

Andreas Holzinger · Peter Kieseberg  
A Min Tjoa · Edgar Weippl (Eds.)

LNCS 11015

# Machine Learning and Knowledge Extraction

Second IFIP TC 5, TC 8/WG 8.4, 8.9, TC 12/WG 12.9  
International Cross-Domain Conference, CD-MAKE 2018  
Hamburg, Germany, August 27–30, 2018, Proceedings



ifip

 Springer

*Commenced Publication in 1973*

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

## Editorial Board

David Hutchison

*Lancaster University, Lancaster, UK*

Takeo Kanade

*Carnegie Mellon University, Pittsburgh, PA, USA*

Josef Kittler

*University of Surrey, Guildford, UK*

Jon M. Kleinberg

*Cornell University, Ithaca, NY, USA*

Friedemann Mattern

*ETH Zurich, Zurich, Switzerland*

John C. Mitchell

*Stanford University, Stanford, CA, USA*

Moni Naor

*Weizmann Institute of Science, Rehovot, Israel*

C. Pandu Rangan

*Indian Institute of Technology Madras, Chennai, India*

Bernhard Steffen

*TU Dortmund University, Dortmund, Germany*

Demetri Terzopoulos

*University of California, Los Angeles, CA, USA*

Doug Tygar

*University of California, Berkeley, CA, USA*

Gerhard Weikum

*Max Planck Institute for Informatics, Saarbrücken, Germany*


More information about this series at <http://www.springer.com/series/7409>

Andreas Holzinger · Peter Kieseberg  
A Min Tjoa · Edgar Weippl (Eds.)

# Machine Learning and Knowledge Extraction

Second IFIP TC 5, TC 8/WG 8.4, 8.9, TC 12/WG 12.9  
International Cross-Domain Conference, CD-MAKE 2018  
Hamburg, Germany, August 27–30, 2018  
Proceedings

*Editors*

Andreas Holzinger   
Graz University of Technology  
Medical University Graz  
Graz  
Austria

Peter Kieseberg  
SBA Research  
Vienna  
Austria

A Min Tjoa  
Vienna University of Technology  
Vienna  
Austria

Edgar Weippl  
SBA Research  
Vienna  
Austria

ISSN 0302-9743                      ISSN 1611-3349 (electronic)  
Lecture Notes in Computer Science  
ISBN 978-3-319-99739-1              ISBN 978-3-319-99740-7 (eBook)  
<https://doi.org/10.1007/978-3-319-99740-7>

Library of Congress Control Number: 2018952241

LNCS Sublibrary: SL3 – Information Systems and Applications, incl. Internet/Web, and HCI

© IFIP International Federation for Information Processing 2018

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

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

# Preface

The International Cross-Domain Conference for Machine Learning and Knowledge Extraction, CD-MAKE, is a joint effort of IFIP TC 5, TC 12, IFIP WG 8.4, IFIP WG 8.9 and IFIP WG 12.9 and is held in conjunction with the International Conference on Availability, Reliability and Security (ARES). The second conference was organized at the University Hamburg, Germany. A few words about IFIP:

IFIP – the International Federation for Information Processing – is the leading multinational, non-governmental, apolitical organization in information and communications technologies and computer sciences, is recognized by the United Nations (UN) and was established in the year 1960 under the auspices of UNESCO as an outcome of the first World Computer Congress held in Paris in 1959.

IFIP is incorporated in Austria by decree of the Austrian Foreign Ministry (September 20, 1996, GZ 1055.170/120-I.2/96) granting IFIP the legal status of a non-governmental international organization under the Austrian Law on the Granting of Privileges to Non-Governmental International Organizations (Federal Law Gazette 1992/174).

IFIP brings together more than 3,500 scientists without boundaries from both academia and industry, organized in more than 100 Working Groups (WGs) and 13 Technical Committees (TCs).

CD stands for “cross-domain” and means the integration and appraisal of different fields and application domains to provide an atmosphere to foster different perspectives and opinions. The conference fosters an integrative machine learning approach, taking into account the importance of data science and visualization for the algorithmic pipeline with a strong emphasis on privacy, data protection, safety, and security. It is dedicated to offering an international platform for novel ideas and a fresh look at methodologies to put crazy ideas into business for the benefit of humans. Serendipity is a desired effect and should lead to the cross-fertilization of methodologies and the transfer of algorithmic developments.

The acronym MAKE stands for “MACHINE Learning and Knowledge Extraction,” a field that, while quite old in its fundamentals, has just recently begun to thrive based on both the novel developments in the algorithmic area and the availability of big data and vast computing resources at a comparatively low price.

Machine learning studies algorithms that can learn from data to gain knowledge from experience and to generate decisions and predictions. A grand goal is to understand intelligence for the design and development of algorithms that work autonomously (ideally without a human-in-the-loop) and can improve their learning behavior over time. The challenge is to discover relevant structural and/or temporal patterns (“knowledge”) in data, which are often hidden in arbitrarily high-dimensional spaces, and thus simply not accessible to humans. Machine learning as a branch of artificial intelligence is currently undergoing a kind of Cambrian explosion and is the fastest growing field in computer science today. There are many application domains, e.g.,

smart health, smart factory (Industry 4.0), etc. with many use cases from our daily lives, e.g., recommender systems, speech recognition, autonomous driving, etc. The grand challenges lie in sense-making, in context-understanding, and in decision-making under uncertainty. Our real world is full of uncertainties and probabilistic inference had an enormous influence on artificial intelligence generally and statistical learning specifically. Inverse probability allows us to infer unknowns, to learn from data, and to make predictions to support decision-making. Whether in social networks, recommender systems, health, or Industry 4.0 applications, the increasingly complex data sets require efficient, useful, and useable solutions for knowledge discovery and knowledge extraction.

To acknowledge all those who contributed to the efforts and stimulating discussions is not possible in a preface with limited space like this one. Many people contributed to the development of this volume, either directly or indirectly, and it is impossible to list all of them here. We herewith thank all colleagues and friends for their positive and supportive encouragement. Finally, yet importantly, we thank the Springer management team and the Springer production team for their smooth support.

Thank you to all!

August 2018

Andreas Holzinger  
Peter Kieseberg  
Edgar Weippl  
A Min Tjoa

# Organization

## Organizers

Andreas Holzinger	Medical University and Graz University of Technology, Austria
Peter Kieseberg	SBA Research, Austria
Edgar Weippl (IFIP WG 8.4 Chair)	SBA Research, Austria
A Min Tjoa (IFIP WG 8.9. Chair, Honorary Secretary IFIP)	TU Vienna, Austria

## Program Committee

Rakesh Agrawal	Microsoft Search Labs, Mountain View, USA
Zeynep Akata	University of Amsterdam, The Netherlands
Jose Maria Alonso	University of Santiago de Compostela, Spain
Sophia Ananiadou	National Centre for Text Mining, Manchester Institute of Biotechnology, UK
Amin Anjomshoa	SENSEable City Laboratory, MIT, USA
Joel P. Arrais	University of Coimbra, Portugal
John A. Atkinson-Abutridy	Universidad de Concepcion, Chile
Chloe-Agathe Azencott	Mines Paris Tech, France
Alexandra Balahur	European Commission Joint Research Centre, Ispra, Italy
Jan Baumbach	Technical University of Munich, Germany
Smaranda Belciug	University of Craiova, Romania
Mounir Ben Ayed	Ecole Nationale d'Ingenieurs de Sfax, Tunisia
Mattia G. Bergomi	Champalimaud Centre for the Unknown, Portugal
Enrico Bertini	New York University, USA
Elisa Bertino	Purdue University, West Lafayette, USA
Tarek R. Besold	City, University of London, UK
Michele Bezzi	SAP Labs France, Sophia Antipolis, France
Jiang Bian	University of Florida, Gainesville, USA
Maria Bielikova	Slovak University of Technology, Slovakia
Chris Biemann	Technische Universität Darmstadt, Germany
Svetla Boytcheva	Bulgarian Academy of Sciences, Bulgaria
Malin Bradley	Vanderbilt University, Nashville, USA
Francesco Buccafurri	Università Mediterranea di Reggio Calabria, Italy
Katja Bühler	VRVis Research Center, Austria
Andre Calero-Valdez	RWTH Aachen University, Germany
Andrea Cerri	IMATI –CNR, Genoa, Italy



