evaluation (1c) and Breast cancer (1d) datasets

RSME scores display similar behavior as R2 scores. Linear regression imputer defeats other methods in the case of Iris and Breast cancer. For the rest of the datasets, linear regression performs just a little better than Mean or Median, which both show results, that are almost the same. Most frequent is still relatively the worst, but in the case of Yeast dataset, RMSE shows it performs practically the same as other algorithms.

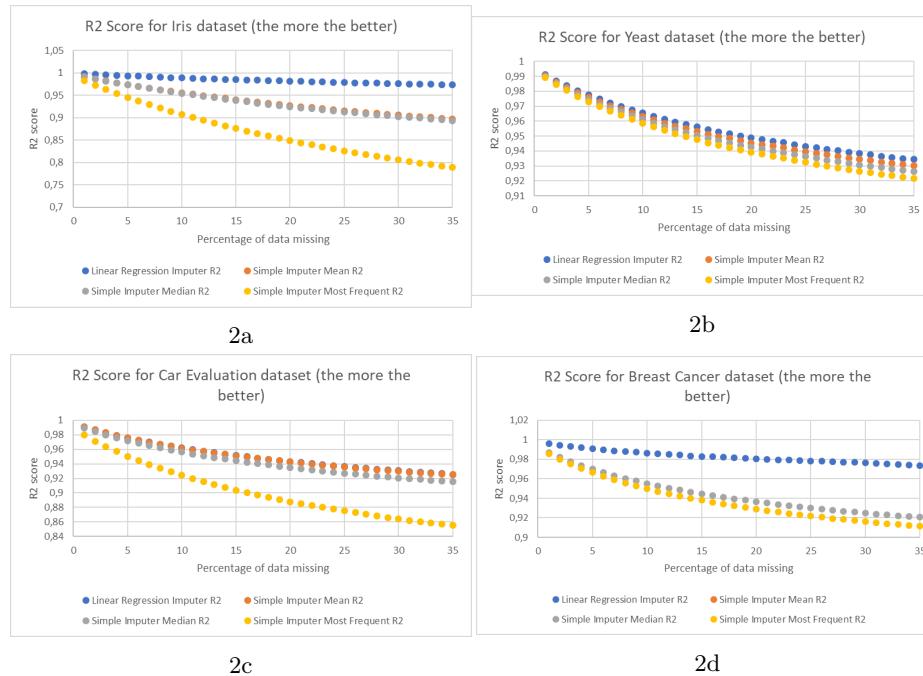


Fig. 4: Results of R2 score for the Iris (2a), Yeast (2b), Car evaluation (2c) and Breast cancer (2d) datasets

The results of R2 score for the linear regression imputer showed major improvements when imputing data in case of Iris and Breast cancer datasets. Mean and median are exhibiting very similar results. When it comes to Car Evaluation and Yeast, linear regression imputer is just a little better than Mean and Median. Most frequent shows the worst performance according to R2 score.

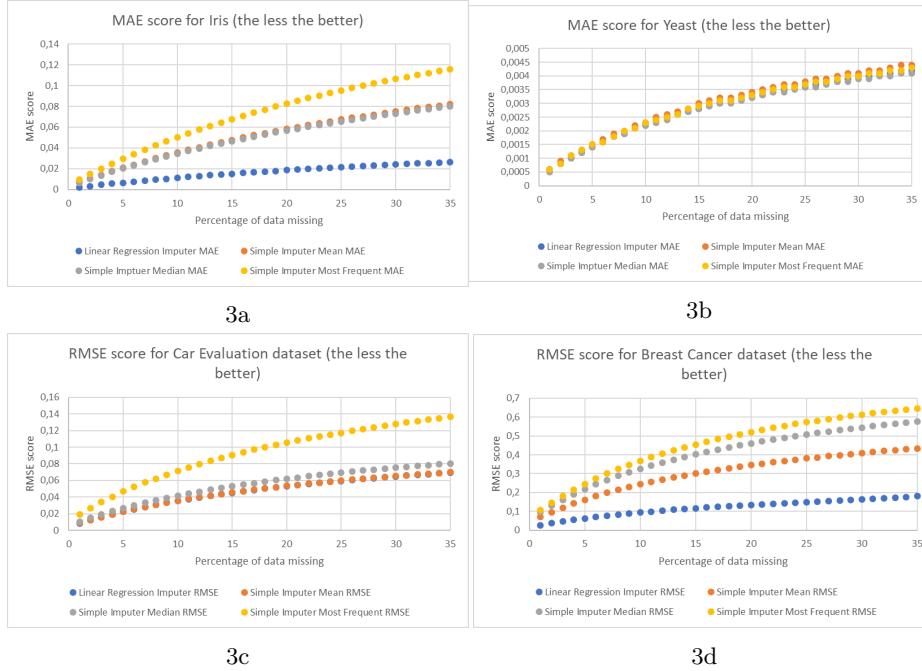


Fig. 6: Results of Mean absolute error score for the Iris (3a), Yeast (3b), Car evaluation (3c) and Breast cancer (3d) datasets

Mean absolute error score confirms the results of other metrics. Still, linear regression imputer outperforms other techniques when we look at Iris and Breast cancer, but results differ when we approach Yeast and Car evaluation. For Yeast, we can see that the mean method achieves its task marginally better than the rest of the methods.

Three different approaches for data imputation, that are used in the literature, were selected. One method for imputation has been implemented additionally by the author. A comparison of these techniques has been done using three separate metrics: R2, RSME, and MAE. For the testing data, four datasets had been picked. Two of the datasets are considered small (Iris and Breast cancer), and two other big (Yeast and Car evaluation). Datasets had been shown in figure 5. All imputers showed similar behaviour – the more data is missing, the worst performance of every imputation method gets. All of the methods presented in this work are using statistics. As the amount of missing data increases, the pool of samples that methods used to come up with imputed values decreases, causing a bigger variance of the imputed values from the original ones. This reaction results in performance worsening over an increasing amount of missing data. The most frequent method turned out to be less effective when applied to bigger datasets, and more effective when applied to smaller sets. Even though the effectiveness has risen with small datasets, most frequent turned out to be

the worst method overall. When it comes to Yeast and Car evaluation dataset Linear regression proved to provide the same or negligibly different results as the mean and the median method. Most frequent stayed behind all the other three. Linear regression proved to be the most efficient amongst all tested processes. Median and mean usually stayed between the best, and the worst method. The difference appears when we look at bigger datasets. Linear regression shows its dominance in lower size datasets. The higher the amount of data goes, the similarity between linear regression and mean rises until they perform almost the same. When we reach a threshold of around 1500 samples, the linear regression imputer method results are indistinguishable from the mean method.

## References

1. Dheeru Dua and Casey Graff. UCI machine learning repository. 2017.
2. Nicholas J. Horton and Ken P. Kleinman. Much Ado About Nothing: A Comparison of Missing Data Methods and Software to Fit Incomplete Data Regression Models. *The American Statistician*, 61:79–90, February 2007.
3. Roderick J A Little and Donald B Rubin. *Statistical Analysis with Missing Data*. John Wiley & Sons, Inc., New York, NY, USA, 1986.
4. D. B. Rubin. *Multiple Imputation for Nonresponse in Surveys*. Wiley, 1987.

# Exploration of performance measurement methods for selected unsupervised machine learning algorithms

Filip Guzy<sup>[0000-0002-8619-7080]</sup>

Wroclaw University of Science and Technology

**Abstract.** Unsupervised learning allows us to find relations in data with no labels. This paper focuses on the comparison of three different algorithms, commonly used in Kohonen maps training: Winner Takes Most (WTM), Conscience Winner Takes Most (CWTM) and Neural Gas (NG). Models are trained on three datasets and compared according to typical criteria of time and quantization error. A custom criterion based on the hybrid network performance is also proposed. Obtained results show that WTM is the fastest one, but CWTM and NG achieve lower quantization errors. Statistical tests that are used to compare hybrid models created from pre-trained Kohonen maps fail to choose the best algorithm.

**Keywords:** Unsupervised learning · Kohonen map · Winner Takes Most · Conscience Winner Takes Most · Neural Gas · Hybrid models

## 1 Introduction

Unsupervised learning is a good strategy when our data samples have no labels. There are a lot of models and algorithms that are used in this kind of problems. Some of the most popular are Adaptive Resonance Theory (ART) networks, characterized by Grossberg and Carpenter [2] or Kohonen maps, described by Alpaydin [1]. This paper focuses on the comparison of three unsupervised learning algorithms described by Osowski [8, 7] that are usually applied in terms of Kohonen maps training: WTM, CWTM and NG. There are several ways to measure algorithm quality. Osowski proposes to calculate the quantization error [8, 7]. Duda, Hart, and Stork suggest measuring the cluster quality by using scatter matrices [5]. There is also a possibility to calculate the cluster purity [6]. In this paper, apart from using standard criteria like time and quantization error, the simple metric involving hybrid models [8, 7] is introduced. Labeled outputs from the pre-trained Kohonen map are employed as a training set for multilayer perceptron with one hidden layer. It allows us to benefit from both unsupervised and supervised methods. Hybrid models are then evaluated with the F1 score metric and compared using Friedman test and post-hoc Nemenyi test, as it is recommended by Alpaydin [1] and Demsar [3]. Obtained hybrid models performance scores can measure the quality of unsupervised algorithms that were used to train input Kohonen maps.

## 2 Experiment and results

Three datasets from the UCI repository [4]: Breast Cancer, Seeds, and Ecoli were chosen to be used in the research. As they are originally imbalanced, the undersampling was made first. The first part of the experiment assumed comparing the fixed number of models with various hyperparameters. After choosing the best algorithm in WTM, CWTM, and NG groups, the best algorithms from each group were compared among themselves. The comparison was made for two simple criteria: time and quantization error. Table 1 presents execution times achieved by the best WTM, CWTM, and NG models.

**Table 1.** Execution times for the best algorithms from each group.

| Algorithm | Time [s]<br>Breast cancer | Time [s]<br>Seeds | Time [s]<br>Ecoli | Mean time [s] |
|-----------|---------------------------|-------------------|-------------------|---------------|
| WTM       | 3.7869                    | 3.7292            | 3.5707            | 3.6956        |
| CWTM      | 3.7709                    | 3.7390            | 3.7650            | 3.7583        |
| NG        | 4.4513                    | 4.4411            | 4.4459            | 4.4459        |

WTM algorithm is, without doubt, the quickest one, because it implements just the base mechanisms which are then extended by CWTM and NG implementations. CWTM model is performing a little bit worse because it requires calculating the neurons potentials during its runtime. NG algorithm presents the longest execution time, as it implements the neurons sorting, which is the most time-consuming computational operation.

Table 2 presents the quantization errors achieved by the best WTM, CWTM, and NG algorithms.

**Table 2.** Quantization errors for the best algorithms from each group.

| Algorithm | Quantization error<br>Breast cancer | Quantization error<br>Seeds | Quantization error<br>Ecoli | Mean quantization<br>error |
|-----------|-------------------------------------|-----------------------------|-----------------------------|----------------------------|
| WTM       | 0.7068                              | 0.1250                      | 0.3865                      | 0.4061                     |
| CWTM      | 0.7068                              | 0.1250                      | 0.3868                      | 0.4062                     |
| NG        | 0.7067                              | 0.1251                      | 0.3858                      | 0.4059                     |

It can be seen that the differences between algorithms are hardly noticeable, but the error achieved by the NG model is the smallest one. This is because the neuron sorting mechanism is the best way to prevent creating dead neurons [8, 7]. It reduces the difference between wages and learning samples on a bigger area of the map, which has a direct influence on a quantization error. The same happens during potentials calculation process in CWTM method, but this effect is not visible in case of chosen datasets.

The second phase of the experiment involved creation of hybrid networks from pre-trained Kohonen maps. The training was performed using 10x10 cross-validation, as Alpaydin suggests [1]. The created perceptron had one hidden layer containing one hundred neurons. The evaluation metric was macro F1 score, as we are dealing with several balanced datasets. Tables 3, 4 and 5 show the F1 scores obtained from training.

**Table 3.** Hybdrid models (WTM) F1 scores.

| Algorithm number | F1 score<br>Breast cancer | F1 score<br>Seeds | F1 score<br>Ecoli | Mean F1 score |
|------------------|---------------------------|-------------------|-------------------|---------------|
| 1                | 0.9051                    | 0.8862            | 0.8373            | 0.8762        |
| 2                | 0.9101                    | 0.9045            | 0.8507            | 0.8884        |
| 3                | 0.9047                    | 0.9113            | 0.8493            | 0.8884        |
| 4                | 0.9121                    | 0.8947            | 0.8394            | 0.8821        |
| 5                | 0.9148                    | 0.8930            | 0.8613            | 0.8897        |
| 6                | 0.9098                    | 0.8974            | 0.8351            | 0.8808        |
| 7                | 0.9066                    | 0.8874            | 0.8586            | 0.8842        |
| 8                | 0.9125                    | 0.8893            | 0.8311            | 0.8776        |
| 9                | 0.9160                    | 0.9111            | 0.8259            | 0.8843        |

**Table 4.** Hybdrid models (CWTM) F1 scores

| Algorithm number | F1 score<br>Breast cancer | F1 score<br>Seeds | F1 score<br>Ecoli | Mean F1 score |
|------------------|---------------------------|-------------------|-------------------|---------------|
| 1                | 0.9119                    | 0.8957            | 0.8499            | 0.8858        |
| 2                | 0.9063                    | 0.8949            | 0.8381            | 0.8798        |
| 3                | 0.9076                    | 0.8926            | 0.8479            | 0.8827        |
| 4                | 0.9085                    | 0.9023            | 0.8639            | 0.8916        |
| 5                | 0.9098                    | 0.8913            | 0.8513            | 0.8842        |
| 6                | 0.9060                    | 0.8973            | 0.8603            | 0.8879        |

**Table 5.** Hybdrid models (NG) F1 scores

| Algorithm number | F1 score<br>Breast cancer | F1 score<br>Seeds | F1 score<br>Ecoli | Mean F1 score |
|------------------|---------------------------|-------------------|-------------------|---------------|
| 1                | 0.9128                    | 0.8896            | 0.8630            | 0.8884        |
| 2                | 0.9157                    | 0.9038            | 0.8560            | 0.8919        |
| 3                | 0.9183                    | 0.8915            | 0.8398            | 0.8832        |
| 4                | 0.9107                    | 0.8926            | 0.8563            | 0.8865        |
| 5                | 0.9157                    | 0.8825            | 0.8536            | 0.8839        |
| 6                | 0.9125                    | 0.8968            | 0.8655            | 0.8916        |

Achieved results were tested using Friedman and post-hoc Nemenyi tests. The null hypothesis ( $H_0$ ) assumed that F1 scores for models from separate groups will be similar. Friedman test rejected  $H_0$  with significance level of  $\alpha = 0.05$ , but the Nemenyi post-hoc test failed to reject it. Summarizing, it is not possible to choose the best hybrid model using current experiment setup.

### 3 Conclusions

Achieved execution times and quantization errors are aligned with Osowski description [8, 7]. If there is a need for the fastest algorithm, we should use WTM, but if we want the algorithm with the lowest quantization error, we should pick NG. According to the hybrid models performance, there are few ways to improve this part of the experiment in the future research. Firstly, we should extend the number of databases, as it is recommended for Friedman test. The low number of datasets leads to poor comparison. We may notice that the mean F1 scores oscillate around 0.88, so probably the capacity of the selected perceptron model may be inadequate. Incrementing the number of hidden layer units or adding more layers may result in better performance, and it may provide different experiment outputs. We can also consider choosing a different network architecture. For example, as we have 2D inputs, we may try convolutional neural networks. After all, it is also possible to validate models using other statistical tests.

### References

1. Alpaydn, E.: Introduction to Machine Learning. Second Edition. Massachusetts Institute of Technology (2010)
2. Carpenter G. A., G.S.: The ART of adaptive pattern recognition by a self-organizing neural network. Computer. 21 7788 (1988)
3. Demsar, J.: Statistical Comparisons of Classifiers over Multiple Data Sets. Journal of Machine Learning Research (2006)
4. Dua, D., Graff, C.: UCI Machine Learning Repository, <http://archive.ics.uci.edu/ml> (2017), access: 15.09.2019
5. Duda R. O., Hart P. E., S.D.G.: Unsupervised Learning and Clustering (1995)
6. Estivill-Castro, V.: Why so many clustering algorithms. A Position Paper. ACM SIGKDD Explorations Newsletter. 4 (2002)
7. Osowski, S.: Sieci neuronowe w ujciu algorytmicznym. Wydawnictwa Naukowo-Techniczne, Warszawa (1996)
8. Osowski, S.: Sieci neuronowe do przetwarzania informacji. Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa (2006)

# A Cluster-Based Approach for AIS Data Analysis and Vessel Trajectory Reconstruction

Marta Mieczyńska<sup>[0000-0002-3793-6641]</sup>, a and Ireneusz Czarnowski<sup>[0000-0003-0867-3114]</sup>, b

<sup>a</sup>Department of Marine Telecommunications

<sup>b</sup>Department of Information Systems, Gdynia Maritime University

Morska 83, 81-225 Gdynia, Poland

m.mieczynska@we. umg.edu.pl, i.czarnowski@umg.edu.pl

**Abstract.** The paper deals with the problem of analysis of data coming from Automatic Identification System (AIS) using a clustering technique. Automatic Identification System provides the information about a vessel. Analysis of AIS data can deliver an important information and knowledge about vessel movements and their behaviors with respect to the maritime safety, as well as detection of unwanted and prohibited maritime operations. On the other hand, AIS data can be incomplete, i.e. can have missing points in their protocol. AIS data can be also incorrect. It means, that AIS data analysis can be impossible or can provide useless results, far from real situation and actual vessel positions. The aim of the paper is to show that the clustering can be helpful in recognition of ships' behaviors and for vessel's trajectory reconstruction, when AIS data are incomplete or incorrect. The proposed procedure of the shipping trajectory clustering based on empirical AIS data is discussed.

**Keywords:** clustering, AIS data analysis, vessel trajectory analysis

## 1 Introduction

The Automatic Identification System (AIS) is an automatic tracking system based on transponders located on ships [1]. Unique ship identification number (MMSI), name, type and dimensions of ship, ship's position, course over ground (COG), speed over ground (SOG) and true heading are transmitted by AIS transponders with regular intervals.

In general, the AIS data have been widely adopted for the navigational safety and maritime traffic management. The AIS message can be received by the other vessels, satellites and terrestrial antennas. These messages can also be collected and used for different analysis, among them for prediction of future vessel's movements. Examples of analysis can also concern ship traffic, for instance in port. The maritime traffic safety needs more attention and analysis of data collected from AIS can help to control and predict of ship's behaviors. Real-time and historical AIS data can contain potentially useful markers for the early identification of anomalous activities of vessels and risk of collision. The AIS data analysis can also be helpful for optimization of different operations or for managing of vessels and transportation operations. Vessel's behavior

analysis are now also important from global scale point of view, for example with respect to maritime terrorism and growing number of acts of piracy [2].

To sum up, the mentioned analyses should be carried out using advanced methods which belong, for example to the family of machine learning methods or based on data mining techniques. Examples of data analysis using data mining tools have been discussed, for example, in [5], [6] or [7].

On the other hand however, machine learning methods or data mining techniques can help in processing AIS data, when they are incomplete or incorrect, i.e. when they include so-called outliers. One reason for incomplete or incorrect data can result from nature of the VHF transmission. AIS message loss may be related to the intermediate processes of the transmission or by a prone of the VHF to environmental conditions such as rain or fog. Additionally restricting the receipt of information can be limited by shadowing caused by land masses or other vessels [8]. From practical point of view, existing of incomplete or incorrect data can be crucial for working of different systems, for example, dedicated to traffic safety, navigational safety or maritime traffic and risk management.

In [8], a problem of reconstruction of the vessel route points, when the missing data within AIS message were identified, has been discussed. Especially, the Random Forest method has been applied to identify the missing position records within vessel trajectories, and next, an artificial neural network approach has been proposed for the reconstruction of vessel trajectories.

Missing or incorrect data or points in the vessel's trajectory are typical for the data collected by the satellites in the satellite Automatic Identification System [3]. The Space-based AIS (SAT-AIS) has the problem of missing data due to the restrictions of the AIS protocol. Reconstruction of the vessel route points, in case when the missing data were identified, has been discussed in [4].

In this paper, a clustering approach using k-means algorithm for vessel trajectory reconstruction, when AIS data are incomplete, is presented. The main research question was, whether the clustering approach can help in the mentioned reconstruction.

The paper is organized as follows - Section 2 contains problem formulation. Section 3 provides details on the proposed approach. Case study using AIS data obtained from the Gulf of Gdansk is presented in section 4. Conclusions and suggestions for future research are included in the final section.

## 2 Problem formulation

The vessel's trajectory can be expressed by set of following vectors:  $T_i^{t_m} = [x_1, x_2, x_3, \dots, x_N]_{t_m}^i$ , where  $T_i^{t_m}$  represents a vessel trajectory point observed at time  $t_m$  and collected basing on AIS messages, where  $i$  is MMSI of the vessel,  $m$  is a number of time steps/points and  $x_1, x_2$  are the longitude and latitude, respectively,  $x_3$  can be the SOG,  $x_4$  can be the COG, and so on. It also means that the vessel's trajectory can be modeled as a time series, where each one vessel trajectory point is a  $N$  dimension ( $N$ -D) vector of reported dynamic information.

In case of incomplete data at least one element of the  $N$ -D vector is missing, i.e.  $\exists_{t_m} \exists_{k:k=1,\dots,N} \{x_k\} = \emptyset$ . Whereas in case of incorrect data at least one element of the  $N$ -D vector has a value, which does not belong to the accepted range.

From practical point of view, AIS messages are streams of data. Thus to get AIS data from the streams synchronization points must be establish. These synchronization points define so-called interpolation intervals. Considering incomplete or incorrect data, it can mean that missing or incorrect positions are detected within the interpolation intervals.

The aim of vessel trajectory reconstruction is to correct data, recover or rebuilt the lost points on the trajectory by their estimation, approximation or prediction. Mathematical or statistical methods can be applied to the process of the vessel's trajectory reconstruction. Another approach is to apply machine learning methods.

### 3 The proposed approach

In this paper for vessel trajectory reconstruction a cluster-based approach is proposed. A set all available multidimensional AIS data are clustered in several different ways, i.e. iteratively for several different values of the parameter defining number of clusters. For clustering the k-means algorithm has been applied. Obtained results have been additionally validated with respect to the consistency within clusters of data using the silhouette technique. The obtained clusters, for which an average value of silhouette is the best (has a highest value), have been returned as the results of the proposed vessel's trajectory reconstruction procedure. It is assumed that outliers represent the abnormal vessel locations. So, it can also mean, that these locations result from incorrect data. In discussed implementation the detected outliers have been added to their nearest clusters, with the silhouette value as a validated factor.

### 4 Case study

In this paper, the vessel trajectory reconstruction is shown based on AIS data obtained from the Gulf of Gdansk. The data has been recorded in 35 minute time period and include 1077 different AIS packages, which identify 23 vessels. Then only part of the packages describing each of the 23 ships have been preserved in original. The remaining packages were damaged. Errors were introduced based on the mechanism of random disturbance of the AIS message structure. It means, that dataset in the experiment consisted of original and damaged messages, i.e. incorrect values. Fig. 1 shows the locations and the vessel's trajectories for the considered data set (a), as well as, the clustering results for AIS data with damaged cases (b). Fig. 1(b) shows also outliers detected which have been merged with data most closeness, i.e. the vessel trajectories have been reconstructed.

### 5 Conclusions

In this paper, a vessel trajectory clustering is discussed. The vessel's trajectory is defined as a set of points (location) obtained basing on the AIS data. The clustering can help to identify anomaly (outliers) that can be result of incorrect AIS data. Next, such

identify can help in reconstruction of loss of vessels' trajectory and for analysis of ships' behaviours. The discussed and very simple approach could be used to enhance security and safety in the maritime systems.

The future work will consist of verification of different decisions concerning the outliers as well as their detections. A future question will also concern the distance measure, which should be used for vessel trajectory reconstruction based on the multidimensional AIS data.

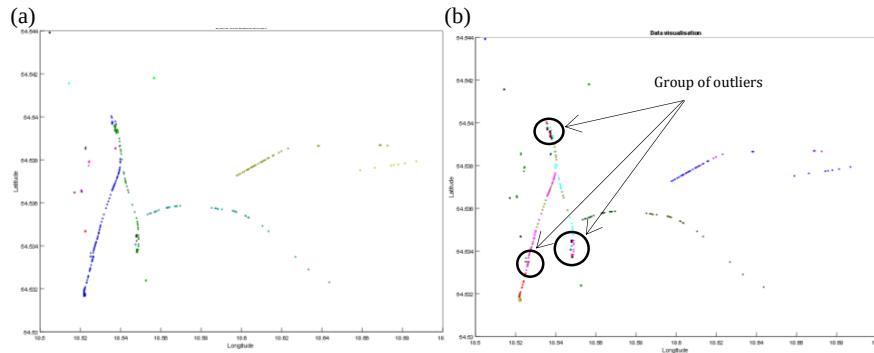


Fig. 1. Collected AIS data for 23 ships, including their representative trajectories (a) and the clustering results for AIS data with damaged cases and with showing the outliers (b).

## References

1. Automatic Identification Systems (AIS), IMO, <http://www.imo.org/en/OurWork/Safety/Navigation/Pages/AIS.aspx> [Accessed in March 2019]
2. Vessel Tracking Pioneer Recalls System's Post-9/11 Origins, <https://portal.midatlanticocean.org/ocean-stories/automatic-identification-system-tracking/> [Accessed in June 2019]
3. Satellite – Automatic Identification System (SAT-AIS) Overview, <https://artes.esa.int/sat-ais/overview> [Accessed in March 2019]
4. Pan, S., Yin, J., Extracting Shipping Route Patterns by Trajectory Clustering Model Based on Automatic Identification System Data. *Sustainability* 10.7 (2018): 2327.
5. Filipiak, D, Strózyna M, Węcel K, Abramowicz W., Big Data for Anomaly Detection in Maritime Surveillance: Spatial AIS Data Analysis for Tankers. *Scientific Journal of Polish Naval Academy* 215 (2018): 5–28
6. Deng, F, Guo, S, Deng, Y, Chu, H, Zhu, Q, Sun, F, Vessel track information mining using AIS data. In: 2014 International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI), 2014
7. Xu, G, Li, F, Chen, Ch-H, Local AIS data analytics for efficient operation management in Vessel Traffic Service. In: 2017 13th IEEE Conference on Automation Science and Engineering (CASE), 2018
8. Liang, M, Liu, R,W, Zhong, Q, Liu, J, Zhang, J, Neural Network-Based Automatic Reconstruction of Missing Vessel Trajectory Data. 2019 IEEE 4th International Conference on Big Data Analytics (ICBDA), China, 2019, doi: 10.1109/ICBDA.2019.8713215

# Aspect-based Sentiment Analysis Summarization using Rhetorical Analysis and Complex Networks

Lukasz Augustyniak, Tomasz Kajdanowicz, and Przemysław Kazienko

Wrocław University of Science and Technology  
`{lukasz.augustyniak,tomasz.kajdanowicz,przemyslaw.kazienko}@pwr.edu.pl`

**Abstract.** This project fills a gap in aspect-based sentiment analysis and aims to present a new method for preparing and analysing texts concerning opinion and generating user-friendly descriptive reports in natural language. We present a comprehensive set of techniques derived from Rhetorical Structure Theory and sentiment analysis to extract aspects from textual opinions and then build an abstractive summary of a set of opinions. Moreover, we propose aspect-aspect graphs to evaluate the importance of aspects and to filter out unimportant ones from the summary. Additionally, the paper presents a prototype solution of data flow with interesting and valuable results.

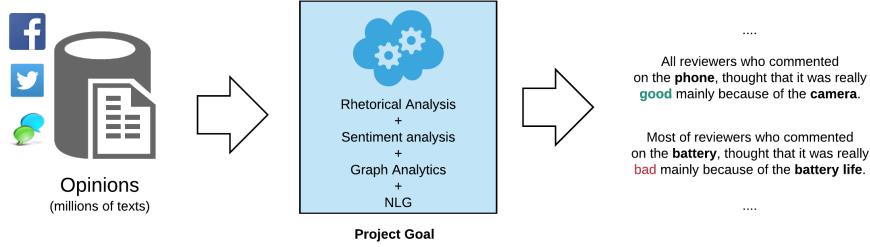
**Keywords:** aspect-based sentiment analysis summarization · opinion mining · aspect extraction ·

## 1 Introduction

Modern society is an information society bombarded from all sides by an increasing number of different pieces of information. The 21st century has brought us the rapid development of media, especially in the internet ecosystem. This change has caused the transfer of many areas of our lives to virtual reality. New forms of communication have been established. Their development has created the need for analysis of related data. Nowadays, unstructured information is available in digital form, but how can we analyse and summarise billions of newly created texts that appear daily on the internet? Natural language analysis techniques, statistics and machine learning have emerged as tools to help us. In recent years, particular attention has focused on sentiment analysis. This area is defined as the study of opinions expressed by people as well as attitudes and emotions about a particular topic, product, event, or person. Sentiment analysis determines the polarisation of the text. It answers the question as to whether a particular text is a positive, negative, or neutral one.

Our goal is to build a comprehensive set of techniques for preparing and analysing texts containing opinions and generating user-friendly descriptive reports in natural language - Figure 1. In this paper, we describe briefly the whole workflow and present a prototype implementation. Currently, existing solutions

for sentiment annotation offer mostly analysis on the level of entire documents, and if you go deeper to the level of individual product features, they are only superficial and poorly prepared for the analysis of large volumes of data. This can especially be seen in scientific articles where the analysis is carried out on a few hundred reviews only. It is worth mentioning that this task is extremely problematic because of the huge diversity of languages and the difficulty of building a single solution that can cover all the languages used in the world. Natural language analysis often requires additional pre-processing steps, especially at the stage of preparing the data for analysis, and steps specific for each language. Large differences can be seen in the analysis of the Polish language (a highly inflected language) and English (a grammatically simpler one). We propose a solution that will cover several languages, however in this prototype implementation we focused on English texts only.



**Fig. 1.**

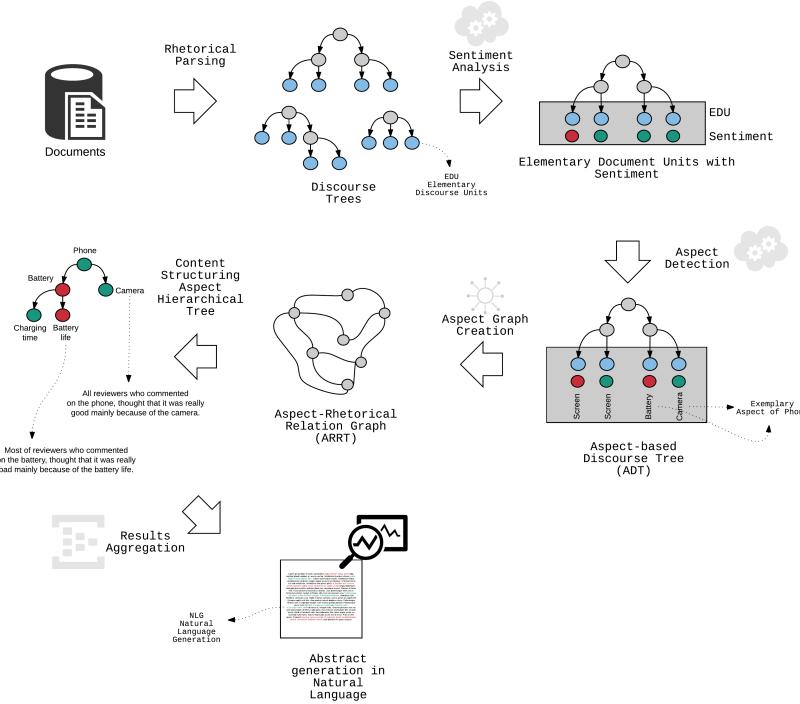
In this paper, we present analysis and workflow inspired by the work of Joty, Carenini and Ng [3]. We experimented with several methods in order to validate aspect-based sentiment analysis approaches and in the next steps we want to customise our implementation for the Polish language.

## 2 Method for aspect-based sentiment analysis

The proposed Rhetorical and Sentiment Analysis flow is divided into four main tasks:

1. Rhetorical analysis with sentiment detection.
2. Aspect detection in textual data.
3. Methods, techniques, and graph analytics of aspect inter-relations.
4. Abstractive summary generation in natural language (not included in prototype workflow yet).

The overall characteristics and flow organisation can be seen in Figure 2. Each of the mentioned steps of the proposed method is described in the following subsections.

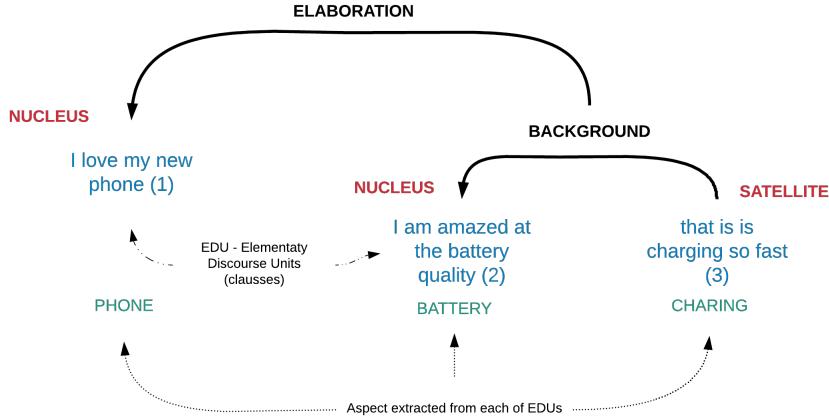


**Fig. 2.** The workflow for Rhetorical and Sentiment Analysis.

## 2.1 Rhetorical Analysis

The goal of discourse analysis in our method is the segmentation of the text for the basic units of discourse structures EDU (Elementary Discourse Units) and connecting them to determine semantic relations. The analysis is performed separately for each source document, and as the output we get Discourse Trees (DT) such as in Figure 3. At this stage, existing discourse parsers will model the structure and the labels of a DT separately. They do not take into account the sequential dependencies between the DT constituents. Then existing discourse parsers will apply greedy and sub-optimal parsing algorithms and build a Discourse Tree. During this stage, and to cope with the mentioned limitation The inferred (posterior) probabilities can be used from CRF parsing models in a probabilistic CKY-like bottom-up parsing algorithm [4] which is non-greedy and optimal. Finally, discourse parsers do not discriminate between intra-sentential parsing (i.e., building the DTs for individual sentences) and multi-sentential parsing (i.e., building a DT for the whole document) [3]. Hence, this part of the analysis extracts for us distributed information about the relationship between different EDUs from parsed texts. Then we assign sentiment orientation to each EDU. What is really important, RST relations proved to be good information source of hierarchical relations between aspects as we investigated in [1].

I love my new phone. I am amazed at the battery quality that it is charging so fast.



**Fig. 3.** An exemplary Discourse Tree based on Rhetorical Structure Theory.

## 2.2 Aspect detection in textual data

The second step covers aspect extraction and creation of aspect-based discourse trees ADT - see Figure 2. We particularly focused on architectures based on long short-term memory (LSTM) with optional conditional random field (CRF) enhancement using different pre-trained word embeddings. Moreover, we analyzed the influence on performance of extending the word vectorization step with character embedding. The experimental results on SemEval datasets revealed that not only does bi-directional long short-term memory (BiLSTM) outperform regular LSTM, but also word embedding coverage and its source highly affect aspect detection performance. An additional CRF layer consistently improves the results as well. Our comprehensive analysis is presented in [2].

## 2.3 Analysis of aspect inter-relations

The third step consists of an Aspect-Rhetorical Relation Graph (ARRG) and content Structuring Aspect Hierarchical Tree (see Figure 2). Discourse Trees of individual documents are processed (the order of EDU is not changed) to form association rules. Then, an Aspect-Rhetorical Relation Graph based on a set of these rules is created. Each node represents an aspect and each edge is one of the relations between the EDU's aspects. A graph will be created for all documents used in the experiment. The graph can be represented with weighted edges (association rules confidence, a number of such relations in the

whole graph etc.), but there is a need to check and compare different types of graph representations. Then, it is possible to characterise the whole graph and each node (aspect) with graph metrics (PageRank [5], degree, betweenness or other metrics). These metrics will be used for estimating the cut threshold – removing uninformative or redundant aspects. Hence, we will end up with only the most important aspects derived from analysed corpora. Then the graph will be transformed into an Aspect Hierarchical Tree. This represents the correlation between aspects and enables us to generate natural language-based descriptions.

## 2.4 Abstractive summary generation in natural language

The last step covers summary (abstract) generation in natural language. Natural language generation models use parameterized templates (very limited and dependent on the size of the rule-based system responsible for the completions of the text), or deep neural networks [6].

## References

1. Augustyniak, L., Kajdanowicz, T., Kazienko, P.: Comprehensive Analysis of Aspect Term Extraction Methods using Various Text Embeddings (sep 2019), <http://arxiv.org/abs/1909.04917>
2. Augustyniak, L., Kajdanowicz, T., Kazienko, P.: Comprehensive Analysis of Aspect Term Extraction Methods using Various Text Embeddings (sep 2019), <http://arxiv.org/abs/1909.04917>
3. Joty, S., Carenini, G., Ng, R.T.: CODRA : A Novel Discriminative Framework for Rhetorical Analysis. *Computational Linguistics* **41** (2015)
4. Jurafsky, D., Martin, J.H.: Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. *Speech and Language Processing An Introduction to Natural Language Processing Computational Linguistics and Speech Recognition* **21**, 0–934 (2009). <https://doi.org/10.1162/089120100750105975>, <http://www.mitpressjournals.org/doi/pdf/10.1162/089120100750105975>
5. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank Citation Ranking: Bringing Order to the Web. *World Wide Web Internet And Web Information Systems* **54**(1999-66), 1–17 (1998). <https://doi.org/10.1.1.31.1768>, <http://ilpubs.stanford.edu:8090/422>
6. Wen, T.H., Gasic, M., Mrksic, N., Su, P.H., Vandyke, D., Young, S., Mrksic, N., Su, P.H., Vandyke, D., Young, S.: Semantically Conditioned LSTM-based Natural Language Generation for Spoken Dialogue Systems. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (September)*, 1711–1721 (2015), <http://aclweb.org/anthology/D15-1199>\n<http://arxiv.org/abs/1508.01745>

# Towards More Efficient and Secure Federated Learning Methods

Mateusz Piwowarczyk<sup>1</sup>[0000–0002–0557–2919]  
and Bogdan Trawiński<sup>1</sup>[0000–0002–2956–6388]

Wrocław University of Science and Technology, Faculty of Computer Science and Management, Wrocław, Poland  
`{mateusz.piwowarczyk, bogdan.trawinski}@pwr.edu.pl`

**Abstract.** In recent years, Machine Learning has played an increasingly important role in the development of new solutions based on Big Data analysis. At the same time, more and more attention is being paid to the privacy of our data. People are slowly realizing not only the opportunities but also threats of sharing their own data. At the same time, the development of machine learning technology allows us to obtain valuable information based on this data. Today's dominant techniques for building useful machine learning models use centralised data resources. Following Internet of Things era makes that, the usage of data scattered across multiple devices while preserving the privacy of them will become an important challenge. In many cases, it is not possible for certain sensitive data to be shared, but at the same time, there is a great need to obtain information from it. Federated Learning seems to be good answer for this type of problems. This paper gives a concise view at the history of secure multi-party computation, present the methods currently used and points out the challenges facing federated learning techniques in the future.

**Keywords:** machine learning · federated learning · multi-party computation · decentralization · prediction models.

## 1 Introduction

Technological development has resulted in large volumes of data aggregated in many places. More and more people in the world have now access to several electronic devices, such as personal computers or smartphones and this trend is growing. The development of fast mobile Internet connections and Internet of Things will make this number grow even more rapidly and all data collected by all these devices will become an extremely valuable material for improving many processes of our everyday life. Some economists predict that data will be the most valuable resource for many businesses in the future. This trend is also raising awareness among users about the usage of their personal data. More people are starting to verify what information about them is aggregated by external companies. Some sort of data could be very sensitive and for example

tell many things about our health status. But this data at the other hand could be very useful and could contribute to improving our health status overall [3].

The General Data Protection Regulation (GDPR) has also played an important role in ensuring the security of private data in the whole European Union [8]. With its entry into force it was necessary to carry out a number of processes aimed at the implementation of these regulations in many companies. Despite providing greater security and control over their own data by citizens, such procedures also introduce numerous difficulties in building predictive models based on various data collected on the basis of user activity. This makes the solutions allowing to create machine learning models in such a way that sensitive data does not leave the client devices and at the same time prediction models built in on this data are sufficiently precise to bring some business value.

The computer science domain that tries to find solutions that meet such requirements is the subfield of cryptography called secure multi-party computation. The first attempts of such computations can be linked with the attempt to solve the famous Yao's Millionaires' problem formulated by Andrew Yao in 1982 [10]. In this problem we want to check which of the two millionaires is richer and at the same time not revealing value of their assets. Later this problem was extended to any number of millionaires and appropriate cryptographic protocols were designed to solve it. The development of machine learning technology in recent years has made the huge focus on applying these cryptography technologies to the process of learning prediction models. These settings are called Federated Learning. Despite many promising research and a huge gain in popularity of these techniques, there are still many challenges ahead of them before they become mature enough to find a wide application in production solutions. In this paper we briefly describe what are the main challenges of Federated Learning and what stopping it from implementation on a large scale in many systems [2].

## 2 Challenges of Federated Machine Learning

One of the main problems faced by Federated Learning is **effective communication** between clients and the server [5,7]. Building an accurate model cannot be based on transferring large amounts of data in many cycles between clients and server, because this transfer could be slow and expensive.

Another challenge rises in terms of **assuring data privacy**. We need to have a huge certainty, that data sent to the server could not be used to retrieve back original dataset. There are many ways of doing it and there are some methods of splitting popular machine learning algorithms on client and server parts. For example in terms of K-means algorithm [9] we could send only centroid shifts and positions without sending actual data points. But there are still many vulnerabilities for these methods.

It is also possible to have an **adversarial attack** on the machine learning models and creating backdoors for personal benefits [1]. Clients knowing that particular prediction models are built from their private data may intentionally corrupt them to achieve some goals. Let's assume that based on data collected

on client devices, we want to customise advertisements or search results for the preferences of individual users. Many companies would like their advertisement or pages to be displayed on the first position. In another situation an online shop want to create a decentralized system of products recommendations [4], which on the basis of customers private data collected on their computers will build a model of recommendations for individual types of users. In this situation, the manufacturer or retailer may intentionally corrupt our model, e.g. by generating artificial users (bots) performing such actions that the data collected from them significantly influence the global results of the model prediction. In other words, users can manipulate their own data to interfere with the learning process in their favour.

Another problem is the correct **balancing of the calculations**. We should assume, that client devices are usually not very efficient in terms of computing power. Additionally, they are often mobile devices with limited battery capacity. At the same time, slowing down the normal usage of these devices by performing additional calculations related to the partial machine learning may negatively affect the comfort of use. It is desirable that only small amount of computing should be performed on the client side and at the same time all necessary privacy and model accuracy should be provided.

The next problem, that facing Federated Learning is the **heterogeneity** of the systems and heterogeneity in the statistical terms [6]. System heterogeneity is caused by different characteristics of devices taking part in a learning process. These devices could differ in terms of computing power, battery life and network connectivity. Due to this fact we should look for solutions that brings out the best features of these devices and combine them to achieve the best performance. Another type of heterogeneity is related to data itself. Data points are not identically distributed across devices [11]. Some devices have large collection of data points and some of them not. Also usually these data points are not available identically all the time. Some of devices could go offline, while some of them will return online.

### 3 Conclusions

In this article, we shortly introduced what are the history and the current state of developing Federated Learning systems. In such systems statistical models are trained in a distributed way. We have pointed out all main challenges and open problems for such systems. These are: communication efficiency, security assurance, corruption protection, concerns about computation load and heterogeneity. We provided brief survey on current domain state and proposed on which aspects future research should focus on. This paper gives bright guidepost for further investigations. The complexity of problems described in this paper requires increased, interdisciplinary efforts but solutions for them will become very valuable especially in an increasingly interconnected world.

## References

1. Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., Shmatikov, V.: How to backdoor federated learning. arXiv preprint arXiv:1807.00459 (2018)
2. Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingberman, A., Ivanov, V., Kiddon, C., Konečný, J., Mazzocchi, S., McMahan, H.B., et al.: Towards federated learning at scale: System design. arXiv preprint arXiv:1902.01046 (2019)
3. Brisimi, T.S., Chen, R., Mela, T., Olshevsky, A., Paschalidis, I.C., Shi, W.: Federated learning of predictive models from federated electronic health records. International journal of medical informatics **112**, 59–67 (2018)
4. Ammad-ud din, M., Ivannikova, E., Khan, S.A., Oyomno, W., Fu, Q., Tan, K.E., Flanagan, A.: Federated collaborative filtering for privacy-preserving personalized recommendation system. arXiv preprint arXiv:1901.09888 (2019)
5. Konečný, J., McMahan, H.B., Yu, F.X., Richtárik, P., Suresh, A.T., Bacon, D.: Federated learning: Strategies for improving communication efficiency. arXiv preprint arXiv:1610.05492 (2016)
6. Li, T., Sahu, A.K., Talwalkar, A., Smith, V.: Federated learning: Challenges, methods, and future directions. arXiv preprint arXiv:1908.07873 (2019)
7. McMahan, H.B., Moore, E., Ramage, D., Hampson, S., et al.: Communication-efficient learning of deep networks from decentralized data. arXiv preprint arXiv:1602.05629 (2016)
8. Tankard, C.: What the gdpr means for businesses. Network Security **2016**(6), 5–8 (2016)
9. Vaidya, J., Clifton, C.: Privacy-preserving k-means clustering over vertically partitioned data. In: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 206–215. ACM (2003)
10. Yao, A.C.: Protocols for secure computations. In: 23rd annual symposium on foundations of computer science (sfcs 1982). pp. 160–164. IEEE (1982)
11. Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., Chandra, V.: Federated learning with non-iid data. arXiv preprint arXiv:1806.00582 (2018)

CV (computer vision)

# **Automatic identification of vitreomacular pathologies based on optical coherence tomography scans**

Agnieszka Stankiewicz<sup>1</sup> [0000-0002-2983-897X], Tomasz Marciniak<sup>1</sup> [0000-0001-6035-7325],  
Adam Dąbrowski<sup>1</sup> [0000-0002-9385-6080], Marcin Stopa<sup>2</sup> [0000-0001-9540-9500],  
Elżbieta Marciniak<sup>2</sup> [0000-0002-7009-0406]

<sup>1</sup> Poznan University of Technology, Division of Signal Processing and Electronic Systems,  
Jana Pawła II 24, 60-965 Poznan, Poland

<sup>2</sup> Poznan University of Medical Sciences, Chair of Ophthalmology and Optometry,  
Rokitnicka5D, 60-806 Poznan, Poland  
tomasz.marciniak@put.poznan.pl

**Abstract.** The paper presents automatic classification of OCT scans of the human eye retina into three categories: healthy people, people with vitreomacular traction (VMT), and those with idiopathic macular hole (IMH). Feature vectors were created from the computed differences between the ILM (internal limiting membrane) and RPE (retinal pigment epithelium) edges automatically segmented from the OCT scans. The training and classification stages were realized using the Python programming language together with the scikit-learn library.

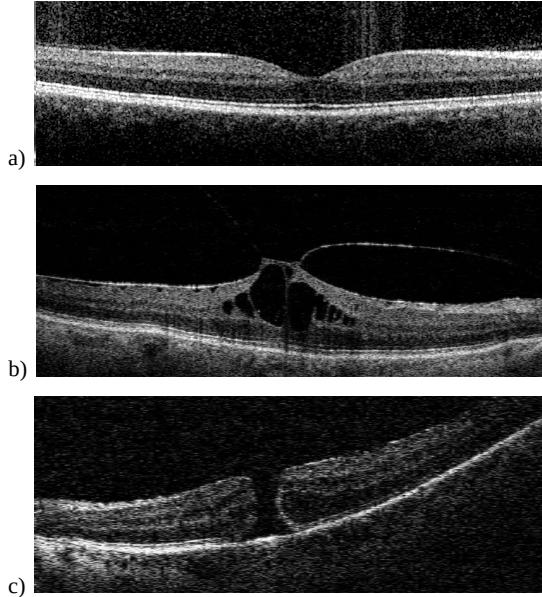
**Keywords:** optical coherence tomography (OCT), VMT, SVM

## **1      Visualisation of vitreomacular interface of human eye using OCT**

Since its invention in 1991, the Optical Coherence Tomography (OCT) is a valuable, non-invasive technology for tissue section imaging using the light scattered on individual layers of the examined tissue [1]. Due to the help of the OCT we obtain retinal cross-sections (B-scans). Many subsequent closely located B-scans can be merged in order to generate three-dimensional retinal models. This proved to be very beneficial for use in medical sciences, providing doctors with the quantitative information on the degree of patient pathologies with the main applications in ophthalmology.

Using OCT enables the ophthalmologists to visualize and monitor the vitreo-retinal interface with a very high accuracy. Thus, it has become an essential tool for clinicians and researchers dealing with the problem of vitreo-macular traction and retinal membrane [2, 3]. Deterioration of visual acuity, image curvature (metamorphopsia) and central vision impairment are typical symptoms of a number of diseases affecting the macula of the human eye. Some of them (vitreous-macular traction, capillary membrane, macular stratum hole, pseudo-hole, full-wall hole) develop in connection with irregularities in the vitreous, at the vitreous and retinal border, and in the retinal architecture.

In this paper we discuss two conditions related to an anomalous PVD (posterior vitreous detachment), namely: vitreomacular traction (VMT) and idiopathic macular hole (IMH).



**Fig. 1.** An example of OCT macula B-scan of a) healthy patient, b) patient with VMT and c) IMH pathologies (images courtesy of Clinical Eye Unit and Pediatric Ophthalmology Service, Heliodor Święcicki University Hospital in Poznan)

## 2 Application of SVM to VRI pathology identification

A group of patients was recruited by the team of participating physicians from the Clinical Eye Unit and Pediatric Ophthalmology Service, Heliodor Święcicki University Hospital in Poznan. Bioethical Committee of the Poznan University of Medical Sciences approved the research and all participants signed the informed consent document before enrollment. Volumetric data was acquired from 21 adults (13 males and 8 females) creating a database of 44 inspections (16 for healthy retina structures, 19 for VMT, and 9 for IMH). The average age in the group was 61 years (ranging from 26 to 90).

In the presented work, the Copernicus HR (Optopol, Poland) device was used to obtain 3-dimensional cross-section images of the central macula with  $8 \times 8$  mm scan width and height, and 2 mm scan depth. The covered volume was represented by a set of 100 OCT B-scan images (a total of 4400) with  $800 \times 1010$  pixel resolution. Each OCT scan was then analyzed with the prepared algorithm.

The experiment was constructed in the following pipeline:

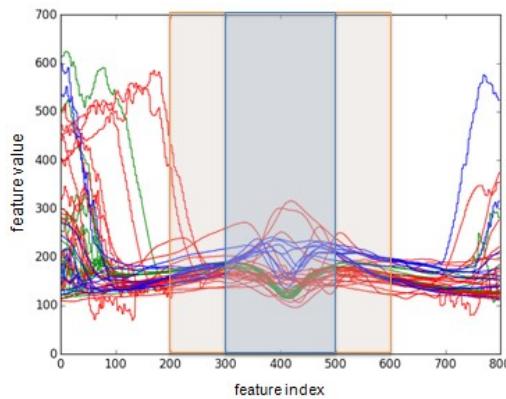
1. The data was subjected to evaluation by experts for the description of existing

- stage of PVD development and subjects classification.
2. Each obtained OCT examination was made with the prepared automatic graph theory-based segmentation algorithm for the segmentation of two layers of the retina, namely the ILM and RPE [4].
  3. Feature vectors were created from the computed differences between ILM and RPE edges, characterizing the depth of the retina along the central OCT B-scan image. Primary feature vectors are of the length equal to the width of the B-scan image, i.e. 800 values.
  4. A SVM system was used in order to establish a supervised classification method able to classify each eye examination into classes based on the feature vectors. Four different models were tested with selected kernels (cf. Table 1).
  5. Calculation of correctness was computed in the final step, followed by the analysis of results of VRI pathology prediction.

To perform the classification the data was divided into two training and testing subcategories with ratio 2:1. Additionally, due to a small number of the obtained OCT examinations the cross-validation strategy needed to be applied. For this purpose the data was cross-validated on three overlapping subsets.

Based on the previous research [5] and analysis of the computed feature vectors presented in Fig. 2 it can be deduced that low quality peripheral regions of the OCT scan significantly impact accuracy of the automatic OCT image analysis leading to errors and false predictions. Furthermore, close analysis of retinal layers shows that crucial information about presence of the pathology is contained to the central region of the OCT scan. Thus, three different lengths of feature vectors were evaluated, to test the influence of lateral image data on the pathology classification.

Fig. 2 presents a set of feature vectors colored according to classes that they are part of (green – healthy patients, red – VMT patients, blue – IMH patients). Overlapping orange and blue areas represent the selected sub-vectors of lengths 400 (equal to  $n/2$ ) and 200 ( $n/4$ ) values, respectively. These areas are centered in the middle of the feature vector. The calculated scores for classification with various kernel models averaged over all samples after 3-fold cross-validation are presented in Tables 1-2.



**Fig. 2.** Values of feature vectors for tested OCT images (green lines – features for healthy patients, red line – VMT patients, blue line – IMH patients)

**Tab. 1.** Average classification accuracy for 3-fold cross-validation predictions

| Feature vector length | Kernel |             |            |         |
|-----------------------|--------|-------------|------------|---------|
|                       | Linear | RBF         | Polynomial | Sigmoid |
| 800                   | 0.68   | 0.62        | 0.68       | 0.43    |
| 400                   | 0.71   | 0.84        | 0.73       | 0.43    |
| 200                   | 0.71   | <b>0.89</b> | 0.77       | 0.43    |

**Tab. 2.** Precision, recall and F1-scores for tested classes obtained with the best classification method

| Class            | Precision   | Recall      | F1-score    | Support   |
|------------------|-------------|-------------|-------------|-----------|
| <b>Healthy</b>   | 0.88        | 0.94        | 0.91        | 16        |
| <b>VMT</b>       | 0.94        | 0.79        | 0.86        | 19        |
| <b>IMH</b>       | 0.82        | 1.00        | 0.90        | 9         |
| <b>Avg/Total</b> | <b>0.89</b> | <b>0.89</b> | <b>0.88</b> | <b>44</b> |

### 3 Conclusions

Automatic identification of retinal diseases based on the OCT image processed using machine learning methods is increasingly entering ophthalmic diagnostics [6].

OCT images often require performing appropriate pre-processing, because their quality may be relatively low, especially in the case of data acquisition in three-dimensional modes [5]. The presented results computed with cross-validation give satisfying accuracy for the proposed classification method.

### References

1. Huang, D., Swanson, E.A., et al.: Optical coherence tomography, *Science* 254, pp. 1178–1181 (1991).
2. Duker, J., et al.: The International Vitreomacular Traction Study Group Classification of Vitreomacular Adhesion, Traction, and Macular Hole, *Ophthalmology*, vol. 120, no. 12, pp. 2611–2619 (2013).
3. Stopa, M., Marciniak, E., Rakowicz, P., Stankiewicz, A., Marciniak, T., Dąbrowski, A., Imaging and Measurement of the Preretinal Space in Vitreomacular Adhesion and Vitreomacular Traction by a New Spectral Domain Optical Coherence Tomography Analysis, *Retina*, 37, (10), pp. 1839-1846 (2017).
4. Stankiewicz, A., Marciniak, T., Dąbrowski, A., Stopa, M., Marciniak, E.: A New OCT-based Method to Generate Virtual Maps of Vitreomacular Interface Pathologies, *Proceedings 18th IEEE International Conference Signal Processing (SPA2014)*, pp. 83-88 (2014).
5. Stankiewicz, A., Marciniak, T., Dąbrowski, A., Stopa, P., Rakowicz, M., Marciniak, E. (2015). Improving segmentation of 3D retina layers based on graph theory approach for low quality OCT images, *Metrol. Meas. Syst.*, 23(2), pp. 269–280 (2015).
6. Kermany et al.: Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning, *Cell*, 172(5), pp. 1122-1131 (2018).

# Band selection with Higher Order Multivariate Cumulants for small target detection in hyperspectral images\*

Przemysław Głomb<sup>[0000–0002–0215–4674]</sup> and Krzysztof Domino<sup>[0000–0001–7386–5441]</sup>, Michał Romaszewski<sup>[0000–0002–8227–929X]</sup>, Michał Cholewa<sup>[0000–0001–6549–1590]</sup>

Institute of Theoretical and Applied Informatics, Polish Academy of Sciences  
Bałycka 5, 44-100 Gliwice, Poland, <https://www.iitis.pl>  
{przemg, kdomino, michal, mcholewa}@iitis.pl

**Abstract.** This paper presents an original approach for hyperspectral band selection applied to small target detection. The proposed method is unsupervised (target-independent) and uses Higher Order Cumulants Tensors to derive statistics of the joint distribution of hyperspectral pixels. We present the algorithm, experimental verification on a real-life dataset and discussion of the results.

**Keywords:** Higher Order Cumulants Tensors; Hyperspectral images; Band selection; Small target detection; Anomaly detection; Outlier detection.

## 1 Introduction

Hyperspectral imaging (HSI) systems capture hundreds of narrow spectral channels, usually in the Visual-Near Infrared (VNIR, 400-1000nm) or Short Wave Infrared (SWIR, 1000-2500nm) regions of the electromagnetic spectrum. This data carries information about materials presents in the scene [11]. HSI has many practical uses including remote sensing of vegetation [32], aiding in art conservation [13], cultural heritage analysis [4], forgery detection [27] or gunpowder residue detection [12].

Among many hyperspectral applications [2], a promising one is the target detection. Classic approaches to hyperspectral target detection [24] include derivations of RX or SVDD detectors for unsupervised case and spectral matched filter (SMF) or subspace projections in the supervised case. In the latter case, the methods based on spectral angle (e.g. Spectral Angle Mapper) are also widely used and effective [26]. These methods are often supported by algorithms from many domains, e.g., basic detectors are often used as a part of a more complicated algorithm that includes other Machine Learning approaches [29], dedicated preprocessing and data-window schemes [20]. Hyperspectral band selection is also a common extension to existing methods, applied as preprocessing before classic detectors [30].

---

\* This work was partially supported by the National Science Centre, Poland, project number 2014/15/B/ST6/05204. Authors would like to thank Adam Glos for his assistance in implementation of the described method.

Band selection is an important component of many HSI processing methods, not limited to detection. Depending on whether the training data is available, band selection methods can be divided into supervised approaches that select bands based on training examples [21] and unsupervised methods [34]. While supervised methods can obtain more discriminative features, dependence on training examples may lead to instability of the solution, therefore unsupervised band selection may be more robust [14].

Modelling the data distribution with multivariate Gaussian distribution has many successful application in pattern recognition, e.g., face [33] and gesture [9] classification or feature selection [28]. Hence a number of existing classification and target detection algorithms are based on the multivariate Gaussian model, however in many cases this model does not represent the statistical behaviour of hyperspectral data [1]. This motivates the use of non-Gaussian model for HSI data processing [15]. It is also well-known that data can have a non-Gaussian joint distribution despite Gaussian marginals [7]. However, while joint distribution is important for HSI analysis, it is difficult to estimate [23], therefore approaches based on copulas are employed instead [35].

In our work we use cumulants of multivariate data for band selection. Higher Order Multivariate Cumulants (HOMC) are represented in our work by tensors of order  $d$  ( $d$ -dimensional arrays) [18]. These tensors have an important property: when data has a multivariate Gaussian distribution, every element of a cumulant tensor of order  $d \geq 3$  equals zero [16, 22]. This property can be utilised when searching for non-Gaussian distributed bands, and leads to creation of the Joint Skewness Band Selection (JSBS) [10] procedure, where a Higher Order Singular Value Decomposition (HOSVD) [5] of a 3-order cumulant's tensor is used to create a measure of non-Gaussianity and select the most informative bands.

In this paper we perform an experimental evaluation of band selection methods based on HOMC for small target detection in hyperspectral images. Our main contributions are: (1) We apply Joint Kurtosis Feature Selection (JKFS) [7] to the problem of band selection and show that it can be effectively applied for hyperspectral small target detection. (2) We introduce a new method of band selection, called Joint Hyper Skewness Feature Selection (JHSFS) that is the extension of JKFS. We discuss its properties and show that in some hyperspectral detection scenarios, the proposed method can outperform both JSBS and JKFS. (3) We propose a uniform derivation of  $d$ -order cumulant-based band selection methods, that derives JSBS (order  $d = 3$ ), JKFS (order  $d = 4$ ) and JHSFS (order  $d = 5$ ) as special cases and can be extended to orders  $d > 5$ . (4) We present a comparison of performance evaluation for cumulant-based methods on real-life hyperspectral data.

## 2 Band selection using Higher Order Multivariate Cumulants

We use the bold uppercase notation for a matrix (e.g.  $\mathbf{X}$ ) and an uppercase for its column vector (e.g. the  $i^{\text{th}}$  vector of matrix  $\mathbf{X}$  is  $X_i$ ). We use the lowercase notation for an element of a matrix or a tensor (e.g.  $x_{ij}$ ). The hyperspectral image data can be represented in the form of a 3-mode tensor:  $\mathcal{X} \in \mathbb{R}^{p_x \times p_y \times n}$ , where first two modes correspond to spatial dimensions of pixels while last mode correspond to spectral channels. Following [10] we consider spectral channels as marginals of a multivariate variable

and reflectance values recorded at each given pixel as a realisation of such variable. Since we are analysing multivariate statistics of data, we are ignoring spatial information in data (pixel positions). A conceptually simple approach to search for a sub-set of non-Gaussian marginals would be to test each univariate  $X_i$ , indexed by  $i = 1, \dots, n$ , for normality. However as discussed in [10], such approach is oversimplified since it does not take into account multi-variate non-Gaussianity. This can be explained using a reference to a copula approach [25], because data can have Gaussian marginal distributions and non-Gaussian copula (non-Gaussian cross correlation between marginals). This problem is described in detail in [7].

## 2.1 Applications of HOMC to band selection

Inspired by the Maximum Ellipsoid Volume (MEV) method of feature selection [28] where at each iteration step, one marginal variable is removed in such a way that the determinant of the covariance matrix (second cumulant) is maximised, authors of [10] introduced a method called JSBS (Joint Skewness Band Selection). They argued that by analogy, the determinant of the following matrix:

$$\mathbb{R}^{[n,2]} \ni \mathbf{M}_3 = (\mathcal{C}_3)_{(1)} \left( (\mathcal{C}_3)_{(1)} \right)^\top, \quad (1)$$

measures the information extracted by the 3<sup>rd</sup> cumulant tensor –  $\mathcal{C}_3$ . Eq. (1) uses the first mode unfolding, because the super-symmetric tensor  $\mathcal{C}_3$  unfolding is mode invariant. Based on this assumption, they introduced the target function:

$$f_{JSBS} = \frac{\sqrt{\det(\mathbf{M}_3)}}{(\det \mathcal{C}_2)^{\frac{3}{2}}}, \quad (2)$$

Band selection is then performed, analogically to MEV algorithm, by iteratively removing marginal variables in such way that in every iteration a target function is maximised. The denominator of the Eq. (2) is a normalisation that, according to [10], reduces the risk of selecting highly correlated marginals.

We observed that the higher the cumulant's tensor order  $d$  is, the more it is sensitive to tails in multivariate distribution, where outliers may appear. In addition, there exist datasets for which 4<sup>th</sup> cumulant tensor generalisation, called Joint Kurtosis Feature Selection (JKFS) [7] is more effective than JSBS. This motivates an introduction of a general,  $d$ -cumulant based method. We define a  $d$ -order dependency matrix and this matrix to define the new target function

$$\mathbb{R}^{[n,2]} \ni \mathbf{M}_d = (\mathcal{C}_d)_{(1)} \left( (\mathcal{C}_d)_{(1)} \right)^\top \quad f_d = \frac{\sqrt{\det(\mathbf{M}_d)}}{(\det \mathcal{C}_2)^{\frac{d}{2}}}. \quad (3)$$

In the case of  $d = 3$  our methods simplifies to JSBS, while in the case of  $d = 4$  it simplifies to JKFS. As a 5<sup>th</sup> cumulant is called hyper-skewness, in the case of  $d = 5$  we call our method the Joint Hyper Skewness Feature Selection (JHSFS). While the use of HOMC is sensitive to outliers located in tails of multi-variate distribution, it comes at a cost of higher estimation error [8]. We present the algorithm for band selection based on HOMC below:

**Algorithm 1** Generalised, HOMC-based band selection algorithm

---

```

1: Input: covariance  $\mathcal{C}_2 \in \mathbb{R}^{[n,2]}$ , cumulant tensor  $\mathcal{C}_d \in \mathbb{R}^{[n,d]}$ , target  $f_d$ , retained number of
   bands  $n_{\text{left}} \leq n$ 
2: Output: a subset (index) of bands that carry meaningful information.
3: function FEATURES SELECT( $\mathcal{C}_2, \mathcal{C}_d, f_d, n_{\text{left}}$ )
4:   for  $n' \leftarrow n$  to  $n_{\text{left}}$  do
5:     for  $i \leftarrow 1$  to  $n'$  do
6:        $m_i = f_d(\mathcal{C}_{2(-i)}, \mathcal{C}_{d(-i)})$ 
7:     end for
8:     set  $r : m_r = \max(\{m_1, \dots, m_{n'}\})$             $\triangleright$  remove band  $r$ 
9:      $\mathcal{C}_2 = \mathcal{C}_{2(-r)}$             $\mathcal{C}_d = \mathcal{C}_{d(-r)}$ 
10:    end for
11:   return remaining  $n_{\text{left}}$  bands
12: end function

```

---

### 3 Experimental results and discussion

This section presents an experimental evaluation of band selection methods based on Higher Order Multivariate Cumulants (HOMC) discussed in previous sections. The experiments are performed for three methods of band selection based on HOMC: JSBS (order  $d = 3$ ), JKFS (order  $d = 4$ ), and JHSFS (order  $d = 5$ ). As a reference, results for MEV, and without band selection are provided. For the detector we use the Spectral Angle Mapper (SAM) [19]. Results of experiments are presented in the form of Receiver Operating Characteristic (ROC) curves and the performance of the detector is measured using the Area Under Curve (AUC) measure.

#### 3.1 Hyperspectral dataset

In order to present the detailed examination of the performance of the proposed method, we use the Cuprite<sup>1</sup> hyperspectral dataset. In conformance with a standard procedure, noisy and water absorption bands were removed To reduce computational complexity, we consider last 50 bands that contain the spectral range of interest [6, 3]. The site has been a subject of many experiments, and its geology was mapped in detail [31]. In order to compare our results with these provided in [10], we focus on Buddingtonite deposit detection, which in Cuprite image has known local surface presence around the area nicknamed the ‘Buddingtonite bump’. Based on the [31], a ground truth map was prepared. For independence of the target spectrum from the image, we use the corresponding reference spectrum<sup>2</sup> from the USGS Spectral Library [17].

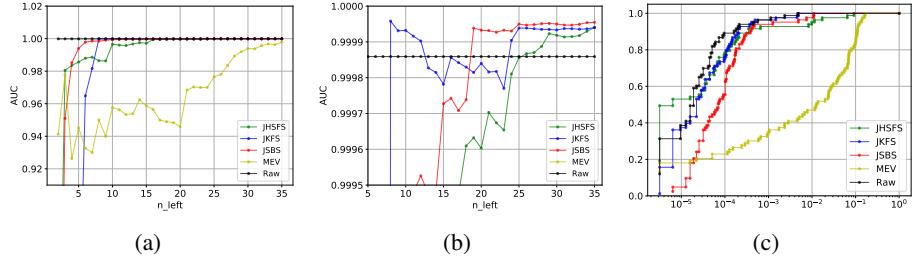
#### 3.2 Results and discussion

The performance of target detector using evaluated methods of band selection is presented in the Figure 1. The detector using HOMC-based methods outperforms the detector based on MEV and for the number of bands greater than  $n_{\text{left}} = 7$ , it also usually

---

<sup>1</sup> Available online at [http://aviris.jpl.nasa.gov/data/free\\_data.html](http://aviris.jpl.nasa.gov/data/free_data.html).

<sup>2</sup> s07\_AV97\_Buddingtnt+Na-Mont CU93-260B\_NIC4b\_RREF



**Fig. 1.** Results of Buddiggonite deposit detection in the Currite dataset compared to the scenario when no band selection is used (black line). Panel (a) presents an Area Under Curve (AUC) as a function of the number of selected bands  $n_{\text{left}}$  for three methods of band selection based on HOMC (JSBS, JKFS, JHSFS) and for band selection with the MEV algorithm. Beginning with values of  $n_{\text{left}} = 8$  bands left, detection with HOMC-based methods outperforms the detection without band selection. The magnification of the vertical axis for this area is presented in the Panel (b). In (c), an illustration of superior performance of the JSHFS if very low false positive probability is required for  $n_{\text{left}} = 14$ .

is better than the detector that does not use band selection. score as detector applied on all bands it can already be viewed as a success, as we reduce the amount of working data. The improved result is thus an additional advantage. From an application point of view, the best performance is achieved by the  $d = 4$  (JKFS) method proposed in this work with  $n_{\text{left}} = 8$ ; it achieves highest score at with the lowest number of retained bands, thus maximizing the detection performance while minimizing the data volume. For large values of the parameter  $n_{\text{left}}$  all methods behave in a similar fashion. However, for small values of the parameter  $n_{\text{left}}$  we can observe sharp breakdowns in the performance of each method: at  $n_{\text{left}} = 4$  bands for the order  $d = 3$  (JSBS), at  $n_{\text{left}} = 7$  bands for the order  $d = 4$  (JKFS) and at  $n_{\text{left}} = 9$  bands for the order  $d = 5$  (JHSFS). The breakdown behaviour is a subject to another study, which is not included here due to the space limitations of this paper. As presented in the Fig. 1(c), when the desired level of false positive rate (FPR) is low, the JHSFS algorithm significantly outperforms other methods, which corresponds to the assumption that it is more sensitive to outliers.

Methods of band selection based on HOMC are a promising approach to hyperspectral small target detection. Our results show that they allow to select a small subset of relevant bands that maintains or improves the performance of the detector while reducing the dimensionality of the data by up to 84%. They also significantly outperform standard methods of band selection such as MEV.

## References

1. Acito, N., Corsini, G., Diani, M.: Statistical analysis of hyper-spectral data: A non-gaussian approach. EURASIP Journal on Applied Signal Processing **2007**(1), 13–13 (2007)
2. Bioucas-Dias, J.M., Plaza, A., Camps-Valls, G., Scheunders, P., Nasrabadi, N., Chanussot, J.: Hyperspectral remote sensing data analysis and future challenges. IEEE Geoscience and remote sensing magazine **1**(2), 6–36 (2013)

3. Clark, R.N., Swayze, G.A., Livo, K.E., Kokaly, R.F., Sutley, S.J., Dalton, J.B., McDougal, R.R., Gent, C.A.: Imaging spectroscopy: Earth and planetary remote sensing with the usgs tetracorder and expert systems. *Journal of Geophysical Research: Planets* **108**(E12) (2003)
4. Cucci, C., Delaney, J.K., Picollo, M.: Reflectance hyperspectral imaging for investigation of works of art: old master paintings and illuminated manuscripts. *Accounts of chemical research* **49**(10), 2070–2079 (2016)
5. De Lathauwer, L., De Moor, B., Vandewalle, J.: A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications* **21**(4), 1253–1278 (2000)
6. Dobigeon, N., Moussaoui, S., Coulon, M., Tourneret, J., Hero, A.O.: Joint bayesian endmember extraction and linear unmixing for hyperspectral imagery. *IEEE Transactions on Signal Processing* **57**(11), 4355–4368 (2009)
7. Domino, K.: Multivariate cumulants in features selection and outlier detection for financial data analysis. arXiv preprint arXiv:1804.00541 (2019)
8. Domino, K., Pawela, Ł., Gawron, P.: Efficient computation of higher-order cumulant tensors. *SIAM J. SCI. COMPUT.* **40**(3), A1590–A1610 (2018)
9. Gawron, P., Głomb, P., Miszczałk, J.A., Puchała, Z.: Eigengestures for natural human computer interface. In: Czachórski, T., Kozielski, S., Stańczyk, U. (eds.) *Man-Machine Interactions 2*. pp. 49–56. Springer Berlin Heidelberg (2011)
10. Geng, X., Sun, K., Ji, L., Tang, H., Zhao, Y.: Joint skewness and its application in unsupervised band selection for small target detection. *Scientific reports* **5**, 9915 (2015)
11. Ghamisi, P., Yokoya, N., Li, J., Liao, W., Liu, S., Plaza, J., Rasti, B., Plaza, A.: Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art. *IEEE Geoscience and Remote Sensing Magazine* **5**(4), 37–78 (2017)
12. Głomb, P., Romaszewski, M., Cholewa, M., Domino, K.: Application of hyperspectral imaging and machine learning methods for the detection of gunshot residue patterns. *Forensic Science International* **290**, 227–237 (2018)
13. Grabowski, B., Masarczyk, W., Głomb, P., Mendys, A.: Automatic pigment identification from hyperspectral data. *Journal of Cultural Heritage* **31**, 1–12 (2018)
14. Jia, S., Tang, G., Zhu, J., Li, Q.: A novel ranking-based clustering approach for hyperspectral band selection. *IEEE Transactions on Geoscience and Remote Sensing* **54**(1), 88–102 (2016)
15. Kajić, V., Považay, B., Hermann, B., Hofer, B., Marshall, D., Rosin, P.L., Drexler, W.: Robust segmentation of intraretinal layers in the normal human fovea using a novel statistical model based on texture and shape analysis. *Optics express* **18**(14), 14730–14744 (2010)
16. Kendall, M.G., et al.: The advanced theory of statistics. The advanced theory of statistics. (1946)
17. Kokaly, R., Clark, R., Swayze, G., Livo, K., Hoefen, T., Pearson, N., Wise, R., Benzel, W., Lowers, H., Driscoll, R., Klein, A.: USGS spectral library version 7. Tech. rep., U.S. Geological Survey Data Series 1035 (2017)
18. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. *SIAM review* **51**(3), 455–500 (2009)
19. Kruse, F., Lefkoff, A., Boardman, J., Heidebrecht, K., Shapiro, A., Barloon, P., Goetz, A.: The spectral image processing system (sips)—interactive visualization and analysis of imaging spectrometer data. *Remote Sensing of Environment* **44**(2), 145 – 163 (1993), airborne Imaging Spectrometry
20. Kwon, H., Der, S.Z., Nasrabadi, N.M.: Adaptive anomaly detection using subspace separation for hyperspectral imagery. *Optical Engineering* **42**(11), 3342–3352 (2003)
21. Li, W., Prasad, S., Fowler, J.E., Bruce, L.M.: Locality-preserving dimensionality reduction and classification for hyperspectral image analysis. *IEEE Transactions on Geoscience and Remote Sensing* **50**(4), 1185–1198 (2012)
22. Lukacs, E.: Characteristic functions. Charles Griffin and Co., Ltd., London (1970)

23. Martínez-Usó, A., Pla, F., Sotoca, J.M., García-Sevilla, P.: Clustering-based hyperspectral band selection using information measures. *IEEE Transactions on Geoscience and Remote Sensing* **45**(12), 4158–4171 (2007)
24. Nasrabadi, N.M.: Hyperspectral target detection: An overview of current and future challenges. *IEEE Signal Processing Magazine* **31**(1), 34–44 (2014)
25. Nelsen, R.B.: An introduction to copulas. Springer Science & Business Media (2007)
26. Pengra, B.W., Johnston, C.A., Loveland, T.R.: Mapping an invasive plant, phragmites australis, in coastal wetlands using the eo-1 hyperion hyperspectral sensor. *Remote Sensing of Environment* **108**(1), 74–81 (2007)
27. Rodionova, O.Y., Houmøller, L.P., Pomerantsev, A.L., Geladi, P., Burger, J., Dorofeyev, V.L., Arzamastsev, A.P.: NIR spectrometry for counterfeit drug detection: a feasibility study. *Analytica Chimica Acta* **549**(1), 151–158 (2005)
28. Sheffield, C.: Selecting band combinations from multispectral data. *Photogrammetric Engineering and Remote Sensing* **51**, 681–687 (1985)
29. Shi, Z., Yu, X., Jiang, Z., Li, B.: Ship detection in high-resolution optical imagery based on anomaly detector and local shape feature. *IEEE Transactions on Geoscience and Remote Sensing* **52**(8), 4511–4523 (2014)
30. Sun, K., Geng, X., Ji, L.: A new sparsity-based band selection method for target detection of hyperspectral image. *IEEE Geoscience and Remote Sensing Letters* **12**(2), 329–333 (2015)
31. Swayze, G., Clark, R., Goetz, A., Livo, K., Breit, G., Kruse, F., Stutley, S., Snee, L., Lowers, H., Post, J., Stoffregen, R., Ashley, R.: Mapping advanced argillic alteration at Cuprite, Nevada using imaging spectroscopy. *Economic Geology* **109**(5), 1179–1221 (2014)
32. Thenkabail, P.S., Lyon, J.G.: Hyperspectral remote sensing of vegetation. CRC Press (2016)
33. Turk, M., Pentland, A.: Eigenfaces for recognition. *Journal of cognitive neuroscience* **3**(1), 71–86 (1991)
34. Yuan, Y., Lin, J., Wang, Q.: Dual-clustering-based hyperspectral band selection by contextual analysis. *IEEE Transactions on Geoscience and Remote Sensing* **54**(3), 1431–1445 (2016)
35. Zeng, X., Durrani, T.S.: Band selection for hyperspectral images using copulas-based mutual information. In: *Statistical Signal Processing, 2009. SSP'09. IEEE/SP 15th Workshop on*. pp. 341–344. IEEE (2009)

# Unsupervised deep learning approach to hyperspectral anomaly detection\*

Bartosz Grabowski<sup>[0000–0002–2364–6547]</sup>, Przemysław Głomb<sup>[0000–0002–0215–4674]</sup>,  
Michał Romaszewski<sup>[0000–0002–8227–929X]</sup>, and Mateusz  
Ostaszewski<sup>[0000–0001–7915–6662]</sup>

Institute of Theoretical and Applied Informatics, Polish Academy of Sciences  
Bałycka 5, 44-100 Gliwice, Poland, <https://www.iitis.pl>  
`{bgrabowski, przemg, michal, mostaszewski}@iitis.pl`

**Abstract.** We investigate a simple, but efficient training scheme that adapts a classification-oriented deep learning network for unsupervised anomaly detection. Through procedurally generated artificial class labelling, we obtain a background model required for anomaly detection. A study of a number of state of art network architectures on several real-life hyperspectral images shows that proposed method provides high accuracy solution to this problem.

**Keywords:** Deep learning · Hyperspectral imaging · Anomaly detection · Unsupervised methods.

## 1 Introduction

Hyperspectral imaging (HSI), or imaging spectroscopy is a technique of acquiring images where each pixel contains a high number of narrow spectral measurements, usually within the 400 – 2500nm wavelength range. This data allows to perform identification of the composition or state of minerals present in the scene [5]. HSI has many practical uses including remote sensing of plants [19], art conservation [8], or forgery detection [18].

Anomaly detection (AD) methods identify patterns in data that do not correspond to a typical or normal behaviour, e.g. noise, outliers or novelty in the signal [4]. It is also an important problem in hyperspectral image analysis [6]. The traditional approach is to assume a Gaussian model and use the likelihood ratio test for detection, denoted Reed-Xiaoli (RX) detector [17]. More advanced methods attempt to model the actual data properties, e.g. with kernel data modelling [2], spatial-spectral processing [20] or sparse representations [11]. An overview of the methods can be found in e.g. [15, 6].

Deep learning combines simple representations into complex features [7]. This class of algorithms has been very successful for computer vision problems, most notably

---

\* This work has been partially supported by the project ‘Application of transfer learning methods in the problem of hyperspectral images classification using convolutional neural networks’ funded from the Polish budget funds for science in the years 2018–2022, as a scientific project under the „Diamond Grant” program, no. DI2017 013847 and by the Polish National Science Center scholarship 2018/28/T/ST6/00429.

classification tasks [10]. Deep learning has been applied to hyperspectral data either by adaptation of a general architecture (e.g. [9]) or by a dedicated approach (e.g. [21]). A recent survey compares the performance of a number of networks in hyperspectral classification scenarios [1].

Hyperspectral anomaly detection with deep learning methods is a new and open field. Published current approaches adapt existing architectures: Deep Belief Network (DBN) [13] or Sparse Autoencoder (SAE) [12] for local background modelling. Current overview of deep learning for anomaly detection [3] shows that while approaches share many common themes, in most cases specialized architectures are proposed. In contrast we propose a learning scheme that is architecture agnostic, simple and effective, and with good generalization potential.

Our approach is based on the transfer learning approach from [14]. We propose an unsupervised artificial labelling scheme that forces the network to learn background model as one of the classes. After training, the feature responses on the late network stages are translated to anomaly score. We test this approach with a number of architectures from [1] and the experiment setting from [6] and achieve promising performance.

## 2 Method

Our motivation stems from the observation that a part of weights in the deep network tend to converge to certain shapes, dependent more on the signal domain than class labels [7] (e.g. universal emergence of Gabor-like filters in image classification). Our experiments with unsupervised transfer learning [14] show that using a priori defined artificial class labelling can significantly improve the accuracy of the classification. This is in accordance with general research, that both low- and high-level features describing the data can be learned during unsupervised training (e.g. self-taught learning [16]). Based on this observations, we propose a simple anomaly detection scheme using an existing classifier and an artificial class assignment.

For our approach to anomaly detection, we first create an artificial label map, by labelling a random number of pixels as anomalies (e.g.  $n_a = 5\%$ ), and the rest as background. We train the network using a regular classification scheme. Finally we evaluate the estimate of probability of either the background or anomaly class, taken from the last layer of the network, as the anomaly score for the image.

In our experiments, we use network architectures studied in the work [1], for which the authors have made available the source code for each method configuration in the PyTorch framework<sup>1</sup>. We use the following models (name codes refer to the respective publications' authors, see [1]): NN, HAMIDA, LEE, HE, HU, LI, LIU. Utilized networks can be divided into three groups according to the method of hyperspectral image analysis as a three-dimensional tensor. So called 1D networks are based on convolutional kernels which work only on the one pixel along spectral dimension. Therefore their inputs are single pixels. Second group (2-D+1-D networks) consists of convolutional neural networks which have two types of kernels. One type works on the spatial (2D) dimensions of the image. Second one works on the spectral dimension (1D).

---

<sup>1</sup> <https://github.com/nshaud/DeepHyperX>.

Last group (3-D networks) utilize three dimensional convolutional kernels. It should be noted, that most of architectures in this group also uses 2D and 1D kernels. Moreover, most of them apply fully connected layers to process the extracted features from convolutional layers and perform classification step.

### 3 Results and discussion

We use experimental setting based on [6]. We use three hyperspectral datasets: HYDICE Urban<sup>2</sup>, HyMap Cooke City<sup>3</sup> and AVIRIS San Diego. All three span wavelength range 400–2500 nm and contain ground truth anomalies for verification. Each evaluated method has an unlabelled image as an input and outputs the anomaly score. The scores are compared as ROC curves. Two standard reference methods (global RX and One Class SVM with RBF kernel) were included for comparison. The deep networks were trained with standard parameters from [1] with  $n_a = 5\%$  artificial anomaly labels randomly (uniformly) spread over the image.

Detection results are presented as ROC curves In Fig. 1. Every architecture is represented by a curve with median value of Area Under Curve (AUC) over  $n = 5$  runs. Datasets used in experiments present different challenges for algorithms. Hydice urban (HU): small, full-pixel anomalies (cars on road/parking), significantly different from their background. San Diego (SD): larger, full-pixel anomalies (planes on airport). Cooke City (CC): subpixel, rare anomalies (colour markers placed in specific locations). For every dataset there are architectures that are comparable or significantly outperform classic anomaly detectors. It seems that some architectures (NN, LEE) are sensitive to full-pixel anomalies, while others, such as HAMIDA or HE are sensitive to subpixel-level rare anomalies. For most architectures the majority of results were stable over multiple runs. The only observed issue was some level discrepancy between probability output for artificial classes, i.e. one of the class adapts better in terms of result score.

### 4 Conclusions

In this paper we present an original approach for adapting a deep learning hyperspectral classification network for anomaly detection in an unsupervised fashion. Our results are promising; the proposed method, while unsupervised and simple is able to outperform state of the art anomaly detectors. Further works will concentrate on examining in detail which elements of the training procedure are have greatest impact on the results, stability analysis across multiple datasets and architectures, and theoretical investigation into the properties of unsupervised training.

---

<sup>2</sup> <http://lesun.weebly.com/hyperspectral-data-set.html>

<sup>3</sup> <http://dirsapps.cis.rit.edu/blindtest>

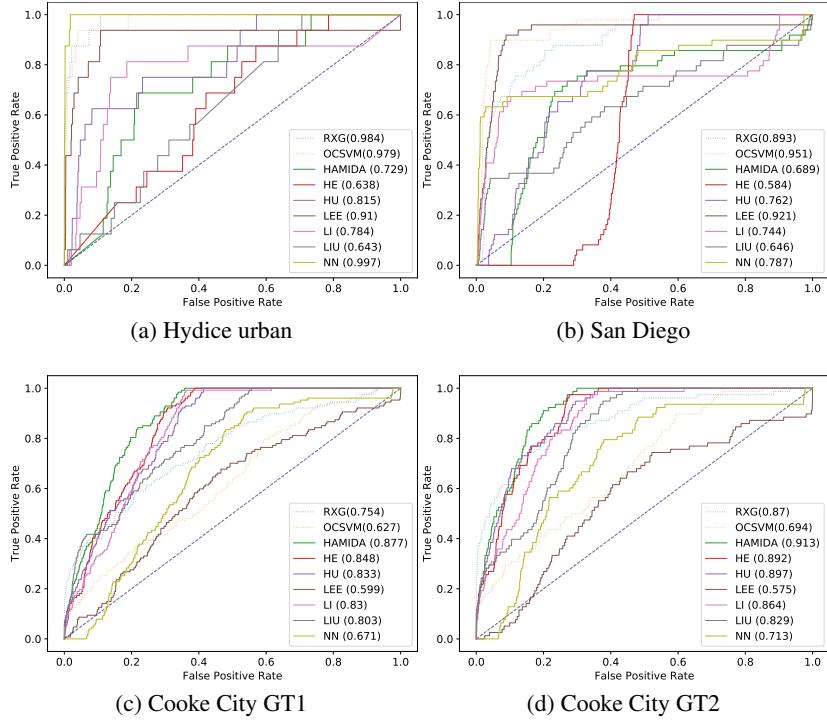


Fig. 1: Detection performance in the form of receiver operating characteristic (ROC) curves for tested detectors and datasets. Value in brackets near the detector's name is the Area Under curve (AUC). Two versions of results for the the Cooke City dataset correspond to the two target maps.

## References

1. Audebert, N., Le Saux, B., Lefevre, S.: Deep learning for classification of hyperspectral data: A comparative review. *IEEE Geoscience and Remote Sensing Magazine* **7**(2), 159–173 (2019)
2. Banerjee, A., Burlina, P., Diehl, C.: A support vector method for anomaly detection in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing* **44**(8), 2282–2291 (2006)
3. Chalapathy, R., Chawla, S.: Deep learning for anomaly detection: A survey. arXiv preprint arXiv:1901.03407 (2019)
4. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. *ACM Computing Surveys* **41**(3), 15:1–15:58 (2009)
5. Ghamisi, P., Yokoya, N., Li, J., Liao, W., Liu, S., Plaza, J., Rasti, B., Plaza, A.: Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art. *IEEE Geoscience and Remote Sensing Magazine* **5**(4), 37–78 (2017)
6. Głomb, P., Romaszewski, M.: Anomaly detection in hyperspectral remote sensing images. In: Pandey, P., Srivastava, P., Balzter, H., Bhattacharya, B., Petropoulos, G. (eds.) *Hyperspectral Remote Sensing: Theory & Applications*. Elsevier (accepted for publication)

7. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016)
8. Grabowski, B., Masarczyk, W., Głomb, P., Mendys, A.: Automatic pigment identification from hyperspectral data. *Journal of Cultural Heritage* **31**, 1–12 (2018)
9. Han, M., Cong, R., Li, X., Fu, H., Lei, J.: Joint spatial-spectral hyperspectral image classification based on convolutional neural network. *Pattern Recognition Letters* (2018)
10. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Curran Associates, Inc. (2012)
11. Li, J., Zhang, H., Zhang, L., Ma, L.: Hyperspectral anomaly detection by the use of background joint sparse representation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **8**(6), 2523–2533 (2015)
12. Ma, N., Peng, Y., Wang, S.: On-line hyperspectral anomaly detection with hypothesis test based model learning. *Infrared Physics & Technology* **97**, 15 – 24 (2019)
13. Ma, N., Peng, Y., Wang, S., Leong, P.H.W.: An unsupervised deep hyperspectral anomaly detector. *Sensors* **18**(3) (2018)
14. Masarczyk, W., Głomb, P., Grabowski, B., Ostaszewski, M.: Effective transfer learning for hyperspectral image classification with deep convolutional neural networks. *arXiv preprint arXiv:1909.05507* (2019)
15. Matteoli, S., Diani, M., Corsini, G.: A tutorial overview of anomaly detection in hyperspectral images. *IEEE Aerospace and Electronic Systems Magazine* **25**(7), 5–28 (2010)
16. Raina, R., Battle, A., Lee, H., Packer, B., Ng, A.Y.: Self-taught learning: Transfer learning from unlabeled data. In: *Proceedings of the 24th International Conference on Machine Learning*, pp. 759–766. ICML '07, ACM (2007). <https://doi.org/10.1145/1273496.1273592>
17. Reed, I.S., Yu, X.: Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution. *IEEE Transactions on Acoustics, Speech, and Signal Processing* **38**(10), 1760–1770 (1990)
18. Rodionova, O.Y., Houmøller, L.P., Pomerantsev, A.L., Geladi, P., Burger, J., Dorofeyev, V.L., Arzamastsev, A.P.: NIR spectrometry for counterfeit drug detection: a feasibility study. *Analytica Chimica Acta* **549**(1), 151–158 (2005)
19. Thenkabail, P.S., Lyon, J.G.: Hyperspectral remote sensing of vegetation. CRC Press (2016)
20. Wang, Y., Lee, L., Xue, B., Wang, L., Song, M., Yu, C., Li, S., Chang, C.: A posteriori hyperspectral anomaly detection for unlabeled classification. *IEEE Transactions on Geoscience and Remote Sensing* **56**(6), 3091–3106 (2018)
21. Zhao, G., Liu, G., Fang, L., Tu, B., Ghamisi, P.: Multiple convolutional layers fusion framework for hyperspectral image classification. *Neurocomputing* (2019)

# Break the curse of small datasets in computer vision tasks with transfer learning methods

Joanna Jaworek-Korjakowska<sup>1</sup>[0000-0003-0146-8652], Andrzej Brodzicki<sup>1</sup>[0000-0001-7713-526X], Dariusz Kucharski<sup>1</sup>[0000-0002-0107-2407], Michał Piekarski<sup>1,2</sup>[0000-0001-9391-4263], and Marek Gorgoń<sup>1</sup>[0000-0003-1746-1279]

<sup>1</sup> Department of Automatic Control and Robotics,  
AGH University of Science and Technology, Krakow, Poland  
[jaworek@agh.edu.pl](mailto:jaworek@agh.edu.pl)

<sup>2</sup> SOLARIS National Synchrotron Radiation Centre, Jagiellonian University,  
Krakow, Poland

**Abstract.** The fundamental problem of using deep learning methods to solve machine vision challenges is the amount and quality of data. To address this issue, we investigate the transfer learning method. In this study, we introduce two different strategies of transfer learning and present the widely-used models. Furthermore, we describe three different experiments from computer vision that confirm the developed algorithms ability to classify images with overall accuracy 87.2-95% which is a state-of-the-art result in melanoma thickness prediction and anomaly detection tasks.

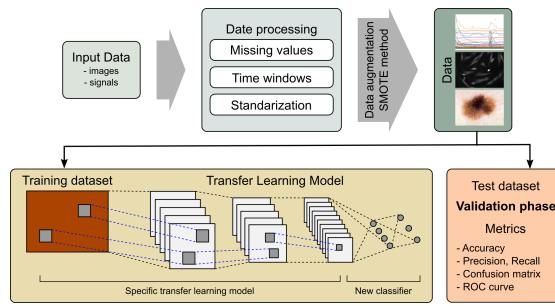
**Keywords:** Deep neural networks · Transfer learning · Signal processing · Image analysis · Anomaly detection

## 1 Introduction to Transfer Learning Methods

The fundamental problem of artificial intelligence including machine learning and deep learning methods is the amount and quality of data. While the process of gathering the data might be unexpectedly expensive or even impossible, there is an idea of sharing, not the data itself, but the machine learning model which actually has 'seen' the data. This kind of approach is called transfer learning. In other words, transfer learning refers to a process where a neural network model is firstly trained on a problem similar to the problem that is being solved. Thanks to the fact that the lower layers of a CNN can typically detect common patterns like lines and edges, the middle layers learn filters that detect parts of objects, while the last layers learn to recognize full objects, in different shapes and positions, the knowledge gained may be reused [2]. The definition of transfer learning is given in terms of domain and task. Given a source domain  $\mathcal{D}_S$  and learning task  $\mathcal{T}_S$ , a target domain  $\mathcal{D}_T$  and learning task  $\mathcal{T}_T$ , transfer learning aims to help improve the learning of the target predictive function  $f_T(\cdot)$  in  $\mathcal{D}_T$  using the knowledge in  $\mathcal{D}_S$  and  $\mathcal{T}_S$ , where  $\mathcal{D}_S \neq \mathcal{D}_T$ , or  $\mathcal{T}_S \neq \mathcal{T}_T$ .

## 2 Deep Transfer Learning Models

One fair question to answer is that 'accuracy not only depends on the network but also on the amount of data available for training'. It has been widely proved that for traditional machine learning algorithms, performance grows according to a power law and then reaches a plateau, while deep learning performance scales with increasing data size. A rapid development could have been noticed in the field of deep learning causing a huge production of many models including AlexNet, Inception, ResNet and VGG which have been shortly described (Fig. 1).



**Fig. 1.** Transfer learning diagram for computer vision tasks including following steps: data processing, data augmentation, training and test data division, training process of the classification layer, and validation phase.

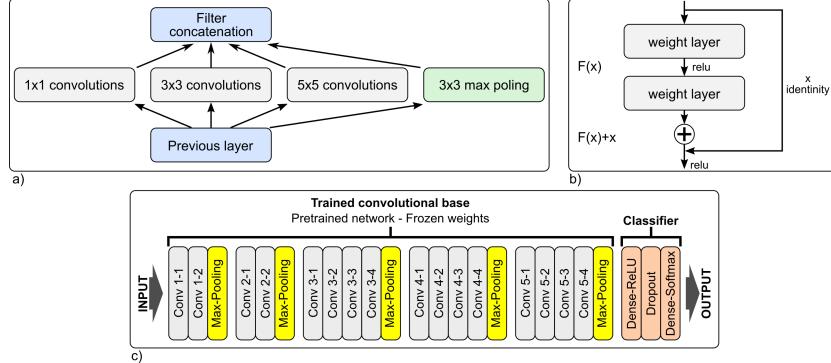
**AlexNet architecture** is a pretty simple model composed of 5 convolutional layers followed by maxpooling layers as the feature extraction part, and 3 fully connected layers with Softmax closing the classification process.

Nowadays **VGG model** is considered to be one of the best models for transfer learning in image recognition tasks because of its simple architecture in general and clarified approaches in creating consecutive layers [1]. The VGG-19 model has roughly 143 million parameters and contains 19 trainable layers including convolutional as well as fully connected layers, max pooling, and dropout (Fig. 2c).

The quintessence of the **Inception model** is to connect several layers parallel in a kind of block instead of stacking up one on another (Fig. 2a). It was assumed that a network which utilizes such an approach will choose the most useful layers rising its weights, while decreasing useless layers at the same time. Moreover,  $1 \times 1$  convolution has been introduced, which helps reducing the feature-map dimension and global average pooling.

In deep learning networks, a residual learning framework (**ResNet Model**) helps to preserve good results through a neural network with many layers. One problem that commonly occurs is degradation and vanishing gradients. The deep

residual network deals with some of these problems by using residual blocks, which take advantage of residual mapping to preserve inputs (Fig. 2b).



**Fig. 2.** Transfer learning models: a) An Inception block example, b) Residual learning building block, c) Schematic overview of the VGG-19 network architecture.

### 3 Classification problems in machine vision with small datasets

To successfully train a deep neural network in machine vision tasks a diverse datasets is needed. The question arises: *can transfer learning be the answer to this issue?* We present three examples where transfer learning solved the problem of small datasets. Table 1 gives a summary of all described projects.

**Table 1.** Summary of the described research projects using transfer learning methods.

| Title                | Dataset (samples)  | Accuracy [%] |
|----------------------|--------------------|--------------|
| Anomaly detection    | 10.000(imbalanced) | 95           |
| Thickness prediction | 244                | 87           |
| Cell classification  | 369                | 93           |

#### 3.1 Anomaly detection solution

Deep convolutional pretrained neural network VGG-19 was used to detect abnormal situations in multivariate diagnostic signals. Motivation of this research was to increase synchrotron's beam stability by detecting anomalies in different subsystems of the machine. The general idea is shown in Fig. 1. The input signals were pressure readings from the storage ring. Due to the fact that anomalies are

rare, problem with the predominant number of samples in non-anomaly dataset occurred. The model achieved very good results, reaching accuracy of 95% by only 10.000 examples in the dataset.

### 3.2 Melanoma Thickness prediction

Thickness is one of the most important factor in melanoma prognosis. To address this problem, we have implemented an effective computer-vision based deep learning tool that can perform the preoperative evaluation. The novelty of our approach is that we directly predict the thickness into one of three classes: less than 0.75 mm, 0.76-1.5 mm, and greater than 1.5 mm. We have used transfer learning of the pretrained, adapted to our application VGG-19 convolutional neural network (CNN) with an adjusted densely-connected classifier. Our database contained only 244 dermoscopy images. Experiments confirm the ability to classify skin lesion thickness with 87.2% overall accuracy what is a state-of-the-art result in melanoma thickness prediction.

### 3.3 Biomedical image classification

An algorithm for automatic classification of *Clostridium Difficile* bacteria cytotoxicity was proposed. 369 fluorescent images depicting both dead and alive human cells were first segmented using classical image processing methods. Three convolutional neural network architectures (mentioned before in 2) were used and compared. The best one, ResNet50, achieved 93% accuracy as well as 93% sensitivity and 94% specificity, with other two being only slightly worse.

## 4 Conclusion and discussion

Presented examples show that transfer learning is a universal method, that may be applied to solve different challenging tasks. However, transfer learning methods have also many limitations. Currently, one of the biggest challenges of transfer learning is the problem of negative transfer, the distribution of the training data which are used to pretrain the model should not vary too much from the test data, and the data should not overfit the model.

**Acknowledgment:** This scientific work was financially supported by AGH University of Science and Technology Status Funds on Decision No. 16.16.120.773.

## References

1. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. ArXiv:1409.1556 (2014)
2. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2. pp. 3320–3328. MIT Press, Cambridge, MA, USA (2014)

# Semisupervised Segmentation using Autoencoder - Comparison of Convergence for Manual and Random Assignments

Przemysław Mazurek<sup>1</sup>[0000–0002–7145–3993],  
Dorota Oszutowska-Mazurek<sup>2</sup>[0000–0001–6403–4160], and  
Oktawian Knap<sup>3</sup>

<sup>1</sup> West Pomeranian University of Technology Szczecin, Department of Signal Processing and Multimedia Engineering, 26. Kwietnia 10 St., 71126 Szczecin, Poland  
[przemyslaw.mazurek@zut.edu.pl](mailto:przemyslaw.mazurek@zut.edu.pl)

<sup>2</sup> Pomeranian Medical University, Department of Histology and Embryology,  
70111 Szczecin, Powstancow Wlkp. 72 St., Szczecin, Poland  
[adorotta@pum.edu.pl](mailto:adorotta@pum.edu.pl)

<sup>3</sup> Pomeranian Medical University, Department of Forensic Medicine, Szczecin, Poland

**Abstract.** Semisupervised learning using autoencoder with local histograms as an input is considered for the segmentation of bone marrow images. Preliminary analysis of the results shows advantages of manual selection of small regions with assignment to appropriate class comparing to random selection of regions. Interesting result is multi modal histogram of segmentation error with three peaks.

**Keywords:** Neural Networks · Autoencoder · Semisupervised Learning · Bone Marrow

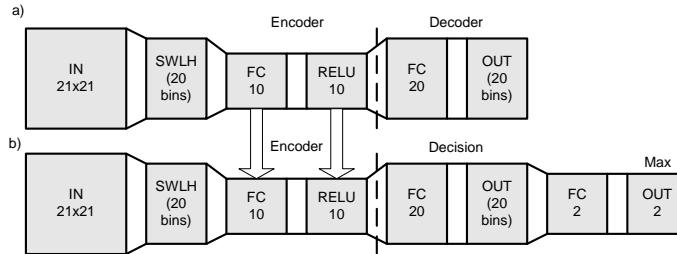
## 1 Introduction

Image segmentation is challenging task and very important for numerous practical applications. The most difficult is the segmentation of microscopic images due to the data size and complexity of structures. Supervised segmentation using neural networks (including deep convolution networks) is the most powerful tool but limited by dataset preparation. Achieved algorithm is dedicated to particular image type and requires enormous effort in manual image segmentation that is the main problem of this approach. Unsupervised pattern recognition algorithms allow finding structures and the segmentation without human effort is possible for specific image types. Alternative approach is semisupervised learning with Human in the Loop, so the image is processed with some human effort. This type of pattern recognition is considered in this work. Semisupervised learning uses iterative process. A human selects regions or specific features in image and assigns them to appropriate classes. A very small portion of image is segmented manually in this way and after this segmentation algorithm is trained. Obtained results are presented and additional regions or features are added by a human

for fixing them and the next learning iteration starts again. Final segmentation is achieved after a few steps typically. Achieved pattern recognition algorithm could be used for other similar images. Semisupervised learning requires the selection of representative regions. Such selection is compared in this work with random selection. The HistAENNseg software [3] is used as a GUI for semisupervised training. This tool uses mouse for the painting of specific classes as well as front-end for neural network software. Dlib [1] is used for training neural network using GPGPU.

HistAENNseg uses autoencoder for unsupervised segmentation using 1% of tiles with  $21 \times 21$  size. The input layer is preserved after training of the autoencoder in the next supervised training process. This process uses segmentation data from a human and after the second training the whole image is processed and presented to a human for acceptance or corrections.

The most critical problem in the segmentation is that the neural network must be very large for processing tiles that could be oriented in any direction. The obtaining of rotational invariance by neural network requires extended processing time and large number of neurons in particular layers. The processing of such training data extends time between human interactions and degrades the workflow. HistAENNseg uses dedicated approach based on processing rotational invariant the data by preprocessing of tiles. Histogram of pixel values from tile is delivered to input layer (and output for autoencoder phase) instead of raw data. The size of the tile influences final quality of the segmentation and should be carefully tested. An example network used in HistAENNseg has the structure shown in Fig. 1.

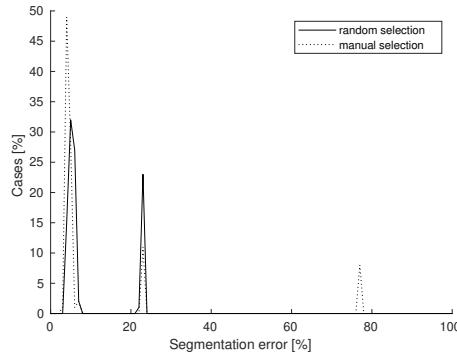


**Fig. 1.** Scheme of HistAENN architecture for both training phases (RELU—Rectified Linear Unit Layer, FC—Full Connection Layer, SWLH—Sliding Window Local Histogram) ((a) autoencoder phase, (b) classifier phase).

## 2 Comparison of Human and Random Selections

Bone marrow images are used with two classes [2] in this work. Images are unbalanced during the autoencoder training because trabeculas occupies about 10% of image and most of image region is the background and other features. The

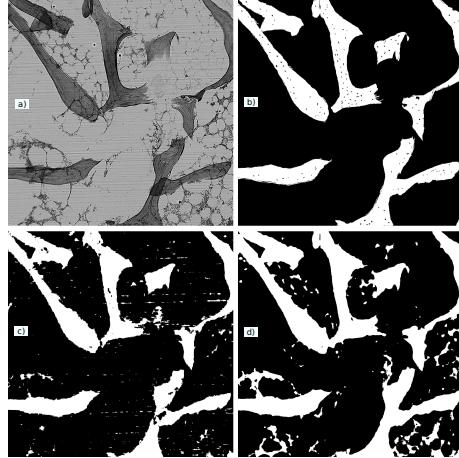
availability of detailed manual segmentation allows automatic generation training samples for the second phase using random number generator. Both classes use 1000 samples and automatic analysis of the segmentation of the algorithm is possible. Alternative approach assumes single only attempt of a human with the selection similar size of training samples. Achieved segmentation results are compared with reference after second phases for random and human selections. Such comparison is interesting from the research point-of-view, because starting point for the optimization process (training process) is very important. There are 100 tests for random selection and 100 test for human selection obtained from 10 manual selections (each manual selection is used for 10 tests) for single image. Manual selections use different strategies related to the features, regions, density, distance to edges. Random selections are dispersed due to the uniform random generator used for image sampling. Example histogram of errors for both selection strategies is shown in Fig. 2. Exemplary results for error near 4% for both selection methods are shown in Fig. 3.



**Fig. 2.** Histograms of error after training.

### 3 Discussion

There are three groups of errors. Left group is related to minimal error achieved after the first iteration. It shows the advantages of manual selection over random selection of training samples, because low segmentation errors (binary difference between achieved mask and reference mask) are achieved with higher probability. The second group is related to very large errors and this group is about two times more probable for random selection. Such segmentation results are not sufficient from user's perspective. The third group is related to convergence problem and is observed for manual selection only. Obtained segmentation image is single value. This group is some surprise, but single value output image could be detected by the algorithm and training process could be restarted for fixing problem. It should be noted that the third group is not specific to single



**Fig. 3.** Segmentation after first iteration (a - original image, b - mask, c - segmentation using HistAENNseg and manual selection, d - segmentation using HistAENNseg and random selection).

manual segmentation. Analysis of results (Fig. 3) shows better control of artifacts for manual selection. There are some artifacts, but shape of trabeculas is achieved. Random selection leads to the selection of other features such the object is classified as bone marrow.

#### 4 Further Work

Further work will be related to the analysis of a few interactions for manual and random selections for convergence analysis. The interesting question about convergence is the mixing of both methods for achieving alternative approach.

**Acknowledgement.** We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X GPU used for this research.

#### References

1. King, D.E.: Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research* **10**, 1755–1758 (2009)
2. Oszutowska-Mazurek, D., Knap, O.: The use of deep learning for segmentation of bone marrow histological images. In: Silhavy, R., Senkerik, R., Kominkova Oplatkova, Z., Prokopova, Z., Silhavy, P. (eds.) *Artificial Intelligence Trends in Intelligent Systems*. pp. 466–473. Springer International Publishing (2017)
3. Oszutowska-Mazurek, D., Mazurek, P., Parafiniuk, M., Stachowicz, A.: Method-induced errors in fractal analysis of lung microscopic images segmented with the use of histaenn (histogram-based autoencoder neural network). *Applied Sciences* **8**(12) (2018)

# Application of GGD Based Preprocessing for Region Based Binarization of Degraded Document Images

Robert Krupiński , Piotr Lech ,  
Hubert Michalak , and Krzysztof Okarma 

Department of Signal Processing and Multimedia Engineering,  
Faculty of Electrical Engineering  
West Pomeranian University of Technology in Szczecin,  
Sikorskiego 37, 70-313 Szczecin, Poland  
`{rkrupinski,piotr.lech,michalak.hubert,okarma}@zut.edu.pl`

**Abstract.** A fast and reliable binarization of degraded document images may be conducted using relatively simple methods, avoiding the necessity of the analysis of the neighbourhood of each pixel typical for adaptive methods. Since the additional improvement of binarization accuracy may be obtained using an appropriate image preprocessing, the application of previously proposed method based on Generalized Gaussian Distribution (GGD) for region based binarization is presented in this paper. The results obtained for region based methods, proposed recently as the balance between the global and adaptive local thresholding, are also compared with typical binarization algorithms with and without the GGD based preprocessing, using well-known DIBCO datasets.

**Keywords:** Document images · Image binarization · Generalized Gaussian Distribution · Monte Carlo method · Thresholding.

## 1 Introduction

One of the most demanding areas of research related to image thresholding, apart from the natural images captured in varying lighting conditions, is the binarization of degraded document images. Many new algorithms are presented during DIBCO competitions held yearly during the leading conferences in this field (ICDAR and ICFHR), where the demanding DIBCO datasets are used for the performance evaluation of binarization algorithms for machine printed and handwritten document images. Since similar methods may be used for natural images e.g. in self-localization and navigation of mobile robots based on machine vision as well as modern autonomous vehicles, they may also be useful for modern Industry 4.0 solutions utilising machine vision.

The most popular image binarization methods may be divided into two major groups, namely global methods, such as Otsu or Kapur, and adaptive local thresholding represented by the algorithms proposed e.g. by Niblack, Sauvola

or Bradley, leading to much better results. Some more advanced methods based on local features and Gaussian mixtures [3] and deep neural networks [5] have been proposed, which require much higher computational efforts due to multiple stages including e.g. background removal, median filtering and morphological processing or long training process.

In the recent paper [1] the application of the GGD based preprocessing for the increase of the performance of some known methods was proposed, whereas in the other article [2] some applications of region based approaches were examined. Hence, the motivation of this paper is to bring together these two approaches and verify the possibilities of their combination using the DIBCO datasets.

## 2 Proposed Approach

The application of the Generalized Gaussian Distribution based preprocessing with additional acceleration using the Monte Carlo method, proposed in the paper [1], is based on the assumption of the similarity of histograms of degraded document images and binary images corrupted by noise of a normal distribution, visible also for real scanned images of historical documents. Therefore, it may be assumed that the real image is similar to the sum of ground truth (GT) binary image with Gaussian noise. Hence, the removal of partial information related to noise, being the right part of the histogram assuming the presence of the dark text on a brighter background, was proposed. The threshold using in this preprocessing, followed by normalization, is determined as the location parameter of the GGD ( $I_{max} = \mu_{GGD}$ ) used for the approximation of the image histogram. The definitions of the parameters and a more detailed analysis can be found in the paper [1].

As the calculation of the GGD parameters for the whole image is relatively slow, the approximated histogram of the image may be determined using the reduced number of samples to speed up the computations, according to the Monte Carlo method. To preserve a rough shape of the histogram and the statistical properties of the image, samples should be chosen randomly assuming their uniform distribution on the image plane. The experiments conducted for the images from DIBCO datasets showed that even for 100 samples the value of the location parameter of the GGD is mostly the same or changes only by 1 or 2 luminance levels. For further increase of the processing speed some predefined numbers od pixels obtained from the pseudorandom number generator with a uniform distribution may be used.

The partial removal of information related to distortions is followed by one of the known binarization methods, e.g. classical Otsu thresholding. Nevertheless, in this paper some region based methods based on stack of regions [2] are assumed as the last step of processing. This method assumes the division of the image into regions with assumed constant threshold, however using the multi-layered approach all the regions are shifted relative to the other layers. Hence, for such stack of regions rapid changes of thresholds at the boundaries of regions are eliminated.

### 3 Experimental Verification

In the paper [1] some results obtained for the fixed threshold (0.5 of the brightness range), global Otsu thresholding and locally adaptive thresholding proposed by Bradley have been presented using some metrics typical for the evaluation of binarization algorithms [4] for DIBCO datasets (<https://vc.ee.duth.gr/dibco2019/#eval>). Regardless that the achieved results may slightly differ, depending on the number of randomly drawn pixels, the application of the GGD preprocessing improved the binarization results for Otsu and Bradley methods to the level comparable with the application of adaptive thresholding, whereas its application for a fixed threshold was unsatisfactory. The results shown in Tables 1 – 3 (lower DRD, higher values of the other metrics) confirm the usefulness of the GGD preprocessing for some other methods as well.

**Table 1.** Results of binarization metrics obtained for DIBCO datasets using adaptive Sauvola method without and with proposed GGD preprocessing.

| DIBCO dataset     | 2009   | 2010   | 2011   | 2012   | 2014   | 2016   | 2017   | All    |
|-------------------|--------|--------|--------|--------|--------|--------|--------|--------|
| Accuracy          | 0.955  | 0.972  | 0.952  | 0.970  | 0.979  | 0.957  | 0.950  | 0.961  |
| Accuracy (GGD)    | 0.964  | 0.973  | 0.961  | 0.974  | 0.980  | 0.967  | 0.960  | 0.967  |
| F-Measure         | 0.786  | 0.811  | 0.786  | 0.811  | 0.903  | 0.801  | 0.779  | 0.806  |
| F-Measure (GGD)   | 0.811  | 0.809  | 0.817  | 0.824  | 0.904  | 0.827  | 0.799  | 0.823  |
| Specificity       | 0.953  | 0.982  | 0.953  | 0.977  | 0.985  | 0.957  | 0.952  | 0.964  |
| Specificity (GGD) | 0.965  | 0.986  | 0.966  | 0.983  | 0.989  | 0.971  | 0.967  | 0.974  |
| PSNR              | 14.378 | 16.228 | 14.371 | 16.210 | 17.935 | 14.710 | 13.637 | 15.135 |
| PSNR (GGD)        | 15.290 | 16.223 | 15.370 | 16.504 | 18.099 | 15.503 | 14.423 | 15.740 |
| DRD               | 13.073 | 7.739  | 12.230 | 9.370  | 3.754  | 10.571 | 11.250 | 10.036 |
| DRD (GGD)         | 10.926 | 7.044  | 9.014  | 7.736  | 3.600  | 8.454  | 9.222  | 8.191  |

**Table 2.** Results of binarization metrics obtained for DIBCO datasets using 2-layers region based method proposed in [2] without and with proposed GGD preprocessing.

| DIBCO dataset     | 2009   | 2010   | 2011   | 2012   | 2014   | 2016   | 2017   | All    |
|-------------------|--------|--------|--------|--------|--------|--------|--------|--------|
| Accuracy          | 0.962  | 0.973  | 0.945  | 0.974  | 0.978  | 0.961  | 0.954  | 0.962  |
| Accuracy (GGD)    | 0.967  | 0.975  | 0.951  | 0.976  | 0.978  | 0.965  | 0.957  | 0.965  |
| F-Measure         | 0.802  | 0.827  | 0.750  | 0.832  | 0.886  | 0.784  | 0.788  | 0.804  |
| F-Measure (GGD)   | 0.815  | 0.835  | 0.761  | 0.839  | 0.887  | 0.797  | 0.794  | 0.813  |
| Specificity       | 0.965  | 0.982  | 0.949  | 0.980  | 0.989  | 0.964  | 0.958  | 0.967  |
| Specificity (GGD) | 0.972  | 0.984  | 0.958  | 0.982  | 0.991  | 0.970  | 0.963  | 0.972  |
| PSNR              | 14.747 | 16.583 | 13.541 | 16.590 | 17.400 | 14.519 | 13.722 | 15.065 |
| PSNR (GGD)        | 15.316 | 16.794 | 13.880 | 16.753 | 17.496 | 14.921 | 13.934 | 15.340 |
| DRD               | 15.084 | 7.534  | 18.682 | 8.616  | 4.544  | 13.860 | 11.869 | 11.857 |
| DRD (GGD)         | 13.994 | 6.801  | 16.744 | 7.789  | 4.397  | 12.941 | 11.209 | 10.916 |

**Table 3.** Results of binarization metrics obtained for DIBCO datasets using 16-layers region based method proposed in [2] without and with proposed GGD preprocessing.

| DIBCO dataset     | 2009   | 2010   | 2011   | 2012   | 2014   | 2016   | 2017   | All    |
|-------------------|--------|--------|--------|--------|--------|--------|--------|--------|
| Accuracy          | 0.964  | 0.974  | 0.948  | 0.975  | 0.979  | 0.964  | 0.957  | 0.964  |
| Accuracy (GGD)    | 0.968  | 0.976  | 0.954  | 0.977  | 0.979  | 0.967  | 0.958  | 0.967  |
| F-Measure         | 0.810  | 0.831  | 0.760  | 0.839  | 0.890  | 0.796  | 0.797  | 0.812  |
| F-Measure (GGD)   | 0.821  | 0.839  | 0.770  | 0.845  | 0.890  | 0.805  | 0.800  | 0.819  |
| Specificity       | 0.967  | 0.983  | 0.953  | 0.981  | 0.990  | 0.967  | 0.961  | 0.970  |
| Specificity (GGD) | 0.973  | 0.985  | 0.961  | 0.983  | 0.992  | 0.972  | 0.965  | 0.974  |
| PSNR              | 15.018 | 16.722 | 13.815 | 16.817 | 17.604 | 14.835 | 13.992 | 15.312 |
| PSNR (GGD)        | 15.527 | 16.918 | 14.106 | 16.970 | 17.655 | 15.162 | 14.123 | 15.537 |
| DRD               | 14.490 | 7.183  | 17.617 | 8.073  | 4.326  | 13.006 | 11.231 | 11.217 |
| DRD (GGD)         | 13.541 | 6.548  | 15.922 | 7.344  | 4.234  | 12.310 | 10.746 | 10.431 |

## 4 Summary and Future Work

Obtained results confirm the usefulness of the proposed GGD based preprocessing for region based thresholding. Due to the use of the Monte Carlo method, the proposed solution may be considered as relatively fast, especially in comparison to some algorithms based on deep CNNs [5], which may easily outperform the proposed solution for the price of time-consuming training process, which additionally requires many test images, which might be unavailable. A natural direction of our further research is the use of some other GGD parameters during preprocessing step and a combination with some other binarization methods.

## References

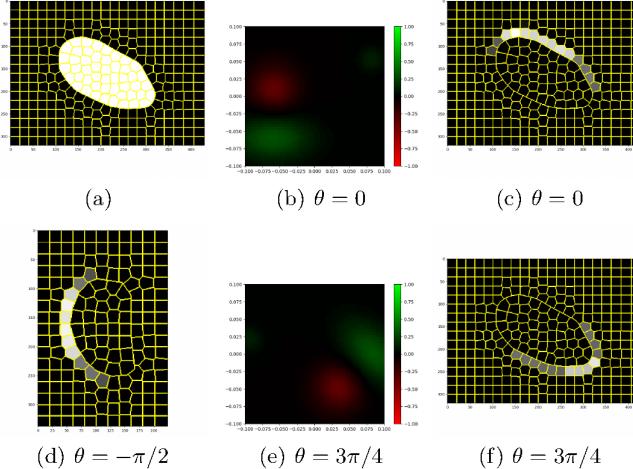
1. Krupiński, R., Lech, P., Teclaw, M., Okarma, K.: Binarization of degraded document images with Generalized Gaussian Distribution. In: Rodrigues, J.M.F. et al. (eds.) Computational Science – ICCS 2019. Lecture Notes in Computer Science, vol. 11540, pp. 177–190. Springer International Publishing (2019). [https://doi.org/10.1007/978-3-030-22750-0\\_14](https://doi.org/10.1007/978-3-030-22750-0_14)
2. Michalak, H., Okarma, K.: Adaptive image binarization based on multi-layered stack of regions. In: Vento, M., Percannella, G. (eds.) Computer Analysis of Images and Patterns. Lecture Notes in Computer Science, vol. 11679, pp. 281–293. Springer International Publishing (2019). [https://doi.org/10.1007/978-3-030-29891-3\\_25](https://doi.org/10.1007/978-3-030-29891-3_25)
3. Mitianoudis, N., Papamarkos, N.: Document image binarization using local features and Gaussian mixture modeling. Image and Vision Computing **38**, 33–51 (2015). <https://doi.org/10.1016/j.imavis.2015.04.003>
4. Ntirogiannis, K., Gatos, B., Pratikakis, I.: Performance evaluation methodology for historical document image binarization. IEEE Transactions on Image Processing **22**(2), 595–609 (2013). <https://doi.org/10.1109/TIP.2012.2219550>
5. Tensmeyer, C., Martinez, T.: Document image binarization with fully convolutional neural networks. In: 14th IAPR International Conference on Document Analysis and Recognition, ICDAR 2017, Kyoto, Japan, November 9–15, 2017. pp. 99–104 (2017). <https://doi.org/10.1109/ICDAR.2017.25>

# Graph CNN with Filter Transformations for Structure Detection and Identification

Arkadiusz Tomczyk<sup>[0000–0001–9840–6209]</sup>

Institute of Information Technology, Lodz University of Technology  
ul. Wolczanska 215, 90-924 Lodz, Poland, [arkadiusz.tomczyk@p.lodz.pl](mailto:arkadiusz.tomczyk@p.lodz.pl)

Geometric deep learning tries to generalize the concept of convolutional neural networks (CNN) for structures less regular than images e.g. graphs. In this work<sup>1</sup> an existing approach, defining convolutional filters using Gaussian mixture model (GMM), is examined. The main contribution of this research lies in applying that kind of network for structure detection in the images and in introducing rotations of trained filters. Rotation of filters simplify training process since only basic analyzed structures need to be presented. To initially verify<sup>2</sup> the proposed concept, it was applied for images which content is represented with a graph of superpixels. The network was trained to identify top, horizontal edges<sup>3</sup>.



**Fig. 1.** Structure detection ( $\theta$  denotes filter rotation angle): (a) - sample input, train image, (b), (e) - one of the GMM filters before and after rotation, (c), (f) - detection result before and after filter rotation, (d) - result of detection for sample test image.

<sup>1</sup> This project has been partly funded with support from National Science Centre, Republic of Poland, decision number DEC-2012/05/D/ST6/03091.

<sup>2</sup> Here only results of work in progress are presented.

<sup>3</sup> In the experiments the PyTorch Geometric library was used.

# Handcrafted features for CNN – is it worth it?\*

Maciej Stefańczyk<sup>1[0000-0001-9948-6319]</sup>, Dawid Seredyński<sup>1[0000-0003-2528-6335]</sup>, and Maciej Węgierek<sup>1[0000-0003-0779-255X]</sup>

Warsaw University of Technology, Institute of Control and Computation Engineering  
[{maciej.stefanczyk,dawid.seredyński,maciej.węgierek}@pw.edu.pl](mailto:{maciej.stefanczyk,dawid.seredyński,maciej.węgierek}@pw.edu.pl)  
<https://www.robota.ia.pw.edu.pl>

**Abstract.** Nowadays, when one needs a system for image recognition, it is mostly a matter of finding pre-trained CNN and, sometimes, adding additional training based on transferred knowledge. This approach, although successful, has some drawbacks. Using such networks in commercial applications may face some law issues, caused by licensing of the network weights or the dataset used for training. There are also some cases, where the problem to be solved is unique or niche, so that no appropriate network or datasets even exist yet. In any case, there is a need for training new network, and it is often based only on (relatively) small dataset. In this paper we try to check, whether using hand crafted features in that kind of scenario makes the learning process and results better than putting unprocessed images as an input.

**Keywords:** Convolutional neural networks · deep learning · computer vision.

## 1 Introduction

Convolutional neural networks have proven their efficiency in image classification tasks multiple times, recognizing common objects [8], faces [7], gestures [1] and more. Those networks were trained on datasets containing hundreds of thousands or even millions of unique images [2], which makes them quiet good at data generalization. Problem appears when (for any reason) one can't neither use pre-trained model nor publicly available dataset to train his own. Situation is even worse, if one has only limited dataset to use, with (relatively) small number of samples. Data augmentation in such case can be insufficient to produce good training results. There is a need then to aid the training and recognition system in some way.

## 2 Idea

A common interpretation of convolutional neural network structure is that first layers extract low level image features, while overall geometric relationships are

---

\* M. Stefańczyk is supported by the National Science Centre, Preludium grant no. UMO-2017/25/N/ST6/02358.

recognized near the end. To train the whole network there is a necessity of having a lot of training samples. In case of low number of training images, we had an idea to prepare low level features by hand, and then pass them as an input to CNN.

The idea of passing handcrafted features as input for CNN is not new. In [5] authors added LBP features as additional input alongside face image to make their system robust to presentation attacks. In [4] different approach is researched – CNN is used as a feature extractor and those features are compared with more traditional (handcrafted) ones. We decided to use gradient (edge) images, to reduce actual dimensionality of input data, but still preserving the spatial relationships between image parts for network to discover. Similar approach is presented in [3], but no comparison is presented to show the difference resulting from using edge images.

### 3 Experiments

Our goal was quiet simple – classify static hand gesture into one of six possible classes (Fig. 1) used for human-machine interaction task. Our training dataset was created by recording three different people, one recording per gesture. In total we have 9000 images, but as they were taken with 20 Hz rate, consecutive images are very similar with each other. In the target system there is a dedicated hand tracking part. The training (and testing) images are thus already cropped, such that hand is more or less in the center of the picture, we have also rough masks for the hand regions.

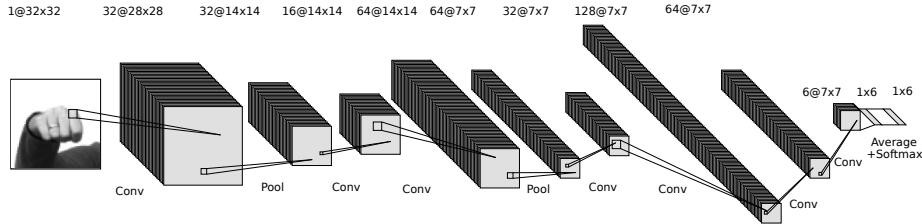


**Fig. 1.** Sample static gestures from training dataset

Cropped hand regions are used as an input to the CNN. The network contains 7 convolutional layers. Instead of the final fully connected layer another convolution is used (with six filters), where each filter corresponds to one gesture. The final decision is made based on the average value for each plane, with softmax layer as the output (Fig. 2).

The same network was trained in two different scenarios. First one is end-to-end training, where the original image is used as an input (let's call it classic way). The other scenario uses preprocessed, gradient images as an input. We trained both networks until they achieved around 95% accuracy.

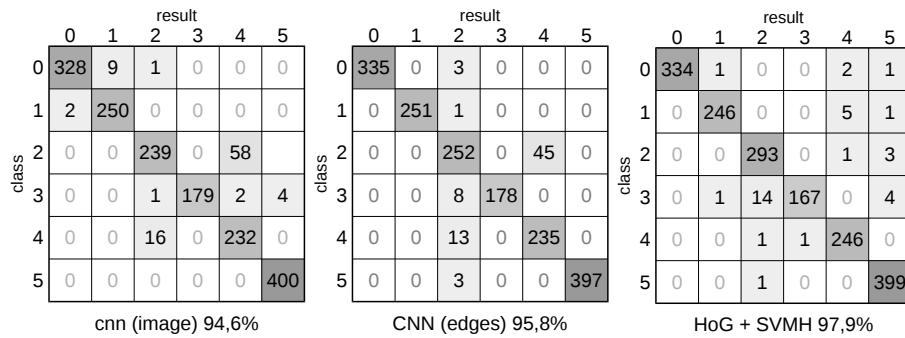
For the comparison one more classifier was prepared, based on HoG features with SVM, which is widely used in similar tasks [6]. It was effectively a 15 binary one-vs-one classifiers (each trained for every pair of gestures) with final voting.



**Fig. 2.** The structure of the neural network classifying static gestures

#### 4 Results and comments

Results achieved by all three classifiers are presented as confusion matrices on Fig. 3. Just by looking at those, all solutions seem to be similar with each other in terms of classification accuracy.



**Fig. 3.** Confusion matrices for static gesture recognition

Difference between each approach is better visible if we take into account number of training samples, that have to be prepared for them. Classical CNN (with unprocessed images as input) required around 100 thousand images (from the augmentation procedure). Using preprocessed images allowed us to cut this number down to 50 thousand. This resulted in shorter training time (half) and gave us slightly better testing accuracy (95,8% vs 94,6%). Error distribution for this classifier is a little bit different than the first one (more spread out), but overall accuracy is still better.

At this point, the answer for the title question is yes – handcrafting features for CNN makes sense and gives better results with shorter training time. There is a catch however. SVM classifier achieved even better accuracy (2,1 p.p. over the second best solution). In terms of training data difference is even bigger. Whole classifier used only 9 thousand original images to train, without any augmentation. It also trained much faster than both CNN classifiers. Does it mean, that

our previous argument is invalid? We don't think so. Our application is rather simple and has very limited training dataset. In that kind of scenarios handcrafting features may result in better accuracy and faster training when using CNN. The question now is different – do we need CNN for every visual classification task? The answer is, of course, no. For well defined applications classical, well established computer vision algorithms may give even better results in shorter time.

## 5 Future works

To better answer original question and additional question from the last paragraph we need to conduct more experiments. Our tests assumed the same CNN structure for both scenarios. Next step is to check, whether handcrafted features may allow for simplifying network structures by removing initial, low level layers without loss in accuracy. It is also an open question, where to put a border between "small" and "big" problems to decide whether to use CNN at all. Last, but not least, is checking whether similar techniques may be used not only in classification, but also in detection. Summing up – the approach seems to be correct and is a good candidate for further research.

## References

1. Koller, O., Ney, H., Bowden, R.: Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3793–3802 (2016)
2. Le, Q.V.: Building high-level features using large scale unsupervised learning. In: 2013 IEEE international conference on acoustics, speech and signal processing. pp. 8595–8598. IEEE (2013)
3. Molchanov, P., Gupta, S., Kim, K., Kautz, J.: Hand gesture recognition with 3d convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 1–7 (2015)
4. Nanni, L., Ghidoni, S., Brahma, S.: Handcrafted vs. non-handcrafted features for computer vision classification. *Pattern Recognition* **71**, 158–172 (2017)
5. Nguyen, D.T., Pham, T.D., Baek, N.R., Park, K.R.: Combining deep and hand-crafted image features for presentation attack detection in face recognition systems using visible-light camera sensors. *Sensors* **18**(3), 699 (2018)
6. Pang, Y., Yuan, Y., Li, X., Pan, J.: Efficient hog human detection. *Signal Processing* **91**(4), 773–781 (2011)
7. Parkhi, O.M., Vedaldi, A., Zisserman, A., et al.: Deep face recognition. In: bmvc. vol. 1, p. 6 (2015)
8. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)

# Towards Color Visual Cryptography with Completely Random Shares

Leszek J. Chmielewski , Grzegorz Gawdzik, and Arkadiusz Orłowski 

Institute of Information Technology, Warsaw University of Life Sciences – SGGW,  
Nowoursynowska 159, 02-775 Warsaw, Poland  
[{leszek\\_chmielewski,arkadiusz.orlowski}@sggw.pl](mailto:{leszek_chmielewski,arkadiusz.orlowski}@sggw.pl)

**Abstract.** The concept of black and white visual cryptography with two completely random shares, recently applied to color images, was improved by mixing the contents of the segments of each coding image, or share, which correspond to one pixel in the secret coded image. The mixing was done in two ways: by permuting the 2 by 2 tiles with which the shares were coded, and by permuting the single pixels within the segments. The results were compared with those of methods without mixing. The immediate improvement was that the column-wise organization of red, green and blue pixels in the shares was removed. This improved the randomness of the shares, without a significant deterioration of image quality. The method of permuting single pixels was found to be superior to the other method considered.

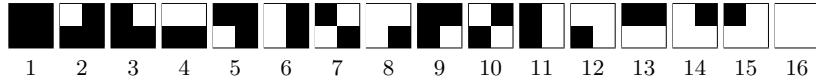
**Keywords:** Visual cryptography · color · true randomness · pixel mixing

## 1 Introduction

Among the visual cryptography methods the algorithm proposed in the '90 of the XX century [6,7] became the basis for numerous other algorithms and as such can be considered classic. It has been extended to gray-level as well as color images (the literature can be found in [1,5]). The domain is still in the development phase. From the most recent literature let us indicate [2] where the images of very good quality are obtained with the CMY color model. In [3] the RGB model is used with the pixel expansion  $5 \times 5$ , and two or three shares are used, with two shares being enough to show the coded secret image.

In the majority of methods, the shares in which the secret image is coded are not perfectly random, but only have no correlation with the secret. The encryption with totally random shares in black-and-white images was proposed and investigated in [9,10]. It was later applied to color images in [8], where the coding of pixels was totally random, but the layout of colors was organized into columns of red, green and blue pixels. The representation of the image with the RGB model was similar to that used in [11], where the pixel expansion was reported to be just three, while in [8,9,10] and in the present paper it is 36.

In this paper we present the visual coding of color images where not only the contents of the pixels, but also their layout is random. The improved randomness



**Fig. 1.** All possible  $2 \times 2$  tiles and their indexes.

of shares can improve the security and confidentiality of the information transfer process. This is done with two methods and the results are compared.

## 2 The method

We shall make a short notice on the basic methods and notions used in visual cryptography and on the transformations of the image which form a foundation for the methods to be compared here. The details can be found in our previous papers [8,9,10], with [8] being the most comprehensive description.

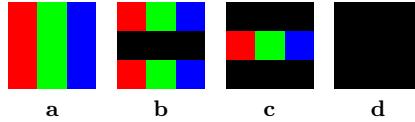
The image to be encoded is called the *secret*. For now let us assume the image is two-level. The secret is encoded into two images called the *shares*. None of them contains any information on the secret. The shares are printed on a transparent medium. The *decoding* consists in precisely overlaying the shares on each other, so the secret becomes visible to a naked human eye.

**Coding a two-level image** A single pixel in the secret corresponds to a square of  $n \times n$  pixels, called the *tile*, in each share; here  $n = 2$ . Hence, 16 different tiles are possible (Fig. 1).

In the classic coding [6,7], in one share, called the *basic share*, the tile corresponding to each pixel of the secret is represented with one of the tiles 4, 6, 7, 10, 11, 13, at random. Now, if the pixel in the secret is black, then in the other share, called the *coding share*, the negated tile is set (e.g., for tile 4 – tile 13). So, when the shares are superimposed, the whole tile corresponding to this pixel is black. If the pixel in the secret is white, then in the coding share the same tile as in the basic share is set. Then, after superposition, the whole tile is half-white.

In the random coding proposed in [9,10] the basic share is drawn at random from among all the 16 possible tiles. This causes differences with respect to the classic decoding, that is, the white pixels can become not only 1/2 white, but also 1/4 or 3/4 white (there are no errors in black pixels). The presence of such errors, analyzed in [10], is the cost of having totally random shares.

**Transforming a color image into a two-level image** To code a gray-level image it can be dithered into the black-and-white one. Similarly, the color image can be dithered into the one containing only black, red, green and blue pixels. Assume that an image is represented with a set of color columns, circularly R, G and B. Let us now represent each pixel in the secret with a  $3 \times 3$  segment. In each color column, the pixels can be either color, or black. Four of such possible segments are shown in Fig. 2. It is convenient to have square segments, so the number of rows is three, as is the number of columns. Such a segment can



**Fig. 2.** Variants of a  $3 \times 3$  pixel segment which encodes a single color pixel (in this case, white, gray, or black). (a) Full brightness – white pixel; (b) brightness  $2/3$  – bright gray; (c) brightness  $1/3$  – dark gray; (d) brightness  $0$  – black pixel.

represent four brightnesses of each color:  $0$ ,  $1/3$ ,  $2/3$  and  $3/3$  of the maximum brightness, by setting  $0$ ,  $1$ ,  $2$  or  $3$  pixels to color in a column.

Three colors, each with four levels of brightness, constitute a 64-color palette, in which the secret can be dithered. Thus, each pixel of the secret is represented as a respective  $3 \times 3$  pixel segment. Such an image can be coded, with either the classic or random coding, as described above. Each pixel of the secret is first replaced with a  $3 \times 3$  segment, and then, each pixel of this segment is replaced by a  $2 \times 2$  tile. Hence, the pixel expansion is  $3^2 \times 2^2 = 36$ .

Now, the R, G and B pixels in the shares and in the decoded image are organized in columns. The values of the pixels in the shares in the random version of coding are totally random; however, the ordered setup of colors reminds that of an old TV screen. This is the state described in [8].

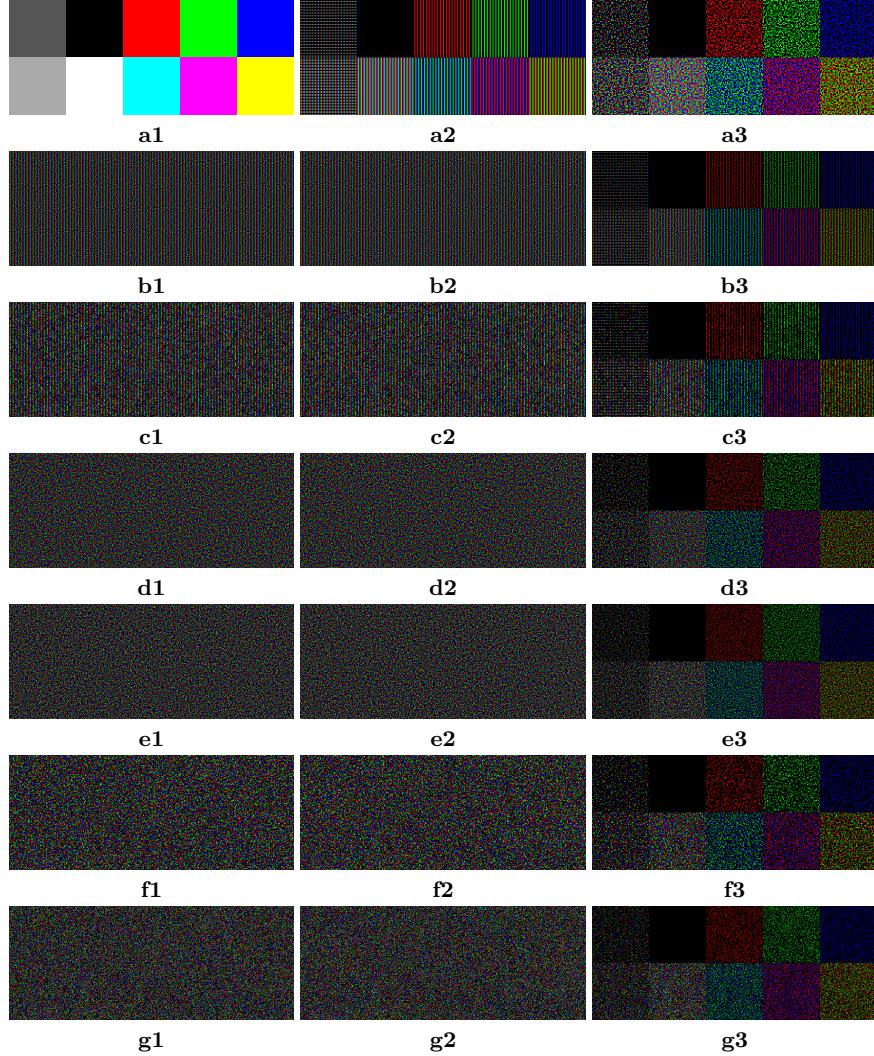
It should be noted that the process of coding with tiles of Fig. 1 implies that in each share, there is one black pixel per one color pixel, so that the number of black pixels equals the sum of numbers of color pixels. At present this makes it impossible for the shares to pass typical randomness tests designed for binary data.

**Mixing the tiles or mixing the pixels** To improve the randomness of the shares, the pixels will be locally randomly mixed, separately within each region of the shares corresponding to one pixel of the secret. This can be done in two ways. First, the pixels can be mixed within the  $3 \times 3$  segment, so that the column-wise organization of colors is removed. Such pixels are further coded randomly. In the shares, the structure of the  $2 \times 2$  tiles remains, although they are not organized into color columns. In the second method of mixing, all the pixels of the shares are mixed, within the regions corresponding to the pixels of the secret. The structure of the tiles is removed. The mixing can be done in the randomly coded as well as classically coded images.

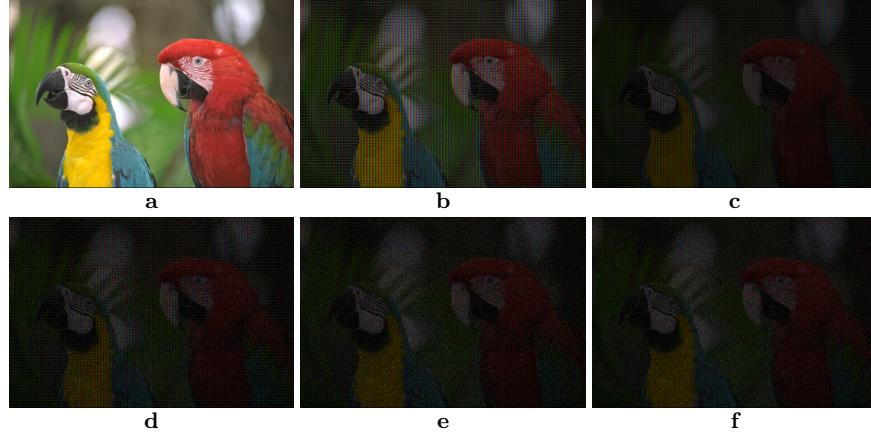
### 3 Examples

**Test image** A simple test image and its coding with the methods described above is shown in Fig. 3. The image contains squares in basic and complementary colors, and four shades of gray. All these colors can be accurately represented in the palette used. The pixels are replaced by  $3 \times 3$  segments with only the necessary pixels set to *on*, which forms the decomposed image in Fig. 3a2.

Some of the stages of the coding and decoding process make the quality of the images go down. The first reason for the loss of quality is the dithering into



**Fig. 3.** Illustration of coding and decoding. (a1) Secret ( $100 \times 40$ ), the dithered image is the same; (a2) decomposed into color stripes for coding ( $300 \times 120$ ); (a3) image a2 mixed with method 1 within segments ( $300 \times 120$ ). Further, (n1, n2) are shares, and (n3) is a decoded image, all ( $600 \times 240$ ). (b) Image a2 coded classically; (c) image a2 coded randomly; (d) image a3 coded classically, mixed with method 1; (e) image a2 coded classically, mixed with method 2; (f) image a3 coded randomly, mixed with method 1; (g) image a2 coded randomly, mixed with method 2.



**Fig. 4.** Coding a natural image. (a) Secret ( $384 \times 256$ ); (b) dithered and decomposed into color stripes ( $1152 \times 768$ ). All further images are ( $2304 \times 1536$ ). (c) Coded classically; (d) coded randomly, no mixing; (e) coded randomly, mixed with method 1; (f) coded randomly, mixed with method 2. Better to be seen in magnification.

the 64-color palette (not present for the test image). The second reason is the decomposition of the dithered image into the two-level image. In the white pixel of the secret all the pixels of the corresponding segment are on, while in the red pixel (green, blue, respectively) only the pixels in the red (green, blue) column are on. Inevitably, the image represented in this way is darker than the original. The coding by using the decomposition into tiles introduces further loss of brightness and contrast, as it is typically in the majority of visual cryptographic methods, where no calculations are admissible in the decoding process. The random coding additionally introduces the errors, mentioned above.

The two methods of mixing tend to smooth the image but introduce granularity, which is smaller in the case of method 2 with respect to that of method 1.

**Natural image** The image “parrots” [4] was chosen due to its bright and varied colors. Its size was reduced by resampling, to show better the coded images which have an increased resolution, Fig. 4. In this example, not only the color evenness, but also the preservation of details can be assessed. Despite the losses of quality, the objects can be recognized and the colors can be noticed.

#### 4 Conclusions and Perspective

The concept of visual coding with shares in which colors are coded in the completely random way was modified by locally mixing the contents of the images. This was done in two ways: by mixing the tiles before the coding is performed, and by mixing the pixels after the coding is done. The results with mixing were compared to those without mixing.

The coding introduces the loss of quality of the images, while the mixing seems to improve it slightly. The quality deterioration is due mainly to the two factors. The first one is the dithering and decomposing the image into a two-level color image, which has a limited color palette and contains only R, G, B and black pixels. The second one is the coding with the tiles, which decreases the image brightness by a half and introduces some granularity. Additional granularity and errors of brightness are introduced by the use of the completely random coding. The mixing of the pixels tends rather to improve than to decrease the image quality. The method in which the mixed elements are single pixels has a more positive effect than that in which the tiles are mixed.

In the future it is planned to work towards balancing the number of black and color pixels in the shares and to check the degree to which they are actually random with the appropriate statistical tests.

## References

1. Cimato, S., Yang, C.N.: Visual Cryptography and Secret Image Sharing (Digital Imaging and Computer Vision). CRC Press, Inc., Boca Raton, FL, USA (2011), <https://www.crcpress.com/Visual-Cryptography-and-Secret-Image-Sharing/Cimato-Yang/9781439837214>
2. Dahat, A.V., Chavan, P.V.: Secret sharing based visual cryptography scheme using CMY color space. Procedia Computer Science **78**, 563–570 (2016). <https://doi.org/10.1016/j.procs.2016.02.103>, 1st International Conference on Information Security & Privacy 2015
3. Dhiman, K., Kasana, S.S.: Extended visual cryptography techniques for true color images. Computers & Electrical Engineering **70**, 647–658 (2018). <https://doi.org/10.1016/j.compeleceng.2017.09.017>
4. Franzen, R.: RWF's Eclectic Miscellany (2015), <http://r0k.us>, [Accessed Apr 2019]
5. Liu, F., Yan, W.Q.: Visual Cryptography for Image Processing and Security: Theory, Methods, and Applications. Springer International Publishing, Cham (2014). <https://doi.org/10.1007/978-3-319-09644-5>
6. Naor, M., Shamir, A.: Visual cryptography. In: De Santis, A. (ed.) Proc. Advances in Cryptology — EUROCRYPT'94. pp. 1–12. Springer, Berlin, Heidelberg (1995). <https://doi.org/10.1007/BFb0053419>
7. Naor, M., Shamir, A.: Visual cryptography II: Improving the contrast via the cover base. In: Lomas, M. (ed.) Proc. Security Protocols. pp. 197–202. Springer, Berlin, Heidelberg (1997). [https://doi.org/10.1007/3-540-62494-5\\_18](https://doi.org/10.1007/3-540-62494-5_18)
8. Orlowski, A., Chmielewski, L.J.: Color visual cryptography with completely randomly coded colors. In: Vento, M., Percannella, G. (eds.) Proc. Int. Conf. on Computer Analysis of Images and Patterns CAIP 2019. Lecture Notes in Computer Science, vol. 11678, pp. 589–599. Springer Nature Switzerland AG, Salerno, Italy (2–6 Sep 2019). [https://doi.org/10.1007/978-3-030-29888-3\\_48](https://doi.org/10.1007/978-3-030-29888-3_48)
9. Orlowski, A., Chmielewski, L.J.: Generalized visual cryptography scheme with completely random shares. In: Petkov, N., Strisciuglio, N., Travieso, C.M. (eds.) Proc. 2nd Int. Conf. Applications of Intelligent Systems APPIS 2019. ACM International Conference Proceeding Series, vol. 1869, pp. 33:1–33:6. ACM, Las Palmas de Gran Canaria, Spain (7–9 Jan 2019). <https://doi.org/10.1145/3309772.3309805>
10. Orlowski, A., Chmielewski, L.J.: Randomness of shares versus quality of secret reconstruction in black-and-white visual cryptography. In: Rutkowski, L., et al. (eds.) Proc. Int. Conf. on Artificial Intelligence and Soft Computing ICAISC 2019. Lecture Notes in Artificial Intelligence, vol. 11509, pp. 58–69. Springer, Cham, Zakopane, Poland (16–20 Jun 2019). [https://doi.org/10.1007/978-3-030-20915-6\\_6](https://doi.org/10.1007/978-3-030-20915-6_6)
11. Yang, C.N., Chen, T.S.: Colored visual cryptography scheme based on additive color mixing. Pattern Recognition **41**(10), 3114–3129 (2008). <https://doi.org/10.1016/j.patcog.2008.03.031>

# Neural Network-based Compressed Image Improvement

Patryk Najgebauer<sup>1[0000–0002–7168–3019]</sup>, Rafał Scherer<sup>1[0000–0001–9592–262X]</sup>

Częstochowa University of Technology  
Al. Armii Krajowej 36, 42-200 Częstochowa, Poland  
 [{patryk.najgebauer, rafal.scherer}@iisi.pcz.pl](mailto:{patryk.najgebauer, rafal.scherer}@iisi.pcz.pl), <http://iisi.pcz.pl>

**Abstract.** We introduce a full convolutional neural network that improves multimedia files compressed by methods based on discrete cosine transform (DCT). In the case of high compression level, such images and video exhibit unwanted artefacts such as halo, ringing or blocks. Here we apply a neural network to reconstruct heavily compressed image by working on square blocks that come from the compression algorithm. We trained our model on clipart sketch images transformed to the JPEG and PNG formats, as the input and output data, respectively. We present examples of reconstruction of real photographs compressed with the JPEG algorithm.

## 1 Introduction

Images and video are often compressed with lossy data compression. In contrast to textual data, multimedia does not lose readability for people in the case of a loss of some data. The most popular methods of lossy compression are based on the discrete cosine transform [1], with the most popular JPEG and MPEG formats. The resulting image or video frames lose some of its original quality what is manifested by artefacts visible as square blocks appearing near edges.

Lossy image compression significantly reduces the size of the stored image, but it leads to a loss of original image reproduction capabilities. In the case of images stored in the JPEG format that uses the discrete cosine transform (DCT), the compression ratio is regulated by the level of compression. With the quality decreasing, a large compression ratio and repeated recompression cause a noticeable problem of block artefacts. This effect is particularly visible on the edges of contrasting uniform areas as well as on gradient-filled surfaces. The easiest way to weaken their visibility is to use blur techniques; however, they affect the whole picture sharpness. Block artefact effect becomes a problem in the case of enlarging the image already saved in the JPEG format; the artefacts become much more noticeable, and in the absence of a source image, it is impossible to display a well-looking magnified slice of the image. One of the easiest ways to remove defects is to use image blur techniques that will reduce the visibility of block artefacts. Blurring lowers the readability of the image – sharp edges and details are no longer visible. Much better results are obtained by machine learning methods that predict pixel values of the image based on previously learned ground truth data.

The effect of block artefacts is created by decompressing the image based on the DCT transform. The compressed image is reproduced in blocks of  $8 \times 8$  each. Each reproduced block of the image is the sum of 64 patterns. In the case of 100 % compression quality, the image is rendered lossless because each pattern coefficients is saved with full accuracy. In the case of lowering the compression quality, the accuracy of saved coefficients is also reduced, resulting in a much smaller output file size. However, the lower accuracy of the coefficients creates a more significant discrepancy between the original and the reproduced image. A particularly visible effect is when the block fragment is on a contrasting part of the image. Then, the distances between the values of coefficients increase and have a direct impact on the increase of errors in the reproduction of the image.

## 2 Fully Convolutional Network Model

The purpose of our experiments was to develop a method that removes artefacts created during JPEG compression by a neural network trained on synthetic drawings. To this end, we developed our own fully convolutional model and



**Fig. 1.** An example of compression artefact removing from a natural image in the JPEG format with the quality  $q = 60$ .

trained it from scratch on a set of vector images converted to JPEG with visible compression artefacts. Our FCNN model uses three blocks of pooling and

convolution layer similar to the VGG16 model [2] that transform the input to a multidimensional feature map representation and then uses three blocks of upsampling and convolution layers (they work as upconvolution) to reconstruct image without artefacts. The purpose of the described model is the aggrega-



**Fig. 2.** Examples of training images used to generate compressed input files.

tion of the closest neighbourhood of the filtered pixel. Each convolutional layer uses standard  $3 \times 3$  filters. Each layer of the first block uses 64 filters, and each next convolutional block use twice as many as the previous block, while on the deconvolution blocks the number of filters is reversed. In order to better reproduce the input image, which can become blurred during the pooling and then upsampling, the output from the first block of convolution is passed to the last deconvolution block. In addition, we added a DCT block that unlike other layers, has the weights fixed and they are initialised by the values of the discrete cosine transform. The DCT layer has 64 filters of  $8 \times 8$  size and sampling each separated channel with 8 pixels offset in accordance with JPEG compression. The shape of the output of this layer at the start was reduced eight times without using pooling and is passed to the input of the first upconvolution block. The DCT block is designed to additionally sensitise the network to artefacts coming from the DCT block but also considering the neighbouring blocks. Usage of the DCT layer changes the way of processing the input image. The image cannot be scaled and modified in any other way. Any modification of the image will make the image compression blocks not overlap with the DCT sampling layer. The input image size is  $256 \times 256$ , so smaller images must be padded, and larger images divided into fragments processed separately with offsets calculated every 8 pixels according to the compression blocks.

### 3 Experiments

We implemented the FCN network in Python with Keras library using TensorFlow 1.4 endpoint on a single Nvidia GTX 1080 GPU. We used the CorelDRAW clipart eleven thousand vector images divided into 81 categories as the training set. We converted the images to the SVG format for easier data preparation and augmentation purposes. The advantage of vector images is their mathematical description allowing any manipulation of the image size without losing quality. In the case of vector images, it is also possible to easily manipulate the image context from the script language level, such as colour change, nodes manipulation, or line thickness change. We converted each image to the PNG and JPEG

format. For the learning process purposes, the JPEG images represent the input set and the PNG ground truth set. During the JPEG image conversion, we used the following compression levels: 80%, 60%, and 40%. Example images are presented in Figure 2. We augmented the data by rotation, scaling and cropping of the image in the vector form and then saving it to the PNG raster form. In this way, the magnified image did not lose its sharpness, especially on the edges of the curves. Images were also cropped so that the content randomly filled the entire space of the image in order to train the model evenly. Augmentation of the data set was done randomly for each batch of data during learning, resulting in very rare repetitions of training examples. We trained the model with the Adam optimisation algorithm from scratch, and initialised all the layers randomly except the DCT layer. The learning process was carried out in steps of 30 epochs and every step was ended by saving the model. For each epoch, we dynamically generated a batch set of 10 elements. We trained the FCNN through 9000 learning epochs. Image reconstruction accuracy was expressed in the mean absolute error (MAE), that is the average difference between the ground truth and the reconstructed pixels value. We also tested the presented model on improving natural photographs for which the content is significantly different from the training set. Although we trained the model on the collection of synthetic images without adding additional noise, it can properly improve real-world photographs (Figure 1).

## 4 Conclusion

In the paper, we developed a fully convolutional network to repair automatically image compression-related artefacts. The experiments showed that the FCN model is an effective tool for graphics processing. Compared to neural models that contain fully connected layers, the presented FCN learned perfectly the mutual spatial distribution of the pattern. This is especially important in the case of removing compression artefacts where we need to recreate the input image on the output with minor pixels corrections based on the pixel neighbourhood. We also obtained interesting effects by using the model trained only on artificial images to improve real-world photography. With such significant differences between the context of these types of images, the method achieved excellent results with strong edge protection, when compared to similar methods from the literature.

## References

- [1] Ahmed, N., Natarajan, T., Rao, K.R.: Discrete cosine transform. *IEEE Trans. Comput.* **23**(1) (January 1974) 90–93
- [2] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)

## RAS (Robotics and Autonomous Systems)

# Low Effort Cross-Modal Learning for 3-D LiDAR Data Segmentation in SLAM

Krzysztof Ćwian, Tomasz Nowak, Michał R. Nowicki, and Piotr Skrzypczyński

Institute of Control, Robotics and Information Engineering,  
Poznań University of Technology, ul. Piotrowo 3A, 60-965 Poznań, Poland  
`{name,surname}@put.poznan.pl`

**Abstract.** This short paper presents a new approach to real-time semantic segmentation of 3-D point clouds employing intensity data yielded by the new generation Ouster LiDAR and the concept of cross-modal transfer learning. This concept, combined with the new representation of the acquired data as 2-D multi-modal images, allows our system to learn from labeled grayscale images, and then transfer the network to the LiDAR intensity domain. Semantic labels of intensity data allow us to decide if the corresponding range data are useful for SLAM.

**Keywords:** Semantic segmentation · Deep learning · SLAM · LiDAR.

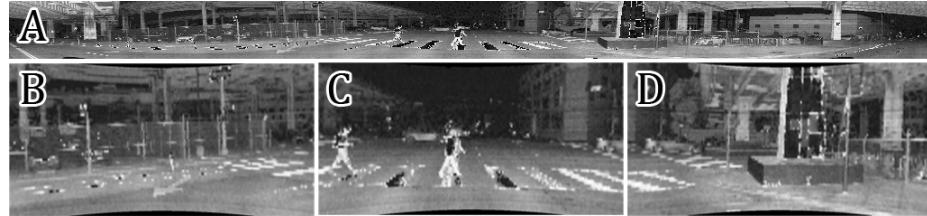
## 1 Introduction

The increasing interest in robust, real-time perception for autonomous vehicles resulted in several technological innovations in the area of LiDARs. An example is the Ouster OS-1 family of sensors that yield depth, signal-intensity and ambient light values, which are spatially and temporally aligned, without any shutter effects. New sensors need new algorithms to process their massive data in real-time. Deep learning architectures are considered state-of-the-art in image processing, but with the new LiDARs obtaining labeled data suitable for supervised learning becomes an issue. Deep neural architectures proposed so far for segmentation and/or object detection in point clouds (e.g. [2, 7]) are either suitable only for small point clouds or do not guarantee real-time processing. In image-based perception, the problem of insufficient training data can be alleviated by using networks pre-trained on large datasets [4]. This method is rarely used in LiDAR-based systems because LiDAR point clouds are sparse with variable point density, in contrast to the uniform matrix-like structure of images.

We show preliminary results of applying a cross-modal transfer learning scheme that exploits the multi-modal output of the new Ouster OS-1 and learns from standard grayscale images, which are commonly available with class and instance-level annotations. We demonstrate that a state-of-the-art deep neural architecture for semantic segmentation can work with the multi-modal Ouster data, inferring accurate semantic labels from the synthesized intensity images. This segmentation is then projected to the depth data domain and used to decide if particular 3-D points should be used or not by a SLAM algorithm.

## 2 Representation of LiDAR data

The Ouster OS-1 measures distances at selected horizontal and vertical angles, returning also the corresponding intensity signal. For this preliminary research, exemplary recordings made publicly available<sup>1</sup> by the sensor manufacturer were used. The sequences were taken with the 64-beam OS-1 rotating at 10 Hz, yielding  $n_{\text{hor}} = 1024$  horizontal measurements with horizontal field of view  $\text{fov}_{\text{hor}} = 360^\circ$ , and  $n_{\text{ver}} = 64$  vertical measurements with vertical field of view  $\text{fov}_{\text{ver}} = 32^\circ$ . A single scan can be represented as a  $1024 \times 64$  matrix/image, where each pixel is mapped according to the angle increment of its corresponding measurement, which results in a non-linear projection. Figure 1A shows an intensity image constructed this way.



**Fig. 1.** Comparison of a directly generated laser intensity image (A), and the camera-like images (B,C,D) for the same scan from one of the publicly available sequences

However, to facilitate learning of neural networks we need to represent OS-1 data as a regular image structure, according to the pin-hole camera model. To this end, we synthesize six  $1024 \times 512$  images with the vertical field of view of  $60^\circ$  and “virtual” camera parameters  $f_x = f_y = 800, c_x = 512, c_y = 256$  (focal length and center in pixels) from a single scan. Examples are shown in Fig. 1B,C,D. Each pixel value in the  $i$ -th image ( $u_{\text{dest}}^i, v_{\text{dest}}^i$ ) is computed as the value for subpixel position ( $u_{\text{laser}}, v_{\text{laser}}$ ) from the array of laser data:

$$\theta = \text{atan}2(u_{\text{dest}}^i - c_x, f_x) + 60^\circ * i, \quad \phi = \text{atan}2(c_y - v_{\text{dest}}^i * \cos(\theta), f_y), \quad (1)$$

$$u_{\text{laser}} = \theta \frac{n_{\text{hor}}}{\text{fov}_{\text{hor}}} + \frac{n_{\text{hor}} + 1}{2}, \quad v_{\text{laser}} = \phi \frac{n_{\text{ver}}}{\text{fov}_{\text{ver}}} + \frac{n_{\text{ver}} + 1}{2}. \quad (2)$$

The image value (intensity or depth) for the subpixel position is computed using a linear interpolation of the neighboring pixel values. Finally, the inpainting algorithm from OpenCV (based on the Fast Marching Method) is applied to each image to remove any “holes” resulting from invalid measurements.

## 3 Cross-modal learning procedure

Real-time semantic segmentation is accomplished using the recent, very efficient ERFNet [3] architecture on the synthesized intensity images. Our aim was to

---

<sup>1</sup> <http://data.ouster.io/sample-data-1.12/index.html>

distinguish between pixels that represent static objects and can be safely used for localization, pixels that represent non-rigid or movable objects, and pixels that represent dynamic objects or clutter, and should be rejected for localization.

| Our labels for SLAM        | CityScapes labels   |
|----------------------------|---|
| Class 1<br>Best for SLAM   | road, sidewalk, parking, building, wall, fence, guard rail, bridge, tunnel, pole, pole group, traffic sign, traffic light |
| Class 2<br>Possibly useful | car, truck, bus, on rails, motorcycle, bicycle, caravan, trailer  |
| Class 3<br>Not useful      | person, rider, sky  |

**Table 1.** Example mapping of class labels for the CityScapes dataset

According to our approach, the ERFNet was pre-trained on about 10 000 labeled images from the KITTI, CityScapes and BDD100K datasets (Fig. 2A). The images were converted to grayscale and rescaled, while the original class labels were merged into three classes, as shown in Tab. 1. In the second stage, the model was fine-tuned on 150 manually labeled images from the publicly available OS-1 sequences (Fig. 2B,C).

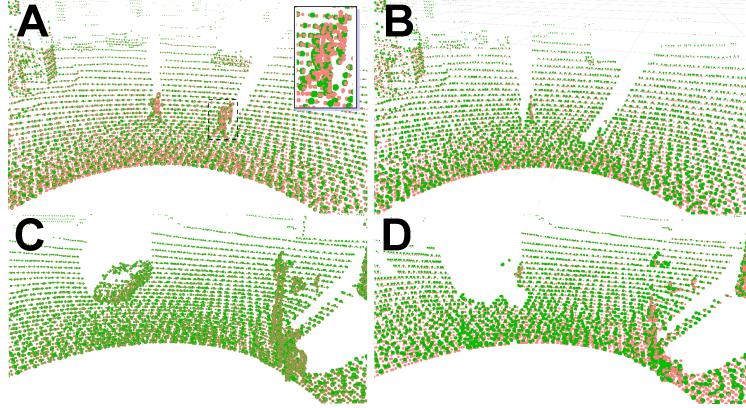


**Fig. 2.** Example from CityScapes used in pre-training (A), intensity image from OS-1 (B) with manual labeling (C): Class 1 is green, Class 2 is red, Class 3 without outline

## 4 Experimental results

The target application of our semantic segmentation method is selection of the LiDAR points that are useful for Simultaneous Localization and Mapping (SLAM). We use a modification of the LOAM algorithm [6] that combines scan-to-scan real-time odometry and registration of the LiDAR scans to an incrementally updated point-based map. While LOAM tries to choose stable points for matching using some heuristics, our results [1] show that this strategy is insufficient for autonomous vehicles operating in a cluttered urban environment.

Figure 3 demonstrates that semantic segmentation of the intensity images from Ouster OS-1 makes it possible to reject almost all points that do not belong to static and rigid objects, while the LOAM heuristics allowed the matching points to be defined e.g. on pedestrians (inset image in Fig. 3A). Unfortunately, having only the publicly available sequences without ground truth trajectories we cannot demonstrate an improvement to the localization accuracy, which makes our results preliminary.



**Fig. 3.** Point clouds before (A,C) and after (B,D) rejection of Class 2 and Class 3 objects. Current laser scan in orange, correspondences to the map points in green. Spurious correspondences are clearly visible in the enlarged fragment (inset image)

## 5 Conclusions

We demonstrate that it is possible to use a representative set of RGB/grayscale images to learn a deep neural network and then transfer the results to the laser intensity domain with a limited number of manually labeled examples. This approach is of practical value, as in spite of the low effort training, the ERFNet can segment the intensity images at the OS-1 frame rate. The preliminary experimental results show that this procedure makes it possible to reject unreliable range data in real-time using learned semantics rather than ad-hoc rules.

## References

1. Nowicki M., Nowak T., Skrzypczyński P.: Laser-based localization and terrain mapping for driver assistance in a city bus. *Automation 2019. Progress in Automation, Robotics and Measurement Techniques*, R. Szewczyk, et al., (eds.), AISC 920, Springer, pp. 502–512 (2019)
2. Qi C., Su H., Mo K., Guibas L. J.: Pointnet: Deep learning on point sets for 3D classification and segmentation. In: *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 652–660 (2017)
3. Romera E., Alvarez J. M., Bergasa L. M., Arroyo R.: ERFNet: Efficient residual factorized ConvNet for real-Ttme semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, **19**(1), pp. 263–272 (2018)
4. Wang M., Deng W.: Deep visual domain adaptation: A survey, *Neurocomputing*, **312**, 135–153 (2018)
5. Wu B., Zhou X., Zhao S., Yue X., Keutzer K.: SqueezesegV2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a LiDAR point cloud. *arXiv preprint arXiv:1809.08495* (2018)
6. Zhang J., Singh S.: Low-drift and real-time LiDAR odometry and mapping. *Autonomous Robots*, **41**(2), 401–416 (2017)
7. Zhou Y., Tuzel O.: Voxelnet: End-to-end learning for point cloud based 3d object detection, In: *IEEE Conference on Computer Vision and Pattern Recognition* (2018)

# Machine learning approach to constrained path planning for intelligent articulated buses<sup>\*</sup>

Piotr Kicki, Tomasz Gawron, and Maciej Marcin Michałek

Institute of Automation and Robotics, Poznan University of Technology (PUT),  
ul. Piotrowo 3A 60-965 Poznan, Poland [piotr.m.kicki@gmail.com](mailto:piotr.m.kicki@gmail.com),  
[maciej.michalek@put.poznan.pl](mailto:maciej.michalek@put.poznan.pl)

**Abstract.** This brief presents the concept of a neural path planner for constrained maneuvering with articulated buses. The planner is a part of a larger motion algorithmization system.

**Keywords:** machine learning · path planning · state constraints · intelligent vehicles · motion algorithmization

## 1 Introduction

Nowadays the efforts put on developing autonomous cars are increasing. Many companies are running their autonomous car programs, whereas only a few consider the autonomization of the vehicles with more complex kinematics, like articulated buses. Articulated buses are ubiquitous in the urban environment, but maneuvering with them is complicated since even humans need many hours of training to drive them properly. That justifies a need for the motion algorithmization for intelligent buses, which will support (Advanced Driver Assistance Systems - ADAS) or replace (autonomy) a driver.

There are several approaches to motion algorithmization of the intelligent vehicles: imitation learning [4], direct perception [1], modular pipeline [5]. In this paper, we consider only the last one - a modular pipeline comprising stages of sensing, planning, and acting. We focus on planning and acting, with the emphasis of the modular algorithmization of a single maneuver of the articulated bus. The main contribution of this paper is the use of neural networks for path planning for articulated buses, which takes into account state and input constraints, namely: static obstacles, joint and steering angle limitations.

## 2 Problem formulation

Motion algorithmization, in general, is broad, but in this paper, we consider only a small part of it, the algorithmization of single monotonic maneuvers, where the

---

\* This work was supported in part by the National Centre for Research and Development (NCBR), Poland, as a grant No. POIR.04.01.02-00-0081/17, and in part by the research subvention 09/93/SBAD/0911 of PUT.

maneuver is meant as reaching the desired terminal pose (position and orientation) with the bus. In the paper, three maneuver types are considered: bus-bay entering, angle parking, and perpendicular parking. The maneuver environment is assumed to be statically cluttered and a free space is represented as a sum of quadrangles. According to the modular pipeline approach, there are two successive phases of the maneuver: motion planning and motion execution i.e., planning and acting.

In the considered problem, motion planning is specified as path planning, because execution time is considered as negligible. The path is defined as a spline constructed from 5th-degree polynomials. The objective is to plan an admissible path (satisfying the configuration constraints) with continuous accelerations, which starts at the actual vehicle pose and ends at the desired configuration. Motion execution is specified as path following using the algorithm which drives the path following errors asymptotically to the 0, addressed here by using a cascade-like feedback control algorithm. Since buses maneuver with small velocities and small accelerations, one can neglect the kinetics and skid/slippage effects and focus solely on articulated bus kinematics presented in [2].

In all scenarios, the initial steering wheel angle is assumed to be equal to zero, as it can be stabilized at any point without moving the vehicle.

### 3 Methods

To solve the tasks specified above two algorithms were used: novel neural network based path planning algorithm and a cascade-like feedback control law.

#### 3.1 Cascade controller with VFO algorithm

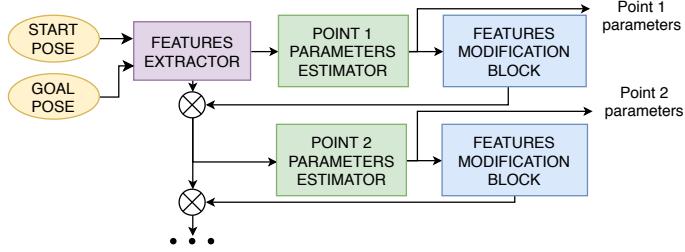
The cascade-like controller consists of an outer loop with the path-following VFO (Vector Field Orientation) control function devised for unicycle kinematics (see [3]), which controls the position and orientation of the vehicle and an inner steering-angle control loop, which controls the orientation of the steering tractor's wheel.

The inner loop task is to produce the steering wheel angular velocity which minimizes the error between desired steering angle computed by the kinematic algorithm and the actual steering angle.

The outer loop computes the linear velocity of the driving axle and the desired steering angle. In the presented solution calculations are ordered as follows: first, the VFO control function appoints desired linear and angular velocities of a tractor body, and next, these velocities are transformed along a vehicle chain into a linear velocity of a driving axle and the desired steering angle for a tractor.

#### 3.2 Neural network based path planning algorithm

Let us consider the path planning function, which for a given task specification, environment representation and a vehicle model returns the path. One can imagine a planning function which, for all prescribed tasks and motion environments



**Fig. 1.** General scheme of the proposed path planning neural network architecture.

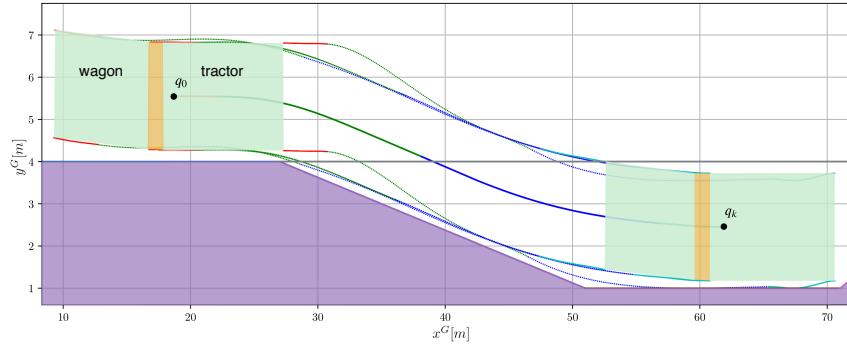
(for which an admissible path exists), finds such an admissible path. Planning function which satisfies the aforementioned definition can be called *oracle planning function*, as it always (if possible) returns the admissible path. Knowing such a function will solve all path planning problems immediately. Unfortunately, it is in general very hard to specify such a function even for relatively simple environments and tasks.

Since it is hard to construct such a function, maybe at least it is possible to approximate the oracle planning function in some narrow range of its parameters. As the oracle function is expected to be complicated and nonlinear one has to propose the model which is expressive and can handle inherent nonlinearities of the constrained path planning problem. One of the possible solutions is to use a neural network, which is indeed the core of the proposed method.

The assumed spline-based path representation affects the proposed neural network architecture which consists of an initial feature extractor(it produces the hidden representation of the problem from an initial state and a desired terminal state representation), and subsequent blocks responsible for gluing points parameters estimation and hidden representation modification. The scheme of the proposed neural network architecture is depicted in Figure 1. Each block consists of a single fully connected layer, except the parameters estimators where are 4 independent layers, one for each parameter. The size of the feature space was set to 128, whereas the number of segments (polynomial paths) was chosen arbitrarily for each task.

To guide the neural network to mimic the oracle planning function, the authors proposed a loss function, which penalizes inadmissible plans i.e. the plans which do not meet the conditions of the oracle planning function. The proposed loss function has two components: collision loss, to force the planner to produce collision-free paths, and constraint loss (steering angle and joint angle limitations), to ensure that produced paths are possible to follow by the state constrained bus.

The training procedure was performed independently for each maneuver type, thus the map perception mechanism could be omitted. For each scenario, there were at most few million starting configurations drawn from the training ranges (which means that only a few possible configurations were included in the training set because the number of possible examples was about  $10^{29}$ ).



**Fig. 2.** Example path generated by the proposed planner for the bus-bay entering task.

## 4 Results

The obtained success rates of the proposed planning algorithm are presented in Table 1. All received rates refer to the points drawn from the training data range but do not contain training examples (with probability less than  $10^{-20}$ ). In Figure 2 one can see the exemplary path in the bus-bay entering scenario. It is worth noting that the time needed to compute such a path is less than 1 ms and to evaluate its correctness is about 20 ms.

**Table 1.** Success rates of the plans computed by the proposed neural planner. Evaluation was performed on 10000 starting configurations located on the grid in training configurations ranges.

| Scenario         | Bus-bay entering | Angle parking | Perpendicular parking |
|------------------|------------------|---------------|-----------------------|
| Success rate [%] | 83.2             | 95.7          | 99.5                  |

## References

- Chen, C., Seff, A., Kornhauser, A., Xiao, J.: Deepdriving: Learning affordance for direct perception in autonomous driving. In: 2015 IEEE International Conference on Computer Vision (ICCV). pp. 2722–2730 (Dec 2015)
- Michalek, M.M.: Modular approach to compact low-speed kinematic modelling of multi-articulated urban buses for motion algorithmization purposes. In: 2019 IEEE Intelligent Vehicles Symposium (IV). pp. 1803 – 1808 (2019)
- Michalek, M.M., Gawron, T.: VFO path following control with guarantees of positionally constrained transients for unicycle-like robots with constrained control input. Journal of Intelligent & Robotic Systems **89**(1), 191–210 (2018)
- Pomerleau, D.A.: Advances in neural information processing systems. chap. ALVINN: An Autonomous Land Vehicle in a Neural Network, pp. 305–313. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1989)
- Thrun, S., et al.: Stanley: The robot that won the DARPA Grand Challenge. Journal of Field Robotics **23**, 661–692 (01 2006)

# You Only Look Once Around: Learnable Object Detection for Bioinspired Visual Localization

Marta Rostkowska

Institute of Control, Robotics and Information Engineering,  
Poznań University of Technology,  
ul. Piotrowo 3A, 60-965 Poznań, Poland  
[marta.a.rostkowska@doctorate.put.poznan.pl](mailto:marta.a.rostkowska@doctorate.put.poznan.pl)

**Abstract.** This paper presents efficient, real-time detection of landmark patterns in the environment using an omnidirectional vision system. The sensor is inspired by its biological counterparts: eyes of insects and peripheral vision mechanisms. A state-of-the-art convolutional neural network architecture is demonstrated to have adaptability to recognize specific patterns on distorted images from the catadioptric camera, and implemented on the embedded computer of our sensor. The solution is shown to outperform a classic, non-learnable approach with respect to both speed and accuracy.

**Keywords:** Object detection · Deep learning · Omnidirectional vision.

## 1 Introduction

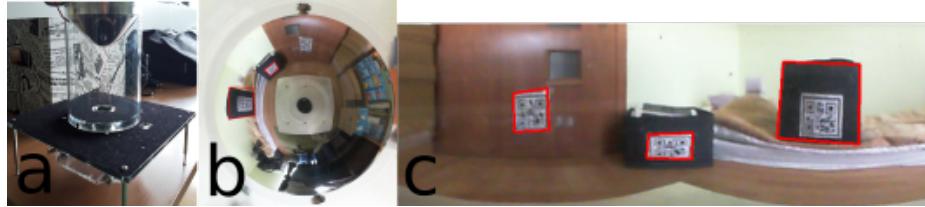
An important feature of autonomous robots is the ability to determine their own location in the environment. Many different approaches using various sensors were proposed to solve this problem, but passive cameras are arguably the most popular sensors for robot navigation. From the robot localization task in 2-D environment the omnidirectional cameras are particularly interesting, as they enable the whole local scene to be seen in one image. In [3] we have presented a hybrid vision sensor, which was inspired by the visual systems of insects and vertebrates. This sensor combines advantages of omnidirectional (catadioptric) and classic (perspective) cameras. Animals recognise objects using the foveal vision, and they notice objects and events by peripheral vision. The hybrid vision sensor is mimicking this concept, adding the omnidirectional field of view like in insects (Fig.1a). The robot can quickly detect relevant objects in an image from the omnidirectional camera, and move the perspective camera (mounted on a servo) to the area of interest in order to acquire a detailed image.

For efficient robot localization it is crucial to detect some salient landmarks in real-time and decide, which of them should be used by the localization procedure. Our hybrid sensor design supports this task by providing on-board image processing using a single-board computer Nvidia Jetson TX2 with GPGPU (General Purpose Graphics Processing Unit). The Jetson is used to rectify the omnidirectional images into undistorted panoramic images, and to compute distances to objects from a pair of images acquired by the sensor. For localization with our sensor artificial landmarks are employed. However, the approach we have proposed in [2] has some important drawbacks that result in too slow detection of

the landmark candidates in the omnidirectional images. Therefore, in this paper we propose a new approach that adopts a Convolutional Neural Network (CNN) to detect the artificial landmarks.

## 2 QR codes in robot localization

Passive landmarks are based on geometric shapes and colours. Using matrix patterns, such as the popular Quick Response (QR) codes, makes it possible to create unique landmarks for many types of objects in the environment. The landmarks can contain rich information about the objects they are attached to, including coordinates in the external reference system and description/class of the object, which allows the robot to reason about the semantics of the environment [4]. The localization algorithm presented in [2] uses QR codes with additional black frames. At first the omnidirectional image (Fig.1b) is transformed to the panoramic one, because the recognition of the landmarks shapes from the raw catadioptric images was unreliable. The extreme parts of the panoramic image are duplicated on the opposing sides in order to obtain a continuous image in the areas where landmarks may appear. Next, the potential landmarks candidates are found (Fig.1c.) using the rectangular shape of the black frame. However, to decide if the candidate shape is actually our landmark, it is necessary to decode the QR code. Consequently, the perspective camera of the sensor is used, which is moved to the area where the given landmark is located, and the QR code gets verified. Although the method from [2] allows the robot to find proper landmarks and localize itself, the calculations necessary to process the omnidirectional image are time consuming, while random rectangular shapes of high contrast are often mismatched with the QR codes.



**Fig. 1.** Hybrid field of view sensor (a), image acquired from the omnidirectional camera (b), and results of the non-learnable landmark detection procedure – a part of panoramic image with landmark candidates marked by rectangles (c)

## 3 Adapting YOLO for detection of QR landmarks

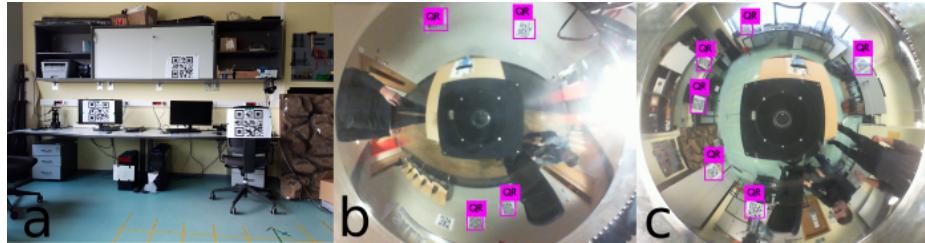
Taking the aforementioned disadvantages into account, it was decided to use a deep learning approach to recognize and localize the QR-based landmarks in images. As detection of objects in images is one of the main tasks CNNs are used for [5], there are many deep learning architectures we could adopt in our system. We decided to use the You Only Look Once (YOLO) version 3 [6][7], which is an extremely fast multi-object detection algorithm. Although it is considered to have worse precision than the popular Mask R-CNN (Region

Convolutional Neural Network) architecture [1] with respect to the localization of objects in images, in our task this is less important, because we only need to have enough information to position the perspective camera mimicking the natural eye fixation mechanism.