Mirko Cesarini	Università di Milano Bicocca, Italy
Chaomei Chen	Drexel University, USA
Elizabeth S. Chen	Brown University, USA
Veronika Cheplygina	Erasmus Medical Center, Rotterdam, The Netherlands
Ajay Chander	Stanford University and Fujitsu Labs of America, USA
Nitesh V. Chawla	University of Notre Dame, USA
Philipp Cimiano	Bielefeld University, Germany
Krzysztof J. Cios	VCU, Richmond, USA
Carlo Combi	University of Verona, Italy
Tim Conrad	Freie Universität Berlin, Germany
Anna Helena Reali Costa	University of Sao Paulo, Brazil
Pierluigi Crescenzi	Università degli Studi di Firenze, Italy
Gloria Cerasela Crisan	Vasile Alecsandri University of Bacau, Romania
Alfredo Cuzzocrea	DIA Department, University of Trieste, Italy
Matthias Dehmer	University for Health and Medical Informatics Tyrol, Austria
Alexiei Dingli	University of Malta, Malta
Pawel Dlotko	University of Pennsylvania, USA
Josep Domingo-Ferrer	Universitat Rovira i Virgili, Spain
Mike Duerr-Specht	Duke University Hospital, USA
Isao Echizen	National Institute of Informatics, Japan
Max J. Egenhofer	University of Maine, USA
Kapetanos Epaminondas	University of Westminster, London, UK
Barbara Di Fabio	Università di Bologna, Italy
Aldo Faisal	Imperial College London, UK
Aasa Feragen	University of Copenhagen, Denmark
Massimo Ferri	University of Bologna, Italy
Sara Johansson Fernstad	Northumbria University, UK
Peter Filzmoser	TU Vienna, Austria
Ana Fred	Technical University of Lisbon, Portugal
Hugo Gamboa	PLUX Wireless Biosensor, and Universidade Nova de Lisboa, Portugal
Luo Gang	University of Washington, Seattle, USA
Aryya Gangopadhyay	University of Maryland, USA
Matthieu Geist	Université de Lorraine France
Panagiotis Germanakos	University of Cyprus, Cyprus
Marie Gustafsson Friberger	Malmö University, Sweden
Randy Goebel	University of Alberta, Canada
Michael Granitzer	University of Passau, Germany
Dimitrios Gunopulos	University of Athens, Greece
Udo Hahn	University of Jena, Germany
Barbara Hammer	Bielefeld University, Germany
Siegfried Handschuh	NUI Galway, Ireland
Pim Haselager	Radboud University Nijmegen, The Netherlands
Dominik Heider	University of Marburg, Germany
Kristina Hettne	Leiden University Medical Center, The Netherlands

Rainer Hofmann-Wellenhof	Graz University Hospital, Austria
Katharina Holzinger	SBA Research gGmbH, Austria
Andreas Hotho	University of Würzburg, Germany
Jun Luke Huan	University of Kansas, USA
Barna Laszlo Iantovics	Petru Maior University, Romania
Beatriz De La Iglesia	University of East Anglia, UK
Xiaoqian Jiang	University of California San Diego, USA
Mateusz Juda	Jagiellonian University, Poland
Igor Jurisica	IBM Life Sciences Discovery Centre, and Princess Margaret Cancer Centre, Canada
Andreas Kerren	Linnaeus University, Växjö, Sweden
Peter Kieseberg	SBA Research gGmbH, Austria
Negar Kiyavash	University of Illinois at Urbana-Champaign, USA
Mei Kobayashi	NTT Communications, Japan
Natsuhiko Kumasaka	RIKEN, Japan
Claudia Landi	Università di Modena e Reggio Emilia, Italy
Robert S. Laramee	Swansea University, UK
Nada Lavrac	Jozef Stefan Institute, Slovenia
Freddy Lecue	Accenture Technology Labs, Ireland and Inria Sophia Antipolis, France
Sangkyun Lee	Dortmund University, Germany
Lenka Lhotska	Czech Technical University, Czech Republic
Ulf Leser	Humboldt-Universität zu Berlin, Germany
Chunping Li	Tsinghua University, China
Brian Y. Lim	National University of Singapore, Singapore
Luca Longo	Trinity College Dublin, Ireland
Oswaldo Ludwig	Nuance Communications, Germany
Andras Lukacs	Hungarian Academy of Sciences and Eotvos University, Hungary
Anant Madabushi	Case Western Reserve University, USA
Ljiljana Majnarić-Trtica	University of Osijek, Croatia
Donato Malerba	Università degli Studi di Bari Aldo Moro, Italy
Vincenzo Manca	University of Verona, Italy
Sjouke Mauw	University of Luxembourg, Luxembourg
Ernestina Menasalvas	Polytechnic University of Madrid, Spain
Facundo Memoli	Ohio State University, USA
Yoan Miche	Nokia Bell Labs, Finland
Silvia Miksch	TU Vienna, Austria
Marian Mrozek	Jagiellonian University, Poland
Tingting Mui	University of Manchester, UK
Nysret Musliu	TU Vienna, Austria
Daniel E. O'leary	University of Southern California, USA
Patricia Ordonez-Rozo	University of Puerto Rico Rio Piedras, Puerto Rico
Vasile Palade	Coventry University, UK
Jan Paralic	Technical University of Kosice, Slovakia
Valerio Pascucci	University of Utah, USA

Gabriella Pasi	Università di Milano Bicocca, Italy
Philip R. O. Payne	Washington University, USA
Roberto Perdisci	University of Georgia, USA
Armando J. Pinho	University of Aveiro, Portugal
Camelia M. Pintea	Technical University of Cluj-Napoca, Romania
Raul Rabadan	Columbia University, USA
Heri Ramampiaro	Norwegian University of Science and Technology, Norway
Fabrizio Riguzzi	Università di Ferrara, Italy
Giuseppe Rizzo	Istituto Superiore Mario Boella Turin, Italy
Jianhua Ruan	University of Texas, San Antonio, USA
Lior Rokach	Ben-Gurion University of the Negev, Israel
Carsten Roecker	Fraunhofer IOSB-INA and Ostwestfalen-Lippe University of Applied Sciences, Germany
Timo Ropinski	Ulm University, Germany
Richard Roettger	Syddansk University, Denmark
Giuseppe Santucci	La Sapienza, University of Rome, Italy
Pierangela Samarati	University of Milan, Italy
Christin Seifert	University of Passau, Germany
Yongtang Shi	Nankai University, China
Andrzej Skowron	University of Warsaw, Poland
Neil R. Smalheiser	University of Illinois at Chicago, USA
Axel J. Soto	Manchester Institute of Biotechnology, UK
Rainer Spang	University of Regensburg, Germany
Irena Spasic	Cardiff University, UK
Gregor Stiglic	Stanford School of Medicine, USA
Simone Stumpf	City, University of London, UK
Shiliang Sun	East China Normal University, China
Xu Sun	Peking University, China
Philipp Thomas	Humboldt-Universität zu Berlin, Germany
A Min Tjoa	TU Vienna, Austria
Dimitar Trajanov	Cyril and Methodius University, Republic of Macedonia
Olof Torgersson	Chalmers University of Technology, Sweden
Shusaku Tsumoto	Shimane University, Japan
Catagaj Turkey	City, University of London, UK
Jean Vanderdonckt	Université catholique de Louvain, Belgium
Karin Verspoor	National Information and Communications Technology Australia, Australia
Dmitry Vetrov	Higher School of Economics, Moscow, Russia
Hubert Wagner	Institute for Science and Technology, Austria
Byron Wallace	Northeastern University, USA
Daniel Weiskopf	Universität Stuttgart, Germany
Edgar Weippl	SBA Research gGmbH, Austria
Janusz Wojtusiak	George Mason University, USA
William Bl Wong	Middlesex University London, UK

Kai Xu	Middlesex University London, UK
Pinar Yildirim	Okan University, Turkey
Martina Ziefle	RWTH Aachen University, Germany
Ping Zhang	IBM T.J. Watson Research Center, USA
Ning Zhong	Maebashi Institute of Technology, Japan
Jianlong Zhou	Commonwealth Scientific and Industrial Research Organisation (CSIRO), Australia
Xuezhong Zhou	Beijing Jiaotong University, China
Blaz Zupan	University of Ljubljana, Slovenia

# Contents

Current Advances, Trends and Challenges of Machine Learning and Knowledge Extraction: From Machine Learning to Explainable AI . . . . .	1
<i>Andreas Holzinger, Peter Kieseberg, Edgar Weippl, and A Min Tjoa</i>	
<b>MAKE-Main Track</b>	
A Modified Particle Swarm Optimization Algorithm for Community Detection in Complex Networks . . . . .	11
<i>Alireza Abdollahpouri, Shadi Rahimi, Shahnaz Mohammadi Majd, and Chiman Salavati</i>	
Mouse Tracking Measures and Movement Patterns with Application for Online Surveys . . . . .	28
<i>Catia Cepeda, Joao Rodrigues, Maria Camila Dias, Diogo Oliveira, Dina Rindlisbacher, Marcus Cheetham, and Hugo Gamboa</i>	
Knowledge Compilation Techniques for Model-Based Diagnosis of Complex Active Systems . . . . .	43
<i>Gianfranco Lamperti, Marina Zanella, and Xiangfu Zhao</i>	
Recognition of Handwritten Characters Using Google Fonts and Freeman Chain Codes. . . . .	65
<i>Alexiei Dingli, Mark Bugeja, Dylan Seychell, and Simon Mercieca</i>	
An Efficient Approach for Extraction Positive and Negative Association Rules from Big Data . . . . .	79
<i>Bemarisika Parfait, Ramanantsoa Harrimann, and Totohasina André</i>	
Field-Reliability Predictions Based on Statistical System Lifecycle Models. . .	98
<i>Lukas Felsberger, Dieter Kranzlmüller, and Benjamin Todd</i>	
Building a Knowledge Based Summarization System for Text Data Mining . . . . .	118
<i>Andrey Timofeyev and Ben Choi</i>	
Spanish Twitter Data Used as a Source of Information About Consumer Food Choice. . . . .	134
<i>Luis G. Moreno-Sandoval, Carolina Sánchez-Barriga, Katherine Espíndola Buitrago, Alexandra Pomares-Quimbaya, and Juan Carlos Garcia</i>	

Feedback Matters! Predicting the Appreciation of Online Articles <i>A Data-Driven Approach . . . . .</i>	147
<i>Catherine Sotirakou, Panagiotis Germanakos, Andreas Holzinger, and Constantinos Mourlas</i>	
Creative Intelligence – Automating Car Design Studio with Generative Adversarial Networks (GAN) . . . . .	160
<i>Sreedhar Radhakrishnan, Varun Bharadwaj, Varun Manjunath, and Ramamoorthy Srinath</i>	
<b>MAKE-Text</b>	
A Combined CNN and LSTM Model for Arabic Sentiment Analysis. . . . .	179
<i>Abdulaziz M. Alayba, Vasile Palade, Matthew England, and Rahat Iqbal</i>	
Between the Lines: Machine Learning for Prediction of Psychological Traits - A Survey . . . . .	192
<i>Dirk Johannßen and Chris Biemann</i>	
LawStats – Large-Scale German Court Decision Evaluation Using Web Service Classifiers. . . . .	212
<i>Eugen Ruppert, Dirk Hartung, Phillip Sittig, Tjorben Gschwander, Lennart Rönneburg, Tobias Killing, and Chris Biemann</i>	
Clinical Text Mining for Context Sequences Identification . . . . .	223
<i>Svetla Boytcheva</i>	
<b>MAKE-Smart Factory</b>	
A Multi-device Assistive System for Industrial Maintenance Operations. . . . .	239
<i>Mario Heinz, Hitesh Dhiman, and Carsten Röcker</i>	
Feedback Presentation for Workers in Industrial Environments – Challenges and Opportunities . . . . .	248
<i>Mario Heinz and Carsten Röcker</i>	
<b>MAKE-Topology</b>	
On a New Method to Build Group Equivariant Operators by Means of Permutants . . . . .	265
<i>Francesco Camporesi, Patrizio Frosini, and Nicola Quercioli</i>	
Topological Characteristics of Digital Models of Geological Core . . . . .	273
<i>Rustem R. Gilmanov, Alexander V. Kalyuzhnyuk, Iskander A. Taimanov, and Andrey A. Yakovlev</i>	

Shortened Persistent Homology for a Biomedical Retrieval System  
with Relevance Feedback . . . . . 282  
*Alessia Angeli, Massimo Ferri, Eleonora Monti, and Ivan Tomba*

**MAKE Explainable AI**

Explainable AI: The New 42?. . . . . 295  
*Randy Goebel, Ajay Chander, Katharina Holzinger, Freddy Lecue,  
Zeynep Akata, Simone Stumpf, Peter Kieseberg, and Andreas Holzinger*

A Rule Extraction Study Based on a Convolutional Neural Network . . . . . 304  
*Guido Bologna*

Evaluating Explanations by Cognitive Value . . . . . 314  
*Ajay Chander and Ramya Srinivasan*

Measures of Model Interpretability for Model Selection . . . . . 329  
*André Carrington, Paul Fieguth, and Helen Chen*

Regular Inference on Artificial Neural Networks . . . . . 350  
*Franz Mayr and Sergio Yovine*

**Author Index** . . . . . 371



# Current Advances, Trends and Challenges of Machine Learning and Knowledge Extraction: From Machine Learning to Explainable AI

Andreas Holzinger<sup>1,2</sup> , Peter Kieseberg<sup>3,4</sup>, Edgar Weippl<sup>3,5</sup>,  
and A Min Tjoa<sup>6</sup>

- <sup>1</sup> Holzinger Group, Institute for Medical Informatics, Statistics and Documentation,  
Medical University Graz, Graz, Austria  
a.holzinger@hci-kdd.org
- <sup>2</sup> Institute of Interactive Systems and Data Science,  
Graz University of Technology, Graz, Austria
- <sup>3</sup> SBA Research, Vienna, Austria
- <sup>4</sup> University of Applied Sciences St. Pölten, St. Pölten, Austria
- <sup>5</sup> Christian Doppler Laboratory for Security and Quality Improvement  
in the Production System Lifecycle, TU Wien, Vienna, Austria
- <sup>6</sup> Information & Software Engineering Group, Institute of Information Systems  
Engineering, TU Wien, Vienna, Austria

**Abstract.** In this short editorial we present some thoughts on present and future trends in Artificial Intelligence (AI) generally, and Machine Learning (ML) specifically. Due to the huge ongoing success in machine learning, particularly in statistical learning from big data, there is rising interest of academia, industry and the public in this field. Industry is investing heavily in AI, and spin-offs and start-ups are emerging on an unprecedented rate. The European Union is allocating a lot of additional funding into AI research grants, and various institutions are calling for a joint European AI research institute. Even universities are taking AI/ML into their curricula and strategic plans. Finally, even the people on the street talk about it, and if grandma knows what her grandson is doing in his new start-up, then the time is ripe: We are reaching a new AI spring. However, as fantastic current approaches seem to be, there are still huge problems to be solved: the best performing models lack transparency, hence are considered to be black boxes. The general and worldwide trends in privacy, data protection, safety and security make such black box solutions difficult to use in practice. Specifically in Europe, where the new General Data Protection Regulation (GDPR) came into effect on May, 28, 2018 which affects everybody (right of explanation). Consequently, a previous niche field for many years, explainable AI, explodes in importance. For the future, we envision a fruitful marriage between classic logical approaches (ontologies) with statistical approaches which may lead to context-adaptive systems (stochastic ontologies) that might work similar as the human brain.