The biological inspiration for both YOLO and R-CNN networks is the brain area responsible for vision [5], where neurons sensitive to small subregions of input signals are located, covering the entire visual field. The cells are well adapted to the strong spatial correlation of brain-processed images and act as filters in the input space. YOLO is a single stage detector, which treats object detection as a regression problem. It applies a  $S \times S$  grid to the whole image capturing global context in prediction of object classes. For each cell of this grid a position of boundary box and confidence score of object detection are computed. Then, for each boundary box the probability of belonging to respective classes is calculated. The final result is a combination of information about the most probable positions of objects belonging to the classes the network has been trained for, and the corresponding class labels. An important advantage of this network is that the learning process considers objects of different sizes belonging to the specified class. This is important in the localization task, where the distances between the robot (sensor) and the landmarks are varying considerably.

The YOLOv3 algorithm was used on the Nvidia Jetson TX2 computer with the dual-core NVIDIA Denver2 and quad-core ARM Cortex-A57 CPU, 8GB 128-bit LPDDR4 memory and integrated 256-core Pascal GPU. In order to determine the impact of the computer's hardware configuration on the data processing speed, two different software configurations (with and without GPU) have been tested. To run the experiment, from 4 to 6 printed landmarks were placed on vertical surfaces in the laboratory room (Fig.2a.) and in the corridor. A dataset of 500 unique images was collected for different lighting conditions (artificial and natural light) and distances between the robot and landmarks (from 0.5m to 4 m). For the network learning process, a set of 300 images was used. The learning procedure was finished after 30000 step, as for more steps overtraining of the model was observed. The learned network was tested on a sample of the remaining 200 photos. We did not use a validation set, because of the rather small size of the whole dataset, which was laborious to acquire and label manually. Moreover, we were using a pre-trained YOLOv3 network, which did not required tuning of the meta-parameters.

The obtained results are presented in Fig. 2b, 2c and in Tab. 1. We can see that the accuracy of landmark recognition is the same for both hardware configurations and for different numbers of landmarks in the scene. Accuracy for the found landmarks was from 51% to 100%, and the average was 86%. As opposed to the non-learnable approach, for all testing images no false positives were detected. However, about 2% of false negatives (i.e. not detected landmarks) occurred. The most significant difference is the time needed to detect and recognize the landmarks. On the Jetson platform, the average time of landmark identification is 73 times smaller when GPU and CUDA are used. Moreover, the average time for selecting landmark's candidates, for previous approach with PC



**Fig. 2.** Operational environment (a), example results for five QR landmarks in the scene with a single false negative (b), example results for six QR landmarks in the scene with succesfull detections (c)

computer and landmarks with black frame, is equal 0.65s. Using solution based on YOLO algorithm and Jetson TX2, landmarks candidates were found faster and with better accuracy than in previous attempt.

| no. of QR<br>in image<br>(no. of images) | time ARM<br>CPU [s] | time<br>GPU [s] | no. of images<br>with all<br>QR detected | no. of images<br>with one QR<br>missing | no. of images<br>with more than one<br>QR missing |
|--|---------------------|-----------------|--|---|---|
| 4 (80)                                   | 2.724               | 0.036           | 74                                       | 6                                       | 0   |
| 5 (40)                                   | 2.741               | 0.035           | 29                                       | 9                                       | 2   |
| 6 (80)                                   | 2.759               | 0.034           | 69                                       | 10                                      | 1   |

**Table 1.** Recognition of QR codes with the adopted YOLO network

#### 4 Conclusions

It has been demonstrated that it is possible to use deep learning architecture for detection of specific pattern (QR code) in real time. Moreover, the results confirm that the discussed approach implemented on the Jeston TX2 is fast and feasible for detecting landmarks in the biologically-inspired scheme.

#### References

- He K., Gkioxari G., Dollár P., Girshick R.: Mask R-CNN, IEEE Int. Conf. on Computer Vision (ICCV), Venice, 2980–2988 (2017)
- Rostkowska M., Skrzypczyński P.: Improving Self-localization Efficiency in a Small Mobile Robot by Using a Hybrid Field of View Vision System, Journal of Automation, Mobile Robotics and Intelligent Systems, **9(4)**, pp. 28–38 (2015)
- Rostkowska M., Skrzypczyński P.: Hybrid field of view vision: From biological inspirations to integrated sensor design, Proc. IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems, Baden-Baden, 653–658 (2016)
- Figat J., Kasprzak W.: NAO-mark vs. QR-code Recognition by NAO Robot Vision, Progress in Automation, Robotics and Measuring Techniques, vol. 2 Robotics, (R. Szewczyk et al., eds.), Springer, Heidelberg, 55–64 (2015)
- Patterson J., Gibson A.: Deep Learning, Helion, Gliwice (2018)
- Redmon J., Divvala S., Girshick R., Farhadi A.: You Only Look Once: Unified, Real-Time Object Detection, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, pp. 779–788 (2016)
- Redmon J., Farhadi A.: YOLOv3: An Incremental Improvement ArXiv, preprint arXiv:1804.02767(2018)

# Usability of Reinforcement Learning Methods in the Task of Manipulation of Deformable Linear Objects

Michał Bednarek, Krzysztof Walas

Poznań University of Technology,  
Institute of Control, Robotics and Information Engineering,  
[michal.bednarek@put.poznan.pl](mailto:michal.bednarek@put.poznan.pl), [krzysztof.walas@put.poznan.pl](mailto:krzysztof.walas@put.poznan.pl)

## 1 Introduction

The manipulation of Deformable Linear Objects (DLO) such as wires, hoses or gaskets is still a challenge in the manufacturing processes, e.g. mechatronics assembly lines or automotive industry. The task is hard as the objects are not firm, and performing manipulation of DLOs is not possible in a fully controlled and repeatable way, as it is the case for rigid bodies. In our work, we have provided a comparison of the reinforcement learning methods in robotic manipulation task of deformable objects.

## 2 Related work

In this section, we provide a brief survey of existing machine learning methods used in a robotic manipulation of Deformable Linear Objects (DLOs). One of the approaches was presented by [4]. The work is mainly focused on modelling and perception of such objects. This work was further extended to a non-rigid registration method applied to a manipulation task [6]. This group presented final results in work [3], where authors are leveraging appearance priors in non-rigid registration. Another approach was presented in [5], where the deep-reinforcement method was used for learning the manipulation of deformable objects from observing humans doing manipulation of a rope. In the latest work [9], authors are introducing the concept of unit manipulations that are basic operations in the knotting procedure.

## 3 Background

In this section, we will formulate the reinforcement learning goal and introduce the required notation. Moreover, we will provide a piece of detailed information about each environment and algorithm used in our experiments. In the end, technical details will be provided.

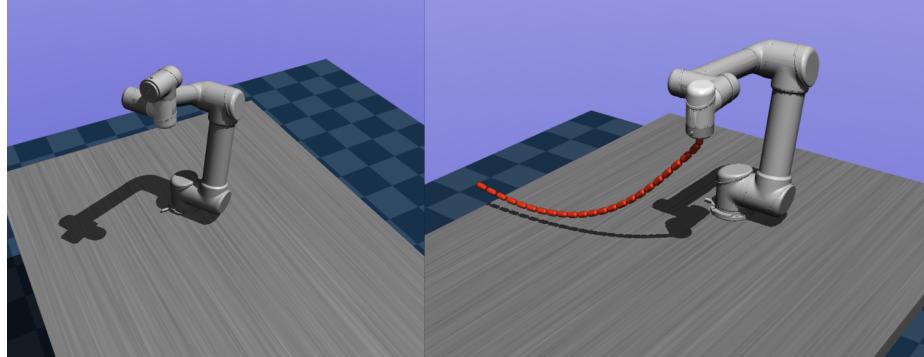
### 3.1 Notation

In our work we address the learning of on-policy and off-policy reinforcement learning algorithms in continuous action spaces. Each manipulation task presented in our paper

can be seen as the Markov Decision Process (MDP). That mathematical framework is depicted as a tuple  $(\mathcal{S}, \mathcal{A}, p, r)$ . State and action spaces  $\mathcal{S}$  and  $\mathcal{A}$  are considered to be continuous. A probability density function of a next state  $s_{t+1} \in \mathcal{S}$  is expressed as  $p : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \rightarrow [0, \infty)$ , given a current state  $s_t \in \mathcal{S}$  and action  $a_t \in \mathcal{A}$ . After each step an environment emits some reward  $r : \mathcal{S} \times \mathcal{A} \rightarrow [r_{min}, r_{max}]$ . In the standard reinforcement learning setup, the goal is to find such a policy  $\pi(s_t | a_t)$  that maximises the objective function defined as the expected sum of rewards:  $\sum_t E_{(s_t, a_t) \sim \pi}[r(s_t, a_t)]$ . In our experiments each algorithm holds that setup except the Soft Actor-Critic, which has the additional term responsible for maximising the entropy  $H$  at each state:  $\sum_t E_{(s_t, a_t) \sim \pi} E[r(s_t, a_t) + H(\pi(\cdot | s_t))]$ .

### 3.2 Environments

We prepared two simulation environments – the reaching task (we consider it to be relatively easy) and the manipulation task (more challenging). Both are presented in the picture 1. In both of them, we used the UR5 collaborative robotic arm where we can control velocity in its joints. The model of the simulated robot is based on the one available at MuJoCo resources website. Reward in the reaching task is presented in the Equation 1 and for the manipulation task in the Equation 2.



**Fig. 1.** Two simulation environments with the UR5 robotic arm.

$$r(s_t, a_t) = -c_1 d - c_2 a^T a \quad (1)$$

Both reward functions depend on the Huber distance  $d$  between an end-effector and a target point/consecutive joints in the rope. There is also an additional *smoothing term*  $a^T a$  that encourages an agent to take smooth actions and prevents applying too large actions. Indexed letters  $c$  are empirically adjusted constants. When the robot successfully finishes its task, a sparse reward term is added.

$$r(s_t, a_t) = -c_1 \left( \sum_{n=1}^N d_n \right) - c_2 a^T a \quad (2)$$

### 3.3 Algorithms

In the paper of Schulman et al. [7] authors presented an effective on-policy learning method called **Proximal Policy Optimisation (PPO)**. PPO aims to reduce the update of a policy network as much as it is possible while keeping reasonable policy updates.

Silver et al. presented one of the most popular off-policy algorithms [8] **Deep Deterministic Policy Gradients (DDPG)**. Authors presented a deep Q-learning approach based on the actor-critic architecture. In this algorithm, a critic tries to minimise the Bellman loss, while actor tries to predict the action that will maximise Q-value. An interesting alternative for previous algorithms is the **Importance Weighted Actor-Learner Architecture (IMPALA)** presented by Espeholt et al. [1]. This method was mainly designed to solve a large number of tasks with a single agent using the *V-trace* learning algorithm. **Soft Actor-Critic (SAC)** method presented by the Haarnoja et al. [2] is the only algorithm that stems from the maximum entropy reinforcement learning framework. It uses the modified version of the standard reinforcement learning objective that aims to maximise the reward, augmented with an entropy term.

### 3.4 Setup

Both simulation environments were prepared in the MuJoCo physics simulator [10]. Communication between RL methods and the simulator was possible by using **mujoco py** and **gym** libraries from OpenAI. PPO, DDPG and IMPALA algorithms were tested using the Python/Tensorflow implementation from **rllib** library. The implementation of SAC came from the Berkeley AI Research repository named *softlearning*. Every algorithm was run using a default **rllib** configuration file, so the SAC algorithm was adjusted to meet the same configuration as others.

## 4 Results

In the results section, we have tested four methods which were described in the previous section. These different approaches were tested in two tasks reaching and manipulation task. In the first task – reaching to the point, the worst-performing method in our particular reaching task is DDPG together with IMPALA. On the opposite, the SAC method is performing the best, reaching positive rewards while other methods stay below zero. In the more complicated task of folding the DLO to the selected shape, the only method which was able to learn without breaking the simulation was SAC. The robot reached the low cumulative distance, which means that it was close to replicating the predefined sinusoidal shape.

## 5 Conclusions

The method which is performing the best in both tasks is the Soft Actor-Critic. It is worth underlying the fact that SAC is the only method that can tackle the DLO manipulation problem. In our work, we presented a comparison of four different reinforcement learning methods, which were applied to two tasks. The first one was reaching, where all the

algorithm were able to provide some results and the best one was SAC. In the second task of DLO manipulation, only SAC was able to deliver the result without braking the simulation and have the convergence of the reward function.

## References

1. Espeholt, L., Soyer, H., Munos, R., Simonyan, K., Mnih, V., Ward, T., Doron, Y., Firoiu, V., Harley, T., Dunning, I., Legg, S., Kavukcuoglu, K.: IMPALA: Scalable distributed deep-RL with importance weighted actor-learner architectures. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 80, pp. 1407–1416. PMLR, StockholmsmÅdssan, Stockholm Sweden (10–15 Jul 2018), <http://proceedings.mlr.press/v80/espeholt18a.html>
2. Haarnoja, T., Zhou, A., Hartikainen, K., Tucker, G., Ha, S., Tan, J., Kumar, V., Zhu, H., Gupta, A., Abbeel, P., Levine, S.: Soft actor-critic algorithms and applications. Tech. rep. (2018)
3. Huang, S.H., Pan, J., Mulcaire, G., Abbeel, P.: Leveraging appearance priors in non-rigid registration, with application to manipulation of deformable objects. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 878–885 (Sep 2015). <https://doi.org/10.1109/IROS.2015.7353475>
4. Javdani, S., Tandon, S., Tang, J., O'Brien, J.F., Abbeel, P.: Modeling and perception of deformable one-dimensional objects. In: 2011 IEEE International Conference on Robotics and Automation. pp. 1607–1614 (May 2011). <https://doi.org/10.1109/ICRA.2011.5980431>
5. Nair, A., Chen, D., Agrawal, P., Isola, P., Abbeel, P., Malik, J., Levine, S.: Combining self-supervised learning and imitation for vision-based rope manipulation. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). pp. 2146–2153 (May 2017). <https://doi.org/10.1109/ICRA.2017.7989247>
6. Schulman, J., Ho, J., Lee, C., Abbeel, P.: Generalization in robotic manipulation through the use of non-rigid registration. In: Proceedings of the 16th International Symposium on Robotics Research (ISRR) (2013)
7. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. ArXiv **abs/1707.06347** (2017)
8. Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., Riedmiller, M.: Deterministic policy gradient algorithms. In: Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32. pp. I–387–I–395. ICML’14, JMLR.org (2014), <http://dl.acm.org/citation.cfm?id=3044805.3044850>
9. Takizawa, M., Yao, Z., Onda, H., Kudoh, S., Suehiro, T.: Learning from observation of tabletop knotting using a simple task model. In: 2019 IEEE/SICE International Symposium on System Integration (SII). pp. 85–91 (Jan 2019). <https://doi.org/10.1109/SII.2019.8700429>
10. Todorov, E., Erez, T., Tassa, Y.: Mujoco: A physics engine for model-based control. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 5026–5033 (Oct 2012). <https://doi.org/10.1109/IROS.2012.6386109>

# Improving Person Re-identification by Segmentation-Based Detection Bounding Box Filtering

Dominik Pieczyński, Marek Kraft, Michał Fularz

Poznań University of Technology

**Abstract.** In this paper, a method for improving the accuracy of person re-identification results is presented. The method is based on the assumption, that including segmentation information into re-identification pipeline discards the automated detections that are of poor quality.

**Keywords:** person re-identification · deep learning · image processing · neural networks.

## 1 Problem definition

Person re-identification is one of the most prominent tasks in video surveillance systems. While the introduction of approaches based on deep neural networks caused a breakthrough change in re-identification accuracy, reaching over 80% rank-1 accuracy on challenging datasets, person re-identification is still a challenging task. Datasets used to train and test the deep learning approaches are based on automatic detection, which might give rise to various problems that the video surveillance system operating under realistic conditions must also be able to cope with.

In this paper, the influence of using segmentation priors on the accuracy of person re-identification is evaluated. The idea is based on the fact, that in video surveillance systems re-identification will most likely be preceded by object detection. Since methods for joint object detection and segmentation, such as Mask-RCNN that is used in this work, are available, the detection bounding box and its internal segmentation result can be used as an input to a simple rule-based system, that can decide which image pairs are potentially invalid and therefore should not be processed. The rules can filter out images that are invalid due to occlusions, misplaced regions of interest (ROI), multiple persons found within a single ROI, etc. using a simple segment number, bounding box fill rate and aspect ratio check.

## 2 Results and conclusions

A method for improving the accuracy of person re-identification was proposed. The method uses segmentation priors to filter out the problematic images, whose analysis might give rise to errors. The computational cost of computing the aspect ratio and fill ratio is low, yet the improvement in re-identification accuracy is noticeable, even though a state of the art method is used as baseline.

# From the Edge to the Datacenter: Evaluating the Throughput and Power Efficiency of Deep Learning Hardware Platforms<sup>\*</sup>

Marek Kraft, Dominik Pieczyński, and Michał Fularz

Poznań University of Technology, Piotrowo 3A, 60-965 Poznań, Poland  
`{name.surname}@put.poznan.pl`

**Abstract.** In this paper, we perform a thorough analysis of performance and power consumption of a wide range of computational platforms in a task representative for smart camera networks – person re-identification based on deep convolutional neural network.

**Keywords:** deep learning · hardware accelerators · GPU.

## 1 Problem definition

Person re-identification is one of the most important tasks in distributed video surveillance systems. Introduction of deep, convolutional neural networks enabled significant gains of performance in a wide range of computer vision tasks, and re-identification is no exception. However, training and prediction using deep neural networks is considered a computationally intensive task.

Since the distributed, edge processing paradigm becomes more and more widespread, it is important to answer the question whether or not the low power, low footprint embedded hardware is up to the task, or is sending the sensor data to a central node for large batch processing a better solution? In this paper, we evaluate a range of deep learning hardware, ranging from desktop and embedded CPUs, through embedded and desktop GPUs, up to dedicated, low power hardware accelerators in an attempt to answer this question.

## 2 Conclusions

The results clearly show, that GPUs are in general a better choice for deep learning inference due to their inherent capability to perform computations in parallel. Desktop GPU consume significant amounts of power and exhibit the best performance, making them an attractive choice for batch processing. The tested embedded GPU is a capable device and can be applied in edge processing. The tested deep learning accelerators are certainly interesting devices, especially in terms of performance to power consumption ratio and one can certainly expect their proliferation in computer vision edge processing in the future.

---

<sup>\*</sup> The authors thank Nvidia for hardware donation under Nvidia Academic Hardware Grant.

# Kinematic Structures Detection and Estimation with Neural Network and Black-box Optimization\*

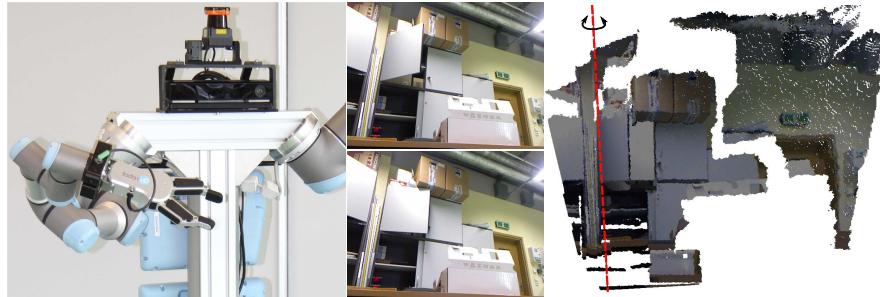
Dominik Belter<sup>1</sup>[0000–0003–3002–9747]

Institute of Control, Robotics and Information Engineering, Poznan University of  
Technology, Poznan, Poland [dominik.belter@put.poznan.pl](mailto:dominik.belter@put.poznan.pl)

**Abstract.** In this paper, we propose a system for the detection and estimation of kinematic structures observed by a mobile manipulating robot equipped with the RGB-D camera. We applied a CNN-based detector to find the articulated object on the pair of the RGB-D images. We propose the optimization-based procedure to find the pose and configuration of the joint detected on the scene. In the experimental results, we show the performance of the proposed method. We also show that the method allows segmenting the scene observed by the robot.

**Keywords:** Object detection, deep learning in robotics, articulated objects

## 1 Introduction

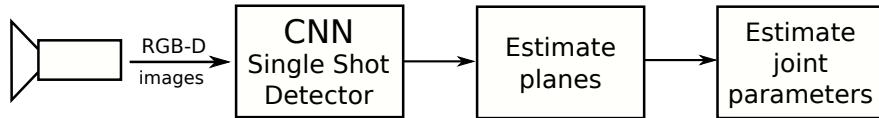


**Fig. 1.** Example scenarios which describe the problem solved in this paper: the robot observes the dynamic scene using RGB-D sensor. From the obtained pairs of RGB-D images, the robot identifies the type, pose, and configuration of the joints in the scene.

Autonomous robots should detect objects in the environment and understand their meaning to operate without human supervision. Recently, great progress

---

\* This work was supported by the National Centre for Research and Development (NCBR) through project LIDER/33/0176/L-8/16/NCBR/2017



**Fig. 2.** Block diagram for the detection and kinematic structure estimation of articulated objects

was made in the object detection [1] and pose estimation [3]. These methods allow for detecting and grasping the objects by a mobile manipulating robot. However, the robot should be also capable to interact with articulated objects like doors to perform manipulation tasks and move freely in the indoor environment.

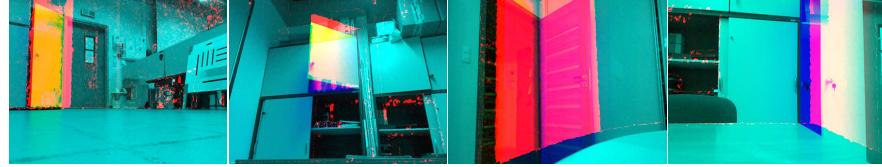
In this paper, we deal with the problem of the detection and kinematic structure estimation of articulated objects like doors. Our mobile manipulating robot (Fig. 1) is equipped with multiple RGB-D cameras which return RGB images and measure the distance to the objects. Data from the sensor are used to detect articulated objects like doors and estimate their kinematic structures. We use two pairs of RGB-D images to find the pose of the joint. The time between the frames does not play an important role because the configuration of the joint is also estimated. We are focused on the object with one degree of freedom (DOF) and we assume that the robot does not move when observing a dynamic scene. The example scenario is shown in Fig. 1.

The problem of identification of kinematic models for the articulated objects has been studied in the literature using formal approaches. Sturm et al. applied a probabilistic framework to learn the kinematic structures from the camera images [7]. Katz and Brock estimate the structure of the object during on-line manipulation but they assume that the robot deals with 2D models only [2]. In our research, we applied a CNN-based object detector to quickly detect the kinematic structures on the RGB-D images. Then, we applied the optimization-based framework to estimate the parameters of the kinematic structure.

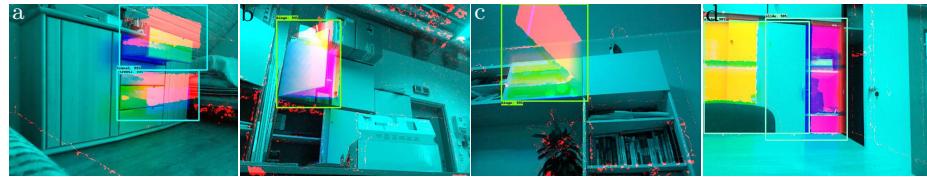
## 2 Kinematic structure estimation of articulated objects

The architecture of the proposed system is presented in Fig. 2. The robot utilizes two pairs of RGB-D images. Then, we use the SSD object detection framework to find articulated objects. The obtained bounding boxes are later used to find the pose and the configuration of the 1 DOF joint. To this end, we find the planes in the selected area. Then, we estimate the parameters of the joint by applying a black-box optimization algorithm.

To detect articulated objects we applied Single Shot Detector (SSD) with lightweight mobilenet v1 architecture for features extraction [5]. We provide the image with three layers on the input of the neural network. Two first layers represent grayscale images of the consecutive frames. The third layer is the difference between two consecutive depth images normalized into range 0–255.



**Fig. 3.** Example input images for the SSD-based object detector

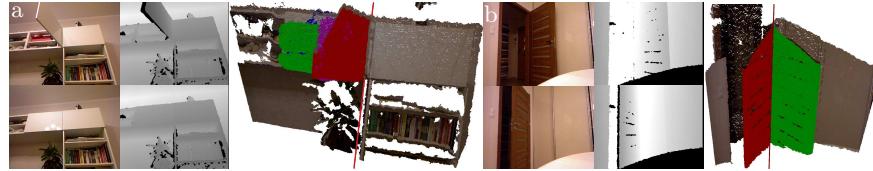


**Fig. 4.** Example detections from the SSD neural network: two drawers (translational joints) (a), two hinge joints (b,c), and sliding doors (c)

Example input images are presented in Fig. 3. We trained the neural networks to detect three types of objects: hinge joints and two types of prismatic joints (sliding doors and drawers). Our detector uses RGB and depth data and works better than the detector with RGB data only. We trained the neural network on the 120 examples. The classification loss and localization loss after 10000 epochs are less than 0.5 and 0.1, respectively.

To find the parameters of the kinematic joint we extract the point cloud from the region determined by the bounding box. First, we determine the plane representing the door before (first frame) and at the end of the motion (second frame). To this end, we applied the Random Sample Consensus (RANSAC) [6] algorithm. The algorithm randomly selects three points from the local point cloud. Using three points, we compute the equation of the plane. Then, we check the fitting of the remaining points to the plane. If the number of outliers is below the threshold and the number of iterations is below the threshold we stop the search with success. We run this procedure to find the planes at the beginning and the end of the motion. The estimated planes are used to find the initial parameters of the joint (position and orientation defined by 6 values) and joint configuration.

The properties of the joint obtained from the RANSAC-based method are used as an initial guess to the optimization-based estimation framework. In this method, we estimate the parameters and configuration of the joint (7 values) using Particle Swarm Optimization [4]. We project the points from the first frame to the second frame to compute the fitness value. This approach allows efficiently explaining the data on both images using the given hypothesis (joint properties and configuration).



**Fig. 5.** Example results of the hinge joint estimation. Each subfigure contains two pairs of the RGB-D images, 3D point cloud and obtained rotation axis. Red and green points represent points covering rotating object.

### 3 Results

The example detection results for the testing dataset are presented in Fig. 4. The proposed method can precisely determine the bounding box on the prepared image from the pair of RGB-D frames. Then, the obtained bounding boxes are used to estimate the kinematic structures. The example results for the hinge joints are presented in Fig. 5. For each example, we present two input frames (RGB-D images) and obtained point clouds with the estimated joint pose. Our method also allows segmenting the scene and extract pixels that belong to the articulated objects (red and green point clouds in Fig. 5).

### 4 Conclusions and future work

In this paper, we present the hybrid CNN and optimization-based method for estimating the kinematic structures in the natural indoor environment. The method is dedicated to the mobile manipulating robots equipped with RGB-D sensors. In the future, we are going to extend the experimental verification of the proposed method and show the results for a multiple of joints and overlapping detections.

### References

1. K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN, IEEE International Conference on Computer Vision, pp. 2980–2988 (2017)
2. D. Katz, Y. Pyuro, O. Brock, Learning to Manipulate Articulated Objects in Unstructured Environments Using a Grounded Relational Representation, Robotics: Science and Systems IV, Zurich, Switzerland (2008)
3. W. Kehl, F. Manhardt, F. Tombariand, S. Ilic, N. Navab, SSD-6D: Making RGB-Based 3D Detection and 6D Pose Estimation Great Again, IEEE International Conference on Computer Vision (ICCV), pp. 1530–1538 (2017)
4. J. Kennedy, R. Eberhart, Particle Swarm Optimization, Proceedings of IEEE International Conference on Neural Networks, pp. 1942–1948 (1995)
5. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, C.S. Reed, C.-Y. Fu, A.C Berg, SSD: Single Shot MultiBox Detector, European Conference on Computer Vision (2016)
6. T. Strutz, Data Fitting and Uncertainty, Springer Vieweg, 2nd edition (2016)
7. J. Sturm, C. Stachniss, W. Burgard, A Probabilistic Framework for Learning Kinematic Models of Articulated Objects, Journal of Artificial Intelligence Research, vol. 41, pp. 477–526 (2011)

**NLP+ASR+CAI** (natural language processing,  
automatic speech recognition, and conversational  
AI)

# Parliamentary election's predictions using social media content

Antoni Sobkowicz<sup>1\*</sup>, Marek Kozłowski <sup>1</sup>

<sup>1</sup>National Information Processing Institute Warsaw, Poland  
[antoni.sobkowicz,marek.kozlowski@opi.org.pl](mailto:antoni.sobkowicz,marek.kozlowski@opi.org.pl)

**Abstract.** Political news are a major part of many information websites and are often emotionally commented on by a huge number of users. Using such data (counted in millions of comments) we searched for an automated way to find coherent groups of users that produce content that is likely to evoke negative emotions towards political parties. Combining time patterns with machine learning methods we build a toolkit to infer an opinion poll using web public data. We proved that such methodology brings very similar results in comparison to the classical approaches used by major opinion polling institutes.

**Keywords:** text classification, neural nets, clustering, word embeddings

## 1 Introduction

Political discussions in social media are overloaded with a high amount of emotional content. The anonymity provided by the Internet allows to write insulting statements, discredit other people and engage people with little to no consequences. While insulting opposition may not directly correspond with being a supporter of other side (for example, insulting PO politicians may not mean that the poster is a PiS supporter), given Poland's mostly two-party system and heavy right-left division, one can argue that this is sufficient if we group support for parties into Left-Right blocks and next predict a poll using such assumption.

## 2 Related Work

Sentiment analysis and emotion detection are very well known topics [6], with papers covering sentiment analysis of Twitter [5], product reviews and movie reviews among others. Most of the approaches are focused on assessing the sentiment of text in relation to what the writer wanted to express, not with the emotion it evokes in the reader.

Analysis of political texts to predict political sentiment provide acceptable results in order to predict election results. In the paper [10] authors used the context of the German federal election to investigate whether Twitter is a relevant forum for political deliberation and whether online messages on Twitter validly mirror offline political sentiment.

The problem of detecting users who produce content that invokes emotions in readers can be connected with the topic of troll detection<sup>1</sup>. The one recent work [7] uses sentiment analysis approach, based on recursive neural network and support vector machines, to detect trolls from Singaporean forum. Similar research [3], using time patterns, posting statistics and user sentiment feedback, was conducted on Bulgarian Internet community forum dnevnik.bg with troll detection accuracy over 90%, although their definition of troll can be debatable, with user being tagged as a troll when sufficient number of other users calls them that.

Surveys on polish political fora using machine learning and NLP methods are rarely published, however with an emphasis on both sentiment analysis done by human annotators with additional keyword-based automatic analysis [9, 8]. In the current work, we move further in order to infer polls using our improved methodology.

### 3 Approach

**Data Set** The surveyed data set consists of 4 000 000 comments scrapped from Interia.pl website<sup>2</sup>. All comments were saved along with information about their position in a discussion tree, date of posting and information about the user. We gathered comments spanning from December 2016 to May 2018. In comparison to the previous research, application of data collected from other major news sites (e.g. onet.pl) was hard because of comments systems that enforce registration, which makes comment volumes drastically smaller.

The poll results were taken from ewybory.eu website. For each month we collected all poll results provided by all major polish opinion polling institutes and averaged them in blocks - Right-aligned block (consisting of Prawo i Sprawiedliwość and Kukiz'15) and Left-aligned block (consisting of Platforma Obywatelska, .Nowoczesna and SLD).

**Political comments evoking emotions** Our psychological analysis shows that most of the users do not read entire comments, what was mentioned also in [2]. The observations suggest that in recent years people increasingly tend to skim through the text and to look for evoking emotions keywords. At the same time, human perception relies largely on transient connotations of common terms with specific contexts, people or situations. The language used in such communication frequently relies on modifications of words intended to be abusive (and understood as such in a given community). Therefore we used a set of negative nicknames of polish political figures and events (called further emotive seeds), as they tend to convey negative emotions of the writer, and are prone to evoke negative emotions among readers.

---

<sup>1</sup> trolls is a user who produces text disrupting discussion by engaging users into emotional discussions

<sup>2</sup> <https://www.interia.pl/>

**Detection of highly emotive words** To detect comments and user polarities we used a list of emotion evoking tokens used by posters to insult opposing party. Emotion evoking tokens is a union of emotive seeds and the similar words retrieved using Word2Vec[4] model. We built the Word2Vec model on all collected comments and then query the model to get the most similar semantically words to emotive seeds. This step allowed us to get 61 highly emotion evoking tokens that are used by Left side to insult right one, and 65 tokens used by Right to insult Left.

**Building comments classifier** As an additional step, to build larger corpora of Right- and Left-aligned comments, we marked all comments written by the user as Right or Left according to appearance of the previously detected highly emotion evoking tokens. This allowed us to generate large corpora of tagged comments, that can be used as a training set (consisting of around 2 millions comments) to build Doc2Vec models. Doc2Vec [1] is an NLP method for representing documents as a vector and a generalization of the Word2Vec method [4], it was selected as it provides good accuracy in classification tasks concerning short texts as posts/comments/tweets.

All comments were preprocessed with a simple pipeline consisting of lower-casing, punctuation removal, lemmatization.

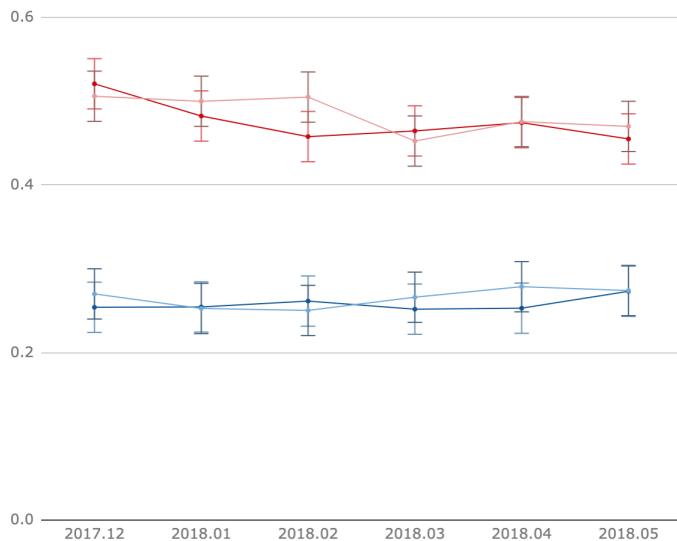
The first classification model was a  $k$ -NN classifier using text embeddings from Doc2Vec model. For each new comment we infer the vector from Doc2Vec model and we find the  $k$  most similar comments from the training set. If all  $k$  (by default 3) comments have the same category we assign it to the new comment, otherwise the comment is left as a neutral one.

The second classifier used averaged Doc2Vec embeddings for each category - and new comments were compared against them, the most similar category vector exceeding the defined similarity threshold determine the final category, otherwise the comment is left as a neutral one.

**Estimation of election poll** We analyzed comments, that were not included in a training set, using above defined classifiers. Next using all tagged comments (the training set and those properly classified) we counted how many commenters are likely to vote for a given party. This data was then grouped by month, and for each month we calculated the percentage of users that were Right- or Left-aligned from all users that posted this month.

## 4 Evaluation

Between December 2017 and May 2018, raw party support (after rescaling based on average poll support to the average number of comments in the whole time-frame) between comment derived data and classical poll data was within the margin of error (3 percent points, typical for poll-based support measurements). Correlation (Person) between comment based support and poll data for each party was - 0.43 - high for Right and 0.07 - a very weak for Left.



**Fig. 1.** Distribution of election poll estimations based on party's classically measured support as e.g CATI/CAVI methods (lighter colors, blue for Left, red for Right side) with machine learning based support (darker colors respectively). Charts show typically around 3 percentages margin between the approaches.

**Summary** Machine learning-based models for social media political party support seem very promising. Ability to deliver raw party support numbers from textual comments - within the margin of error of standard poll-based methods and higher correlation of D2V based method may lead - after further improvements - into the development of a new way to measure changes in political stance in the country.

## References

1. Le, Q., Mikolov, T.: Distributed representations of sentences and documents. In: International conference on machine learning. pp. 1188–1196 (2014)
2. Liu, Z.: Reading behavior in the digital environment. Journal of Documentation 61(6), 700–712 (2005)
3. Mihaylov, T., Georgiev, G., Nakov, P.: Finding Opinion Manipulation Trolls in News Community Forums. Proceedings of the Nineteenth Conference on Computational Natural Language Learning pp. 310–314 (2015)
4. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013)
5. Pak, A., Paroubek, P.: Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In Proceedings of the Seventh Conference on International Language Resources and Evaluation pp. 1320–1326 (2010)

6. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Foundations and trends in information retrieval 2(1-2), 1–135 (2008)
7. Seah, C.W., Chieu, H.L., Chai, K.M.A., Teow, L.n., Yeong, L.W.: Troll Detection by Domain-Adapting Sentiment Analysis. 18th International Conference on Information Fusion pp. 792–799 (2015)
8. Sobkowicz, A., Kozłowski, M.: An application of automatic sentiment analysis methods in web-political discussions
9. Sobkowicz, P., Sobkowicz, A.: Two-year study of emotion and communication patterns in a highly polarized political discussion forum. Social Science Computer Review pp. 448–469 (2012)
10. Tumasjan, A., Sprenger, T.O., Sandner, P.G., Welpe, I.M.: Predicting elections with twitter: What 140 characters reveal about political sentiment. ICWSM 10, 178–185 (2010)

# Neural style transfer for non-native speech recognition

Kacper Radzikowski<sup>1</sup>, Robert Nowak<sup>2</sup>, and Osamu Yoshie<sup>1</sup>

<sup>1</sup> Waseda University, Graduate School of Information, Production and Systems

<sup>2</sup> Warsaw University of Technology, Institute of Computer Science

**Abstract.** Automatic speech recognition (ASR) has been an object of extensive research since the second half of the previous century. ASR systems achieve high accuracy rates, however, only when the system is used for recognizing the speech of native speakers. The score drops in case the ASR system is being used with a non-native speaker of the language to be recognized, as the pronunciation is affected by the patterns of the mother tongue. Traditional approaches for developing speech recognition classifiers are based on supervised learning, relying on the existence of large labeled datasets. In case of non-native speech such datasets do not always exist and even if they do, the number of samples is not always high enough to train accurate classifiers. We have dealt with the problem of the non-native speech in our previous research using different approach of dual-supervised learning [12]. This time, we try tackling the problem using the style transfer methodology. We designed a pipeline for modifying the non-native speech, so that it resembles the native one to a higher extent. In this research, we plan to tackle the problem of non-native accent, using style transfer methodology. We adjust style transfer to the domain of speech and sound, to create an algorithm for real-time accent modification. Such an approach could allow to modify non-native speaker's voice on-the-fly, so that the ASR system can recognize the speech with higher accuracy. Our methodology could potentially be used as a wrapper for existing ASR system, reducing the necessity of training new algorithms for non-native speech.

**Keywords:** speech recognition, style transfer, non-native speaker, machine learning, deep learning, neural networks

## 1 Introduction

Automatic speech recognition is a task that has been an object of extensive research since the second half of the previous century. Its main purpose was to allow communication between a human and a machine, using the medium which is the most natural way for a human to convey a message.

Speech recognition techniques and methodologies that were developed recently, can recognize speech with up to 90-95% accuracy, depending on the used dataset and benchmark test ([20]). Such accuracy levels however, can be reached only when the system is used for recognizing the speech of native speakers (e.g.

English language for North American people). In case of non-native speakers, even the most advanced speech recognition systems, can achieve an accuracy not higher than 50-60%. The main reason for such a drop, is that non-native speakers have a different mother tongue than the one that is being recognized. Most people use their native language most often, which makes their pronunciation affected by the patterns and characteristics of their mother tongue. This makes their pronunciation of a foreign language biased to some extent, therefore speech recognition systems fail in such cases ([16, 10, 17, 18]). Global integration creates the need to properly recognise non-native speakers, who are nowadays the vast majority of users. Similar problem is improving ASR results for children[11].

In traditional methods of training speech recognition classifiers, supervised learning techniques are usually utilized ([2, 4, 5, 9, 1, 3]). While perfectly well fitted for cases of recognizing speech of tens of most popular languages worldwide, supervised learning methodologies will not supply classifiers of a decent quality in case of non-native speakers. The main reason is lack of labeled datasets, large enough, for non-native speakers. We have dealt with the problem of the data scarcity regarding the non-native speech in our previous research ([12, 14, 15, 13]). Our idea was to use unlabeled datasets (e.g. Japanese people that speaks English words without labels).

The application of style transfer in audio domain is not new itself. In [7] authors investigated how to transfer the style of a reference audio signal to a target audio content. They proposed a flexible framework for the task, which uses a sound texture model to extract statistics characterizing the reference audio style, followed by an optimization-based audio texture synthesis to modify the target content. In contrast to mainstream optimization-based visual transfer method, the process proposed by the authors is initialized by the target content instead of random noise and the optimized loss is only about texture, not structure.

In [19] authors presented a new machine learning technique for generating music and audio signals. The focus of their work was to develop new techniques parallel to what has been proposed for artistic style transfer for images by others. They presented two cases of modifying an audio signal to generate new sounds. A feature of their method is that a single architecture can generate these different audio-style-transfer types using the same set of parameters which otherwise require complex hand-tuned diverse signal processing pipelines.

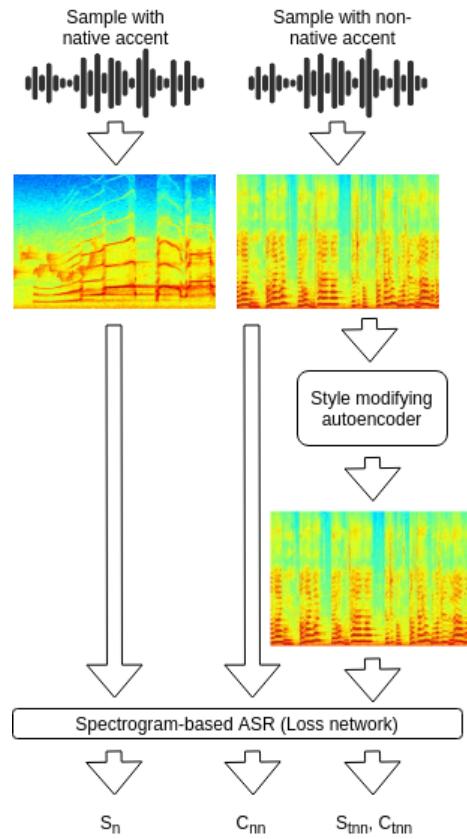
This time, in our research, we plan to tackle the problem of non-native accent, using style transfer methodology ([6, 8]). Broadly speaking, we plan to apply and adjust style transfer to the domain of speech and sound. Having done that, it would enable the possibility to create a wrapper over already existing and trained ASR systems. Such an approach could allow to modify non-native speaker's voice on-the-fly, so that the ASR system being used at the time can recognize the speech with higher accuracy.

## 2 Proposed methodology

In this paper we propose an approach for handling the problem related to specific, non-native accent. In this approach, we decided to employ a style transfer methodology adapted for the domain of speech and sound. Specifically, we decided to create a method resembling the style transfer feedforward method for the graphical domain.

In the problem of the graphical style transfer we try to modify an image in a way that its style resembles style of another, so called, style image. At the same time, the content of the image ideally should not be modified.

The general flow of the accent modification using style transfer is depicted in Fig. 1.



**Fig. 1.** The basic diagram of style transfer-based accent modification

The very first step is training a network (here called Loss network) which will be used as a speech recognizer in the style transfer approach. Its role is to separate speech spectrograms into multiple layers using convolutional network. It will be used for extracting content (related to the utterance) and style (related to the accent and pronunciation) from the spectrogram. For automatic speech recognition task, we combined properties of convolutional and recurrent layers. Convolutional neural networks has been proven to give outstanding results when applied to images, here spectrograms. They are able to detect and learn local features which are later passed on to recurrent layers. The architecture of neural network is depicted in Table 1.

**Table 1.** Detailed architecture of the CNN-RNN model used as the Loss network in style transfer approach

| Layer                 | Output shape    | Parameters |
|-----------------------|-----------------|------------|
| InputLayer            | None, None, 161 | 0          |
| Conv1D                | None, None, 220 | 389840     |
| Conv1D                | None, None, 220 | 389840     |
| Maxpool               | None, None, 220 | 880        |
| Conv1D                | None, None, 150 | 265800     |
| Conv1D                | None, None, 150 | 265800     |
| Maxpool               | None, None, 150 | 600        |
| Conv1D                | None, None, 100 | 177200     |
| Conv1D                | None, None, 100 | 177200     |
| Maxpool               | None, None, 100 | 400        |
| Conv1D                | None, None, 80  | 141760     |
| Conv1D                | None, None, 80  | 141760     |
| Maxpool               | None, None, 80  | 320        |
| Conv1D                | None, None, 80  | 141760     |
| Conv1D                | None, None, 80  | 141760     |
| Maxpool               | None, None, 80  | 320        |
| Conv1D                | None, None, 80  | 141760     |
| Conv1D                | None, None, 80  | 141760     |
| Bidirectional         | None, None, 400 | 505200     |
| BatchNormalization    | None, None, 400 | 1600       |
| TimeDistributed       | None, None, 29  | 11629      |
| Dropout               | None, None, 29  | 0          |
| TimeDistributed       | None, None, 29  | 870        |
| SoftmaxActivation     | None, None, 29  | 0          |
| Total params:         |                 | 3,038,059  |
| Trainable params:     |                 | 3,038,059  |
| Non-trainable params: |                 | 0          |

Second step is training the style modifying autoencoder. Its architecture is described in details in Table 2. During the training step, the samples with native

**Table 2.** Detailed architecture of the autoencoder

| Layer                 | Output shape          | Parameters |
|-----------------------|-----------------------|------------|
| Conv2D                | None, None, None, 32  | 417344     |
| Conv2D                | None, None, None, 64  | 18496      |
| Conv2D                | None, None, None, 128 | 73856      |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2D                | None, None, None, 128 | 147584     |
| Conv2DTranspose       | None, None, None, 64  | 73792      |
| Conv2DTranspose       | None, None, None, 32  | 18464      |
| Conv2D                | None, None, None, 32  | 82976      |
| Total params:         |                       | 2,160,768  |
| Trainable params:     |                       | 2,160,768  |
| Non-trainable params: |                       | 0          |

accent are transformed into the spectrograms and fed into the Loss network, which extracts style matrix  $S_n$  from certain layers. Next, the sample with native accent is transformed into a spectrogram, which is fed to the same Loss network for the extraction of content matrix  $C_{nn}$ . The sample is also fed into the style modifying autoencoder, after which its output gets fed into the Loss network to extract transformed non-native style matrix  $S_{tnn}$  and content matrix  $C_{tnn}$ .

After having received  $S_n$ ,  $C_{nn}$ ,  $S_{tnn}$ ,  $C_{tnn}$ , we can formulate the style loss and content loss. Content loss is calculated as:

$$L_c = \sum_l \sum_{i,j} (\alpha C_{nn}^l_{i,j} - \alpha C_{tnn}^l_{i,j})^2 \quad (1)$$

where  $l$  is the set of convolutional layers representing the speech content.

Style loss is calculated as:

$$L_s = \sum_l \sum_{i,j} (\beta G_{nn}^l_{i,j} - \beta G_{tnn}^l_{i,j})^2 \quad (2)$$

where  $G_{nn}^l$  is the Gram matrix of  $l$ th layer of  $C_{nn}$  received from the Loss network and  $G_{tnn}^l$  is the Gram matrix of the  $l$ th layer of  $C_{tnn}$ .

Therefore, the final loss function is represented as:

$$L = L_s + L_c \quad (3)$$

After having formulated our loss function, we use backpropagation algorithm to train the style modifying autoencoder network for the task of accent modification.

### 3 Experiments and evaluation

#### 3.1 Datasets

We use a set of 75000 recordings of *English Speech Database Read by Japanese Students (UME-ERJ)* containing Japanese pronouncing English sentences as well as Americans pronouncing the same utterances, which consist of:

1. Sentences for learning phonemic pronunciation:
  - 460 phonetically-balanced sentences,
  - 32 sentences including phoneme sequences difficult for Japanese to pronounce correctly,
  - 100 sentences designed for test set,
  - 302 minimal-pair words,
  - 300 phonemically balanced words.
2. Sentences for learning prosody of speech:
  - 94 sentences with various intonation patterns,
  - 120 sentences with various accent and rhythm patterns,
  - 109 words with various accent patterns.

The same dataset was used in our previous work[12].

Another dataset used in the research is LibriSpeech. It was used to train the ASR module used as the last part of our pipeline. The algorithms and the relative increase in the accuracy were decided based on the results yielded by the module trained on LibriSpeech.

Another application of the dataset in our research is training the Loss network for the style transfer-based variant of accent modification process.

#### 3.2 Experiments and metrics

In our research we designed separate experiments for several processes in our pipeline. Namely, we performed experiments and evaluated the relative improvement in the speech recognition accuracy in case of audio style transfer-based accent modification.

#### 3.3 Metrics

We employed two different evaluation metrics depending on the experiment type.

As a quality metric for a speech recognition neural network (Loss network and ASR module), we chose Character Error Rate, which is expressed as:

$$CER = \frac{i + s + d}{n}$$

where:

i number of insertions,  
 s number of substitutions,  
 d number of deletion,  
 n total number of characters.

As for the evaluation of the accent modification itself, we decided to present a relative decrease in CER yielded by the ASR module from the last part of our pipeline.

### 3.4 Results

The ASR module was trained using *LibriSpeech train-clean-360* subset. It was evaluated using the *test-clean* dataset, with which it achieved **15.7%** CER. This model evaluated on a 10% test subset of the *UME-ERJ* dataset achieved only **46.3%** CER. The results are an average of 10 runs of respective experiment.

After activating the style transfer-based accent modification in our pipeline, we performed several experiments on the same test subset of the *UME-ERJ* dataset, each differing in numbers of style and content layers. The best setup gave a result of **31.7%** CER (evaluation of the same model trained only on the *LibriSpeech* training set). Therefore, it yielded a **32%** relative improvement in terms of CER. All experimental results of this approach are presented in Table 3.

**Table 3.** Relative improvement depending on the content and style layers

| Style layers | Content layers | Relative improvement |
|--------------|----------------|----------------------|
| 1-10         | 6-12           | 32%                  |
| 1-8          | 8-12           | 29.6%                |
| 1-10         | 10-12          | 30.1%                |
| 1-5          | 5-12           | 26.7%                |
| 1-4          | 4-12           | 15.6%                |

## 4 Discussion and future work

This manuscript contains preliminary study of audio style transfer methods used for improve ASR quality for non-native speakers.

First of the future steps in our research, would be to conduct more experiments, i.e. the evaluation of the autoencoder described in methodology description using more datasets of non-native speech.

One of the steps to realize in terms of future work, is approaching the problem of accent neutralization in a slightly different manner.

The other task we consider, is to focus on the most challenging cases, and build separate models for each of them. It could improve overall accuracy of non-native speech recognition.

We are planning to develop our idea for non-native speech recognition further and to constantly improve the quality of designed methodology. Furthermore, additional experiments will be conducted, i.e. using multiple nationalities of non-native English speakers, as well as using different datasets including samples of languages other than English.

## References

1. A. Graves, A. Mohamed, G.H.: Speech recognition with deep recurrent neural networks. Proc. ICASSP. IEEE (2013)
2. Abdel-Hamid, O., Mohamed, A.R., Jiang, H., Deng, L., Penn, G., Yu, D.: Convolutional neural networks for speech recognition. IEEE/ACM Trans. Audio, Speech and Lang. Proc. **22**(10), 1533–1545 (Oct 2014). <https://doi.org/10.1109/TASLP.2014.2339736>, <http://dx.doi.org/10.1109/TASLP.2014.2339736>
3. Amodei, D., Anubhai, R., Battenberg, E., Case, C., Casper, J., Catanzaro, B., Chen, J., Chrzanowski, M., Coates, A., Diamos, G., Elsen, E., Engel, J., Fan, L., Fougner, C., Han, T., Hannun, A., Jun, B., LeGresley, P., Lin, L., Narang, S., Ng, A., Ozair, S., Prenger, R., Raiman, J., Satheesh, S., Seetapun, D., Sengupta, S., Wang, Y., Wang, Z., Wang, C., Xiao, B., Yogatama, D., Zhan, J., Zhu, Z.: Deep speech 2: End-to-end speech recognition in english and mandarin (2015)
4. Dave, N.: Feature extraction methods lpc, plp and mfcc in speech recognition. International Journal for Advance Research in Engineering and Technology **1**, 1–5 (7 2013)
5. Dehak N., Kenny P. J., D.R., Dumouchel P., O.P.: Front-end factor analysis for speaker verification. Trans. Audio, Speech and Lang. Proc. **19**(4), 788–798 (May 2011). <https://doi.org/10.1109/TASL.2010.2064307>, <http://dx.doi.org/10.1109/TASL.2010.2064307>
6. Gatys, L.A., Ecker, A.S., Bethge, M.: A neural algorithm of artistic style (2015)
7. Grinstein, E., Duong, N., Ozerov, A., Prez, P.: Audio style transfer (2017). <https://doi.org/10.1109/ICASSP.2018.8461711>
8. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution (2016)
9. Li M., Han K. J., N.S.: Automatic speaker age and gender recognition using acoustic and prosodic level information fusion. Computer Speech Language (2013)
10. Livescu, K., Glass, J.: Lexical modeling of non-native speech for automatic speech recognition. IEEE International Conference on Acoustics, Speech and Signal Processing (2000)
11. Metallinou, A., Cheng, J.: Using deep neural networks to improve proficiency assessment for children english language learners. In: Fifteenth Annual Conference of the International Speech Communication Association (2014)
12. Radzikowski, K., Wang, L., Yoshie, O., Nowak, R.: Dual supervised learning for non-native speech recognition. EURASIP Journal on Audio, Speech and Music Processing **2019:3**, 1–10 (2019), doi:10.1186/s13636-018-0146-4, <https://rdcu.be/bgUxy>
13. Radzikowski Kacper, Wang Le, Y.O.: Non-native speech recognition using characteristic speech features, with respect to nationality. Proceedings of the conference of institute of electrical engineers of japan, electronics and information systems division (2017)

14. Radzikowski Kacper, Wang Le, Yoshie Osamu: Non-native english speaker's speech correction, based on domain focused document. In: Proceedings of the Conference of Institute of Electrical Engineers of Japan, Electronics and Information Systems Division (2016)
15. Radzikowski Kacper, Wang Le, Yoshie Osamu: Non-native english speakers' speech correction, based on domain focused document. In: Proceedings of the 18th International Conference on Information Integration and Web-based Applications and Services. pp. 276–281. iiWAS '16, ACM, New York, NY, USA (2016). <https://doi.org/10.1145/3011141.3011169>
16. T. Drugman, T.D.: Glottal closure and opening instant detection from speech signals. Proc. Interspeech (2009)
17. Tan, T., Besacier, L.: Acoustic model interpolation for non-native speech recognition. Proc. ICASSP (2007)
18. Tomokiyo, L.M.: Recognizing Non-native Speech: Characterizing and Adapting to Non-native Usage in LVCSR. Ph.D. thesis, Carnegie Mellon University (2001)
19. Verma, P., Smith, J.O.: Neural style transfer for audio spectrograms (2018)
20. Xiong, W., Wu, L., Alleva, F., Droppo, J., Huang, X., Stolcke, A.: The microsoft 2017 conversational speech recognition system (2017)

# **Sub-word units in Polish text generation.**

Emilia Zawadzka-Gosk<sup>1[0000-0002-0166-9254]</sup> Krzysztof Wolk<sup>1[0000-0001-5030-334X]</sup>

<sup>1</sup> Polish-Japanese Academy of Information Technology, Warsaw MZ 02-008, Poland  
ezawadzka@pja.edu.pl, kwolk@pja.edu.pl

**Abstract.** In this paper we introduce the research on natural language generation for Polish. This domain have various applications in different fields as art, literature, medicine or information technology. In presented solution the recurrent neural network architecture was adhibited. The network consists of three main layers: encoder, decoder and Gated Recurrent Unit (GRU) layer. The GRU layer is the composed one, and consists of several layers. To contrive high probability results we applied beam search algorithm. Important aspect of our research is an approach to text units provided to the network during training. The sub-word unit was chosen as the solution ensuring balance between results' quality and efficiency, although more advanced options are also described in the paper and considered in future work. The experiments were conducted with the use of literary texts in Polish coming from different eras and written in different styles. Their outcome demonstrate that automatic text generation by machine learning methods gives promising results.

**Keywords:** text generation, written art, subword units, machine learning , neural text generation.

## **1      Introduction**

Neural networks (NN) are successfully used in natural language processing tasks. Automatic text generation is the one that becomes a popular research domain in recent years, as it might positively affect various fields, as art, medicine, IT, science. NN can create poetry, but also could be used to generate specific, but repeatable written work, as medical records [5]. The deficiency of data, due to the lack of digitized documents or privacy issues, is a great problem of contemporary scientific research, which could be supported by natural language generation solutions. Also in the domains like software testing the availability of specific data is important. All of these needs could be met by the automatic text generation.

## **2      Sub-word units**

The most common application of subword units is the so-called open dictionary idea. The idea is to divide some or all of the words in the corpus into smaller units than words. These can be e.g. syllables, stemming with cut-off prefixes and suffixes, etc. Currently, a very popular approach is to use BPE (Byte Pair Encoding) algorithm.

It allows to accomplish this task by encoding rare and unknown words as a sequence of units smaller than words. This method is based on the fact that different word classes can be segregated, which allows for a sort of data compression. An example of such a division:

Kuzynka nazwała synka Xavier => Kuzynka nazwała synka Xia@@@ vi@@@ er

In the examples, @@ symbols have been introduced, which allow to "remember" the places where the units join into full words in order to possibly connect and process them. The disadvantage is that the approach cannot be applied to languages that do not use spaces to separate words such as Chinese. Nor can we encode common multi-word units such as "European Union" or "black hole". [4]

## 2.1 Subword units for Polish

Our approach is to divide texts into syllables or into core, prefix and suffix, and to indicate how they were separated. For the Polish language, the tool for such divisions has been implemented within the Clarin<sup>1</sup> project and is available as an online service<sup>2</sup>. [9] It allows text to be divided either into syllables:

mo++ --ja ku++ --zyn++ --ka na++ --zwa++ --ła sy++ --nka Xavier

or into a core with suffixes and prefixes.

moj++ --a kuzyn++ --ka na++ --zw++ --ała syn++ --ka Xavier

Note that in this type of approach there are additional ++ and -- tags that, like @@ in BPE, allow us to remember how the divisions were made, and how these units merge into full word forms.

Note, however, that this approach is language-specific and limited by different rules. The problem will be foreign words not included in dictionaries and rules, such as our example Xavier. This tool is capable of algorithmically dividing foreign words, not based on rigid rules:

mo++ --ja ku++ --zyn++ --ka na++ --zwa++ --ła sy++ --nka Xia++ --vier

But there is nothing to prevent different solutions from being combined one after the other as a pipeline. For example, we could use BPE first and then this tool. [7]

## 3 Generation methodology

Presented solution is a three layer Recurrent Neural Network (RNN) with one decoder layer, one Gated Recurrent Unit (GRU) [1] layer and one encoder layer. GRU layer

---

<sup>1</sup> <http://clarin-pl.eu>

<sup>2</sup> <http://ld.clarin-pl.eu:5000>

could consist of several layers. GRU is an improved recurrent network unit. Thanks to its architecture, composed of update gate and reset gate it is able to track long dependencies in sequences. Moreover, we decided to use beam search algorithm [3]. It keeps only predetermined number of best partial solutions, which are used to calculate final sub-word unit candidate. There are also other RNN solutions worth to consider. To accelerate the solution the GRU layer is replaced by Simple Recurrent Unit (SRU) layer [6]. SRU unit is a simplified version of GRU or LSTM units and has two main components, a light recurrence and highway network, with additional optimization of computations. Moreover, to improve the results of text generation, more than one language model could be used. Training the models on different corpuses enables to find most satisfying statistical property of the solution.

#### 4 Results and discussion

In our experiment we used two texts in Polish. The first one was Polish epic poem ‘Sir Thaddeus’ written by Adam Mickiewicz. The network was trained with the following parameters: 500 hidden units, 3 GRU layers, 15 epochs. The experiment was conducted in Google Colaboratory environment and took 52 minutes. The network generated stylized poem:

‘Litwo! Ojczyzno moja! Zdrowie, Ty, widzianego chłodzie!  
 Powstał, na jego, na jutro sto zamku owoje,  
 To Dziś odszedł, i Sopmu na Piotrkona strzewem,  
 Czy nie chciał, roży, i s pamienne życie żegły,  
 Tam nigdy już odskocha, już na pomiele.  
 Tak, to gości udeporząsza, na pierś poraga  
 Zabawą na włami zasłoła lecz cieka gadać,  
 Gdy patrzył za konia, a prawie gospodają

The results of an experiment were promising. The resultant text follows style of the source poem. Metrum line is close to Polish alexandrine, but not all the verses have 13 syllables. Vocabulary is similar to Mickiewicz’s. Although the text still has no coherent meaning.

The second experiment was conducted using polish translation of ‘Harry Potter and the Deathly Hallows’ novel written by J. K. Rowling. The network was trained with the following parameters: 256 hidden units, 2 GRU layers, 20 epochs. Training took 9 hours 24 minutes. RNN generated following text:

‘Wszystko okej, wciąż uderzyłem, ale  
 zobaczy, że się przebrali...  
 - Wiadomo już nie uśmieają - odrzekł Harry, ale czolo przeszył mu ból tak straszny,  
 że blizna zapiekła mu. - Wygląda tak, jakby czuje się, że  
 sprawiała. - Taak... Cedrik...  
 - Oni są na wszelki wypadek, Sami- Wiecie- Kto bardzo krwi, siedząc.

- Jesteś też? - odparł, trochę zaskoczony, że nie zdają sobą...  
 - Ale w tym momencie porzuciłeś'

Generated text is a dialog, the style of a book was preserved. The names of the book's main characters appears in presented fragment. The vocabulary seems to be similar to Rowling's.

## 5 Conclusions and future plans

Results of text generation by recurrent neural networks are promising. The resultant texts still suffer from some issues like incorrect words or sentences, and not consistent meaning of the whole text. Therefore in our future plans we intend to introduce new solutions and ideas. The environment used for the presented experiment imposes restrictions, the training cannot last longer than 12 hours, so in the first step we would like to change it. The experiments will be conducted with different network configurations, as SRU and adoption of more than one language model, on various texts styles. Finally we plan to compare popular language models as BERT[2] or GPT-2[8] with our solution and verify their effectiveness for Polish language.

## References

1. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN en-coder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.
2. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
3. Freitag, M., & Al-Onaizan, Y. (2017). Beam search strategies for neural machine translation. arXiv preprint arXiv:1702.01806.
4. Kudo, T. (2018). Subword regularization: Improving neural network translation models with multiple subword candidates. arXiv preprint arXiv:1804.10959.
5. Lee, S. H. (2018). Natural language generation for electronic health records. NPJ digital medicine, 1(1), 63.
6. Lei, T., Zhang, Y., Wang, S. I., Dai, H., & Artzi, Y. (2018). Simple recurrent units for highly parallelizable recurrence. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (pp. 4470-4481).
7. Liu, X., Hieronymus, J. L., Gales, M. J., & Woodland, P. C. (2013). Syllable lan-gauge models for Mandarin speech recognition: Exploiting character language mod-els. The Journal of the Acoustical Society of America, 133(1), 519-528.
8. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI Blog, 1(8).
9. Wolk, K., & Marasek, K. (2018, September). Survey on neural machine transla-tion into polish. In International Conference on Multimedia and Network Infor-mation System (pp. 260-272). Springer, Cham.

# Comparison of topic modelling algorithms in text clustering problem

Tomasz Walkowiak<sup>1</sup>[0000–0002–7749–4251] and Mateusz Gniewkowski<sup>1</sup>[0000–0002–0620–8123]

University of Science and Technology,  
Faculty of Electronics,  
Wybrzeze Wyspianskiego 27, Wroclaw 50-370, Poland

**Abstract.** The paper shows the quality of document clustering in a topic space obtained by the *LDA* (MALLET) and *ARTM* (bigARTM) algorithms. Following clustering algorithms were used: *spectral clustering*, *agglomerative hierarchical clustering* and *kmeans* with different distance measures: Euclidean, cosine and correlation. For evaluation purpose we used *Adjceusted Mutual Information (AMI)* score.

**Keywords:** topic modelling, distance, clustering evaluation, text analysis, LDA, ARTM

## 1 Introduction

Topic modelling is a complex and powerful text-mining method of generating sets of words (topics) that characterise the content of documents. Its aim is to extract collections of words which are supposed to be significant in a given corpus of documents and relatively coherent. Each document typically is a mixture of multiple topics in different proportions.

Topic modelling can be thought of as a method of reducing the dimensionality of BoW representation of documents and therefore, it can be used as a preparation part of proper dimensionality reduction method or clusterisation. The main problem we focus on in this paper is about the second one. The reason to group documents in topic space is to facilitate interpretation topic modelling results, specially that topics usually contain similar set of words. We have already dealt with similar problems in [4]. This article shows our attempts to improve the results by using the alternative topic modelling algorithm, ARTM [3]. Results of this research are part of the development of the tools included in the CLARIN-PL infrastructure.

The paper is structured as follows. In Section 2 briefly discuss tools and methods that were used in our work. Next, in Section 3 we describe data sets we used for our tests, results and conclusion.

## 2 Methods

Our goal is to properly group documents, described in a topic space, into meaningful clusters. To do this, we must first use topic modelling algorithm to obtain

topic representations, then cluster the data and finally evaluate the results. In this section, we briefly describe each of these steps.

## 2.1 Topic Modelling

In this work, we focus on two approaches for topic modelling. First of them is Latent Dirichlet Allocation (LDA) [1] which results in two matrices (topics over words and documents over topics) generated from prior Dirichlet distributions. The main disadvantage of LDA is the memory usage and computation time. The second method, called Additive Regularization of Topic Models (ARTM) [3], aims to eliminate these flaws by increasing the sparsity of the output matrices. What is more, zero probabilities in the output matrices can have a positive effect on clustering results. For our experiments, we used MALLET implementation of LDA algorithm and bigARTM implementation of ARTM.

## 2.2 Metrics and Clustering

In order to group documents we used following clustering algorithms:

1. *K-means* algorithm is a classic method that assigns labels to the data, basing on a distance to the nearest centroid. Centroids are moved iteratively until all clusters stabilise.
2. *Agglomerative hierarchical clustering (AC)* is a method that iteratively joins subgroups basing on a *linkage criterion*. In this paper, we present result for the average linkage clustering.
3. *Spectral clustering (SC)* is based on the Laplacian matrix of the similarity graph and its eigenvectors. The least significant eigenvectors create new, lower dimensional space that is used with a *K-means* algorithm.

One of the most crucial part of using clustering algorithm is to choose the proper distance function. The task of the distance function is to determine which samples are similar and which differ from each other. The most intuitive measure is Euclidean distance, but in natural language processing most commonly used is a cosine distance as i.e. it does not distinguish documents, described as a vector of most frequently occurred words in the corpus, that have a linear dependence between features. We also used a variation of the cosine distance, called correlation distance. Thanks to measuring the angle between mean vectors, the problem is reduced by one dimension. All of the above functions were used in experiments. However, K-means works only with Euclidean (in general any  $L_p$  metric) distance since it requires a method to calculate centroids.

## 2.3 Quality Metrics

Evaluation of clustering quality may be performed in two different ways: with external knowledge of sample membership or without it. Because we have original document labels, we can the first group of methods. Two most common metrics

are *Adjusted Rand Index* (ARI) and *Adjusted Mutual Information* (AMI). AMI is better suited to our problem as documents types are often unbalanced. It gives more weight to a clustering solutions with purer small groups than to minor mistakes in bigger ones. In our work we use AMI which was proposed in [2].

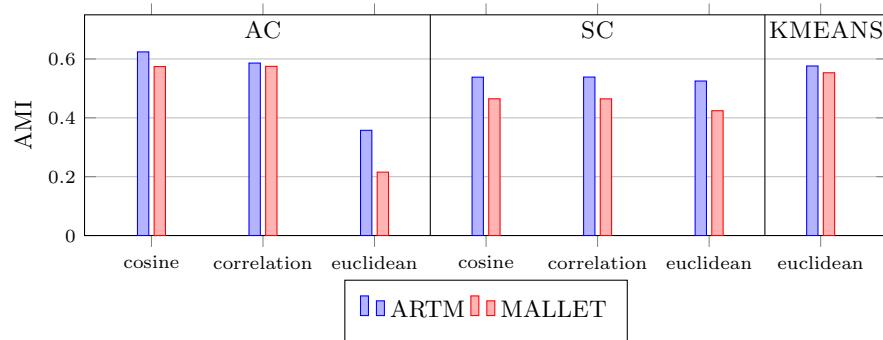
### 3 Experiments

#### 3.1 Data sets

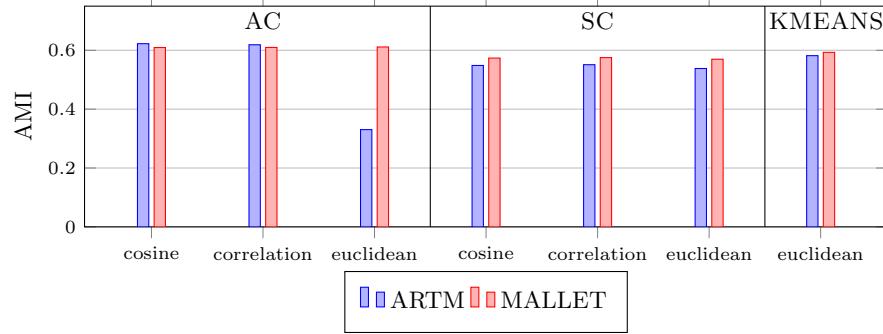
For our experiments we used two collections of text documents: *Wiki* and *Press*. The first corpus (*Wiki*) consists of 9,837 articles extracted from the Polish language Wikipedia. This corpus has also good quality, however some of the class assignments may be doubtful. It is characterised by a significant number of 34 subject categories. The second corpus (*Press*) consists of 6564 Polish press news. It is a good example of a complete, high quality and well defined data set. The texts were assigned by press agency to 5 subject categories. All the subject groups are very well separable from each other and each group contains reasonably large number of members.

#### 3.2 Results

The goal of our work is to find the best method for clustering documents in a topic space with an emphasis on generating that space. In order to fulfil this goal and find the answer, we performed our experiments as follows: given the documents  $d_i \in D$  over topics  $t_i \in T$  (the number of topics was set to 100) matrices as the result of the LDA (MALLET) and ARTM (bigARTM) for two data sets described in the previous section, we performed several tests that were evaluating the quality of clustering documents using different distance/similarity measures (like cosine distance, correlation distance, with different clustering algorithms (Agglomerative Clustering, Spectral Clustering, K-means). We have made the evaluation using AMI score and known classes  $c_i \in C$  for every document  $d_i$ . The number of clusters to find was twice as large as the number of original classes.



**Fig. 1.** *PRESS* corpus, AMI score for different clustering approaches



**Fig. 2.** WIKI corpus, AMI score for different clustering approaches

The results are given in Figure 2 and Figure 1. It can be observed that Agglomerative Clustering with cosine or correlation distance performs slightly better than the method we used as our base method, K-means. On the other hand, using Spectral Clustering reduces the quality of results. In the context of the tool used for topic modelling, ARTM gives better scores. For PRESS it is always true, but in Figure 2 it is noticeable that on average Mallet performs very similar to ARTM (excluding Agglomerative Clustering with Euclidean distance). Nevertheless, ARTM achieved the best result. The ARTM advantage is most likely caused by the sparsity feature. Because of that, the input of the clustering algorithm is less *noisy* and therefore it is easier for distance functions to distinguish samples.

As a conclusion, we would like to suggest using Agglomerative Clustering with cosine distance as a solution to the given problem and recommend ARTM as an alternative to commonly used LDA algorithm. ARTM is constantly being improved by its creators and gives better and better results.

The work was funded by the Polish Ministry of Science and Higher Education within CLARIN-PL Research Infrastructure.

## References

1. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *Journal of machine Learning research* **3**(Jan), 993–1022 (2003)
2. Hubert, L., Arabie, P.: Comparing partitions. *Journal of Classification* **2**(1), 193–218 (Dec 1985). <https://doi.org/10.1007/BF01908075>, <https://doi.org/10.1007/BF01908075>
3. Vorontsov, K., Potapenko, A.: Additive regularization of topic models. *Machine Learning* **101**(1), 303–323 (Oct 2015). <https://doi.org/10.1007/s10994-014-5476-6>, <https://doi.org/10.1007/s10994-014-5476-6>
4. Walkowiak, T., Gniewkowski, M.: Distance measures for clustering of documents in a topic space. In: International Conference on Dependability and Computer Systems. pp. 544–552. Springer (2019)

# Does the syntax matter for composing semantic representation of medical products therapeutic indications? \*

Wojciech Jaworski<sup>[0000-0002-7838-7781]</sup>

<sup>1</sup> Institute of Informatics, University of Warsaw

<sup>2</sup> Lekseek Polska

**Streszczenie** In this paper, we present the task of creating meaning representation for therapeutic indications for medicines. We analyze the extent to which syntax of documents is useful for extraction their semantics.

**Keywords:** Drug Characteristics Processing · Semantics

## 1 Drug indications

Therapeutic indications for medicines are described in documents named *Summary of Product Characteristics*. They define the relation between drug, active substance, patient, illness and therapy. In the course of our R&D project we analyze Polish versions of these documents called *Charakterystyka Produktu Leczniczego*. One of the goals of the project is to formalize the above mentioned relation.

Grounding provides us the following structure: there must be a patient in order for illness to exist. Treatment on the other hand requires both a patient and an illness (usually also a drug).

However the question appears, whether and to what extent syntax of sentences helps in acquiring their semantic representation. Consider the following examples<sup>3</sup>:

[*The ready-to-use solution*]Drug[*is indicated*]Verb[*for continuous hemofiltration*]Treatment  
[*in patients*]Person[*with acute renal failure of various origins*]Illness  
[*who are in intensive care units*]Condition.

\* This work is supported by project POIR.01.01.01-00-0328/17 financed by European Regional Development Fund as a part of The Intelligent Development Program 2014-2020

<sup>3</sup> We present examples in English to make them easier to understand for readers who are not familiar with Polish, and because the phenomena presented are mainly language independent

[*Velfofent*]Drug[is indicated]Verb[for the treatment]Treatment  
[of breakthrough pain (BTP)]Illness[in adults]Person[with cancer]Illness  
[who are already receiving maintenance opioid therapy for chronic cancer pain]Condition.

Our goal is to extract 5-tuple composed of drug, illness, patient, treatment and a list of conditions. We can observe that the examples may be split into phrases that represent semantic units of our interest. However, syntactic relations between these phrases may significantly differ, also they are ambiguous (in both sentences Person may be subordinate of Treatment or the Verb). Most of these relations is not relevant for obtaining semantic representation, but they are crucial for recognizing that in the second sentence *breakthrough pain* is an illness treated by the drug and *cancer* is in fact a condition.

## 2 Coordination

Let us consider a sentence:

*Each film-coated tablet contains abacavir hydrochloride equivalent to 600 mg abacavir and 300 mg lamivudine.*

The proper assignment of phrase boundaries in this sentence is crucial for constructing its semantic representation, yet the sentence is syntactically ambiguous and may be disambiguated in the following manners:

[*abacavir hydrochloride*] equivalent to [*600 mg abacavir and 300 mg lamivudine*]  
[*abacavir hydrochloride* equivalent to *600 mg abacavir*] and [*300 mg lamivudine*]  
*abacavir [hydrochloride equivalent to [600 mg abacavir and 300 mg lamivudine]].*

The first interpretation is proper for the typical use of coordination, however in this case the second interpretation is the correct one.

Moreover, the sentence below shows us that sometimes phrase bracketing for drug composition description behaves differently:

*1 ml ready-to-use solution contains [40 mg erythromycin and 12 mg zinc acetate] in the form of erythromycin-zinc complex.*

## 3 Non-sentential utterances

Apart from regular sentences various abbreviated notation is present in the domain of drug administration, for example:

*The usually recommended doses are: infants and children: 30 mg/kg bw./day,  
adolescents and adults: 20-30 mg/kg bw./day.*

In this case we need to create the semantic representation out of each of divided by colon phrases and compose it into the semantics of utterance without clues provided by syntactic relations.

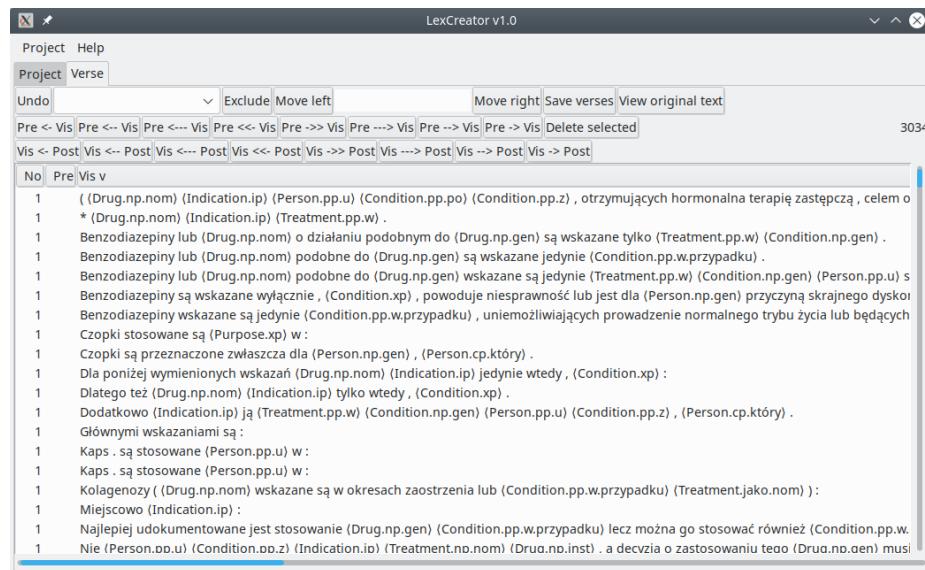
## 4 Our solution

First, we divide text into 'segments' according to the following rule: colons, semicolons and sentence ending dots are borders of segments. We make here exclusions for inclusions in parentheses which are treated as single segments and special notations such as proportions.

Then, we split segments into phrases and we assign phrases with their semantic types. Here we do not have obvious phrase-ending markers. We define phrases according to their semantics — we extract text fragments that describe persons, substances doses, etc.

We obtain meaning representation of each phrase independently. Composition of segment meanings out of phrase meanings may be performed by means of simple grammar which ignores most syntactic and semantic features. In first example phrases that have different semantic types compose different fields in resulting tuple. This may be done using grammar that collects sequence of phrases ignoring all their features. In case of second example the only work for grammar is to distinguish the role of first and second Illness phrase. For the third example we need only simple rules that make lists out of semantics of coordinated phrases.

When we build text semantics upon segment meanings we pay attention to punctuation. Semicolon and dot connect list elements while semicolon introduces a list header — information that should be applied to every element of subsequent list.



Rysunek 1. LexCreator

## 5 Phrase extraction

Typical phrase is a preposition phrase, noun phrase or relative sentential clause. Phrase borders and types may be recognized using machine learning models. However these models require training corpora. Since phrase division is a substantial decision in the process of meaning extraction training corpus must be representative for the domain

It is also possible perform phrase splitting manually on whole corpus and maintain a team on annotators for preparing new texts.

In order to create phrase division we developed a special tool: **LexCreator**.

LexCreator is a publicly available tool created for extracting phrases from text. It presents each text (segment) as a row in a three column table. User may select rows and move tokens belonging to them between columns as well as save contents of selected rows from middle column to a lexicon. Rows may be also sorted according to the beginning or end of each column.

LexCreator (see Fig. 1) is capable of replacing known phrases with their types. This makes sentence fragments similar and helps in extracting next phrases. One requirement in this process is that a phrase must be disambiguous, i. e. for a given phrase there must be only one semantic type assigned to it.

## 6 Conclusions

Now, we analyze the advantages that segments and phrases division provides for the process of creating semantic representation.

First, we may parse each phrase alone using deep syntactic parser and obtain semantic representation of its contents. Since phrases are shorter and syntactically simpler than sentences problems with ambiguity and computational complexity are significantly reduced.

Second, semantics is more 'flat' than syntactic derivation tree. That is why not all syntactically defined constituents needs to belong to phrase. This solves the problem syntactic constituents which are not semantically related to the semantics of the rest of the phrase.

Moreover, for many phrases we do not need precise semantic representation, it is enough to have semantic type of phrase and raw text of its contents. This apply for example to various conditions of drug indications.

The other advantage is that the failure in parsing a single phrase does not imply the failure of creating the representation for segment and text.

Now, we may conclude, answering the question posed in the title. Syntax is important for processing short unit of texts (i.e. phrases). Yet, the process of their composition in longer units (sentences etc.) depends mainly on chosen semantic representation.

# Integrating Polish Language Tools and Resources in spaCy

Ryszard Tuora<sup>[0000–0001–9610–8048]</sup> and Łukasz Kobyliński<sup>[0000–0003–2462–0020]</sup>

Institute of Computer Science, Polish Academy of Sciences  
Jana Kazimierza 5, 01-248 Warszawa, Poland  
[ryszardtuora@gmail.com](mailto:ryszardtuora@gmail.com), [lkobylinski@gmail.com](mailto:lkobylinski@gmail.com)

**Abstract.** This paper summarizes the ongoing work aiming at integrating existing Polish language tools and resources into the spaCy pipeline. spaCy is an easy to use Python framework, commonly used by the Natural Language Processing community. While many tools for processing Polish already exist, they require specific installation environments. By combining the ease of use of spaCy with existing NLP resources for Polish we are hoping to promote the idea of using NLP solutions in Polish.

**Keywords:** Natural Language Processing · spaCy · Polish language

## 1 Introduction

Natural Language Processing of Polish has a long history of research and a large number of tools and resources have been produced to date [1]. However, many of these tools and resources are not easily used by researchers not already involved in Polish NLP, and also not easily used in commercial, production environments. There is also no common pipeline for processing text that would use all of the state of the art processing methods.

spaCy<sup>1</sup> is one of the more commonly used NLP frameworks, developed in Python, and offering a complete NLP pipeline for English and many other languages. spaCy is easy to use and promises high efficiency and accuracy of the implemented methods. Unfortunately, spaCy works best for English, while other languages have been implemented with different level of maturity. Polish is one of the languages with almost no out-of-the box support.

In this work we aim to integrate existing language resources for Polish into spaCy, in order to be able to use all of the state-of-the art processing methods in an easy to use framework. The currently released model is already available for download at the following address: <http://zil.ipipan.waw.pl/SpacyPL>.

## 2 Previous Work

Some work concerning implementing Polish processing into spaCy has already been done<sup>2</sup>, but the strategy was very much different from what we propose in

---

<sup>1</sup> <https://spacy.io/>

<sup>2</sup> <http://spacypl.sigmoidal.io>

this contribution. Previous work on this subject involved creating new language resources and methods that have not been used or evaluated previously in a research environment. For example, the authors propose to create a lemmatizer that is based on the freely available ispell dictionary.

Our aim is to recreate a full processing pipeline, which incorporates already published, well-evaluated methods that are based on several years long research on Polish language and incorporate the linguistic aspect of the language in the results produced.

### 3 Issues in Implementing Polish Support in spaCy

#### 3.1 General issues concerning spaCy

spaCy is focused on ease of access and integration for production use. It consists of a standard library, which includes the general-purpose pipeline, and Language classes defined for 52 languages. These classes include very basic data for NLP in the respective language, and they can be supplanted with language-specific models which are downloaded separately. A full model for a given language for spaCy should include solutions for representing the vocabulary, tokenization, tagging, dependency parsing, named entity recognition, and some basic word classification capabilities. Models for each component can utilize the same vector representation. For our purposes we have found that using the KGR10 100-dimensional fasttext vectors [2]<sup>3</sup> yields the best results. We have pruned the vectors to 800.000 most frequent words, and created the model around them.

#### 3.2 Tagging

spaCy requires a tagger model, which classifies tokens into a language specific tagset, which is subsequently mapped onto the 17 UD part of speech (POS) tags. We have chosen to rely on the National Corpus of Polish [3] (NKJP) POS tagset, which includes 35 different tags for grammatical classes. We have trained our tagging model on a corpus consisting of the 1 million segment manually annotated subcorpus of the NKJP corpus<sup>4</sup> and the Frequency Corpus of the 1960s Polish language<sup>5</sup>. Achieving high accuracy on POS tagging (i.e. excluding morphological features such as grammatical gender) is fairly easy. Nevertheless mapping the NKJP tagset onto UD proves quite problematic as the two tagsets aim at different purposes, the latter including some syntactic distinctions. For example the NKJP tagset utilizes the 'FIN' tag for all finite verbs, whereas the UD tagset differentiates between auxiliary verbs and content verbs. The same form, e.g. "jest", can be assigned 'AUX', or 'VERB' depending on the context. Also the UD tagset includes a 'PROPN' tag, which is attached to all proper

---

<sup>3</sup> These embeddings are available for download here: <https://clarin-pl.eu/dspace/handle/11321/606>

<sup>4</sup> <http://clip.ipipan.waw.pl/NationalCorpusOfPolish>

<sup>5</sup> <http://clip.ipipan.waw.pl/PL196x>

names. This amounts to named entity recognition at the level of tagging. In the NKJP tagset, all proper names are assigned ‘SUBST’ tags, which correspond to the ‘NOUN’ tag in UD. Therefore even though using the UD tagset for Polish is possible, the UD tag of a given word is not a function of its NKJP tag in the proper sense of the term, as is required by spaCy. This acts as a glass ceiling for UD POS tagging results.

We want to try out several different approaches to this problem, but for now, we have decided to rely on the original POS tagset, with a simple mapping to UD. Because spaCy allows users to access both the language-specific, and UD tags of a given word, the user can use all the information provided by our model.

### 3.3 Lemmatization

For our lemmatizer we have chosen two parallel strategies. The first involves implementing a look-up table, using a lemma dictionary imported from the Morfeusz 2 morphological analyzer [4]. We use the functionalities for look-up tables introduced to spaCy in version 2.2. This solution allows to achieve competitive lemmatization speed, with high accuracy, compared to rule-based approaches.

The second strategy, is to integrate our model with Morfeusz as an external dependency. This requires some additional steps from the user during the installation process, but provides substantial improvements. Morfeusz is imported and used in our custom ”preprocessing” pipeline component which tokenizes, attaches tags, and provides lemmas for the input text. Because Morfeusz provides multiple possible interpretations, we need some method of disambiguation. At the moment we simply choose the first interpretation which agrees with the tags provided by our tagger, despite its simplicity, the improvements are still noticeable.

### 3.4 Dependency Parsing

For training a dependency parser, we’ve used the PDB UD treebank<sup>6</sup>. It is bigger, and linguistically more robust than LFG used by previous Polish attempts with spaCy. This latter factor entails that PDB is also harder to learn, which has been observed with other parsers, e.g. UDPipe obtaining 90.9% UAS on LFG, and only 87.6% on PDB when parsing raw text<sup>7</sup>. The results with respect to UAS and LAS measures are not on par with some state of the art dependency parsers yet. Some of this can be improved by more elaborate optimization, but some deficiencies are inherent in spaCy, as the parser component utilizes only the vector embeddings of words as input.

### 3.5 Named Entity Recognition (NER)

Our work in providing support for NER is still in the early stages. We have prepared some models trained on the NKJP coprus, but it has a limited set

---

<sup>6</sup> [https://universaldependencies.org/treebanks/pl\\_pdb/index.html](https://universaldependencies.org/treebanks/pl_pdb/index.html)

<sup>7</sup> <http://ufal.mff.cuni.cz/udpipe/models>

of 6 labels (persons, organizations, dates, time, geographic and anthropogenic locations). Extending this dataset with other corpora (e.g. the PWr corpus) is difficult because of the incompatibility between the sets of labels used for tagging. At the moment, we achieve an accuracy (measured as F-1 score value) of 87.52 on a 500-document test set extracted out of NKJP.

## 4 Experimental Results

Table 1 compares the results of our model, with those achieved by previous attempts at modelling Polish in spaCy. Additionally we list the results for an initial version of a different model we work on, which includes the Morfeusz library for tokenizing, tagging and lemmatization. The results have been obtained using the CONLLU 2018 Evaluation script<sup>8</sup>.

The first row lists the results on the PDB test subset, the second lists the results for LFG test subset. Our models were trained on PDB, whereas the one by Sigmoidal was trained on LFG. Each column represents accuracy results on the following tasks: 1. tokenization, 2. internal tagset part of speech tagging, 3. UD tagset part of speech tagging, 4. lemmatization, 5. Unlabelled and 6. Labelled Attachment Scores for dependency parsing. All values are calculated as F1 scores.

**Table 1.** spaCy model evaluation results.

|     |            | Tokens | XPOS  | UPOS  | Lemmas | UAS   | LAS   |
|-----|------------|--------|-------|-------|--------|-------|-------|
| PDB | IPI PAN    | 98.8%  | 90.9% | 83.9% | 91.0%  | 86.1% | 83.2% |
|     | +MORFEUZZ2 | 98.3%  | 92.7% | 85.9% | 93.7%  | 90.7% | 87.8% |
|     | SIGMOIDAL  | 98.8%  | 90.4% | 83.6% | 72.7%  | 75.1% | 68.0% |
| LFG | IPI PAN    | 96.7%  | 96.0% | 83.2% | 90.7%  | 84.3% | 79.3% |
|     | +MORFEUZZ2 | 99.5%  | 98.9% | 86.1% | 94.6%  | 90.6% | 85.4% |
|     | SIGMOIDAL  | 96.7%  | 95.4% | 82.8% | 73.9%  | 85.9% | 83.2% |

## 5 Conclusions and Future Work

The currently released version of Polish language models and resources for spaCy allow for tokenization, lemmatization, POS tagging, dependency parsing, and basic NER of Polish using the standard spaCy pipeline. The evaluation results presented in the previous section show that our implementation of Polish language support allows for producing results of higher accuracy than the previously proposed approach.

For future work, we are planning to further optimize our current models, and prepare full support for named entity recognition. Additionally, we aim to

<sup>8</sup> <http://universaldependencies.org/conll18/evaluation.html>

integrate the Morfeusz library with our spaCy models, in order to give users an easy way of accessing it's proven solutions for NLP in Polish. This step would allow to easily achieve good tokenization, which is nontrivial, and important for improving further processing.

## References

1. Dziubalska-Kolaczyk, K., Ogrodniczuk, M. (eds.): Current state of the art in language technology for Polish, special issue of Poznan Studies in Contemporary Linguistics, vol. 55(2). De Gruyter (2019)
2. Kocon, J., Gawor, M.: Evaluating KGR10 Polish word embeddings in the recognition of temporal expressions using BiLSTM-CRF. *Schedae Informaticae* **27** (2018). <https://doi.org/10.4467/20838476SI.18.008.10413>, <http://www.ejournals.eu/Schedae-Informaticae/2018/Volume-27/art/13931/>
3. Przepiorkowski, A., Bańko, M., Górska, R.L., Lewandowska-Tomaszczyk, B. (eds.): Narodowy Korpus Języka Polskiego. Wydawnictwo Naukowe PWN, Warsaw (2012)
4. Woliński, M.: Morfeusz reloaded. In: Calzolari, N., Choukri, K., Declerck, T., Loftsson, H., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., Piperidis, S. (eds.) Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014. pp. 1106–1111. ELRA, Reykjavík, Iceland (2014), <http://www.lrec-conf.org/proceedings/lrec2014/index.html>

# Bias-variance tradeoff problem in dialogue agents: a nearly infinite size state space model postulate

Daniel Oklesinski<sup>1,2</sup>

<sup>1</sup> Institute of Informatics, Warsaw University, Banacha 2, 02-097 Warsaw, Poland

<sup>2</sup> Selidor, sp. j., omiaska 20b, 01-685 Warsaw, Poland

**Abstract.** In this paper, I propose a solution to the bias-variance tradeoff problem in the area of dialogue managers. While the current paradigm in dialogue system is to rely on machine learning solutions, I aim to show that joining a nearly infinite size state space model with an extended expert system can result in an efficient, commercially attractive task-oriented dialogue agent that is able to perform elaborate actions. I present bot that is able to book services within existing booking system Reservis by conversing with a user in the Polish language as a proof of concept. It aims to make human-computer interaction as natural as possible and not to limit user's choice by making inexplicit assumptions (and therefore showing only part of available results). The complex design of state space and slots allows the system to accurately react to many corner cases.

**Keywords:** Bias-variance tradeoff · dialogue manager · state space

## 1 Introduction

Since the frame-driven GUS system for travel planning [3], most dialogue agents are based on slot filling and frame design. The next step in the progress of the field was the rise in the widespread use of machine learning [9]. Strict structure of dialogue introduces a high bias, while dependence on training set results in a high variance. It was not until the recent development of the second branch of conversational agents, task-oriented dialogue managers, that dialogue systems have become ubiquitous (eg. Amazon Alexa and Apple's Siri). Most of them are personal assistants. They are able to handle a variety of tasks, but are generally unable to process many turns of dialogue [5]. Multi-turns conversational agents have started appearing only recently. Areas they are developed in include e-commerce [2] and movie booking [7].

The presented solution goes against the ML trend and focuses on extending the state space as to ensure that many types of situations would be supported in the desired way [8]. Because of that, an attempt to create an expert system based on a complex and exhaustive state space has been made. The result allows for creating very specific responses based on current situation. By using nearly infinite state space, the system has mixed initiative. It is able to independently perform a full process of booking through the messenger application [1].

Although the use of machine learning has been often proven to be useful [10, 6], it comes with its own limitations. Because the system needs to handle many different specific situations, we did not want to limit state definition and, even with ML solution, would still have to perform a substantial design work requiring an expert knowledge.

## 2 Limitations of machine learning

Machine learning solutions require a small state definition and by such, a state often holds a simplified register of a conversation. To limit retained data, a structure of information is omitted and, most often, even not recognized by NLU module (that performs only shallow analysis). Because of these, such solutions have a high variance and are not precise. The more specific scenario, the less chance of the satisfying output.

The need to recognize user needs requires using the NLU module with deep syntactic and semantic analysis. Designing simple state structure that holds all information from NLU module and is appropriate for automatic learning purposes is nearly impossible. Furthermore, any change in bot's desired set of features would result in necessity to both redesign state structure and prepare a new set of learning data.

## 3 System overview

Reservis is a universal booking system for businesses. It allows any business based on client service (eg. beauty parlours) to manage its own bookings. It is highly adaptable and configurable and allows business managers to control the process of booking. The role of the dialogue manager is to enable customers to make a reservation in any business subscribed to Reservis. Therefore, the primary requirement is to ensure that an app will be able to handle a variety of categories and any service configuration settings.

A state of conversation is defined by its current knowledge, which consists mainly of merged pieces of information extracted from the clients utterances and history of the dialogue managers actions. Every turn, the clients utterance is parsed, disambiguated and merged into already acquired knowledge.

Raw query text made by a client is fed to the NLU module which returns structural representation of content. No context is given to the module and the assumption is that the module returns every possible correct interpretation of a given utterance. NLU module is a separate system ENIAM that performs deep syntactic and semantic analysis and implements fuzzy pipeline (no heuristic desambiguation), which guarantees the correctness of acquired interpretations [4]. After disambiguating JSON data and updating inner knowledge, the system acquires data concerning availability of terms from Reservis database and chooses a reply according to the current state and the set of rules.

## 4 Implementation

The corpus with utterances regarding booking was created. Participants of the research were asked to make a reservation via bot in any field from a predefined set. Answers of the bot were mocked by a researcher. 89 participants took part in the research. Based on the created corpus, the following slots were created: action, service, time, location, organisation, doer, patient, price and rating.

A unique structure for every slot aims to exhaustively describe given parameter. For example, utterances about time like we wtorki (on Thursdays), w godzinach porannych (in the morning hours), and w drugim tygodniu listopada

(in the second week of november) can be found in the corpus. All slots' structures also support negation, conjunction, alternative, preferences, points of freedom, referring to dialogue history and ambiguity.

#### 4.1 Communication with NLU module

JSON data extracted from client's utterance by NLU module is formatted according to predefined information schema. The core of schema consists of 9 attributes (corresponding to slots). Every attribute is additionally parametrized according to a design for a given slot (eg. location can include amongst other town, quarter, street parameters). The schema also allows modifiers of these core attributes (such as before or aprox) and operations on them (such as conjunction or alternative).

After extracting, the JSON data is disambiguated and merged into current representation of knowledge. Although the space of responses from the NLU module is finite, the size of combinations makes effectively the number of possible states infinite. With such a design, nearly every possible state of conversation can be encoded in it. With the assumption of perfect NLU module, the only limitation is information schema. If some information from the client cannot be put in the schema, it cannot be merged into the dialogue state.

#### 4.2 Expert system

Every slot has a definition of being adequately filled. Eg, time is considered filled if user mentions anything about it as opposed to service that must point to a specific service in the Reservis database with a high probability. The system, after analyzing current dialogue knowledge and data from Reservis database, decides what action to take. Then, a reply is generated. Replies can be either open questions, specifying questions (accompanied by a finite set of suggested responses) or acknowledgements. The system infers both client's and business' demands and constraints from possessed data.

One of the main factors in decision making is the current aim of conversation (eg. to determine what service the client seeks). However, the client can in any moment change a topic, revert the previous statement or retract to the previous point in the conversation. Thus, the dialogue, like in natural conversation, has mixed initiative.

The system seeks not to make decisions for the client. If necessary, it enquires for additional data, so it can present the best results. It also takes into account which medium of communication is used, as different media require different dialogue flow.

The multitude of known data allows to handle accurately any corner case and generate very specific and tailored to circumstances responses. For example, if the client defined their service and time preferences, the system may present available times (if there are only a few results), ask for another parameter/preferable hour/preferable day of the week, inform that the value of the time parameter is invalid (eg. 30th February).

Thanks to the specific structure for every slot, the dialogue manager is able to take into the account more than a traditional slot-filling dialogue manager. It allows to implement specific extending methods for a given slot. (U: I would like to book a hairdresser for tomorrow 8pm. DM: Unfortunately, there are no dates available. What about tomorrow 7pm?).

While adding new features requires a substantial implementation work, it does not require creating and tagging new dataset or new methodology. It is easy to add handling new configuration setting. With the assumption of a perfect NLU module and having a full control over the dialogue flow, we can safely give the initiative to a client.

## 5 Conclusion

Although machine learning based solutions can be, and often are, useful in creating dialogue managers, when it comes to creating a task-oriented dialogue agent that has to be able to perform complicated multi-turn actions, a variation of an expert system has more advantages.

By having a complex state space and being an elaborate expert system, it is able to handle very specifically many various points in a conversation. Having a mixed initiative, the flow of conversation is more natural than in most existing solutions. While extending bot does require work, it is easy to add precisely new features. Although the work may be substantial, it is not methodologically hard.

## Acknowledgments

This work was co-financed by the European Regional Development Fund under the Regional Operational Programme for Mazowieckie Voivodeship 2014-2020 project RPMA.01.02.00-14-5681/16-00.

## References

1. Reservis Chatbot. <http://m.me/ChatbotReservis>. (2019)
2. Amir Reza Asadi and Reza Hemadi. Design and implementation of a chatbot for e-commerce. (2018)
3. Daniel G Bobrow, Ronald M Kaplan, Martin Kay, Donald A Norman, Henry Thompson, and Terry Winograd. Gus, a frame-driven dialog system. *Artificial intelligence*, 8(2) (1977)
4. Wojciech Jaworski and Jakub Kozakoszczak. Eniam: Categorial syntactic-semantic parser for polish. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations, pages 243247 (1977)
5. Dan Jurafsky and James H. Martin. Speech and language processing (3rd ed. draft). <https://web.stanford.edu/~jurafsky/slp3/> (2019)
6. Olivier Lemon and Olivier Pietquin. Machine learning for spoken dialogue systems. In European Conference on Speech Communication and Technologies (Interspeech07), pages 26852688. (2007)
7. Xiujun Li, Zachary C. Lipton, Bhuwan Dhingra, Lihong Li, Jianfeng Gao, and Yun-Nung Chen. A user simulator for task-completion dialogues. CoRR, abs/1612.05688. (2016)
8. Daniel Oklesinski and Wojciech Jaworski. Implementation of current state-of-the-art technology for obtaining a working dialogue agent. In 9th Language Technology Conference. (2019)
9. Tim Paek and Roberto Pieraccini. Automating spoken dialogue management design using machine learning: An industry perspective. *Speech communication*, 50(8-9):716729. (2008)
10. Satinder Singh, Diane Litman, Michael Kearns, and Marilyn Walker. Optimizing dialogue management with reinforcement learning: Experiments (2002)

# Fine-Grained Named Entity Recognition for Polish using Deep Learning<sup>\*</sup>

Michał Marciniuk<sup>[0000–0002–3269–6378]</sup>

Department of Computational Intelligence  
Wrocław University of Science and Technology  
Wrocław, Poland  
`michal.marcinczuk@pwr.edu.pl`

**Abstract.** In the paper, we present a work in progress on fine-grained named entity recognition for Polish. The recent works on language modeling and deep learning had a significant impact on many natural language processing tasks, including named entity recognition. However, the focus was mainly on the evaluation of coarse-grained named entity models. In our work, we focus on fine-grained models for Polish. We evaluate a deep learning approach utilizing three different FastText language models. The results are compared with the previous state-of-the-art model based on Conditional Random Fields. We obtained a significant improvement from 63% to more than 73% of F-measure.

**Keywords:** Information Extraction · Named Entity Recognition · Deep Learning.

## 1 Introduction

The recent works on language modeling and deep learning had a significant impact on many natural language processing tasks. Language models are robust because they can be used to transform words into points in a multi-dimensional space and the distances between those points reflect the similarity between the words. Representation of words as real number vectors allows applying powerful methods such as deep learning to text processing. The PolEval 2018 task on named entity recognition has contributed to the improvement of NER for Polish [11]. The CRF-based system like Liner2 [7] and NERF [9] were outperformed by systems based on deep learning. The score improvement was from 81.0 for Liner2 and 73.9 for NERF to 86.6 for Parallel LSTM-CRFs [1] and 85.1 for PolDeepNer [7]. Recent improvements in the NER are reported solely for coarse-grained categories of named entities. For English the Conll-03 dataset [10] which is commonly used for NER evaluation consists of only four categories, i.e. *person*, *location*, *organization* and *miscellaneous* entities. For Polish, the evaluation dataset used in the shared task was based on the NKJP guidelines [8] which defines 12 categories of named entities. In the paper, we present the evaluation for fine-grained named entity recognition for Polish using the recent methods.

---

<sup>\*</sup> Work financed as part of the investment in the CLARIN-PL research infrastructure funded by the Polish Ministry of Science and Higher Education.

## 2 Fine-Grained Named Entity Recognition Baseline

*KPWr annotation guidelines* [6] defines a hierarchy of named entities containing more than 100 categories organized in a multi-level hierarchy. The top level consists of nine categories: *event*, *facility*, *living*, *location*, *organization*, *product*, *adjective*, *numerical* and *other*. The hierarchy was inspired by the Sekine's Extended Named Entity Hierarchy<sup>1</sup>. The guidelines were used to annotate the KPWr corpus [2]. Since not all categories were sufficiently numerous in the training subcorpus only 82 of them were selected for training.

The baseline was obtained using Liner2 [5]. Liner2 utilizes Conditional Random Field method and a rich set of features, including orthographic, structural, morphological, lexicon-based and wordnet-based. Before deep learning become popular, the CRF-based models were successfully applied to different sequence tagging tasks, including NER but also chunking and morphological tagging. In the case of fine-grained NER there was a problem with model learning efficiency. For 82 categories it took a couple days to train the model. This caused difficulties in adjusting the parameters as every single evaluation was very time-consuming.

## 3 Evaluation of Deep Learning and Language Models

We used the same neural network architecture as for the coarse-grained model [7]. The network consists of the following layers: *Word Embeddings*, *Dropout* with a rate of 0.5, *Bidirectional Gated Recurrent Unit*, *Dense* and *Conditional Random Fields*. For word embeddings we used three FastText language models [3]. `cc.pl.300` [3] and `kgr10/plain.skipgram.dim300.neg10` [4] were pretrained. `cc.pl.deduped.maca.skipgram.300.mc10` was trained from the scratch on Polish texts from Common Crawl<sup>2</sup>. The idea behind training the model from scratch was that the `cc.pl.300` was trained on text tokenized with a generic tokenizer which does not take into account the specific rules of tokenization for Polish. Thus, we tokenized the texts using the same rules as in the reference dataset and retrained the language model.

All three models significantly outperformed the baseline model by 5–9 pp of F-measure. The retrained model (2) was better than the `cc.pl.300` what confirmed that the language model trained on texts tokenized using language-specific rules was better than using the generic tokenizer. The (3) model obtained the best result. This might be caused by the fact that the model was trained on a corpus which contains high-quality texts (books, articles, blogs, etc.), while the Common Crawl corpus used for the other two language models contains solely texts crawled from the Internet and it may contain many low-quality texts (short snippets, web page templates, etc.). We also evaluated the ensemble of the three models and obtained improvement by another 2 pp up to 73.28% of F-measure. The ensemble was based on a majority voting and in case of three different votes, the decision from the model (3) was taken. Figure 1 contains detailed results

---

<sup>1</sup> <http://nlp.cs.nyu.edu/ene/>

<sup>2</sup> <https://commoncrawl.org/>

for every named entity category. As one can observe the infrequent categories obtained in general lower performance.

**Table 1.** Evaluation of different FastText language models for Polish

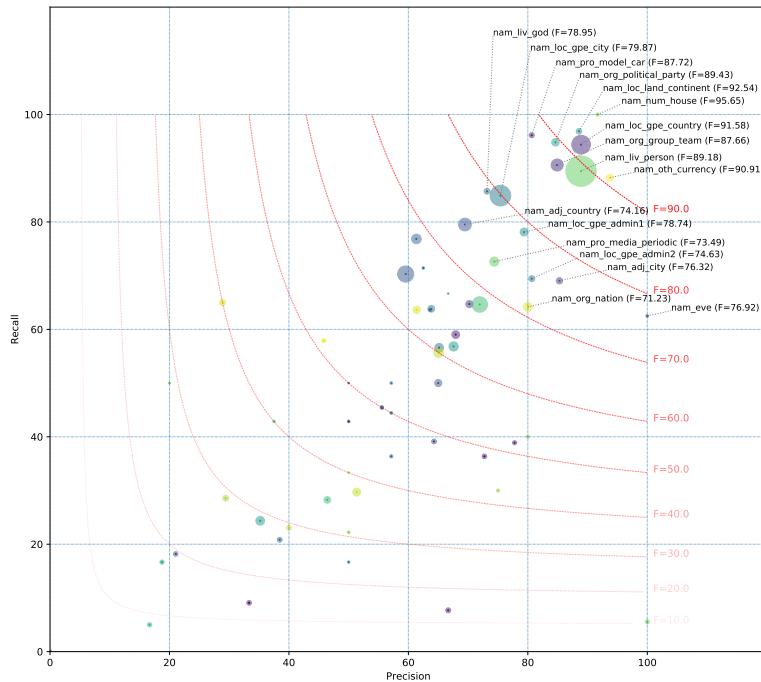
| No Language model                        | Precision | Recall | F-measure |
|--|-----------|--------|-----------|
| 0. Liner2 baseline [5]                   | 67.65     | 58.83  | 62.93     |
| 1. cc.pl.300 [3]                         | 71.40     | 66.43  | 68.83     |
| 2. cc.pl.deduped.maca.skipgram.300.mc10  | 72.74     | 68.89  | 70.76     |
| 3. kgr10.plain.skipgram.dim300.neg10 [4] | 71.36     | 71.49  | 71.43     |
| 4. Majority voting of 1, 2 and 3         | 74.70     | 71.92  | 73.28     |

## 4 Summary

The deep learning approach utilizing FastText language models significantly outperformed the recent CRF-based state-of-the-art models. The deep learning approach has many advantages over the CRF-based and some disadvantages. The most important benefit is the performance — we obtained up to 9 pp improvement of F-measure. The other advantage is much shorter training time — training a single neural network took less than 60 minutes, while training CRF model takes several days. The last benefit is shorter processing time — a single neural network processes the testing dataset in less than 1 minute, while the CRF-based model requires more than four minutes. We observed only two disadvantages: a much larger model size and higher memory requirement to run the model. The CRF-based model has a size of fewer than 0.5 GB and requires 3 GB of memory. The best single neural network requires a language model that has a size of near 11 GB (20 times larger) and requires 13 GB of memory (4 times more).

## References

1. Borchmann, L., Gretkowski, A., Graliński, F.: Approaching nested named entity recognition with parallel lstm-crf. In: Ogrodniczuk, M., Kobyliński, (eds.) Proceedings of the PolEval 2018 Workshop. pp. 63–73. Institute of Computer Science, Polish Academy of Science (2018)
2. Broda, B., Marciniak, M., Maziarz, M., Radziszewski, A., Wardyński, A.: KPWr: Towards a free corpus of polish. In: Proceedings of the 8th International Conference on Language Resources and Evaluation, LREC 2012 (2012)
3. Grave, E., Bojanowski, P., Gupta, P., Joulin, A., Mikolov, T.: Learning word vectors for 157 languages. In: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018) (2018)
4. Kocouć, J., Gawor, M.: Evaluating KGR10 Polish word embeddings in the recognition of temporal expressions using BiLSTM-CRF



**Fig. 1.** Precision, recall, and F-measure of fine-grained NER for each category. The size of the circle indicates the number of examples of a given category in the test dataset.

5. Marcińczuk, M., Wawer, A.: Named entity recognition for Polish. *Poznan Studies in Contemporary Linguistics* **55**(2) (2019). <https://doi.org/10.1515/pscl-2019-0010>
6. Marcińczuk, M., Oleksy, M., Dziob, A.: KPWr annotation guidelines — named entities (2016), <http://hdl.handle.net/11321/294>, CLARIN-PL digital repository
7. Marcińczuk, M., Kocoń, J., Gawor, M.: Recognition of named entities for polish-comparison of deep learning and conditional random fields approaches. In: Ogrodniczuk, M., Kobyliński, (eds.) *Proceedings of the PolEval 2018 Workshop*. pp. 77–92. Institute of Computer Science, Polish Academy of Science (2018)
8. Przeźiórkowski, A., Bańko, M., Górska, R.L., Lewandowska-Tomaszczyk, B.: *Narodowy Korpus Języka Polskiego* (2012)
9. Savary, A., Waszczuk, J.: Narzędzia do anotacji jednostek nazewniczych. In: Przeźiórkowski, A., Bańko, M., Górska, R.L., Lewandowska-Tomaszczyk, B. (eds.) *Narodowy Korpus Języka Polskiego*, pp. 225–252. Wydawnictwo Naukowe PWN, Warsaw (2012)
10. Tjong Kim Sang, E.F., De Meulder, F.: Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In: *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003* (2003)
11. Wawer, A., Małek, E.: Results of the poleval 2018 shared task 2: Named entity recognition. In: Ogrodniczuk, M., Kobyliński, (eds.) *Proceedings of the PolEval 2018 Workshop*. pp. 53–62. Institute of Computer Science, Polish Academy of Science (2018)

# Information retrieval from biomedical text strongly depends on parameters: application to the bioCADDIE benchmark\*

Artur Cieślewicz<sup>1</sup>, Jakub Dutkiewicz<sup>2</sup>, and Czesław Jędrzejek<sup>2</sup>

<sup>1</sup> Department of Clinical Pharmacology  
Poznan University of Medical Sciences, Poznan, Poland  
<sup>2</sup> IARII, Poznan University of Technology  
Maria Skłodowska-Curie Sq. 5, 60-965 Poznan, Poland

**Abstract.** This work presents some improvements upon our previous results of bioCADDIE 2016. The Terrier LGD baseline, combined with query expansion using word embedding calculated for a biomedical corpus, allowed us to achieve infNDCG close to the original winning result of bioCADDIE. Across all measures our results are significantly better compared to existing results. The most important factor is assigning much smaller weights to expanded terms compared to the original terms in a query. We also calculate measures used in NTCIR and find they are more stable upon changes compared to the inferred measures of TREC and bioCADDIE.

**Keywords:** information retrieval · query expansion · bioCADDIE.

## 1 Introduction

In this work we address the challenge of information retrieval effectiveness over time for medical documents. The existence and use of standard test collections in information retrieval experimentation, in principle, allows results to be compared between research groups, methods and over time. This is important because the information search process is one of the key tasks performed by IT systems. In 2009 [2] had performed an analysis of published IR results in SIGIR and CIKM proceedings from 1998-2008 and uncovered the fact that ad hoc retrieval was not measurably improving. The authors noticed consistent improvement of one element of computations, namely increasing measures by deliberately choosing weak baselines. With the progress of machine learning, particularly neural networks, neural information retrieval should follow other areas of NLP, surpassing classical methods. However, recent meta-analysis of papers that have reported experimental results on the TREC Robust04 [17] test collection finds no evidence of an upward trend in effectiveness over time. The Lin's group [20] divided methods used in these papers into 16 neural [11] and 92 nonneural approaches<sup>3</sup> and revealed that the neural methods did not perform well, contrary to what had been

---

\* Supported by the PUT DS grant no 04/45/DSPB/0197 and 04/45/DSMK/0200

<sup>3</sup> <https://github.com/lintool/robust04-analysis>

claimed. In particular, the much hyped method BERT is able to obtain MAP measure equal 0.3278 [21]. This compares with historically best MAP=0.3686 [9]. For the last several years, the National Institute of Standards of Technology's the Text REtrieval Conference (TREC) has concentrated on finding the most relevant PubMed articles and clinical trial data in response to selected medical records within its Clinical Decision Support (CDS) Track evolving into Precision Medicine. This methodology, similar to the TREC CDS/PM (Precision Medicine) methodology, was used in the bioCADDIE 2016 challenge. After the challenge, the Database journal devoted a Special Virtual Issue analysing the methodology [8] and overall results of the challenge [12] together with particular methods of participants that were published in 2017 and 2018. Our work is based a retrieval system that obtained good results in the TREC 2016 CDS, TREC 2017 PM, TREC 2018 PM and the bioCADDIE 2016 challenge as published in [7]. The bioCADDIE Challenge was conducted using a collection of metadata from biomedical datasets, generated from 794 992 XML documents extracted from a set of 20 individual repositories. A set of representative example queries for biomedical data, determined by domain experts, were provided for system development. Evaluation was conducted using a manually annotated benchmark dataset, that consisted of a 15 queries, with relevance judgments for datasets in the provided collection. The datasets were annotated as relevant, partially relevant and not relevant to the query [8]. Among the papers published in the virtual issue in the Database journal we achieved the best result for infAP and the second best result for infNDCG [7]. Our method used the Terrier LGD baseline combined with query expansion using word embedding, calculated for biomedical corpus; this allowed us to achieve infNDCG close to the original winning result of bioCADDIE. The most important factor is assigning much smaller weights of expanded terms compared to original terms in a query.

This work aims at improving the bioCADDIE 2016 challenge benchmark. We apply several enhancements to investigate whether they add up to better evaluation measures. Inspection of participant contributions revealed that in general, leading teams that achieved higher infNDCG results suffered from poorer infAP and P@10 measures, and vice versa. Therefore, we extend the bioCADDIE results by another set of measures: bpref, AP and Q. AP and Q are mostly used for NTCIR, a Japanese version of TREC [14]. Interestingly, to our knowledge infAP, infNDCG, AP and Q have never been analysed together.

## 2 Evaluation procedure and applied measures

When a document set is close or exceeds one million documents, its complete evaluation is impossible. Only a fraction of documents is judged. The simplest method is pooling by taking 10-100 topmost documents from each participant's runs. This is an exhaustive but shallow stratum. From several possible more complex strategies the bioCADDIE organizers chose so-called 2stratum [18] - an exhaustively judged small initial stratum plus a moderately sized, sampled second stratum. The bioCADDIE organizers selected the all documents in ranks

1-10 plus 5% of documents in the ranks 11-100. The submitted ranking list has a size of 1000 documents. TREC and the bioCADDIE used inferred metrics, namely infAP and infNDCG that are complex and calculated on the assumption that we know all relevant documents in the corpus [22], [3]. Tests on TREC sets at that time showed that the neglecting relevant documents above the so-called "depth100" is acceptable. This strategy is partly motivated by the fact that it allows precision P (first stratum depth) to be computed exactly. Altogether 20184 datasets were judged and among them only 812 were fully relevant. NTCIR was of a different opinion and have not used inferred measures [13].

## 2.1 Applied Measures

Due to the vast resources required to evaluate Information Retrieval systems, recent Information Retrieval challenges (TREC, Biocaddie) use measures designated for the incomplete judgements sets and measures which consider only the top scoring documents, such as

- inferred Average Precision (infAP) - this estimated measure simulates Average Precision, a measure which benefits systems, which tend to discover most of the relevant documents,
- inferred Normalized Discounted Cumulative Gain - a measure, which is intended to award the systems, which put the documents in a proper order – fully relevant documents at the top, partially relevant documents in the middle, and non-relevant documents at the bottom of a score list,
- Average Precision at 10 - this measure highlights the systems, which find the most relevant documents and put them at the top of the scoring list,
- Normalized Discounted Cumulative Gain at 10 - a measure similar to the infNDCG, but limited to the top 10 documents.

In our evaluation we also include a set of measures designed for the incomplete judgement sets and used by NTCIR. These measures use the idea of condensed score lists - lists with the non-annotated and unjudged documents removed. Detailed description of those measures can be found in [13]. In the following formulas, the  $rel(i)$  symbol denotes relevance of the  $i$ -th document on the list and the  $count(i)$  symbol indicates the number of relevant documents up to the  $i$ -th document. The  $ideal(i)$  represents the relevance of the  $i$ -th document on the ideally ordered list of documents. Symbols and functions which use the condensed lists are denoted by an apostrophe(').

$$NDCG@10 = \frac{\sum_{i=1}^{i=10} \frac{rel(i)}{\log_2 i+1}}{\sum_{i=1}^{i=10} \frac{ideal(i)}{\log_2 i+1}} \quad (1)$$

$$P@10 = \frac{count(10)}{10} \quad (2)$$

$$AP' = \frac{1}{R} \sum_{i=1}^{i=R} rel'(i) \cdot \frac{count'(i)}{i} \quad (3)$$

$$Q' = \frac{1}{R} \sum i = 1^{i=R} rel'i \cdot \frac{\beta \cdot \sum_{j=1}^{j=i} rel'(i) + count'(j)}{\beta \cdot \sum_{j=1}^{j=i} ideal'(i) + j} \quad (4)$$

$$bpref = \frac{1}{R} \sum_{i=1}^{i=R} rel'(i) \left( 1 - \frac{\min(R, i - count'(i))}{R} \right) \quad (5)$$

### 3 Results

As in [7], this study adopts the Divergence From Randomness search model [1] and indexing based on the implementation the Terrier's engine [10]. Terrier also performs the information retrieval process, taking as an input a set of queries and a collection of documents, and constitutes a reference model (the so-called baseline). As in the bioCADDIE 2016 challenge the various Terrier models (mostly LGD and BB2) serving as baselines were combined with query expansion using word embedding, calculated for biomedical corpus based on the Pubmed abstracts allowed us achieving infNDCG close to the original winning result of the bioCADDIE challenge. The most important factor is assigning much smaller weights of expanded terms compared to original terms in a query. This new result has not been theoretically explained so far. In the bioCADDIE benchmark for the infNDCG measure, among the papers published in the Virtual issues our result (the second best) was around 0.06 lower than the winning result of 0.5132, which remains the best result today.

This work uses two improvements. Instead of using word embedding, based on the Pubmed abstracts, we use the corpus proposed by [6]. Assigning much smaller weights to the expanded terms compared to original terms in a query is crucial and increased infNDCG by over 0.04 but at the same time decreased infAP and P@10. This is an overall effect, however for some topics the [6] method is not always better. The second source of improvement comes from the the fusion of the two best DFR systems using the Borda count method [4]. The current status of the bioCADDIE results together with this results is presented in Table 1.

We also perform calculations using condensed score lists omitting the unjudged documents from evaluation algorithms (the bpref [5] - not currently used in the TREC evaluation, and its improvements AP and Q). The results obtained with these measures express better robustness compared to the inferred measures' results. This raises conjecture that the conditions of the application of inferred algorithms are not met in the bioCADDIE evaluations, similar to the TREC PM evaluations.

In Table 1 two set of entries represent data for slightly different evaluation conditions. These are the results of [19], [7] and our results. For the submission to the original challenge, documents for judgment were selected based on each participant's highest runs (based on top 10 + randomly selected samples). Less of documents used in our work and [19] are judged documents; therefore the results were obtained in a slightly different evaluation environment. There is a

**Table 1.** Selected,top results in the contest and the virtual volume of the bioCADDIE

| Team               | Submission                        | inf AP        | inf NDCG      | NDCG @10      | P@10 + partial | P@10 - partial | bpref        | AP'          | Q'           |
|--------------------|-----------------------------------|---------------|---------------|---------------|----------------|----------------|--------------|--------------|--------------|
| UCSD [19]          | all PSD                           | 0.2792        | 0.4980        | 0.612         | 0.7600         | 0.3267         |              |              |              |
| UCSD challenge     | armyofucsdgrads -3.txt            | 0.1468        | <b>0.5132</b> | 0.5303        | 0.7133         | 0.2400         |              |              |              |
| SIBTex [16]        | sibtex-50.txt                     | 0.3664        | 0.4188        | 0.6271        | 0.7533         | 0.3467         |              |              |              |
| Elsevier [15]      | elsevier4.txt                     | 0.3049        | 0.4368        | <b>0.6861</b> | <b>0.8267</b>  | <b>0.4267</b>  |              |              |              |
| Poznan (paper,[7]) | LGD word2vec and Terrier Rocchio  | 0.3978        | 0.4539        | 0.6375        | 0.7700         | 0.4000         | 0.578        | 0.649        | 0.65         |
| This work          | LGD word2vec (embedding [6])      | 0.3387        | 0.4988        | 0.624         | 0.7367         | 0.406          | 0.587        | 0.657        | 0.659        |
| This work          | LGD (word2vec) and BB2 (word2vec) | 0.3385        | 0.5079        | 0.612         | 0.7667         | 0.333          | <b>0.599</b> | <b>0.659</b> | <b>0.662</b> |
| This work          | LGD (baseline) and BB2 (word2vec) | <b>0.4092</b> | 0.4508        | 0.633         | 0.81           | 0.393          | 0.576        | 0.637        | 0.638        |

discrepancy in the UCSD results between the challenge results and this paper. Remaining differences may come from the variation in testing environments.

The striking feature in these results is the negative correlation between infAP and infNDCG. Possible explanations of this observation include:

1. Due to a very complicated nature of the search, there is a larger than expected human evaluation error,
2. The number of judged documents is too small and assumption of the incomplete judgment theory that all relevant documents are within so called "depth-100" [3] set is not fulfilled.

Analysis of the original bioCADDIE 2016 results reveals disturbing features. Comparing all bioCADDIE runs based on the infAP, infNDCG, NDCG and P@10 there is surprisingly little correlation. The same trend happens with the results of this work: we significantly improved infNDCG but at the same time a value of infAP correspondingly deteriorated. We have already achieved infNDCG above 0.52 (which would be the best result for the benchmark) but infAP went below 0.1 which we do not consider realistic correlation between evaluated results for these measures.

This could be an effect of incompatibility of theoretical derivation of inferred formulae vs procedures used in the bioCADDIE evaluation (similar procedures were used in TREC PM 2017 and 2018). Another possibility is errors in the sample and evaluating codes (particularly, action of these programs when k=0 is

met). During a discussion at the 27 TREC meeting at Gaithersburg , November, 2018 such concern was voiced.

To shed light on this behavior we plan to simulate qrels. These simulations might reveal:

1. Possible inadequate stability of the infNDCG results,
2. Queries should be normalized and # relevant documents per query should be similar; otherwise additional variance is introduced.

The result of the bioCADDIE challenge show that infNDCG is the least correlated measure with others. We speculate that the error in obtained result for infNDCG could be as large as 0.15 <sup>4</sup>. The most favorable case would be if # of relevant documents per topic should be little less than the cutoff k=100. K=1000 and very small or none # of relevant documents per topic distort the results of evaluations and could have impacted the evaluation of bioCADDIE 2016 contest.

## 4 Conclusions

In this research we summarize the status of current Biomedical Information Retrieval benchmarks. We studied a set of improvements on our previous methodology applied to the bioCADDIE challenge. It turns out that implementation of a Borda Count method to several results improves the scores achieved by the IR system for all of the evaluation measures.

We also study the impact of word embedding based query expansion on the evaluation measures. The two classical measures - infAP and infNDCG are ambiguous in that matter. With the word embedding expansions added, we observe an increase in infNDCG and a decrease in infAP. We used an alternative set of measures to verify that issue. Ultimately, the evaluation gives an edge to the systems, which use the word embedding based query expansion. We believe, this research gives the legitimacy of using such a type of query expansion in all settings.

## References

1. Amati, G., Van Rijsbergen, C.J.: Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Trans. Inf. Syst.* **20**(4), 357–389 (Oct 2002). <https://doi.org/10.1145/582415.582416>, <http://doi.acm.org/10.1145/582415.582416>
2. Armstrong, T.G., Moffat, A., Webber, W., Zobel, J.: Improvements that don't add up: ad-hoc retrieval results since 1998. In: *Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM 2009, Hong Kong, China, November 2-6, 2009.* pp. 601–610 (2009). <https://doi.org/10.1145/1645953.1646031>, <https://doi.org/10.1145/1645953.1646031>

---

<sup>4</sup> our forthcoming paper

3. Aslam, J.A., Pavlu, V., Yilmaz, E.: A statistical method for system evaluation using incomplete judgments. In: Efthimiadis, E.N., Dumais, S.T., Hawking, D., Järvelin, K. (eds.) SIGIR 2006: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, USA, August 6-11, 2006. pp. 541–548. ACM (2006). <https://doi.org/10.1145/1148170.1148263>
4. Benham, R., Culpepper, J.S.: Risk-reward trade-offs in rank fusion. In: Proceedings of the 22nd Australasian Document Computing Symposium, ADCS 2017, Brisbane, QLD, Australia, December 7-8, 2017. pp. 1:1-1:8 (2017). <https://doi.org/10.1145/3166072.3166084>
5. Buckley, C., Voorhees, E.M.: Retrieval evaluation with incomplete information. In: SIGIR 2004: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, UK, July 25-29, 2004. pp. 25–32 (2004). <https://doi.org/10.1145/1008992.1009000>
6. Chiu, B., Crichton, G., Korhonen, A., Pyysalo, S.: How to train good word embeddings for biomedical NLP. In: Proceedings of the 15th Workshop on Biomedical Natural Language Processing. pp. 166–174. Association for Computational Linguistics, Berlin, Germany (Aug 2016). <https://doi.org/10.18653/v1/W16-2922>
7. Cieslewicz, A., Dutkiewicz, J., Jedrzejek, C.: Baseline and extensions approach to information retrieval of complex medical data: Poznan’s approach to the biocaddie 2016. Database **2018**, bax103 (2018). <https://doi.org/10.1093/database/bax103>
8. Cohen, T., Roberts, K., Gururaj, A.E., Chen, X., Pournejati, S., Alter, G., Hersh, W.R., Demner-Fushman, D., Ohno-Machado, L., Xu, H.: A publicly available benchmark for biomedical dataset retrieval: the reference standard for the 2016 biocaddie dataset retrieval challenge. Database **2017**, bax061 (2017). <https://doi.org/10.1093/database/bax061>
9. Cormack, G.V., Clarke, C.L.A., Büttcher, S.: Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009, Boston, MA, USA, July 19-23, 2009. pp. 758–759 (2009). <https://doi.org/10.1145/1571941.1572114>
10. Macdonald, C., McCreadie, R., Santos, R.L., Ounis, I.: From puppy to maturity: Experiences in developing terrier. Proc. of OSIR at SIGIR pp. 60–63 (2012)
11. Rao, J., Yang, W., Zhang, Y., Türe, F., Lin, J.: Multi-perspective relevance matching with hierarchical convnets for social media search. In: The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019. pp. 232–240 (2019), <https://aaai.org/ojs/index.php/AAAI/article/view/3790>
12. Roberts, K., Demner-Fushman, D., Voorhees, E.M., Hersh, W.R., Bedrick, S., Lazar, A.J., Pant, S.: Overview of the TREC 2017 precision medicine track. In:

- TREC. vol. Special Publication 500-324. National Institute of Standards and Technology (NIST) (2017)
13. Sakai, T.: Alternatives to bpref. In: SIGIR 2007: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, July 23-27, 2007. pp. 71–78 (2007). <https://doi.org/10.1145/1277741.1277756>, <https://doi.org/10.1145/1277741.1277756>
  14. Sakai, T.: Graded relevance assessments and graded relevance measures of NTCIR: A survey of the first twenty years. CoRR **abs/1903.11272** (2019), <http://arxiv.org/abs/1903.11272>
  15. Scerri, A., Kuriakose, J., Deshmane, A.A., Stanger, M., Cotroneo, P., Moore, R., Naik, R., de Waard, A.: Elsevier’s approach to the biocaddie 2016 dataset retrieval challenge. Database **2017**, bax056 (2017). <https://doi.org/10.1093/database/bax056>, <https://doi.org/10.1093/database/bax056>
  16. Teodoro, D., Mottin, L., Gobeill, J., Gaudinat, A., Vachon, T., Ruch, P.: Improving average ranking precision in user searches for biomedical research datasets. Database **2017**, bax083 (2017). <https://doi.org/10.1093/database/bax083>, <https://doi.org/10.1093/database/bax083>
  17. Voorhees, E.M.: The TREC robust retrieval track. SIGIR Forum **39**(1), 11–20 (2005). <https://doi.org/10.1145/1067268.1067272>, <https://doi.org/10.1145/1067268.1067272>
  18. Voorhees, E.M.: The effect of sampling strategy on inferred measures. In: The 37th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR ’14, Gold Coast , QLD, Australia - July 06 - 11, 2014. pp. 1119–1122 (2014). <https://doi.org/10.1145/2600428.2609524>, <https://doi.org/10.1145/2600428.2609524>
  19. Wei, W., Ji, Z., He, Y., Zhang, K., Ha, Y., Li, Q., Ohno-Machado, L.: Finding relevant biomedical datasets: the UC san diego solution for the biocaddie retrieval challenge. Database **2018**, bay017 (2018). <https://doi.org/10.1093/database/bay017>, <https://doi.org/10.1093/database/bay017>
  20. Yang, W., Lu, K., Yang, P., Lin, J.: Critically examining the “neural hype”: Weak baselines and the additivity of effectiveness gains from neural ranking models. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019. pp. 1129–1132 (2019). <https://doi.org/10.1145/3331184.3331340>, <https://doi.org/10.1145/3331184.3331340>
  21. Yang, W., Zhang, H., Lin, J.: Simple applications of BERT for ad hoc document retrieval. CoRR **abs/1903.10972** (2019), <http://arxiv.org/abs/1903.10972>
  22. Yilmaz, E., Aslam, J.A.: Estimating average precision with incomplete and imperfect judgments. In: Yu, P.S., Tsotras, V.J., Fox, E.A., Liu, B. (eds.) Proceedings of the 2006 ACM CIKM International Conference on Information and Knowledge Management, Arlington, Virginia, USA, November 6-11, 2006. pp. 102–111. ACM (2006). <https://doi.org/10.1145/1183614.1183633>, <https://doi.org/10.1145/1183614.1183633>

KE (knowledge engineering)

# Disambiguation of experts and patent inventors in the Chinese database

Robert Nowak and Wiktor Franus

Institute of Computer Science, Warsaw University of Technology,  
Nowowiejska 15/19, 00-665 Warsaw  
[robert.nowak@pw.edu.pl](mailto:robert.nowak@pw.edu.pl)

## 1 The Chinese experts database

In our research we worked with the Chinese experts database, created and shared with us by the Shanghai Science and Technology Talent Development Center. The database contains information about scientific publications, patents, authors of papers, authors with significant number of publications referred to as experts and affiliations of all authors. The metadata is stored in English, full documents are in English or Chinese. The proposed system uses only English metadata. The data comes from different sources and it is not completely consistent, nor normalized. The majority of tables comes from Elsevier, the company specialized in providing scientific, technical and medical information. This data includes information about papers journals, authors and affiliations. On the other hand, data about patents was supplied by another vendor, not Elsevier. It includes application and publication numbers and dates, titles, abstracts and names of inventors, among other things.

The main issue with the data concerns inconsistency of authors' identifiers, originally assigned by Elsevier in their database named Scopus [1]. In theory, author unique identifiers (*aid*) are unique keys identifying every real author indexed by Scopus. However, it was discovered from the data that one real author may have a couple of *aid* or alternatively, one *aid* may point to a couple of authors in real world. There are various causes of these problems. Firstly, the database stores expert data from China, where a lot of people share the same family name, which is very short. Secondly, the order of given name and family name in the database is inconsistent. Lastly, translation of Chinese names to English is not standardized. It is problematic for Elsevier's indexing algorithm to well differentiate between authors and as a result one record in author's table can represent two or more real authors. The opposite situation is also possible when a single real author has more than one *aid*. In our work we try to predict which *aid* may refer to the same person. We call it the experts disambiguation problem.

Another problem is disambiguation of patent inventors. It is known that for the majority of patents present in the database at least one of inventors is also an author of one or more papers. It means that there is a record related to inventor in the expert's table. However these patent-expert relation is not

present, because data about authors (and experts) and patent come from two different sources. Direct mapping of inventors into experts results in poor quality, because metadata about patent's inventors consists only of their names. Exact name matching should be avoided due to reasons described above. For now only manual linking of records gives good results.

## 2 Contributions

We plan to solve the experts disambiguation problem using hierarchical clustering. For each expert in the database we extracted three groups of features: (1) his/her English name, (2) organizations (affiliations) and (3) area of interest. While first two are stored directly in the tables, the area of interest for each author is inferred using ASJC [2] (All Science Journal Classification) codes assigned to the author's papers. The most common ASJC code is taken for each author. We experimented with agglomerate hierarchical where each author record starts in its own cluster and two most similar clusters are merged until there is only one big cluster in the final step. We are exploring different stop-criteria for this clustering algorithm at the moment.

For the inventor-expert linking problem we decided to match inventors into experts using not only their names, but also their estimated areas of interest. The latter is already defined for experts in the form of ASJC codes, based on their previous publications. For patent inventors we lacked such codes, hence we decided to infer their areas of interest based on IPC [3] (International Patent Classification) codes assigned to patents. IPC hierarchical codes describe areas of technology to which patents pertain. Our current work showed that manual mapping of IPC codes into ASJC leads to many ambiguities. As next step we propose to train a ML model for this task based on 300k patent-expert pairs already present in the database. If it will not improve quality significantly, we plan to utilize DBpedia [4] ontology and find common entities for inventors and experts. Entities will be extracted from patent abstracts using DBpedia Spotlight tool [5] in case of inventors and from paper-related keywords in case of experts.

## References

1. Scopus Homepage, <https://www.scopus.com/>. Last accessed 10 Oct 2019
2. List of ASJC codes, [https://service.elsevier.com/app/answers/detail/a\\_id/15181/supporthub/scopus/](https://service.elsevier.com/app/answers/detail/a_id/15181/supporthub/scopus/). Last accessed 10 Oct 2019
3. IPC Homepage, <https://www.wipo.int/classifications/ipc/en/>. Last accessed 10 Oct 2019
4. Lehmann, J. et al. DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. Semantic Web Journal. 6. 10.3233/SW-140134. (2015)
5. Mendes, P. N., Jakob, M., Garcia-Silva, A., Bizer, C. DBpedia Spotlight: shedding light on the web of documents. In: Proc. of I-SEMANTICS (2011)

# Mining Cardinality Restrictions in OWL

Jedrzej Potoniec<sup>[0000–0002–6115–6485]</sup>

Center for Artificial Intelligence and Faculty of Computing, Poznan University of  
Technology, Poznan, Poland  
`jpotoniec@cs.put.poznan.pl`

**Abstract.** We propose an approach for mining cardinality restrictions to be used in class inclusion axioms of OWL 2. The approach is based on kernel density estimation using Aitchison and Aitken kernel and we present preliminary experimental evidence that it is substantially better than a baseline approach using frequency estimation.

**Keywords:** ontology learning · semantic web · kernel density estimation

## 1 Introduction

The Semantic Web is an envisioned variant of the web where the content is easily readable for both machines and people, by means such as providing data in unambiguous representations and using formalized vocabulary with explicitly defined semantics. A standardized way to provide the data is to use Resource Description Framework (RDF) [12], and the semantics is provided using ontologies in OWL 2 Web Ontology Language [8].

Unfortunately, ontology engineering suffers from knowledge acquisition bottleneck, making it a long and tedious process requiring serious effort of an ontology engineer in order to collect and then formalize required knowledge. One of the ways to alleviate the issue is to use methods of (semi-)automatic ontology constructions, e.g., by using statistical and machine learning approaches in order to propose new axioms, that potentially could extend the ontology. This simplifies the process greatly: the engineer is responsible only for selecting and cleaning the axioms, instead of constructing them from scratch.

Learning requires a data source and various setups were considered in the literature, from raw text, through semi-structured data such as formalized textual definitions, to database tables, to RDF. In this paper we assume the last variant and concentrate on mining axioms of the following form, expressed in the Manchester syntax [4]:  $C \text{ SUBCLASSOF: } p \text{ MIN } n \text{ } D$  and  $C \text{ SUBCLASSOF: } p \text{ MAX } n \text{ } D$  where  $C$  and  $D$  are named classes,  $p$  is an object property and  $n$  is a natural number.

## 2 Related work

Ontology learning is an established subfield of research on the Semantic Web and on artificial intelligence, however, the research so far concentrated mostly

on less detailed axioms or on axioms in less expressive profiles of OWL. For example, considered were axioms providing partial definitions of classes in the OWL 2 EL profile [11]; axioms introducing new defined subclasses [10]; disjointness axioms [15]; domain and range restrictions [2]. Recently neural network-based approaches were considered, e.g., [9,14]. A comprehensive overview of approaches to ontology learning is presented in [7].

Mining logical constructions with formalized vocabulary was not limited to ontology learning. For example, algorithms were proposed to complete knowledge bases [3], to solve classification tasks [13], to discover frequent patterns [6].

### 3 Mining cardinality restrictions

Let  $\mathbf{G}$  be an RDF graph and  $\mathbf{O}$  a preexisting ontology, which is to be extended by the ontology engineer. Let  $C$  be a class of interest and  $S = \{s : \mathbf{G} \cup \mathbf{O} \models C(s)\}$  a set of individuals belonging to this class. Let  $F = \{(p, D) : \exists s \in S \exists o, \mathbf{G} \cup \mathbf{O} \models p(s, o) \wedge D(o)\}$  be a set of all possible features for the individuals of  $S$ .

Now, for each feature  $f = (p, D) \in F$ , we compute a histogram  $h_f$ :

$$h_f(s) = |\{o : \mathbf{G} \cup \mathbf{O} \models p(s, o) \wedge D(o)\}|$$

From the histogram, use kernel density estimation for discrete random variables, to obtain an approximate probability distribution of the number of occurrences of the feature  $f$  in the individuals of the set  $S$ . Denote by  $F_f(x)$  its cumulative distribution function [5]:

$$F_f(x) = P(h_f(\cdot) \leq x) = \sum_{k \in \{0, 1, \dots, x\}} \frac{1}{|S|} \sum_{s \in S} K(h_f(s), k)$$

where  $K$  is Aitchison and Aitken kernel parametrized by a constant  $\lambda$  [1]

$$K(x, y) = \begin{cases} 1 - \lambda & x = y \\ \frac{\lambda}{c-1} & \text{otherwise} \end{cases}$$

In the kernel  $c$  is the number of different values in the histogram  $h_f(s)$ . To compute  $\lambda$  we use the *normal reference*:  $\lambda = 1.06 \cdot \hat{\sigma} \cdot |S|^{-\frac{1}{5}}$  where  $\hat{\sigma}$  is the sample standard deviation of  $\{h_f(s) | s \in S\}$ . Observe that for  $\lambda = 0$ , this estimator collapses to frequency estimation.

Let  $\theta$  be a user-defined probability threshold. We now find two positive integer values,  $x_f^{\min}$  and  $x_f^{\max}$  such that  $F_f(x_f^{\min}) < \frac{1-\theta}{2}$  and  $F_f(x_f^{\max}) \geq 1 - \frac{1-\theta}{2}$ . These values correspond to a range  $[x_f^{\min}, x_f^{\max}]$  such that, with probability  $\theta$ , the number of occurrences of the feature  $f$  is within this range and the remaining probability  $1-\theta$  is equally divided between the values above and below the range. It is possible that there are no such values and then  $x_f^{\min} = 0$ , i.e., there is no lower limit on the number of occurrences or  $x_f^{\max} = \infty$ , i.e., there is no upper limit.

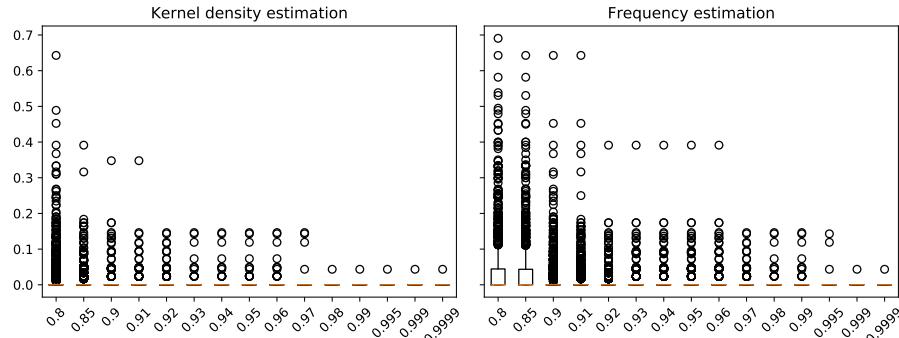
From this, we use a simple strategy for construction axioms. For a feature  $f = (p, D)$  if  $x_f^{\min} > 0$ , then yield the axiom  $C \text{ SUBCLASSOF: } p \text{ MIN } x_f^{\min} D$  and if  $x_f^{\max} < \infty$ , then yield the axiom  $C \text{ SUBCLASSOF: } p \text{ MAX } x_f^{\max} D$ .

## 4 Preliminary experimental evaluation

One may wonder whether using such a complex machinery offers any advantage over a simple frequency estimation. In order to address this question we performed the following experiment. We used DBpedia 2016-10 and we randomly selected 20 classes having between 1000 and 10000 individuals. To account for variability due to the selection of the threshold  $\theta$ , we performed computations for 15 different values ranging from 0.8 to 0.9999. For a given class and a given threshold, we mined cardinality restrictions as described in the previous section, and the computed the number  $iv$  of individuals violating the mined restrictions:

$$iv = \frac{\left| \{s \in S : \exists f \in F, h_f(s) \notin [x_f^{min}, x_f^{max}] \} \right|}{|S|}$$

To make the results more robust we used 100-fold cross validation: for each pair class– $\theta$ , we split the set  $S$  into 100 parts, and repeated computations 100 times, every time using 99 parts to estimate  $x_f^{min}$  and  $x_f^{max}$ , while computing  $iv$  on the remaining part. For comparative purposes we performed an experiment under the same condition using frequency estimate, i.e. assuming  $\lambda = 0$ . The results aggregated over all 20 classes are presented in Figure 1, while class by class comparison is available on-line at <https://doi.org/10.6084/m9.figshare.9885470>. We observe that kernel density estimation always yields number of violating individuals no greater, and often significantly smaller, than frequency estimation, which suggest that the results obtained with kernel density estimation are more robust to noise than while using frequency estimation.



**Fig. 1.** Number of violating individuals under 100-fold cross-validation aggregated over all considered classes. The left chart presents a boxplot for kernel density estimation, while the right chart for frequency estimation.

## 5 Conclusions and future work

We presented an approach to mine OWL cardinality restrictions that can be used in class inclusion axioms. The approach is based on kernel density estimation and we provided preliminary evidence that the approach is substantially better than a baseline based on frequency estimation.

There are multiple avenues to extend the work. First, we plan to consider more elaborate strategies of transforming features and corresponding estimations to axioms, in a way that would take into account the preexisting ontology. Then, we would like to analyze noise-tolerance of the proposed approach and see whether it is more robust than frequency estimation. Finally, using the normal reference to compute  $\lambda$  is not a well-founded approach in this context and we would like to consider more robust approaches.

**Acknowledgement.** We acknowledge support from the grant 09/91/DSMK/0659.

## References

1. AITCHISON, J.,AITKEN, C.G.G.: Multivariate binary discrimination by the kernel method. *Biometrika* **63**(3), 413–420 (12 1976)
2. Fleischhacker, D., et al.: Mining RDF data for property axioms. LNCS, vol. 7566, pp. 718–735. Springer (2012). [https://doi.org/10.1007/978-3-642-33615-7\\_18](https://doi.org/10.1007/978-3-642-33615-7_18)
3. Galárraga, L., Teflioudi, C., Hose, K., Suchanek, F.M.: Fast rule mining in ontological knowledge bases with AMIE+. VLDB J. **24**(6), 707–730 (2015)
4. Horridge, M., Patel-Schneider, P.: OWL 2 web ontology language manchester syntax (second edition). W3C note, W3C (Dec 2012)
5. Ju, G., et al.: Nonparametric estimation of multivariate CDF with categorical and continuous data. In: Adv. in Econometrics, pp. 291–318. Emerald Group Publishing
6. Lawrynowicz, A., Potoniec, J.: Pattern based feature construction in semantic data mining. Int. J. Semantic Web Inf. Syst. **10**(1), 27–65 (2014)
7. Lehmann, J., Völker, J.: Perspectives on Ontology Learning, Studies on the Semantic Web, vol. 18. IOS Press (2014). <https://doi.org/10.3233/978-1-61499-379-7-i>
8. Parsia, B., Rudolph, S., Patel-Schneider, P., Hitzler, P., Krötzsch, M.: OWL 2 web ontology language primer (second edition). Tech. rep., W3C (Dec 2012)
9. Petrucci, G., et al.: Expressive ontology learning as neural machine translation. J. Web Semant. **52–53**, 66–82 (2018). <https://doi.org/10.1016/j.websem.2018.10.002>
10. Potoniec, J., Lawrynowicz, A.: Combining ontology class expression generation with mathematical modeling for ontology learning. In: Bonet, B., Koenig, S. (eds.) Proc. of the 29<sup>th</sup> AAAI Conf. on AI. pp. 4198–4199. AAAI Press (2015)
11. Potoniec, J., et al.: Swift linked data miner: Mining OWL 2 EL class expressions directly from online RDF datasets. J. Web Semant. **46–47**, 31–50 (2017)
12. Raimond, Y., Schreiber, G.: RDF 1.1 primer. W3C note, W3C (Jun 2014), <http://www.w3.org/TR/2014/NOTE-rdf11-primer-20140624/>
13. Rizzo, G., et al.: Tree-based models for inductive classification on the web of data. J. Web Semant. **45**, 1–22 (2017). <https://doi.org/10.1016/j.websem.2017.05.001>
14. Song, W., et al.: Explainable knowledge graph-based recommendation via deep reinforcement learning. CoRR **abs/1906.09506** (2019)
15. Völker, J., et al.: Automatic acquisition of class disjointness. J. Web Semant. **35**, 124–139 (2015). <https://doi.org/10.1016/j.websem.2015.07.001>

# **Towards smart enterprises: supporting the business processes using artificial intelligence**

Marcin Hernes<sup>1</sup>

<sup>1</sup>Wrocław University of Economics and Business, ul. Komandorska 118/120, 53-345 Wrocław,  
Poland  
[marcin.hernes@ue.wroc.pl](mailto:marcin.hernes@ue.wroc.pl)

**Abstract.** Business processes' supporting in enterprises is implemented using economic, statistical, mathematical and other methods. However these traditional methods do not have the ability to extract the deep relationship between data and to adapt to permanent changes occurring in the organization's environment. Therefore their performance can be not sufficient to effective business processes supporting. The artificial intelligence tools can be used to resolve these problems. The main aim of researches is to develop methods to support the implementation of selected business processes using artificial intelligence technologies (mainly deep learning and cognitive technologies). On the basis of preliminary research experiment's results, related to financial investments, it can be concluded that the deep learning model is characterized by the higher level of performance than traditional methods.

**Keywords:** Smart enterprises, artificial intelligence, deep learning, business processes, cognitive architectures.

## **1 Introduction**

Modern enterprise development trends are directed at Industry 4.0, which is associated with the concept of smart enterprise (smart factory) including the integration and automation of planning, steering and monitoring of technological and environmental processes, as well as the application of artificial intelligence methods and technologies and Big Data in industrial diagnostics [1]. In [2] it was stated that in order to support business processes, mainly statistical methods and traditional data mining methods, such as regression analysis and decision trees, are used. Limitations of these methods have also been shown, such as: they do not extract the deep relationship between factors affecting the efficiency of business processes; they do not allow to identify cause-and-effect relationships between these variables; data mining models do not have the ability to adapt to permanent changes occurring in the organization's environment; they are insufficient in case the heterogeneity of business processes.

These limitations can be largely eliminated by using artificial intelligence technologies, such as neural networks, expert systems, genetic algorithms, or cognitive technologies [3]. These technologies allow for support, among others, such areas as financial forecasts, e.g. [4], production management, e.g. [5], energy consumption planning [6]. However, there are many areas of business activity in which artificial intelligence

technologies have not yet been applied or are just beginning to be used. The main aim of these researches is to develop methods to support the implementation of selected business processes (such as: automation of the customer service process, automatic analysis of textual opinions, automation of Environmental Life Cycle Costing -ELCC, financial investments, Internet of Things security using BlockChain) by application the artificial intelligence technologies (mainly deep learning and cognitive technologies). The justification for taking up the problem is the dynamic development of artificial intelligence technologies and their increasing use in supporting the management, especially in solutions in the field of Industry 4.0. At the same time, it should be noted that there is currently a research gap regarding the use of artificial intelligence in supporting indicated business processes. The researches are performed by the Intelligent Management Systems Center ([www.imscenter.pl](http://www.imscenter.pl)) research team. The researches are financed by the Ministry of Science and Higher Education in Poland under the programme "Regional Initiative of Excellence" 2019 - 2022 project number 015/RID/2018/19 total funding amount 10 721 040,00 PLN"

## **2 Research methodology**

The Design Science Research methodology is adopted. It consists of following steps:

1. Awareness of problem: Analysis of existing ways to support selected business processes. Research methods: literature analysis, observation of on-site phenomena in enterprises.
2. Suggestion: Development of assumptions, concepts regarding the methods being developed as well as the simulation environment and the research stand. Research methods: modeling, analysis and design of IT systems.
3. Development: Acquisition, analysis and modeling of data from cyberspace and internal enterprise databases. Research methods: quantitative (statistical) methods, database modeling; Development of a simulation environment - an IT application allows for simulation of selected business processes using simulated or real data acquired from enterprises. Research methods: modeling, analysis and design of IT systems; Development of methods to support selected business processes. Research methods: case study, analysis of deep learning and cognitive technologies, modeling.
4. Evaluation: Development of test scenarios and testing the developed methods of supporting selected business processes. Research methods: A research experiment using a simulation environment.
5. Conclusions: Elaboration of research results.

## **3 Preliminary results**

Our initial research has been focused on financial investment. This was performed on the basis of the needs of one of the enterprises that is a member of our research Center. The subject of the study was the development of a module for the automatic

assessment of the customer's creditworthiness for a leasing institution using deep learning. The set of data obtained from the leasing company contained historical data of several thousand of its partners. The entire collection presented 54 000 records. Each record was characterized by a set of 156 attributes containing cross-sectional information about the lessee (the attributes relate to the areas: contract details, applicant's finances, legal forms, accounting data, historical data on the applicant's operations, management structure, administrative data). The output of the model is binary variable named *default* - in the case of the value "0" the customer has fulfilled the contract, in the case of the value "1" - the customer has not fulfilled the contract, i.e. he has stopped paying off the leasing installments. The deep learning model was developed using the Keras framework with the TensorFlow engine. The created model is a binary classifier. The model consists of six interconnected neural networks. The output of each of the six neural networks is the input to the next neural network, at the end of which there is one output neuron. The output neuron assumes values in the range of 0 - 1. The result is interpreted as follows: if its value is close to unity, we assume that leasing will be successfully completed; however, if its value is close to zero, leasing will be discontinued. Embedding layers have been used in some submodels. The main model is a classic multiperceptron. For training and testing, a data set of 15060 records, 11700 records with the *default* = 0 and 3360 records with the *default* = 1 was selected. A data set consisting of 10542 records was used during the training of the model. Accuracy is calculated based on a set of 4518 records (testing set). Figure 1 shows the model accuracy values (140 learning epoch).

**Fig. 1.** The model accuracy

The average accuracy of the developed experimental prototype is 89.07%. Accuracy was also tested within each value class of the *default* variable and compared to multiple regression model). For 3533 records for which the *default* equals 0 the accuracy 93.24% was obtained (using multiple regression – 92.43% ).

With reference to 985 records where the default equals 1, an accuracy 74.11% was achieved (using multiple regression – only 1%).

#### 4 Conclusions

The artificial intelligence tools allow for improving the business processes supporting. On the basis of preliminary research experiments it can be concluded, that in the financial investment field the deep learning model allow for achieve highest accuracy than traditional statistical methods. The model presented in this paper has been implemented in leasing decision support system and will be used by final customers.

The further research works are planned, such as developing deep learning model for food demand forecasting or developing ELCC module using the cognitive agents' architectures.

## References

1. Lasi, H., Fettke, P., Kemper, H. G., Feld, T., & Hoffmann, M.: Industry 4.0. Business & information systems engineering, 6(4), 239-242 (2014).
2. Hassouna, M., Tarhini, A., Elyas, T., & AbouTrab, M. S: Customer churn in mobile markets a comparison of techniques. arXiv preprint arXiv:1607.07792. . (2016).
3. Folguera, L., Zupan, J., Cicerone, D., & Magallanes, J. F.: Self-organizing maps for imputation of missing data in incomplete data matrices. Chemometrics and Intelligent Laboratory Systems, 143, 146-151 (2015).
4. Dixon, M., Klabjan, D., & Bang, J. H.: Classification-based financial markets prediction using deep neural networks. Algorithmic Finance, (Preprint), 1-11. (2017).
5. Mehdiyev, N., Lahann, J., Emrich, A., Enke, D., Fettke, P., & Loos, P.: Time series classification using deep learning for process planning: a case from the process industry. Procedia Computer Science, 114, 242-249 (2017).
6. Mocanu, E., Nguyen, P. H., Gibescu, M., & Kling, W. L.: Deep learning for estimating building energy consumption. Sustainable Energy, Grids and Networks, 6, 91-99 (2016).
7. Vaishnavi, V., Kuechler, W., Petter, S. (Eds.) "Design Science Research in Information Systems" last updated December 2017, (2017).

# Automatic Translation of Ontology Competency Questions into SPARQL-OWL Queries

Dawid Wisniewski<sup>1</sup>[0000–0003–1194–7921] and Agnieszka Lawrynowicz<sup>1,2</sup>[0000–0002–2442–345X]

<sup>1</sup> Faculty of Computing

<sup>2</sup> Center for Artificial Intelligence and Machine Learning (CAMIL)

Poznan University of Technology

{dwisniewski, alawrynowicz}@cs.put.poznan.pl

**Abstract.** Competency Questions (CQs) are questions expressed in natural language, used to define the scope of the ontology. Ontology engineers may use them in order to track how mature the ontology being built is and decide at which moment one can stop working on it. CQs are later formalized according to the language used to represent the ontology to test the developed ontology. In our work, we consider expressing CQs in an ontology query language SPARQL-OWL as their formalisation.

Because the process of translation from natural language into SPARQL-OWL is non-trivial and time-consuming, our goal is to develop methods for (semi-)automatic translation from CQs into SPARQL-OWL queries. This poses several challenges due to: i) high expressiveness of the language, i.e. numerous, potentially very convoluted CQ formulations, ii) freedom in the way of ontology modeling, i.e. knowledge can be expressed in different ways, iii) semantic equivalence of classes, which is expressed in a non-explicit way syntactically, e.g. *domestic mammal that barks*  $\equiv$  *dog* (sometimes in a CQ there appears an entity, e.g. "dog", which is not explicitly mentioned in an ontology, then, in many cases, we need to construct a "dog" using different vocabulary, iv) limited resources, what constitutes a problem for machine learning methods, v) there are patterns in the sets of CQs, but these patterns are often common within one ontology only and often do not appear in other ontologies.

As a first step, we have gathered a set of CQs and analysed them in two principle ways: (i) performing a linguistic analysis of the natural language text itself, i.e., a lexico-syntactic analysis without any presuppositions of ontology elements, and (ii) a subsequent step of semantic analysis in order to find CQ patterns and SPARQL-OWL query templates. Subsequently, we have developed a tagger to extract so-called Glossary of Terms from CQs, which for the given sequence of words in a CQ decides whether it should be considered as a suggestion of vocabulary (a class, an instance or a property) in the created ontology, and in this way being a good candidate entry to the Glossary of Terms. These components serve as building blocks for the overall automatic method for recommending translations of ontology CQs into SPARQL-OWL.

Ultimately, we hope that our work will contribute to establishing good practices, templates and user tools that will support CQ formulation, formalisation, and general management.

# Brief Overview of Research Directions in Artificial Intelligence Methods for Business Process Management<sup>\*</sup>

Krzysztof Kluza<sup>1[0000–0003–1876–9603]</sup>, Piotr Wiśniewski<sup>1[0000–0003–3777–642X]</sup>,  
Weronika T. Adrian<sup>1[0000–0002–1860–6989], [0000–0003–3323–9328]</sup>, Antoni  
Ligęza<sup>1[0000–0002–6573–4246]</sup>, Marek Adrian<sup>1[0000–0002–0435–0994]</sup>, Bernadetta  
Stachura-Terlecka<sup>1[0000–0003–2887–5936]</sup>, and Krystian  
Jobczyk<sup>1[0000–0001–6194–2737]</sup>

AGH University of Science and Technology  
al. A Mickiewicza 30, 30-059 Krakow, Poland  
`{kluza,wpiotr,wta,ligeza,madrian,bstachur,jobczyk}@agh.edu.pl`

**Abstract.** In this paper, we present the overview of our research related to Artificial Intelligence methods in Business Process Management area. The paper presents the big picture of this research area and focuses on our contribution in this field as well as future works of our research group.

**Keywords:** artificial intelligence · business process management · business process models · information systems

## 1 Background and Motivation

It can be observed that more and more researchers in Business Process Management (BPM) exploit various techniques coming from all areas of Artificial Intelligence (AI). These include machine learning, planning methods, knowledge representation and reasoning, information extraction, multi-agent systems, natural language processing, constraint satisfaction, robotics, etc.

The aim of this paper is to give an overview of the research in AI methods for BPM conducted in the Department of Applied Computer Science at AGH University of Science and Technology in Krakow.

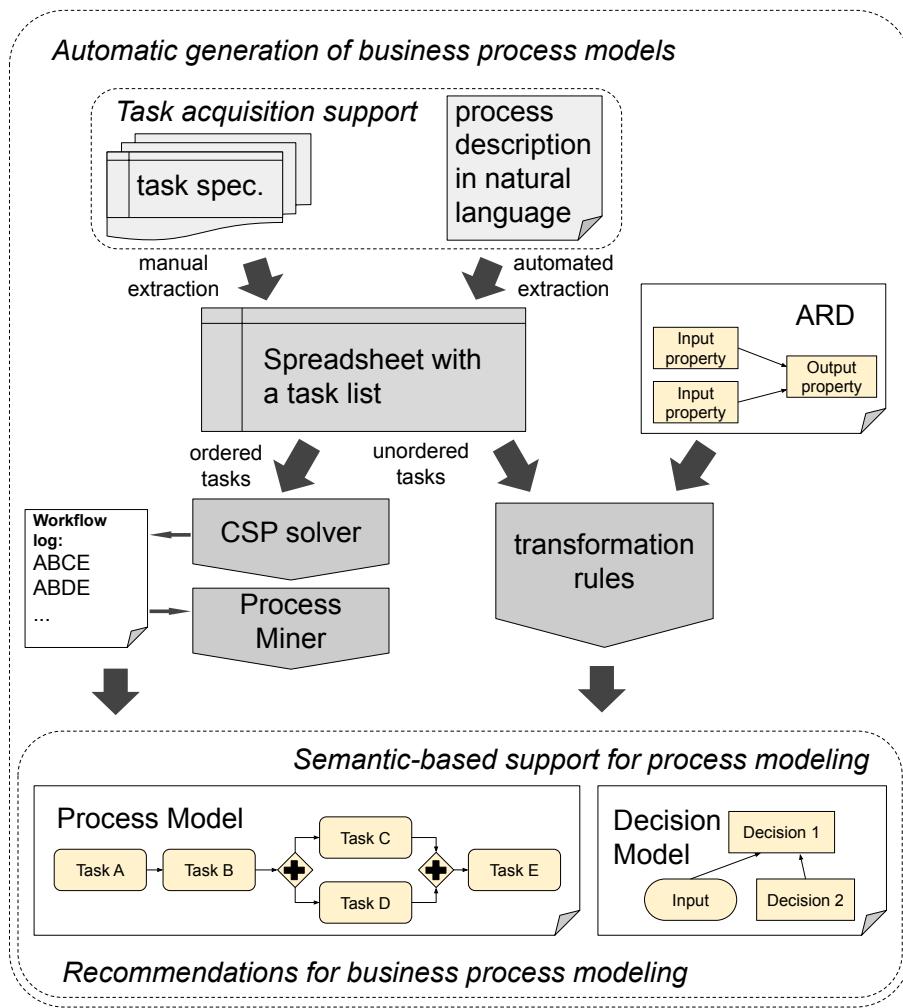
## 2 Our contribution to the AI for BPM field

The research of our group in the BPM area focuses on business process modeling, especially acquisition of models using different AI methods. The overview of our research threads is illustrated in Figure 1. Among them one can distinguish:

---

<sup>\*</sup> The paper is supported by AGH UST research grant.

- Semantic-based support for business process modeling:** As process models can be ambiguous, semantic technologies, such as ontologies, are introduced to BPM solutions [5]. They help to support intelligent functions in BPM software, like web services discovery or intelligent suggestions. Such techniques often use semantic annotations based on formally specified ontologies, what improves business process management environments. This area also includes our former research concerning recommendation techniques in business process modeling [3], especially using Bayesian networks as a recommendation tool [1], what makes process modeling faster and less error-prone.



**Fig. 1.** Overview of our research directions in AI for BPM

Among semantic-based supported methods in process modeling, we developed a knowledge acquisition method which helps process analysts with facilitating knowledge gathered from domain experts by means of analysis of similarity between different tasks within the modeled process [4]. This system enables smart merging of the data provided by different users, and results in declarative process specification that can be used as a basis for BPMN process model generation.

2. **Automatic generation of business process models:** As manual extraction of business process models from technical documentation is a time-consuming task, some of our research threads focus on automatic generation of business process models [7]. In this area, we contributed by introducing several methods of obtaining process models, such as acquiring process models from natural language description [2], from other models [6] or declarative specification of tasks [9].

In [2], we presented a concept of a novel method for extracting business process from a natural language text through intermediate process model based on the spreadsheet representation [8]. Our method is enhanced with semantic analysis of the text, which allows to filter out unnecessary content. We also proposed a method that facilitates prototyping and semi-automatic construction of the integrated process and decision models [6]. Our method supports generating business processes with decisions represented in the BPMN and DMN standards.

Another approach to obtaining process model is our participatory business process modeling method [9], that uses constraint programming and graph composition for generating a BPMN model. The method is used as a support for business analysts or process designers in visualizing the workflow without the need of designing the model explicitly in a graphical editor.

As our main research threads focus on acquiring business process models, our future works will concern developing new model acquiring methods and extending the existing ones, especially for integrated process and decision models. In particular, the following threads seem to be interesting: techniques for discovering and automating tasks, processes and rules from unstructured data, automation of exception handling, AI-enabled virtual assistants to simplify interaction with processes, as well as impact of AI technology on BPM-related standards such as BPMN, CMMN and DMN.

### 3 Summary

We gave a brief overview of our research threads in Business Process Management area that exploit various Artificial Intelligence methods. The presented research has been conducted in the Department of Applied Computer Science at AGH University of Science and Technology in Krakow. We highlighted our contribution to the AI for BPM field and provided a brief overview of our future research topics in this area.

## References

1. Bobek, S., Baran, M., Kluza, K., Nalepa, G.J.: Application of bayesian networks to recommendations in business process modeling. In: Proceedings of the Workshop AI Meets Business Processes 2013 co-located with the 13th Conference of the Italian Association for Artificial Intelligence (AI\*IA 2013), Turin, Italy, December 6, 2013. pp. 41–50 (2013)
2. Honkisz, K., Kluza, K., Wiśniewski, P.: A concept for generating business process models from natural language description. In: International Conference on Knowledge Science, Engineering and Management. pp. 91–103. Springer International Publishing, Cham (2018). [https://doi.org/10.1007/978-3-319-99365-2\\_8](https://doi.org/10.1007/978-3-319-99365-2_8)
3. Kluza, K., Baran, M., Bobek, S., Nalepa, G.J.: Overview of recommendation techniques in business process modeling. In: Proceedings of 9th Workshop on Knowledge Engineering and Software Engineering (KESE9) co-located with the 36th German Conference on Artificial Intelligence (KI2013), Koblenz, Germany, September 17, 2013. (2013)
4. Kluza, K., Kagan, M., Wiśniewski, P., Adrian, W.T., Ligeza, A.: Semantic-based support system for merging process knowledge from users. In: Mercier-Laurent, E., Owoc, M.L., Ritter, W. (eds.) Proceedings of the 7th IFIP International Workshop on Artificial Intelligence for Knowledge Management (AI4KM 2019) AI for Humans August 11th, 2019, Macao, China (2019)
5. Kluza, K., Nalepa, G.J., Ślażyński, M., Kutt, K., Kucharska, E., Kaczor, K., Łuszaj, A.: Overview of selected business process semantization techniques. In: Advances in Business ICT: New Ideas from Ongoing Research, pp. 45–64. Studies in Computational Intelligence, Springer International Publishing, Cham (2017). [https://doi.org/10.1007/978-3-319-47208-9\\_4](https://doi.org/10.1007/978-3-319-47208-9_4)
6. Kluza, K., Wiśniewski, P., Adrian, W.T., Ligeza, A.: From attribute relationship diagrams to process (BPMN) and decision (DMN) models. In: International Conference on Knowledge Science, Engineering and Management. pp. 615–627. Springer International Publishing, Cham (2019). [https://doi.org/10.1007/978-3-030-29551-6\\_55](https://doi.org/10.1007/978-3-030-29551-6_55)
7. Wiśniewski, P., Kluza, K., Jobczyk, K., Stachura-Terlecka, B., Ligeza, A.: Overview of generation methods for business process models. In: International Conference on Knowledge Science, Engineering and Management. pp. 55–60. Springer International Publishing, Cham (2019). [https://doi.org/10.1007/978-3-030-29563-9\\_6](https://doi.org/10.1007/978-3-030-29563-9_6)
8. Wiśniewski, P., Kluza, K., Kucharska, E., Ligeza, A.: Spreadsheets as interoperability solution for business process representation. *Applied Sciences* **9**(2), 345 (2019)
9. Wiśniewski, P., Kluza, K., Ligeza, A.: An approach to participatory business process modeling: BPMN model generation using constraint programming and graph composition. *Applied Sciences* **8**(9), 1428 (2018)

# Semantic Information Extraction and Knowledge Graph Analysis<sup>\*</sup>

Weronika T. Adrian<sup>1[0000–0002–1860–6989]</sup>, Marco Manna<sup>2[0000–0003–3323–9328]</sup>,  
Giovanni Amendola<sup>2[0000–0002–2111–9671]</sup>, and Rafael  
Peñaloza<sup>3[0000–0002–2693–5790]</sup>

<sup>1</sup> AGH University of Science and Technology,  
al. A.Mickiewicza 30, 30-059 Krakow, Poland,  
[wta@agh.edu.pl](mailto:wta@agh.edu.pl)

<sup>2</sup> University of Calabria, via Pietro Bucci  
Arcavacata di Rende 87036 (CS), Italy,  
[{manna,amendola}@mat.unical.it](mailto:{manna,amendola}@mat.unical.it)

<sup>3</sup> University of Milano-Bicocca,  
Piazza dell'Ateneo Nuovo, 1 - 20126, Milano, Italy  
[rafael.penaloza@unimib.it](mailto:rafael.penaloza@unimib.it)

**Abstract.** In this paper, we present our research in the field of semantic information extraction and knowledge graph analysis. In particular, we tackle the problem of a semi-automatic lexicon generation with use of knowledge graphs. We introduce the problem and resulting challenges that arise from the domains of knowledge representation and reasoning. We report the activities performed so far, summarize the obtained results and outline future work directions and perspectives.

**Keywords:** information extraction · knowledge graphs · lexicon generation · semantic resources · information systems · entity set expansion.

## 1 Introduction

Semantic Information Extraction (SIE) [1,2,6] uses methods from the Knowledge Representation domain to improve the quality of information extracted from complex documents. In particular, ontology-based- [13] Information Extraction uses lexicons, dictionaries and taxonomies to organize the data discovered from the analyzed input into ready-to-use knowledge bases. Unfortunately, building an ontology on which the information extraction process can rely is a time-consuming and error-prone task. To alleviate these problems, several approaches to semi-automatic lexicon generation have been undertaken.

One of the many ways to build a semantic schema (and in particular to populate the *classes* of the target ontology with their *instances*) is to start with sets of individuals and treat them as samples to learn from. “Growing a class” of objects from the given initial instances –called “seeds”– is known in literature

---

\* This paper is supported by AGH UST grant.

as the Entity Set Expansion (ESE) problem. The objective of ESE is, given a set of example instances that belong to the same (potentially unknown) class, to find more objects “of the same kind” [9].

The problem of Entity Set Expansion is intuitive and can be simply formulated. However, more detailed analysis reveals several challenges and sub-problems that can be basically grouped into two questions:

1. How to “understand” the seeds and recognize similarities among them so that a sort of *category description* can be formulated for them?
2. How to extend the initial set with more instances?

The problem has practical applications in both “personal” information management (cf. Google Sets, now discontinued and with the technique protected by a patent) and “enterprise” solutions e.g., in IE systems used in recruitment etc.

## 2 Entity Set Expansion from Knowledge Graphs

The problem of Entity Set Expansion has been traditionally addressed using a textual corpus to learn *patterns* in which the initial words appear and then use these patterns to extract more words found in similar contexts. This iterative process consists in alternating the discovery of textual patterns and extraction of new terms. Later on, the vast potential of the Web –that is a large collection of inter-connected and semi-structured documents– has been used to learn and expand the categories of entities. For instance, one could use the HTML markup to identify lists of items within websites and this way find sets of items that belong to the same category. This allowed to abstract from language-specific textual patterns [12]. However, the *semantics* of the links between the documents and the objects described in them was used only to a limited extent.

Currently, significant amount of knowledge is stored in so-called *knowledge graphs* [10], of which prominent examples are Google’s Knowledge Graph, Facebook’s Social Graph, BabelNet [8] or DBpedia [5]. These knowledge bases represent information about *classes*, *instances* and *relations* among them in a defined structured manner. Knowledge graphs vary in the level of formalization and scope, from encyclopedic semi-structured databases such as Wikipedia, through lexical databases like WordNet [7], up to formal ontologies defined in logic [11] — and from close domain-specific databases to interlinked Web-based resources. Our research hypothesis is that appropriate usage of (fragments of) these resources allows to:

1. better “understand” the words given as seeds in the ESE problem
2. formally define the desired class of objects to which the seed words refer
3. improve the quality of the expansion results by finding more instances of the well-defined classes efficiently.

### 3 Obtained results and further research

To overcome the limitations of existing approaches to automatic lexicon generation, we proposed to use knowledge available on the Web, specifically, stored in selected semantic resources that represent semantics of objects, their categorization and relations with other objects [4]. We used these resources to get and, in some cases, disambiguate word senses, to discover commonalities among objects represented with them and to formulate their common category. We integrated information available in the selected semantic resources to combine the strengths and minimize weaknesses of them. To reason over the integrated knowledge, we represented it with a single model called *entity network*. An entity network is a four-tuple  $\mathcal{N} = \langle Uni, Rel, Con, type \rangle$  where: (i)  $Uni$  is a set of knowledge units, both classes and instances; (ii)  $Rel$  is a set of semantic relations; (iii)  $Con \subseteq Uni \times Uni$  is a set of ordered pairs denoting that two units are connected via some (one or more) semantic relations; and (iv)  $type : Con \rightarrow (2^{Rel} \setminus \emptyset)$  is a function that assigns to each connection a set of semantic relations.

In [3], we have proposed an algorithm that traverses selected knowledge bases (BabelNet [8], WordNet [7] and Wikidata<sup>4</sup>) in a prescribed order and using the equivalence relations between the objects. The algorithm proceeds in phases: first, for the seed words, it creates an entity network out of the hypernymy relations found in the semantic resources. This network then serves to disambiguate the word senses of the seeds. In particular, the *optimal common ancestors* of the seed nodes are determined and a set of mappings from words to objects is obtained. Then, for the objects identified as the “best” senses of the seeds, another entity network is created – this time including all the known relations of the objects of interest. A common category description is inferred from the network and in the next step this description is used to query the resources for more instances that are consistent with it.

Experimental evaluation revealed the inefficiency of the Web-based tool that performs multiple queries to external knowledge bases. Unfortunately, even adopting some limitations: using only one resource (BabelNet) and its local dump resulted in unacceptable performance time. These experiments inspired on the one hand a theoretical study of the ESE problem (that is currently being carried out), and on the other raised question about certain “parametrization” of the problem, be it in the expressiveness of the description language or the depth of analysis of the used knowledge bases. In general, given the vast amount of the related research fields, we have moved our interest towards more abstract study of the problem, analysing the ESE problem given *arbitrary* knowledge base, description language and the complexity of the problem in different classes of these languages.

---

<sup>4</sup> See <https://www.wikidata.org>.

## 4 Future work

Future research directions are of both theoretical and practical nature. As for the former, we will study the complexity of building the category descriptions in different languages. As for the latter, we aim to develop a publicly available tool that would not only allow users to expand their seed list, but also give an explanation on why particular entities have been discovered. In fact, an open publicly available tool for ESE is hard to find. The pioneer Google Sets have been discontinued and the solutions behind them has been protected by a patent ever since. Few proof-of-concept tools are available, but usually not maintained or they do not provide any “explanation” why certain terms have been found.

## References

1. Adrian, W.T.: Ontology-driven Information Extraction. Ph.D. thesis, University of Calabria (July 2017)
2. Adrian, W.T., Leone, N., Manna, M.: Semantic views of homogeneous unstructured data. In: International Conference on Web Reasoning and Rule Systems. pp. 19–29. Springer (2015)
3. Adrian, W.T., Manna, M.: Navigating online semantic resources for entity set expansion. In: Proc. of PADL’18. pp. 170–185 (2018), [https://doi.org/10.1007/978-3-319-73305-0\\_12](https://doi.org/10.1007/978-3-319-73305-0_12)
4. Adrian, W.T., Manna, M., Leone, N., Amendola, G., Adrian, M.: Entity set expansion from the web via asp. In: Technical Communications of the 33rd International Conference on Logic Programming (ICLP 2017). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2018)
5. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: A nucleus for a web of open data. In: The semantic web, pp. 722–735. Springer (2007)
6. Manna, M., Oro, E., Ruffolo, M., Alviano, M., Leone, N.: The *h1ex* system for semantic information extraction. In: Transactions on Large-Scale Data-and Knowledge-Centered Systems V, pp. 91–125. Springer (2012)
7. Miller, G.A.: Wordnet: a lexical database for english. Communications of the ACM **38**(11), 39–41 (1995)
- 8.Navigli, R., Ponzetto, S.P.: Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artificial Intelligence **193**, 217–250 (2012)
9. Sarmento, L., Jijkoun, V., de Rijke, M., Oliveira, E.: "More like these": growing entity classes from seeds. In: Proc. of CIKM’07. pp. 959–962 (2007)
10. Singhal, A.: Introducing the knowledge graph: Things, not strings. Google Official Blog (May 2012), retrieved September 20, 2019.
11. Waliszko, J., Adrian, W.T., Ligęza, A.: Traffic danger ontology for citizen safety web system. In: International Conference on Multimedia Communications, Services and Security. pp. 165–173. Springer (2011)
12. Wang, R.C., Cohen, W.W.: Language-independent set expansion of named entities using the web. In: Seventh IEEE international conference on data mining (ICDM 2007). pp. 342–350. IEEE (2007)
13. Wimalasuriya, D.C., Dou, D.: Ontology-based information extraction: An introduction and a survey of current approaches (2010)

# Context-Based Inference in Technical Diagnostics

Anna Timofiejczuk<sup>1</sup>

<sup>1</sup> The Silesian University of Technology, Gliwice, Poland

<sup>2</sup>anna.timofiejczuk@polsl.pl

**Keywords:** technical diagnostics, context-based inference, classification

One of the main problems of technical diagnostics of such objects as machines or industrial processes, is the determination of their technical condition. The identification of the condition of a technical object during the stage of its use is carried out by means of diagnostic tests, which are most often carried out during normal operation of the object. The stages of these tests are: signal registration, signal analysis, signal interpretation and diagnostic inference. The purpose of the described research [1] was to develop a methodology for interpreting the results of signal analysis and diagnostic inference.

A special feature of the proposed method of inference is to discover associations that are considered. The process is carried out taking into account the context of the object's operation. The method was developed basing on the assumptions, which are related to the type of data analyzed, the method of their analysis and coding, context definition, context identification and the method of inference. A global approach to interpreting the results of the analysis of signals recorded during the operation of technical facilities has been developed. Important parts of the developed method include the methods of data coding, a method of automatically creating a database of examples underlying the assessment of identified contexts and rules, and a method of diagnostic inference based on determining diffuse assessments. The most important part of the developed method should be the identification of contexts that may occur during the operation of the object. The context identification method based on the results of signal analysis was based on the use of evolutionary algorithm. Verification studies of the developed method were based on three groups of data. The first group is data generated on the basis of mathematical models, which take into account the impact of changes in selected parameters on other parameters. The second group of signals was recorded during the operation of the laboratory stand, enabling modeling of typical rotor machine failures. During this part of the experiment, several simple states of object operation and their combinations were simulated. The third group of signals are real signals recorded during the operation of the longwall shearer.

The next stage of the research described in [2] concerned tuning parameters of inference algorithms during the training process. The problem of data optimization was also described. The second part of the research concerns the formal description and verification of detection methods and damage isolation based on the class committees using information on the context of the diagnosed device. Currently, the problem of technical diagnostics using classifiers and other classic methods is very well known and described in the literature. In the case of more complex technical measures, in

which the impact of damage does not give unequivocal symptoms, the use of single classifiers may not guarantee satisfactory results. Over the years, a number of methods have been created whose task is to combine the classifiers in committees in order to obtain the result resulting from the analysis of the base outputs of the classifiers. Such a solution usually allows for more satisfactory results, but at the same time entails a problem related to learning the classifiers and increasing their complexity, which directly affects the time needed to train the classifier or its use in the target system.