**Keywords:** Machine learning · Knowledge extraction  
Artificial intelligence · Explainable AI · Privacy

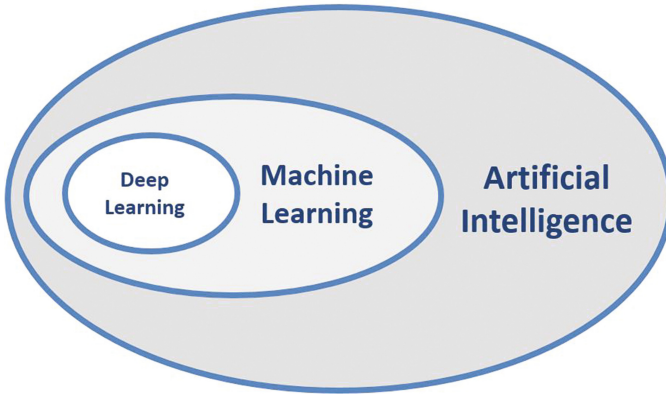
## 1 Introduction

Artificial intelligence (AI) has a long tradition in computer science, reaching back to 1950 and earlier [24]. In the first three decades, industry, governments and the public had extremely high expectations to reach the “mythical” human-level machine intelligence [9,17]. As soon as it turned out that the expectations were too high, and AI could not deliver these high promises, a dramatic “AI winter” affected the field; even the name AI was avoided at that time [8].

The field recently gained enormous interest due to the huge practical success in Machine Learning & Knowledge Extraction. Even in famous journals including Science [12] or Nature [15] the success of machine learning was recently presented. This success is visible in many application domains of our daily life from health care to manufacturing. Yet, many scientists of today are still not happy about the term, as “intelligence” is not clearly defined and we are still far away from reaching human-level AI [18].

Maybe the most often asked question is: “What is the difference between Artificial Intelligence (AI) and Machine Learning (ML) – and is deep learning (DL) belonging to either AI or ML?”. A formal short answer: Deep Learning is part of Machine Learning is part of Artificial Intelligence:  $DL \subset ML \subset AI$

This follows the popular Deep Learning textbook by Ian Goodfellow, Yoshua Bengio & Aaron Courville (2016, see Fig. 1):



**Fig. 1.** A question most often asked: What is the difference between AI, ML and DL, see also [6].

## 2 Trend Indicators

### Industry as Trend Indicator

Many global industrial players from Amazon to Zalando have now concerted international efforts in AI. The topic is so hot, that e.g. Google Brain has recently itself renamed to Google AI. Start-ups are emerging at an unprecedented rate - AI spring is here.

### Funding as Trend Indicator

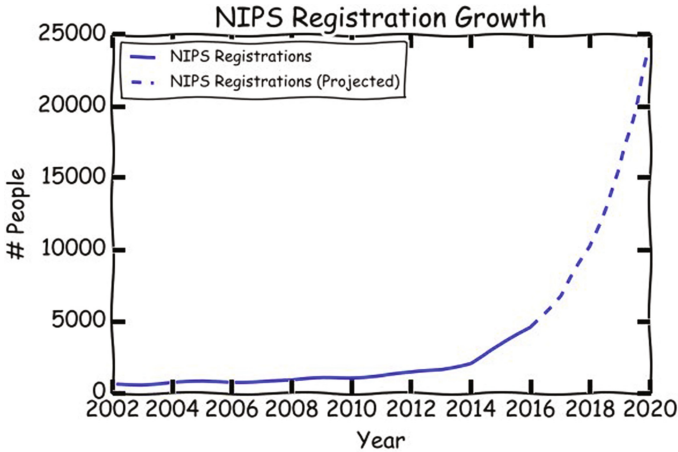
Worldwide, enormous grants are now fostering AI research generally and machine learning specifically: DARPA in the US or Volkswagen Stiftung in Germany are only two examples. The European Union targets for a total of 20 BEUR bringing into AI research in the future across both, public and private sectors. Health is one of the central targets, which is easy to understand as it is a topic that affects everybody. The primary direction was set in the last Horizon2020 initiative: The goal is to develop an European AI ecosystem, bringing together knowledge, algorithms, tools and resources available and making it a compelling solution for users, especially from non-tech sectors (such as health). The aim is to mobilize the European AI community including scientists, businesses and start-ups to provide access to knowledge, algorithms and tools. On the EU agenda are particularly ELSE aspects, where ELSE stands for Ethical, Legal and Socio-Economic issues.

At the same time there is the ELLIS initiative (<https://ellis-open-letter.eu>) which urges for seeing machine learning at the heart of a technological and societal artificial intelligence revolution involving multiple sister disciplines, with large implications for the future competitiveness of Europe. The main critique is that currently Europe is not keeping up: most of the top laboratories, as well as the top places to do a PhD, are located in North America or Canada; moreover, ML/AI investments in China and North America are significantly larger than in Europe. As an important measure to address these points, the ELLIS initiative proposes to found a *European Lab for Learning & Intelligent Systems* (working title; abbreviated as “ELLIS”), involving the very best European academics while working together closely with researchers from industry, ensuring to have economic impact and the creation of AI/ML jobs in Europe. This mission is meanwhile supported by IFIP TC 12.

In the UK the House of Lords (see the report by Wendy Hall and Jerome Presenti from October, 15, 2017: [bit.ly/2HCEXhx](http://bit.ly/2HCEXhx)) is convinced that the UK can lead in AI by building on a historically strong research program, which proposes five principles [19]: 1. AI should be developed for the common good and benefit of humanity. 2. AI should operate on principles of intelligibility and fairness. 3. AI should not be used to diminish the data rights or privacy of individuals, families or communities. 4. All citizens have the right to be educated to enable them to flourish mentally, emotionally and economically alongside AI. 5. The autonomous power to hurt, destroy or deceive human beings should never be vested in AI.

## Conferences as Trend Indicator

A good indicator for the importance of machine learning is the conference on Neural Information Processing Systems (NIPS) - which is now trying to re-name itself. This conference was first held in Denver in December 1987 as a small meeting. The conference beautifully reflects the success of statistical learning methods attracting more and more researchers from machine learning (see Fig. 2).



**Fig. 2.** NIPS 2017 in Long Beach was the most popular ML conference yet, attracting over 8,000 registered attendees, following the 2016 event with 6,000 registered attendees in Barcelona (image taken from Ian Goodfellow’s tweet on June, 15, 2018).

## 3 Main Problems Today

A major issue with respect to explaining machine learning algorithms lies in the area of privacy protection: Trust is one of the core problems when dealing with personal, and potentially sensitive, information, especially when the algorithms in place are hard or even impossible to understand. This can be a major risk for acceptance, not only by the end users, like e.g. hospital patients, or generally in safety-critical decision making [10], but also among the expert engineers that are required to train the models, or, in case of an expert-in-the-loop approach, partake in the daily interaction with the expert system [14]. One option is to include risk management practice early in the project to manage such risks [11]. Trust and Privacy are actually a twofold problem in this regard; an example from the medical domain shall illustrate this: The patients need to be able to trust the machine learning environment that their personal data is secured and protected against theft and misuse, but also that the analytical processes working on their data are limited to the selection they have given consent to.

For the expert, on the other hand, there is the need to trust the environment that their input to the system is not manipulated later on. Furthermore, usability is a fundamental factor for successfully integrating experts into AI systems, which, again, requires the designers of the interfaces to understand the fundamentals of the system in place. Here it must be noted that usability and security are often considered fundamental opposites, hence research in the so-called area of *usable security* [3] is urgently needed.

A topic closely related to the issue of security and privacy, but still different in nature, is the issue of fingerprinting/watermarking information [22]. Many approaches in utilizing data for data driven research face the problem that data must be shared between partners, i.e. data sets are either sent to a central analysis repository for further processing, or directly shared between the partners themselves. While the earlier approach allows for some kind of control over the data by the trusted third party operating the analysis platform, in the later one, the original owner potentially gives up control over the data set. This might not even be a problem with respect to privacy, as the data shared with the other partners will in most cases obey data protection rules as put forth by various regulations, still, this data might be an asset of high (monetary) value. Thus, when sharing the data with other partners, it must be made sure that the data is not further illegally distributed. A typical reactive approach to this problem is the implementation of so-called *fingerprints* or *watermarks*; these can also be used to embed information that helps to detect collusion in deanonymization attacks [13,21]. Both terms, fingerprinting and watermarking, are often used synonymously by authors, while others differentiate them as watermarks being mechanisms that prove the authenticity and ownership of a data set and fingerprints actually being able to identify the data leak by providing each partner with the same basic set marked with different fingerprints.

Throughout the past decades, watermarking and fingerprinting of information has gained a lot of attention in the research community, most notably regarding the protection of digital rights in the music and movie industries [23]. Approaches for marking data have also been put forth (e.g. [1]) and while a lot of them exist nowadays, most of them only focus on marking whole data sets and fail with partially leaked sets. Thus, in order to provide transparency with respect to privacy, as well as explainability, we propose that a fingerprinting mechanism within data driven research requires the following criteria:

1. **Single Record Detection:** The detection of the data leak should be possible with only one single leaked (full) record. This is a major obstacle for most algorithms that rely on adding or removing so-called *marker*-records from the original data set.
2. **Collusion Protection:** Several partners being issued the same fingerprinted data set might collude in order to extract and remove the fingerprints, or even frame another partner. The fingerprinting algorithm is required to be stable against such kinds of attacks.
3. **High Performance:** In order to make this protection mechanism usable, it must not require a lot of resources, neither with respect to calculation time

(for both, the generation of the fingerprint, as well as the detection), nor with respect to additional storage requirements.

4. **Low distortion:** The algorithm must not introduce a large amount of additional distortion, thus further reducing the value of the data used in the analysis.

The development of novel techniques in this area is thus another open problem that has a high potential for future research. When developing new solutions contradicting requirements including future improvements in “counter-privacy”, aka. forensics [2], have to be considered.

Last, but not least, the need to understand machine learning algorithms is required to deal with distortion: Due to novel regulations in the European Union, especially the General Data Protection Regulation (GDPR), the protection of privacy has become extremely important and consent for processing personal information has to be asked for rather narrow use cases, i.e. there is no more “general consent”. Thus, research labs tend to consider anonymizing their data, which makes it non-personal information and thus consent-free to use. Still, as it has already been shown [16], many standard anonymization techniques introduce quite a large amount of distortion into the end results of classical machine learning algorithms. In order to overcome this issue, additional research in the area of Privacy Aware Machine Learning (PAML) is needed: The distortion needs to be quantified in order to be able to select the anonymization algorithm/machine learning algorithm pairing that is ideal with respect to the given data set. Explainable AI can be a major enabler for this issue, as understanding decisions would definitely help in understanding and estimating distortions. In addition, algorithms (both, for anonymization and machine learning) need to be adapted in order to reduce the distortion introduced, again, a task where the black-box characteristics of machine learning nowadays is an issue. Thus, explainable AI could be the key to designing solutions that harness the power of machine learning, while guaranteeing privacy at the same time.

## 4 Conclusion

To provide an answer to the question “*What are the most interesting trends in machine learning and knowledge extraction?*”: the most interesting ones are not known yet. What we know is that the driver for the AI hype is success in machine learning & knowledge extraction. A promising future approach is the combination of ontologies with probabilistic approaches. Traditional logic-based technologies as well as statistical ML constitute two indispensable technologies for domain specific knowledge extraction, actively used in knowledge-based systems. Here we urgently need solutions on how the two can be successfully integrated, because to date both technologies are mainly used separately, without direct connection.

The greatest problem, however, is the problem of black box algorithms. These make machine decisions intransparent and non-understandable, even to the eyes of experts, which reduces trust in ML specifically and AI generally.

Another field that requires more research is the intersection between security (and especially privacy related) research and ML - be it in the form of privacy aware machine learning, where the distortion from data protection mechanisms is mitigated, or rather in the areas of protecting ownership on information or providing trust into the results of ML algorithms. All of these areas could greatly benefit from explainable AI, as the design of novel mechanisms to achieve these security and privacy tasks cannot be soundly done without further insight into the internal workings of the systems they are protecting.

A final remark of applications: According to the ML initiative of the Royal Society the greatest benefit of AI/ML will be in improved medical diagnosis, disease analysis and pharmaceutical development. This on the other hands needs making results transparent, re-traceable and to understand the *causality of learned representations* [4,20].

Consequently, the most promising field in the future is what is called explainable AI [5] where DARPA has already launched a funding initiative in 2016 [7]. This calls for a combination of logic-based approaches (ontologies) with probabilistic machine learning to build *context adaptive systems*.

**Acknowledgements.** The authors thank their colleagues for valuable feedback, remarks and critics on this editorial introduction. The competence center SBA Research (SBA-K1) is funded within the framework of COMET – Competence Centers for Excellent Technologies by BMVIT, BMDW, and the federal state of Vienna, managed by the FFG. This research was also funded by the CDG Christian Doppler Laboratory SQI and by the KIRAS program of the FFG.

## References

1. Agrawal, R., Kiernan, J.: Watermarking relational databases. In: VLDB 2002: Proceedings of the 28th International Conference on Very Large Databases, pp. 155–166. Elsevier (2002)
2. Frühwirt, P., Kieseberg, P., Schrittwieser, S., Huber, M., Weippl, E.: Innodb database forensics: reconstructing data manipulation queries from redo logs. In: 2012 Seventh International Conference on Availability, Reliability and Security (ARES), pp. 625–633. IEEE (2012)
3. Garfinkel, S., Lipford, H.R.: Usable security: history, themes, and challenges. Synthesis Lectures on Information Security, Privacy, and Trust **5**(2), 1–124 (2014)
4. Gershman, S.J., Horvitz, E.J., Tenenbaum, J.B.: Computational rationality: a converging paradigm for intelligence in brains, minds, and machines. *Science* **349**(6245), 273–278 (2015)
5. Goebel, R.: Explainable ai: the new 42? In: Holzinger, A., et al. (eds.) CD-MAKE 2018. LNCS, vol. 11015, pp. 295–303. Springer, Cham (2018)
6. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press, Cambridge (MA) (2016)
7. Gunning, D.: Explainable artificial intelligence (XAI): Technical report Defense Advanced Research Projects Agency DARPA-BAA-16-53. DARPA, Arlington, USA (2016)
8. Hendler, J.: Avoiding another ai winter. *IEEE Intell. Syst.* **23**(2), 2–4 (2008)

9. Hernández-Orallo, J.: *The Measure of all Minds: Evaluating Natural and Artificial Intelligence*. Cambridge University Press, Cambridge (2016)
10. Holzinger, K., Mak, K., Kieseberg, P., Holzinger, A.: Can we trust machine learning results? artificial intelligence in safety-critical decision support. *ERCIM News* **112**(1), 42–43 (2018)
11. Islam, S., Mouratidis, H., Weippl, E.R.: An empirical study on the implementation and evaluation of a goal-driven software development risk management model. *Inf. Softw. Technol.* **56**(2), 117–133 (2014)
12. Jordan, M.I., Mitchell, T.M.: Machine learning: trends, perspectives, and prospects. *Science* **349**(6245), 255–260 (2015)
13. Kieseberg, P., Schrittwieser, S., Mulazzani, M., Echizen, I., Weippl, E.: An algorithm for collusion-resistant anonymization and fingerprinting of sensitive microdata. *Electron. Markets* **24**(2), 113–124 (2014)
14. Kieseberg, P., Weippl, E., Holzinger, A.: Trust for the doctor-in-the-loop. *ERCIM News* **104**(1), 32–33 (2016)
15. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* **521**(7553), 436–444 (2015)
16. Malle, B., Kieseberg, P., Schrittwieser, S., Holzinger, A.: Privacy aware machine learning and the right to be forgotten. *ERCIM News* **107**(10), 22–3 (2016)
17. McCarthy, J.: *Programs with common sense*. pp. 75–91. RLE and MIT Computation Center (1960)
18. McCarthy, J.: From here to human-level ai. *Artif. Intell.* **171**(18), 1174–1182 (2007)
19. Olhede, S.: The AI spring of 2018. *Significance* **15**(3), 6–7 (2018)
20. Peters, J., Janzing, D., Schölkopf, B.: *Elements of causal inference: foundations and learning algorithms*. Cambridge, MA (2017)
21. Schrittwieser, S., Kieseberg, P., Echizen, I., Wohlgemuth, S., Sonehara, N., Weippl, E.: An algorithm for  $k$ -anonymity-based fingerprinting. In: Shi, Y.Q., Kim, H.-J., Perez-Gonzalez, F. (eds.) *IWDW 2011*. LNCS, vol. 7128, pp. 439–452. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-32205-1\\_35](https://doi.org/10.1007/978-3-642-32205-1_35)
22. Shih, F.Y.: *Digital Watermarking and Steganography: Fundamentals and Techniques*. CRC Press, Boca Raton (2017)
23. Swanson, M.D., Kobayashi, M., Tewfik, A.H.: Multimedia data-embedding and watermarking technologies. *Proc. IEEE* **86**(6), 1064–1087 (1998)
24. Turing, A.M.: Computing machinery and intelligence. *Mind* **59**(236), 433–460 (1950)