One of the solutions limiting these disadvantages is the use of a contextual feature that can significantly reduce the complexity of the classifier committee while maintaining its high efficiency. These studies are a combination of the results of works [1] and [2]. Additional elements of the proposed methods are encapsulating the training process of the classifier committee with an optimization algorithm whose purpose function allows determining the significance of individual technical states of the facility. The goal of the optimization algorithm is to fine-tune the parameters of the class of filters and feature selection algorithms used to minimize the value of the objective function. The optimization process is carried out globally, taking into account each of the classifiers separately.

1. Timofiejczuk A.: Methodology of context based reasoning in technical diagnostics. Gliwice, Politechnika Śląska, 2011 (monography, in Polish).
2. Kalisch M., Przystałka P., Timofiejczuk A.: A concept of meta-learning schemes for context-based fault diagnosis. XV International Technical Systems Degradation Conference. TSD International Conference, Liptovsky Mikulas, 30 March - 2 April 2016. Ed. J. Mączak. Faculty of Automotive and Construction Machinery Engineering. Warsaw University of Technology [et al.]. Warszawa : Polskie Naukowo-Techniczne Towarzystwo Eksplotacyjne, 2016, p. 113-114.

# Towards data-event-driven approach in ADVISOR project

Dariusz Król[0000–0002–2715–6000]

Department of Information Systems, Faculty of Computer Science and Management,  
Wrocław University of Science and Technology, Poland  
Dariusz.Krol@pwr.edu.pl

**Abstract.** The ADVISOR project addresses the problem of data-event-driven smart manufacturing operations by leveraging inherent complexity, uncertainty and dynamicity of the industrial process activities to the business success of the firm. The motivation for this research project is to provide a solution for filling the gap between a constant stream of engineering data collecting from the factory shop-floor monitoring, control systems and the higher-level management data. The general research question is: How to acquire relevant knowledge for efficient quality improvement of manufacturing processes and how to apply it to the decision-making process at operational, tactical and even strategic levels? Such fully responsive and flexible integration should facilitate effective communication between day-to-day intelligent production and strategic decision-makings at board levels. This automatic data driven transition including continuous KPI dynamics and correlation-based self-optimization into business recommendations is a shift toward an innovation-based Smart Economy. This requires implementing the various cyber-physical technologies inherent in Industry 4.0 – including advanced analytics, cloud manufacturing, AI and cognitive technologies, and new business KPIs – to connect assets and facilities, make sense of data, and digitize operations into manager's business support system.

This situation calls for (1) modelling of manufacturing resources using ontology approach to structure the engineering and management data in a sound Smart Factory Information System, (2) developing methods for identification and classification dependencies between signals generated by production units (sockets) and manufacturing efficiency indicators (KPIs defined in ISO 22400 standard) related to business KPIs, and (3) developing methods for identification and classification business KPIs and determining their dependencies with manufacturing efficiency indicators.

The specific objectives of the research project therefore are: (1) to develop a model to structure the data of a sound management information system, (2) to define a set of KPIs (ratios) to support manager's business decisions, (3) to develop an ontology to describe the interdependencies between data and the KPIs and (4) to develop methods for intelligent business decision making.

**Keywords:** Business process management · Decision making · Ontology engineering · Smart manufacturing · Industry 4.0.

NI (neuroinformatics)

# Kernel Current Source Density (kCSD) as an example of applied Machine Learning \*

Jakub M. Dzik<sup>1</sup>[0000–0003–3745–1000] ♠, Marta Bejtka<sup>1</sup>[0000–0002–4005–8464], Chaitanya Chintaluri<sup>1,2</sup>[0000–0003–4252–1608], and Daniel K. Wójcik<sup>1</sup>[0000–0003–0812–9872]

<sup>1</sup> Laboratory of Neuroinformatics, Nencki Institute of Experimental Biology of Polish Academy of Sciences, 3 Pasteur Str., 02-093 Warsaw, Poland  
<sup>2</sup> Centre for Neural Circuits and Behaviour, Department of Physiology Anatomy and Genetics, University of Oxford, Oxford, UK  
♠ Corresponding author: j.kowalski@nencki.gov.pl

**Abstract.** Kernel Current Source Density (kCSD) method uses kernel learning to estimate the sources of brain activity from measured electric potentials.

**Keywords:** Current Source Density · Electric Activity of Brain · Neuroinformatics · Machine Learning.

## 1 Introduction

### 1.1 From Electric Potential to the Underlying Brain Activity

Measurements of extracellular potential are commonly used by neuroscientists to investigate electric activity of the brain. The potential gradient results in ion motion in the extracellular space which can be related to current sources located at membranes of the active cells. The relation between the current sources and recorded potential is well established [2,4,5]. As the electric field is long range [14,6,7,12], a reconstruction of the *current source density* (CSD) gives a better insight in the underlying neural activity.

Since 1950s several methods of CSD reconstruction have been introduced [10,8,9,14,13]. In 2012 a non-parametric, kernel-based method of CSD estimation (*kernel CSD*; kCSD) was developed in our laboratory [11,3]. A major advantage of that method over its predecessors was that it allowed for nonregular setup of electrodes, which was not possible before.

Here we discuss kCSD and advocate that it is an example of a *supervised learning* algorithm.

---

\* Project funded from the Polish National Science Centre's OPUS grant (2015/17/B/ST7/04123).

## 1.2 Source Reconstruction Based on Kernel Interpolation

The simplest variant of kCSD<sup>3</sup> interpolates extracellular potential recorded by  $N$  electrodes with a *kernel function*  $K(\mathbf{x}, \mathbf{x}')$ :

$$V^*(\mathbf{x}) = \sum_{j=1}^N \beta_j K(\mathbf{x}, \mathbf{x}_j), \quad (1)$$

where  $\mathbf{x}_j$  is the location of  $j$ -th electrode. The  $\beta$  vector of coefficients is obtained by solving a linear equation:

$$\mathbf{K}\beta = \mathbf{V}, \quad (2)$$

where  $\mathbf{V}$  is the vector of potentials measured by electrodes and  $\mathbf{K}$  is a *kernel matrix* ( $\mathbf{K}_{i,j} \equiv K(\mathbf{x}_i, \mathbf{x}_j)$ ). The kernel function is defined as:

$$K(\mathbf{x}, \mathbf{x}') \equiv \sum_{k=1}^M b_k(\mathbf{x}) b_k(\mathbf{x}'), \quad (3)$$

where  $b_k : M_V \rightarrow \mathbb{R}$  is  $k$ -th of  $M$  *potential basis functions*, which represents an electric potential in space  $M_V$  generated by an associated *basis function*  $\tilde{b}_k : M_C \rightarrow \mathbb{R}$  defined in the CSD reconstruction space  $M_C$ , which may be different than  $M_V$ . For example,  $M_C$  may be a region of Euclidean three-dimensional space where the sources are, while  $M_V$  may be another region surrounding electrodes located at  $\{x_1, x_2, \dots, x_N\}$ .

From equations 1 and 2 we can write the kernel-interpolated potential function  $V^* : M_V \rightarrow \mathbb{R}$  as a linear combination of the potential basis functions:

$$V^*(\mathbf{x}) = \sum_{k=1}^M \alpha_k b_k(\mathbf{x}), \quad (4)$$

where:

$$\alpha_k = \sum_{j=1}^N \beta_j b_k(\mathbf{x}_j). \quad (5)$$

By substituting the potential basis functions with their counterparts in CSD space we obtain the reconstructed CSD function  $C^* : M_C \rightarrow \mathbb{R}$  as:

$$C^*(\mathbf{y}) = \sum_{k=1}^M \alpha_k \tilde{b}_k(\mathbf{y}). \quad (6)$$

With a *cross-kernel function* defined as:

$$\tilde{K}(\mathbf{y}, \mathbf{x}) \equiv \sum_{i=k}^M \tilde{b}_k(\mathbf{y}) b_k(\mathbf{x}). \quad (7)$$

---

<sup>3</sup> For noisy data we need to regularize which leads to approximation rather than interpolation.

the reconstructed CSD may be written in a cross-kernel form:

$$C^*(\mathbf{y}) = \sum_{j=1}^N \beta_j \tilde{K}(\mathbf{y}, \mathbf{x}_j), \quad (8)$$

which avoids explicit estimation of contributions from individual basis functions. For more details see [11,3].

## 2 kCSD as Instance-Based Machine Learning Algorithm

Note that the kernel matrix can be written as:

$$\mathbf{K} = \sum_{k=1}^M \mathbf{B}^k, \quad (9)$$

where  $\mathbf{B}_{i,j}^k \equiv b_k(x_i)b_k(x_j)$ . As matrix  $\mathbf{B}^k$  is a covariance matrix of recorded potentials originating from a *basis current source* of density given as  $X_k \tilde{b}_k(y)$  (where  $X_k$  is a random variable following a standard normal distribution  $\mathcal{N}(0, 1)$ ), the kernel matrix is also a covariance matrix of the measured potentials originating from all  $M$  basis sources (given that  $X_k$  variables are mutually independent).

The kernel matrix can also be factorized as:

$$\mathbf{K} = \Phi^T \Phi, \quad (10)$$

where  $\Phi_{k,j} \equiv b_k(x_j)$ . As through *singular value decomposition* (SVD)  $\Phi$  matrix can be further factorized as:

$$\Phi = \mathbf{U}^T \mathbf{S} \mathbf{E}, \quad (11)$$

which gives an *eigendecomposition* of the kernel matrix:

$$\mathbf{K} = \mathbf{E}^T \mathbf{S}^2 \mathbf{E}, \quad (12)$$

where rows of  $\mathbf{S} \mathbf{E}$  may be seen as *principal components* of the measured potentials originating from all  $M$  independent basis sources.

From equations 5, 2, 12 and 11 we have that the  $\alpha$  coefficient vector can be written as:

$$\alpha = \mathbf{U}^T \mathbf{W} \mathbf{V}, \quad (13)$$

where  $\mathbf{W} = \mathbf{S}^{-1} \mathbf{E}$  is a *whitening transformation* matrix for a vector of random variables of covariance matrix  $\mathbf{K}$  (as  $\mathbf{W}^T \mathbf{W} = \mathbf{K}^{-1}$ ), which transforms vector  $\mathbf{V}$  to the principal component space. As  $i$ -th row of  $\mathbf{U}$  matrix may be seen as an  $L^2$ -normalized vector quantifying similarity of base sources to the  $i$ -th principal component, kCSD may be seen as an instance-based supervised machine learning algorithm, where pairs of base functions  $b$  and  $\tilde{b}$  are a training set.

## References

1. Bédard, C., Destexhe, A.: Generalized theory for current-source-density analysis in brain tissue. *Phys Rev E Stat Nonlin Soft Matter Phys* **84**(4 Pt 1), 041909 (Oct 2011)
2. Buzsáki, G., Anastassiou, C.A., Koch, C.: The origin of extracellular fields and currents — EEG, ECoG, LFP and spikes. *Nat Rev Neurosci* **13**(6), 407–420 (Jun 2012). doi: 10.1038/nrn3241
3. Chintaluri, C., Kowalska, M., Średniawa, W., Czerwiński, M., Dzik, J.M., Jędrzejewska-Szmek, J., Wójcik, D.K.: kcsd-python, a tool for reliable current source density estimation. Preprint in: bioRxiv (2019). doi: 10.1101/708511, <https://www.biorxiv.org/content/early/2019/07/19/708511>
4. Einevoll, G.T., Kayser, C., Logothetis, N.K., Panzeri, S.: Modelling and analysis of local field potentials for studying the function of cortical circuits. *Nat Rev Neurosci* **14**(11), 770–785 (Nov 2013). doi: 10.1038/nrn3599
5. Gratiy, S.L., Halnes, G., Denman, D., Hawrylycz, M.J., Koch, C., Einevoll, G.T., Anastassiou, C.A.: From maxwell's equations to the theory of current-source density analysis. *The European journal of neuroscience* **45**, 1013–1023 (Apr 2017). doi: 10.1111/ejn.13534
6. Hunt, M.J., Falinska, M., Łęski, S., Wójcik, D.K., Kasicki, S.: Differential effects produced by ketamine on oscillatory activity recorded in the rat hippocampus, dorsal striatum and nucleus accumbens. *J Psychopharmacol* **25**(6), 808–821 (Jun 2011). doi: 10.1177/0269881110362126
7. Lindén, H., Tetzlaff, T., Potjans, T.C., Pettersen, K.H., Grün, S., Diesmann, M., Einevoll, G.T.: Modeling the spatial reach of the LFP. *Neuron* **72**(5), 859–872 (Dec 2011). doi: 10.1016/j.neuron.2011.11.006
8. Nicholson, C., Freeman, J.A.: Theory of current source-density analysis and determination of conductivity tensor for anuran cerebellum. *J Neurophysiol* **38**(2), 356–368 (Mar 1975)
9. Pettersen, K.H., Devor, A., Ulbert, I., Dale, A.M., Einevoll, G.T.: Current-source density estimation based on inversion of electrostatic forward solution: effects of finite extent of neuronal activity and conductivity discontinuities. *J Neurosci Methods* **154**(1-2), 116–133 (Jun 2006). doi: 10.1016/j.jneumeth.2005.12.005
10. Pitts, W.: Investigations on synaptic transmission. In: Cybernetics, Trans. 9th Conf. Josiah Macy, New York. pp. 159–162 (1952)
11. Potworowski, J., Jakuczun, W., Łęski, S., Wójcik, D.K.: Kernel current source density method. *Neural Comput* **24**(2), 541–575 (Feb 2012). doi: 10.1162/NECO\_a\_00236
12. Łęski, S., Lindén, H., Tetzlaff, T., Pettersen, K.H., Einevoll, G.T.: Frequency dependence of signal power and spatial reach of the local field potential. *PLoS Comput Biol* **9**(7), e1003137 (Jul 2013). doi: 10.1371/journal.pcbi.1003137
13. Łęski, S., Pettersen, K.H., Tunstall, B., Einevoll, G.T., Gigg, J., Wójcik, D.K.: Inverse Current Source Density method in two dimensions: Inferring neural activation from multielectrode recordings. *Neuroinformatics* **9**(4), 401–425 (Dec 2011). doi: 10.1007/s12021-011-9111-4
14. Łęski, S., Wójcik, D.K., Tereszczuk, J., Swiejkowski, D.A., Kublik, E., Wróbel, A.: Inverse Current-Source Density method in 3D: reconstruction fidelity, boundary effects, and influence of distant sources. *Neuroinformatics* **5**(4), 207–222 (2007). doi: 10.1007/s12021-007-9000-z

# Computational investigation of biochemical foundations of learning and memory

Ziemowit Sławiński<sup>1</sup>[0000-0002-4577-9885], Joanna Jedrzejewska-Szmk<sup>1</sup>[0000-0002-2336-0848], and Daniel K. Wójcik<sup>1</sup>[0000-0003-0812-9872]

Nencki Institute of Experimental Biology  
Polish Academy of Sciences  
Ludwika Pasteura 3, 02-093 Warsaw, Poland  
<http://neuroinflab.pl>

**Abstract.** We present a modelling study of molecular mechanisms underlying synaptic plasticity in a morphology comprised of a single dendritic spine located in the hippocampal CA1 pyramidal neuron. We focus on Spike-Timing Dependent Plasticity (STDP) protocol and its neuro-modulatory control. In agreement with experimental data, our model predicts that the timing between synaptic stimulation and induction of the action potential (AP) can modify synaptic strength. Specific neuro-modulation can switch the synaptic response to stimulus from forgetting to learning. Understanding these effects may inspire novel Artificial Neural Networks models.

**Keywords:** Neuroscience · Neural networks · Synaptic plasticity · Hippocampus · Hebbian learning Rule · LTP · LTD · STDP

## 1 Introduction

Spike-Timing Dependent Plasticity (STDP) is a biological process, which, depending on the relative timing of neural input and output, leads either to an increase of synaptic strength and synaptic growth, a process called long-term potentiation (LTP), or shrinking of the synapse reducing its weight (long-term depression, LTD). STDP eliciting experimental protocols combine synaptic stimulation with the induction of action potentials back-propagating (bAP) through the dendritic tree of the postsynaptic neuron. Typically, if the synapse is stimulated ( $t_{syn}$ ) before the action potential induction ( $t_{bAP}$ ) LTP is elicited, LTD is elicited in the opposite case. We can define  $\Delta t$ :

$$\Delta t = t_{syn} - t_{ap} \quad (1)$$

Thus  $\Delta t > 0$  leads to LTP and  $\Delta t < 0$  leads to LTD (Fig. 1). STDP-eliciting paradigms fulfil the Hebbian Postulate, which states that if two neurons fire together they wire together. Obviously, not every  $\Delta t$  leads to synaptic plasticity induction. Experiments show that for most neurons synaptic plasticity occurs when  $\Delta t$  is in the range from -50 ms to 50 ms.

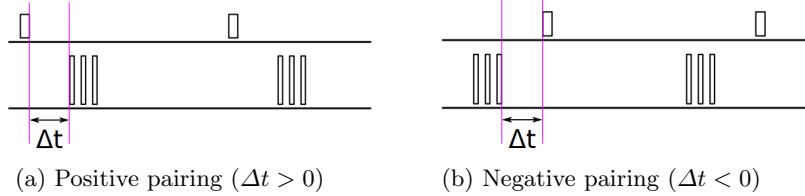


Fig. 1: Positive and negative STDP paradigms

Neuromodulators such as dopamine, acetylcholine or noradrenaline affect both the magnitude and the polarity of synaptic plasticity [4, 2, 3, 7]. For example, bath application of dopamine (DA) accompanying negative STDP pairing ( $\Delta t < 0$ ) can switch LTD to LTP.

## 2 Methods

The hippocampus is a part of the brain associated with memory consolidation and seems a natural choice to study synaptic plasticity. Most studies of the hippocampal synaptic plasticity target pyramidal neurons located in the CA1 sub-region. Pyramidal neurons form excitatory connections between different brain regions or subregions. Excitatory synapses (connections) involve dendritic spines – mushroom-like protrusions that typically receive input from a single axon located on dendrites. The out-most part of the spine is called Post-Synaptic Density (PSD) and ensures that receptors are close to the axon terminals (of the presynaptic cells).

Here we created two distinct modes of simulation, A) a multi-compartmental model of the electrical activity of a single CA1 pyramidal neuron (Fig. 2A); B) a multi-compartmental model of biochemical dynamics of a single dendritic spine (Fig. 2B). The first model contains explicit dendritic spines and spatially distributed ion channels: sodium (Na), potassium (K) and calcium (Ca) channels [5]. The neuron model was implemented in the NEURON simulation environment (Fig. 2.A). The second model is a biochemical model of the dendritic spine which implements stochastic reaction-diffusion equations to track molecular interactions based on known reaction constants for all modelled molecules. The dendritic spine model is implemented in the NeuroRD environment [6] (Fig. 2.B).

From the molecular point of view maintaining changes of synaptic strength for more than 2 h requires synthesis of new proteins, which entails activation of extracellular signal-regulated kinase (ERK) via distinct signalling pathways. The biochemical pathways model comprises Ca-activated pathways, Dopamine receptor (D1R) and beta-adrenergic receptor ( $\beta$ AR) activated pathways, and cAMP-activated pathways (Fig. 3).

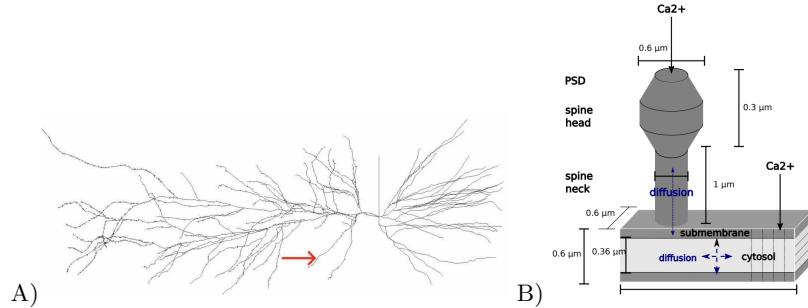


Fig. 2: A) Pyramidal neuron simulated in the NEURON environment. Red arrow points to the particular spine where we collect data for further biochemical simulations. B) Morphology of the dendritic spine simulated in the NeuroRD environment. Ca dynamics is collected from the NEURON simulation and injected to the biochemical model created in NeuroRD.

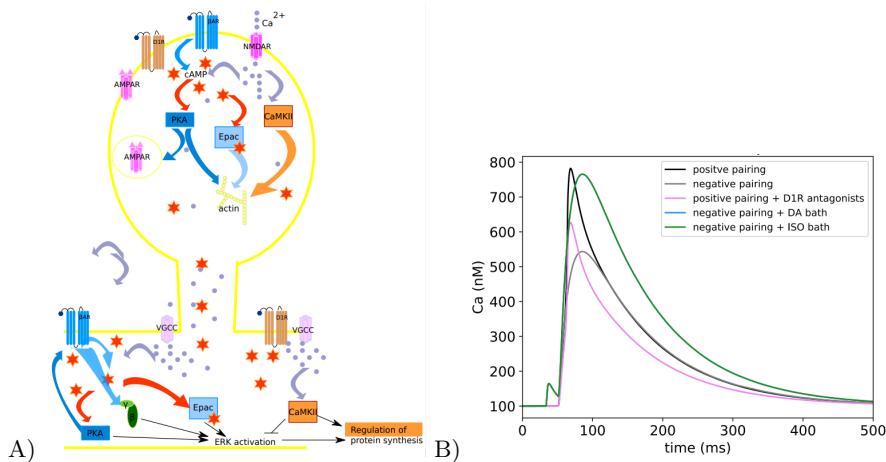


Fig. 3: A) Pathways upstream of ERK- and protein synthesis. B) Neuromodulation changes Ca dynamics in the spine: D1R antagonist — D1R receptor activator; DA bath — application of dopamine (D1R receptor activator) before the stimulation; ISO bath — application of isoproterenol ( $\beta$ AR receptor activator) before the stimulation. Green and blue curves overlap completely on the graph which means that induction of STDP negative pairing with D1R or  $\beta$ AR activation leads to similar calcium dynamics.

### 3 Results

Molecular activity is frequently redundant: many pathways lead to activation of the same molecular target. We believe that there is a set of key molecules whose collective dynamics can describe ERK activation and as a consequence allow

predicting polarity and induction of synaptic plasticity. Conceptually, this may be compared to the eigendecomposition of a matrix. Key molecules and their nonlinear dynamics probably vary with brain region, cell type, and learning paradigm which we are investigating. For synaptic plasticity of CA1 pyramidal neurons, literature provides clues of the identity of these molecules and our simulations suggest what their dynamics might look like.

Based on the previous studies [6] we suspect that synaptic plasticity can be described as a function of activity of 4 proteins: calcium-calmodulin activated kinase II (CaMKII), two cAMP targets: protein kinase A (PKA) and exchange protein directly activated by cAMP (epac) and  $\beta\gamma$  subunit of inhibitory G-protein (Gi), as well as the dynamics of two ions: sodium and calcium. As a result, we expect to get two distinct behaviours, which we call *signatures* of LTP and LTD, respectively. To our knowledge, only linear models of the plasticity signature were proposed [6, 1].

To better understand those key behaviours we investigated 6 different STDP paradigms, which have been shown experimentally to elicit LTP or LTD. In agreement with experimental results, our simulations show that positive pairing ( $\Delta t = 10ms$ ) leads to molecular dynamics which can be associated with LTP and negative pairing ( $\Delta t = -20ms$ ) leads to the dynamics which can be associated with LTD.

Activation of D1R and  $\beta$ AR leads to elevation of phosphorylation of the NMDA receptors and in consequence increased calcium currents. As a result, spine calcium for positive pairing and negative pairing accompanied by dopamine (D1R agonist) or isoproterenol ( $\beta$ AR agonist) result in similar molecular dynamics (Fig. 3.B) leading to LTP.

#### 4 Conclusions

Synaptic plasticity is a complex and redundant process that involves many biochemical pathways. The signatures of LTP and LTD may bring substantial insight into plasticity-related dynamics of synapses. In consequence, those signatures can affect how we build bio-inspired artificial neural networks. Accurate modelling of such signatures seems crucial for our understanding of how neurons learn. Nevertheless, there are still other essential areas to investigate before we would be able to faithfully recreate a bio-inspired artificial neural network. One is subthreshold dendritic depolarization. In most cases, this process takes place without the postsynaptic neuron generating action potentials. Most synaptic inputs create some level of depolarization in the nearby spines and synapses. So to some extent, we can say that even without induction of an action potential, synapses can learn during fluctuations of neural activity. Second, we must understand excitatory and inhibitory balance in dendritic compartments. Excitation and inhibition seem to be complementary for a particular distance on the dendrite. After activation of excitatory synapses, we see a higher inhibition within  $3\mu m$  for other excitatory activations. Outside that zone, inhibition is reduced which may produce another compartment of excitation.

## References

1. Blackwell, K.T., Salinas, A.G., Tewatia, P., English, B., Hellgren Kotaleski, J., Lovinger, D.M.: Molecular mechanisms underlying striatal synaptic plasticity: relevance to chronic alcohol consumption and seeking. *European Journal of Neuroscience* **49**(6), 768–783 (2019)
2. Brzosko, Z., Schultz, W., Paulsen, O.: Retroactive modulation of spike timing-dependent plasticity by dopamine. *Elife* **4**, e09685 (2015)
3. Brzosko, Z., Zannone, S., Schultz, W., Clopath, C., Paulsen, O.: Sequential neuro-modulation of hebbian plasticity offers mechanism for effective reward-based navigation. *Elife* **6**, e27756 (2017)
4. Foncelle, A., Mendes, A., Jedrzejewska-Szmk, J., Valtcheva, S., Berry, H., Blackwell, K.T., Venance, L.: Modulation of spike-timing dependent plasticity: towards the inclusion of a third factor in computational models. *Frontiers in computational neuroscience* **12**, 49 (2018)
5. Graham, B.P., Saudargiene, A., Cobb, S.: Spine head calcium as a measure of summed postsynaptic activity for driving synaptic plasticity. *Neural computation* **26**(10), 2194–2222 (2014)
6. Jedrzejewska-Szmk, J., Luczak, V., Abel, T., Blackwell, K.T.:  $\beta$ -adrenergic signaling broadly contributes to LTP induction. *PLoS Computational Biology* **13**(7), e1005657 (7 2017). <https://doi.org/10.1371/journal.pcbi.1005657>, <https://dx.plos.org/10.1371/journal.pcbi.1005657>
7. Zhang, J.C., Lau, P.M., Bi, G.Q.: Gain in sensitivity and loss in temporal contrast of stdp by dopaminergic modulation at hippocampal synapses. *Proceedings of the National Academy of Sciences* **106**(31), 13028–13033 (2009)

ML (machine learning)

# Boolean Biclustering Review and Perspectives\*

Marcin Michalak<sup>[0000–0001–9979–8208]</sup>

Silesian University of Technology, ul. Akademicka 16, 41-100 Gliwice, Poland  
[Marcin.Michalak@polsl.pl](mailto:Marcin.Michalak@polsl.pl)

**Abstract.** The Boolean reasoning based approach to biclustering is a new way of expressing biclustering requirements. Due to its mathematical roots, all proposed solutions have strong backgrounds in their mathematical proofs. This paper summarizes the developed algorithms and presents the perspectives of future works in this area.

**Keywords:** Boolean reasoning · Biclustering · Prime implicants.

## 1 Introduction

Biclustering is a way of analyzing two-dimensional homogeneous data, proposed by Hartigan in 1970' [1]. A bicluster is an ordered pair of a subset of rows and a subset of columns of the given input matrix. The intersection of these subsets defines a submatrix. The cells of this submatrix fulfill the assumed criteria of equality, comparability or even incomparability. In this paper a review of a new approach to biclustering — Boolean reasoning based — is presented and new challenging issues in this area are invoked.

## 2 Proved Approaches

The research carried out in Boolean reasoning based biclustering brought many satisfactory results. It was possible to prove in the mathematical way that prime implicants of a well defined Boolean function, which represents the data, correspond directly to inclusion-maximal biclusters in the data. The definition of the Boolean function is strongly connected with the data type and the demanded properties of the obtained biclusters. The following examples will provide a better look of such a biclustering approach.

Let us consider two matrices as they are presented in Table 1. They consist of three rows and three columns.

**Discrete Biclustering** Let us define the discernibility function in the way as follows ( literals coding rows and columns are denoted in the same way as the row and cell label,  $B$  - set of rows,  $X$  - set of columns):

$$f_M = \bigwedge (a \vee b \vee x) \wedge \bigwedge (c \vee y \vee z) \text{ where } a, b, c \in B \quad x, y, z \in X$$

---

\* The work was carried out within the statutory research project of the Institute of Informatics, BK-204/RAU2/2019.

**Table 1.** Two sample matrices: discrete (left) and binary (right).

|   | a | b | c |   | a | b | c |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 2 | 1 | 1 | 0 | 1 |
| 2 | 1 | 1 | 0 | 2 | 1 | 0 | 0 |
| 3 | 1 | 1 | 1 | 3 | 1 | 1 | 1 |

such that ( $a(x) = x(a)$  = value of the  $a^{th}$  row and  $x^{th}$  column):

$$\forall_{a,b,c,x,y,z} \quad (a(x) \neq b(x) \wedge a \neq b) \vee (c(y) \neq c(z) \wedge y \neq z)$$

It was proved that prime implicants of such defined function code inclusion–maximal exact biclusters: a bicluster contains such rows and columns whose corresponding literals are not present in the prime implicant. The Boolean formula that corresponds to the data in the discrete matrix from Table 1 is as follows:

$$\begin{aligned} f_M = & (1 \vee a \vee b) \wedge (1 \vee a \vee c) \wedge (1 \vee b \vee c) \wedge (2 \vee a \vee c) \wedge (2 \vee b \vee c) \wedge \\ & \wedge (b \vee 1 \vee 2) \wedge (b \vee 1 \vee 3) \wedge (c \vee 1 \vee 2) \wedge (c \vee 1 \vee 3) \wedge (c \vee 2 \vee 3) \end{aligned} \quad (1)$$

and its form that contains only prime implicants looks as follows:

$$f_M = (1 \wedge 2) \vee (1 \wedge c) \vee (b \wedge c) \vee (1 \wedge 3 \wedge a \wedge b) \vee (2 \wedge 3 \wedge a \wedge c) \vee (2 \wedge 3 \wedge a \wedge b) \quad (2)$$

Let us start from the interpretation of the second prime implicant  $1 \wedge c$ : the missing literals are 2, 3 and  $a, b$ , so that means that the bicluster  $(\{2, 3\}, \{a, b\})$  is the inclusion–maximal bicluster in this matrix. As we can see this bicluster contains only cells with the value 1 (so it is the exact one) and neither a row or column can be added without violating such a restriction (the bicluster is inclusion–maximal). The interpretation of other prime implicants is analogous.

**Binary Biclustering** In the case of binary data biclustering the definition of Boolean function is simplified: if we are looking for biclusters of zeros we just code all cells with ones in the formula (and vice versa). Let  $\square \in \{0, 1\}$ , then if we are looking for biclusters of  $\neg \square \in \{0, 1\} \setminus \{\square\}$ , the appropriate Boolean function is defined as follows:

$$f_{Mb(\square)} = \bigwedge (a \vee x), \quad a \in B, x \in X, a(x) = \square \quad (3)$$

The formula for the binary data from Table 1 that codes zeros (and helps to find biclusters of ones) takes the following form:

$$f_{Mb(0)} = (1 \vee b) \wedge (2 \vee b) \wedge (2 \vee c) = (1 \wedge 2) \vee (2 \wedge b) \vee (b \wedge c) \quad (4)$$

The interpretation of prime implicants remains the same: the prime implicant  $1 \wedge 2$  codes the bicluster of ones that is the whole last row of the matrix (the bicluster contains last row and all columns). It is very important to notice that the previous operation — finding inclusion–maximal exact biclusters in the discrete data — would become much faster if it is decomposed to several issues of finding biclusters in the binary data. More details (theorems and proofs) of such an approach are described in [4].

**Heuristic Approach** The presented two approaches try to find all inclusion–maximal biclusters in the data. But sometimes it is not so important to find all biclusters (to cover all of considered cells of specific values). For that reason the modification of the classical Johnson’s heuristic for finding prime implicant approximations was developed. In the classical approach the frequency of literals in formula clauses was analyzed. The required modification takes into consideration two classes of literals: frequencies of row literals and column literals are considered separately. This avoids generating biclusters that are empty: an empty bicluster contains all rows and no columns or vice versa. The modification of Johnson’s strategy is described in details in [3].

**Similarity Biclustering** Typically, finding biclusters in the continuous data means finding subregions of the matrix whose cells have similar values: e.g. the maximal bicluster values difference does not exceed an assumed level. Such a condition may also be described in terms of Boolean reasoning: the proper Boolean function is defined as follows:

$$f_\sigma = \bigvee (a \vee b \vee x \vee y) \text{ where } a, b \in B; x, y \in X; (a \neq b) \vee (x \neq y) \quad (5)$$

such that  $|a(x) - b(y)| > \sigma$  where  $\sigma$  is the maximal assumed difference between bicluster values (the bicluster range). Also this time, the interpretation of prime implicants of such defined Boolean formula remains the same.

**Dissimilarity Biclustering** However, it is very intuitive and simple to revert the condition from the previous section:  $|a(x) - b(y)| \leq \sigma$ . The interpretation of the obtained biclusters (from the formula prime implicants) remains the same as in previous cases. However, in such an approach a found bicluster has no pair of different cells whose absolute difference is lower than  $\sigma$ . The proofs of correctness and maximality of such approaches can be found in [5].

**Centre–Based Biclustering** In the paper [5] it was also proved that it is possible to search all sensible similarity (and dissimilarity) biclusters with just one formula. All possible levels of (dis)similarity are derived from all possible absolute differences between any two cells in the data. But what if we are only interested in biclusters whose values gather around some well defined value with the given tolerance level? This leads to the definition of the centre–based bicluster. Let us assume a centre value  $\mu$  and a tolerance level  $t$ . We want to obtain a bicluster whose all values fulfill a criterion  $|a(x) - t| \leq \mu$ . It occurs that the Boolean formula of the form as follows gives the solution as before:

$$f_{t,\mu} = \bigwedge (x \vee a), \quad a \in B, x \in X, \text{ such that } |a(x) - t| > \mu$$

Proofs and perspectives of such an approach application are described in [2].

### 3 Conclusions and Challenges

All provided definitions of biclusters and Boolean functions that are used to induce them confirm the correctness of the Boolean reasoning approach to biclustering. However, the obtained results and theorems do not exhaust the possible fields of application.

In the paper [5] the methods of finding all biclusters of an assumed level of similarity and all sensible biclusters in the continuous data were presented. However, it may become interesting to search such biclusters, whose range of similarity has its lower and upper bound. That would help to remove small but quite similar biclusters (a small number of cells but also with small differences between them) and of course to remove big biclusters of very high range. Such a limitation would have smaller computational complexity than the exhaustive search of all biclusters, which requires further postprocessing.

Another challenge deals with the generalization of Boolean biclustering for more than two-dimensional data. It is expected that similar theorems that join prime implicants of a Boolean function with tri-, tetra- or any- clusters can be proved.

Due to the high computational complexity of Boolean formula processing it is worth to consider some parallelized approaches. The parallelization should be considered in two aspects: parallel multiplication of clauses pairs and parallel simplification of the multiplication result (application of absorption laws). A smart parallel version of formula processing should take into consideration that initially a big number of short clauses are multiplied (multiplications should be then parallelized) while in the end a small number of very long clauses are multiplied (clause simplification should be then parallelized).

Concluding, the proposed approach for biclustering — based on Boolean reasoning — brings interesting solutions for many definitions of bicluster and for many types of the input data. Nevertheless, there are still many interesting issues in this area that can be defined, solved and mathematically proved.

### References

1. Hartigan, J.A.: Direct Clustering of a Data Matrix. *Journal of the American Statistical Association* **67**(337), 123–129 (1972)
2. Michalak, M.: Induction of Centre-Based Biclusters in Terms of Boolean Reasoning. *Advances in Intelligent Systems and Computing* **1061**, 239–248 (2020)
3. Michalak, M., Jaksik, R., Ślęzak, D.: Heuristic Search of Exact Biclusters in Binary Data. *International Journal of Applied Mathematics and Computer Science* (2019/20 (to appear))
4. Michalak, M., Ślęzak, D.: Boolean Representation for Exact Biclustering. *Fundamenta Informaticae* **161**(3), 275–297 (2018)
5. Michalak, M., Ślęzak, D.: On Boolean Representation of Continuous Data Biclustering. *Fundamenta Informaticae* **167**(3), 193–217 (2019)

# New Methods of Generating Random Parameters in Feedforward Neural Networks with Random Hidden Nodes<sup>\*</sup>

Grzegorz Dudek<sup>[0000–0002–2285–0327]</sup>

Electrical Engineering Department, Czestochowa University of Technology,  
Czestochowa, Poland  
`dudek@el.pcz.czest.pl`

**Abstract.** The standard method of generating random parameters in randomized neural network learning selects randomly hidden node weights and biases both from the same fixed interval regardless of the data scope and activation function type. This leads to the poor approximation property of the network. Recently more sophisticated methods have been developed which treat weights and biases separately. They distribute the sigmoids inside the input hypercube and adjust their slopes to the target function complexity using different approaches. These lead to improvement in approximation performance of the network and allow us to control the generalization degree of the model.

**Keywords:** Feedforward neural networks · Neural networks with random hidden nodes · Randomized learning algorithms.

## 1 Introduction

In feedforward neural networks (FNNs) the weights are learned iteratively from data using a gradient descent method. Due to a layered structure of the network and nonconvex character of the optimization problem, the gradient-based learning is time consuming, sensitive to the initial settings and leads to the local optima of the error function. In randomized learning the parameters of the hidden neurons are selected by random and stay fixed. The only parameters which are learned are the weights of output neurons. The optimization problem becomes convex, and to solve it the standard linear least-squares can be used [1]. This significantly speeds up the learning process and simplifies its implementation.

The single-hidden-layer FNN with random hidden nodes has universal approximation capability when the random parameters are selected from a symmetric interval according to any continuous sampling distribution [2]. The open questions are: what should be the interval bounds and the parameter distribution? These are the most important research gaps in FNN randomized learning [3]. The interval widely used in practice,  $[-1, 1]$ , is misleading and many authors

---

<sup>\*</sup> Supported by Grant 2017/27/B/ST6/01804 from the National Science Centre, Poland.

criticize it as devoid of scientific justification, independent on data and activation function (AF) type. It is common practice recently to optimize this interval by looking for its bounds [4].

In this work we present three new approaches for generating FNN random parameters which have been developed recently. In these approaches the weights and biases of the hidden nodes are treated separately, due to their different functions. The biases, which are related to the AF location in space, are determined in such a way that the steepest AF fragments, which are the most useful for modeling the target function fluctuations, are introduced into the input hypercube (IH). The weights, related to the AF slopes, are selected randomly using different approaches, taking into account the TF complexity.

## 2 Improved Methods of Generating Random Parameters

In this study we consider sigmoids as the AFs of hidden nodes:

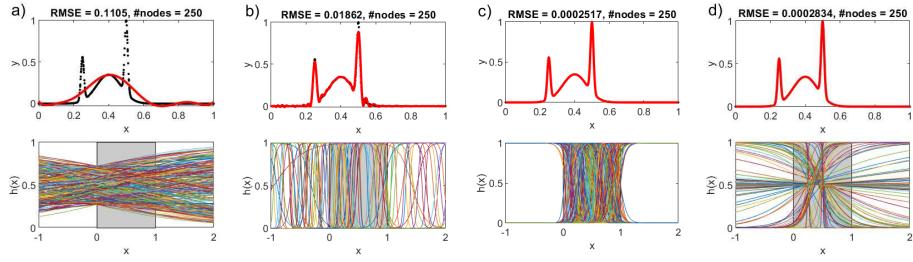
$$h(\mathbf{x}) = 1 / (1 + \exp(-(\mathbf{a}^T \mathbf{x} + b))) \quad (1)$$

where weights  $\mathbf{a} = [a_1, \dots, a_n]^T$  decide about sigmoid slopes in different directions and bias  $b$  decides about a shift of the sigmoid.

A linear combination of the sigmoids with coefficients  $\beta$  (output weights selected using a linear least-squares method) gives a fitted function which approximates TF. According to the standard approach, the weights and biases are selected both from the uniform distribution over the same fixed symmetrical interval,  $[-u, u]$ . As shown in [5], a drawback of this method is that many of the sigmoids have their steepest fragments, which are around their inflection points, outside of the IH. So, they cannot be used for modeling the TF nonlinearities. Because the interval is common for biases and weights, it is difficult to select it optimally for both parameters in the same time.

Fig. 1 shows an example of single-variable function approximation when using the standard approach with the fixed intervals:  $[-1, 1]$  (a) and  $[-100, 100]$  (b). The bottom charts show the sigmoids whose linear combination forms the function fitting data (shown as a solid line in the upper charts). For  $[-1, 1]$  case, the sigmoids are flat and their distribution in the input interval  $[0, 1]$  (shown as a grey field) does not correspond to the TF fluctuations. This results in a very weak fit. For  $[-100, 100]$  case, the sigmoids are steeper but many of them have their steepest fragments outside of the input interval. So, many of them are wasted. Moreover, the insufficient number of steep sigmoids in the input interval causes fluctuations of the fitted curve on the flat parts of the TF.

To improve the performance of the model in [5] the rsM method have been proposed which distributes the sigmoids across the IH and adjusts their slopes to the TF fluctuations. It takes into account the IH location and activation function type. According to this method the weights of the  $i$ -th hidden node for  $n$ -dimensional input are calculated as follows:



**Fig. 1.** Fitted curves and distribution of the hidden node sigmoids for: the standard method with interval  $[-1, 1]$  (a) and  $[-100, 100]$  (b), RARSM (c) and D-DM (d).

$$a_{i,j} = \Sigma_i \zeta_j / \sum_{l=1}^n \zeta_l, \quad j = 1, 2, \dots, n \quad (2)$$

where  $\zeta_1, \zeta_2, \dots, \zeta_n \sim U(-1, 1)$  are i.i.d. numbers and  $\Sigma_i$  is the sum of weights of the  $i$ -th node, which is randomly chosen from the interval:

$$|\Sigma_i| \in [\ln((1-r)/r), s \cdot \ln((1-r)/r)] \quad (3)$$

where  $r \in (0, 0.5)$  and  $s > 1$  are the parameters controlling the sigmoid slopes.

The biases of the hidden nodes are determined setting the inflection points of the sigmoids (around these points the sigmoids are steepest) at some points  $x^*$  randomly selected from the IH or randomly chosen from the training set:

$$b_i = -\mathbf{a}_i^T \mathbf{x}^* \quad (4)$$

Another method of improving the randomized learning performance was proposed recently in [6] – random angle, rotation and shift method, RARSM. Firstly, it randomly chooses the slope angles of the sigmoids from an interval adjusted to the TF complexity. Then, the activation functions are randomly rotated around the y-axis and finally, they are distributed across the IH according to data distribution. Their biases are calculated from (4) and weights from:

$$a_{i,j} = -4a'_{i,j}/a'_{i,0}, \quad j = 1, 2, \dots, n \quad (5)$$

where  $a'_{i,j}$  are components of the normal vector  $\mathbf{n}$  to the hyperplane, which is tangent to the sigmoid at their inflection points.

To determine the normal vector  $\mathbf{n}$ , we randomly select its slope angle  $\alpha \in (\alpha_{min}, \alpha_{max})$ , where  $\alpha_{min}$  and  $\alpha_{max}$  are adjusted to the TF. The sigmoid rotation is determined by selecting randomly normal vector components  $a'_1, \dots, a'_n \sim U(-1, 1)$  and calculating the component  $a'_0$  from:

$$a'_0 = (-1)^c \sqrt{(a'_1)^2 + \dots + (a'_n)^2} / \tan \alpha \quad (6)$$

where  $c \sim U\{0, 1\}$ .

To fit the sigmoids to the data more closely, in [7] a data-driven randomized learning have been proposed (D-DM). According to it, for each node an input space region is selected by choosing randomly one of the training points  $\mathbf{x}^*$ . Then the hyperplane  $T$  is fitted to  $\mathbf{x}^*$  and its  $k$  nearest neighbors. Assuming that  $T$  is tangent to the sigmoid at its inflection points, we determine the weights as:

$$a_{i,j} = 4a''_{i,j}, \quad j = 1, 2, \dots, n \quad (7)$$

where  $a''_{i,j}$  are the hyperplane coefficients. The bias of the sigmoid is determined from (4), so that its inflection point is in  $\mathbf{x}^*$ .

Fig. 1 (c) shows the sigmoid distribution in the input interval for RARSM (similar distribution we obtain for rsM). Note that all sigmoids have their inflection points in the input interval and their slopes are adjusted to the TF. This results in decreasing of the fitting error to 0.00025. Similar error level we get when using D-DM. In this case the sigmoid slopes are adjusted to the TF fragments locally, so their slopes change across the input interval (see Fig. 1 (d)).

### 3 Conclusions

In this work we present three new approaches for generating FNN random parameters, which have been developed recently. According to them, the weights and biases of the hidden nodes are treated separately, due to their different functions. The biases, which are related to the activation function location, are determined in such a way that the steepest activation function fragments, which are the most useful for modeling the TF fluctuations, are introduced into the IH. The weights, related to the activation function slopes, are selected randomly taking into account the TF fluctuations. These new approaches improve the performance of randomized learning and lead to the more compact FNN architecture.

### References

1. Principe, J., Chen, B.: Universal approximation with convex optimization: Gimmick or reality? *IEEE Comput Intell Mag* **10**, 68–77 (2015)
2. Husmeier, D.: Random vector functional link (RVFL) networks. In: Neural Networks for Conditional Probability Estimation: Forecasting Beyond Point Predictions, chapter 6, 87–97, Springer-Verlag London (1999)
3. Weipeng, C., Wang, X., Ming, Z., Gao, J.: A review on neural networks with random weights. *Neurocomputing* **275**, 278–287 (2018)
4. Li, M., Wang, D.: Insights into randomized algorithms for neural networks: Practical issues and common pitfalls. *Information Sciences* **382–383**, 170–178 (2017)
5. Dudek, G.: Generating random weights and biases in feedforward neural networks with random hidden nodes. *Information Sciences* **481**, 33–56 (2019)
6. Dudek, G.: Improving randomized learning of feedforward neural networks by appropriate generation of random parameters. *IWANN 2019, LNCS*, vol. 11506, pp. 517–530. Springer, Heidelberg (2019).
7. Dudek, G.: Data-Driven Randomized Learning of Feedforward Neural Networks. arXiv:1908.03891v1 (2019).

# Solving Inconsistencies of the Perfect Clustering Concept

Robert A. Kłopotek<sup>1</sup> and  
Mieczysław A. Kłopotek<sup>2</sup>

<sup>1</sup> Faculty of Mathematics and Natural Sciences. School of Exact Sciences,  
Cardinal Stefan Wyszyński University in Warsaw, Poland

r.kłopotek@uksw.edu.pl

<sup>2</sup> Institute of Computer Science,  
Polish Academy of Sciences, Warsaw, Poland  
mieczysław.kłopotek@ipipan.waw.pl

Ackerman and Dasgupta [1] study clusterability properties of incremental clustering algorithms. They introduce an incremental version of very popular  $k$ -means algorithm (for an extensive overview of  $k$ -means versions see [3]).

They introduced the *perfect clustering* with the property that the smallest distance between elements of distinct clusters is larger than the distance between any two elements of the same cluster. They demonstrate that there exists an incremental algorithm discovering the *perfect clustering* that is linear in  $k$  with respect to space. But their incremental (*sequential*)  $k$ -means fails to do so.

Their case study is interesting because it demonstrates that the cluster shape plays a role - each cluster has to be enclosed into a convex envelope. The problem of incremental  $k$ -means is caused by the fact that this envelope is not ball-shaped.

**Data:** the data points  $\mathbf{x}_i$ ,  $i = 1, \dots, m$ , the required number of clusters  $k$

**Result:**  $T$  - the set of cluster centres

Set  $T = (t_1, \dots, t_k)$  to the first  $k$  data points ;

Initialize the counts  $n_1, n_2, \dots, n_k$  to 1 ;

**while** any data point unvisited **do**

    Acquire the next example,  $t_{k+1}$ . Set  $n_{k+1} = 1$  ;

**if**  $t_i$  is the closest centre to  $t_j$ ,  $j \neq i$  **then**

        Replace  $t_i = (t_i n_i + t_j n_j) / (n_i + n_j)$ , thereafter  $n_i = n_i + n_j$  ;

**if**  $j \neq k + 1$  **then**

            | replace  $t_j = t_{k+1}$ ,  $n_j = n_{k+1}$ ;

            | **end**

        | **end**

    | **end**

**end**

**Algorithm 1:** Sequential (incremental)  $k$ -means, our modification

Let us discuss at this point a bit the notions of *perfect separation*. In their Theorem 4.4. Ackerman and Dasgupta [1] show that the incremental  $k$ -means algorithm, as introduced in their Algorithm 2.2 , is not able to cluster correctly

data that is *perfectly clusterable* (their Definition 4.1). The reason is quite simple. The perfect separation refers only to separation of data points, and not to points in the convex hull of these points. But during the clustering process, the candidate cluster centres are moved in the convex hulls, so that they can occasionally get too close to data points of the other cluster. To avoid this effect, let us introduce the concept of *perfect-ball-separation*:

**Definition 1.** *We shall say that clusters centred at  $A$  and  $B$  and enclosed in balls centred at  $A, B$  and with radius  $\rho_{AB}$  each are nicely ball-separated, if the distance between  $A, B$  is at least  $4\rho_{AB}$ . If all pairs of clusters are nicely ball separated with the same ball radius, then we shall say that they are perfectly ball-separated.*

We demonstrated elsewhere [2] that

**Theorem 1.** *If the distance between any two cluster centres  $A, B$  is at least  $4\rho_{AB}$ , where  $\rho_{AB}$  is the radius of a ball centred at  $A$  and enclosing its cluster (that is cluster lies in the interior of the ball) and it also is the radius of a ball centred at  $B$  and enclosing its cluster, then once each cluster is seeded the clusters cannot lose their cluster elements for each other during  $k$ -means-random and  $k$ -means++ iterations.*

Under the *perfect-ball-separation* as introduced here their incremental  $k$ -means Algorithm 2.2. (Sequential  $k$ -means) will discover the structure of the clusters after a modification (Algorithm 1):

|  |
|--|
| <b>Data:</b> $T = (t_1, \dots, t_k)$ be the resulting set of cluster centres from the<br>Algorithm 1.<br><b>Result:</b> Clusterability decision<br>Initialize the furthest neighbours $f_1, f_2, \dots, f_k$ with $t_1, t_2, \dots, t_k$ respectively;<br><b>while</b> any data point unvisited <b>do</b><br>Acquire the next example, $x$ . ;<br><b>if</b> $t_i$ is the closest centre to $x$ and $x$ is further away from $t_i$ than $f_i$ <b>then</b><br>  Replace $f_i$ with $X$ ;<br>  <b>end</b><br><b>end</b><br>Compute distances between corresponding $t_i$ and $f_i$ , pick the highest one, ;<br>Compute distances between each pair $t_i, t_j$ and pick the lowest one. ;<br><b>if</b> the latter is 4 times or more higher than the former one <b>then</b><br>  We got a perfect ball clustering<br><b>else</b><br>  Perfect ball clustering was not found<br><b>end</b> |
|--|

**Algorithm 2:** Sequential  $k$ -means, our modification - second pass

The reason is as follows. Perfect ball separation ensures that there exists an  $r$  of the enclosing ball such that the distance between any two points within the

same ball is lower than  $2r$  and between them is bigger than  $2r$ . So whenever Ackerman's incremental  $k$ -mean merges two points, they are the points of the same ball. And upon merging the resulting point lies again within the ball.

**Theorem 2.** *The incremental  $k$ -means algorithm will discover the structure of perfect-ball-clustering.*

**Data:** The set  $S$  of data points  $t$   
**Result:** The set of candidating cluster centres  
 Run single linkage on  $S$  to get a tree (distances between  $t$  are used) ;  
 Assign each leaf node the corresponding data point ;  
 Moving bottom-up, assign each internal node the ;  
 $n = n_L + n_R$ ,  $t = (t_L n_L + t_R n_R)/n$ , L,R indicating left and right child.;  
**Return** all points at distance  $< k$  from the root;

**Algorithm 3:** CANDIDATES( $S$ ) algorithm, our modification

Let us note at this point, however, that the incremental  $k$ -means algorithm would return only a set of cluster centres without stating whether or not we got a perfect ball clustering. But it is important to know if this is the case because otherwise the resulting set of cluster centres may be arbitrary and under unfavourable conditions it may not correspond to a local minimum of  $k$ -means-ideal at all. However, if we are allowed to inspect the data for the second time, such an information can be provided. See Algorithm 2: A second pass for other algorithms from Ackerman and Dasgupta section 2 would not yield such a decision.

Let us further turn to their concept of *nice clustering* (their Def. 3.1.). The clusters are *nicely separated*, as defined by [1], if a distance between an element and any other element of the same cluster is lower than the distance from this element to an element outside of the cluster.

As they show in their Theorem 3.8., nice clustering cannot be discovered by an incremental algorithm with memory linear in  $k$ . In Theorem 5.3 they show that their incremental algorithm 5.2. with up to  $2^{k-1}$  cluster centres can detect points from each of nice clusters. Again it is not the incremental  $k$ -means that may achieve it (see their Theorem 5.7.) even under *nice convex* conditions and with such a large memory. Though, when looking at the issue with randomly generated sequence of data, a memory linear in  $k$  suffices for incremental  $k$ -means with some probability.

Surely our concept of nice-ball-clustering is even more restrictive than their *nice-convex* clustering. But if we upgrade their CANDIDATES( $S$ ) algorithm so that it behaves like  $k$ -means that is if we replace the step "Moving bottom-up, assign each internal node the data point in one of its children" with the assignment to the internal node the properly weighted (with respective cardinalities of leaves) average, then the algorithm 5.2. upgraded to incremental  $k$ -means version will in fact return the *refinement* of the clustering. See Algorithm 3 What is more, if we are allowed to have a second pass through the data, then we can pick out the real cluster centres using an upgrade of the CANDIDATES( $S$ ) algorithm. The other algorithms considered in their section 5 will fail to do this on the second pass through the data (because of deviations from true cluster cen-

tre). The Algorithm 4 is needed: It is clear that if  $k$ -means random or  $k$ -means++

```

Data: Take the tree with  $t$  values assigned, from the Algorithm 3.
Result: Real cluster centres
Assign each node an  $f$  value identical to  $t$  value. ;
while any data point unvisited do
    Acquire the next example,  $x$  ;
    Find the leaf with  $t$  closest to  $x$  ;
    Update its  $f$  value with  $x$  if it is further away from  $t$  than  $f$  ;
    Pass  $x$  to all direct and indirect ancestors (internal) nodes of this leaf,
        where in each of these nodes execute: if  $x$  is further away from  $t$  than  $f$ 
        then
            | update its  $f$  value with  $x$  ;
        end
end
For each cut of the tree engaging exactly  $k$  nodes check if the nice ball
clustering condition is fulfilled for balls rooted at  $t$  with radii  $\|f - t\|$  ;
if for any such a cut the condition holds then
    | the nice ball clustering is found
else
    | it is not
end
```

**Algorithm 4:** Real cluster centres

gets initiated in such a way that each initial cluster centre hits a different cluster, then upon subsequent steps the cluster centres will not leave the clusters. One gets stuck in a minimum, not necessarily the global one. Note that the incremental- $k$ -means will discover the perfect-ball-clustering if it exists and will confirm/reject the existence of such a clustering in the second pass. This means that incremental- $k$ -means has a distinct optimization criterion from  $k$ -means++ and  $k$ -means-random.

But recall the fact that, as with perfect clustering (see [1]), also if there exists a perfect ball clustering into  $k$  clusters, then there exists only one such clustering. So if it exists, it is the global optimum among perfect ball clusterings.

## References

1. Ackerman, M., Dasgupta, S.: Incremental clustering: The case for extra clusters. In: Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada. pp. 307–315 (2014)
2. Kłopotek, R., Kłopotek, M.: On the discrepancy between kleinberg's clustering axioms and k-means clustering algorithm behavior. <https://arxiv.org/abs/1702.04577> (2017)
3. Wierzchoń, S., Kłopotek, M.: Modern Clustering Algorithms. Studies in Big Data 34, Springer Verlag (2018)

# On the Shape of $k$ -means Clusters and Their Motion Consistency

Mieczysław A. Kłopotek<sup>1</sup>, Sławomir T. Wierzchon<sup>1</sup>, and  
Robert A. Kłopotek<sup>2</sup>

<sup>1</sup> Institute of Computer Science,

Polish Academy of Sciences, Warsaw, Poland

[mieczyslaw.kłopotek@ipipan.waw.pl](mailto:mieczyslaw.kłopotek@ipipan.waw.pl)

<sup>2</sup> Faculty of Mathematics and Natural Sciences. School of Exact Sciences,

Cardinal Stefan Wyszyński University in Warsaw, Poland

[r.kłopotek@uksw.edu.pl](mailto:r.kłopotek@uksw.edu.pl)

Since the paper of Kleinberg [2] a number of invariant properties of clustering functions have been investigated. They included such properties as inner or outer consistency [1].  $k$ -means, one of the most popular algorithms occurs to violate most of these properties. For example it has no inner consistency property and the outer consistency is mostly not applicable to it. We will refer here to the versions random  $k$ -means, that is one with random initial seeding of clusters, and to  $k$ -means++, that is one with seeding of clusters according to a heuristic minimizing distance to the closest cluster. We consider so-called batch versions. For an extensive overview of these and other versions of  $k$ -means see [3]).

Below we shall look at one way of defining  $k$ -means properties that resembles the outer consistency, but is more suitable to  $k$ -means. It is a variant of the property of motion consistency.

Recall that *outer consistency* is defined as preservation of partition of data by a clustering algorithm, if the following change to distances between objects are made: distances between objects in the same cluster remain unchanged, while distances between objects in different clusters are increased or unchanged.

In the  $\mathbb{R}^n$  it is impossible to perform outer consistency operation by changing distances to elements of a single  $k$ -means cluster only if that cluster has elements in the convex hull of elements of other clusters. However, we ask if we can weaken the outer consistency conditions by requiring only that the distances between cluster centres are increased.

**Property 1** *A clustering method has the property of motion-consistency, if it returns the same clustering after motion transform, i.e. when the distances of cluster centers are increased by moving each point of a cluster by the same vector without leading to overlapping of the convex regions of clusters.*

Let us look at two neighboring clusters. The Voronoi regions, associated with  $k$ -means clusters, are in fact polyhedrons, such that the *outer* polyhedrons (at least one of them) can be moved away from the rest without overlapping any other region.

So is such an operation on regions permissible without changing the cluster structure? A closer look at the issue tells us that it is not. As  $k$ -means ter-

minates, the neighboring clusters' polyhedra touch each other via a hyperplane such that the straight line connecting centers of the clusters is orthogonal to this hyperplane. This causes that points on the one side of this hyperplane lie more closely to the one center, and on the other to the other one. But if we move the clusters in such a way that both touch each other along the same hyperplane, then it happens that some points within the first cluster will become closer to the center of the other cluster and vice versa.

Moving the clusters generally will change their structure (points switch clusters) unless the points lie actually not within the polyhedrons but rather within *paraboloids* with appropriate equations. Then moving along the border hyperplane will not change cluster membership (locally, that is the data points of the two considered clusters will not switch cluster membership given that we fixed all other clusters and consider reclustering of these two clusters only). But the intrinsic cluster borders are now *paraboloids*. The problem will occur again if we relocate the clusters allowing for touching along the *paraboloids*.

Hence the question can be raised: What shape should the  $k$ -means clusters have in order to be (locally) immune to movement of whole clusters?

Let us consider the problem of susceptibility to class membership change within a 2D plane containing the two cluster centers. Let the one cluster center be located (for simplicity) at a point  $(0,0)$  in this plane and the other at  $(2x_0, 2y_0)$  for some  $x_0, y_0$ . Let further the border of the first cluster be characterized by a (symmetric) function  $f(x)$  and let the shape of the border of the other one  $g(x)$  be the same, but rotated by  $180^\circ$  around  $(x_0, y_0)$ :  $g(x) = 2y_0 - f(2x_0 - x)$ . Let both have a touching point (we excluded already a straight line and want to have convex smooth borders). From the symmetry conditions one easily sees that the touching point must be  $(x_0, y_0)$ . As this point lies on the surface of  $f()$ ,  $y_0 = f(x_0)$  must hold. Any point  $(x, f(x))$  of the border of the first cluster must be closer to its center  $(0, 0)$  than to the center  $(2x_0, 2y_0)$  of the other:

$$(x - 2x_0)^2 + (f(x) - 2f(x_0))^2 - x^2 - f^2(x) \geq 0 \quad (1)$$

That is

$$-x_0(x - x_0) - f(x_0)(f(x) - f(x_0)) \geq 0$$

Let us consider only positions of the center of the second cluster below the  $X$  axis ( $y_0 < 0$ ). In this case  $f(x_0) < 0$ . Further let us concentrate on  $x > x_0$ . We get

$$\frac{f(x) - f(x_0)}{x - x_0} \geq \frac{x_0}{-f(x_0)}$$

In the limit, when  $x$  approaches  $x_0$ ,

$$f'(x_0) \geq \frac{x_0}{-f(x_0)}$$

By analogy for  $x < x_0$  in the limit  $x \rightarrow x_0$  we get:

$$f'(x_0) \leq \frac{x_0}{-f(x_0)}$$

This implies

$$f'(x_0) = \frac{-1}{\frac{f(x_0)}{x_0}} \quad (2)$$

$\frac{f(x_0)}{x_0}$  is the directional tangent of the straight line connecting both cluster centers.  $f'(x_0)$  is the tangential of the borderline of the first cluster at the touching point of both clusters. The equation above means both are orthogonal. But this property implies that  $f(x)$  must define (a part) a circle centered at  $(0, 0)$ . As the same reasoning applies at any touching point of the clusters, a  $k$ -means cluster would have to be (hyper)ball-shaped in order to allow the movement of the clusters without elements switching cluster membership. We shall call batch  $k$ -means style algorithms *iterative algorithms*.

**Property 2** *Let an iterative clustering algorithm in an iterative step reach a (local or global) optimum. Let the distances of cluster centers be increased by moving each point of a cluster by the same vector for that cluster without leading to overlapping of the convex regions of clusters. Let then continue the iterative phase of the clustering algorithm. If no change in clustering occurs whatever moving operation was applied, then this clustering method has the property of reiterative-motion-consistency,*

**Theorem 1.**  *$k$ -means-random and  $k$ -means++ have the property of reiterative-motion-consistency for clusterings for which each cluster can be enclosed in a ball centered at cluster center and no two balls intersect.*

For  $k = 2$ , this is obvious from the above consideration. For  $k > 2$  consider just each pair of clusters to see that no cluster change occurs.

The tendency of  $k$ -means to recognize best ball-shaped clusters has been known long ago, but we are not aware of presenting such an argument for it.

It has to be stated however that reiterative-motion-consistency does not imply motion-consistency, in particular for  $k$ -means algorithms, even if the global optimum has the property of reiterative-motion-consistency. A sufficient separation between the enclosing balls is needed, as we will show again for  $k = 2$ .

Let us consider, under which circumstances a cluster  $C_1$  of radius  $r_1$  containing  $n_1$  elements would take over  $n_{21}$  elements (i.e. subcluster  $C_{21}$ ) of a cluster  $C_2$  of radius  $r_2$  of cardinality  $n_2$ , if we perform the motion-consistency transform. Let  $n_{22} = n_2 - n_{21}$  be the number of the remaining elements (subcluster  $C_{22}$ ) of the second cluster. Let the enclosing balls of both clusters be separated by the distance (gap)  $g$ . Let us consider the worst case that is that the center of the  $C_{21}$  subcluster lies on a straight line segment connecting both cluster centers. The center of the remaining  $C_{22}$  subcluster would lie on the same line but on the other side of the second cluster center. Let  $r_{21}$ ,  $r_{22}$  be distances of centers of  $n_{21}$  and  $n_{22}$  from the center of the second cluster. The relations

$$n_{21} \cdot r_{21} = n_{22} \cdot r_{22}, \quad r_{21} \leq r_2, \quad r_{22} \leq r_2$$

must hold. Let us denote with  $SSC(C)$  the sum of squared distances of elements of the set  $C$  to the center of this set.

So in order for the clusters  $C_1, C_2$  to constitute the global optimum

$$SSC(C_1) + SSC(C_2) \leq SSC(C_1 \cup C_{21}) + SSC(C_{22})$$

must hold. But

$$\begin{aligned} SSC(C_2) &= SSC(C_{21}) + SSC(C_{22}) + n_{21} \cdot r_{21}^2 + n_{22} \cdot r_{22}^2 \\ SSC(C_1 \cup C_{21}) &= SSC(C_1) + SSC(C_{21}) + \frac{n_1 n_{21}}{n_1 + n_{21}} (r_1 + r_2 + g - r_{21})^2 \end{aligned}$$

Hence

$$r_{21} \sqrt{\frac{n_2}{n_1} \frac{n_1 + n_{21}}{n_2 - n_{21}}} - r_1 - r_2 + r_{21} \leq g$$

Let us consider the worst case when the elements to be taken over are at the edge of the cluster region ( $r_{21} = r_2$ ). Then

$$r_2 \sqrt{\frac{n_2}{n_1} \frac{n_1 + n_{21}}{n_2 - n_{21}}} - r_1 \leq g$$

The lower limit on  $g$  will grow with  $n_{21}$ , but  $n_{21} \leq 0.5n_2$ , because otherwise  $r_{22}$  would exceed  $r_2$ . Hence in the worst case

$$r_2 \sqrt{2(1 + 0.5n_2/n_1)} - r_1 \leq g \quad (3)$$

In case of clusters with equal sizes and equal radius this amounts to

$$g \geq r_1(\sqrt{3} - 1) \approx 0.7r_1$$

So we can conclude

**Theorem 2.** *Given a k-means clustering  $\Gamma$  has the property that each cluster can be enclosed in a ball and the gaps between balls fulfill the condition (3), then k-means algorithm has Pairwise-Motion-Consistency property for this data set.*

whereby

**Property 3** *Let a clustering be pairwise optimal, that is we cannot decrease the clustering cost function by reclustering any pair of clusters into a new cluster pair. If a pairwise optimal clustering produced by a clustering method (as a local or global optimum) is transformed by the motion-consistency transform and the same clustering of the transformed data points is again pairwise optimal under the same clustering method, then the clustering method has the property of Pairwise-Motion Consistency.*

## References

1. Ackerman, M., Ben-David, S., Loker, D.: Towards property-based classification of clustering paradigms. In: Proc. NIPS 2010, pp. 10–18 (2010), <http://papers.nips.cc/paper/4101-towards-property-based-classification-of-clustering-paradigms.pdf>
2. Kleinberg, J.: An impossibility theorem for clustering. In: Proc. NIPS 2002. pp. 446–453 (2002), <http://books.nips.cc/papers/files/nips15/LT17.pdf>
3. Wierzchoń, S., Kłopotek, M.: Modern Clustering Algorithms. Studies in Big Data 34, Springer Verlag (2018)

# Analytical Forms of Normalized and Combinatorial Laplacians of Grid Graphs

Mieczysław A. Kłopotek<sup>1</sup> Sławomir T. Wierzchon<sup>1</sup> and  
Robert A. Kłopotek<sup>2</sup>

<sup>1</sup> Institute of Computer Science,

Polish Academy of Sciences, Warsaw, Poland

[mieczyslaw.klopotek@ipipan.waw.pl](mailto:mieczyslaw.klopotek@ipipan.waw.pl)

<sup>2</sup> Faculty of Mathematics and Natural Sciences. School of Exact Sciences,

Cardinal Stefan Wyszyński University in Warsaw, Poland

[r.klopotek@uksw.edu.pl](mailto:r.klopotek@uksw.edu.pl)

A neighbourhood matrix  $S$  of any graph shall be defined as a matrix with entries  $s_{jk} > 0$  if there is a link between nodes  $j, k$ , and otherwise it is equal 0.  $s_{jj} = 0$ . An unnormalised (combinatorial) Laplacian  $L$  of such a graph is defined as  $L = D - S$ , where  $D$  is the diagonal matrix with  $d_{jj} = \sum_{k=1}^n s_{jk}$  for each  $j = 1, \dots, n$ . The respective unoriented Laplacian  $K$  of a graph is defined as:  $K = D + S$ . A normalized Laplacian  $\mathfrak{L}$  of a graph is defined as  $\mathfrak{L} = D^{-1/2}LD^{-1/2} = I - D^{-1/2}SD^{-1/2}$ . A random walk Laplacian  $\mathbb{L}$  of a graph is defined as  $\mathbb{L} = LD^{-1} = I - SD^{-1}$ . Foundations of Laplacians are explained e.g. in [5].

Eigenvalues of  $\mathbb{L}$  and  $\mathfrak{L}$  are identical, while they differ from those of  $L$  and  $K$ , whereas the eigenvectors differ in each case. Eigenvectors of random walk Laplacian can be effortlessly derived from those of normalized Laplacian. Let  $\mathbf{v}$  be the eigenvector of  $\mathfrak{L}$  with eigenvalue  $\lambda$  that is  $\lambda\mathbf{v} = \mathfrak{L}\mathbf{v}$ . This implies  $\lambda D^{1/2}\mathbf{v} = \mathbb{L}D^{1/2}\mathbf{v}$ . Hence  $D^{1/2}\mathbf{v}$  is the eigenvector of  $\mathbb{L}$  for the eigenvalue  $\lambda$ . The eigenvalues of  $L$  and  $K$  will differ unless we have to do with a bipartite graph which is the case with a grid graph. A two-dimensional (unweighted) grid graph [4], (called also a square grid graph, or rectangular grid graph, or  $m \times n$  grid) is an  $m \times n$  lattice graph  $G_{(m,n)}$ , meaning the graph Cartesian product  $P_m \times P_n$  of path graphs on  $m$  and  $n$  vertices resp. Let us define a weighted generalized grid graph as  $G_{(n_1)(w_1)}$  being a weighted path graph of  $n_1$  vertices with weight  $w_1$  for any link in this graph, and the  $d$  dimensional weighted grid graph  $G_{(n_1, \dots, n_d)(w_1, \dots, w_d)}$  being the weighted graph Cartesian product  $G_{(n_1, \dots, n_{d-1})(w_1, \dots, w_{d-1})} \times G_{(n_d)(w_d)}$ . Let us remind a special way of assigning (integer) identities to weighted grid graph  $G_{(n_1, \dots, n_d)(w_1, \dots, w_d)}$  nodes, following the ideas of [1, 3]. The *node identity numbers* run consecutively from 1 to  $\prod_{j=1}^d n_j$ . Each node identity number  $i$  is uniquely associated with a *node identity vector*  $\mathbf{x} = [x_1, \dots, x_d]$  via the (invertible) formula:

$$i = 1 + \sum_{j=1}^d (x_j - 1) \cdot \prod_{k=j+1}^d n_k$$

Let  $\mathbf{i}(i)$  be a function turning the node identity number  $i$  to the corresponding *node identity vector*  $\mathbf{x}$ . A node with identity vector  $[x_1, \dots, x_d]$  is connected for

each  $j$  with the node  $[x_1, \dots, x_j-1, x_d]$  if  $x_j > 1$  and with node  $[x_1, \dots, x_j+1, x_d]$  if  $x_j < n_j$  with weight  $\mathbf{w}_j$  and there are no other edges in the graph.

We will index the eigenvalues and the corresponding eigenvectors with an *eigen identity vector* of  $d$  integers  $\mathbf{z} = [z_1, \dots, z_d]$ . Consider now the similarity matrix  $S$  of the weighted grid graph  $G_{(n_1, \dots, n_d)(\mathbf{w}_1, \dots, \mathbf{w}_d)}$ . It is a  $(\prod_{j=1}^d n_j) \times (\prod_{j=1}^d n_j)$  matrix with  $s_{il} = \mathbf{w}_j$  if nodes with identities  $i, l$  are connected and their connection is in dimension  $j$  and  $s_{il} = 0$  otherwise. Let  $n = \prod_{j=1}^d n_j$  for simplicity.

Combinatorial Laplacians of weighted grid graphs are easily derived from path graphs Laplacians via combination proposed e.g. by Fiedler [2, 5]. Let us define

$$\lambda_{[z_1, \dots, z_d]} = \sum_{j=1}^d 2\mathbf{w}_j \cdot \left( 1 - \cos \left( \frac{\pi z_j}{n_j} \right) \right) \quad (1)$$

where for each  $j = 1, \dots, d$   $z_j$  is an integer such that  $0 \leq z_j \leq n_j - 1$ . Define furthermore

$$\nu_{[z_1, \dots, z_d], [x_1, \dots, x_d]} = \prod_{j=1}^d \cos \left( \frac{\pi z_j}{n_j} (x_j - 0.5) \right) \quad (2)$$

where for each  $j = 1, \dots, d$   $x_j$  is an integer such that  $1 \leq x_j \leq n_j$ . Finally define the  $n$  dimensional vector  $\mathbf{v}_{[z_1, \dots, z_d]}$  such that

$$\mathbf{v}_{[z_1, \dots, z_d], i} = \nu_{[z_1, \dots, z_d], [x_1, \dots, x_d]} \quad (3)$$

**Theorem 1.** *Given the combinatorial Laplacian  $L$  of the weighted grid graph  $G_{(n_1, \dots, n_d)(\mathbf{w}_1, \dots, \mathbf{w}_d)}$ , for each vector of integers  $[z_1, \dots, z_d]$  such that for each  $j = 1, \dots, d$   $0 \leq z_j \leq n_j - 1$ , the  $\lambda_{[z_1, \dots, z_d]}$ , as defined by (1), is an eigenvalue of  $L$  and  $\mathbf{v}_{[z_1, \dots, z_d]}$ , as defined by (3) is a corresponding eigenvector.*

This theorem is proven in [3].

Like in case of unweighted grid graphs, there exists an elegant solution to the eigen-problem of the unoriented Laplacian. The unoriented Laplacian is defined as  $K = D + S$ .

**Theorem 2.** *The unoriented Laplacian eigenvalues for a weighted grid graph are of the same form as for the combinatorial Laplacian that is*

$$\lambda_{[z_1, \dots, z_d]} = \sum_{j=1}^d \mathbf{w}_j \left( 2 \sin \left( \frac{\pi z_j}{2n_j} \right) \right)^2 \quad (4)$$

*The corresponding eigenvectors have components of the form [3]*

$$\nu_{[z_1, \dots, z_d], [x_1, \dots, x_d]} = \prod_{j=1}^d (-1)^{x_j} \cos \left( \frac{\pi z_j}{n_j} (x_j - 0.5) \right) \quad (5)$$

The approach to the eigen-problem of normalised Laplacian differs strongly from combinatorial Laplacian in spite of some formal resemblance. The principal difference is that *the path combination of Fiedler [2] does not work due to the normalization*. Normalization causes that the eigenvectors of weighted grid graph normalized Laplacians, contrary to their combinatorial counterparts, depend also on weights, because the respective eigenvalues depend on them.

If  $\mathbf{v}$  is the eigenvector of  $\mathfrak{L}$  for some eigenvector  $\lambda$ , then

$$D^{-1/2}LD^{-1/2}\mathbf{v} = \lambda\mathbf{v} \Rightarrow LD^{-1/2}\mathbf{v} = \lambda D^{1/2}(D^{1/2}D^{-1/2})\mathbf{v}$$

Setting  $\mathbf{w} = (D^{-1/2}\mathbf{v})$  we state that  $L\mathbf{w} = \lambda D\mathbf{w}$ , i.e. if  $(\lambda, \mathbf{v})$  is an eigenpair of the normalized Laplacian, then  $(\lambda, \mathbf{w})$  solves generalized eigenproblem  $L\mathbf{w} = \lambda D\mathbf{w}$ . In [3] it has been shown that

**Theorem 3.** *The normalized Laplacian  $\mathfrak{L}$  of a  $d$ -dimensional weighted grid graph with at least one inner node has the eigenvalues of the form*

$$\lambda_{\mathbf{z}} = 1 + \sum_{j=1}^d \frac{\mathfrak{w}_j}{\sum_{j=1}^d \mathfrak{w}_j} \cos \left( \frac{1}{n_j - 1} (z_j \pi - 2\delta_j) \right) \quad (6)$$

with the  $\boldsymbol{\delta}^{\mathbf{z}}$  vector, called shift vector, defined as a solution of the equation system consisting of the subsequent equation (8) and the equations (9) for each  $l = 1, \dots, d$ . The corresponding eigenvectors  $\mathbf{v}_{\mathbf{z}}$  have components of the form

$$\begin{aligned} \nu_{\mathbf{z},[x_1, \dots, x_d]} &= D_{[x_1, \dots, x_d], [x_1, \dots, x_d]}^{1/2} \\ &\prod_{j=1}^d (-1)^{x_j} \cos \left( \frac{x_j - 1}{n_j - 1} (z_j \pi - 2\delta_j^{\mathbf{z}}) + \delta_j^{\mathbf{z}} \right) \end{aligned} \quad (7)$$

The defining equations for  $\delta$  (shifts) are:

$$2\mathfrak{w}_{\Sigma} \lambda_{\mathbf{z}} = \sum_{j=1}^d \mathfrak{w}_j \left( 2 + 2 \cos \left( \frac{1}{n_j - 1} (z_j \pi - 2\delta_j^{\mathbf{z}}) \right) \right) \quad (8)$$

$$\begin{aligned} \lambda_{\mathbf{z}} &= 1 + \cos \left( \frac{1}{n_l - 1} (z_l \pi - 2\delta_l^{\mathbf{z}}) \right) \\ &+ \tan(\delta_l^{\mathbf{z}}) \sin \left( \frac{1}{n_l - 1} (z_l \pi - 2\delta_l^{\mathbf{z}}) \right) \end{aligned} \quad (9)$$

By combining the equation (8) with equations (9) for each  $l$  we get an equation system of  $d + 1$  equations from which  $\lambda$  and  $\delta$ s can be determined.

The equation (9) may be transformed to:

$$(\lambda_{\mathbf{z}} - 1) \cos(\delta_l^{\mathbf{z}}) = + \cos(\delta_l^{\mathbf{z}}) \cos \left( \frac{1}{n_l - 1} (z_l \pi - 2\delta_l^{\mathbf{z}}) \right)$$

$$+ \sin(\delta_l^z) \sin \left( \frac{1}{n_l - 1} (z_l \pi - 2\delta_l^z) \right)$$

that is

$$(\lambda_z - 1) \cos(\delta_l^z) = \cos(\delta_l^z - \frac{1}{n_l - 1} (z_l \pi - 2\delta_l^z)) \quad (10)$$

which is simpler to solve for  $\delta$  knowing  $\lambda$ . The solution can be obtained using the bisectional method on  $\lambda$  using the above formula to obtain  $\delta$ s, and then using (6) to get the value of  $\lambda'$  and then reducing bisectionally the difference between  $\lambda$  and  $\lambda'$  down to zero.

The eigenvalues and eigenvectors for Random Walk Laplacians could be conveniently derived from those for Normalized Laplacians. Thus [3]

**Theorem 4.** *The random walk Laplacian  $\mathbb{L}$  of a weighted  $d$ -dimensional grid with at least one inner node, has the eigenvalues of the form*

$$\lambda_z = 1 + \sum_{j=1}^d \frac{\mathfrak{w}_j}{\mathfrak{w}_{\Sigma}} \cos \left( \frac{1}{n_j - 1} (z_j \pi - 2\delta_j^z) \right) \quad (11)$$

with the  $\delta^z$  vector defined as a solution of the equation system consisting of the preceding equation (8) and the equations (9) for each  $l = 1, \dots, d$ . The corresponding eigenvectors  $\mathbf{v}_z$  have components of the form

$$\begin{aligned} \nu_{z,[x_1, \dots, x_d]} &= D_{[x_1, \dots, x_d], [x_1, \dots, x_d]} \\ &\cdot \prod_{j=1}^d (-1)^{x_j} \cos \left( \frac{x_j - 1}{n_j - 1} (z_j \pi - 2\delta_j^z) + \delta_j^z \right) \end{aligned} \quad (12)$$

Note that unweighted grid graphs can serve as a testbed for clustering algorithms in case we want no structure in the data to be detected, while the weighted grid graphs may present a challenge for the case that regular clusters are to be detected. Analytical solutions to the Laplacian determination allow for analytical approach to clusterig algorithm analysis

## References

1. Edwards, T.: The discrete Laplacian of a rectangular grid. [https://sites.math.washington.edu/reu/papers/2013/tom/Discrete\(2013\)](https://sites.math.washington.edu/reu/papers/2013/tom/Discrete(2013))
2. Fiedler, M.: Algebraic connectivity of graphs. Czech. Math. J. **23**(98), 298–305 (1973)
3. Kłopotek, M.A.: Spectral analysis of Laplacian of a multidimensional grid graph - combinatorial versus normalized and random walk Laplacians. ArXiv e-prints 1707.05210 (Sep 2019)
4. mathworld.wolfram.com: Grid graph. <http://mathworld.wolfram.com/GridGraph.html> (2017)
5. Wierzchoń, S., M.A.Kłopotek: Modern Clustering Algorithms, Studies in Big Data, vol. 34. Springer Verlag (2018)

# Return of Investment in Machine Learning: Crossing the Chasm between Academia and Business

Jan Mizgajski<sup>1</sup>, Adrian Szymczak<sup>1</sup>, Piotr Żelasko<sup>1[0000-0002-8245-0413]</sup>, Mikołaj Morzy<sup>2[0000-0002-2905-9538]</sup>, Łukasz Augustyniak<sup>1,3[0000-0002-4090-4480]</sup>, and Piotr Szymański<sup>1,3[0000-0002-7733-3239]</sup>

<sup>1</sup> Avaya

{mizgajski.jan,adrian.dominik.szymczak,piotr.andrzej.zelasko}@gmail.com

<sup>2</sup> Poznan University of Technology mikolaj.morzy@put.poznan.pl

<sup>3</sup> Wrocław University of Science and Technology {lukasz.augustyniak, piotr.szymanski}@pwr.edu.pl

**Abstract.** Academia remains the central place of machine learning education. While academic culture is the predominant factor influencing the way we teach machine learning to students, many practitioners question this culture, claiming the lack of alignment between academic and business objectives. Drawing on a unique set of professional experiences from both sides of the divide, we outline the main points of contention and we try to explain why certain parts of academic curriculum on machine learning may be less than optimal for future machine learning practitioners. We provide recommendations and practical insights into applied aspects of machine learning.

**Keywords:** machine learning · applied science · teaching.

## 1 Introduction

Machine learning (ML) is quickly becoming one of the defining technologies of modern computer science (CS). It attracts attention from both academia and business, with some going as far as calling it the fuel of the future economy. However, the implementation of ML-backed solutions in popular applications and services remains restrained due to significant practical obstacles, one of the most consequential being the lack of alignment between the way ML is presented and taught in academia, and industrial requirements [1, 2]. In this paper, we identify the main divergences between academic and applied ML and we propose an agenda aiming at the better alignment of education with market requirements. Drawing on a unique set of experiences resulting from the involvement in scientific research, educational activities, and working on exclusively privately funded projects, we identify the essential issues in future ML practitioners' education.

## 2 Business Objectives

The quality of an ML model is evaluated using a set of metrics. Students learn about many different metrics designed for particular tasks; examples include accuracy, precision, specificity, Dunn index, silhouette score, cross-entropy, and many others. Unfortunately, the reality is much more complex and requires a more holistic approach to model evaluation.

Firstly, the objective function used for model training is not synonymous with the critical metric. One must not forget that there are essential upstream and downstream metrics which define the quality of an ML model. A simple example is model latency; in several applications, there exist strict response time limits defined by business requirements (e.g., a model must come up with a prediction within 100 ms). Often individual metrics are combined by complex relationships. In many NLP tasks, the objective function is to optimize recall given a fixed minimum precision, a task which is relatively difficult to perform.

Upstream metrics for model evaluation usually revolve around infrastructure costs. For example, a simple model incurring minimum CPU usage cost is preferred to a more complex model requiring more compute, which translates directly into increased costs. On the other hand, downstream metrics postpone the evaluation of ML models - e.g., the quality of word embeddings is measured by their ability to capture sentiment when combined with a classifier. In practical applications, we also see a confusion of simple metrics, useful for model training, with more sophisticated measures. In the area of automatic speech recognition, a popular metric is that of word error rate (WER). This metric is lackluster when it comes to a real assessment of the ASR quality, which requires the analysis of grammatical correctness, verb confusion, or semantic understanding of the text. Finally, A/B tests by human annotators can be used to evaluate the quality of models practically.

In our view, academia does not pay enough attention to the practical ramifications of model training and evaluation. ML models do not operate in a vacuum; they constitute a broader ecosystem, and we must evaluate them in the context of business objectives they serve. This principle may be difficult to apply in an academic environment, but we need to continuously remind the students of the necessity to think critically about model training and evaluation. They should pay particular attention to possible adversarial effects of overfitting, model bias, lack of robustness, computational constraints, or infrastructure costs.

## 3 Data

Typically, the most effective machine learning models are data greedy. Many courses and textbooks stress the importance of data and feature engineering, but it is challenging to comprehensively present all intricacies of the process in an academic setting. What we often see is that students learn about the essential feature engineering techniques (normalization, binning, feature extraction, feature encoding) and then proceed to learn about various algorithms. On the

other hand, every practitioner will recognize that data preparation is the most crucial step in the entire machine learning pipeline. In a practical setting, they will spend the majority of time on data and feature engineering, and it will be time well-spent.

Below we present some practical hints on how to make working with data more exciting and attractive to students.

- Successful training of ML models requires access to large training datasets. If not enough labeled examples are available, techniques such as active learning [3] and unsupervised pre-training with supervised fine-tuning [4] can be employed.
- Many pre-trained models have been published over recent years. An auspicious direction is to use these models in new application domains via transfer learning [5], especially when training of models from scratch is prohibitively expensive. This practice has become the de facto standard approach in computer vision and NLP tasks.
- Students are rarely taught about the possibility of manual data annotation, even though simple and effective tools for annotation are readily available. Data annotation can be done directly from Jupyter notebooks [16, 17], it can be outsourced to tools such as Figure Eights [18], or Prodigy [6].

We also observe bias in academia towards supervised learning. In practice, unsupervised learning has proven itself to be surprisingly useful and is significantly cheaper in terms of labeled data availability. Nevertheless, obtaining quality training data poses a substantial challenge, even for unsupervised learning. While providing students with links to Kaggle competition datasets allows them to work on real-world problems, it does not prepare them for real-world data acquisition challenges. We strongly suggest combining academic courses on machine learning with data acquisition and processing courses.

## 4 Algorithms

For many students, the algorithms lay at the heart of machine learning. To be competent in ML means to know the inner workings of an algorithm, be able to adjust its hyper-parameters, and interpret the results. Many ML courses focus on a detailed presentation of a broad scope of algorithms for classification, regression, or clustering. However, focusing solely on these aspects results in overstressing of the role of individual algorithms and is detrimental to the application of ML to solve real-world problems.

In our practice, we find that starting every new experiment with a simple baseline is not only a good strategy but a necessity. General-purpose classification algorithms such as Random Forest, Gradient Boosting, or Naive Bayes can provide strong baselines in many classification tasks, whereas k-means and DBSCAN tend to perform well for clustering. In case of NLP tasks, one does not have to resort to training an expensive model with hundreds of millions of parameters such as the Transformer - pre-computed word embeddings combined

with Support Vector Machines tend to work remarkably well across a wide range of problems. The same can be said of image recognition, stripping layers from large pre-computed models such as VGG19, ResNet, or InceptionV3 is a very reasonable starting point.

What we find most problematic is the insufficient focus on the model training process. Students should consider the influence of an objective function on the model. A mismatch between the training criterion and real objectives has led YouTube to recommend content full of racism and conspiracy theories because it maximized the number of views [7]). A practitioner should always check for potential information leaks between training and evaluation sets, biases in the training datasets and model over-fitting.

Applied machine learning is a highly iterative process, with many loops involving data selection, data engineering, model training, and model evaluation. In our experience, visual tools that help to interpret and explain trained models are invaluable, but we find that often, the coverage of these tools at academia is insufficient. Visualization can be easily introduced into the machine learning workflow using tools such as Yellowbrick [19] or Visdom [20]. Text datasets can be visualized using Scattertext [8]. Deep neural nets can be inspected using TensorBoard [21], GanLab [22], saliency maps [9], or Deep Visualization [10]. Even recurrent neural networks can be easily visualized to reveal activations and attention [11–13].

## 5 Process

Finally, we arrive at the most contentious point: a total lack of support for the machine learning process from academia. It is here where we observe the most significant discrepancy between the standards of academia and industry. The main reason for this chasm is the fundamental difference in research culture, objectives, and incentives. Let us briefly present the fundamentals of ML culture in most business organizations. They outline the most significant points of issue between academia and business.

- *The quickest path to invalidate a hypothesis:* the main incentive in academic research is the publication in a prestigious venue, which enforces certain constraints on the minimum scope of the experiment. In business, however, we are interested in the most cost-effective way of validating or invalidating a hypothesis.
- *Minimal viable dataset:* many scientific experiments require the curation of a high-quality training dataset. Since in applied ML, we focus on fast pruning of unpromising paths, we can quickly verify hypotheses on small datasets.
- *Data and exploration first:* thoroughly knowing one’s data is the key requirement for the successful application of machine learning in real-world problems. Out of the box models and algorithms usually require sophisticated domain adaptation, which cannot be performed without a deep understanding of data and its characteristics. We can imagine a keyword extraction problem

for scientific papers (a standard problem in academia) and its application to transcribed calls in the call center (a very domain-specific problem).

- *Embrace failure and learn:* this is one of the differentiating points between theoretical and applied machine learning. Academia has a strong publication bias for positive results; publishing of negative results is fairly difficult and not very common. On the other hand, most ideas and approaches do not work in practice, so being able to let go of a failed idea and move to the next experiment is essential in applied research. Failures are not synonymous with wasted resources, but trying to squeeze microscopic improvements over baseline solutions erratically is.
- *Document everything:* often, it is more important to know what did not work and what was considered but rejected and why. Everything should be documented: rationales, tools, assumptions, solutions, outcomes, requirements like memory, CPU or GPU. It can be partially achieved using the principles of reproducible science, but in our opinion, the students of ML should be made aware of the importance of documentation of ML workflows and its reproducibility.

Of course, research culture forms the backbone of the organization, but equally important is the ML process itself. Far too often, we see ML problems solved as independent software services, in separation from the rest of the technological stack. In reality, the ML model will be a small piece in a much larger puzzle, albeit incurring disproportionately large technical debt [14]. It is essential to employ all the best software engineering practices when working on an ML project. These practices include:

- *Use agile processes all the way:* run ML project using a flexible methodology (e.g. kanban), with short daily re-caps, planning, definitions of done, retrospectives, and frequent syncing between team members to brainstorm and validate ideas.
- *Do code reviews:* this is one of the best and quickest ways to obtain high-quality code. Besides, it promotes the best programming practices and supports horizontal knowledge transfer in the organization.
- *Know when to stop:* there are clear diminishing returns of doing more research and gathering more data. Sometimes it can lead to paralysis by analysis when incremental model improvements do not contribute to solving the business problem anymore. Avoid it by checking the alignment of current research with business values and objectives often. It could be an excellent idea to time-box complex research tasks and try not to dig too deep sometimes.
- *Test (almost) everything:* seldom do we see ML code presented to the students of ML accompanied by unit tests, integration tests and automatic (or at least reproducible) builds. At the same time, no sane person would allow releasing an untested code in a production environment. ML code must be tested at least as thoroughly as the rest of the codebase. However, testing of ML code is not simple. It may involve more dependencies, evolving libraries, and frameworks. It also depends on train/test/validation datasets, which may not be time-invariant.

- *Outsource model serving complexity*: there are many services that may be used to provide stable and robust execution environments for ML processes, think in advance about integration with TensorFlow Serving [23], Weights & Biases [25], Amazon SageMaker [24] or Neptune [26].
- *Avoid over-engineering*: refrain from premature optimization at all costs and follow the key rule of extreme programming: *you ain't gonna need it* (YAGNI). ML workflows tend to become convoluted quite quickly, so do not add any unnecessary complexity.

Last but not least, academia rarely teaches students to think of ML models through the prism of the final product. This perspective requires careful consideration of several factors which may or may not fall outside of the area of expertise of machine learning researchers and engineers. Apart from apparent issues of customer research, profitability, marketing (it always lies about ML, let them!), the expertise of ML practitioners can be beneficial for the development of the final product. There are several possible strategies that a product can employ to gain popularity: being first to the market, being cheaper than the competition, being better than the competition (claimed with data), being a fast follower, or maybe just being good enough. People working on ML models and workflows should be aware of the strategy the product is employing and should evaluate their progress in the context of that strategy. The final goal is not to be perfect, but to be sellable.

## 6 Conclusions

In this paper, we marked issues which hinder the quick adaptation of students of machine learning who leave academia for business environments. We are not addressing the question of the relevance of basic research on machine learning to the actual needs of the market, nor are we criticizing the way research is conducted at academia. Far from it, we focus exclusively on the problem of teaching machine learning. We note that few people involved in teaching ML at universities have extensive practical experience. This problem is exacerbated by the constant brain drain of top ML talent by large companies. A recent study [15] reveals an exponential growth of the number of tenure-track and tenured faculty leaving universities for the industry. The researchers show how this "corporate poaching" of top ML talent negatively impacts the overall economy by stifling ML entrepreneurship among the alumni, weakening ML brainpower available to government and universities, and lowering the rate of innovation and disruptive creativity. We believe that a better alignment of academia and business is possible and that it can be mutually beneficial. We welcome the discussion which our paper hopefully sparks among academics.

## References

1. VanderPlas, Jake, "The Big Data Brain Drain: Why Science is in Trouble" [jakevdp.github.io/blog/2013/10/26/big-data-brain-drain/](https://jakevdp.github.io/blog/2013/10/26/big-data-brain-drain/), (accessed September 2019)

2. VanderPlas, Jake, "Hacking Academia: Data Science and the University" [jakevdp.github.io/blog/2014/08/22/hacking-academia/](https://jakevdp.github.io/blog/2014/08/22/hacking-academia/), (accessed September 2019)
3. Gilyazev, R. A., and D. Yu. Turdakov, "Active Learning and Crowdsourcing: A Survey of Optimization Methods for Data Labeling." *Programming and Computer Software* 44.6 (2018): 476-491.
4. Howard, Jeremy, and Ruder, Sebastian, "Universal language model fine-tuning for text classification." arXiv preprint arXiv:1801.06146 (2018).
5. Weiss, Karl, Taghi M. Khoshgoftaar, and Wang, DingDing, "A survey of transfer learning." *Journal of Big data* 3.1 (2016): 9.
6. Montani, Ines, and Honnibal, Matthew, "Prodigy: A new annotation tool for radically efficient machine teaching." *Artificial Intelligence*, to appear (2018).
7. Tufekci, Zeynep, "YouTube's Recommendation Algorithm Has a Dark Side" <https://bit.ly/2m09tvZ>, (accessed September 2019)
8. Kessler, Jason, "Scattertext: a Browser-Based Tool for Visualizing how Corpora Differ". ACL System Demonstrations. 2017.
9. Zeiler, Matthew D., and Fergus, Rob, "Visualizing and understanding convolutional networks." European Conference on Computer Vision. Springer, Cham, 2014.
10. Yosinski, Jason, et al. "Understanding neural networks through deep visualization." arXiv preprint arXiv:1506.06579 (2015).
11. Karpathy, Andrej, "The unreasonable effectiveness of recurrent neural networks." Andrej Karpathy blog 21 (2015).
12. Strobelt, Hendrik, et al. "Seq2seq-vis: A visual debugging tool for sequence-to-sequence models." *IEEE transactions on visualization and computer graphics* 25.1 (2018): 353-363.
13. Wang, Junpeng, et al. "Dqnviz: A visual analytics approach to understand deep q-networks." *IEEE Transactions on Visualization and Computer Graphics* 25.1 (2018): 288-298.
14. Sculley, David, et al. "Hidden technical debt in machine learning systems." *Advances in neural information processing systems*. 2015.
15. Gofman, Michael, and Zhao, Jin, "Artificial Intelligence, Human Capital, and Innovation", [http://gofman.info/AI/AI\\_GofmanZhao.pdf](http://gofman.info/AI/AI_GofmanZhao.pdf) (accessed September 2019)
16. <https://github.com/agermanidis/pigeon>
17. <https://github.com/chestrays/jupyanno>
18. <https://www.figure-eight.com/>
19. <https://www.scikit-yb.org>
20. <https://github.com/facebookresearch/visdom>
21. <https://www.tensorflow.org/tensorboard/>
22. <https://poloclub.github.io/ganlab/>
23. <https://github.com/tensorflow/serving>
24. <https://aws.amazon.com/sagemaker>
25. <https://www.wandb.com>
26. <https://neptune.ml>

# **SOUП-Bagging: a new approach for multi-class imbalanced data classification**

Mateusz Lango and Jerzy Stefanowski

Institute of Computer Science, Poznań University of Technology, Poznań, Poland  
`{mlango, jstefanowski}@cs.put.poznan.pl`

## **1 Introduction**

Learning from imbalanced data is an important and prevalent issue in machine learning research and applications [2]. Class-imbalanced data occur in many applications such as fraud detection, network intrusion detection, sentiment analysis, predictive maintenance and in the medical domain. As the standard classifiers fail to sufficiently recognize the minority classes, many novel algorithms have been developed in recent years. They are usually categorized into three groups: data-level, algorithmic-level and ensemble methods. The first data-level methods modify the original distribution of the data in order to improve the classification of minority classes. They can be used with virtually any classification algorithm. Algorithmic techniques modify a particular classification algorithm trying to make it more accurate for class-imbalanced problems. The ensemble methods exploit both these directions in the construction of combined classifier set, e.g. they generalize bagging and boosting schema and additionally incorporate data pre-processing of the training subsets. Following [1] they are quite efficient in improving prediction measures for minority classes.

Most research has been devoted to a binary version of imbalanced data with a single minority class and a single majority class. There are also imbalanced problems with several important minority classes and in the last years, the increasing research interest has been observed on that issue [7]. It has been addressed mainly by adapting binary decomposition strategies (e.g. pairwise ensembles) or by proposing simple modifications of re-sampling methods. However, some of these researchers questioned the initial belief that multi-class imbalanced learning can be solved by simple decomposition into binary problems [4]. In particular, they postulated that these techniques are insufficiently dealing with complex interrelations which occur between classes. For instance, a class of average size can act as a minority class in the region dominated by majority class, at the same time causing difficulties in the recognition of other, smaller classes.

In our previous research, we introduced a new approach for examining the interrelations of multiple classes in imbalanced data [6]. It is based on analyzing the neighborhood of minority class examples and on the additional information about similarities between classes. Recently, we exploited this idea in the new resampling approach called Similarity Oversampling and Undersampling Preprocessing (SOUP) [3]. Even though in the experimental evaluation SOUP proved

to be an efficient approach for dealing with multiple imbalanced classes, the possibility of constructing an ensemble classifier was not considered. In this paper, we put forward the proposition of SOUP-Bagging - a bagging-based ensemble algorithm which leverages SOUP during its training.

## 2 SOUP Pre-processing Algorithm

Due to page limits we will only provide a brief description of Similarity Oversampling and Undersampling Preprocessing (SOUP) [3] which is directly related to this paper. A reader interested in more details has to consult [6, 3].

SOUP, as its name suggests, combine oversampling with undersampling to achieve a balanced class distribution in the training set. After SOUP resampling all classes have the same cardinality being equal the mean of the biggest minority and the smallest majority class sizes. This causes that (except corner cases) all the minority classes are oversampled and all the majority classes are undersampled by the algorithm. Both under- and oversampling is not performed randomly and the weight based selection of instances to be resampled is the key ingredient of SOUP algorithm, where inspirations from [6] are utilised to establish the level of difficulty of each example.

During undersampling, SOUP tries to clear the decision boundary from majority instances at the same time strengthening the minority class concepts with oversampling. To this end, a notion of a safe level is used. The safe level of an instance  $x$  belonging to class  $C_i$  is defined as

$$\text{safe}(x_{C_i}) = \frac{1}{k} \sum_{j=1}^l n_{C_j} \mu_{ij} \quad (1)$$

where  $n_{C_j}$  is the number of  $k$ -nearest neighbors of  $x$  which belong to  $C_j$  class and  $\mu_{ij}$  is the special *degree of similarity between classes*  $C_i$  and  $C_j$ , which allows us to model interrelations between classes [6]. This degree is defined<sup>1</sup> by

$$\mu_{ij} = \frac{\min(|C_i|, |C_j|)}{\max(|C_i|, |C_j|)} \quad (2)$$

where  $|C_i|$  is the size of  $C_i$  class. Safe level of an examples is higher in the clear homogenous regions, dominated by the example's class. In the presence of instances from other classes the safe level decreases, taking into account sizes of surrounding classes. If the classes are of roughly the same size, safe level does not drop significantly, but together with bigger discrepancies between class sizes the decrease is more notable. SOUP uses this properties of the safe level to clean decision boundary from majority examples by undersampling instances with the lowest safe level values. On the other hand, safe regions of minority class are enlarged by duplicating examples with highest safe levels.

---

<sup>1</sup> In the original SOUP paper, authors suggest that  $\mu_{ij}$  should be provided by a domain expert. Here, for simplicity we provide a heuristic which is used in SOUP-Bagging.

---

**Algorithm 1** Similarity Oversampling and Undersampling Preprocessing (SOUP)

---

**Input:**  $D$ : original training set of  $|D|$  examples with  $c$  classes;  $C_{min}$ : indexes of minority classes;  $C_{maj}$ : indexes of majority classes

**Output:**  $D'$ : balanced training set

```

1: Split dataset  $D$  into  $c$  homogeneous parts  $D_1, D_2, \dots, D_c$ . Each  $D_i$  contains all
   examples from  $i$  class
2:  $D' = \emptyset$ 
3:  $m \leftarrow \text{mean}(\min_{i \in C_{maj}} |D_i|, \max_{j \in C_{min}} |D_j|)$ 
4: for all  $i \in C$  do
5:   for all  $x \in D_i$  do
6:     find  $k$  nearest neighbours of  $x$ 
7:     calculate safe level of  $x$ , according to Eq. 1
8:   end for
9:   if  $|D_i| > m$  then
10:    remove  $|D_i| - m$  examples with the lowest safe level values from  $D_i$ 
11:   else
12:    duplicate  $m - |D_i|$  examples with the highest safe level values in  $D_i$ 
13:   end if
14:    $D' \leftarrow D' \cup D_i$ 
15: end for
16: return  $D'$ 
```

---

The experimental evaluation of SOUP was performed over 19 imbalanced datasets [3] and it was compared against decomposition ensembles, resampling methods, and Multi-class Roughly Balanced Bagging (MRBB) [5]. SOUP stood out as the best performing method for decision trees (J48) and k-nearest neighbour classifier, loosing only to MRBB while using PART rules. Nevertheless, this result raises a question about the possibility of further improvement by ensembling techniques.

### 3 SOUP-Bagging

We investigate the possibility of improving SOUP by combining it with bagging which was often successfully generalized for binary complex imbalanced datasets. Moreover, bagging-based MRBB proved to be useful in multi-class problems [5] and it worked better than decomposition-based ensembles [3].

We introduce SOUP-Bagging algorithm whose pseudocode is presented in Alg. 2. The method iteratively resamples the original dataset with replacement, applies SOUP preprocessing technique and constructs a classifier. While resampling the dataset, stratified sampling is used. Predictions of the component classifiers are aggregated by the majority voting.

We carry out its experimental evaluation using the same real datasets used in [3]. Table 1 presents average ranks (as in the Friedman test) of G-mean measure while using J48 classifier. SOUP-Bagging stood out as the best-performing

**Algorithm 2** SOUP-Bagging

**Input:**  $D$ : original training set of examples of size  $N$ ,  $k$ : number of bootstrap samples,  $LA$ : learning algorithm;

**Output:**  $C^*$  bagging ensemble with  $k$  component classifiers

Learning phase:

- 1: **for**  $i = 1 \rightarrow k$  **do**
- 2:    $S_i \leftarrow N$ -element sample drawn with replacement from  $D$
- 3:    $S_i \leftarrow \text{SOUP}(S_i)$
- 4:    $C_i \leftarrow LA(S_i)$
- 5: **end for**

Prediction phase:

$$C^*(x) = \arg \max_y \sum_{i=1}^k p_{C_i}(y|x)$$

**Table 1.** Average rank (like in the Friedman test) of G-mean obtained by algorithms with J48 classifier.

| Algorithm    | SOUP-Bagging | SOUP | OVO  | RUS | OVO  | ROS  | MRBB | Global-CS | Static-SMOTE |
|--------------|--------------|------|------|-----|------|------|------|-----------|--------------|
| Average rank | 2.80         | 3.30 | 3.30 |     | 3.56 | 3.90 | 4.93 | 6.20      |              |

approach, however, the difference between SOUP and SOUP-Bagging is not statistically significant according to the pairwise Wilcoxon test. Nevertheless, the difference between SOUP-Bagging and the other bagging-based approach which achieved the best results in our previous studies is statistically significant.

**Summary:** In this paper we promote new approach to deal with complex interrelation between multiple imbalanced classes. We partly summarize earlier research and introduce their new generalization into SOUP-Bagging.

## References

1. Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F.: A review on ensembles for the class imbalance problem: bagging-, boosting, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4), 463–484, (2011).
2. Haibo He, Edwardo A. Garcia: Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9), 1263–1284, (2009).
3. Janicka M., Lango M., Stefanowski J.: Using information on class interrelations to improve classification of multi-class imbalanced data: a new re-sampling algorithm, *International Journal of Applied Mathematics and Computer Science*, 29 (4), (2019).
4. Krawczyk, B.: Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5(4), 221–232 (2016).
5. Lango M., Stefanowski J.: Multi-class and Feature Selection Extensions of Roughly Balanced Bagging for Imbalanced Data. *JIIS*, 50 (1), 97–127 (2017) .
6. Lango, M., Napierala, K. and Stefanowski, J.: Evaluating difficulty of multi-class imbalanced data, Proc. 23rd Inter-national Symposium ISMIS, pp. 312–322, (2017).
7. Wang, S., Yao, X: Multiclass imbalance problems: Analysis and potential solutions. *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, 42(4), 1119–1130 (2012).

# Streaming approach to Big Data analysis

Piotr Duda<sup>1[0000–0001–7182–1349]</sup> and Leszek Rutkowski<sup>1[0000–0001–6960–9525]</sup>

Czestochowa University of Technology, Czestochowa, Poland  
piotr.duda@iisi.pcz.pl

**Keywords:** Big data · Stream data mining · nonstationarity

**Abstract.** The issues of Big Data analysis are currently a great challenge for researchers. In this paper, we present a possibility to apply a stream data approach to handle this task. Our approach is based on the ensemble method and can be successfully applied to various types of nonstationary environments.

## 1 Motivations

A rapid increase in internet access allows moving many of the everyday activities to the world web. Over two billion people are connected to the net. They use it every day, both as a work tool and for private purposes. The Internet has found applications in many fields such as financial markets, healthcare, security systems or social communications. In the same time, a requirement to processes digital data has also significantly increased. Upload of digital images or videos is a common practice especially in view of the expansion of mobile technologies. In the consequence of a still growing number of the produced data, the new challenges have emerged. In the literature, the aforementioned issue is known under the name big data. This is a very hot topic of research and much theoretical work is needed to develop the appropriate methods and algorithms [1].

One of the crucial challenges of Big Data analysis is its size. The size of the data caused a problem, not only with their analysis but also with their storage. There are many attempts to handle this limitation. Sometimes the data are stored in the form of the uniform relational databases, but more often the data are gathered in many physical servers, operating in different locations. Access do the data can be provided in different ways, like cloud storage, data warehouses, or OLAP. The huge size of the data and some difficulties with gathering them in one place at the same time, make impossible to apply directly most of the traditional data analysis techniques. In consequence, some limitations have been put on the algorithms, to make them useful in the Big Data analysis. One of the most promising approaches is known in the literature under the name of the Stream Data Mining (SDM). In particular, the well-designed SDM algorithm has to fulfill the following criteria:

- the data cannot be stored and considered as a whole. The data elements should be analyzed as fast as it is possible, and the memory should be released for a new coming data,

- the processing time should be as limited as it is possible, as the number of the new coming data can be arbitrarily fast,
- the distribution of the data can change during processing of the stream. Such an event is called a concept-drift and it can occur in every moment. The algorithm should be able to adjust to a time-varying environment.

In this paper, various issues concerning the adaptation of the ASE algorithm to the case of the non-stationary environment are presented.

## 2 Research methods

The ensemble methods are a great tool to process the big data set. They have a natural feature to adjust to the new concepts, coming from time-varying environments, by applying the appropriate methods to manage to add and removing the weak classifiers. Most of the methods proposed in the literature are only focused on optimizing the accuracy on a current chunk of the data, without trying to justify decision based on the broader context. In [2] and [3] we developed the Automatically Sized Ensemble (ASE) algorithm, which guarantees with the assumed probability that addition of a new component will increase the accuracy, not only for the current data chunk but also for the whole data stream. The proposed procedure has mathematical justification and allows to obtain satisfactory results in a real-world problem. However, its results could be significantly improved if we assume some knowledge about the type of non-stationarity. For example in a case of recurring changes, the algorithm, which was initially trained on data chunks generated from the first concept, will try to fit itself to a new concept by replacing older components by the new ones. After the next change of concept (again to the initial concept), the accuracy will significantly decrease once again. The reason behind such behavior will be the rejection of the previously gathered knowledge, during adjusting the ensemble to a new environment. In [4], we assumed that the data stream is a sequence of the data, generated by the different distributions, which appear and disappear alternately. We applied a KullbackLeibler divergence to compare the usability of a single component on the stage of removing component from the ensemble. Similar considerations allow us to improve the ASE algorithm, in the case of gradual changes, with the application of Hellinger distances, see [5]. We also investigate, as it was suggested in [6], the ensemble method for solving regression problems in a time-varying environment.

In the coming years, the Big Data analysis will be an important subject of research. Stream data processing can still provide many algorithms that can be used to solve real-world problems. In future work, we are planning to develop an algorithm able to tracking nonstationary density functions. Some efforts have been put, by application of Parzen-kernel estimators [8].

## References

1. Erl, T., W. Khattak, and P. Buhler, (2016) Big data fundamentals: concepts, drivers and techniques. Prentice Hall Press.
2. Pietruczuk, L. L. Rutkowski, M. Jaworski, and P. Duda, (2016) A method for automatic adjustment of ensemble size in stream data mining, in IEEE International Joint Conference on Neural Networks (IJCNN), pp. 915.
3. Pietruczuk, L., Rutkowski, L., Jaworski, M., Duda, P.: How to adjust an ensemble size in stream data mining? Information Sciences 381, 4654 (2017)
4. Duda, P., M. Jaworski and L. Rutkowski, (2017) On ensemble components selection in data streams scenario with reoccurring concept-drift, in IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, HI, pp. 1-7.
5. Duda, P., (2018) On Ensemble Components Selection in Data Streams Scenario with Gradual Concept-Drift, Artificial Intelligence and Soft Computing, Lecture Notes in Computer Science, vol 10842. Springer, Cham, p. 311-320.
6. Duda P., Jaworski M., Rutkowski L. (2018) Online GRNN-Based Ensembles for Regression on Evolving Data Streams. Advances in Neural Networks, ISNN 2018, Lecture Notes in Computer Science, vol 10878. Springer, Cham, pp. 221-228.
7. Duda, P., Jaworski, M., Rutkowski, L.: International Journal Neural Systems, vol 28, 1750048 [23 pages] (2018).
8. Duda P., L. Rutkowski, M. Jaworski, D. Rutkowska, On the Parzen kernel-based probability density function learning procedures over time-varying streaming data with applications to pattern classification, IEEE transactions on cybernetics, PP, 2019

# Various Aspects of Data Distribution Monitoring Using the Restricted Boltzmann Machine

Maciej Jaworski<sup>[0000–0002–8410–4231]</sup> and Leszek Rutkowski<sup>[0000–0001–6960–9525]</sup>

Institute of Computational Intelligence, Czestochowa University of Technology  
Czestochowa, Poland  
`{maciej.jaworski, leszek.rutkowski}@iisi.pcz.pl`

**Abstract.** In this paper, we take under considerations a very challenging issue of computer science research, i.e. the data stream mining. We analyze the applicability of the Restricted Boltzmann Machine (RBM) as a concept drift detector. It turns out that the properly learned RBM can be successfully used to monitor possible changes in the data distribution. We also discuss the resource-awareness of the proposed method and its capability for handling missing values.

**Keywords:** Restricted Boltzmann Machine · Data stream mining · Data distribution estimation · Concept drift detection.

## 1 Introduction

In recent years, data stream mining became a very challenging issue in computer science research. In the case of data stream classification tasks, algorithms which received the most interest seem to be those based on decision trees [7], [10], [11], [12], [13]. Another group of methods naturally applicable in data stream scenarios are ensemble algorithms [9]. For the regression task in the time-changing environment, several non-parametric methods have been proposed, e.g. based on Parzen kernel functions [2] or orthogonal series [3]. Such non-parametric methods can be also successfully applied for unsupervised learning of data distributions in time-varying environments [1] or to track non-stationary noise variances of data [6]. In this paper, various issues concerning the active concept drift detection in time-varying data stream mining are discussed. We investigate the applicability of the Restricted Boltzmann Machine (RBM) as a tool for monitoring the changes in data distribution [4]. The RBMs are neural networks able to learn generative models of data. After training, they contain compressed information about the distribution from which the training data were drawn. We suppose that such an RBM learned on a part of the data stream can then be used to determine whether the new data elements from the stream are drawn from the same distribution or not. Two indicators are proposed to be used for evaluation of incoming data: the free energy and the reconstruction error. Their computation is relatively fast, hence they are suitable for data stream scenario. The RBM is suited for unsupervised learning scenario. However, after adding

softmax layer it can be easily adapted to handle labeled data [5]. Preliminary experimental results demonstrate that the proposed tool can deal with different types of concept drift, e.g. the sudden or the gradual.

We also consider the problem of resource-awareness in data stream mining with an application of the Restricted Boltzmann Machine (RBM) [8]. If the data incoming rate is very fast, an appropriate algorithm should work as fast as possible. In our research two RBM learning algorithms are investigated, i.e. the Contrastive Divergence and the Persistent Contrastive Divergence. We test three strategies for dealing with a buffer overflow in the case of high-speed data streams: load shedding, mini-batch resizing, and controlling the number of Gibbs steps in the learning algorithm. Considered approaches are verified on the real MNIST dataset which is treated as a part of a data stream.

Additionally, we consider another problem which often occurs in real-life data streams, i.e. incomplete data. We propose two modifications of the RBM learning algorithms to make them able to handle missing values. The first one inserts an additional procedure before the positive phase of the Contrastive Divergence. This procedure aims at inferring the missing values in the visible layer by performing a fixed number of Gibbs steps. The second modification introduces dimension-dependent sizes of minibatches in the stochastic gradient descent method. The proposed methods are verified experimentally, demonstrating their usability for concept drift detection in data streams with incomplete data.

## References

1. Duda, P., Rutkowski, L., Jaworski, M., Rutkowska, D.: On the parzen kernel-based probability density function learning procedures over time-varying streaming data with applications to pattern classification. *IEEE Transactions on Cybernetics* pp. 1–14 (2018)
2. Duda, P., Jaworski, M., Rutkowski, L.: Convergent time-varying regression models for data streams: Tracking concept drift by the recursive parzen-based generalized regression neural networks. *International Journal of Neural Systems* **28**(02), 1750048 (2018)
3. Duda, P., Jaworski, M., Rutkowski, L.: Knowledge discovery in data streams with the orthogonal series-based generalized regression neural networks. *Information Sciences* **460-461**, 497 – 518 (2018)
4. Jaworski, M., Duda, P., Rutkowski, L.: On applying the Restricted Boltzmann Machine to active concept drift detection. In: *Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence*. pp. 3512–3519. Honolulu, USA (2017)
5. Jaworski, M., Duda, P., Rutkowski, L.: Concept drift detection in streams of labelled data using the Restricted Boltzmann Machine. In: *2018 International Joint Conference on Neural Networks (IJCNN)*. pp. 1–7 (2018)
6. Jaworski, M.: Regression function and noise variance tracking methods for data streams with concept drift. *International Journal of Applied Mathematics and Computer Science* **28**(3), 559–567 (2018)

7. Jaworski, M., Duda, P., Rutkowski, L.: New splitting criteria for decision trees in stationary data streams. *IEEE Transactions on Neural Networks and Learning Systems* **29**(6), 2516–2529 (2018)
8. Jaworski, M., Rutkowski, L., Duda, P., Cader, A.: Resource-aware data stream mining using the Restricted Boltzmann Machine. In: Rutkowski, L., Scherer, R., Korytkowski, M., Pedrycz, W., Tadeusiewicz, R., Zurada, J.M. (eds.) *Artificial Intelligence and Soft Computing*. pp. 15–24. Springer International Publishing, Cham (2019)
9. Pietruczuk, L., Rutkowski, L., Jaworski, M., Duda, P.: How to adjust an ensemble size in stream data mining? *Information Sciences* **381**(C), 46–54 (2017)
10. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: The CART decision tree for mining data streams. *Information Sciences* **266**, 1–15 (2014)
11. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: Decision trees for mining data streams based on the Gaussian approximation. *IEEE Transactions on Knowledge and Data Engineering* **26**(1), 108–119 (2014)
12. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: A new method for data stream mining based on the misclassification error. *IEEE Transactions on Neural Networks and Learning Systems* **26**(5), 1048–1059 (2015)
13. Rutkowski, L., Pietruczuk, L., Duda, P., Jaworski, M.: Decision trees for mining data streams based on the McDiarmid's bound. *IEEE Transactions on Knowledge and Data Engineering* **25**(6), 1272–1279 (2013)

# Improving Evolutionary Instance Selection with Clustering and Ensembles

Mirosław Kordos<sup>1</sup>[0000–0002–2031–7561] and  
Marcin Blachnik<sup>2</sup>[0000–0003–3336–4962]

<sup>1</sup> Department of Computer Science and Automatics, University of Bielsko-Biała,  
Willowa 2, 43-309 Bielsko-Biała, Poland

[mkordos@ath.bielsko.pl](mailto:mkordos@ath.bielsko.pl)

<sup>2</sup> Department of Industrial Informatics, Silesian University of Technology,  
Akademicka 2A, 44-100 Gliwice, Poland

[marcin.blachnik@polsl.pl](mailto:marcin.blachnik@polsl.pl)

**Abstract.** Genetic algorithms (GA) are a useful tool for a broad range of optimization problems, frequently allowing to find a better solution than classical methods. However, they suffer from two problems: high computational complexity related to fitness function evaluation and decrease in performance when the chromosome gets too long. In the paper we propose a solution which utilizes GA for efficient training set size reduction in regression tasks. To overcome GA limitations we use fuzzy c-means (FCM) clustering algorithm to reduce the size of the chromosome and also to solve the clusters border problem, by allowing the clusters to overlap. Finally the individual solutions obtained by each GA application are aggregated by applying the concept of ensembles of instance selection. This allows as to improve the selection results in terms of the accuracy-compression balance.

**Keywords:** Instance selection · Genetic algorithms · Clustering · Ensembles

## 1 Introduction

Instance selection is an important element in machine learning tasks [5]. It allows for training size reduction by eliminating the training samples which do not bring any relevant information to the training process. It also increases prediction performance by removing noisy instances and outliers. However, instance selection is an NP-hard problem and it is impossible to examine all the combinations of selected sets in order to find the best one.

Typical instance selection algorithms such as CNN, ENN, RNN, DROP family, ICF, or HMN family [6] apply some kind of best first search strategy or greed search, what leads to local optima. To overcome this limitation several other solutions were proposed, as evolutionary based methods [8] and ensembles of instance selection [3]. However, as our experiments and the experiments of other authors showed [8,2], GA-based instance selection can frequently produce better results than the classical methods.

In our method of GA-based instance selection, the process consists of two steps. In the first step the distance matrix is calculated and sorted. In the second step the GA-based optimization is executed, utilizing the pre-computed distance matrix. (The distance matrix contains distances between each pair of points in the training set, which are used by k-NN to calculate the *rmse* inside the instance selection process.) This allows obtaining the value of the fitness function very efficiently by reading only a few entries from the matrix. Also most of classical instance selection algorithms need to calculate and sort the distance matrix and for that reason the computational complexity of our method is comparable to that of the classical methods, and the most expensive is step one, which is of order  $O(n^2)$ .

An important issue of GA is the decrease of efficiency with the increase of chromosome length. As Goldberg wrote [7], genetic algorithms work "by building short, low order, and highly fit schemata (blocks), which are recombined (crossed over), and re-sampled to form strings of potentially higher fitness". When the solution of the problem is directly optimized by GA, it can be enough to increase the number of iterations. However, in instance selection one of the criterion is data size reduction of the training set and the second one is minimization of the prediction error (usually *rmse*) on the test set. Yet, the test set is unknown at the instance selection time, so the prediction error on the training set is temporarily substituted for this criterion (similarly as in classifier learning). Thus by using too many GA iterations, the over-fitting starts within individual blocks, while in other parts of the chromosome the algorithm still requires more iterations.

## 2 Method

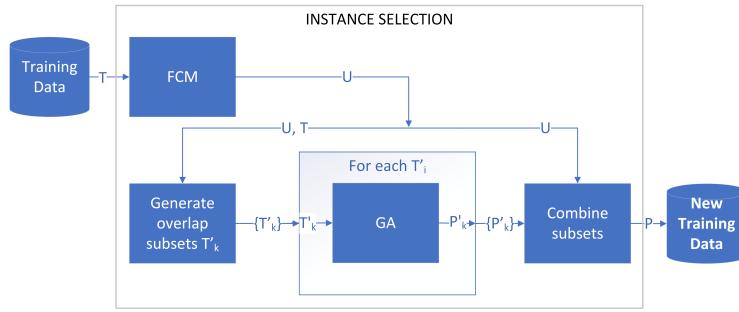
To remedy both of these problems, we apply data partitioning [1]. It allows to run the optimization in subspaces of the dataset for an optimal number of iterations. Data partitioning (clustering) is a more effective solution in regression problems, where the changes tend to be more smooth and more equally distributed in data space than in classification tasks.

Each cluster  $\mathbf{T}'_k$  contains the instances of the training set which are closest to the cluster center  $\mathbf{p}_k$ . This allows to perform independent instance selection  $\mathbf{P}'_k = \text{InstanceSelection}(\mathbf{T}'_k)$  in each of  $c$  subsets, and then a simply aggregation of all  $\mathbf{P}'_k$  constituting superset  $\mathbf{P} = \bigcup_{0 < k < c} \mathbf{P}'_k$ .

However, crisp clustering such as k-means faces the problem of border effect, where some closest neighbors of the instances from one cluster belong to another cluster and thus do not take part in determining this point output by k-NN used within the instance selection process. To overcome this limitation we use Fuzzy C-means (FCM) clustering [4], which not only splits the data into subsets, but also provides information of the fuzzy membership function (how strong particular instance belongs to given cluster). Each instance can belong to more than one cluster. We define a so called threshold  $\eta$ . If an instance has larger membership value in a given cluster than the threshold it is included into the cluster, otherwise it is excluded.

At the instance selection phase, GA is applied to each cluster  $\mathbf{T}'_k$  independently and the the fuzzy membership is ignored. It is used again at the results aggregation phase where the subsets of selected instance denoted as  $\mathbf{P}'_k$  are combined into  $\mathbf{P}$ . Here, instead of a simple union operation we again use fuzzy membership to combine the votes of the clusters. The process can be seen as building an ensemble of instance selection methods, and only when the sum of collected votes weighted by the membership function is higher than the threshold  $\theta$  the sample is included in the final subset  $\mathbf{P}$ .

The parameters of the GA determine the properties of instance selection within each cluster, while the clustering and voting parameters determine the way of aggregating the partial results. The proposed instance selection method is rather stable and small changes of the parameters do not influence the final results in a noticeable way. The sketch of the process is presented in figure 1.



**Fig. 1.** The instance selection process

### 3 Experimental Evaluation

In the experiments we used genetic algorithms with population size of 96 individuals, crossover with 48 parents and 48 cross-over points, mutation probability 0.03. We used the number of clusters  $c = 0.2 * \sqrt{n}$  rounded to the nearest integer,  $fuzziness = 2.5 - 2\log(n)$  of the FCM methods and thresholds respectively  $\eta = 0.3$  and  $\theta = 0.4$ . The results presented in Table 1 were obtained exactly with the same settings for each test to ensure fair comparison. We performed the experimental evaluation using our own software, available from [www.kordos.com/pprai2019](http://www.kordos.com/pprai2019). Because of space limitation, in Table 1 we present only the results obtained on 10 regression datasets from the Keel Repository averaged over five times repeated 10-fold crossvalidation. The results represent prediction error denoted as  $rmse$  and retention rate denoted by  $ret$ . The columns called  $rmse^{FCM}$  and  $ret^{FCM}$  represent performances obtained with our new approach, and the columns denoted  $rmse$  and  $ret$  represent performances obtained without clustering. According to the Wilcoxon signed-rank test the obtained performances for our new method are significantly better ( $\alpha = 0.05$ ).

**Table 1.** Results of the experimental evaluation

| dataset    | $n$ (inst.) | attr. | $rmse$ | $rmse^{FCM}$ | $ret$ | $ret^{FCM}$ |
|------------|-------------|-------|--------|--------------|-------|-------------|
| stock      | 950         | 9     | 0.134  | 0.134        | 0.499 | 0.464       |
| laser      | 993         | 4     | 0.256  | 0.256        | 0.486 | 0.461       |
| concrete   | 1030        | 8     | 0.632  | 0.625        | 0.500 | 0.458       |
| plastic    | 1650        | 2     | 0.615  | 0.600        | 0.483 | 0.428       |
| quake      | 2178        | 3     | 1.309  | 1.272        | 0.485 | 0.430       |
| abalone    | 4177        | 8     | 0.888  | 0.852        | 0.493 | 0.390       |
| delta-ail  | 7128        | 5     | 0.723  | 0.716        | 0.493 | 0.423       |
| california | 20640       | 8     | 0.676  | 0.669        | 0.497 | 0.486       |
| house      | 22784       | 16    | 0.885  | 0.869        | 0.672 | 0.641       |
| mv         | 40767       | 10    | 0.235  | 0.223        | 0.677 | 0.668       |

## 4 Conclusions

We presented a GA-based instance selection for regression tasks, which uses the concept of both: splitting the data and ensembles. As the experimental evaluation showed, this improved the results allowing as well for lower  $rmse$  on the test set as for smaller size of the training set. It also accelerated the calculations, decreasing the complexity for big datasets approximately from  $O(n^2)$  to  $O(cn)$  for clustering and  $O((\beta n/c)^2)$  for instance selection within clusters. This is obviously an initial approach and there are still many issues to further investigate and improve, nevertheless the current results are quite promising.

## References

- Ali, F.A., A.N.: Differential evolution algorithm with space partitioning for large-scale optimization problems. *Intell. Syst. Appl.* **11**, 49–59 (2015)
- Antonelli, M., Ducange, P., Marcelloni, F.: Genetic training instance selection in multiobjective evolutionary fuzzy systems: A coevolutionary approach. *Fuzzy Systems, IEEE Transactions on* **20**(2), 276–290 (April 2012)
- Blachnik, M.: Ensembles of instance selection methods. a comparative study. *International Journal of Applied Mathematics and Computer Science* **29**(1) (2019)
- Dunn, J.: A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*. **3**, 32–57 (1973)
- García, S., Luengo, J., Herrera, F.: Tutorial on practical tips of the most influential data preproc. algorithms in data mining. *Knowledge-Based Systems* **98**, 1–29 (2016)
- García, S., Derrac, J., Cano, J.R., Herrera, F.: Prototype selection for nearest neighbor classification: Taxonomy and empirical study. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **34**(3), 417–435 (2012)
- Goldberg, D.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley (1989)
- Kordos, M., Lapa, K.: Multi-objective evolutionary instance selection for regression tasks. *Entropy* **20**(10), 746 (2018)

# Evaluation of Musical Data Representation for Music Information Retrieval<sup>\*</sup>

Mariusz Kleć<sup>[0000-0003-4103-2597]</sup>, Krzysztof Marasek<sup>[0000-0003-1344-3524]</sup>, and Krzysztof Szklanny<sup>[0000-0001-6540-1671]</sup>

Polish-Japanese Academy of Information Technology,  
Multimedia Department, Warsaw, Poland.  
(mklec, kmarasek, kszkannya)@pjwstk.edu.pl

**Abstract.** The paper describes the evaluation of various audio representations for music classification, with the respect of first CNN layer. The results will contribute in building multi-representation model, ready to be used for various Music Information Retrieval (MIR) tasks - directly or in the process of transfer learning. The first results from the experiments towards this direction are described in this paper.

**Keywords:** Scattering Wavelet Transformation · Genre Recognition · Convolution Neural Networks · BiLSTM.

## 1 Introduction

Convolutional Neural Networks (CNN), are very successful in image recognition these days. Their performance exceeds the ability to recognize images by people [10, 11]. It was possible mainly due to the ability of Deep Learning techniques (DL) to discover hidden features from raw data (pixels). However, DL with raw audio data is computationally awkward. In this aspect, the right choice of audio representation for a specific task is significant.

The audio needs to be partitioned using windowing techniques. It leads to smoothing signal characteristics what might affect the results. Nevertheless, the community of speech processing use Fourier-based Mel-frequency Cepstral Coefficients (MFCC) with short window lengths (20-40ms). The MFCC is robust to small intra-class changes and provides sufficient spectral envelope to solve speech recognition task with prominent successes [12, 7]. However, the musical signal is much more locally nonstationary, especially in the case of signals that contain the mix of many instruments.

The author believes that tuning the audio representation with the first layer can introduce improvements to Music Genre Recognition (MGR) at the last layer of the deep architecture. The following experiments are to validate this thesis.

---

<sup>\*</sup> Supported by organization Polish-Japanese Academy of Information Technology.

## 2 The First Phase of Experiments

The purpose of the first phase of experiments is the inspection of how various signal representations affect the performance of MGR. Authors chose five audio representations that seemed to be relevant for music processing: Spectrogram (SPT), Scattegirng Wavelet Transformation (SWT) [1], Autocorrelogram (AUT) [5], Chromagram (CHR) [8] and Self-Similarity matrix (SSL) [4]. One thousand songs from GTZAN<sup>1</sup> data-set were encoded with each representation, taking into account the following window lengths (with half overlapping):  $W_L = \{0.0925s, 0.185s, 0.37s, 0.555s, 0.74s, 0.925s, 1.11s, 1.295s, 1.48s, 1.665s, 1.85s, 2.035s, 2.22s, 2.405s, 2.59s, 2.775s, 2.96s\}$ . The model for MGR was simple one convolutional layer and no pooling (in order not to discard the spatial information about the representation components). It was used in five cross-validation tests to derive the final accuracy.

Let  $W_L$  denotes the window length of the audio representation and  $F_W$  the filter (kernel) width in the convolution layer. Since the kernel performs convolution operations over the input, the original signal length  $S_L$  that it covers depends on  $F_W$  and the window length  $W_L$ . The kernel performed the convolution along time dimension with the stride equal one audio frame. It means that the created feature map is also one-dimensional. Finally, fully-connected layer performed MGR.

The experiments took into account 32 values of  $F_W = 1, 3, 6, \dots, 64$ , together with each value of  $W_L$ . This combination covers the  $S_L$  from 0.0925s up to 5.92s. However, all these combinations influence the capacity of the model (tendency to overfit the data) as produce different number of training examples. Therefore, in order to deliver a fair comparison, the capacity was regulated by keeping the number of learnable parameters (number of filters) equal in each case of  $W_L$ . The models have been optimised using Adaptive Moment Estimation (ADAM) until the validation loss has not been improving for the previous eight epochs. These experiments were to research the following:

1. How the accuracy of MGR change using different audio representations.
2. How the accuracy of MGR change when using different  $W_L$  and  $F_W$ .
3. Whether the signal length  $S_L$ , that  $W_L$  and  $F_W$  covers together, impacts the result of MGR.

### 2.1 Results For the First Phase

The results confirmed the expectation from SWT [1] to give significantly higher average accuracy compared to others (SWT: 66%, SPT: 47%, AUT: 37%, CHR: 26%, SSL: 35%). The highest results with SWT were possible to obtain when  $W_L$  was over 1s and short  $F_W$  (1 to 3 audio frames). The highest value was 81%. Additionally, the strong negative linear correlation between the accuracy and  $F_W$  was observed in this case (-0.93). All the other representations showed

---

<sup>1</sup> <http://marsyas.info/downloads/datasets.html>

similar correlation characteristics. However, the AUT was different. The highest results were possible to obtain using short  $W_L$  and longer  $F_W$ . Additionally, instead of linear relationship, results seem to go along polynomial functions of the second order.

Finally, to research the effect of  $S_L$  (regardless the  $F_W$  and  $F_L$ ) on accuracy, the  $S_L$  were divided into three groups: A=[0.0925s-1.48s>, B=[1.48s-2.96s>, C=[2.96s-5.92s]. Next, an analysis of variance showed that the effect of  $S_L$  on accuracy was significant only in the case of SPT ( $F(2,90) = 17.57$ ,  $p=3.62e-07$ ) and SWT ( $F(2,75) = 10.42$ ,  $p=0.0001$ ). In all the other representations, the signal length does not influence the final results. Post hoc comparisons estimated the differences between groups A and C for SPT and SWT. They were -8% and -11% accuracy respectively. Based on the conclusion, the most effective settings for SWT was derived ( $W_L = 2.775s$  and  $F_W = 3$ ). These settings will be validated in the second phase of experiments.

### 3 The Second Phase of Experiments

In order to validate the findings, additional experiments were carried out with FMA-medium data-set [2]. It provides 25000, 30 sec. clips with unbalanced 16 genres. The authors of this data-set reports 63% accuracy as the highest result - further treated as a baseline. However, due to uneven class distribution, besides the accuracy, F1-score will also be reported.

First, the initial CNN layer (with 64 filters) was trained with the settings of SWT derived from the previous experiments. The activations from trained model formed a new data representation with the dimension of [64 x 20], where rows denote the feature maps from each filter. Next, this new representation (NRP) was used in the following experiments:

1. Training deep CNN architecture with NRP from scratch (201 layers), inspired by GoogLe Net [10], using 1-D convolution.
2. Training the same architecture with NRP, using 2-D convolution.
3. Training one BiLSTM layer with NRP.

The results are presented in the following Table.

| # of Experiment | F1-score | Recall | Precision | Accuracy (%) |
|-----------------|----------|--------|-----------|--------------|
| 0               | 0.411    | 0.418  | 0.403     | 61.8         |
| 1               | 0.372    | 0.354  | 0.391     | 62.5         |
| 2               | 0.419    | 0.395  | 0.446     | 62.6         |
| 3               | 0.414    | 0.407  | 0.419     | <b>63.9</b>  |

**Table 1.** The results from the second phase of experiments. The values present the means of measure from each class. The zero experiment refers to results obtained from the first CNN layer from which NRP was derived.

## 4 Conclusions and Future Plans

Comparing the results with the baseline (63% accuracy in [2]) we can conclude that the settings of SWT, evaluated in the first experiment, performed better when processed by BiLSTM network (64%). It is worth noting that the baseline was obtained using SVN applying nine feature sets. The experiments in this paper used the same division into training (22427 songs) and test-set (2573 songs). However, it is possible to find in literature much better results for FMA-medium [9, 6]. But they describe the results from the Challenge [3] where all 25K songs was used for training and test-set was provided by the organizer. Therefore the author plans to compare different approaches of combining audio representations (including CNN, BiLSTM, Autoencoders and XGBoost) in various publicly available MIR benchmarks. Additionally, the goal is to build and evaluate big corpus of musical pieces, including balanced, rich and researched genre taxonomy. The corpus will contribute for training deep, multi-representation model, ready to be used directly in various MIR tasks or in the process of transfer learning.

## References

1. Andén, J., Mallat, S.: Deep scattering spectrum. *IEEE Transactions on Signal Processing* **62**(16), 4114–4128 (2014)
2. Defferrard, M., Benzi, K., Vandergheynst, P., Bresson, X.: Fma: A dataset for music analysis. arXiv preprint arXiv:1612.01840 (2016)
3. Defferrard, M., Mohanty, S.P., Carroll, S.F., Salathé, M.: Learning to recognize musical genre from audio. arXiv preprint arXiv:1803.05337 (2018)
4. Foote, J.: Visualizing music and audio using self-similarity. In: ACM Multimedia (1). pp. 77–80 (1999)
5. Kale, H., Maye, S.L.: Autocorrelation of a sound signal. IOSR Journal of Electrical and Electronics Engineering (IOSE-JEEE) pp. 50–53 (2014)
6. Kim, J., Won, M., Serra, X., Liem, C.: Transfer learning of artist group factors to musical genre classification. In: Companion Proceedings of the The Web Conference 2018. pp. 1929–1934. International World Wide Web Conferences Steering Committee (2018)
7. Korzinek, D., Marasek, K., Brocki, L., Wołk, K.: Polish read speech corpus for speech tools and services. arXiv preprint arXiv:1706.00245 (2017)
8. Lerch, A.: An introduction to audio content analysis: Applications in signal processing and music informatics. Wiley-IEEE Press (2012)
9. Murauer, B., Specht, G.: Detecting music genre using extreme gradient boosting. In: Companion Proceedings of the The Web Conference 2018. pp. 1923–1927. International World Wide Web Conferences Steering Committee (2018)
10. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–9 (2015)
11. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826 (2016)
12. Wolk, K., Marasek, K.: Survey on neural machine translation into polish. In: International Conference on Multimedia and Network Information System. pp. 260–272. Springer (2018)

# **SAFAIR: Secure and Fair AI Systems for Citizens**

Michał Choras<sup>1,2</sup>, Marek Pawlicki<sup>1,2</sup>, and Rafał Kozik<sup>1,2</sup>

<sup>1</sup> ITTI Sp. z o.o., Poland ,

<sup>2</sup> UTP University of Science and Technology, Bydgoszcz, Poland  
chorasm@utp.edu.pl

**Keywords:** Artificial Intelligence · Security · Adversarial Machine Learning · Fairness · Explainability

## **1 Towards Security and Fairness in AI systems and Machine Learning**

Recent advances in machine learning (ML) and the surge in computational power have opened the way to the proliferation of Artificial Intelligence (AI) in many domains and applications. With the real-world applications of AI came the realization that its security requires immediate attention. Malicious users, called 'Adversaries' in the AI world, can skilfully influence the inputs fed to the AI algorithms in a way that changes the classification or regression results. Regardless of the machine learning's ubiquity, the awareness of the security threats and ML susceptibility to adversarial attacks is fairly uncommon. Apart from security, another aspect that requires attention is the explainability of ML and ML-based decision systems. Many researchers and systems architects are now using deep-learning capabilities to solve AI/ML tasks. However, in most cases, the results are provided by algorithms without any justification.

Therefore, the SAFAIR Programme (Secure and Fair AI Systems for Citizens) of the H2020 SPARTA project focuses on security, explainability, and fairness of AI/ML systems, especially in the cybersecurity domain. Both the SPARTA project and the SAFAIR Programme have started in 2019, and the results are expected by 2022.

## **Acknowledgement**

This work is funded under the SPARTA project, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 830892.

# MDFS – a statistical filter for multivariate interactions

Krzysztof Mnich<sup>2[0000–0002–6226–981X]</sup>, Witold R. Rudnicki<sup>1,2,3[0000–0002–7928–4944]</sup>, and Radosław Piliszek<sup>2[0000–0003–0729–9167]</sup>

<sup>1</sup> Institute of Informatics, University of Białystok, Białystok, Poland

<sup>2</sup> Computational Center, University of Białystok, Białystok, Poland

<sup>3</sup> Interdisciplinary Centre for Mathematical and Computational Modelling,  
University of Warsaw, Warsaw, Poland

**Abstract.** Identification of informative variables in an information system is often performed using univariate filtering procedures that discard information about interactions between variables. Here we present an algorithm that performs identification of informative variables taking into account synergistic interactions between multiple descriptors and the decision variable. The algorithm is implemented as an R language package **MDFS** (Multi-Dimensional Feature Selection).

**Keywords:** feature selection · exhaustive search · weak relevance.

## Background

Identification of variables that are related to the decision variable is an important step in data analysis. The simplest and fastest way is to perform a test of statistical association between each descriptive variable and the decision variable. However, such univariate filters do not detect synergistic interactions between variables. Here we introduce an R package implementing a filter based on information theory. The algorithm can identify synergistic relevant variables by performing an exhaustive search of low-dimensional combinations of variables.

## 1 Theory

The key notion used in the algorithm is *weak relevance*, introduced by Kohavi and John [4]. A variable  $x_i \in X$  is weakly relevant if there exists such a subset of variables  $S \subset X : x_i \notin S$  that extending the subset  $S$  with the variable  $x_i$  increases information about the decision variable  $y$ . Mnich and Rudnicki [5] introduced the notion of  $k$ -weak relevance, that restricts the original definition by Kohavi and John to  $(k - 1)$ -element subsets  $S$ . Our aim was to identify all the  $k$ -weakly relevant variables for a given  $k$ .

The algorithm implements the definition of  $k$ -weak relevance directly by exploring all possible  $k$ -tuples of variables  $x_i \cup S^{k-1} \equiv x_i \cup \{x_{m_1}, x_{m_2}, \dots, x_{m_{k-1}}\}$  for  $k$ -dimensional analysis. For each  $k$ -tuple the conditional mutual information

is computed as a measure of relevance of the variable  $x_i$  with respect to the remaining ones.

$$I(y; x_i | S^{k-1}) = H(y|S^{k-1}) - H(y|x_i, S^{k-1}), \quad (1)$$

where  $H$  is a (conditional) information entropy.

The test statistic for a variable  $x_i$  is computed as a maximum value of  $I(y; x_i | S^{k-1})$  over all the subsets  $S^{k-1}$ .

$$I_{\max}^k(x_i) = \max_{S^{k-1} \subset X} I(y; x_i | S^{k-1}) \quad (2)$$

For  $k = 1$ ,  $I_{\max}^k$  reduces to the mutual information between  $x_i$  and the decision variable, i.e. to the well-known Fisher's  $G$ -test statistic.

The values of entropy are computed directly, assuming that both explanatory variables and the decision variable are discrete:

$$H(y|x_1, \dots, x_k) = - \sum_{d=1}^{c_y} \sum_{i_1=1}^{c_{x_1}} \dots \sum_{i_k=1}^{c_{x_k}} p_{i_1, \dots, i_k}^d \log(p_{i_1, \dots, i_k}^d), \quad (3)$$

where the value of  $p_{i_1, \dots, i_k}^d$  is computed as

$$p_{i_1, \dots, i_k}^d = \frac{N_{i_1, \dots, i_k}^d + \beta^d}{\sum_d (N_{i_1, \dots, i_k}^d + \beta^d)}. \quad (4)$$

$N_{i_1, \dots, i_k}^d$  is the count of class  $d$  in a  $k$ -dimensional voxel with coordinates  $i_j$ ;  $\beta^d \simeq 1$  is a pseudocount corresponding to class  $d$ , supplied by the user.

The null-hypothesis distribution for conditional mutual information (1) computed in this way is well-known. For a sample big enough,  $2NI(y; x_i | S^{k-1})$  (where  $N$  is a sample size) follows the  $\chi^2$  distribution. This allows to evaluate the null-hypothesis distribution of  $I_{\max}(x_i)$  up to one unknown parameter:

$$p(2NI_{\max}^k(x_i) < \alpha) = [p_{\chi^2}(2NI_{\max}^k(x_i) < \alpha)]^\gamma, \quad (5)$$

where  $\gamma$  corresponds to the number of independent tests for each variable  $x_i$ .

The parameter  $\gamma$  is estimated by fitting the theoretical distribution to the irrelevant part of the data. The recommended way to do that is to extend the data set by a number of permuted variables, that are irrelevant by design.

The eventual result of the procedure is a  $p$ -value for each variable  $x_i$ . Since many variables are investigated, the  $p$ -values should be adjusted using well-known FWER [3] or FDR [1] control technique.

## 2 Implementation

The algorithm is implemented as an R language package MDFS [6]. The package consists of two main parts. One is an R interface to two computational engines.

These engines utilise either CPU or NVIDIA GPU and are implemented in standard C++ and in CUDA C, respectively. Either computational engine returns the  $I_{\max}^k$  distribution for a given dataset. The other part is a toolkit to analyse results. It is written entirely in R.

The current implementation is limited to a binary decision variable  $y \in \{0, 1\}$  and up to 5-dimensional analysis. It is dedicated to analysis of data sets containing continuous explanatory variables  $x$ , so the first step is the discretisation of them.

The discretisation is performed using randomised rank-based approach. The values of the variable are split into the approximately equiprobable categories, with some random dispersion. By default, multiple random discretisations are performed and  $I_{\max}^k$  is computed as a maximum over them.

### 3 Results

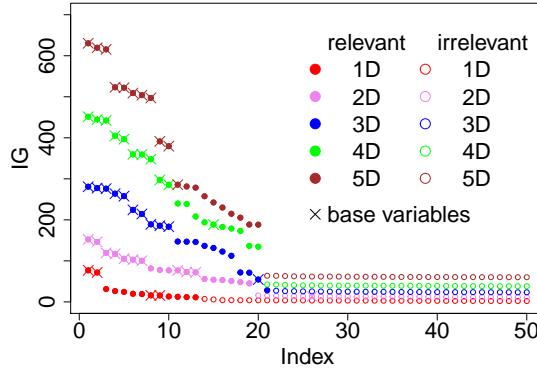
For demonstration of the **MDFS** package we used the training subset of the well-known Madelon dataset. It is an artificial set with 2000 objects and 500 variables. The decision was generated using a 5-dimensional random parity function based on variables drawn from normal distribution. The next fifteen variables were obtained as linear combinations of the 5 input ones (6 of them are almost exact copies of 4 of the base variables) and the remaining 480 variables were drawn randomly from the normal distribution. The data set can be accessed from the UCI Machine Learning Repository [2].

We conducted the analysis by **MDFS** in all possible dimensionalities using both CPU and GPU versions of the code. Additionally, a standard  $t$ -test was performed for reference. We examined computational efficiency of the algorithm and compared the results obtained by performing analysis in varied dimensionalities. The execution time of the tests are shown in Table 1. The use of GPU allows to perform up to 5-dimensional tests in a reasonable time.

|     | <i>t</i> -test | 1D    | 2D    | 3D   | 4D     | 5D      |
|-----|----------------|-------|-------|------|--------|---------|
| CPU | 0.01s          | 0.01s | 0.44s | 42s  | 1h:58m | 249h    |
| GPU | -              | -     | 0.23s | 0.2s | 9.8s   | 59m:37s |

**Table 1.** Execution times of MDFS tests for the Madelon dataset.

The values of the test statistic for various tests are shown in Fig. 1. The univariate statistic test omitted some of the relevant variables, while 3- and more-dimensional filters detected all of them. Note that the base variables appeared as the strongest ones in the 5-dimensional test.



**Fig. 1.** Test statistic values obtained by the MDFS algorithm using 1-, 2-, 3-, 4-, and 5-dimensional variants of the algorithm for the Madelon dataset.

#### 4 Conclusions

We have introduced a new package for identification of informative variables in multidimensional information systems which takes into account interactions between variables. The implemented method is significantly more sensitive than the univariate statistic tests when interactions between variables are present in the system. When applied to the well-known five-dimensional problem, i.e. Madelon dataset, the method not only discovered all relevant variables but also produced the correct estimate of their relative relevance.

#### References

1. Benjamini, Y., Hochberg, Y.: Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)* **57**(1), 289–300 (1995). <https://doi.org/10.1111/j.2517-6161.1995.tb02031.x>
2. Dheeru, D., Karra Taniskidou, E.: UCI machine learning repository (2017), <http://archive.ics.uci.edu/ml>
3. Holm, S.: A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics* **6**(2), 65–70 (1979). <https://doi.org/10.2307/4615733>
4. Kohavi, R., John, G.H.: Wrappers for feature subset selection. *Artif. Intell.* **97**(1-2), 273–324 (Dec 1997). [https://doi.org/10.1016/S0004-3702\(97\)00043-X](https://doi.org/10.1016/S0004-3702(97)00043-X)
5. Mnich, K., Rudnicki, W.R.: All-relevant feature selection using multidimensional filters with exhaustive search. *CoRR* **abs/1705.05756** (2017), <http://arxiv.org/abs/1705.05756>
6. Piliszek, R., Mnich, K., Migacz, S., Tabaszewski, P., Sułecki, A., Polewko-Klim, A., Rudnicki, W.: MDFS: MultiDimensional Feature Selection in R. *The R Journal* (2019). <https://doi.org/10.32614/RJ-2019-019>, <https://doi.org/10.32614/RJ-2019-019>

# Prediction of Drug-induced Liver Injury using different integration techniques

Wojciech Lesiński<sup>1[0000-0001-6628-9466]</sup>, Agnieszka Kitlas Golińska<sup>1[0000-0001-8737-765X]</sup>, Krzysztof Mnich<sup>2[0000-0002-6226-981X]</sup>, and Witold R. Rudnicki<sup>1,2,3[0000-0002-7928-4944]</sup>

<sup>1</sup> Institute of Informatics, University of Białystok, Białystok, Poland

<sup>2</sup> Computational Center, University of Białystok, Białystok, Poland

<sup>3</sup> Interdisciplinary Centre for Mathematical and Computational Modelling, University of Warsaw, Warsaw, Poland

**Abstract.** Drug-induced liver injury (DILI) is one of the primary problems in drug development. Early prediction of DILI, based on the chemical properties of compounds and experiments performed on cell lines, may significantly reduce the cost of clinical trials. The current study is aimed at building predictive models of DILI. Several models were obtained using both chemical properties of compounds, as well as gene expression levels in drug-affected cell-lines. Individual models were then integrated using Super Learner approach. The resulting composite model has a significantly improved accuracy (AUC=0.74, MCC=0.32). The model allows for the division of compounds into low-risk and high-risk classes.

**Keywords:** gene expression · integration data · feature selection · machine learning · random forest.

## Background

Drug-induced liver injury (DILI) is a common problem in drug development since nearly all classes of medications can cause liver disease [2]. Current paper reports work in progress, that aims at creating predictive models for DILI. The models should provide estimates of risk of DILI for new compounds using both their chemical properties and gene expression profiles in candidate-drug-affected cancer cell lines.

The data for CAMDA 2019 CMap Drug Safety Challenge was used in this study. It consists of the gene expression profiles for the thirteen human cell lines: MCF7, PC3, A375, A549, ASC, HA1E, HCC515, HPEG2, HT29, NPC, PHH, SKB, VCAP. These cell lines were exposed to various doses of 1314 drug compounds and then 12328 gene expression profiles were obtained using L1000 Platform[7], developed for Connectivity Map [4] at the Broad Institute. Samples were collected using several doses of drugs and expression profiles were measured at three different times, however, only single measurement per compound was used in the current study. Additionally the chemical structure of drug compounds

was described by SMILES (Simplified Molecular-Input Line-Entry System) [8]. The DILI classification is provided for 233 of these compounds, using 2 classification schemes. In the first one four classes are defined: *most DILI concern* – 39 compounds, *less DILI concern* – 90 compounds, *ambiguous DILI concern* – 50 compounds, *no DILI concern* – 54 compounds. In the second one classes 2. and 3. are merged into a single class *less DILI concern*.

## Materials and Methods

The aim of the study was to develop a robust approach to building predictive models for DILI using statistical analysis and machine learning. To this end we concentrated on a subset of data provided by challenge organisers, namely the molecular properties and gene expression profiles for 13 cell lines treated with these 233 compounds for which their DILI classification was provided.

The modelling approach is based on the following general protocol:

- Split the data into training and validation set;
- Identify informative variables in the training set;
- Select variables for model building;
- Build model on training set;
- Estimate model's quality both on training and validation set,

The procedure outlined above is cast within the 10 repeats of the 10-fold cross-validation scheme. The predictive models were build using Random Forest (RF) algorithm [1].

Both types of data contain very large number of descriptive variables - 12328 in the case of gene expression profiles and 1613 molecular descriptors, hence the selecting only informative variables for model building is very important. The identification of informative variables was performed with the help of two methods, Welch t-test for differences in sample means and multidimensional filter based on information theory developed in our laboratory [5, 6] and implemented in the R package *MDFS*.

The initial modelling performed on the original DILI classifications did not result in the predictive models. Therefore we decided to use aggregated classification scheme in which substances are either *DILI-concern* (179 compounds from original classes 1, 2 and 3) and *no DILI concern* (54 compounds from the original class 4). Such a split results in the unbalanced data set, hence the usage of robust performance measures was crucial.

Statistic tests exhibited very weak dependence between particular features and the response variable. However, the false discovery rate analysis suggested, that about a half of the 100-200 top-rated variables should be relevant. Therefore, we simply used 100 highest-scoring variables (after removal of strongly correlated ones) to build predictive models. The experiments were performed in three stages: first we built individual models using single source of data, then we explored different methods for integration of data from different sources, and finally we used generalised Super Learner approach [3] to obtain final model. We

**Table 1.** AUC for combined predictive models

| Method                 | AUC         |           |                           |                          |
|------------------------|-------------|-----------|---------------------------|--------------------------|
|                        | Internal CV | Nested CV | Std. dev. due to sampling | Std. dev. on unseen data |
| Best single result     | 0.70        | 0.66      | 0.03                      | 0.04                     |
| Mean of all results    | 0.68        | 0.69      | 0.01                      | 0.04                     |
| Mean of 5 best results | 0.74        | 0.74      | 0.02                      | 0.04                     |
| Linear combination     | 0.75        | 0.72      | 0.02                      | 0.04                     |
| Random Forest          | 0.84        | 0.71      | 0.03                      | 0.04                     |

have tested several methods for combination of models within Super Learner, including linear combination, average of best N models and Random Forest.

## Results and Discussion

### Individual cell lines.

The best results were obtained for MCF7 cell line with  $AUC = 0.61$  and  $MCC = 0.19$  measured in the fully cross-validated manner. Additionally the results obtained for VCAP, A549, HA1E, HCC515 and SKB cell lines suggest weak but non-random association with DILI signal. Results for other cell lines were very weak and not significantly different from random.

The best results from individual data source was obtained for a model built on chemical descriptors of compounds derived from their molecular structure ( $AUC = 0.66$  and  $MCC = 0.24$ ).

**Integration with chemical properties of drugs.** Feature selection on data sets containing both molecular and gene expression descriptors returns predominantly gene expression variables with small admixture of molecular descriptors, despite that model built on molecular descriptors is better than any model built on gene expression profiles. Interestingly, presence of the molecular descriptors improves predictions in most, but not all, cases. Best results ( $AUC = 0.69$ ,  $MCC=0.29$ ) were also in this case obtained for MCF7 cell line.

**Integration using Super Learner methodology** The composite model significantly outperforms any individual one, with  $AUC=0.74\pm0.04$ . The quality of the composite model was not changed when five most important variables are used, and was slightly decreased for composite model built on three variables ( $AUC=0.71\pm0.01$ ). For more result see Table 1.

The quality of final model is certainly not sufficient for predicting DILI status of any compound with good precision. Nevertheless it can be used to enrich results in given class, see Table 2. In this case we were obtained significant enrichment in no DILI class.

## Conclusions

Weak predictive models for DILI can be obtained using either gene expression profiles of some cell lines exposed to drug compounds or molecular properties of

**Table 2.** Enrichment of non-DILI compounds in low risk class

| Method                 | Enrichment |           |           |
|------------------------|------------|-----------|-----------|
|                        | no DILI    | less DILI | most DILI |
| Best single result     | 1.92       | 0.77      | 0.70      |
| Mean of all results    | 2.23       | 0.63      | 0.94      |
| Mean of 5 best results | 3.48       | 0.55      | 0.74      |
| Linear combination     | 3.00       | 0.63      | 0.67      |
| Random Forest          | 2.84       | 0.64      | 0.68      |

these compounds. Integration of gene expression profiles with chemical properties of drug compounds leads to slightly improved models. On the other hand the quality can be significantly improved when combination of individual models into a composite model based on Super Learner methodology is performed. Five cell lines with best individual models were used for best composite model with significantly improved predictive power. The results of this study suggest that search for cell lines that are predictive for DILI should be extended to more cell lines. The model developed in the current study is not suitable for individual prediction for a single compound, nevertheless it can be useful for discrimination between compounds possessing high and low risk of causing DILI. The predictive power of the model is limited by the composition of the dataset. It is strongly unbalanced with nearly 4:1 ratio between *DILI concern* and *no DILI concern* class. One should also note that significant overfitting is present in most methods of combining the results. A mean of 5 best models is both resistant to overfitting and best on external validation.

## References

1. Breiman, L.: Random forests. *Machine Learning* **45**, 5–32 (2001)
2. David S, H.J.: Drug-induced liver injury. *US gastroenterology and hepatology review* (6), 73 – 80 (2010)
3. van der Laan Mark J., Polley Eric C., H.A.E.: Super learner. *Statistical Applications in Genetics and Molecular Biology* **6**(1) (2007)
4. Lamb, J.: The connectivity map: a new tool for biomedical research. *Nature Reviews Cancer* (7), 54 – 60 (2007)
5. Mnich, K., Rudnicki, W.R.: All-relevant feature selection using multidimensional filters with exhaustive search. *arXiv preprint arXiv:1705.05756* (2017)
6. Piliszek, R., Mnich, K., Migacz, S., Tabaszewski, P., Sułecki, A., Polewko-Klim, A., Rudnicki, W.: MDFS: MultiDimensional Feature Selection in R. *The R Journal* (2019). <https://doi.org/10.32614/RJ-2019-019>, <https://doi.org/10.32614/RJ-2019-019>
7. Subramanian, A.: A next generation connectivity map: L1000 platform and the first 1,000,000 profiles. *Cell* **171**(6), 1437 – 1452 (2017)
8. Weininger, D.: Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences* **28**(1), 31–36 (1988). <https://doi.org/10.1021/ci00057a005>

# Weighted Context-free Grammar Induction— a preliminary report<sup>\*</sup>

Olgierd Unold[0000–0003–4722–176X] and Mateusz Gabor

Department of Computer Engineering  
Wrocław University of Science and Technology  
Wyb. Wyspiańskiego 27, 50-370 Wrocław, Poland  
[olgierd.unold@pwr.edu.pl](mailto:olgierd.unold@pwr.edu.pl)

**Abstract.** In this paper, we address the problem of inducing weighted context-free grammar on data given. The induction is performed by using a new model of grammatical inference, i.e., stochastic Grammar-based Classifier System. Some preliminary results over standard context-free languages are presented.

**Keywords:** Grammatical Inference, Weighted Grammar, Probabilistic Grammar, Implicit Negative Evidence, Contrastive Estimation

## 1 Introduction

Grammatical inference is a part of symbolic Artificial Intelligence and deals with the induction of formal structures like grammars or trees from data [1]. Among different types of grammars, the weighted context-free grammars (weighted CFG, WCFGs) or equally expressive probabilistic context-free grammars (PCFGs) play a special role and have found use in many areas of syntactic pattern matching.

The task of learning WCFGs/PCFGs from data consists of two subproblems: determining a discrete structure of the target grammar and estimating weighted/probabilistic parameters in the grammar. One of the few systems enabling learning both structure and grammar parameters is Grammar-based Classifier system (GCS), introduced in [3] and dedicated initially to learn crisp context-free grammar. GCS was extended to fuzzy version [4], and recently to stochastic one (stochastic GCS, sGCS).

In [6], two different algorithms to estimate probabilistic parameters of WCFG were compared. Both are the part of a new stochastic GCS, which is the subject of this paper.

## 2 Stochastic Grammar-based Classifier System

According to the principles of grammatical inference, sGCS receives as the input data set in the form of positive and negative labeled sentences, as the output the

---

\* The research was supported by the National Science Centre Poland (NCN), project registration no. 2016/21/B/ST6/02158.

WCFG is induced. The set of weighted CFG rules in Chomsky's normal form is the core of the system. The CKY parser checks if the given input sentence belongs to induced context-free grammar. To learn the structure of the grammar, i.e., the set of CFG rules, a genetic algorithm supported by a covering mechanism is applied. To estimate weighted parameters in the grammar, i.e., weights assigned to each grammar rule, the modified inside-outside (IO) method is used. The standard IO has been extended by Contrastive Estimation (IOCE) approach to deal with the sentences not belonging to the target language [6]. The correction module removes CFG rules with weights less than 0.001. The general architecture of sGCS is given in Fig. 1.

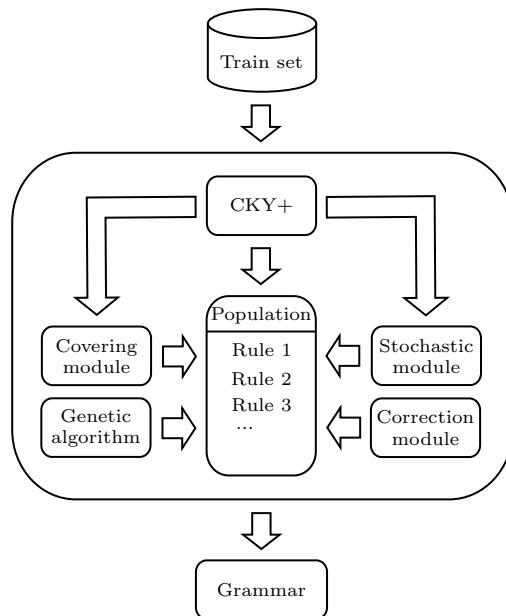


Fig. 1: The architecture of sGCS

### 3 Test study

This paper presents the preliminary test study over not-overlapping training and test sets, containing both positive and negative sentences taken from three context-free languages [2], i.e., **ab** - the language of all strings consisting of equal numbers of *as* and *bs*, **bra3** - the language of balanced parentheses, and **pal2** - palindromes over  $\{a, b\}$ . Metrics of training and test sets are given in Tab 1. The experiments were carried out according to the experimental protocol described in Algorithm 1, using jGCS library [5].

To evaluate the quality classifications, we use the classification results stored in a confusion matrix. The following four scores were defined as  $tp$ ,  $fp$ ,  $fn$ , and  $tn$ ,

**Algorithm 1** Experimental protocol

---

**Input:** Training set  
**Output:** Weighted context-free grammar

- 1: Run CKY algorithm with Covering (on training set)
- 2: **while** NOT stop criterion **do**
- 3:     Run genetic algorithm
- 4:     **for**  $i \leftarrow 1$  to IOCE iterations **do** ▷ IOCE iterations=8
- 5:         Run CKY with estimation method on training set
- 6:     **end for**
- 7:     Remove zero probabilities rules
- 8:     Run CKY on the test set
- 9:     Calculate the metrics
- 10: **end while**
- 11: Run the grammar correction module
- 12: **return** WCFG and results

---

| Set        | Size | Positive sentences | Negative sentences | Max. length of sentence |
|------------|------|--------------------|--------------------|-------------------------|
| Train ab   | 199  | 112                | 87                 | 12                      |
| Test ab    | 101  | 57                 | 44                 | 12                      |
| Train bra3 | 199  | 93                 | 106                | 12                      |
| Test bra3  | 101  | 47                 | 54                 | 12                      |
| Train pal2 | 199  | 118                | 81                 | 14                      |
| Test pal2  | 101  | 60                 | 41                 | 14                      |

Table 1: Train and test sets metrics

representing the numbers of true positives (the positive labeled test sentences regarded as positive sentences), false positives (the negative labeled test sentences regarded as positive sentences), false negatives (the positive labeled test sentences regarded as negative sentences), and true negatives (the negative labeled test sentences regarded as negative sentences), respectively. Based on the values stored in the confusion matrix, we calculate the widely used estimators: Sensitivity =  $tp/(tp+fn)$  and Specificity =  $tn/(tn+fp)$ .

Fig 2 shows the induction of individual languages, and Tab 2 summarizes the exemplary induced grammars together with the source grammars, i.e., the grammars applied to generate the positive labeled sentences. The negative sentences were generated randomly. Comparing induced structures with source ones indicates that the system was able to find all relevant grammar patterns. The languages ab and pal2 have reached Specificity and Sensitivity values equal to one fully finding the grammar for all sentences. Induction of bra3 achieved a slightly lower result, obtaining Sensitivity and Specificity around 0.97, not quite finding the full grammar.

The introduced stochastic Grammar-based Classifier system proved to be useful in weighted context-free grammar induction. The sGCS model is continuously under development, and numerous improvements and extensions are to

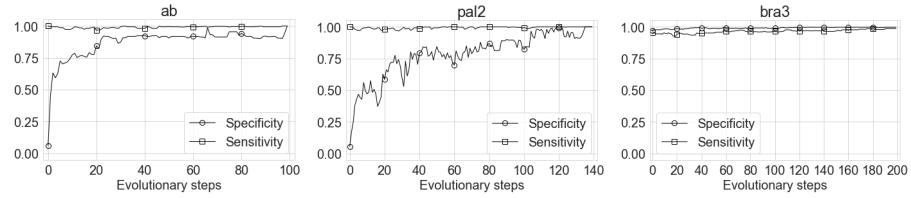


Fig. 2: Induction of ab, pal2 and bra3 languages

be implemented, as well as new fields of application, such as the induction of grammars describing structures of amyloid peptides.

|                    |  |
|--------------------|--|
| <b>ab</b>          | $A \rightarrow a, B \rightarrow b, S \rightarrow SS, S \rightarrow AB, S \rightarrow BA, A \rightarrow SA, A \rightarrow AS$   |
| <b>source ab</b>   | $A \rightarrow a, B \rightarrow b, S \rightarrow SS, S \rightarrow BF, S \rightarrow BA, S \rightarrow AB, S \rightarrow AD, F \rightarrow SA, D \rightarrow SB$   |
| <b>bra3</b>        | $A \rightarrow a, B \rightarrow b, C \rightarrow c, D \rightarrow d, E \rightarrow e, F \rightarrow f, S \rightarrow SS, S \rightarrow SF, S \rightarrow AS, S \rightarrow AD, S \rightarrow BE, S \rightarrow CS, S \rightarrow CF, B \rightarrow BS, C \rightarrow CF, D \rightarrow DA, F \rightarrow AD, F \rightarrow BE, F \rightarrow CF$ |
| <b>source bra3</b> | $A \rightarrow a, B \rightarrow b, C \rightarrow c, D \rightarrow d, E \rightarrow e, F \rightarrow f, S \rightarrow AD, S \rightarrow BE, S \rightarrow CF, S \rightarrow SS, S \rightarrow AS, C \rightarrow CS, B \rightarrow BS$   |
| <b>pal2</b>        | $A \rightarrow a, B \rightarrow b, S \rightarrow AA, S \rightarrow BB, A \rightarrow SA, B \rightarrow BS$   |
| <b>source pal2</b> | $A \rightarrow a, B \rightarrow b, S \rightarrow AA, S \rightarrow BB, S \rightarrow OA, S \rightarrow BH, O \rightarrow AS, H \rightarrow SB$   |

Table 2: Comparison of source and induced grammars

## References

1. de la Higuera, C.: Grammatical Inference: Learning Automata and Grammars. Cambridge University Press (2010)
2. Keller, B., Lutz, R.: Evolving stochastic context-free grammars from examples using a minimum description length principle. In: Workshop on Automatic Induction, Grammatical Inference and Language Acquisition (1997)
3. Unold, O.: Context-free grammar induction with grammar-based classifier system. Archives of Control Sciences 15(4), 681–690 (2005)
4. Unold, O.: Fuzzy grammar-based prediction of amyloidogenic regions. In: International Conference on Grammatical Inference. pp. 210–219 (2012)
5. Unold, O.: jGCS. <https://github.com/ounold/jGCS> (2019)
6. Unold, O., Gabor, M.: How implicit negative evidence improve weighted context-free grammar induction. In: International Conference on Artificial Intelligence and Soft Computing. pp. 595–606. Springer (2019)

# Robust Machine Learning protocol with estimation of biases

Aneta Polewko-Klim<sup>1</sup>[0000–0003–1987–7374], Wojciech Lesiński<sup>1</sup>[0000–0001–6628–9466], Krzysztof Mnich<sup>2</sup>[0000–0002–6226–981X], Radosław Piliszek<sup>2</sup>[0000–0003–0729–9167], Bogumił Sapinski<sup>3</sup>, and Witold R. Rudnicki<sup>1,2,3</sup>[0000–0002–7928–4944]

<sup>1</sup> Institute of Informatics, University of Białystok, Białystok, Poland

<sup>2</sup> Computational Center, University of Białystok, Białystok, Poland

<sup>3</sup> Interdisciplinary Centre for Mathematical and Computational Modelling, University of Warsaw, Warsaw, Poland

**Abstract.** Machine Learning predictive models for patients suffering different medical conditions are often developed using molecular descriptors. Generally these datasets consist of at most a few hundreds of cases that are described with tens of thousands (e.g. gene expression profiles) or even millions (e.g. genetic markers) of variables. In such cases modelling procedures are complicated and usually involve feature selection and model selection. Both steps introduce biases. Furthermore, additional biases are introduced by dividing a dataset into the training and validation subsets. These biases are rarely properly estimated and sometimes even completely neglected. In the current study we present a simple protocol that allows to build predictive models with robust estimates of the different biases introduced by the modelling procedure. The protocol is then applied to build predictive models for neuroblastoma patients.

**Keywords:** feature selection · cross-validation · bias estimation · robust modelling.

## Background

The modelling using machine learning methods generally consist of the following steps: selection of variables for model building; splitting data into training and validation sets; building model on the training set; estimating model's quality both on training and validation set. It is often performed within cross-validation framework, which allows to use entire dataset both as a training and validation sets. Each of the steps may introduce biases, in particular when the dataset used in the study has few objects – what is often the case in the biomedical studies. The biases arise at each modelling step – feature selection, model building and even splitting data between training and validation set. The size of the bias strongly depends on the number of objects in the dataset and balance of classes as well as on the strength of the signal in the data. The biases are strongest when the signal associated with individual variables is weak and the number of objects

in the least numerous class is small. Unfortunately it is very common situation in the biomedical research. The studies are often limited to a small number of patients not only due to the cost. In many cases the number of patients is limited either by the low frequency of cases or by the legal limitations to data sharing. This is exemplified by the dataset used in the current study, where the less numerous class consists of 38 cases.

The problem of overfitting of models built on molecular markers was recognised very soon after such datasets became available [1, 6]. Nevertheless, the rigorous estimates of the biases and variances are very often missing in the research articles published in the field. Two factors contribute to this unfortunate situation. Firstly, the researchers in the area lack the statistical background and settle to established practices, even if these practices are not based on a solid statistical ground. Another example is wide-spread misuse of p-values in life sciences [3, 7]. Secondly, there is a lack of tools that would help to build models in a way that allows robust and unbiased estimates of model errors without requiring user to have in-depth knowledge of statistical issues.

In the current study we present a protocol for building predictive Machine Learning models. It allows to estimate different biases and different contributions to the variance of metrics used for evaluation of the models' quality. The prototype implementation of the protocol in an R library has been also carried out using Random Forest classification algorithm [2, 4] and MDFS feature selection library [5]. Currently we work on generalised implementation of the interfaces to multiple classification algorithms and multiple feature selection algorithms. Also the extension of the library that will handle robustly biases due to optimisation of the parameters will be implemented.

## Materials and Methods

### Protocol

In protocol is designed to give estimates of biases and variances that arise due to: feature selection, internal variability of modelling algorithm, insufficient sample size, and variable composition of training and validation sets. In the current implementation it does not handle bias due to selection model parameters. The protocol consists of three stages. In the first stage, we perform several repeats of modelling procedure co using entire dataset. This stage is well suited to algorithms that provide unbiased estimate of performance on the training set - such as OOB estimates in Random Forest. In the case of fully deterministic algorithms only single run is necessary. In the second stage the feature selection step is performed once for entire dataset. Then model building is performed within  $N - times$  repeated  $K - fold$  cross-validation scheme. Finally, in the third stage entire modelling procedure is performed within the cross-validation scheme identical to that applied in the second stage. In each step the estimates of model quality metrics are collected and their average values and variances are collected.

The first step provides estimate of the internal variability of the pair of algorithms used for feature selection and building ML model. It also gives reference for estimation of negative bias of performance of cross-validation scheme that arises due to smaller size of the training set. The repeated cross-validation in the stage 2 and 3 allow for better estimate of the variance due to particular split between training and validation sets. Since the third step is performed within identical cross-validation scheme as the second step, the difference between their results arises entirely due to moving the feature selection step inside the cross-validation. This allows for estimate of the bias due to feature selection.

## Data

The datasets used in the current study were is a subset of data provided within Neuroblastoma Data Integration Challenge (<http://camda.info>). Genetic information for 498 patients was collected using profiling of gene expression (GE) by means RNA sequencing for 60 778 probes. The data collection procedures and design of experiments were described in the original studies, see [8] and references therein. The data is alternatively accessible in Gene Expression Omnibus (<https://www.ncbi.nlm.nih.gov/geo/>) with accession number GSE49711. Due to the design of the original study a cohort of 145 patients for whom additional information was available was studied separately using identical protocol.

## Results and Discussion

Several tests were performed using different different values of  $K$  in  $K - fold$  cross-validation, only a subset of results is presented below. The number of repeats was set at 50 to obtain reliable estimates of the variance. Three feature selection methods were applied – Welch t-test, followed by selection of the optimal set with LASSO method, as well as one- and two-dimensional variants of MDFS and twenty most important variables were used to build models. The summary of results for the  $10 - fold$  cross-validation for the both cohorts is presented in Table 1.

The OOB estimates of error are generally consistently upward biased for all stages. Moreover, the cross-validated results obtained at the stage 2 are biased in the similar way. One can observe small downward bias of MCC, due to decreased training set in the Stage 2 in comparison with stage 1. Interestingly, this bias is not present in the estimate of AUC. It is also clearly visible that cross-validation introduces significantly higher variance than internal variability of the classifier. What is more, variances are larger in the smaller set, although differences are not very big. Finally, large drop of both metrics is observed when models are constructed in a fully crop-validated manner. The upward bias in MCC varies between 0.33, for model built using t-test with LASSO for smaller dataset, and 0.55 for model built using MDFS-2D for larger dataset. The upward bias in AUC varies between 0.074, for model built using t-test with LASSO for smaller dataset, and 0.016 for model built using MDFS-1D for larger dataset. Interestingly, the AUC was within standard deviation for all fully cross-validated models, whereas statistically significant differences were observed for MCC.

**Table 1.** Model quality measured with MCC and AUC for neuroblastoma patients. MDFS-1D and MDFS-2D denote models built on variables selected using univariate and two-dimensional MDFS variants, respectively.

| FS metod      | 145 patients |           |       |           | 498 patients |           |       |           |
|---------------|--------------|-----------|-------|-----------|--------------|-----------|-------|-----------|
|               | MCC          | AUC       | MCC   | AUC       | MCC          | AUC       | MCC   | AUC       |
|               | Mean         | Std. Dev. | Mean  | Std. Dev. | Mean         | Std. Dev. | Mean  | Std. Dev. |
| Stage 1       |              |           |       |           |              |           |       |           |
| t-test        | 0.772        | 0.022     | 0.964 | 0.003     | 0.656        | 0.016     | 0.937 | 0.002     |
| MDFS-1D       | 0.785        | 0.005     | 0.945 | 0.003     | 0.474        | 0.017     | 0.882 | 0.002     |
| MDFS-2D       | 0.768        | 0.021     | 0.940 | 0.003     | 0.413        | 0.018     | 0.871 | 0.002     |
| Stage 2       |              |           |       |           |              |           |       |           |
| t-test (OOB)  | 0.765        | 0.011     | 0.963 | 0.001     | 0.651        | 0.007     | 0.936 | 0.001     |
| t-test (CV)   | 0.766        | 0.036     | 0.965 | 0.007     | 0.657        | 0.018     | 0.937 | 0.003     |
| MDFS-1D (OOB) | 0.763        | 0.006     | 0.944 | 0.002     | 0.465        | 0.006     | 0.882 | 0.001     |
| MDFS-1D (CV)  | 0.765        | 0.031     | 0.945 | 0.014     | 0.458        | 0.027     | 0.883 | 0.004     |
| MDFS-2D (OOB) | 0.743        | 0.009     | 0.947 | 0.001     | 0.495        | 0.006     | 0.889 | 0.001     |
| MDFS-2D (CV)  | 0.742        | 0.039     | 0.949 | 0.010     | 0.495        | 0.022     | 0.891 | 0.004     |
| Stage 3       |              |           |       |           |              |           |       |           |
| t-test (OOB)  | 0.817        | 0.012     | 0.972 | 0.002     | 0.615        | 0.009     | 0.928 | 0.002     |
| t-test (CV)   | 0.591        | 0.054     | 0.898 | 0.018     | 0.460        | 0.030     | 0.869 | 0.010     |
| MDFS-1D (OOB) | 0.766        | 0.009     | 0.945 | 0.002     | 0.482        | 0.008     | 0.886 | 0.001     |
| MDFS-1D (CV)  | 0.601        | 0.048     | 0.902 | 0.017     | 0.426        | 0.027     | 0.870 | 0.005     |
| MDFS-2D (OOB) | 0.745        | 0.011     | 0.945 | 0.002     | 0.470        | 0.010     | 0.885 | 0.002     |
| MDFS-2D (CV)  | 0.591        | 0.046     | 0.901 | 0.015     | 0.415        | 0.029     | 0.868 | 0.006     |

## References

1. Ambroise, C., McLachlan, G.J.: Selection bias in gene extraction on the basis of microarray gene-expression data. *Proceedings of the National Academy of Sciences of the United States of America* **99**(10), 6562–6566 (2002)
2. Breiman, L.: Random forests. *Machine Learning* **45**, 5–32 (2001)
3. Ioannidis, J.P.: Why most published research findings are false. *PLoS medicine* **2**(8), e124 (2005)
4. Liaw, A., Wiener, M.: Classification and regression by randomforest. *R News* **2**(3), 18–22 (2002), <http://CRAN.R-project.org/doc/Rnews/>
5. Piliszek, R., Mnich, K., Migacz, S., Tabaszewski, P., Sułecki, A., Polewko-Klim, A., Rudnicki, W.: MDFS: MultiDimensional Feature Selection in R. *The R Journal* (2019). <https://doi.org/10.32614/RJ-2019-019>, <https://doi.org/10.32614/RJ-2019-019>
6. Varma, S., Simon, R.: Bias in error estimation when using cross-validation for model selection. *BMC bioinformatics* **7**(1), 91 (2006)
7. Wasserstein, R.L., Lazar, N.A., et al.: The asa's statement on p-values: context, process, and purpose. *The American Statistician* **70**(2), 129–133 (2016)
8. Zhang, W., Yu, Y., Hertwig, F., Thierry-Mieg, J., Zhang, W., Thierry-Mieg, D., Wang, J., Furlanello, C., Devanarayanan, V., Cheng, J., et al.: Comparison of rna-seq and microarray-based models for clinical endpoint prediction. *Genome biology* **16**(1), 133 (2015)

# Recent Advances in Cross-Domain Sentiment Analysis of Polish Texts \*

Arkadiusz Janz<sup>1</sup>[0000-0002-9203-5520] and Jan Kocon<sup>1</sup>[0000-0002-7665-6896]

Wrocław University of Science and Technology, Wrocław, Poland  
`{arkadiusz.janz|jan.kocon}@pwr.edu.pl`

**Abstract.** Sentiment analysis is a hot research topic of Natural Language Processing with its main focus on emotive analysis of textual opinions. The task of sentiment recognition is highly domain-dependent, thus, there is a great need for designing the methods with decent domain adaptation abilities. In this paper we present a brief overview of existing datasets and methods connected with the task of cross-domain sentiment analysis in the area of natural language processing, with the special focus given to Polish. Presented work is mainly a result of two large scientific projects: CLARIN-PL and Sentimenti with their specific parts focused on sentiment recognition.