**MAKE-Main Track**





# A Modified Particle Swarm Optimization Algorithm for Community Detection in Complex Networks

Alireza Abdollahpouri<sup>1</sup>(✉), Shadi Rahimi<sup>1</sup>,  
Shahnaz Mohammadi Majd<sup>2</sup>, and Chiman Salavati<sup>1</sup>

<sup>1</sup> Department of Computer Engineering, University of Kurdistan, Sanandaj, Iran  
Abdollahpouri@gmail.com

<sup>2</sup> Department of Mathematics, Islamic Azad University of Sanandaj,  
Sanandaj, Iran

**Abstract.** Community structure is an interesting feature of complex networks. In recent years, various methods were introduced to extract community structure of networks. In this study, a novel community detection method based on a modified version of particle swarm optimization, named PSO-Net is proposed. PSO-Net selects the modularity  $Q$  as the fitness function which is a suitable quality measure. Our innovation in PSO algorithm is changing the moving strategy of particles. Here, the particles take part in crossover operation with their personal bests and the global best. Then, in order to avoid falling into the local optimum, a mutation operation is performed. Experiments on synthetic and real-world networks confirm a significant improvement in terms of convergence speed with higher modularity in comparison with recent similar approaches.

**Keywords:** Community detection · Complex network · PSO · Modularity

## 1 Introduction

Most of real-world complex systems can be represented as complex networks. Social networks such as Facebook, collaboration networks such as scientific networks, technological networks such as the Internet and biological networks such as protein interaction networks are only some examples. Networks are modeled as graphs, where vertices represent individual objects and edges indicate relationships among these objects. One of the important properties of complex networks is “community structure” [1]. The term community is considered as a group of nodes within a graph with more internal connections than external connections to the rest of the network [2]. The detection of community structure, is a great important research topic in the study of complex networks, because it can detect the hidden patterns existing in complex systems. Therefore, a significant amount of efforts have been devoted to develop methods that can extract community structures from complex networks [1, 3–6].

Fortunato in [7] studied the community discovery methods in detail and divided them into several categories. Although special strategies adopted are different, most of the algorithms are mainly divided into two basic categories including: hierarchical

clustering methods [1, 3–6, 8–11] and optimization based methods [12–21]. In hierarchical clustering, a network is grouped into a set of clusters in multiple levels, which each level presents a particular partition of the network. Hierarchical clustering methods can be further divided into two groups, depending on how they build the clustering tree: divisive algorithms [1, 4, 6, 9] and agglomerative algorithms [3, 8, 11, 22]. In divisive methods, which is a top-down approach, in each iteration, the graph is divided into two groups. This process is continued until each node is assigned by a distinct cluster label. On the other hand, in agglomerative approaches (i.e., bottom-up methods), clusters are iteratively merged if their similarity is sufficiently high.

In optimization based algorithms, the community detection task is transformed into an optimization problem and the goal is to find an optimal solution with respect to a pre-defined objective function. Network modularity employed in several algorithms [1, 3, 23] and cut criteria adopted by spectral methods [24, 25], are two examples of objective functions. Evolutionary algorithms (EAs) have been successfully applied to identify community structures in complex networks [14, 19, 20]. Genetic algorithm (GA) as a well-known EA, have been frequently used for community detection among the other EA methods [15, 17, 19, 20, 26, 27]. The existing GA-based algorithms have some advantages such as parallel search and some drawbacks such as slow convergence [28]. Also, it has been shown that the GA may stick at local optimal solution and therefore, can hardly find the optimal solution [27]. There are also some challenging problems regarding GA based community detection methods such as discovering reasonable community structure without prior knowledge, and further improvement of the detection accuracy. On the other hand, swarm intelligence-based methods such as particle swarm optimization (PSO) have been successfully used in the literature to solve optimization problems [29]. PSO is a global search method which is originally developed by Kennedy and Eberhart and inspired by the paradigm of birds flocking [29]. PSO initialize the system with a population of random particles. Each particle keeps track of its coordinates in space which are associated with the best solution it has obtained (local optima) and the best solution of the population (global optima). The particles in any movement try to minimize their distances from these two positions. PSO has the advantage of easy implementation and inexpensive computationally for many problems.

In this paper, a novel PSO based approach, called PSO-Net is proposed to discover communities in complex networks. PSO-Net explores the search space without the need to know the number of communities in advance. In the proposed method a specific modularity measure is used to compute the quality of discovered communities, and then a PSO based search process is employed to explore the search space. In PSO-Net two crossover operators are applied to update particle positions and then a mutation operator is used to spread the solutions through the search space. Experiments on a synthetic and several well-known real-world networks such as Zachary’s Karate Club network, the Dolphin social network, American College Football and the Books about US politics network, show the capability of the PSO-Net method to correctly detect communities with better or competitive results compare with other approaches.

The rest of this paper is organized as follows. Section 2 describes the description of the problem and related research on community detection. In Sect. 3, the proposed modified particle swarm optimization algorithm (PSO-Net) for community detection is

presented. Section 4 presents the experimental results on synthetic and real world networks with their related analysis, and finally, the conclusion is provided in Sect. 5.

## 2 Community Definition and Related Works

### 2.1 Community Definition

Let us consider a network  $N$  which is modeled as a graph  $G = (V, E)$ , where  $V$  denotes a set of nodes, and  $E$  is a set of edges linking each two nodes. Community is defined as a group of nodes (sub-graph) that has more intra-edges than inter-edges. Most formal definition for community has been introduced in [2]. Suppose that, adjacency matrix of  $G$  is  $A$ , where the element  $a_{ij}$  is 1 if there is an edge between node  $i$  and node  $j$ , and 0 otherwise. The degree of node  $i$  is defined as  $k_i = \sum_j a_{ij}$ . Suppose, the node  $i$  is placed to a sub-graph  $S \subset G$ , the degree of  $i$  with respect to  $S$  can be split as  $k_i(s) = k_i^{in}(s) + k_i^{out}(s)$ , where  $k_i^{in}(s)$  is the number of edges connecting  $i$  to the nodes of  $S$ , and  $k_i^{out}(s)$  is the number of edges connecting node  $i$  the outside of  $S$  (i.e.,  $G \setminus S$ ).