**Keywords:** sentiment analysis · cross-domain polarity recognition · machine learning

## 1 Background

Natural Language Processing (NLP) is a modern interdisciplinary computational science that links artificial intelligence (AI) with linguistics, psychology and philosophy to make the computers understand statements written by humans in natural language. One of the hot research topics in the area of NLP is Sentiment Analysis (SA). Sentiment Analysis is a process of identifying and categorising textual opinions in a way that provides a clear information about emotional load associated with analysed text. A basic task in sentiment analysis is to recognize the polarity of words, short phrases, sentences, or full documents. Usually the task of polarity recognition is limited to ternary classification case where the main aim is to determine if a text expresses a *positive*, *negative*, or *neutral* opinion. The research covers also the topics of subjectivity detection, aspect-based polarity recognition, and emotion identification.

### 1.1 Emotive Lexicons

Lexicons are an important, inherent part of sentiment analysis and opinion mining systems. There are three general approaches to compile sentiment lexicon i.e.

---

\* Work co-financed as part of the investment in the CLARIN-PL research infrastructure funded by the Polish Ministry of Science and Higher Education and in part by the National Centre for Research and Development, Poland, under grant no POIR.01.01.01-00-0472/16.

*corpus-based* approach, *dictionary-based*, and *manual* [13]. Manual approaches are laborious and time-consuming, so there is a great need for fast, automated methods designed to automatically generate sentiment lexicons, especially for low-resourced languages. One of the greatest problems of lexical analysis is connected with polysemy phenomenon, which means that the words can express different meaning depending on the context, as the context determines the meaning of word which is the main assumption of distributional hypothesis. Thus, the words can express different polarity and different emotions among their senses. To tackle the issue of polysemy the researchers decided to focus on building the lexicons for word senses and analyse the texts on sense-level rather than word-level.

One of the possible ways to construct a sense-aware sentiment lexicon is to use large sense inventories like wordnets (i.e. a dictionary-based approach). These kind of approaches generally aim at extending a small set of seed words with known polarity using lexical relations of a wordnet, mainly Princeton WordNet[5], e.g. hypernymy, synonymy, antonymy, etc. SentiWordNet[4] annotates senses with three scores in the range  $< 0, 1 >$ , describing how positive and negative they are. The annotation was mainly automatic, only 10% of adjectives were manually annotated and treated as a seed. The authors used machine learning approach to construct polarity recognition classifier and propagate the annotation of their seed over all wordnet senses. MLSentiCon[2] built upon the idea presented in SentiWordNet, but the initial seed was slightly expanded by taking WordNet-Affect[16] annotation and General-Inquirer lexicon[15]. Finally, the expanded seed was propagated using the same method as proposed in SentiWordNet. The main drawback of presented approaches is that they are based on a narrow, small initial seed, so the automatic process of emotive propagation introduces a lot of errors (e.g. the word 'be' is +0.125 in SentiWordNet).

| Measure             | BASE  | CPP-N        | CPP-S |
|---------------------|-------|--------------|-------|
| $Fscore_{Negative}$ | 75.52 | <b>79.91</b> | 79.81 |
| $Fscore_{Neutral}$  | 93.95 | <b>95.34</b> | 95.35 |
| $Fscore_{Positive}$ | 66.77 | <b>74.99</b> | 74.61 |

**Table 1.** F1-score (F) for separate classes of polarity. Baseline results (BASE) are compared to CPP-N and CPP-S.

To solve the issue of limited emotive seed for propagation a large-scale annotation process based on plWordNet[17] was initiated as a part of CLARIN-PL project. The senses in plWordNet-emo [17, 6] were annotated with following categories: *strong positive* (+m), *weak positive* (+s), *strong negative* (-m), *weak negative* (-s), *ambiguous* (amb), *neutral* (0). This annotation was expanded in [6] where the authors proposed the next version of plWordNet-emo. In [9, 10] the authors proposed a fully automated method called *Classifier-based Polarity Propagation* (henceforth CPP) based on plWordNet-emo. The classifier was trained using a rich set of features based on a very broad semantic context from a

wordnet (a subgraph of wordnet senses connected by lexico-semantic links) taking into account the distant neighbourhood of synsets and including an extended set of semantic relations. The results obtained with CPP method in comparison to baseline approaches are presented in Table 1.

## 2 Polarity Recognition in CLARIN-PL

The research on sentiment analysis of customer reviews was conducted in 2018 within CLARIN-PL<sup>1</sup> [14] and it consisted of the pilot stage and the main stage. The preliminary part of analysis involved 3,000 opinions from the Web. Each text was manually annotated by two annotators: a psychologist and a linguist. In the sentiment annotation the same set of tags was used as they appeared in plWordNet 3.0 emo.

Three different classifiers for the recognition tasks were selected: (1) logistic regression provided by fastText [7]; (2) sequential BiLSTM model, (3) BiLSTM augmented with *sentiment dictionary*, and (4) BERT transformer network. The logistic regression approach with pre-trained embeddings (here, a pre-trained model from [8] was used) is much faster both in training and testing than deep learning classifiers [7]. Although fastText is able to analyse all words in the context, it does not provide information about word order. BiLSTM with its sequential nature captures the order of words as they appear in text. Texts are divided into tokens and converted to corresponding word embedding vectors generated by fastText [1]. BERT was designed to provide a pre-trained deep bidirectional representations conditioning left and right context [3] simultaneously, therefore it achieves best performance on text fragments rather than just single sentences. Its architecture allows to fine-tune these representations by adding one additional output layer which suits the needs of given task. To test generalization abilities of modern classifiers in the area of sentiment analysis we decided to measure their performance in a multi-domain setting.

Here we present the results of *MD (Mixed Domains)* evaluation variant [11], where the evaluation sets were randomly sampled from all domains. Table 2 presents the values of F1-score for each label (columns 3-8), global F1-score (column 9), micro-AUC and macro-AUC (columns 10-11) for all evaluation types related to the texts. Further details of these results are presented in work [11].

## 3 Emotion Recognition in Sentimenti

This year, the first results of the Sentimenti<sup>2</sup> project were published. The main aim of the project was to design the methods of analysing textual data available on the Web in terms of the emotions expressed by the authors as well as the emotional impact of the texts on their readers. Within the project, a large database has been created in which 30,000 lexical units from plWordNet database and

---

<sup>1</sup> <https://clarin-pl.eu>

<sup>2</sup> <https://sentimenti.com/>

| Model | +m           | +s           | 0            | -s           | -m           | amb          | F1           | micro        | macro        |
|-------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|       |              |              |              |              |              |              |              |              |              |
| C1    | 83.20        | 40.27        | <b>97.14</b> | 10.91        | 85.28        | 17.72        | 76.83        | 88.92        | 68.04        |
| C2    | 81.21        | 41.03        | 96.75        | 09.68        | 83.36        | 21.57        | 74.35        | 92.90        | 74.79        |
| C3    | 81.82        | 00.00        | 96.39        | <b>10.96</b> | 80.75        | <b>27.64</b> | 72.70        | 87.67        | 74.19        |
| C4    | <b>86.12</b> | <b>50.00</b> | 94.65        | 00.00        | <b>86.87</b> | 22.86        | <b>77.78</b> | <b>95.78</b> | <b>78.85</b> |

**Table 2.** F1-scores for multi-domain text-oriented evaluation. Classifiers: C1 - fastText, C2 - BiLSTM, C3 - BiLSTM with word embeddings extended using polarity dictionary, C4 - BERT.

7,000 texts were evaluated, most of which are consumer reviews from the domain of hotels and medicine. The lexical units and texts were evaluated by 20,000 unique Polish respondents in the Computer Assisted Personal Interview survey and more than 50 marks were obtained for each element, which gives more than 1.8 million annotations. Within each mark, polarisation of the element, stimulation and basic emotions aroused by the recipients are determined.

In the experimental part, the authors decided to use a popular baseline model based on fastText algorithm [1] as a reference method for the evaluation. The authors used *one-versus-all cross-entropy* loss and 250 training epochs, with KGR10 pretrained word vectors [12]. They decided also to adopt the multi-labelled BiLSTM networks and expand the research to the more challenging task of emotion detection. As an input for BiLSTM networks, pre-trained fastText embeddings trained on KGR10 corpus [12] were used.

| Model | Valence <sub>p</sub> | Valence <sub>n</sub> | Arousal      | Joy          | Surprise     | Anticip.     | Trust        | Sadness      | Anger        | Fear         | Disgust      |
|-------|----------------------|----------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|       |                      |                      |              |              |              |              |              |              |              |              |              |
| C1    | 75.36                | 81.01                | <b>68.31</b> | 75.51        | <b>64.69</b> | 74.87        | 71.52        | 82.14        | 79.70        | 73.24        | 71.85        |
| C2    | <b>80.72</b>         | <b>85.22</b>         | 66.56        | <b>80.92</b> | 63.54        | <b>80.77</b> | <b>78.16</b> | <b>84.72</b> | <b>85.66</b> | <b>74.51</b> | <b>77.41</b> |

**Table 3.** F1-scores for Sentimenti dataset evaluation in the task of emotion recognition. Classifiers: C1 - fastText, C2 - BiLSTM.

Table 3 shows the results of evaluation for mixed-domain setting. BiLSTM classifier outperformed fastText in 9 out of 11 cases. These models were also tested in terms of their ability to generalize over different domains (the authors used *1-domain-out* training schema). A more detailed analysis of domain-adaption for the task of sentiment analysis was also presented in [12].

## References

- Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics

- tics **5**, 135–146 (2017)
2. Cruz, F., Troyano, J., Pontes, B., Ortega, F.J.: Building layered, multilingual sentiment lexicons at synset and lemma levels. *Expert Systems with Applications* **41**, 5984–5994 (10 2014)
  3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
  4. Esuli, A., Sebastiani, F.: SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining. In: Proceedings of 5th Conference on Language Resources and Evaluation LREC 2006. pp. 417–422 (2006)
  5. Fellbaum, C. (ed.): WordNet – An Electronic Lexical Database. The MIT Press (1998)
  6. Janz, A., Kocoń, J., Piasecki, M., Zaśko-Zielińska, M.: plWordNet as a Basis for Large Emotive Lexicons of Polish. In: LTC'17 8<sup>th</sup> Language and Technology Conference. Fundacja Uniwersytetu im. Adama Mickiewicza w Poznaniu, Poznań, Poland (Nov 2017)
  7. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. pp. 427–431. Association for Computational Linguistics (April 2017)
  8. Kocoń, J., Gawor, M.: Evaluating KGR10 Polish word embeddings in the recognition of temporal expressions using BiLSTM-CRF. *Schedae Informaticae* **27** (2018)
  9. Kocoń, J., Janz, A., Piasecki, M.: Classifier-based Polarity Propagation in a Wordnet. In: Proceedings of the 11<sup>th</sup> International Conference on Language Resources and Evaluation (LREC'18) (2018)
  10. Kocoń, J., Janz, A., Piasecki, M.: Context-sensitive Sentiment Propagation in WordNet. In: Proceedings of the 9<sup>th</sup> International Global Wordnet Conference (GWC'18) (2018)
  11. Kocoń, J., Zaśko-Zielińska, M., Milkowski, P.: Multi-Level Analysis and Recognition of the Text Sentiment on the Example of Consumer Opinions. In: Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2019 (2019)
  12. Kocoń, J., Janz, A., Piotr, M., Riegel, M., Wierzba, M., Marchewka, A., Czoska, A., Grimaling, D., Konat, B., Juszczak, K., Klessa, K., Piasecki, M.: Recognition of emotions, polarity and arousal in large-scale multi-domain text reviews. In: Vetulani, Z., Paroubek, P. (eds.) Human Language Technologies as a Challenge for Computer Science and Linguistics, pp. 274–280. Wydawnictwo Nauka i Innowacje, Poznań, Poland (2019)
  13. Liu, B.: Sentiment analysis: Mining opinions, sentiments, and emotions. Cambridge University Press (Jan 2015)
  14. Piasecki, M.: User-driven language technology infrastructure—the case of clarin-pl. In: Proceedings of the Ninth Language Technologies Conference. Ljubljana, Slovenia (2014)
  15. Stone, P.J., Dunphy, D.C., Smith, M.S., Ogilvie, D.M.: The General Inquirer: A Computer Approach to Content Analysis. MIT Press (1966)
  16. Strapparava, C., Valitutti, A.: WordNet-Affect: An affective extension of WordNet. In: Proceedings of the 4th International Conference on Language Resources and Evaluation. pp. 1083–1086 (2004)
  17. Zaśko-Zielińska, M., Piasecki, M., Szpakowicz, S.: A large wordnet-based sentiment lexicon for polish. In: Proceedings of the International Conference Recent Advances in Natural Language Processing. pp. 721–730 (2015)

# Bayes optimal prediction for NDCG@ $k$ in extreme multi-label classification

Kalina Jasinska<sup>1,2</sup> and Krzysztof Dembczyński<sup>1</sup>

<sup>1</sup> Institute of Computing Science, Poznań University of Technology, Poland

{kjasinska, kdembczynski}@cs.put.poznan.pl

<sup>2</sup> Allegro.pl, Poznań, Poland

**Abstract.** Extreme multi-label classification (XMLC) is a supervised learning problem in which instances are labeled with a number of relevant labels from a very large set of target labels. For many performance metrics used in XMLC, such as Hamming loss, macro F-measure, or precision@ $k$ , the Bayes optimal prediction is a function of label marginal probabilities. Interestingly, this is not the case for NDCG@ $k$ . In this paper we recall the form of the Bayes optimal prediction for this performance metric, and show how to apply this result to improve empirical results of decision tree-based classifiers.

**Keywords:** Extreme classification · multilabel classification · statistical decision theory · normalized discounted cumulative gain

## 1 Problem statement

Let  $\mathcal{X}$  denote a feature space and  $\mathcal{L} = \{1, \dots, m\}$  be a finite set of  $m$  class labels. Each subset of labels  $\mathcal{L}_+ \subset \mathcal{L}$  can be represented by a binary vector  $\mathbf{y}$ , with  $y_j = 1$  if and only if  $j \in \mathcal{L}_+$ . We use  $|\mathbf{y}|$  to denote the number of labels in the vector  $\mathbf{y}$ , i.e.,  $\sum_j y_j = |\mathcal{L}_+|$ . The label vector space is denoted by  $\mathcal{Y}$ . Observations  $(\mathbf{x}, \mathbf{y})$  are generated i.i.d. according to the distribution  $\mathbf{P}(\mathbf{X} = \mathbf{x}, \mathbf{Y} = \mathbf{y})$  (denoted later by  $\mathbf{P}(\mathbf{x}, \mathbf{y})$ ) defined on  $\mathcal{X} \times \mathcal{Y}$ .

The XMLC problem can be stated as finding a *classifier*  $\mathbf{h}(\mathbf{x}) = (h_1(\mathbf{x}), \dots, h_m(\mathbf{x}))$ , defined as a mapping  $\mathcal{X} \rightarrow \mathcal{R}^m$ , that minimizes the *expected loss* or *risk*:

$$L_\ell(\mathbf{h}) = \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \mathbf{P}(\mathbf{x}, \mathbf{y})} (\ell(\mathbf{y}, \mathbf{h}(\mathbf{x}))),$$

where  $\ell(\mathbf{y}, \hat{\mathbf{y}})$  is the (*task*) *loss*. The expected loss for a single  $\mathbf{x}$ ,  $\mathbb{E}\ell_{\text{log}}(\mathbf{h} | \mathbf{x})$ , is called a *conditional risk*. The optimal classifier, the so-called *Bayes classifier*, for a given loss function  $\ell$  is the one minimizing the risk,  $\mathbf{h}_\ell^* \in \arg \min_{\mathbf{h}} L_\ell(\mathbf{h})$ .

## 2 Bayes optimal predictions

To better clarify the result for NDCG@ $k$ , we contrast it with the result for precision@ $k$  which is defined as a fraction of positive labels among  $k$  predicted labels,

$$p@k(\mathbf{y}, \mathbf{x}, \mathbf{h}) = \frac{1}{k} \sum_{j \in \rho_k(\mathbf{h})} y_j,$$

where  $\rho_k(\mathbf{h})$  is a set of  $k$  labels predicted by  $\mathbf{h}$  for  $\mathbf{x}$ . Usually, the labels are predicted as top  $k$  labels from some ranking. However, the definition of precision@ $k$  does not require the labels to be sorted.

To define NDCG@ $k$ , we first discuss DCG@ $k$ . Let  $\sigma$  be a permutation of labels returned by classifier  $\mathbf{h}$ , such that  $\sigma(\mathbf{h}, r) \in \mathcal{L}$  is a label on the  $r$ -th rank. By adding to the definition of precision@ $k$  a discounting factor, usually  $\frac{1}{\log_2(r+1)}$ , and removing the normalizing factor  $\frac{1}{k}$ , we get the discounted cumulative gain:

$$\text{DCG}@k(\mathbf{y}, \mathbf{x}, \mathbf{h}) = \sum_{r=1}^k \frac{y_{\sigma(\mathbf{h}, r)}}{\log_2(r+1)}.$$

The best possible, or ideal, DCG@ $k$  for a given label vector  $\mathbf{y}$  is  $\text{IDCG}@k(\mathbf{y}) = \sum_{r=1}^{\min(k, |\mathbf{y}|)} \frac{1}{\log_2(r+1)}$ . By normalizing DCG@ $k$  by this factor we get the normalized discounted cumulative gain:

$$\text{NDCG}@k(\mathbf{y}, \mathbf{x}, \mathbf{h}) = N_k(\mathbf{y}) \text{DCG}@k(\mathbf{y}, \mathbf{x}, \mathbf{h}),$$

where, for simplicity of notation, we define  $N_k(\mathbf{y}) = \text{IDCG}^{-1}(\mathbf{y})$ .

Many multi-label learning algorithms are suited for estimating marginal probabilities

$$\mathbf{P}(y_j = 1 | \mathbf{x}) = \sum_{\mathbf{y}: y_j = 1} \mathbf{P}(\mathbf{y}, \mathbf{x}) = \sum_{\mathbf{y}} y_j \mathbf{P}(\mathbf{y}, \mathbf{x}).$$

Sorting the labels by these estimates and selecting top values is an easy way of obtaining final predictions. For many such algorithms precision@ $k$  and NDCG@ $k$  are frequently reported together. As we show in [1],<sup>3</sup> the Bayes optimal decisions for those metrics differ, so by optimizing one of them, one does not necessarily optimize the other one. Let us briefly recall these results and discuss their consequences. All considered measures are utilities, so in order to use them in the formal framework defined above, we will change them to loss functions, when necessary.

According to [3], the conditional risk for precision@ $k$  is

$$L_{p@k}(\mathbf{h} | \mathbf{x}) = -\frac{1}{k} \sum_{j \in \hat{\rho}_k} \mathbf{P}(y_j = 1 | \mathbf{x}).$$

This value is minimized when  $\sum_{j \in \hat{\rho}_k} \mathbf{P}(y_j = 1 | \mathbf{x})$  is maximized. The maximal sum is obtained by choosing  $k$  labels with the highest marginal probabilities  $\mathbf{P}(y_j = 1 | \mathbf{x})$ . Therefore, for precision@ $k$  selecting  $k$  labels with highest marginal probabilities is indeed the optimal strategy.

For NDCG@ $k$  the optimal strategy is different. Let  $\Delta_j(k, \mathbf{x})$  denote a specific marginalized value, similar to the marginal probability:

$$\Delta_j(k, \mathbf{x}) = \sum_{\mathbf{y}: y_j = 1} N_k(\mathbf{y}) \mathbf{P}(\mathbf{y} | \mathbf{x}).$$

---

<sup>3</sup> See <https://da2pl.cs.put.poznan.pl/programme/detailed-programme/da2pl2018-abstract-29.pdf>

Notice that  $\Delta_j(1, \mathbf{x}) = \mathbf{P}(y_j = 1 | \mathbf{x})$ . The conditional risk for NDCG@ $k$  is:

$$\begin{aligned} L_{NDCG@k}(\mathbf{h} | \mathbf{x}) &= - \sum_{r=1}^k \frac{1}{\log_2(r+1)} \sum_{\mathbf{y} \in \mathcal{Y}} y_{\sigma(\mathbf{h}, r)} \mathbf{P}(\mathbf{y} | \mathbf{x}) N_k(\mathbf{y}) \\ &= - \sum_{r=1}^k \frac{1}{\log_2(r+1)} \sum_{\mathbf{y}: y_{\sigma(\mathbf{h}, r)} = 1} \Delta_j(k, \mathbf{x}) \end{aligned}$$

The risk is minimized by top  $k$  labels sorted in the descending order of the values of  $\Delta_j(k, \mathbf{x})$ . Notice that the optimal decision for NDCG@ $k$  depends on  $k$ : not only the values of  $\Delta_j(k, \mathbf{x})$  may be different for different values of  $k$ , but also their relative order may change. Therefore an optimal decision for NDCG@ $k$  may be different than the one for NDCG@ $n$ ,  $n \neq k$ .

As we have seen, Bayes optimal classifiers for those two metrics are in general different. However, in specific situations those two solutions match. For example, this is the case for conditionally independent labels. It is also easy to see that when  $k = 1$ , NDCG@ $k$  and precision@ $k$  differ only by a constant, so in this case the Bayes optimal decisions match as well.

### 3 NDCGXML

To demonstrate how to apply our theoretical findings in practice, we modify the FastXML [2] algorithm to deliver predictions suited for NDCG@ $k$ . FastXML learns a forest of decision trees suited for XMLC problems. A single FastXML tree divides recursively the feature space with linear classifiers until each region, corresponding to a leaf, contains at most a certain number of training instances. A linear classifier in a given node is sought by optimizing a criterion that takes into account NDCG@m at the child nodes and the log loss of classifier's weights. During prediction a test example  $\mathbf{x}$  traverses the tree down to the corresponding leaf. The prediction is based on frequencies of labels among training instances in the region corresponding to the leaf. Formally, let  $\mathcal{X}_l$  denote a subset of training instances in the feature space region corresponding to leaf  $l$ . The prediction for any test instance  $\mathbf{x}$  ending up in  $l$ , is

$$\mathbf{h}_l^{\text{FastXML}}(\mathbf{x}) = \frac{1}{|\mathcal{X}_l|} \sum_{(\mathbf{x}_i, \mathbf{y}_i): \mathbf{x}_i \in \mathcal{X}_l} \mathbf{y}_i.$$

The frequencies can be treated as estimates of marginal probabilities of labels in this region of the feature space. We see that FastXML delivers solutions suited for precision@ $k$ .

Now we briefly discuss how to modify FastXML to deliver better predictions with respect to NDCG@ $k$  (we call the modified algorithm NDCGXML). To this end, we change the prediction made by leaves. Instead of the frequencies of labels we return:

$$\mathbf{h}_{l,k}^{\text{NDCGXML}}(\mathbf{x}) = \frac{1}{|\mathcal{X}_l|} \sum_{(\mathbf{x}_i, \mathbf{y}_i): \mathbf{x}_i \in \mathcal{X}_l} \mathbf{y}_i N_k(\mathbf{y}_i), \quad (1)$$

which corresponds to a vector of empirical values of  $\Delta_j(k, \mathbf{x})$ ,  $j \in \mathcal{L}$ , in region corresponding to leaf  $l$ .

**Table 1.** Results of NDCGXML (optimized for NDCG@5) and FastXML with a single tree.

| Classifier     | p@k     |         |         |         |         |         | NDCG@k  |         |         |         |         |         |
|----------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
|                | train   |         |         | test    |         |         | train   |         |         | test    |         |         |
|                | $k = 1$ | $k = 3$ | $k = 5$ | $k = 1$ | $k = 3$ | $k = 5$ | $k = 1$ | $k = 3$ | $k = 5$ | $k = 1$ | $k = 3$ | $k = 5$ |
| EurLex         |         |         |         |         |         |         |         |         |         |         |         |         |
| FastXML        | 92.17   | 73.45   | 60.51   | 47.86   | 39.30   | 31.58   | 92.17   | 78.09   | 71.11   | 47.86   | 41.50   | 37.11   |
| NDCGXML        | 92.17   | 73.43   | 60.51   | 47.91   | 39.41   | 31.67   | 92.17   | 78.11   | 71.33   | 47.91   | 41.59   | 37.21   |
| RCV            |         |         |         |         |         |         |         |         |         |         |         |         |
| FastXML        | 88.31   | 72.38   | 53.86   | 81.91   | 66.38   | 47.48   | 88.31   | 87.69   | 89.05   | 81.91   | 80.81   | 81.65   |
| NDCGXML        | 88.12   | 72.28   | 53.81   | 81.73   | 66.32   | 47.47   | 88.12   | 87.83   | 89.26   | 81.73   | 80.91   | 81.79   |
| Wiki10         |         |         |         |         |         |         |         |         |         |         |         |         |
| FastXML        | 94.18   | 78.31   | 66.56   | 70.77   | 56.12   | 45.08   | 94.18   | 81.99   | 72.91   | 70.77   | 59.59   | 51.03   |
| NDCGXML        | 94.18   | 78.31   | 66.56   | 70.74   | 56.10   | 45.08   | 94.18   | 82.00   | 72.92   | 70.74   | 59.56   | 51.02   |
| AmazonCat13    |         |         |         |         |         |         |         |         |         |         |         |         |
| FastXML        | 97.41   | 88.44   | 74.26   | 71.31   | 61.79   | 50.12   | 97.41   | 96.89   | 96.97   | 71.31   | 68.23   | 67.07   |
| NDCGXML        | 97.34   | 88.40   | 74.25   | 71.33   | 61.80   | 50.12   | 97.34   | 96.93   | 97.06   | 71.33   | 68.29   | 67.15   |
| DeliciousLarge |         |         |         |         |         |         |         |         |         |         |         |         |
| FastXML        | 70.95   | 59.05   | 52.98   | 35.67   | 31.29   | 28.74   | 70.95   | 62.33   | 58.11   | 35.67   | 32.33   | 30.38   |
| NDCGXML        | 70.55   | 58.83   | 52.85   | 35.26   | 30.99   | 28.50   | 70.55   | 62.62   | 58.86   | 35.26   | 32.01   | 30.11   |
| WikiLSHTC      |         |         |         |         |         |         |         |         |         |         |         |         |
| FastXML        | 72.81   | 49.06   | 37.36   | 30.34   | 18.54   | 12.64   | 72.81   | 68.91   | 70.48   | 30.34   | 26.45   | 25.06   |
| NDCGXML        | 72.46   | 48.84   | 37.30   | 30.46   | 18.52   | 12.62   | 72.46   | 70.29   | 72.63   | 30.46   | 26.57   | 25.19   |
| Amazon670      |         |         |         |         |         |         |         |         |         |         |         |         |
| FastXML        | 49.75   | 43.32   | 39.14   | 17.39   | 15.52   | 14.17   | 49.75   | 45.75   | 43.62   | 17.39   | 16.41   | 15.76   |
| NDCGXML        | 48.72   | 42.84   | 38.85   | 16.97   | 15.18   | 13.87   | 48.72   | 46.75   | 45.72   | 16.97   | 16.20   | 15.63   |

## 4 Empirical evaluation

We compare FastXML to NDCGXML with predictions computed using  $h_{l,5}^{\text{NDCGXML}}$ . Notice that  $h_{l,1}^{\text{NDCGXML}}(\mathbf{x})$  is equal to  $h^{\text{FastXML}}(\mathbf{x})$ , so there would not be a difference in performance between two algorithms. We perform the evaluation on benchmark datasets from the XMLC Repository.<sup>4</sup> For training we use default hyper-parameters of FastXML except the number of trees set to 1.

Table 1 contains results for  $p@k$  and NDCG@ $k$  for  $k = 1, 3, 5$  on train and test datasets. Certainly the values of  $p@1$  and NDCG@1 match, as discussed before. We see that on training data, FastXML outperforms NDCGXML in terms of  $p@k$  for all  $k$ , while NDCGXML outperforms FastXML on NDCG@5. The results on the test data may vary due to randomness. Notice that NDCGXML usually outperforms FastXML for NDCG@3, but the gains are smaller than for NDCG@5. This is so because the  $\Delta_j(5, \mathbf{x})$  is closer to  $\Delta_j(3, \mathbf{x})$  than  $\Delta_j(1, \mathbf{x})$ , or  $\mathbf{P}(y_j = 1 | \mathbf{x})$ , is.

## References

1. K. Jasinska and K. Dembczyński. Bayes optimal prediction for ndcg@ $k$  in extreme multi-label classification. In *DA2PL*, 2018.
2. Y. Prabhu and M. Varma. FastXML: A fast, accurate and stable tree-classifier for extreme multi-label learning. In *KDD*, pages 263–272. ACM, 2014.
3. M. Wydmuch, K. Jasinska, K. Dembczyński, M. Kuznetsov, and R. Busa-Fekete. A no-regret generalization of hierarchical softmax to extreme multi-label classification. In *NIPS*, 2018.

<sup>4</sup> See <http://manikvarma.org/downloads/XC/XMLRepository.html> for info on the datasets.

## Preliminary tests of a real-valued Anticipatory Classifier System\*

Norbert Kozłowski<sup>[0000-0003-4873-6730]</sup> and Olgierd Unold<sup>[0000-0003-4722-176X]</sup>

Department of Computer Engineering  
Wrocław University of Science and Technology  
Wyb. Wyspiańskiego 27, 50-370 Wrocław, Poland  
[norbert.kozlowski,olgierd.unold@pwr.edu.pl](mailto:norbert.kozlowski,olgierd.unold@pwr.edu.pl)

**Abstract.** The paper describes the first attempts toward designing and evaluating anticipatory classifier systems working in a real-valued input domain using interval predicates representation. Promising results are obtained by testing two environments - real-valued multiplexer and checkerboard from the classical XCSR problem domain.

**Keywords:** Anticipatory Learning Classifier Systems, OpenAI Gym

### 1 Introduction

Anticipatory Classifier System (ACS) [2] is a variant of Learning Classifier System (LCS) extending the classical human-interpretable, rule-based model with the psychological theory of anticipations. Every situation is accompanied by consequences after performing each possible behavior. ACS is tested in both single- and multi-step interaction processes like knowledge discovery or controlling mobile robot's arm. By design, all LCS-es are versatile towards the chosen alphabet and can work with complex adaptive systems. This work describes the first approach towards implementing interval predicates representation in ACS models. Traditionally, ACS use ternary alphabet to encode environmental state. Proposed modification enables to widen range of applicable problems by enabling adaptive interval ranges representation to model continuous-valued features.

### 2 Real-valued interval representation

In all further descriptions an assumption is made that single perception  $\sigma$  obtained from the environment is bounded to range  $\sigma \in [0, 1]$ . If it's not true an additional pre-processing step should be applied beforehand.

---

\* This abstract was created based on the detailed article that was presented at the conference The Genetic and Evolutionary Computation Conference, GECCO 2019

## 2.1 Existing alphabets

Most popular LCS representation for binary data uses *ternary representation*, where possible attributes values are encoded with three symbols -  $\{0, 1, \#\}$ . For continuous input value still there are many approaches possible, that will be briefly explained.

The most simple case is to binary encode both interval ranges, expand the rule condition and effect parts twice and reuse the ternary alphabet without any algorithm modification. This technique is most primitive, because of the vast increase in problem complexity.

Unold and Mianowski [4] extended the binary alphabet for more states (partitions) where each one was mapped to different data range. The algorithm was evaluated on two real-valued multi-step environments: the 1D linear corridor and the 2D continuous gridworld environments showing promising results. The drawback is, however, interval partitions are created upfront and cannot be changed in the process.

Dedicated alphabets were also created for XCS version of the algorithm (XCSR). In 1999 Wilson proposed a *center-spread representation* (CSR) [5], where interval was represented by two numbers - center of the range and the spread. Later in 2003 Stone and Bull analyzed two new representations *ordered-bounded* (OBR) and *unordered-bounded* (UBR) [3]. Both of them represent the range by using left  $x_1$  and right  $x_2$  bounds, but in OBR  $x_1 < x_2$ . By neglecting the ordering in UBR the crossover operator has greater chances of introducing better classifiers. In 2005 Dam and Abbass proposed also a *min-percentage* (MP) representation [1] trying to overcome some of the UBR drawbacks. In order to apply CSR, OBR, UBR and MP representations in XCS operators like covering, mutation and GA subsumption needed to be adjusted.

## 3 Real-valued ACS (rACS)

For the case of experiment, UBR was chosen for representing discretized interval predicates. The original ACS2 algorithm cannot work "out-of-the-box" with that representation, so several changes were required. Most important are listed below.

- **Don't care symbol.** In rACS the feature attributes consists solely of interval ranges. The "*don't care*" and "*pass-through*" symbol is represented as a full-ranged interval.
- **Covering** The covering process introduces randomness when a new classifiers is added into population. A new parameter - *covering noise*  $\epsilon_{cover}$  defines the maximum noise that can alter current perception. The noise  $\epsilon$  is drawn from uniform random distribution  $U[0, \epsilon_{cover}]$ . When creating a new classifier each condition and effect attribute is spread  $UBR(x_1 - \epsilon, x_2 + \epsilon)$  accordingly.
- **Mutation** Similarly, a new parameter - *mutation noise*  $\epsilon_{mutation}$  is used for introducing slight disturbances. For each attribute of condition and effect

perception string a noise  $\epsilon$  is drawn from uniform distribution  $U[-\epsilon_{mutation}, \epsilon_{mutation}]$  and added to the current value.

- **Subsumption** The mechanism was extended accordingly to analyze incorporating ranges.
- **Marking** Classifier's mark stores only single encoded exceptional perceptions (not intervals).

## 4 Experiments

Reproducible experiments<sup>1</sup><sup>2</sup> were performed on two single-step environments used for evaluating XCSR - real-multiplexer (rMPX) [5] and checkerboard [3]. In order observe anticipations a validation bit was appended to perception string and was activated when the agent proposed the correct action.

Additionally, all experiments were executed in *explore-exploit* mode alternating in each trial. In exploring phase the agent was set to choose action fully randomly, in exploiting one the action from best-fitted classifier was chosen. Detailed metrics used for creating plots were collected every 5 trials.

**Real Multiplexer** Parameters:  $\beta = 0.05$ ,  $\gamma = 0.95$ ,  $\theta_r = 0.9$ ,  $\theta_i = 0.2$ ,  $\epsilon = 1.0$ ,  $\theta_{GA} = 100$ ,  $m_u = 0.1$ ,  $\chi = 1.0$ ,  $\epsilon_{cover} = 0$ ,  $\epsilon_{mutation} = 0.25$ .

The ability of modeling environment was examined on 3-bit rMPX. Preliminary tests are promising to show that the agent is able to capture the feature interaction. When using encoding with 1 or 2 bits (i.e. Figure 1) the number of classifiers stabilizes after about 10000 trials. Using more accurate encoding is problematic. 3 and 4 bit encoding shows that exploit phase is performing better, but an uncontrolled classifier growth is observed.

**Checkerboard** Parameters:  $\beta = 0.05$ ,  $\gamma = 0.95$ ,  $\theta_r = 0.9$ ,  $\theta_i = 0.3$ ,  $\epsilon = 0.9$ ,  $\theta_{GA} = 100$ ,  $m_u = 0.2$ ,  $\chi = 0.6$ ,  $\epsilon_{cover} = 0.1$ ,  $\epsilon_{mutation} = 0.25$ .

All checkerboard experiments were performed using 2-dimensional checkerboard with 3 splits in each dimension. Input perception was encoded with 4 bits. The reward for providing the correct answer was  $\rho = 1$ .

Figure 2 demonstrates that answers given in explore phase are near  $\rho = 0.5$ . This behavior is expected because in this phase agent is performing mostly random guessing. For the exploit phase, the average reward is noticeably better over time, meaning that the agent is able to learn the environment rules. The rACS agent is unable to fully learn the environment due to the vast number of classifiers created.

---

<sup>1</sup> <https://github.com/ParrotPrediction/pyalcs>

<sup>2</sup> <https://github.com/ParrotPrediction/openai-envs>

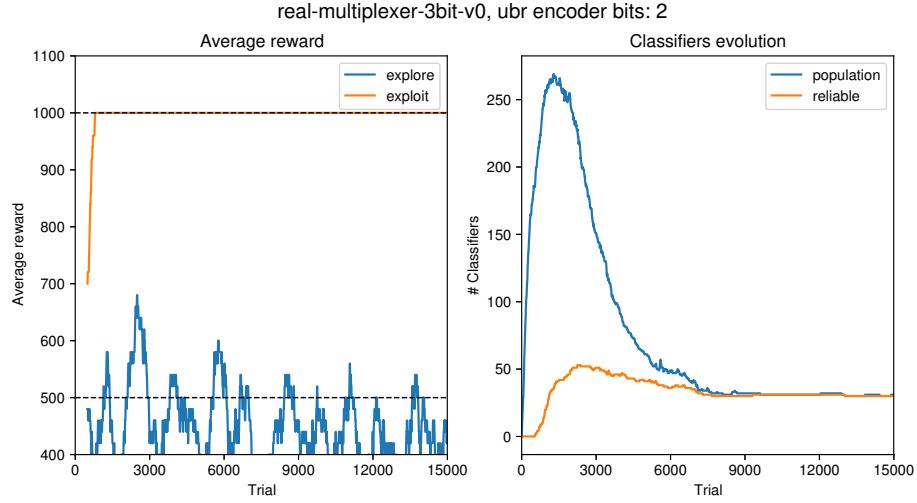


Fig. 1: 3-bit rMPX. 2-bit continuous value encoding. The agent is able to perfectly exploit the environment and the number of classifier converges. Moving average calculated from the last 50 metrics is presented

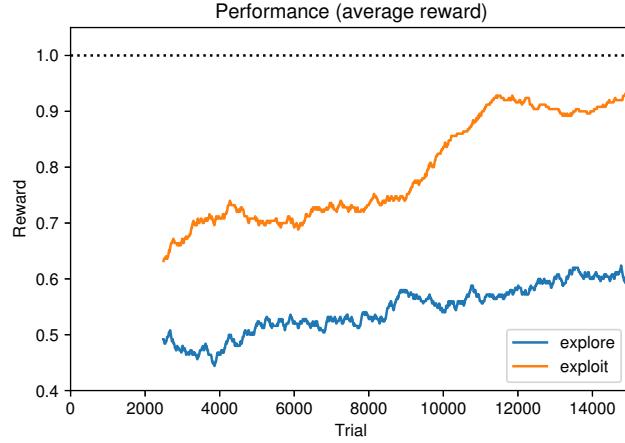


Fig. 2: Average reward obtained in 2-d checkerboard problem with 4-bit binary encoding. Reward is averaged using 250 past metrics.

## 5 Conclusions

The possibility of extending anticipatory classifier systems with the capability of representing intervals looks promising. Preliminary tests with two environments revealed that the algorithm is able to exploit gathered knowledge substantially better than by random guessing.

However, there are some issues preventing the rACS from satisfying the assumptions established by Stolzmann - the created classifiers should be accurate and maximally general. The most serious problem identified so far is related to the uncontrolled growth of the classifiers (see Checkerboard environment). The problem is caused by the fact that newly created classifiers are not compared to those already existing in the population  $[P]$ . The result is the creation of duplicated and overlapping off-springs covering the same niches. It's not the trivial issue because a new mechanism needs to be capable of subsuming and merging classifiers with similar ranges (condition and effect parts) meanwhile favoring more general classifiers.

Other aspects requiring further investigation is the impact of choosing proper number of bits for encoding, other alphabets for representing intervals (except from UBR) and adjustment of LCS operators for interval predicates.

## 6 Acknowledgments

The work was supported by the statutory grant of the Wrocław University of Science and Technology, Poland.

## References

1. Dam, H.H., Abbass, H.A., Lokan, C.: Be real! xcs with continuous-valued inputs. In: Proceedings of the 7th annual workshop on Genetic and evolutionary computation. pp. 85–87. ACM (2005)
2. Stolzmann, W.: Antizipative classifier systems. Ph.D. thesis, Fachbereich Mathematik/Informatik, University of Osnabrück (1997)
3. Stone, C., Bull, L.: For real! xcs with continuous-valued inputs. Evolutionary Computation 11(3), 299–336 (2003)
4. Unold, O., Mianowski, M.: Real-valued ACS classifier system: A preliminary study. In: Burduk, R., Jackowski, K., Kurzyński, M., Woźniak, M., Żołnierzek, A. (eds.) Proceedings of the 9th International Conference on Computer Recognition Systems CORES 2015. pp. 203–211. Springer International Publishing, Cham (2016)
5. Wilson, S.W.: Get real! xcs with continuous-valued inputs. In: International Workshop on Learning Classifier Systems. pp. 209–219. Springer (1999)

PS+O (problem solving and optimization)

# Influence of Traffic Type on Traffic Prediction Quality in Dynamic Optical Networks with Service Chains

Daniel Szostak [0000-0003-3268-5465] and Krzysztof Walkowiak [0000-0003-1686-3110]

Wrocław University of Science and Technology, Faculty of Electronics, Wrocław, Poland  
daniel.szostak@pwr.edu.pl

**Abstract.** Machine Learning (ML) algorithms can be employed for solving wide range of optical network optimization tasks. One of them is predicting future traffic in order to improve network performance (e.g., routing task). In this paper, we compare quality of various ML classifiers and examine impact of network traffic type on the quality of prediction. We consider an optical network, in which the basic service is related to service chains based on the Network Function Virtualization (NFV) paradigm. However, as a second type of traffic, normal background traffic also occurs in the network. We present and discuss numerical results of experiments run on two representative networks.

**Keywords:** Dynamic Optical Networks, Traffic Prediction, Machine Learning Classifiers Comparison.

## 1 Introduction

Knowledge about traffic incoming to network can be of utmost importance for operators, as it allows efficient resource management. To adjust network performance to current conditions, a concept of a *cognitive optical network* has been proposed. A cognitive optical network has a cognitive process, which analyzes network circumstances and adapt network operation to those conditions. The cognitive process typically uses ML techniques [1]. More about application of ML techniques in optical networks can be found in [2].

In this paper, we consider a dynamic optical network, in which traffic is represented by demands incoming to network within a time scale. Moreover, we assume that all network functions are virtualized according to Network Function Virtualization (NFV) paradigm. Virtualized functions can be run, on demand, on servers located in nodes, instead of running on specific hardware [3]. It makes network more flexible and can decreases costs of network building. At times, before arriving to end user, data flow has to go through a number of network functions. Therefore, permanent and repeatable network function chains can be observed in network traffic. Set of ordered VNF is called Service Function Chain (SFC) [4]. In consequence, new type of traffic can be distinguished, namely, *chain traffic*.

This paper is a continuation of our previous work [5], where we introduced a ML methodology for traffic prediction in dynamic optical network serving VNF chain

traffic. As a novelty, we examine impact of traffic type in network on classifiers prediction quality.

## 2 Problem definition

In our work, we assume that network is modeled as a directed graph composed of  $n$  nodes and a set of links connecting them (representing the set of physical optical links between nodes). In each node none, one or more VNFs can be located. Single VNF can belong to one or more chains. Traffic in network is represented by demands, which arrive to the network within a time scale divided into *time intervals* (TIs), also called *iterations*. Each demand can be described using three features: TI in which it appears, source node, and destination node. We consider two possible types of traffic – *chain* and *non-chain*. The *chain* traffic means that demands creating a given SFC occur in adjacent TIs. In particular, for chain from node  $v_1$  to node  $v_2$  related to a single VNF located in node  $v_3$ , two demands will be established. The first demand occurs in TI  $i$  and is established from node  $v_1$  to  $v_3$  and the second demand occurs in the next TI ( $i+1$ ) and is established from node  $v_3$  to node  $v_2$ . In this way, the whole chain is created. In the *non-chain* traffic, demands occur randomly with probability inversely proportional to the distance between end nodes of the demand.

Our objective of this work is to examine influence of traffic type in network on quality of predicting traffic in next TIs. We consider prediction task as a classification problem. For the sake of that, the number of possible classes can be equal to the number of possible demands. For a network with a higher number of nodes and possible demands, it makes multiclass prediction quite complex. To overcome this, we transform the problem to a simple binary problem, by technique of classifiers learning and then, by the way of prediction.

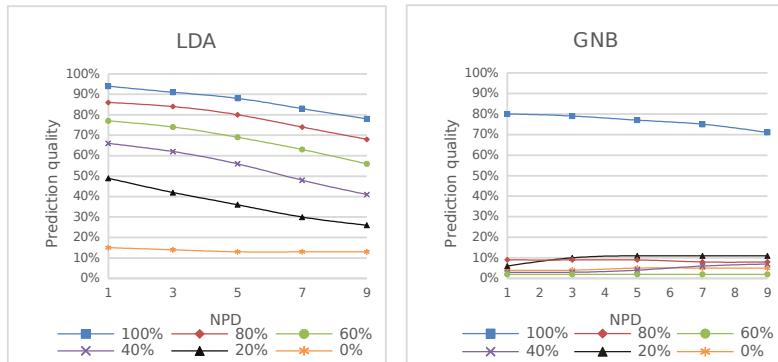
Instances in the learning dataset are created based on sets of demands arriving to network in consecutive TIs. Based on a single TI,  $n*(n-1)$  instances can be created (this responds to the number of possible demands in the network). Each instance is described by  $n*(n-1)+1$  features from which first  $n*(n-1)$  depict binary state of previous TI (if single demand occurs in previous TI, the corresponding position in vector is equal 1, 0 otherwise) and last is linked to the index of the demand. For each demand, possible binary classes are: positive, if the particular demand appears in the considering TI, and negative, if the demand does not appear in the considering TI. More detailed description of learning dataset creation can be found in [5].

In our machine learning approach, the number of used classifiers is equal to the number of possible demands. Each single classifier creates a model based on data related to one demand. The classification task is made for each demand by classifier related to the considered demand. As input classifier takes the binary state of previous TI and demand index. As output it returns support for each of binary class. We examined two classifiers – Linear Discriminant Analysis (LDA) and Gaussian Naive Bayes (GNB). In our previous work [5], those two classifiers obtained the best classification quality for all considered datasets with only *chain* traffic.

Evaluation quality of prediction is done by considering a ratio of correctly predicted demands assuming a fixed number of predicted demands (NPD). To illustrate this, let's assume that we want to predict 4 demands in next TI (NPD=4) and 3 of 4 predicted demands will appear in next TI, then the ratio is 0.75. All datasets generated for experiment based on *pol12* and *euro16* topologies. For each topology 6 datasets were created. Specifically, datasets differ from each other with ratio of *chain* to *non-chain* traffic. First dataset is composed of 0% *chain* and 100% *non-chain* traffic and in every next dataset amount of *chain* traffic increase by 20% and amount of *non-chain* traffic decrease by 20%. Average number of demands in each TI is equal to 10, chains consist of 3 demands.

### 3 Numerical results

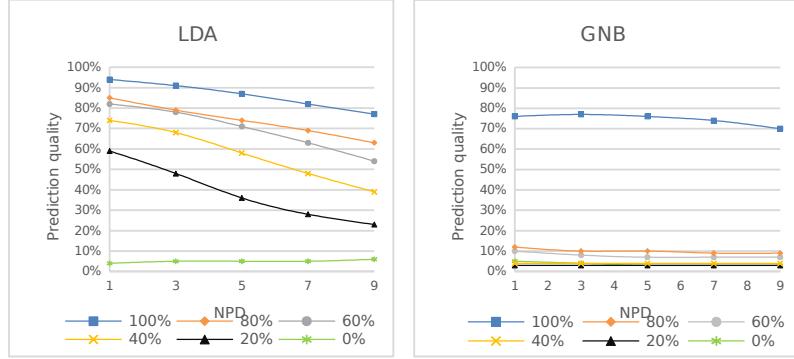
Figs 1 and 2 present numerical results obtained for *pol12* and *euro16* networks, respectively. Vertical axis shows classifiers prediction quality and horizontal axis shows NPD values. Results for particular datasets are represented by different lines, i.e., 40% means that dataset is composed of 40% of *chain* traffic and 60% of *non-chain* traffic.



**Fig. 1.** Prediction quality, datasets based on *pol12* topology

Fig. 1 depicts classifiers prediction quality for datasets based on *pol12* topology. For LDA a general trend can be noticed – with increasing of percentage of *non-chain* traffic in dataset, classifier prediction quality decreases. GNB obtains good results for 100% *chain* traffic dataset, however in case of mixed traffic significant drop of prediction quality can be seen. In case on both classifiers, with increasing of NPD value, prediction quality decreases.

In turn, Fig. 1, summarizes results for datasets based on *euro16* topology. Classifiers general trends are the same as in case of datasets based on *pol12* topology. To compare performance for both networks, we can see that LDA obtains better results, for smaller NPD values, in case of *euro16* datasets, when for larger NPD values it provides better results for *pol12* datasets. Furthermore, difference between prediction quality for the smallest and the largest NPD value is smaller in case of *pol12* datasets.



**Fig. 1.** Prediction quality, datasets based on *euro16* topology

## 4 Conclusions

In this paper, we examined influence of traffic type on quality of traffic prediction in dynamic optical networks with service chains. Our study showed that in the employed classification, quality is correlated with the percentage of *chain* traffic in the whole traffic in network. Specifically, with decreasing of *chain* traffic, classification quality also decreases. LDA obtained promising results for lower NPD values, thus applied approach can be used to predict future traffic in optical network.

## Acknowledgements

This work was supported by National Science Centre, Poland under Grant 2017/27/B/ST7/00888.

## References

1. V. W. S. Chan and E. Jang: Cognitive all-optical fiber network architecture, *ICTON*, Girona, 2017
2. F. Musumeci and et all: An Overview on Application of Machine Learning Techniques in Optical Networks, *IEEE Communications Surveys & Tutorials*, vol. 21, pp. 1383 – 1408, 2019.
3. J. Simmons, Optical Network Design and Planning, 2nd edit., *Springer*, 2014
4. V. Nikam and et all: VNF Service Chaining in Optical Data Center Networks, *NFV-SDN*, November 2017.
5. D. Szostak, K. Walkowiak: Machine Learning Methods for Traffic Prediction in Dynamic Optical Networks with Service Chains, *ICTON*, Angers, 2019.

# Social Impact Assessment and Multicriteria Optimization of AI Tools for Online Knowledge Provision

Andrzej M.J. Skulimowski [0000-0003-0646-2858]

<sup>1</sup>AGH University of Science and Technology, Chair of Automatic Control and Robotics,  
Decision Science Laboratory, 30-059 Kraków, Poland

<sup>2</sup>International Centre for Decision Sciences and Forecasting,  
Progress & Business Foundation, 30-048 Kraków, Poland  
e-mail: ams(at)agh.edu.pl

**Abstract.** This paper proposes a social impact model for AI-supported web-based knowledge repositories. The model applies multi-level simulation and a self-contained multicriteria analysis framework. It combines several simulation and forecasting methods, such as controlled discrete-event systems, learning cellular automata, user group dynamics, and anticipatory networks. These all serve first to monitor, measure, and assess the social impact of knowledge repository uses, then to optimise user community building strategies and cyber- and AI threats mitigation. The management of the above activities is organised as a decision support system capable of solving complex combinatorial planning problems. The primary application of this model has been an innovative digital knowledge repository and learning platform developed within a Horizon 2020 research project. *Ex-ante* impact assessment was intended to improve the repository services planning as well as its technological development and exploitation strategy. Operational recommendations to the knowledge repository management staff were driven by observed and forecasted user behaviour. Finally, the analysis of user responses to various service innovations and community building activities points out AI-related digital policies to implement and may be a contribution to AI development strategies.

**Keywords:** AI impacts, Knowledge repositories, Multicriteria analysis, User community modelling, Social impact assessment, Anticipatory networks

## 1 Introduction

Social impact assessment of Artificial Intelligence (AI) tools is increasingly relevant for the design of complex cloud-based and AI-enabled knowledge provision systems that strongly interact with their human users. This paper is concerned with the social impact of knowledge repositories that evolve towards Global Expert Systems (GES). The latter are a subclass of AI-based knowledge provision tools defined in [9] as “*all knowledge and information sources, such as sensors, databases, repositories, experts and processing units, regardless of whether they are human, artificial or hybrid, provided that all they are mutually connected and endowed with a holistic information management system*”. The development of GES is driven by salient ICT and AI trends, specifically by the web information systems integration and interoperability, the growing complexity and coverage of search engines, the development of Internet of Things

(IoT), Big Data tools, and intelligent human-computer interfaces, including the natural language communication and BCI. GES and related AI tools have been further investigated in [11], [6], and [14]. An example of a successful GES-targeted implementation is the knowledge platform developed within a recent Horizon 2020 project [6], [17].

One of the ultimate goals of this repository is to provide an efficient training and research support tool for young researchers, students, and public administrators. Its user community building approach presumes a wide use of existing cooperation networks and exploring the opportunities provided by different social media. According to the digital strategies in force in most OECD countries, the achievement of social impact goals of a knowledge repository financed from public funds grows with:

- The number of users and the composition of the community of users that should include members of the target groups determined by the funding institution,
- The user capability and engagement to effectively using the AI tools provided on the platform, such as intelligent recommenders, autonomous webcrawlers, chatbots etc., averaged over the community of users,
- User subjective satisfaction from the knowledge provision that may manifest in a readiness to promote the tool to further potential users,
- The prevention and mitigation of cyber- and AI-threats that maybe faced by the users of knowledge repositories or caused by them to such systems.

Reaching the above objectives for a repository requires an efficiency of disseminating the information about the content and functionalities offered to potential users. It involves also gathering user feedback, tracing their activity as system users with DOM events and other tools, and investigating the mechanisms of spontaneous social information diffusion in user communities which may lead to their growth according to the snowball principle. The latter can be formally described by learning cellular automata (CA) [2]. An active repository marketing, including market research, publication of popular and professional articles, conference presentations, repository positioning, appropriate uses of social media and other Internet tools, makes it a playground of social impact modelling with additional tools such as controlled discrete event systems (DES), and anticipatory global influence models [10].

This paper uses a bottom-up approach to present the social impact assessment problem and its solution. It studies the real-life case of a knowledge repository, but strives to generalise the methods used, proposes procedures and formulates rules that are potentially applicable to manage and optimise social impact of a large class of AI-enabled knowledge provision systems. The here presented social impact modelling approach provides a new look for the assessment of open knowledge repositories. This is due to the fact that the difficulties with the estimation of financial characteristics of the repository uses made the common social return on investment (SROI) method [13] infeasible. On the other hand, the innovation diffusion models, e.g. the well-known Norton-Bass model [5] do not grasp the immediate and cloud-based character of information diffusion and social recommendation propagation in the Internet. This led us to formulate a new methodology of hybrid modelling, simulation, optimisation and assessment of intelligent knowledge repositories social impact and to implement it as a group decision support system (DSS). It uses controlled polymorphic [7] or learning cellular automata [2],[3],[16],[17], and Bayesian network refers to earlier research projects dealing with advanced ICT social impact [4]. The fusion and

simultaneous use of modelling methods brings a systemic added value over the sum of individual experiences with each of the single approaches.

To optimise the user community-building, we created an action plan theory that includes a formulation of a multicriteria repository development planning problem. The user community evolution is investigated with the controlled discrete-event systems (DES) to model single users and learning cellular automata (CA) to model their cooperation in the virtual communities of practice [18], the dissemination of information [8] and the subsequent community growth. We assumed that the community building activities act on a random scale-free network [1], whereas the user community forms a subnetwork with a variable structure. The decision support system (DSS) implementing the above model is capable of providing exploitation –related recommendations to repository owners and stakeholders. The DSS uses anticipatory networks [9] to model the consequences of resource allocation decisions and find a compromise action plan. Beyond optimizing social goals, the above social impact modelling framework can also be a source of general rules of AI impact modelling and threat monitoring, which can be presented as digital policy recommendations.

## 2 Social impact modelling principles

An implementation-friendly exploitation strategy of an information system that sets goals in a realistic manner, to be achieved in an optimal way is an important, yet often undervalued, component of any software project financed by public funds. We assume that strategic goal attainment can be evaluated by the set of quantitative criteria  $\tilde{F} := \{\tilde{F}_1, \dots, \tilde{F}_N\}$  in a multicriteria optimization problem  $F = (F_1, \dots, F_N) \rightarrow \min$ . The criteria should fulfill the following conditions:

- (i) better values of each criterion correspond to a higher satisfaction of user or stakeholder preferences related to at least one of the project goals (*representativeness*),
- (ii) every change in user or stakeholder satisfaction as regards goal attainment can be expressed equivalently as a change in values of at least one of the criteria functions (*completeness*), and
- (iii) condition (ii) fails to be satisfied after removing any of the coordinate functions of  $\tilde{F}$  (*strong non-redundancy*).  
If the criteria values are skewed by statistical errors, (iii) may be replaced by:
- (iv) if multiple stochastic criteria describe the attainment of the same goal (or goals) then they are independent random variables (*weak non-redundancy*).

In case of the repository [6] the social goals have been formulated as  $G_j, j=1,2,3$ :

$G_1$  - reaching a given number of satisfied users,

$G_2$  - reaching the prescribed number and quality of services offered,

$G_3$  - reaching the content quality and quantity in predefined fields that is assessed as satisfactory by the users.

These goals are then quantified as the social impact criteria  $F := \{F_1, \dots, F_n\} \subset \tilde{F}$ , which relate to a population of users or potential users  $\Psi$ , at certain time  $t$ , cf. Table 1.

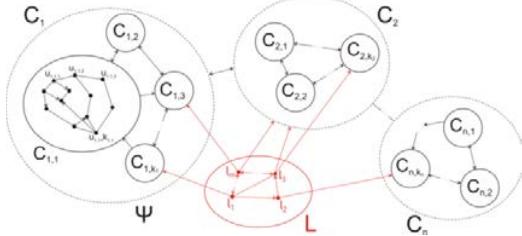
**Table 1.** The relations between the social goals and the quantitative social impact criteria

| $F_i$ | Criteria description   | Source of data, verification means  |
|-------|--|---|
| $F_1$ | Number of target group members in $\Psi$ informed about the repository   | User data continually reported with the dedicated feedback forms  |
| $F_2$ | Number of repository's active users in the population $\Psi$   | Estimations based on the efficiency of dissemination activities. Verification with user logs                                |
| $F_3$ | Number of end users in $\Psi$ assessing the pservices as at least satisfactory                                   | The share of satisfied users estimated <i>ex-ante</i> with a Delphi survey [12], user feedback forms                        |
| $F_4$ | Number of end users in $\Psi$ assessing the repository services as excellent and recommending them to the others | User feedback forms the share of new users encouraged by the others based on voluntary information provided at registration |
| $F_5$ | Number of stakeholders that assess the repository operation as satisfactory                                      | Delphi survey to elicit <i>ex-ante</i> estimations of relevant variables, user feedback forms [12]                          |
| $F_6$ | Aggregated stakeholder substantial assessment of the available repository content, services and functionalities  | Feedback forms and structured interviews with the stakeholders, to be confronted with the development expenses $F_9$        |

The multicriteria optimization problem that is equivalent to the attainment of the above defined social goals  $G_1, G_2, G_3$  and criteria  $F$  with resource constraints has been solved with anticipatory networks [10], [13], where the causal component of the network modelled the anticipated consequences of splitting the available resources into a) communication and promotion activities, b) implementing new functionalities and acquiring new content, and c) providing an active cyber threat detection and mitigation services. The resulting optimal strategy presumes the following actions:

- Building polymorphic cellular learning automata and controlled discrete event system models to elicit the user and user group evolution rules.
- Formulating and solving multicriteria optimization problems to maximise the social impact criteria and the impact/expenditure ratio (cf. [13] for a discussion of SROI) using statistical aggregation and anticipatory networks [9].
- Deriving strategic recommendations to the repository management, operational hints to the users and staff, as well as policy recommendations to stakeholders.

Fig. 1 below presents a typical structure of a population  $\Psi$  where community building actions are planned. Individual users  $u_i$  form a social network composed of their communities of practice  $C_j$ . The communities may form larger structures  $C_k$ , for example, learning groups at a university are grouped into faculties. The expected connections between users are initialised as a random network, e.g. of Barabasi-Albert [1].

**Fig. 1.** A scheme of the population  $\Psi$  with two levels of communities of practice and other user groups ( $C_{i,j}$  and  $C_i$ ) and the community of innovation leaders ( $L$ )

A special role is played by the community  $L$  whose members  $l_k$  are open innovation leaders and experts capable to influence entire communities. The user state transition model is a controlled DES. The logic of modelling the community of users and its impact on them at individual and learning group level may be presented as follows.

For each group of users, a transition pattern is built, where a user in a certain state  $q$  can be moved to another state. State transitions result from an intervention modelled with external controls  $v$  acting on users, from the influence of other users, or they may occur spontaneously, either randomly, or as a result of a natural process depending on  $q$  or the state of the environment. In the first case, the target state is preferred by the repository management. Other transitions may lead to less favoured states.

### 3 Discussion and conclusions

Ex-ante and forecasted social impact of ICTs has been a subject of intensive research since several decades, recently focusing on AI impacts. A diversified spectrum of ICT/AI social impact aspects, focusing on impact of web-based applications involving a large number of non-professional users, referred to as '*ICT-enabled social innovations*' (IESI, [7]), has been a subject of study of several recent EU-financed projects. The interest in IESI is due to the fact that they are capable of reaching social policy goals related to education, employment and proactive attitudes in a timely, relatively low-cost manner, and according to common citizens' preferences. Despite of much efforts spent on elaborating a methodology of social impact modelling for digital social innovations, the results have not reached out much beyond SROI [15] and have a restricted applicability area. Therefore the model based on user community growth, composition and quality presented in this paper constitutes a relevant and applicable contribution to multicriteria social impact assessment of AI-based tools.

The approach here presented is dedicated in the first order of importance to digital learning platforms that can be regarded as a relevant class of web-based social innovations [4]. They can be characterised as open-access knowledge repositories, endowed with intelligent and cognitive functionalities and automatic content updating mechanisms featuring autonomous web crawlers that can search for statistical, patent, or bibliographic information. Beyond impact simulation, the procedures proposed in this paper can generate practical recommendations primarily concerning the user community building strategy implementation of similar digital knowledge repositories, in a clear formal way. The rigorous impact planning will also facilitate aligning the repository's services to the future trend of implementing more cognitive features and functionalities and to an increasingly autonomous and complex user interactions [11].

Moreover, the impact modelling approach presented in this paper can be regarded a general framework of social impact assessment for a broader class of AI-based information systems. It can be applied to other open-source research and learning content repositories, as well as to digital libraries [4],[13]. The selection of modelling methods may vary from case to case, to include the approaches suitable for specific systems or goals, taking into account a variety of AI-based social services delivery and the diversity of their users.

## References

1. Albert, R.; Barabási, A.-L.: Statistical mechanics of complex networks. *Rev. Mod. Phys.* 74, 47–97 (2002)
2. Beigz, H.; Meybodi, M.R.: Asynchronous cellular learning automata. *Automatica* 44, 1350–1357 (2008)
3. Dabbaghian, V.; Spicerb, V.; Singh, S.K.; Borwein, P.; Brantingham, P.: The social impact in a high-risk community: A cellular automata model. *J. Comput. Science* 2, 238-246 (2011)
4. IESI project (“ICT-Enabled Social Innovation to support the implementation of the Social Investment Package”) web site: <https://ec.europa.eu/jrc/en/iesi> [accessed: July 31, 2019]
5. Jiang, Z.; Jain, D.C.: A generalized Norton-Bass model for multigeneration diffusion. *Management Sci.* 58(10), 1887–1897 (2012)
6. MOVING project (“Training towards a society of data-savvy information professionals to enable open leadership innovation”) web site: [www.moving-project.eu](http://www.moving-project.eu) [acc.: Aug. 31, 2019]
7. Sekanina, L.; Komenda, T.: Global Control in Polymorphic Cellular Automata. *Journal of Cellular Automata* 6(4-5), 301-321 (2011)
8. Skulimowski A.M.J.: Optimizing the structure of a partitioned population. In: Henry J., Yvon J.P. (eds), *System Modelling and Optimization*. LNCIS, 197. pp. 771-782, Springer, Berlin, Heidelberg (1994), <https://doi.org/10.1007/BFb0035527>
9. Skulimowski, A.M.J.: Universal intelligence, creativity, and trust in emerging global expert systems. In: L. Rutkowski et al. (eds.), 12<sup>th</sup> International Conference on Artificial Intelligence and Soft Computing (ICAISC 2013), Zakopane, Proceedings, Part II, LNAI 7895, Springer, Berlin–Heidelberg, pp. 582-592 (2013), [https://doi.org/10.1007/978-3-642-38610-7\\_53](https://doi.org/10.1007/978-3-642-38610-7_53)
10. Skulimowski, A.M.J.: Anticipatory Network Models of Multicriteria Decision-Making Processes. *Int. J. Systems Sci.* 45(1), 39-59 (2014), DOI: 10.1080/00207721.2012.670308
11. Skulimowski, A.M.J.: Impact of Future Intelligent Information Technologies on the Methodology of Scientific Research. In: 16th IEEE Int. Conf. on Computer and Information Technology, Nadi, Fiji, Dec. 7-10, 2016, IEEE, pp.238-247 (2016), DOI: [10.1109/CIT.2016.118](https://doi.org/10.1109/CIT.2016.118)
12. Skulimowski, A.M.J.: Expert Delphi Survey as a Cloud-Based Decision Support Service, IEEE 10th International conference on Service-Oriented Computing and Applications (SOCA 2017), 22–25 Nov. 2017, Kanazawa, Japan. IEEE, Piscataway, pp. 190–197 (2017), <https://ieeexplore.ieee.org/document/8241542>
13. Skulimowski, A.M.J. Anticipatory Networks. In: Poli R. (eds) *Handbook of Anticipation*. Springer, Cham, pp. 995-1030 (2019), [https://doi.org/10.1007/978-3-319-91554-8\\_22](https://doi.org/10.1007/978-3-319-91554-8_22)
14. Skulimowski, A.M.J.; Badecka, I.; Klamka, J.; Kluz, D.; Ligeza, A.; Okoń-Horodyńska, E.; Pukocz, P.; Rotter, P.; Tadeusiewicz, R.; Wiśla, R.: Trends and Scenarios of Selected Information Society Technologies. *Advances in Decision Sciences and Futures Studies*, vol.1. Progress & Business Publishers, Kraków (2017)
15. Then, V.; Schober, Ch.; Rauscher, O.; Kehl, K.: *Social Return on Investment Analysis. Measuring the Impact of Social Investment*, Palgrave Studies in Impact Finance, Palgrave Macmillan, p.406 (2017), <https://doi.org/10.1007/978-3-319-71401-1>
16. Vafashoar, R.; Meybodi, M.R.: Multi swarm optimization algorithm with adaptive connectivity degree. *Appl. Intell.* 48, 909–941 (2018)
17. Vagliano, I.; Günther, F.; Heinz, M.; Apaolaza, A.; Bienia, I.; Breitfuss, G.; Blume, T.; Collyda, C.; Fessl, A.; Gottfried, S.; Hasitschka, P.; Kellermann, J.; Köhler, T.; Maas, A.; Mezaris, V.; Saleh, A.; Skulimowski, A.M.J.; Thalmann, S.; Vigo, M.; Wertner, A.; Wiese, M.; and Scherp, A. Open Innovation in the Big Data Era With the MOVING Platform. *IEEE MultiMedia* 25(3), 8-21 (2018), DOI: 10.1109/MMUL.2018.2873495
18. Wang, J.; Zhang, R.; Hao ,J-X.; Chen, X.: Motivation factors of knowledge collaboration in virtual communities of practice: a perspective from system dynamics, *Journal of Knowledge Management* 23(3), 466-488 (2019), <https://doi.org/10.1108/JKM-02-2018-0061>

# Dynamic signature verification using AI methods

Marcin Zalasiński<sup>1</sup>[0000–0002–0009–6124]  
and Krzysztof Cpałka<sup>1</sup>[0000–0001–9761–118X]

Częstochowa University of Technology,  
Institute of Computational Intelligence, Poland  
[{marcin.zalasinski,krzysztof.cpalka}@iisi.pcz.pl](mailto:{marcin.zalasinski,krzysztof.cpalka}@iisi.pcz.pl)

**Abstract.** Dynamic signature is a biometric attribute which can be used for identity verification. It is very useful because it is commonly accepted in the society, however, verification with the use of this biometric characteristic is a difficult process. Artificial intelligence methods can be used to improve the effectiveness of identity verification. Examples of them are described in this paper.

**Keywords:** Dynamic signature · Artificial intelligence · Biometrics.

## 1 Introduction

Signature is a commonly used and socially acceptable form of authorization. It is also a characteristic used in behavioral biometrics for identity verification. Dynamic signature is a special kind of this characteristic, which contains also information about the dynamics of signing process (e.g. pen pressure, instant pen velocity, pen tilt angle, etc.) in the form of signals changing over time [4]. These signals describing signature are acquired using a digital input device, e.g. graphic tablet. The dynamic signature analysis allows us to obtain many pieces of information characteristic for individual signer what makes verification process more effective than in case of using a so-called static signature which contains only information about the signature's shape.

Dynamic signature is a very interesting biometric attribute which can be very useful in practice, however, identity verification using this biometric characteristic is a difficult task. Researchers involving this process can meet various problems, for example, lack of biometric samples derived from forgers at the training phase of the biometric system, the different discriminative power of descriptors describing the signature, changes in the behavior of the biometric system users over time, etc. However, artificial intelligence methods can be very useful to resolve these problems. Some of them are presented in this paper.

## 2 AI methods for the dynamic signature verification

In this section, we describe some artificial intelligence methods which can be used to improve effectiveness of the identity verification process.

## 2.1 Classification using neuro-fuzzy one-class classifier

A certain challenge for IT systems that verify identity based on behavioral biometric attributes is the selection and training of the appropriate classifier. This is due to the fact that in practice at the stage of system learning there are no samples of false biometric features that in real conditions are generated by forgers. The forgers are often "skilled" [3] and have knowledge about the behavior of the biometric system users. This problem is often solved in such a way that biometric features of other users of the system are treated as random false samples [3] at the stage of system learning. However, it is not consistent with the real conditions prevailing during identity verification, because a system learned by random false samples may not be protected against the attack of a skilled forger. Therefore, in our previous papers, we focused on the creation of an effective classifier solving this problem [1, 6, 8]. As a result, we have developed a new neuro-fuzzy one-class classifier of behavioral biometric attributes. A characteristic feature of our classifier is that the verification of test biometric features is based on the responses of the flexible neuro-fuzzy system. This type of system is characterized by an action based on the set of rules of the form "if-then" and the possibility of selecting parameters in the learning process. In addition, the flexible system allows us to take into account the hierarchy of importance of the premises of rules and entire rules [2]. Its parameters in the considered problem must be selected individually for each user included in the system's database. Our classifier meets all the requirements for identity verification algorithms based on biometric features, i.e.: 1) it works regardless of the number of users (its accuracy does not depend on the number of users in the database), 2) it has the ability to easily expand with the characteristics of new users, 3) it does not take into account the characteristics of other users during identity verification of the considered user. In many applications, machine learning (e.g. gradient or evolutionary) is used to select parameters of the fuzzy system rules (see e.g. [5]), but it is not suitable for use in combination with identity verification algorithms. This is due to the fact that the effectiveness of the system trained using machine learning depends largely on the number and quality of training samples. In the case of biometric systems, during the learning phase, only a small number of reference samples of a given biometric feature is available and samples of forgers are not available. Machine learning is also an iterative process that should be carried out independently for each database user. Due to this, we proposed a new structure of a flexible fuzzy one-class classifier whose parameters depend on the reference descriptors of biometric features. The parameters are determined analytically (not in the supervised learning process) and individually for each user (his reference features). Moreover, the classifier works with very good accuracy.

## 2.2 Evolutionary selection of the most characteristic descriptors

Our research showed that not all descriptors describing the dynamic signature are equally important in the context of identity verification for different users

of the biometric system. Therefore, we have developed methods which automatically select the most characteristic descriptors in the context of each signer. These methods combine the following features: 1) they use a population algorithm to select biometric features descriptors, eliminating features that can affect the accuracy of identity verification, 2) they operate individually for each signer, 3) during the learning phase, they do not require patterns in the form of false attributes or features of other users, because they use only biometric attributes of currently considered signer, 4) they are not dependent on the number of elements in the set of features that can be arbitrarily reduced or enlarged. For example, in [9] we have presented the method of individual selection of biometric features descriptors which uses the classic genetic algorithm with a specially designed evaluation function. In this method, each of the chromosomes contains binary encoded information about the descriptor set that will be evaluated. To evaluate the usefulness of selected descriptors of signatures in the context of the effectiveness of identity verification, a measure of their similarity to the template is determined, which is expressed in the form of the Mahalanobis distance. In this work, the proposed method has been tested using global descriptors of dynamic signature, but it is a universal method. The results obtained during the simulation confirm that the use of the individual selection of biometric features descriptors increases the effectiveness of the identity verification process. Moreover, the research confirmed that for each user the selected set of descriptors can be individual.

### **2.3 Prediction of biometric attributes changes over time**

An important problem of behavioral biometrics is the susceptibility of biometric features to changes occurring over time. They result, for example, from physical changes occurring in the human body. This may cause decreasing the effectiveness of identity verification systems based on biometric attributes. Therefore, we have developed methods for predicting changes in biometric features' descriptors. The purpose of the prediction was improving the effectiveness of the identity verification process in the case when the time interval between individual biometric acquisition sessions was large.

For example, in paper [7] we have presented a method to predict changes of the dynamic signature descriptors which uses the capabilities of fuzzy systems and can work for any number of biometric features' descriptors. The input values of the system used for the prediction are values of the features of a single previous acquisition session. The evolutionary strategy ( $\mu + \lambda$ ) was used to learn this system. The considered method is based on a set of descriptors determined for features acquired in subsequent training sessions, which took place at certain intervals. The range of descriptor values within individual sessions is usually different (although it is considered for each user independently). Therefore, their values were normalized. The averaged values of the descriptors are part of the training and test sequences used in the learning and testing phase of the system used for prediction. This system is created for each user independently and in accordance with the assumption that the prediction is based only on the

descriptor values from the previous session (in order to increase the accuracy, the prediction may also include a larger number of previous sessions). For the prediction, the Mamdani type fuzzy system is used. To assess the operation of the fuzzy system in the learning phase, a standard RMSE error was used (its use makes sense thanks to the implemented normalization of features). This error is used in the phase of evolutionary learning to evaluate individuals encoding the parameters used for the prediction of the fuzzy system (the purpose of the learning algorithm is to minimize the error). As part of this work, simulations were carried out to predict the values of the global dynamic signature descriptors. They confirmed that for each user it is possible to predict the values of the considered descriptor set with good accuracy.

### 3 Conclusions

In this paper, we present a brief description of artificial intelligence methods which can be used to support identity verification using the dynamic signature. Our previous research has shown that they can significantly improve verification accuracy. Moreover, they can be used also in case of other behavioral biometric attributes which are described by signals changing over time, like the dynamic signature.

### References

1. Cpałka, K., Zalasiński, M. and Rutkowski, L. A new algorithm for identity verification based on the analysis of a handwritten dynamic signature. *Applied Soft Computing*, 43, pp. 47–56 (2016)
2. Cpałka, K. *Design of Interpretable Fuzzy Systems*, Springer (2017)
3. Fierrez, J. and Ortega-Garcia, J. On-Line Signature Verification. W: Jain A.K., Flynn P., Ross A.A. (ed) *Handbook of Biometrics*. Springer, Boston, MA (2008)
4. Nanni, L., Maiorana, E., Lumini, A. and Campisi, P. Combining local, regional and global matchers for a template protected on-line signature verification system. *Expert Systems with Applications*, 37, 3676–3684 (2010)
5. Prasad, M., Liu, Y.T., Li, D.L., Lin, Ch.T., Shah, R.R., Kaiwartya, O.P. A New Mechanism for Data Visualization with TSK-type Preprocessed Collaborative Fuzzy Rule based System, *Journal of Artificial Intelligence and Soft Computing Research*, vol. 7, 33–46 (2017)
6. Zalasiński, M., Cpałka, K. and Rakus-Andersson, E. An Idea of the Dynamic Signature Verification Based on a Hybrid Approach. *Lecture Notes in Computer Science*, 9693, 232–246 (2016)
7. Zalasiński, M., Łapa, K., Cpałka, K. and Saito, T. A Method for Changes Prediction of the Dynamic Signature Global Features over Time. *Lecture Notes in Computer Science*, 10245, 761–772 (2017)
8. Zalasiński, M. and Cpałka, K. (2018). A Method for Genetic Selection of the Dynamic Signature Global Features' Subset. *Advances in Intelligent Systems and Computing*, 655, 73–82 (2018)
9. Zalasiński, M., Cpałka, K. and Rutkowski, L. Fuzzy-Genetic Approach to Identity Verification Using a Handwritten Signature. *Advances in Data Analysis with Computational Intelligence Methods*, 738, 375–394 (2018)

# The use of new space properties of binary vectors in the set partitioning problem

Zbigniew Pliszka<sup>1</sup> and Olgierd Unold<sup>2[0000-0003-4722-176X]</sup>

<sup>1,2</sup>Wroclaw University of Science and Technology

Department of Computer Engineering

Wyb. Wyspianskiego 27, 50-370 Wroclaw, Poland

<sup>1</sup>zbigniew.pliszka@pwr.edu.pl

<sup>2</sup>olgierd.unold@pwr.edu.pl

**Abstract:** In the paper, we show the use of a transformed unit cube in a mutation tree (a special form of the tree described in [1]) to obtain the answer for the partition of a numerical set (the partition problem or number partitioning) being a question from the NP-complete problem class [3]. Tree cutting operations proved to be an indispensable tool, being (as it was shown in [2]) classes of abstractions for finite ordered sequences of positive integers. Definitions of conciliatory sets were also introduced as an extension of the relation of equal sums for two sets.

**Keywords:** optimization, trees, unit cube, number partitioning

## 1 The mutation tree

The  $n$ -dimensional unit (hyper)cube has the vertices in  $[0,1]^n$  (see Fig. 1). Such cubes are called boolean cubes and have always played and still play a primary and foremost role in modern computer science.

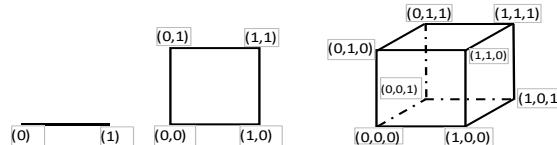


Fig. 1 The unit (boolean) cubes for  $n=1, 2$  and  $3$ .

The transformation of an  $n$ -dimensional unit cube into a tree called the mutation tree (Fig. 2) using the TreeM algorithm [1], allows to see in the tree structure the distribution of all combinations without repetition of an  $n$ -element set in such a way that for an arbitrary  $k$  ( $0 \leq k \leq n$ ) all combinations of  $n$  choose  $k$  are on one level. Another property, described in the same article, is that if you take an element  $(0, \dots, 0)$  as the root, then the positions of occurrences of "ones" are inherited by the descendants. Hence for the description  $(W_a, \{a_{n-1}, \dots, a_0\})$ , for which the weights and the constraint meet the conditions:

$$W_a \geq a_{n-1} \geq a_{n-2} \geq \dots \geq a_0,$$

assuming that  $\alpha_i = 1$  means including the weight  $a_i$  to the subset in question, and  $\alpha_i = 0$  means the opposite situation - we do not include it to this subset, for each element of the tree  $(\alpha_{n-1}, \dots, \alpha_0)$  (where  $\forall i \in \{0, \dots, n-1\} \alpha_i \in \{0, 1\}$ ), we have a bijectively assigned subset of weight indices  $\{x_{k-1}, \dots, x_0\}$ ,

$$\forall x \in \{n-1, \dots, 0\} \quad \forall \alpha_x \in \{0, 1\} \quad (x \in \{x_{k-1}, \dots, x_0\} \Leftrightarrow \alpha_x = 1).$$

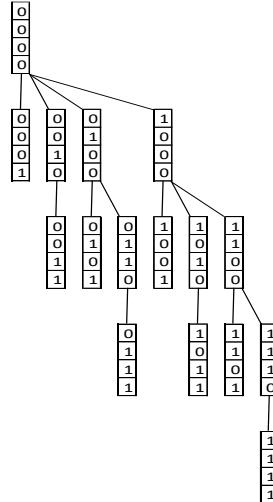


Fig. 2 The mutation tree for  $n=4$ .

### 3 The cut

In [2], attention was drawn to the similarities between the numerical sets due to the subsets contained in them, with constraints assigned to each of the sets. As an example, let us designate for the set  $\{2, 4, 5, 3\}$  all subsets of which the sum of elements does not exceed 5, and for the set  $\{1, 2, 2, 1\}$  let us designate all subsets of which the sum of elements does not exceed 2. Systematically we will perform the above task by sorting both sets (e.g., from the right side in non-decreasing order) and assigning indices to the elements according to the order obtained after the sorting operation (see Fig. 3).

For the set  $\{5, 4, 3, 2\}$  let us write out all non-empty sets of element indices (our answer will become universal) for which the sum of elements does not exceed 5:  $\{0\}, \{1\}, \{2\}, \{3\}, \{1, 0\}$ .  $\{5, 4, 3, 2\}$   
 $\uparrow \uparrow \uparrow \uparrow$   
 $3, 2, 1, 0$  ← indices  
 $\downarrow \downarrow \downarrow$   
 $\{2, 2, 1, 1\}$

Fig. 3 Indexing elements of sets.

All ten remaining (combinations of) sets do not meet the accepted condition (the sum of the elements exceeds 5). Note that if for the set  $\{2, 2, 1, 1\}$  the number limiting the sum of the elements of the searched subsets is 2, then we have the same division into subsets of indices meeting the conditions as in the first example.

Let there be a non-decreasing  $n + 1$ -element sequence of positive integers (for the whole paper, let us assume that  $n$  is a fixed, though arbitrary, positive integer):

$W_a \geq a_{n-1} \geq a_{n-2} \geq \dots \geq a_0$ .  
In each such sequence, let us assume the names:  $W_a$  - constraint,  $a_i$  - weights, and let the whole sequence be called a description of an ordered pair  $(W_a, \{a_{n-1}, \dots, a_0\})$ , in short, a description, when we know which ordered pair we are dealing with. Let us create a new description  $(W_b, \{b_{n-1}, \dots, b_0\})$  that meets the same conditions:

$$W_b \geq b_{n-1} \geq b_{n-2} \geq \dots \geq b_0.$$

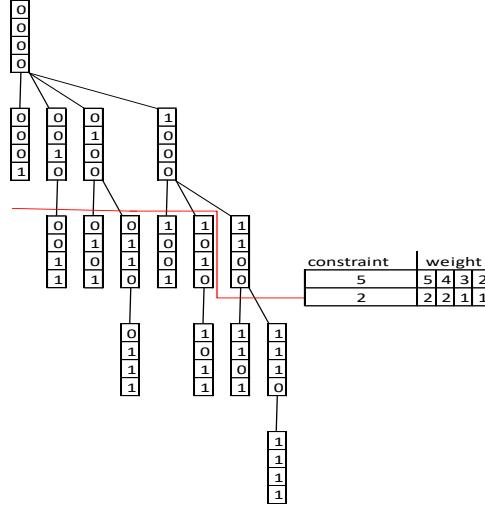


Fig. 4 Two descriptions belong to the single cut.

Definition: We shall state that two descriptions  $(W_a, \{a_{n-1}, \dots, a_0\})$  and  $(W_b, \{b_{n-1}, \dots, b_0\})$  belong to a single cut (or describe the same cut) if and only if for any subset  $\{x_{k-1}, \dots, x_0\}$  of the weight indices ( $\{x_{k-1}, \dots, x_0\} \subseteq \{n-1, \dots, 0\}$  for  $k \leq n$ ), exactly one of the two inequality systems is met:

$$\begin{cases} \sum_{j=0}^{k-1} a_{x_j} \leq W_a \\ \sum_{j=0}^{k-1} b_{x_j} \leq W_b \end{cases} \quad \text{xor} \quad \begin{cases} \sum_{j=0}^{k-1} a_{x_j} > W_a \\ \sum_{j=0}^{k-1} b_{x_j} > W_b. \end{cases} \quad (1)$$

For example, see Fig. 4.

Lemma

With a fixed  $n$ , for all descriptions, the relation of belonging to a single cut is the equivalence relation and divides the set of all descriptions into abstraction classes. The full proof is given in [2].

#### 4. Classification of cuts

In the description  $(W_a, \{a_{n-1}, \dots, a_0\})$ , the set of all weights  $\{a_{n-1}, \dots, a_0\}$  will be called conciliatory if there exist two non-empty sets of weight indices  $\{x_{k-1}, \dots, x_0\}$  and  $\{y_{t-1}, \dots, y_0\}$  are a dichotomy of a set of indices, and at the same time, the following alternative is true:

$$\begin{aligned}
 & \text{if } W_a < \sum_{i=0}^{n-1} a_i \leq 2W_a, \text{ then} & \sum_{i=0}^{k-1} a_{x_i} \leq W_a & \text{and} & \sum_{i=0}^{t-1} a_{y_i} \leq W_a \\
 & \text{or} & & & \\
 & \text{if } 2W_a < \sum_{i=0}^{n-1} a_i, \text{ then} & \sum_{i=0}^{k-1} a_{x_i} > W_a & \text{and} & \sum_{i=0}^{t-1} a_{y_i} > W_a.
 \end{aligned}$$

From the statements and conclusions contained in the work [2], we can conclude (see Fig. 5) that for two descriptions  $(W_a, \{a_{n-1}, \dots, a_0\})$  i  $(W_b, \{b_{n-1}, \dots, b_0\})$ :

- If  $\frac{\sum a_i}{W_a} \in (0,1]$  and  $\frac{\sum b_i}{W_b} \in (0,1]$ , then the descriptions  $(W_a, \{a_{n-1}, \dots, a_0\})$  and  $(W_b, \{b_{n-1}, \dots, b_0\})$  belong to the same cut. (Conclusion W2.)
- If  $\frac{\sum a_i}{W_a} \in (1,2]$  and  $\frac{\sum b_i}{W_b} \in (2, n]$ , while the sets  $\{a_{n-1}, \dots, a_0\}$  and  $\{b_{n-1}, \dots, b_0\}$  are not conciliatory in their descriptions and each solid compound holds simultaneously in both descriptions, then the descriptions  $(W_a, \{a_{n-1}, \dots, a_0\})$  and  $(W_b, \{b_{n-1}, \dots, b_0\})$  belong to a single cut. (Theorem 3.)
- If  $\frac{\sum a_i}{W_a} \in (1,2]$  and  $\frac{\sum b_i}{W_b} \in (2, n]$ , while the sets  $\{a_{n-1}, \dots, a_0\}$  and  $\{b_{n-1}, \dots, b_0\}$  are conciliatory in their descriptions, then the descriptions  $(W_a, \{a_{n-1}, \dots, a_0\})$  and  $(W_b, \{b_{n-1}, \dots, b_0\})$  do not belong to a single cut. (Conclusions W4 and W5.)

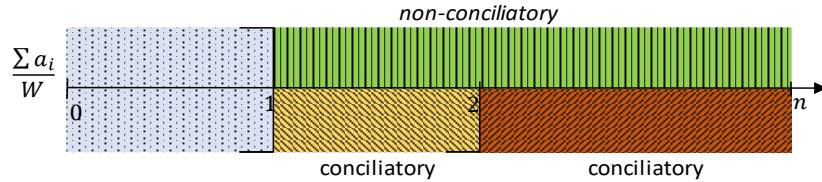


Fig. 5 Four ranges of abstraction classes of descriptions.

## 5. Summary

The paper shows the relationship between the unit cube, mutation tree, and the problem of division of sets. A cut (so far only used in graph theory) is the binder as an abstraction class for finite number sequences.

## References

1. Pliszka, Z., Unold, O.: On Transforming Unit Cube into Tree by One-Point Mutation. In: J. Świątek et al. (Eds.): ISAT 2018, AISC 853, 71-82 Springer (2019).
2. Pliszka, Z.: On some similarity of finite sets (and what we can say today about old problems) (2019). Paper in review.
3. Mertens, S.: The easiest hard problem: Number partitioning. Computational Complexity and Statistical Physics 125(2), 125-139 (2006).

# Ain't Nobody Got Time for Coding: Structure-Aware Program Synthesis from Natural Language

Jakub Bednarek, Karol Piaskowski, Krzysztof Krawiec

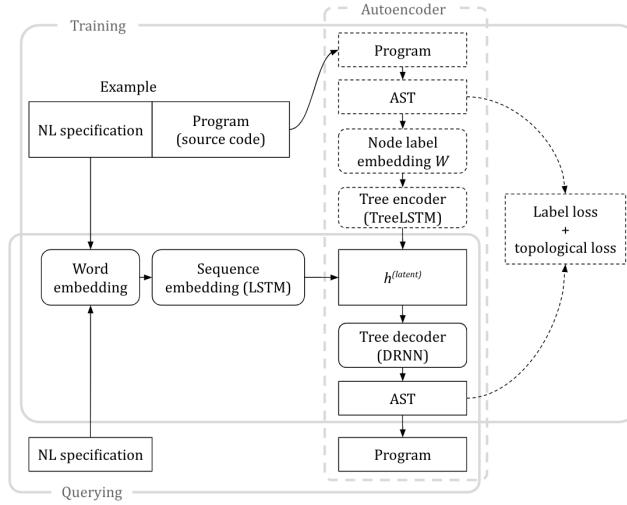
Institute of Computing Science  
Poznan University of Technology  
Poznań, 60-965, Poland  
`{jakub.bednarek,krzysztof.krawiec}@put.poznan.pl`  
`kar.piaskowski@gmail.com`

**Abstract.** We propose SAPS, an efficient method for synthesizing program source code from natural language, realized as an end-to-end neural network capable of mapping relatively complex, multi-sentence NL specifications to snippets of executable code. SAPS is trained on abstract syntax trees and uses a pretrained word embedding and a bi-directional multi-layer LSTM for processing of word sequences. SAPS performs on par or better than a method proposed in a previous study, producing correct programs in over 92% of cases. In contrast to other methods, it does not require post-processing code refinement, and uses a fixed-dimensional latent representation as the only interface between the NL analyzer and the source code generator.

## 1 Introduction

Program synthesis, i.e. automatic or semi-automatic generation of programs from specifications, can be posed in several ways. It is most common to assume that specification has the form of input-output pairs (tests). This approach is limited by its inductive nature: even if a program passes all provided tests, little can be said about generalization for other inputs. This is one of the rationales for synthesis from formal specifications, which are typically expressed as a pair of logical clauses (a *contract*) comprising a *precondition* and a *postcondition*. Programs so synthesized are correct by construction, but the task is NP-hard, and preparing specifications can be difficult for programmers.