### 2.2 Related Works

In recent years, community detection methods have been successfully applied in different research areas such as sociology, physics, biology, and computer science [1–4, 15, 18, 20, 23]. Community detection methods can be divided into two approaches including; hierarchical and optimization-based approaches. As mentioned previously, hierarchical clustering method groups data objects into a tree of clusters to produce multilevel clustering. This type of clustering is further divided to divisive and agglomerative methods. In divisive methods, a given graph is split iteratively into smaller and smaller subgraphs. Up to now, several divisive methods have been proposed in the literature. For example, the Girvan-Newman (GN) algorithm proposed in [1, 4] is a divisive method that extracts the network's communities removing the edges with the highest value of edge betweenness. This process is continued until the graph is divided into two separate subgraphs. The betweenness of an edge is defined as the number of shortest paths which are passing from that edge [30, 31]. A variation of GN algorithm is proposed by Fortunato et al. in [9]. In their method, the concept of information centrality [32] as a way to measure edge centrality, is used instead of edge betweenness. In their method, communities are discovered by repeatedly identifying and removing the edges with the highest centrality measure. In [6], another divisive algorithm is proposed to find communities based on the principle of GN method. In order to quantify the relevance of each edge in a network, the authors applied three edge centralities based on network topology, walks and paths, respectively.

Agglomerative hierarchical clustering is a bottom-up clustering method. Till now, several agglomerative graph clustering methods have been proposed in the literature. For example, in [3] an agglomerative clustering algorithm called Fast-Newman (FN) is proposed. In this method, a modularity measure is used to merge clusters iteratively until there is no improvement in modularity. Another example of this type of clustering,

is the method proposed in [8]. This algorithm begins with a community division using prior knowledge of the network structure (degrees of the nodes), and then combines the communities as an iterative optimization process for modularity until a clear partition is obtained.

On the other hand, optimization based methods employ an objective function in their processes to evaluate the quality of found clusters. This process is continued until an optimal clustering result is found in the whole solution space. For instance in [4] an objective function called Q-modularity is used in community detection process. In this case, the community detection becomes a modularity optimization problem. In general, the obtained communities are more accurate when the value of Q is larger. Also, Brandes et al. in [33] showed that searching for the optimal modularity value is a NP-complete problem and therefore, it cannot be solved in polynomial time. Thus, many metaheuristic algorithms such as: ant colony optimization [16, 34], genetic algorithm [17, 19, 20, 27] and Extremal Optimization (EO) [13] and other metaheuristic algorithms [12, 14, 21] have been applied to solve community detection problem.

Generally, the metaheuristic methods are defined as an iterative process which employing a learning strategy to effectively explore the search space. Several metaheuristic based methods have been proposed to identify communities in complex networks. For example, taking advantage of genetic algorithm, Pizzuti proposed a new algorithm (named GA-Net), for this purpose [19]. This approach introduced the concept of community score to measure the quality of identified communities. Shang et al. [27] proposed an improved genetic algorithm for community discovery method based on the modularity concept. The computational complexity of this method is very high compare to the traditional modularity-based community detection methods. To overcome this problem, Liu et al. in [34] proposed an ant colony optimization based method for community discovery. The authors employed movement, picking-up and dropping-down operators to perform node clustering in email networks. The authors of [20] proposed a multiobjective approach for community discovery, considering both *community score* and *community fitness* concepts as its objectives. In [21], a hybrid algorithm based on PSO and EO was proposed by employing a special encoding scheme based Ji et al. proposed an ant colony clustering algorithm with an accuracy measure to identify communities in complex networks. Their algorithm focuses on the strategy of ant perception and movements and the method of pheromone diffusion and updating, and searches for an optimal partitioning of the network by ant colony movements [16].

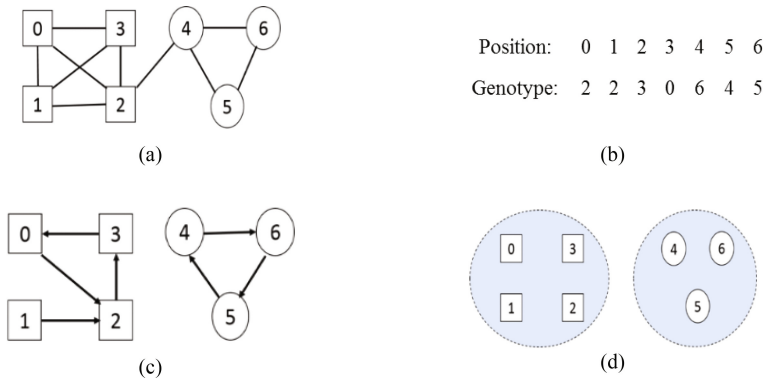
### 3 Proposed Method

In this section, the proposed community detection method called PSO-Net is described in detail. The proposed method consists of two main steps including; *Initialization* and *Moving*. In initialization step, first a suitable representation for a solution which demonstrates a partitioning of a network is considered. Afterward, the solutions are randomly initialized. Then in the next step, inspired from PSO search strategy, the solutions are moved around the search space to optimize an objective (modularity)

function. In the search process of the proposed method, the solutions are moved toward local and global best solutions which are performed by means of a specific crossover operator. Moreover, in order to expand the solution space, a random mutation operation is performed on each particle. The pseudo code of the proposed method is shown in Algorithm 1. Additional details of the steps in proposed method are described in their corresponding sections.

### 3.1 Initialization

The proposed method exploits the locus-based adjacency representation (LAR) [35]. In LAR scheme, each solution considered as an array of  $N$  genes, each of which belongs to a node and each gene takes its values in the range of  $1, 2, \dots, N$ . Each solution represents a new graph which the value of  $j$  for the gene  $i$ , means that, there is a link between node  $i$  and node  $j$  in this graph and each connected component represent a cluster. For example, Fig. 1, illustrates LAR scheme for a network with seven nodes. In Fig. 1(a) the graph structure of the network is drawn. Figure 1(b), shows a solution that was represented by the LAR scheme. As can be seen, for each gene, a value in the range of 1 to  $N$  is assigned. According to the Fig. 1(c), the seventh node with position of 6, takes the value of 5, meaning that, in corresponding graph, there is a link from node 6 to node 5. Thereupon, these two nodes are placed in a same cluster, which can be seen in Fig. 1(d).



**Fig. 1.** Locus-based adjacency representation. (a) The topology of the graph. (b) One possible genotype. (c) Translation of (b) to the graph structure. (d) The community structure

**Algorithm1. Particle swarm based community detection (PSO-Net)**


---

**Input**     A complex network modeled by  $G = (V, E)$   
*I*: Number of iteration that algorithm repeated  
*p*: Number of particles  
@*Init*: Function that generates particles with given number by Locus-Based Representation Scheme.  
@*Decoder*: Function that decodes a given particle and resulted a partitioning of a given graph.  
@*Modularity*: Function that computes modularity of a given clustering.  
@*TwoPointCrossover*: Function that performs a two point crossover between two given particles  
@*Mutation*: Function that mutates a given particle

**Output**     $C = \{Cluster_1, Cluster_2, \dots, Cluster_k\}$

---

```

1:  Begin algorithm
2:  Population =  $\emptyset$ , Clusterings =  $\emptyset$ , Fitness =  $\emptyset$ , Personalbest =  $\emptyset$ , Globalbest =  $\emptyset$ 
3:  Run @Init by p parameter to produce p particles and assign to Population array.
4:  Apply @Decoder to obtain community structure,  $\forall i = 1 \dots p$  and then, assign to Clusterings array.
5:  Apply @Modularity to set the modularity of solutions,  $\forall i = 1 \dots p$  and then, assign to Fitness array.
6:  for i=1 to p
7:     Personalbest(i) = Population(i)
8:  end for
9:  Find the particle with maximum value in Fitness array and assign to Globalbest
10: for i=1 to I
11:   for j = 1 to p
12:    Temp_Particle =  $\emptyset$ , Child1 =  $\emptyset$ , Child2 =  $\emptyset$ , Clus1 =  $\emptyset$ , Clus2 =  $\emptyset$ , Tempclus =  $\emptyset$ 
13:    Initialize Child1 and Child2 by applying @TwoPointCrossover on Population(i) and Personalbest(i)
14:    Initialize Clus1 and Clus2 by applying @Decoder on Child1 and Child2
15:    Apply @Modularity to compute the modularity of Clus1 and Clus2
16:    Select the Child with maximum modularity among Clus1 and Clus2 and then, assign to Temp_Particle
17:    Update Child1 and Child2 by applying @TwoPointCrossover on Temp_Particle and Globalbest
18:    Update Clus1 and Clus2 by applying @Decoder on Child1 and Child2
19:    Apply @Modularity to compute the modularity of Clus1 and Clus2
20:    Select the Child with maximum modularity among Clus1 and Clus2 and then, assign to Temp_Particle
21:    Apply @Mutation on Temp_Particle
22:    Population(i) = Temp_Particle
23:    Apply @Decoder on Population(i) and then, assign to Clustering(i)
24:    Apply @Modularity to compute modularity of Clustering(i), and then, update Fitness(i)
25:    if (Fitness(i)  $\geq$  Fitness(Personalbest(i))) then
26:       Personalbest(i) = Population(i)
27:    end if
28:  end for
29:  Update Globalbest by the particle that has highest maximum Fitness value
30: end for
31: Return Globalbest
32: End algorithm