From the practical perspective, the most intuitive and convenient way of specifying programs is natural language. This way of formulating synthesis tasks has been rarely studied in the past, but the recent progress in NLP and deep neural networks made it more realistic. Here, we propose Structure-Aware Program Synthesis (SAPS), an end-to-end approach to program synthesis from natural language. SAPS receives a short NL description of requested functionality and produces in response an executable snippet of code. We combine generic word and sequence embeddings with doubly-recurrent decoders trained on abstract syntax trees (ASTs), with the following contributions: (i) folding the entire NL specification into a fixed-dimensional latent space, which is then mapped by decoder onto the AST of a code snippet; (ii) modular architecture that facilitates usage of pretrained components; (iii) new signal propagation strategies in the decoder; and (iv) a ‘soft attention’ mechanism over the latent space.



**Fig. 1.** Overall architecture of SAPS. Rectangles mark data entities, rounded boxes are operations.

## 2 SAPS architecture

SAPS pipeline (Fig. 1) comprises (i) a word embedding for preprocessing of the specification, (ii) a mapping of the embedded word sequence tokens to a fixed-size latent representation  $h^{(latent)}$ , and (iii) a decoder that unfolds  $h^{(latent)}$  into an AST.

For word embedding, we rely on Common Crawl, the smallest pretrained GloVe embedding that covers all terms occurring in our NL specifications [5]. Given an NL query phrase of  $n$  tokens, this module produces a sequence of  $n$  300-dimensional vectors, which we pad with a special out-of-vocabulary value. To map the sequences of word embeddings of the NL specification to the latent space  $h^{(latent)}$ , we employ a multilayer bidirectional LSTM [74]. Because some variants of SAPS engage pretraining, the concatenated outputs of both vertical and horizontal LSTMs is passed through a tanh layer in order to match the  $h^{(latent)}$  dimensionality and the  $(-1, 1)$  range of values produced by the tree encoder. The final state of the topmost forward LSTM cell and the final state of the topmost backward cell are concatenated to form the final output.

We rely on the tree2tree autoencoder architecture we introduced in [3], based on TreeLSTM [8] for encoding and Doubly-Recurrent NN (DRNN) [1] for decoding. Our decoder comprises two LSTM cells that separately capture the *vertical* and *horizontal* traversal of the resulting AST tree. The state of the former is passed vertically downwards from parents to children, and of the latter horizontally left-to-right to consecutive siblings. Contrary to the original DRNN, we do not reset the state of the horizontal LSTM before processing the children of each node, as we found out that this might bring significant improvements (code pieces can be *semantically* related even if they reside in different subtrees of AST). We consider thus three variants  $H$ ,  $V$  and  $HV$ , in which the states of the horizontal and vertical LSTM cells are accordingly propagated or not. SAPS involves also an attention mechanism that relies only on  $h^{(latent)}$ .

**Table 1.** Percentage of programs passing all tests; non-SAPS results from [6].

| Model                | Validation    | Test          |
|----------------------|---------------|---------------|
| Attentional Seq2Seq  | 54.4%         | 54.1%         |
| Seq2Tree             | 61.2%         | 61.0%         |
| SAPSpre VH Att (256) | <b>86.67%</b> | <b>83.80%</b> |
| Seq2Tree + Search    | 86.1%         | 85.8%         |

### 3 Experiment

We assess the performance of SAPS on the set of NL-based program synthesis tasks proposed in [6] and formulated using AlgoLisp, a simple dialect of Lisp. A single program is composed as a nested list of instructions and operands of three types: `string`, `Boolean` and `function`. The dataset comprises 99506 examples, each being a pair of NL specification (e.g., *You are given an array a; find the smallest element in a, which is strictly greater than the minimum element in a.*), and the corresponding AlgoLisp program (respectively `(reduce (filter a (partial0 (reduce a inf) <)) inf min)`).

In Table [1] we juxtapose our best configuration, SAPSpre VH Att (256), with the results reported in [6]. The comparison metric is the percentage of programs that pass all tests provided (alongside with NL specifications and target programs) in the AlgoLisp database. SAPS performs on par with the Seq2Tree combined with external search algorithm, which was the best approach reported in the cited work (last row of the table). Though SAPS does not match its test-set performance, i.e. 85.8%, it is worse only by 2 percent point, while being trained end-to-end using gradient, rather than with the help of a sophisticated search mechanism and input-output tests.

Table [2] presents examples of programs synthesized with SAPS. Remarkably, the network produces correct output even when the input is very simplified and laconic. We evaluated also generalization by replacing operations (*multiplied* → *minimum*), applying different ranges of arrays, and complex modifications affecting multiple parts of NL specification simultaneously.

### 4 Conclusion

SAPS manages to achieve state-of-the-art test-set accuracy, on par with that of [6], and does so with a bare neural model, without explicit search, additional postprocessing or other forms of guidance. This remains in stark contrast to that study, where network was queried repeatedly in a Tree-Beam search heuristics to produce the target program step by step, testing the candidate programs on provided tests. There are also many differences with respect to [2], where a network was used to prioritize search conducted by an external algorithm. SAPS's architecture can provide similar accuracy with purely neural mechanisms and gradient descent as the learning mechanism. We find it likely that methods like SAPS may be successfully applied for practical synthesis of short code snippets, for instance in end-user programming. In future works, we plan to devise means to address the composite character of both specifications and program code.

**Acknowledgment.** We thank the authors of [6] for publishing their code and data.

**Table 2.** The effects of modifications of NL specification. The first specification in each group is an original task from the validation set, and those that follow are its modified variants (modifications marked in bold). Except for  $a \cdot b$ ,  $a + b$ , all programs are consistent with specification.

| Specification  | Synthesized program   |
|--|---|
| <i>you are given numbers <math>a</math> and <math>b</math>, your task is to find <math>a + b</math></i>  | $(+, a, b)$   |
| <i>you given numbers <math>a \cdot b</math>, your is find <math>a + b</math></i>   | $(+, a, b)$   |
| <i>given <b>a numbers</b> <math>b</math>, find <math>a + b</math></i>  | $(+, a, b)$   |
| <i>given a numbers <math>b</math>, <math>a + b</math></i>  | $(+, a, b)$   |
| <i><math>a \cdot b</math>, <math>a + b</math></i>  | $(+, (+, a, b), c)$   |
| <i>you are given numbers <math>a</math> and <math>b</math>, your task is to find <b>a multiplied by b</b></i>  | $(\star, a, b)$   |
| <i>you are given numbers <math>a</math> and <math>b</math>, your task is to find <b>minimum a and b</b></i>  | $(\min, a, b)$  |
| <i>given a number <math>a</math> and an array of numbers <math>b</math>, find the length of the longest subsequence of range from <b>0 to a inclusive</b> that is a prefix of <math>b</math></i> | $(\text{reduce}, (\text{range}, 0, (+, a, 1)), 0, (\lambda \text{arg2}, (\text{if}, (==, \text{arg2}, (\text{if}, (<, \text{arg1}, (\text{len}, b)), (\text{deref}, b, \text{arg1}), 0))), (+, \text{arg1}, 1), \text{arg1})))$ |
| <i>given a number <math>a</math> and an array of numbers <math>b</math>, find the length of the longest subsequence of range from <b>1 to a exclusive</b> that is a prefix of <math>b</math></i> | $(\text{reduce}, (\text{range}, 1, a), 0, (\lambda \text{arg2}, (\text{if}, (==, \text{arg2}, (\text{if}, (<, \text{arg1}, (\text{len}, b)), (\text{deref}, b, \text{arg1}), 0))), (+, \text{arg1}, 1), \text{arg1})))$         |
| <i>given an array of numbers <math>a</math>, find <b>median</b> of values in <math>a</math> after only keeping <b>first half</b></i>   | $(\text{deref}, (\text{sort}, (\text{slice}, a, 0, (/, (\text{len}, a), 2))), (/, (\text{len}, (\text{slice}, a, 0, (/, (\text{len}, a), 2))), 2))$   |
| <i>given an array of numbers <math>a</math>, find <b>mean</b> of values in <math>a</math> after only keeping <b>second half</b></i>  | $(/, (\text{reduce}, (\text{slice}, a, (/, (\text{len}, a), 2), (\text{len}, a)), 0, +), (\text{len}, (\text{slice}, a, (/, (\text{len}, a), 2), (\text{len}, a)), 2), (\text{len}, a))))$                                      |

## References

1. Alvarez-Melis, D., Jaakkola, T.S.: Tree-structured decoding with doubly-recurrent neural networks (2017)
2. Balog, M., Gaunt, A.L., Brockschmidt, M., Nowozin, S., Tarlow, D.: Deepcoder: Learning to write programs. CoRR (2016)
3. Bednarek, J., Piaskowski, K.: Efficient tree-2-tree autoencoders for end-to-end processing of abstract syntax trees (2018), master's Thesis
4. Graves, A., Fernández, S., Schmidhuber, J.: Bidirectional LSTM networks for improved phoneme classification and recognition. In: Proceedings of the 15th International Conference on Artificial Neural Networks: Formal Models and Their Applications - Volume Part II. pp. 799–804. ICANN'05, Springer-Verlag, Berlin, Heidelberg (2005)
5. Pennington, J., Socher, R., Manning, C.: Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). p. 1532–1543. Association for Computational Linguistics (2014)
6. Polosukhin, I., Skidanov, A.: Neural program search: Solving programming tasks from description and examples p. 11 (2018)
7. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing **45**, 2673–2681 (November 1997)
8. Tai, K.S., Socher, R., Manning, C.D.: Improved semantic representations from tree-structured long short-term memory networks. In: ACL (2015)

# Population-based Algorithms for Selecting Parameters and Structures of Various Crisp and Fuzzy Systems

Krystian Lapa<sup>1[0000-0002-3926-5685]</sup> and Krzysztof Cpałka<sup>1[0000-0001-9761-118X]</sup>

Institute of Computational Intelligence, Częstochowa University of Technology,  
Częstochowa, Poland  
`{krystian.lapa,krzysztof.cpalka}@iisi.pcz.pl`

**Abstract.** In certain optimization problems, the solutions and their qualities are defined by systems whose parameters are being optimized. Such systems include, for example, fuzzy systems and cascade PID controllers. However, the ideal structures of these systems are usually not known, and most often they are experimentally selected for the problem under consideration. Therefore solutions that enable the simultaneous selection of not only system parameters, but also its structure are sought. Such solutions can be obtained by combining, fine-tuning and extending population-based algorithms operating on different types of parameters. Selected proposed solutions, ideas and future aspects of the development and application of such methods are discussed in this paper.

**Keywords:** Population-based Algorithms · Evolutionary Computation · Structure and Parameter Selection · Fuzzy Systems · Control Systems.

## 1 Introduction

Population-based algorithms are usually inspired by nature, and although the number of new inspirations has peaked in recent years [1], many variations of existing algorithms are still emerging at a tremendous rate (see e.g. [2]). Their large number results from the freedom of their use in any optimization problems and the fact that there is no single best algorithm to solve all problems [3]. The list of such algorithms can be found e.g. in [4]. Most of them allow optimization of only problems for which solutions can be encoded by real number parameters (e.g. differential evolution) or by binary parameters (e.g. genetic algorithm). In certain optimization problems, the solutions and their qualities are defined by systems whose parameters are being optimized (e.g. fuzzy systems used for solving regression tasks). However, the ideal structures of such systems are usually not known, and most often they are experimentally selected for the problem under consideration. Therefore solutions that enable the simultaneous selection of not only system parameters, but also its structure are sought. Such solutions can be obtained by combining algorithms operating on different types of parameters. Selected proposed solutions and ideas will be discussed further in the paper.

## 2 Cascade PID controllers

The first example of systems in which both the structure and parameters can be selected are control systems based on P, I and D elements (PID controllers). These controllers are most often used in practice, but they have a number of disadvantages such as the ability to process only one signal. Cascade PID [5] systems become the solution to a one signal issues, but they require expert knowledge to develop a proper cascade structure. To solve this we proposed an approach that combines the possibilities of genetic programming (for the selection of cascade structure connections), genetic algorithm (for the selection of PID block structure) and evolutionary strategy (for the selection of PID block parameters) [6]. This combination requires, among others good development of operators of parameter and structure modification, selection of proper algorithm parameters (the smallest changes in structure have a great impact on the operation of the control system) and development of new system structures in which binary parameters can affect the reduction of its elements. The issue worth discussion is also taking into account the complexity of the system in the evaluation function and determining its impact on the obtained results (complexity vs accuracy). Therefore, the user is released from the required knowledge about the problem, and at the same time he can determine the importance of the system complexity.

## 3 Fuzzy systems

The second example of systems in which both the structure and parameters can be selected simultaneously are fuzzy systems. They can be used to solve regression, classification or control problems. In fuzzy systems, it is particularly important to obtain a system with good accuracy and the simplest possible structure. The simple structure translates to an increase in interpretability [7] of the system's operation - the most important advantage of fuzzy systems. For this problem, we propose new aggregation operators that take into account parameters that decide about the reduction of system elements (e.g. fuzzy rules and antecedences) and the use of mechanisms derived from the firework algorithm to select the real number parameters of the system [8]. In addition, the proposed system structure enabling dynamic exclusion and inclusion of system elements also enabled the possibilities of development and implementation of new interpretability criteria that can be included in the evaluation function. In this case, the right combination of modification mechanisms for various types of parameters allows for an optimal selection of the system structure, increasing its readability and at the same time ensuring adequate accuracy.

## 4 Proposed approaches and aspects of generalization

In our research we also investigated the use of multi-population algorithms [9], introducing new population initialization methods [10], and design of hybrid systems [11]. The proposed methods verified experimentally that both parameters

and system structures can be selected at the same time. It is worth noting, that none of the considered solutions proposed an algorithm inspired by a specific and new phenomenon in nature. It can be related to the fact that many of the new population-based algorithms have forcibly added inspiration - which may cause that many people question its originality [13]. It is worth considering whether general solutions should be created using the most effective mechanisms derived from various algorithms instead. We presented preliminary work on such idea in [12] where each individual of the population determines how it should be modified (which operators from different algorithms should be used to modify individual parameters and which operators should be spread in the population at a given time). At the moment we are working on a thorough examination of such mechanisms and the results obtained initially seem satisfactory.

## 5 Discussion and future development

The problem of using population algorithms to choose parameters and structures is still current. This is not only due to possibilities of using new population-based algorithms or mechanisms included therein but also due to not completely solved issues with proper maintaining of the balance in optimization mechanisms, increasing possibilities of using these algorithms in various fields but also due to an increase of computing power. The increase in computing power means that new solutions that use parallelization at different levels of the algorithm may arise (in the simplest case, launching many cooperating populations simultaneously). Due to the increase of computing power, also meta-optimization solutions can be found in the literature. Regarding applications, it is worth mentioning that methods for optimization of Convolutional Neural Network structures are being under development. The future aspects of developing parameters and structure selection population-based algorithms are therefore very promising.

## 6 Conclusions

In this paper, aspects of using population-based algorithms to simultaneously optimize the parameters and structures of various systems have been discussed and presented in examples.

## References

1. Fister Jr, I., Mlakar, U., Brest, J., Fister, I.: A new population-based nature-inspired algorithm every month: is the current era coming to the end. In Proceedings of the 3rd Student Computer Science Research Conference, University of Primorska Press, pp. 33–37 (2016).
2. Dawar, D., Ludwig, S. A.: Effect of strategy adaptation on differential evolution in presence and absence of parameter adaptation: an investigation. Journal of Artificial Intelligence and Soft Computing Research, vol. 8(3), pp. 211–235 (2018).

3. Maaranen, H., Miettinen, K., Penttinen, A.: On initial populations of a genetic algorithm for continuous optimization problems. *Journal of Global Optimization*, vol. 37(3), pp. 405 (2007).
4. Evolutionary Computation Bestiary, <https://github.com/fcampelo/EC-Bestiary>. Last accessed 12 Sept 2019
5. Bo, G., Xin, L., Hui, Z., Ling, W.: Quadrotor helicopter Attitude Control using cascade PID. In 2016 Chinese Control and Decision Conference (CCDC), pp. 5158–5163 (2016).
6. Cpałka, K., Łapa, K., Przybył, A.: Genetic Programming Algorithm for Designing of Control Systems. *Information Technology And Control*, vol. 47(4), pp. 668–683 (2018).
7. Liu, H., Gegov, A., Cocea, M.: Rule based networks: an efficient and interpretable representation of computational models. *Journal of Artificial Intelligence and Soft Computing Research*, vol. 7(2), pp. 111–123 (2017).
8. Łapa, K., Cpałka, K., Rutkowski, L.: New Aspects of Interpretability of Fuzzy Systems for Nonlinear Modeling. In *Advances in Data Analysis with Computational Intelligence Methods*, pp. 225–264 (2018).
9. Łapa, K., Cpałka, K., Paszkowski, J.: Hybrid Multi-population Based Approach for Controllers Structure and Parameters Selection. In *International Conference on Artificial Intelligence and Soft Computing*, pp. 456–468 (2019).
10. Łapa, K., Cpałka, K., Przybył, A., Grzanek, K: Negative space-based population initialization algorithm (NSPIA). In *International Conference on Artificial Intelligence and Soft Computing*, pp. 449–461 (2018).
11. Łapa, K., Cpałka, K. (2017). Flexible fuzzy PID controller (FFPIDC) and a nature-inspired method for its construction. *IEEE Transactions on Industrial Informatics*, vol. 14(3), pp. 1078–1088 (2017).
12. Łapa, K.: Population-Based Algorithm with Selectable Evolutionary Operators for Nonlinear Modeling. In *International Conference on Information Systems Architecture and Technology*, pp. 15–26 (2017).
13. Sørensen, K.: Metaheuristics—the metaphor exposed. *International Transactions in Operational Research*, vol. 22(1), pp. 3–18 (2015).

# On the development of the ASDM method

Paweł Wawryński, Paweł Zawistowski, and Łukasz Lepak<sup>1</sup>

Institute of Computer Science  
Warsaw University of Technology  
Nowowiejska 15/19, 00-665 Warsaw  
<http://ai.ii.pw.edu.pl/>

**Abstract.** We report on work-in-progress development of a method that automatically tunes step-sizes and momentum decay factors in ADAM algorithm. The approach is based on the estimation of the short- and long-term influence of these hyperparameters on the loss value and achieves very good results in the conducted experimental study.

**Keywords:** on-line learning · neural networks · gradient descent

## 1 Introduction and related work

In this paper, we consider the typical setting for on-line learning: we wish to optimize a parameter,  $\theta \in \mathbb{R}^d$ , of a learning system. For each time step, a known (momentary) loss function,  $J(\theta, \xi)$ , exists where  $\xi$  denotes a randomly generated data sample. The goal of learning is to find the point  $\theta^* \in \mathbb{R}^d$  for which the global loss function  $\bar{J}(\theta) = E J(\theta, \xi)$  attains its minimum. We assume that only the gradient of the momentary loss function,  $\nabla_\theta J(\theta, \xi)$ , is available, which is an unbiased estimate of the (unavailable) global loss gradient  $\nabla \bar{J}(\theta)$ .

Most fundamental methods of on-line learning, like classic momentum (CM) [4] require tuning hyperparameters called step-sizes and momentum decay factors that generally depend on the problem and process stage. In practice this is usually conducted using trial-and-error, which is time consuming and not satisfying. There have been attempts, like AdaGrad [1] or ADAM [3], to address this issue. However, these algorithms also require step-sizes and momentum decay factors, and their default values do not guarantee good performance for all learning problems.

In order to get rid of free parameters, the classic methods have been combined with gradient normalization in multiple approaches. AdaGrad [1] scales the gradient descent step at time  $t$  with an inversed diagonal matrix constructed using a cumulative sum of gradient products from times  $1, \dots, t$ , which unfortunately leads to decaying learning rates. The AdaDelta [7] and RMSProp [5] try to address these issues.

ADAM and ADAMAX [3] are currently among the most widely used optimization algorithms that utilize first and second-order moment corrections in their parameter update rules.

In this paper, we extend the method presented in [6], which optimizes the step-size and the momentum decay factor in ADAM while these methods are working. The main contribution of this paper is a novel method of analyzing the long-term influence of the parameters on the momentary loss value.

## 2 Problem formulation

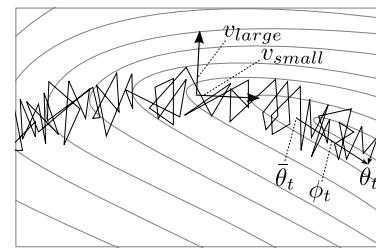
Algorithms like CM and ADAM utilize the history of past gradients observed during the optimization process while performing each iteration. This historical information, called momentum, gets updated in each iteration with the current gradient value. These updates in iteration  $t$  are steered by two hyperparameters:  $\lambda_t$  and  $\beta_t$ . The former controls forgetting of the historical information, while the latter scales the influence of the current gradient on the momentum value. The problem considered here is how to tune  $\beta_t$  and  $\lambda_t$  on the run of the optimization procedure to make it most efficient.

The main motivation behind the ASDM algorithm is, that analyzing the influence of the previous  $\beta$  and  $\lambda$  hyperparameter settings on the current values  $\theta_t$  should be a feasible way of tuning these hyperparameters. The values of  $\beta$  and  $\lambda$  are being incrementally adjusted in the direction that if these parameters were pushed before, current position of  $\theta_t$  would be better.

The idea of optimization of the current loss,  $J(\theta_t, \xi_t)$ , by manipulating previous values of  $\beta$  and  $\lambda$  has already been utilized in the original ASDM method. However, when applied directly, this yields small hyperparameter values, as these suppress random fluctuations of  $\theta_t$  thereby minimizing the current loss. Unfortunately this eventually slows down learning, therefore here we analyze the long term influence of  $\beta$  and  $\lambda$  on this process. The intuition behind this analysis is presented below.

Let us visualize the optimization process as a descent down a multidimensional valley between steep slopes (Fig. 1). The goal is to descend in the direction of the bottom of this valley. However, the strategy we want to follow is to make this descent fast, possibly at the cost of bouncing between these slopes, rather than slowly and steadily progressing along the flatter bottom. In such a case, locating the slopes and the bottom of the valley may be performed by analyzing its curvature modelled using the Hessian  $\nabla^2 \bar{J}(\theta_t)$ .

Large eigenvalues of the Hessian are associated with eigenvectors along which the gradient changes fast ( $v_{large}$  in Fig. 1), which makes  $\theta_t$  oscillate along these directions. This corresponds to the directions of "slopes". On the other hand, small eigenvalues are associated with eigenvectors along which the gradi-



**Fig. 1.** Black polygonal chain illustrates the  $\theta_t$  trajectory. Smooth gray lines: contour lines of  $\bar{J}$ .

ent changes slowly ( $v_{small}$  in Fig. 1), which makes  $\theta_t$  move stably along these directions. This corresponds to the direction of "the bottom".

Thus, the described strategy can be realized with proper hyperparameter settings. When manipulating  $\beta_i$  and  $\lambda_i$ , one observes that increasing  $\beta_i$  and  $\lambda_i$  makes  $\theta_t$  bounce higher up the slopes, and causes larger momentary  $J$ . This is the short-term influence of  $\beta_i$  and  $\lambda_i$  on  $\theta_t$ , which was already analyzed in [6].

Also, increasing  $\beta_t$  and  $\lambda_t$  makes  $\theta_t$  move faster along the valley, and causes smaller future  $J$ . This long-term influence is of  $\beta_i$  and  $\lambda_i$  is analyzed by focusing on  $\bar{\theta}_t$ . In our valley metaphor,  $\bar{\theta}_t$  plays the role of the projection of  $\theta_t$  on the valley. Thus optimizing  $\beta_t$  and  $\lambda_t$  to approximately minimize a linear combination of  $J(\theta_t, \xi_t)$  and  $J(\bar{\theta}_t, \xi_t)$  includes both short- and long-term influence.

The aggregation of the impact of  $\beta_i$  and  $\lambda_i$  on future values of  $\theta_t$  is done by means of an operator,  $\mathcal{S}_\gamma$  [6], which is defined for  $\gamma \in (0, 1]$  as  $S_\gamma \frac{dv}{d\alpha_k} = \sum_{i \leq k} \gamma^{k-i} \frac{dv}{d\alpha_i}$ . Using  $\mathcal{S}_\gamma$ , it is easy to quantify the influence of  $\beta$  and  $\lambda$  on  $\theta_t$  and  $\bar{\theta}_t$ .

Combining these observations with the method presented in [6] and ADAM yields the ASDM2/n algorithm, where /n stands for gradient normalization.

### 3 Experimental study

This section reports experiments with the algorithms presented in the previous section. The new algorithm ASDM2/n, is compared with the following known ones: CM, AG, ADAM, AdaGrad, and AdaDelta considered in two settings: with optimized and default hyperparameters. The optimized hyperparameters are the momentum decay factor and the step-size selected from the Cartesian product  $\{0.9, 0.99\} \times \{0.1, 0.05, 0.02, 0.01\}$ . The default parameters are ones applied by Tensorflow when their values are not provided when calling the algorithm. The algorithms are tested in training shallow neural classifiers for 10 arbitrary classification problems from the UCI Machine Learning Repository [2].

The results are depicted in Tab. 1 in the form of average losses attained at the end of training and their standard deviations. Only training losses are reported, as here we focus on optimization rather than the actual quality of the models. The smallest losses for each problem are indicated by bold face font.

In most cases (8 cases out of 10) the winner is ASDM2/n. The manual optimization of the parameters of ADAM yields significant improvement of its behavior. Still, its default parameters make that algorithm perform well in comparison to others, except all variants of ASDM2. CM had its moment of glory in the case of Robot. That algorithm outperformed all the rest so strongly that we had to double-check the setting and repeat all the experiments with Robot. However, the results were the same. The performance of AdaGrad and AdaDelta is especially disappointing. Those algorithms were presented as a way to optimize the step-size on-the-fly in SGD and CM, respectively. That does not check out in our experiments.

**Table 1.** Final mean loss estimates for the tested algorithms averaged from 10 independent runs. /d — default step-size and momentum decay factor, /o — optimized ones. The standard deviation is presented with the same number of digits after the decimal point but without leading zeros e.g.,  $15.35 \pm 31$  denotes  $15.35 \pm 0.31$ , and  $0.0055 \pm 4$  denotes  $0.0055 \pm 0.0004$ .

| Alg.<br>Problem | CM                               | AG              | ADAM            |                                 | AdaGrad         | AdaDelta        |                 | ASDM2/n                         |
|-----------------|----------------------------------|-----------------|-----------------|---------------------------------|-----------------|-----------------|-----------------|---------------------------------|
|                 | /d                               | /o              | /d              | /o                              | /d              | /o              | /d              | /d                              |
| CCard           | 0.262 $\pm 1$                    | 0.264 $\pm 1$   | 0.262 $\pm 1$   | 0.260 $\pm 1$                   | 0.266 $\pm 1$   | 0.342 $\pm 2$   | 0.265 $\pm 1$   | <b>0.256 <math>\pm 1</math></b> |
| Dota2           | 0.422 $\pm 1$                    | 0.418 $\pm 1$   | 0.412 $\pm 1$   | 0.394 $\pm 1$                   | 0.405 $\pm 1$   | 0.500 $\pm 1$   | 0.423 $\pm 1$   | <b>0.379 <math>\pm 1</math></b> |
| Htru2           | 0.0326 $\pm 7$                   | 0.0300 $\pm 3$  | 0.0313 $\pm 5$  | 0.0293 $\pm 5$                  | 0.0294 $\pm 4$  | 0.1033 $\pm 16$ | 0.0321 $\pm 5$  | <b>0.0291<math>\pm 4</math></b> |
| Motor           | 0.0642 $\pm 10$                  | 0.0480 $\pm 8$  | 0.0668 $\pm 90$ | 0.0415 $\pm 5$                  | 0.0829 $\pm 9$  | 0.802 $\pm 16$  | 0.0956 $\pm 13$ | <b>0.0413<math>\pm 7</math></b> |
| Poker           | 0.451 $\pm 10$                   | 0.315 $\pm 13$  | 0.448 $\pm 15$  | 0.327 $\pm 3$                   | 0.477 $\pm 7$   | 0.569 $\pm 1$   | 0.524 $\pm 3$   | <b>0.302 <math>\pm 3</math></b> |
| Robot           | <b>0.0642<math>\pm 21</math></b> | 0.0908 $\pm 23$ | 0.1198 $\pm 14$ | 0.1064 $\pm 12$                 | 0.1270 $\pm 13$ | 0.6178 $\pm 19$ | 0.1251 $\pm 17$ | 0.0925 $\pm 44$                 |
| Shuttle         | 0.0064 $\pm 7$                   | 0.0045 $\pm 3$  | 0.0084 $\pm 3$  | <b>0.0038<math>\pm 2</math></b> | 0.0093 $\pm 3$  | 0.2768 $\pm 41$ | 0.0135 $\pm 4$  | 0.0059 $\pm 2$                  |
| Skin            | 0.0062 $\pm 2$                   | 0.0055 $\pm 1$  | 0.0074 $\pm 1$  | 0.0043 $\pm 2$                  | 0.0074 $\pm 2$  | 0.2573 $\pm 75$ | 0.0089 $\pm 2$  | <b>0.0037<math>\pm 2</math></b> |
| Spam            | 0.0245 $\pm 10$                  | 0.0230 $\pm 3$  | 0.0279 $\pm 3$  | 0.0172 $\pm 4$                  | 0.0257 $\pm 3$  | 0.4270 $\pm 56$ | 0.0390 $\pm 5$  | <b>0.0171<math>\pm 3</math></b> |
| Theo            | 0.442 $\pm 2$                    | 0.409 $\pm 1$   | 0.430 $\pm 2$   | 0.393 $\pm 1$                   | 0.440 $\pm 1$   | 0.720 $\pm 1$   | 0.454 $\pm 1$   | <b>0.381 <math>\pm 2</math></b> |

## 4 Conclusions

In this paper we present a work-in-progress on automatic tuning of hyperparameters in on-line learning. We apply the way of analyzing influence of the hyperparameters on  $\theta_t$  taken from [6] to analyze the influence of the hyperparameters on the center of mass of the recent trajectory of  $\theta_t$ . That leads to a method of hyperparameters tuning that yields very encouraging preliminary results.

**Acknowledgments** We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research.

## References

1. Duchi, J., Hazan, E., Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research* **12**, 2121–2159 (2011)
2. Frank, A., Asuncion, A.: UCI machine learning repository (2010), <http://archive.ics.uci.edu/ml>
3. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: CoRR. vol. abs/1412.6980 (2014)
4. Polyak, B.T.: Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics* **4**, 1–17 (1964)
5. Tieleman, T., Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude (2012)
6. Wawrzynski, P.: ASD+M: Automatic parameter tuning in stochastic optimization and on-line learning. *Neural Networks* **96**, 1–10 (2017)
7. Zeiler, M.D.: Adadelta: An adaptive learning rate method. In: arXiv:1212.5701 (2012)

# Differential Evolution Strategy: a differential evolution version of the Covariance Matrix Adaptation Evolution Strategy

Jarosław Arabas and Dariusz Jagodziński

Warsaw University of Technology, Institute of Computer Science  
[jarabas@elka.pw.edu.pl](mailto:jarabas@elka.pw.edu.pl), [d.jagodzinski@elka.pw.edu.pl](mailto:d.jagodzinski@elka.pw.edu.pl)

**Abstract.** According to the experimental results using CEC and BBOB benchmark set families, Differential Evolution (DE) and the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) are the best performing metaheuristic algorithms for numerical optimization. We present a crossover between these two — a Differential Evolution Strategy which is a version of DE which processes points in a way similar to CMA-ES. The contribution is based on the article [2].

**Keywords:** CMA-ES method, Differential Evolution

## 1 Covariance Matrix Evolution Strategy and its equivalent based on differential mutation

The starting point for our considerations is a version of the CMA-ES method [3] with equal weights used for the update of midpoint and covariance matrix. CMA-ES maintains multivariate normal distribution which is characterized with the reference point  $\mathbf{m}^{(t)}$ , the covariance matrix  $\mathbf{C}^{(t)}$  and the step-size multiplier  $\sigma^{(t)}$ ; symbol  $t$  stands for the iteration index. Initially, the covariance matrix  $\mathbf{C}^{(t)}$  is the identity matrix  $\mathbf{I}$ , and values of  $\mathbf{m}^{(t)}, \sigma^{(t)}$  are given by the user. In each iteration, this distribution is realized  $\lambda$  times and yields a population of points from the search space. Then the points are evaluated according to the fitness function and the subset of  $\mu$  best points is used to update the distribution parameters.

The generation of new points is a two-stage process. In the first stage, a population of vectors  $D^{(t)} = \{\mathbf{d}_i^{(t)}, i = 1, \dots, \lambda\}$  is generated as  $\lambda$  independent normal variates with zero expectation and the covariance matrix  $\mathbf{C}^{(t)}$ . We shall call them the difference vectors. Then, points  $\mathbf{x}_i^{(t)}$ , called the individuals, are created by scaling the difference vectors with the parameter  $\sigma^{(t)}$  and adding them to the reference point  $\mathbf{m}^{(t)}$ . Once the population of individuals has been created and evaluated, individuals are sorted according to the fitness function. The population of  $\mu$  difference vectors which correspond to the best  $\mu$  individuals is the base to update the covariance matrix for the next iteration,  $\mathbf{C}^{(t+1)}$ , which is a weighted average of the current covariance matrix  $\mathbf{C}^{(t)}$  and the so-called rank-1 and rank- $\mu$  update matrices,  $\mathbf{C}_1^{(t)}$  and  $\mathbf{C}_\mu^{(t)}$ . Coefficients that define the covariance matrix update rule satisfy the conditions  $c_{cov} = c_1 + c_\mu$  and  $0 \leq c_1, c_\mu, c_{cov} \leq 1$ .

**Algorithm 1** Covariance Matrix Adaptation Evolution Strategy (CMA-ES)

---

```

1:  $\mathbf{p}_c^{(1)} \leftarrow \mathbf{0}$ ,  $\mathbf{p}_\sigma^{(1)} \leftarrow \mathbf{0}$ ,  $\mathbf{C}^{(1)} \leftarrow \mathbf{I}$ ,  $t \leftarrow 1$ 
2: initialize( $\mathbf{m}^{(1)}, \sigma^{(1)}$ )
3: while !stop do
4:   for  $i = 1$  to  $\lambda$  do
5:      $\mathbf{d}_i^{(t)} \sim N(\mathbf{0}, \mathbf{C}^{(t)})$ 
6:      $\mathbf{x}_i^{(t)} \leftarrow \mathbf{m}^{(t)} + \sigma^{(t)} \mathbf{d}_i^{(t)}$ 
7:   end for
8:   evaluate ( $X^{(t)} = \{\mathbf{x}_1^{(t)}, \dots, \mathbf{x}_\lambda^{(t)}\}$ )
9:    $\Delta^{(t)} \leftarrow \frac{1}{\mu} \sum_{i=1}^{\mu} \mathbf{d}_i^{(t)}$ ,  $\mathbf{m}^{(t+1)} \leftarrow \mathbf{m}^{(t)} + \sigma^{(t)} \Delta^{(t)}$ 
10:   $\mathbf{p}_\sigma^{(t+1)} \leftarrow (1 - c_\sigma) \mathbf{p}_\sigma^{(t)} + \sqrt{\mu c_\sigma (2 - c_\sigma)} \cdot (\mathbf{C}^{(t)})^{-\frac{1}{2}} \Delta^{(t)}$ 
11:   $\mathbf{p}_c^{(t+1)} \leftarrow (1 - c_c) \mathbf{p}_c^{(t)} + \sqrt{\mu c_c (2 - c_c)} \cdot \Delta^{(t)}$ 
12:   $\mathbf{C}_1^{(t)} \leftarrow \mathbf{p}_c^{(t)} \left( \mathbf{p}_c^{(t)} \right)^T$ ,  $\mathbf{C}_\mu^{(t)} \leftarrow \frac{1}{\mu} \sum_{i=1}^{\mu} \mathbf{d}_i^{(t)} \left( \mathbf{d}_i^{(t)} \right)^T$ 
13:   $\mathbf{C}^{(t+1)} \leftarrow (1 - c_{cov}) \mathbf{C}^{(t)} + c_1 \mathbf{C}_1^{(t)} + c_\mu \mathbf{C}_\mu^{(t)}$ 
14:   $\sigma^{(t+1)} \leftarrow \sigma^{(t)} \exp \left( \frac{c_\sigma}{d_\sigma} \left( \frac{\|\mathbf{p}_\sigma^{(t+1)}\|}{E \|N(\mathbf{0}, \mathbf{I})\|} - 1 \right) \right)$ 
15:   $t \leftarrow t + 1$ 
16: end while

```

---

The dynamics of the covariance matrix adaptation process is improved by an additional element — the step size multiplier  $\sigma^{(t)}$  which adjusts the overall step size in an adaptive way.

In its original formulation, CMA-ES is based on matrix operations which are costly. We propose the Covariance Matrix Adaptation Differential Evolution method which generates individuals in a way similar to CMA-ES, except for the step size multiplier adaptation.

**Theorem 1.** *The probability distribution which is used by CMA-DE to generate the difference vectors  $\mathbf{d}_i^{(t)}$  has zero expectation vector and its covariance matrix  $\Sigma[\mathbf{d}_i^{(t)}]$  equals the covariance matrix  $\mathbf{C}^{(t+1)}$  maintained by CMA-ES*

$$\begin{aligned} \Sigma[\mathbf{d}_i^{(t)}] &= \mathbf{C}^{(t+1)} = c_\mu \frac{\mu - 1}{\mu} \sum_{\tau=1}^t (1 - c_{cov})^{t-\tau} \mathbf{S}(D_\mu^{(\tau)}) + c_1 \sum_{\tau=1}^t (1 - c_{cov})^{t-\tau} \mathbf{p}^{(\tau)} \left( \mathbf{p}^{(\tau)} \right)^T \\ &\quad + c_\mu \sum_{\tau=1}^t (1 - c_{cov})^{t-\tau} \Delta^{(\tau)} \left( \Delta^{(\tau)} \right)^T + (1 - c_{cov})^t \mathbf{I} \end{aligned} \quad (1)$$

where  $D_\mu^{(\tau)}$  is the set of  $\mu$  difference vectors selected according to the fitness of their corresponding individuals in generation  $\tau$  and  $\mathbf{S}(D_\mu^{(\tau)})$  is the empirical covariance matrix of the set  $D_\mu^{(\tau)}$ .

Note that, despite the equivalence of expectation vectors and covariance matrices between CMA-DE and CMA-ES, the probability distributions to generate

**Algorithm 2** Covariance Matrix Adaptation Differential Evolution (CMA-DE)

---

```

1:  $\mathbf{p}^{(0)} \leftarrow 0$ , initialize  $(X^{(1)} = \{\mathbf{x}_1^{(1)}, \dots, \mathbf{x}_{\lambda}^{(1)}\})$ ,  $\mathbf{m}^{(1)} = \frac{1}{\lambda} \sum_{i=1}^{\lambda} \mathbf{x}_i^{(1)}$ ,  $t \leftarrow 1$ ,
2: evaluate  $(X^{(1)})$ 
3: while !stop do
4:    $\mathbf{m}^{(t+1)} = \frac{1}{\mu} \sum_{i=1}^{\mu} \mathbf{x}_i^{(t)}$ ,  $\Delta^{(t)} \leftarrow \mathbf{m}^{(t+1)} - \mathbf{m}^{(t)}$ 
5:    $\mathbf{p}^{(t)} \leftarrow (1 - c_c)\mathbf{p}^{(t-1)} + \sqrt{\mu c_c(2 - c_c)}\Delta^{(t)}$ 
6:   for  $i = 1$  to  $\lambda$  do
7:      $\tau_1, \tau_2, \tau_3 \sim G_{cov}\{1, \dots, t\}$ 
8:      $j, k \sim U(1, \dots, \mu)$ 
9:      $\mathbf{d}_i^{(t)} \leftarrow \sqrt{\frac{c_1}{\alpha(t)c_{cov}}}\mathbf{p}^{(\tau_1)} \cdot N(0, 1) + \sqrt{\frac{c_\mu}{\alpha(t)c_{cov}}}\Delta^{(\tau_2)} \cdot N(0, 1)$ 
10:     $+ \sqrt{\frac{c_\mu}{2\alpha(t)c_{cov}}} (\mathbf{x}_j^{(\tau_3)} - \mathbf{x}_k^{(\tau_3)}) + (1 - c_{cov})^{t/2} \cdot N(\mathbf{0}, \mathbf{I})$ 
11:     $\mathbf{x}_i^{(t+1)} \leftarrow \mathbf{m}^{(t+1)} + \mathbf{d}_i^{(t)}$ 
12:   end for
13:   evaluate  $(X^{(t+1)})$ 
14:    $t \leftarrow t + 1$ 
15: end while

```

---

individuals by both methods are different. In CMA-ES, they are generated with a single multivariate normal random variable, whereas CMA-DE uses a mixture of multivariate normal, univariate normal, and discrete random variables.

## 2 Differential Evolution Strategy

In [2] we discuss another algorithm inspired by CMA-ES. The method, called DES (Differential Evolution Strategy), uses a simple moving average instead of the exponential smoothing in the formula that defines the covariance matrix of difference vectors  $\Sigma [\mathbf{d}^{(t)}]$ . Then the number of values to be recorded is of the order  $H \cdot \lambda \cdot n$ , where  $H$  is the window size. An additional feature of DES is the evaluation of the fitness function in the population midpoint  $\mathbf{m}^{(t)}$ . This feature was inspired by [1] where the authors show that efficiency of various evolutionary methods may be increased by evaluating the population midpoint.

Similarly to CMA-ES, CMA-DE and DES are also inherently insensitive to translation, rotation, reflection, and scaling, since the methods involve only affine transformations of individuals. They are also insensitive to any order-preserving transformation of the fitness function since, in the selection phase, individuals are selected on the basis of their positions in the list sorted according to their fitness.

According to the results of benchmarking on BBOB and CEC family benchmark sets, DES yields competitive results to CMA-ES — see [2] for the detailed data. Example ECDF curves for are depicted in Fig. 1.

**Algorithm 3** Differential Evolution Strategy (DES)

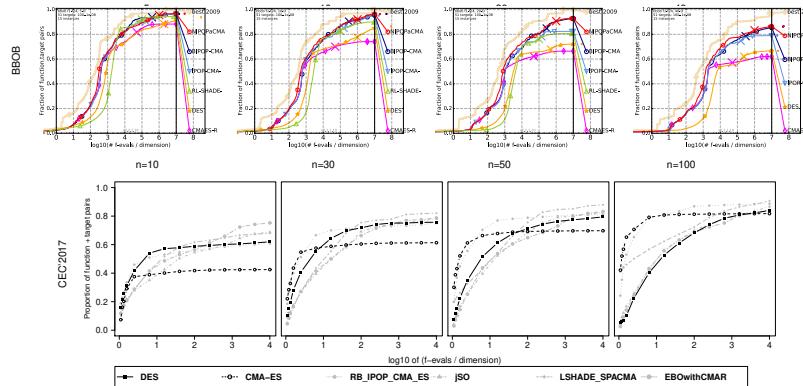
---

```

1: initialize  $(X^{(1)} = \{\mathbf{x}_1^{(1)}, \dots, \mathbf{x}_\lambda^{(1)}\})$ ,  $\mathbf{m}^{(1)} = \frac{1}{\lambda} \sum_{i=1}^{\lambda} \mathbf{x}_i^{(1)}$ ,  $t \leftarrow 1$ 
2: evaluate  $(X^{(1)}, \mathbf{m}^{(1)})$ 
3: while !stop do
4:    $\mathbf{m}^{(t+1)} = \frac{1}{\mu} \sum_{i=1}^{\mu} \mathbf{x}_i^{(t)}$ ,  $\Delta^{(t)} \leftarrow \mathbf{m}^{(t+1)} - \mathbf{m}^{(t)}$ 
5:    $\mathbf{p}^{(t)} \leftarrow (1 - c_c)\mathbf{p}^{(t-1)} + \sqrt{\mu c_c(2 - c_c)}\Delta^{(t)}$ 
6:   for  $i = 1$  to  $\lambda$  do
7:     pick at random  $\tau_1, \tau_2, \tau_3 \in \{1, \dots, H\}$ ,  $j, k \sim U(1, \dots, \mu)$ 
8:      $\mathbf{d}_i^{(t)} \leftarrow \sqrt{\frac{c_d}{2}} (\mathbf{x}_j^{(t-\tau_1)} - \mathbf{x}_k^{(t-\tau_1)}) + \sqrt{c_d}\Delta^{(t-\tau_2)} \cdot N(0, 1)$ 
9:      $+ \sqrt{1 - c_d}\mathbf{p}^{(t-\tau_3)} \cdot N(0, 1) + \varepsilon \cdot (1 - c_e)^{t/2} \cdot N(\mathbf{0}, \mathbf{I})$ 
10:     $\mathbf{x}_i^{(t+1)} \leftarrow \mathbf{m}^{(t+1)} + \mathbf{d}_i^{(t)}$ 
11:   end for
12:   evaluate  $(X^{(t+1)}, \mathbf{m}^{(t+1)})$ 
13:    $t \leftarrow t + 1$ 
14: end while

```

---

**Fig. 1.** ECDF curves of results yielded by DES and CMA-ES vs. selected other methods

## References

- Arabas, J., Biedrzycki, R.: Improving evolutionary algorithms in a continuous domain by monitoring the population midpoint. *IEEE Trans. Evol. Comput.* **21**(5), 807–812 (2017)
- Arabas, J., Jagodziński, D.: Towards a matrix-free Covariance Matrix Adaptation Evolution Strategy. *IEEE Transactions on Evolutionary Computation* (2019)
- Hansen, N.: The CMA evolution strategy: a comparing review. In: Lozano, J.A. (ed.) *Towards a new evolutionary computation: advances on estimation of distribution algorithms*, pp. 75–102. Springer (2006)

# Optimization of ultra-thin magnetron sputtered aluminum films with the use of AI models

Eryk Warchałski<sup>1</sup> and Robert Mroczynski<sup>2[0000-0002-7067-6247]</sup>  
and Jarosław Arabas<sup>1[0000-0002-5699-947X]</sup>

<sup>1</sup> Warsaw University of Technology, Institute of Computer Science

<sup>2</sup> Warsaw University of Technology, Institute of Microelectronics and Optoelectronics

**Abstract.** We report on a successful application of AI models in the microelectronics technology field. The process of aluminum layer fabrication was modeled with various regression models. Then two metaheuristic methods were applied to minimize Al layer roughness which was predicted by the regression models. The resulting optimal values of the fabrication process parameters were applied to control the process. In effect, the Al layer roughness was reduced by the order of magnitude. The contribution is based on the paper [1].

**Keywords:** neural networks, support vector regression, CMA-ES

## 1 Scope of the contribution

Conductive materials fabricated from aluminum films are commonly used in nowadays semiconductor technologies in the form of connections or inter-metal dielectric layers. Aluminum films find also numerous applications in optoelectronics and photovoltaics as reflective coatings and lateral current collective material. The inhomogeneities of thickness are of little importance in these applications, however, modern technologies of nanoelectronic and photonic devices demand the fabrication of conductive films in the thin, and ultra-thin regime (up to 20 nm). The roughness of such films is becoming increasingly important, as it influences the optical properties and integrity of multi-layer structures.

This study presents the results of application of metaheuristics-based optimization of parameters of the Al film fabrication process. The optimization is based on the process model. After optimum settings of the process models has been found, the settings are used to control the real fabrication process. In this contribution we verify the accuracy of various models to be applied for the considered process. Then we present the results of optimization of the process model with two different metaheuristic methods. Finally, we overview the quality of Al films which have been obtained with the optimized setup of the manufacturing technology, and we compare it with the state-of-the-art solution that is based on Taguchi orthogonal tables approach [2].

## 2 Modeling and optimization of the Al film manufacturing

### 2.1 The Al manufacturing process

In this study, aluminum films were fabricated by means of magnetron sputtering method using a PlasmaLab System 400s made by Oxford Instruments Plasma Technology. The thickness and structural properties of investigated Al films were characterized by means of spectroscopic ellipsometry, Atomic Force Microscopy (AFM), Scanning Electron Microscopy (SEM), and High-Resolution Transmission Electron Microscopy (HR-TEM).

### 2.2 Modeling methods and results

The Al film fabrication process was modeled using several types of models. We started from linear regression as a baseline model, we considered two versions of spline models, the Support Vector Machine with two different kernels, and the multi-layer perceptron.

The manufacturing process was characterized by four parameters: the chamber pressure, the heating facility power, the argon flow, and the process duration. Since the manufacturing process is costly and time-consuming, only 50 combinations of the aforementioned parameter values were applied in the phase of model preparation. For each considered combinations, we measured the resulting Al roughness. Thus we obtained 50 samples to perform the regression task, with Al roughness being the dependent variable and the process parameters being the independent variables. The regression was aimed at minimization of the mean square error. The models were built using solely the experimental data.

Each model, except for the linear regression, was built with the use of 10-fold crossvalidation. Table 1 lists the list of considered models along with the mean squared error value obtained for the best model from each class.

**Tab. 1.** The best MSE values obtained for various types of models for the experimental data.

| model type                                    | best MSE value |
|---|----------------|
| Support Vector Regression (polynomial kernel) | 2.13e-01       |
| Support Vector Regression (radial kernel)     | 1.52e-01       |
| Multivariate adaptive regression splines      | 3.81e-01       |
| Thin plate spline                             | 10.1e-02       |
| Multilayer perceptron                         | 2.05e-01       |
| Linear model                                  | 6.12e-01       |

### 2.3 Optimization methods and results

During the optimization process, the model was used to predict the Al film roughness which was subject to optimization with respect to the process parameters. The optimization methods included CMA-ES and DES. According to the results of black-box continuous optimization benchmarking, CMA-ES is the leading method together with Differential Evolution (DE). DES is a crossover of CMA-ES and DE. A comprehensive discussion of these methods, together with the comparison of their efficiency, can be found in [3].

CMA-ES and DES were used to solve the optimization problem where the objective function was defined as the predicted Al roughness, and the optimizer was allowed to play with different settings of the Al fabrication process parameters. Each optimization method was run 10 times and the best solution was reported. The parameter values defined by that solution were applied to control the fabrication process. Then the roughness of the Al layer was measured and reported.

Results of optimization are presented in tab 2. According to the results, DES and CMA-ES revealed highly similar performance so the name of the particular method that yielded the best parameter settings for each model is omitted.

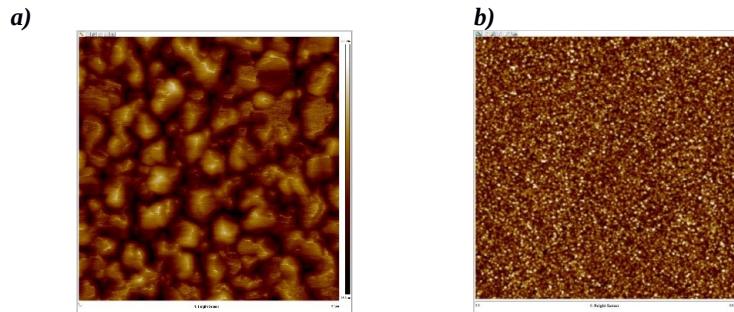
**Tab. 2.** Roughness of Al layers fabricated by the process controlled with settings that resulted from optimization with the various models that predicted the roughness.

| model type to predict the Al roughness        | Al roughness [nm] obtained with the use of considered model |
|---|---|
| Support Vector Regression (polynomial kernel) | 0.22  |
| Support Vector Regression (radial kernel)     | 0.26  |
| Multivariate adaptive regression splines      | 0.32  |
| Thin plate spline                             | 0.71  |
| Multilayer perceptron                         | 0.11  |
| Linear model                                  | 1.68  |

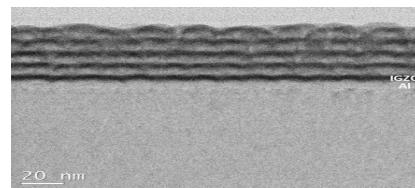
## 3 Discussion of results

Current practice of setting the Al manufacturing process parameters is based on so-called Taguchi orthogonal tables. In Fig. 1 we compare magnified images of the Al covered plates yielded by the manufacturing process whose parameter values were tuned either with Taguchi tables or by the modeling and optimization approach defined in the previous section. In the comparison we demonstrate micrographs of Al layers obtained with the Taguchi tables based approach, where the Al layer roughness was 6 nm, and with the best settings obtained by the optimization method supported with the regression models.

The ultimate application of optimized processes was demonstrated in Fig. 2. The technology of multi-layered structure composed of alternating Indium-Gallium-Zinc Oxide and Aluminum was developed. The fabricated multi-layered structure contains 5x IGZO/Al homogeneous stacks in terms of thickness.



**Fig. 1.** AFM micrograph of Al surface obtained by the fabrication process controlled by the Taguchi tables (a) and by the parameters optimized with the regression models (b).



**Fig. 2.** Cross-section of a multi-layer structure composed of alternating IGZO and Al layers, each with a thickness of 5 nm.

## References

1. D. Iwanicki, E. Warchulski, M. Ożga, M. Świniarski, A. Gertych, I. Pasternak, M. Zdrojek, M. Godlewski, J. Arabas, R. Mroczynski: Optimization of ultra-thin magnetron sputtered aluminum films, in: 3th Conference „Electron Technology” ELTE, Wrocław, Poland, (2019)
2. G. Z. Yin, “Orthogonal design for process optimization and its application in plasma etching”, Intel Corp., (1986).
3. J. Arabas and D. Jagodziński: Towards a Matrix-free Covariance Matrix Adaptation Evolution Strategy, *IEEE Transactions on Evolutionary Computation*. doi: 10.1109/TEVC.2019.2907266 (2019)

# Generalized Self-Adapting Particle Swarm Optimization algorithm with model-based optimization enhancements

Mateusz Zaborski<sup>1</sup>, Michał Okulewicz<sup>1</sup>, and Jacek Mańdziuk<sup>1</sup>

Warsaw University of Technology  
Faculty of Mathematics and Information Science  
`{M.Zaborski,M.Okulewicz,J.Mandziuk}@mini.pw.edu.pl`

**Abstract.** This paper investigates the performance of an improved version of Generalized Self-Adapting Particle Swarm Optimization (GAPSO) – a hybrid global optimization framework. In particular, the possibility of utilizing model-based optimization in parallel with sampling-based methods (like PSO or DE) within GAPSO framework is discussed. The research on GAPSO approach is based on two assumptions: (1) it is possible to improve the performance of an optimization algorithm through utilization of more function samples than standard PSO sample-based memory, (2) combining specialized sampling methods (i.e. Particle Swarm Optimization, Differential Evolution, locally fitted functions) will result in a better algorithm performance than using each of them separately. The inclusion of these model-based enhancements indicated the necessity of extending GAPSO framework by means of an external samples memory - this enhanced model is referred to as M-GAPSO in the paper. The key feature of M-GAPSO is collection of already computed function samples in R-Tree based index and their subsequent use in model-based enhancements. Moreover, M-GAPSO incorporates a global swarm restart mechanism from JADE algorithm, instead of resetting each particle separately, as opposed to GAPSO. COCO benchmark set is used to assess the M-GAPSO performance against the original GAPSO and the state-of-the-art KL-BIPOP-CMAES algorithm.

**Keywords:** Particle Swarm Optimization · global optimization · metaheuristics

## 1 Introduction

Particle Swarm Optimization (PSO) [2] is a well-known global optimization metaheuristic with many possible variants. For instance, Nepomuceno and Engelbrecht [3] proved that appropriate mix of heterogeneous versions of PSO can lead to significant performance improvement. Yamaguchi and Akimoto [6] presented the usage of search history for more efficient algorithm initialization after restart. The above works confirmed that various optimization enhancements and storing samples in memory are all promising directions of global optimization

research. This work presents an approach which combines both of the above-mentioned features.

## 2 M-GAPSO framework description

This section describes the proposed Generalized Self-Adapting Particle Swarm Optimization framework with external samples memory (M-GAPSO), which is an enhancement of the GAPSO approach [5].

The GAPSO optimization framework has been designed on the basis of PSO. It allows the usage of virtually any optimization algorithm behavior, whose performance is evaluated during the optimization process. The enhancements that brought the highest improvement to the estimated optimum and used relatively more frequently.

Within GAPSO, *particles* act independently and may behave differently. From the swarm's point of view internal behavior of the *particle* (i.e. function sampling scheme) is irrelevant. Each *particle* is only obliged to update its velocity and maintain its current and best positions. Therefore, the well-known algorithms, such as Differential Evolution (DE), can be easily implemented within GAPSO framework by means of the appropriate scheme for updating the velocity vector.

However, in order to include efficient model-based enhancements, an additional external memory module has to be implemented for storing the already sampled function values. Moreover, the implementation of the new features revealed that a global JADE-like restart mechanism [4] is more beneficial for the algorithm's performance, than the original GAPSO's particle-by-particle method.

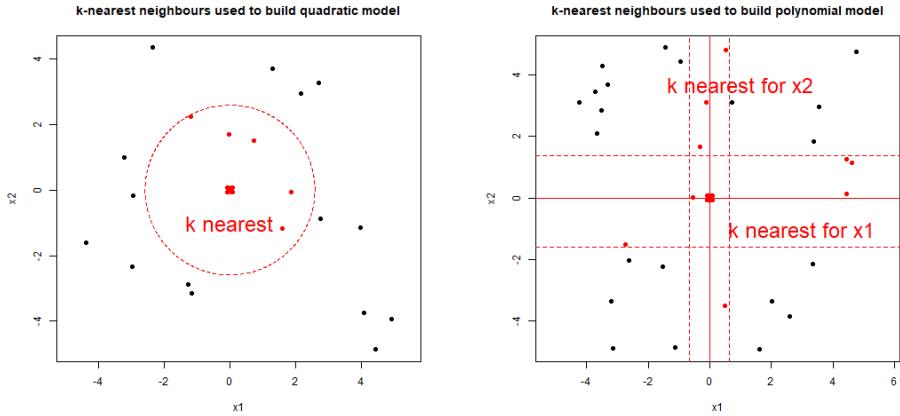
M-GAPSO is maintained in the publicly available source code repository<sup>1</sup>.

### 2.1 External memory

Gathering samples (understood as point coordinates and function values at these points) is a key enhancement compared to the initial work on GAPSO [5]. The main idea is to take advantage of already gathered samples and use them in model-based optimization enhancements. In order to store and retrieve samples in an efficient way M-GAPSO utilizes a multidimensional R-Tree index. It allows quick access to the desired subset of samples, such as surrounding of a selected point.

### 2.2 Model-based optimization enhancements

Model-based enhancements (quadratic as well as polynomial models) have been applied in order to support quick convergence to local optima. In both cases the same principles of particle's behavior are applied. At the beginning, the model is fitted using specified sample collection. Then, the algorithm finds a function



**Fig. 1.** Comparison of a samples data sets used for fitting quadratic and polynomial models

optimum in relation with the field boundaries. Finally, the particle is moved to coordinates that match the estimated optimum.

**Quadratic model** is fitted on a data set composed of  $k$  nearest samples (in the sense of the Euclidean metric) to a  $particle.x_{best}$  location for which the quadratic behavior has been selected. See Fig. 1 as an example. Quadratic function-based approach fits the following model:

$$\hat{f}_{quadratic.local}(x) = \sum_{d=1}^{dim} (a_d x_d^2 + b_d x_d) + c \quad (1)$$

**Polynomial model** enhances the quadratic model in the following way:

$$\hat{f}_{polynomial.local}(x_d) = \sum_{i=1}^p a_{i,d} x_d^i + c \quad (2)$$

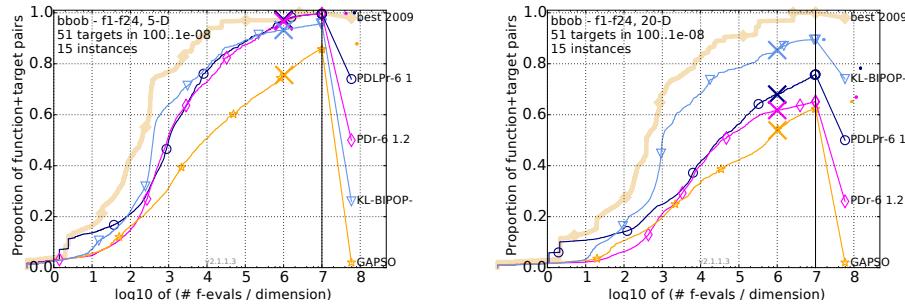
Furthermore, the polynomial model is fitted on separated data sets in each dimension. These data sets are made of  $k$  samples closest to a line with coordinates fixed to the current location, except for the dimension  $d$  for which the model is currently fitted. The differences between methods of gathering samples (quadratic vs. polynomial) are depicted in Fig. 1.

### 3 Results and future work

The enhancements implemented in M-GAPSO improved performance over GAPSO. Moreover, results became comparable with the state-of-the-art CMA-ES[6], mostly for lower dimensions. Comparisons for 5 and 20 dimensions are shown in Fig. 2.

<sup>1</sup> <https://bitbucket.org/pl-edu-pw-mini-optimization/basic-psو-de-hybrid>

Evaluation was made on 24 noiseless continuous functions from COCO BBOB benchmark data set [1].



**Fig. 2.** Results of M-GAPSO configurations including model-based optimization (PDLPr) and without model based optimization (PDr) against GAPSO and state-of-the-art variation of CMA-ES for  $DIM * 10^6$  optimization budget.

On a general note, M-GAPSO results are promising, although, many improvements can still be applied, in particular other local methods for handling samples gathered in external memory, as well as global modeling schemes.

## References

1. Hansen, N., Brockhoff, D., Mersmann, O., Tusař, T., Tusař, D., ElHara, O.A., Sampaio, P.R., Atamna, A., Varelas, K., Batu, U., Nguyen, D.M., Matzner, F., Auger, A.: COmparing Continuous Optimizers: numbbbo/COCO on Github (2019). <https://doi.org/10.5281/zenodo.2594848>, <https://doi.org/10.5281/zenodo.2594848>
2. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. Proceedings of IEEE International Conference on Neural Networks. IV pp. 1942–1948 (1995)
3. Nepomuceno, F.V., Engelbrecht, A.P.: A Self-adaptive Heterogeneous PSO Inspired by Ants. In: International Conference on Swarm Intelligence, pp. 188–195 (2012). [https://doi.org/10.1007/978-3-642-32650-9\\_17](https://doi.org/10.1007/978-3-642-32650-9_17)
4. Poafă, P., Klema, V.: JADE, an adaptive differential evolution algorithm, benchmarked on the BBOB noiseless testbed. In: Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference companion - GECCO Companion '12. p. 197. ACM Press, New York, New York, USA (2012). <https://doi.org/10.1145/2330784.2330814>
5. Uliński, M., Żychowski, A., Okulewicz, M., Zaborski, M., Kordulewski, H.: Generalized Self-adapting Particle Swarm Optimization Algorithm. In: Lecture Notes in Computer Science, vol. 3242, pp. 29–40. Springer, Cham (2018). [https://doi.org/10.1007/978-3-319-99253-2\\_3](https://doi.org/10.1007/978-3-319-99253-2_3)
6. Yamaguchi, T., Akimoto, Y.: Benchmarking the novel CMA-ES restart strategy using the search history on the BBOB noiseless testbed. In: GECCO '17 Proceedings of the Genetic and Evolutionary Computation Conference Companion. pp. 1780–1787 (2017). <https://doi.org/10.1145/3067695.3084203>

# Exploring Constraint Programming. Approaching a Practical Optimization Problem<sup>\*</sup>

Weronika T. Adrian<sup>1[0000–0002–1860–6989]</sup>, Mateusz Ślażyński<sup>1[0000–0001–7269–8215]</sup>, Antoni Ligęza<sup>1[0000–0002–6573–4246]</sup>, Marco Manna<sup>2[0000–0003–3323–9328]</sup>, Nicola Leone<sup>2[0000–0002–9742–1252]</sup>, Marek Adrian<sup>1[0000–0002–0435–0994]</sup>, Krystian Jobczyk<sup>1[0000–0001–6194–2737]</sup>, Krzysztof Kluza<sup>1[0000–0003–1876–9603]</sup>, Bernadetta Stachura-Terlecka<sup>1[0000–0003–2887–5936]</sup>, and Piotr Wiśniewski<sup>1[0000–0003–3777–642X]</sup>

<sup>1</sup> AGH University of Science and Technology,  
al. A.Mickiewicza 30, 30-059 Krakow, Poland,  
`{wta,mslaz,ligeza,madrian,jobczyk,kluza,bstachur,wpiotr}@agh.edu.pl`

<sup>2</sup> University of Calabria, via Pietro Bucci  
Arcavacata di Rende 87036 (CS), Italy,  
`{manna,leone}@mat.unical.it`

**Abstract.** This paper outlines some lines of exploration of a production scheduling problem with constraints. The problem is defined and three approaches based on ASP, Prolog and Simulated Annealing implemented in Julia are mentioned. A summary of experimental results is provided.

**Keywords:** discrete optimization, constraint programming, declarative programming, answer set programming, simulated annealing

## 1 Introduction

Discrete Constraint Programming and Discrete Constraint Optimization are inspiring domains for investigation and areas of important practical applications. Over recent decades wide and in-depth theoretical studies have been carried out, and a number of techniques and tools have been developed [3, 4]. A state-of-the-art is summarized in [7]. Unfortunately, the tackled problems are not only computationally intractable; they are also diversified w.r.t. their *internal structure*, *types of constraints*, and *formalization possibility*. Some specialized methods working well for small problems e.g. [6] are not easy to translate to large ones.

In this paper we re-explore selected issues using a particular example of a discrete optimization problem<sup>3</sup> [1]. The problem has a small, training case of the size 15 variables, and a hard, realistic instance of the size 150 variables; it employs complex constraints and nontrivial goal function.

---

<sup>\*</sup> This paper is supported by AGH UST grant.

<sup>3</sup> The problem was defined by Alessio Bonetti, Michele Lombardi and Pierre Schaus and presented as a challenge at the 2014 Constraint Programming Summer School in Bologna, Italy.

## 2 Problem Description

Consider a Production Scheduling Optimization Problem. A set of items is to be produced on a single machine and delivered on time, following the rules:

- **Items Delivery** — there is a number  $m$  of different types of items to be produced and delivered at predefined date;
- **Due-Date Constraints** — an item can be delivered exactly at or before its due-date, but not after it (hard constraints);
- **Inventory Cost** — there is a penalty for storing an item produced before the due-date for each day (the inventory cost; soft constraint);
- **Single Machine – Transition Cost** — there is a single machine available, and there is a matrix  $m \times m$  defining the transition cost;
- **An optimal solution is the one having minimal cost (total inventory costs + total transition costs).**

Below we show an exemplary instance of a size 15 (time slots). In the input data, the number of periods (15) and the number of item types (8) are given. There is one line of length 15 (number of periods) for each item type with a 1 if an order must be delivered at this particular date. In the second column the inventory cost is given (10). Then the change-over cost matrix (8x8) is specified. For instance a change from type 2 to type 3 costs 193.

|                                     |                              |
|-------------------------------------|------------------------------|
| 15                                  |                              |
| 8                                   |                              |
| 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0     | 10                           |
| 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0     | 0 78 86 93 120 12 155 20     |
| 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0     | 165 0 193 213 178 12 90 20   |
| 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0     | 214 170 0 190 185 12 40 20   |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0   | 178 177 185 0 196 12 155 66  |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 | 201 199 215 190 0 12 155 20  |
| 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1   | 201 100 88 190 14 0 75 70    |
| 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0   | 50 44 155 190 111 12 0 20    |
| 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0     | 201 199 215 190 123 70 155 0 |

## 3 Declarative Encoding using Answer Set Programming

The first approach consists in solving the problem using Answer Set Programming – a powerful knowledge representation and reasoning paradigm. The syntax of ASP follows the logic programming style, and the stable models of the program constitute the solutions of the original problem. The inference uses SAT-based approach for calculating minimal stable models. The problem was coded with the *Guess-Check-Optimize* (GCO) methodology:

- *Guess*: Definition of the search space with regular rules;
- *Check*: Checking the admissibility of solutions with hard constraints, and
- *Optimize*: Optimization with two optimization criteria – weak constraints: minimization of the sum of inventory costs, and minimization of the sum of change-over costs.

In the experiments the WASP [2] solver was used.

## 4 Prolog Encoding

Another experiment was carried out with SWI-PROLOG<sup>4</sup> applied as a tool for solving the optimization problem. Both instances are taken into consideration, a training one of the size 15 and the hard instance of the size 150. The strategy lines can be summarized as follows:

- Find an initial admissible solution; then use the Prolog machine;
- *Search* - standard backtracking, depth-first search was applied;
- *Improve* - a better solution replaces the stored one;
- *Constraints* - the domain is maintained; only admissible solutions are searched for; systematic constraint propagation was carried out.

## 5 Simulated Annealing

Simulated Annealing algorithm, presented by Kirkpatrick in [5], was implemented in Julia, a high-level programming language for numerical and technical computing<sup>5</sup>. Due to the efficiency reasons, the state was encoded in a dual form:

1. as an array of ordered items where the value in the array represents a time slot, when the item is going to be produced;
2. as an array of time slots where the value in the array represents the item produced in this slot (0 if there is no item produced).

Some principles of the experiment are as follows:

- initial solution was found with greedy search;
- search was enriched with additional meta-heuristics to escape local minima;
- the core consisted of three phases: (i) reheat the annealing process, (ii) make a random series of moves, and (iii) restart search from the best state. Note that the search can be easily parallelized.

## 6 Summary and Outlook

Some numerical results are best summarized in the following tables.

| EASY INSTANCE (15 SLOTS) |                   |                  |                          |
|--------------------------|-------------------|------------------|--------------------------|
| Solver                   | first solution    | optimal solution | time to prove optimality |
| <b>ASP</b>               | 1437 in 0.1 s     | 784 in 7.85 s    | 1h 10min                 |
| <b>Prolog</b>            | 1287 in 0.001 s   | 784 in 274.639 s | 762.020 s                |
| <b>Julia SA</b>          | 1675 in 0.00001 s | 784 in 0.0449 s  | not possible             |

| HARD INSTANCE (150 SLOTS) |                  |                               |
|---------------------------|------------------|-------------------------------|
| Solver                    | first solution   | best solution (6h time limit) |
| <b>ASP<sup>6</sup></b>    | 45765 in 6.24 s  | 42050                         |
| <b>Prolog</b>             | 34808 in 0.003 s | 34705                         |
| <b>Julia SA</b>           | 56042 in 0.1 s   | 25914                         |

<sup>4</sup> <http://www.swi-prolog.org/>

<sup>5</sup> <http://julialang.org/>

- ASP encoding best mimics the formal definition of the problem and ensures that the reached solutions are correct. However, because the reasoning first generates *all* possible models and then uses different algorithms to find a minimum, the process may become really slow for big instances.
- The backtracking search — and its implementation in PROLOG — has two advantages: (i) memory consumption is very low, and (ii) there is no repetition of the search. On the other hand, operations on lists surely slow down the search in a significant way. Further, it is hard to employ heuristic search component using the standard backtracking strategy.
- The Simulated Annealing algorithm implemented in Julia programming language, being an stochastic method, easily copes with the large combinatorial search spaces and finds a decent solution within a short period of time. Despite its excellent results, it has an inherent disadvantage — it is not able to tell whether the found solution is globally optimal.

For future work, we plan to further improve the logic-based encodings, and also work on a hybrid representation for such problems that would enable to use the advantages of both logic-based approaches and stochastic methods. New tools (MiniZinc, Picat) are also intended to be explored.

## References

1. Adrian, W.T., Leone, N., Ligeza, A., Manna, M., Ślażyński, M.: Constraint optimization production planning problem. a note on theory, selected approaches and computational experiments. In: et al., L.R. (ed.) Artificial Intelligence and Soft Computing: 14th International Conference, ICAISC 2015. pp. 541–553. Lecture Notes in Computer Science ; ISSN 0302-9743. Lecture Notes in Artificial Intelligence ; LNAI 9120, Springer, Warsaw; Los Alamitos (2015)
2. Alviano, M., Dodaro, C., Leone, N., Ricca, F.: Advances in wasp. In: International Conference on Logic Programming and Nonmonotonic Reasoning. pp. 40–54. Springer (2015)
3. Dechter, R.: Constraint Processing. Morgan Kaufmann Publishers, San Francisco, CA (2003)
4. Hentenryck, P.V., Michel, L.: Constraint-Based Local Search. MIT Press, Cambridge, Massachusetts; London, England (2005)
5. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. SCIENCE **220**(4598), 671–680 (1983)
6. Ligeza, A.: Improving efficiency in constraint logic programming through constraint modeling with rules and hypergraphs. In: Ganzha, M., Maciaszek, L., Paprzycki, M. (eds.) Proceedings of the Federated Conference on Computer Science and Information Systems 2012. pp. 101–107. Polskie Towarzystwo Informatyczne, IEEE Computer Society Press, Warsaw; Los Alamitos (2012)
7. Rossi, F., van Beek, P., Walsh, T. (eds.): Handbook of Constraint Programming. Elsevier (2006)

# Employing supervised learning algorithms in the task of dynamic spectrally-spatially flexible optical networks optimization

Paweł Ksieniewicz<sup>1</sup>, Mirosław Klinkowski<sup>2</sup> and Krzysztof Walkowiak<sup>1</sup>

<sup>1</sup> National Institute of Telecommunications, Warsaw, Poland

<sup>2</sup> Wrocław University of Science and Technology, Wrocław, Poland

**Abstract.** Following work focuses on optimization of dynamic spectrally-spatially flexible optical networks aided using supervised learning methods. Such kind of networks have distance-adaptive, spectral super-channel transmission realized over weakly-coupled multi-core fibers. Article proposes employing a pattern recognition approach with the goal to estimate the effective TRs of particular MFs so that to optimize network performance in terms of bandwidth blocking probability (BBP). The experimental study confirmed the high efficiency of the proposed algorithm and its superiority to the reference models.

**Keywords:** pattern recognition · optimization · regression · optimization · space division multiplexing

## 1 Introduction

The growing popularity of *artificial intelligence*, resulting mainly from the wide interest in learning using deep models [7], has significantly contributed to the extension of the area of interest in *pattern recognition* algorithms to other spaces, including optimization of computer networks [10]. Therefore, the concept of *cognitive optical networks* appeared, which may be described as networks that are somewhat *aware* of their current state and dynamically adapting itself to the rules prevailing within it [4].

*Machine learning* techniques try to be the main tool of this *awareness*. An important issue, however, is the fact that the overall automation of a computer networks is a very complex task, whose simplification to the form of patterns, the basic teaching elements of artificial intelligence algorithms [2], is definitely non-trivial and requires a deep understanding of both the operation of computer networks and the pattern recognition algorithms themselves [3].

Pattern recognition systems can, in principle, be divided into two groups [8]. In the first, unsupervised learning, we feed the algorithm with a set of structured, but non-described events, the second group, being the focus of the following work, are supervised learning algorithms, in which there is also a set of learning objects, but each of them is described by a continuous value (regression task) or assignment to one of the predefined categories (classification task).

*Spectrally-spatially flexible optical networks* (ss-FON-s) allow to combine the space division multiplexing (SDM) technology [9], enabling parallel transmission

of co-propagating spatial modes in suitably designed optical fibers, such as MCFs [1], with flexible-grid elastic optical network (EON) technologies that enable *SCh* (i.e., multi-carrier) and distance-adaptive transmission [5].

The following work attempts to describe the process of optimizing modulation format selection ranges aided by an ensemble of regressors, reducing the number of necessary, complex simulations and allowing for the possibly reliable prediction of metrics describing its effectiveness and, as a consequence, the precise location of a near optimal solution.

## 2 Network model

Following work realizes experiments on ss-FON composed of weakly-coupled MCF links and signals transmitted in the form of spectral *Schs*. Network itself applies *flexgrid* equipped with coherent transceivers, each with re-configurable bit-rates and MF-s compatible with optical link. The network nodes do not perform the switching of cores; i.e., the same core is assigned to a lightpath in all MCF links belonging to its routing path. The above assumptions are frequently considered in the literature [6].

We assume that a set of candidate routing paths is given for each node pair. Additionally, each MCF core on each path an MF from a set of available MF-s is pre-selected. The selected MFs are used to determine the number of frequency slices required to carry the bit-rate demand of a connection request on a given path-core pair. Moreover, the selected MFs are necessary in the *QoT* estimation according to Model. For each path-core pair, we select the most spectrally efficient MF subject to its TR in a given MCF core exceeding the path length.

## 3 Regression-aided optimization

Overall optimization is performed over tuple of *transmission reaches* (TR) for MF-s: *QPSK*, *8QAM*, and *16QAM*, respectively, which determines the selection of MFs. The transmission reach of *BPSK* is not included, since we assume that it have the maximum feasible value. General approach is:

1. A pool of random TR samples (1000 in performed experiments) are generated with the requirement to satisfy the conditions of modulation.
2. For each sample an XT-aware dynamic routing algorithm is run to obtain BBP resulting from using MF-s selected according to it.
3. Interpreting inputs of dynamic routing algorithm (generated, random TR-s) as a set of *patterns* and its output (BBP) as collection of *target values* leads to creation of dataset dedicated to create the regression model. Trained model is able to predict BBP for new, yet unknown patterns, so we perform a simple grid search to find the best TR, giving the lowest prediction of BBP.
4. At the last step, XT-aware dynamic routing algorithm calculates BBP for best TR found for a trained model.

The main task of the applied pattern recognition approach is to train a model that is able to predict the expected value, based on the regression algorithm. The input data to the model are the results of the initial simulations, where the set of input parameters is considered as the analyzed feature space while the results

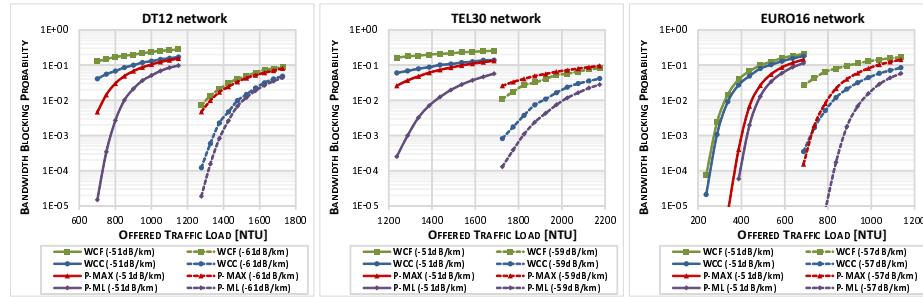
of carried out simulations is collection of labels. The approach used to develop a model of the highest possible quality is the construction of a homogeneous ensemble of estimators, diversified by dividing the training set into  $k$  disjoint subsets. Following approach allows to minimize the influence of uncontrollable processing factors, such as outliers, on the prediction result of the model. Each of the regressors in the pool is trained, according to the 5-fold cross-validation rule. The measure of the partial quality of each model, is later used as a weight employed in the decision *fuser*, implemented as a weighted average of the objective function prediction. Overall proposed optimization approach is later described as the P-ML algorithm.

#### 4 Experimental evaluation

To make a proper choice of base regression algorithm, a preliminary experiment was carried out, examining the value of the  $r^2$  metric for the model built in accordance with the above proposal, depending on the used regressor. Standard approaches were included in the group of tested methods. The summary of the experiment may be found in Table 1. As may be seen, the kNN algorithm is better than the others in the strong majority of cases and there is no situation in which any of the competitors achieves better statistical results.

**Table 1.** Comparative analysis of base regressors used in preliminary experiment. Value in the cell indicates the number of datasets in which the method from the column statistically significantly improves the method from the row.

|     | kNN | LIN | LAS | SVR |
|-----|-----|-----|-----|-----|
| kNN | 0   | 0   | 0   | 0   |
| LIN | 40  | 0   | 0   | 0   |
| LAS | 42  | 35  | 0   | 0   |
| SVR | 42  | 41  | 41  | 0   |



**Fig. 1.** BBP performance of P-ML and reference methods as a function of offered traffic load for different values of XT in analyzed networks.

In Figure 1 we show the BBP performance of P-ML and the reference methods in the analyzed networks, assuming different XT scenarios. The right-side results of BBP in particular networks are obtained with different values of XT. In general, P-ML significantly outperforms the reference methods in the studied networks.

The second best method is either WCC or P-MAX, depending on the network and XT scenario. These trends follow mainly from the differences in sizes of analyzed networks and the TR values applied in particular methods.

## 5 Conclusion

Following article tackled the problem of dynamic and crosstalk-aware routing in SS-FONS with multi-core fibers. The key idea of our method was to utilize data from initial XT-aware dynamic simulations run with randomly selected transmission reaches of modulation formats as an input for the regression model. The regression approach allows to find the best configuration of transmission reaches in terms of the estimated blocking. Finally, these transmission reaches are used in XT-aware dynamic routing algorithm to obtain the final results. Experiments run on three representative network topologies with different XT levels have shown that the P-ML method proposed significantly outperforms three reference methods previously proposed in the literature for the considered problem.

## Acknowledgements

The work of P. Ksieniewicz and K. Walkowiak was supported by National Science Centre, Poland under Grant No. 2017/27/B/ST7/00888. The work of M. Klinkowski was supported by the National Science Centre, Poland under Grant No. 2016/21/B/ST7/02212.

## References

1. Awaji, Y., Sakaguchi, J., Puttnam, B.J., Luis, R.S., Mendinueta, J.M.D., Klaus, W., Wada, N.: High-capacity transmission over multi-core fibers. *Opt. Fiber Technol.* **35**, 100–107 (2017)
2. Bishop, C.M.: *Pattern Recognition and Machine Learning*. Springer (2006)
3. Chan, V.W.S., Jang, E.: Cognitive all-optical fiber network architecture. In: Proc. of ICTON. pp. 1–4 (Jul 2017). <https://doi.org/10.1109/ICTON.2017.8025063>
4. de Miguel, I., Durán, R.J., Jiménez, T., Fernández, N., Aguado, J.C., Lorenzo, R.M., Caballero, A., Monroy, I.T., Ye, Y., Tymecki, A., Tomkos, I., Angelou, M., Klonidis, D., Francescon, A., Siracusa, D., Salvadori, E.: Cognitive dynamic optical networks [invited]. *IEEE/OSA J. of Opt. Commun. and Netw.* **5**(10), A107–A118 (Oct 2013). <https://doi.org/10.1364/JOCN.5.00A107>
5. Goścień, R., Walkowiak, K., Klinkowski, M.: Distance-adaptive transmission in cloud-ready elastic optical networks. *IEEE/OSA J. of Opt. Commun. and Netw.* **6**(10), 816–828 (2014)
6. Klinkowski, M., Lechowicz, P., Walkowiak, K.: Survey of resource allocation schemes and algorithms in spectrally-spatially flexible optical networking. *Opt. Switch. and Netw.* **27**, 58–78 (2018)
7. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *nature* **521**(7553), 436 (2015)
8. Mitchell, T., Buchanan, B., DeJong, G., Dietterich, T., Rosenbloom, P., Waibel, A.: Machine learning. *Ann. Rev. of Comput. Sci.* **4**(1), 417–433 (1990)
9. Saridis, G.M., Alexandropoulos, D., Zervas, G., Simeonidou, D.: Survey and evaluation of space division multiplexing: From technologies to optical networks. *IEEE Commun. Surv. and Tutorials* **17**(4), 2136–2156 (2015)
10. Wei, W., Wang, C., Yu, J.: Cognitive optical networks: Key drivers, enabling techniques, and adaptive bandwidth services. *IEEE Comm. Mag.* **50**(1), 106–113 (January 2012). <https://doi.org/10.1109/MCOM.2012.6122540>

## Author Index

### A

- Adrian, M. 243, 384  
Adrian, W. T. 243, 247, 384  
Amendola, G. 247  
Arabas, J. 372, 376  
Augustyniak, L. 107, 285

### B

- Baczkiewicz, A. 59  
Bednarek, J. 360  
Bednarek, M. 173  
Bejtka, M. 255  
Belter, D. 179  
Biedrzycki, J. 29  
Bielak, P. 33  
Bieronski, M. 33  
Blachnik, M. 302  
Borczyk, W. 2  
Brodzicki, A. 133

### C

- Chintaluri, C. 255  
Chmielewski, L. J. 150  
Cholewa, M. 121  
Choras, M. 15, 310  
Cieslewicz, A. 223  
Cpalka, K. 352, 364  
Culer, L. 55  
Cwian, K. 161  
Cwiklinski, B. 15  
Czarnecki, W. 25  
Czarnowski, I. 103

### D

- Dabrowski, A. 117  
Dembczynski, K. 73, 332  
Domino, K. 121  
Duda, P. 296  
Dudek, G. 269  
Dutkiewicz, J. 223  
Dzik, J. M. 255

### F

- Falkiewicz, M. 41  
Filipowicz, W. 6  
Forczmanski, P. 84  
Franus, W. 232  
Fularz, M. 177, 178

### G

- Gabor, M. 319  
Gaciarz, M. 92  
Gambin, T. 10  
Gawdzik, G. 150  
Gawron, P. 11, 19, 25  
Gawron, T. 165  
Gielczyk, A. 15  
Gladki, P. 2  
Glomb, P. 121, 128  
Gniewkowski, M. 202  
Gorecki, T. 68  
Gorgon, M. 133  
Grabowski, B. 128  
Guzy, F. 99

### H

- Hernes, M. 238  
Hullermeier, E. 73

### J

- Jagodzinski, D. 372  
Janz, A. 327  
Jasinska, K. 332  
Jaworek-Korjakowska, J. 133  
Jaworski, M. 299  
Jaworski, W. 206  
Jedrzejek, C. 223  
Jedrzejewska-Szmek, J. 259  
Jobczyk, K. 243, 384  
Jozwiak, M. 33

### K

- Kaczmarek, A. 55

- Kajdanowicz, T. 107  
Kania, K. 37  
Kazienko, P. 107  
Kicki, P. 165  
Kitlas-Golinska, K. 315  
Klec, M. 306  
Klikowski, J. 53  
Klinkowski, M. 388  
Klopotek, M. A. 273, 277, 281  
Klopotek, R. A. 273, 277, 281  
Kluza, K. 243, 384  
Knap, O. 137  
Kobylinski, L. 210  
Kocon, J. 327  
Kordos, M. 302  
Kosturek, M. 37  
Kotowski, K. 49  
Kowalek, P. 88  
Kozik, R. 310  
Kozlowski, M. 184  
Kozlowski, N. 336  
Kraft, M. 177, 178  
Krawiec, A. 11  
Krawiec, K. 360  
Krol, D. 253  
Krupinski, R. 141  
Ksieniewicz, P. 388  
Kucharski, D. 133  
Kusmirek, W. 10
- L**
- Lango, M. 292  
Lapa, K. 364  
Lawrynowiczi, A. 242  
Lech, P. 141  
Leone, N. 384  
Lepak, L. 368  
Lesinski, W. 315, 323  
Ligeza, A. 243, 384  
Loch-Olszewska, H. 88
- M**
- Mandziuk, J. 380  
Manna, M. 247, 384  
Marasek, M. 306  
Marcinczuk, M. 219  
Marciniak, E. 117  
Marciniak, T. 117
- Mazurek, P. 137  
Michalak, H. 141  
Michalak, M. 265  
Michalek, M. M. 165  
Mieczynska, M. 103  
Mizgajski, J. 285  
Mnich, K. 311, 315, 323  
Mortier, T. 73  
Morzy, M. 285  
Mroczynski, R. 376  
Mułek, P. Z. 78
- N**
- Najgebauer, P. 156  
Neumann, L. 10  
Nowak, R. 10, 189, 232  
Nowak, T. 161  
Nowicki, M. 161
- O**
- Okarma, K. 141  
Oklesinski, D. 215  
Okulewicz, M. 380  
Orłowski, A. 150  
Ostaszewski, M. 128  
Oszutowska-Mazurek, D. 137
- P**
- Palczewski, K. 63  
Pawlak, M. 310  
Pelka, P. 21  
Penaloza, R. 247  
Piasecki, P. 68  
Piaskowski, K. 360  
Pieczyński, D. 177, 178  
Piekarski, M. 133  
Piliszek, R. 311, 323  
Piwowarczyk, M. 112  
Pliszka, Z. 356  
Polewko-Klim, A. 323  
Potoniec, J. 234
- R**
- Radzikowski, K. 189  
Romaszewski, M. 121, 128  
Rostanski, M. 2  
Rostkowska, M. 169

Rudnicki, W. 311, 315, 323  
Rutkowski, L. 296, 299

**S**

Sapinski, B. 323  
Scherer, R. 156  
Seredynski, D. 146  
Skrzypczynski, P. 161  
Skulimowski, A. M. J. 346  
Slawinski, Z. T. 259  
Slazynski, M. 384  
Smolinski, A. 84  
Sobkowicz, A. 184  
Stachura-Terlecka, B. 243, 384  
Stankiewicz, A. 117  
Stapor, K. 49  
Stefanczyk, M. 146  
Stefanowski, J. 292  
Stopa, M. 117  
Szklanny, K. 306  
Szostak, D. 342  
Szwabinski, J. 88  
Szymanski, P. 285  
Szymczak, A. 285

**T**

Timofiejczuk, A. 251  
Tomczyk, A. 145  
Topolski, M. 92  
Trawinski, B. 78, 112  
Tuora, R. 210

**U**

Unold, O. 55, 319, 336, 356  
Urbaniaik, K. 45

**W**

Waegeman, T. 73  
Walas, K. 173  
Walkowiak, K. 342, 388  
Walkowiak, T. 202  
Warchulski, E. 376  
Wawrzynski, P. 368  
Wegierek, M. 146  
Wierzchon, S. T. 277, 281  
Wisniewski, D. 242  
Wisniewski, P. 243, 384  
Wojciechowski, S. 77  
Wojcik, D. 255, 259  
Wolk, K. 198  
Wydmuch, M. 73

**Y**

Yoshie, O. 189

**Z**

Zaborski, M. 380  
Zalasinski, M. 352  
Zawadzka-Gosk, E. 198  
Zawistowski, P. 368  
Zelasko, P. 285  
Zyblewski, P. 54