```

---

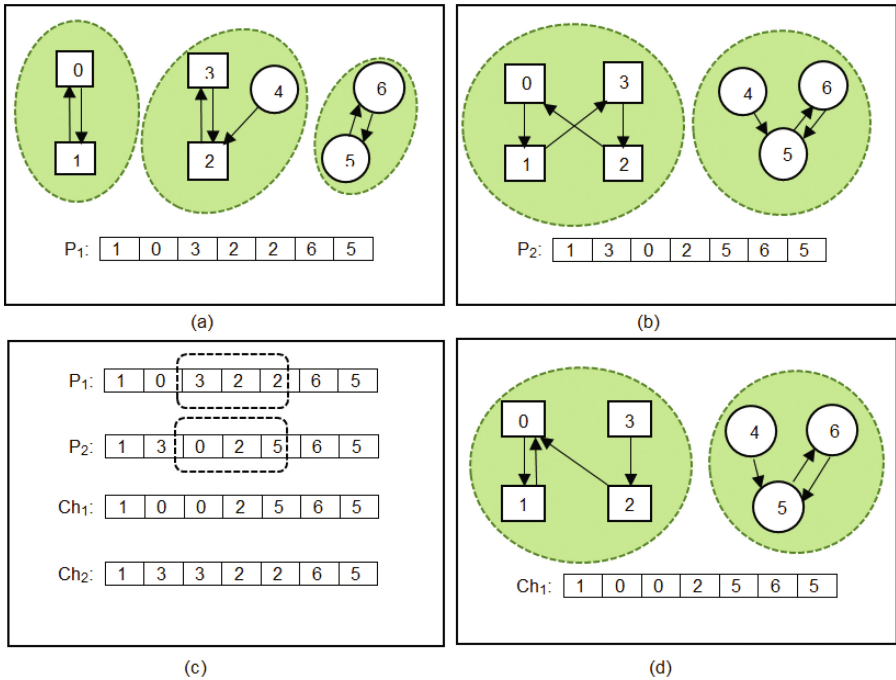
The LAR encoding scheme has some benefits. First, it is dispensable to determine the number of communities in advance, because of automatically determination in the

decoding step. Besides, the decoding process can be done in a linear time. Then, standard crossover operators can be easily employed over these types of representation. To initialize the system, a population of random individuals is generated such that for each node  $i$ , the value of  $g_i$  is randomly chosen among one of its neighboring nodes, which indicates the edge  $(i, j)$  in the graph. This type of initialization improves the convergence of the algorithm, due to restriction of the solution space.

### 3.2 Search Strategy

In order to move each solution towards the best positions, we use genetic operators, i.e., crossover and mutation operators as follows.

**Moving Toward Personal Best.** At first, for each particle a two-point crossover with its personal best is performed and then as a result, two new solutions are obtained. For example, given two parents  $P_1$  and  $P_2$  and two random points  $i$  and  $j$ , binary string from beginning of chromosome to the crossover point  $i$  is copied from parent  $P_1$ , the part from crossover point  $i$  to the crossover point  $j$  is copied from the parent  $P_2$  and the rest is copied from the parent  $P_1$ . This action creates the first child. To produce the second child, this action is done in reverse order. (See Fig. 2). Finally, a solution with higher fitness value, i.e., higher modularity, is selected as a temporary position of current particle.



**Fig. 2.** Two point crossover. (a) P1 and corresponding graph structure. (b) P2 and corresponding graph structure. (c) A random two-point crossover of the genotypes yields the children Ch1 and Ch2. (d) Ch1 and its graph structure.

**Moving Toward Global Best.** To move towards the global best, a two-point crossover is performed between a particle and the global best of population. In this case, two new solutions are obtained. The one with a higher modularity value is selected as temporary state of current particle.

### 3.3 Enhancing Search Ability

Finally, to move the solutions around the whole search space, one-point neighbour-based mutation is performed on all particles. Such that, for each particle, a gene  $i$  is picked randomly and the possible values for this gene are limited to its neighbours to guarantee that -solution space has only possible solutions.

### 3.4 Fitness Computation

Modularity of a network [4], measures the goodness of identified communities. A quantitative definition of the modularity can be the fraction of the edges that fall within the clusters minus the anticipated value of this fraction while edges fall at random in a network regardless of the community structure. Let  $k$  be the number of clusters found inside a network, the modularity  $Q$  is defined as (Eq. 1).

$$Q = \sum_{s=1}^k \left[ \frac{l_s}{m} - \left( \frac{d_s}{2m} \right)^2 \right] \quad (1)$$

Where,  $l_s$  is total number of edges connecting vertices inside the cluster of  $s$ , and  $d_s$  is the sum of the degrees of nodes of  $s$ , and  $m$  is the total number of edges in the network. The possible values for this criterion is in the range of  $[-0.5, 1]$  and for most real-networks this value is in the range of  $[0.3, 0.7]$ . Actually, values larger than 0.3, indicate a meaningful community structure.

## 4 Experimental Results

In this section, we study the effectiveness of our approach and compare the results obtained by PSO-Net w.r.t. the algorithms of GA-Net, FN and FC on the Girvan-Newman benchmark and then on real-world networks including the Zachary's Karate Club network, the American College Football network, the Bottlenose Dolphin network and the Books about US Politics network. Moreover, the proposed method was compared to three community detection methods which are listed below:

- *Fast Newman* (FN) [3] is an agglomerative hierarchical method which aims to maximizing modularity of obtained communities.
- *GA-Net* [19] is an optimization-based community detection method, which adopts Genetic Algorithm to optimize the community score measure.
- *Fuzzy Clustering* (FC) [5] is a community detection method based on fuzzy transitive rules. This method uses the edge centralities such as edge betweenness centrality to measure the similarity among nodes of a network. Then, by forming a



fuzzy relation on the network and applying transitive rules on the relation, when the relation achieve to the stable state, the clusters are discovered. In this study, we report the best results obtained by this method.

#### 4.1 Parameter Setting

The PSO-Net algorithm was implemented in visual studio 2010. The experiments have been performed on a computer having Intel® Core™ i5 CPU 2.67 GHz and 4 GB (3.9 GB usable) of memory. The number of generations for all data sets in both PSO-Net and GA-Net was set to 100. The population size is customized according to the size of data sets. In this way, size of population for karate club network is 100, for dolphin network is 200, and for football network, Political Books network and Girvan-Newman benchmark are set to 400. We used the following parameters for implementation of GA-Net: crossover rate of 0.8, mutation rate of 0.2, and tournament selection function. Since PSO-Net and GA-Net algorithms, are the random optimization methods, all the results obtained from these two methods are computed over 10 independent runs.

#### 4.2 Evaluation Metrics

In order to compare PSO-Net and other approaches, two measures, the normalized mutual information (NMI) [36] and Modularity [4], mentioned in Sect. 3.3, are used. NMI criterion is employed to measure the similarity between the real community structure of a network and the structure detected by the proposed method. Assume two different types of partitioning for a network  $A = \{A_1, \dots, A_R\}$  and  $B = \{B_1, \dots, B_D\}$ , that  $R$  and  $D$  are the number of communities in the partitioning  $A$  and  $B$ , respectively. A confusion matrix  $C$  is formed first, where an entry  $C_{ij}$  is the number of nodes that appear in both communities  $A_i \in A$  and  $B_j \in B$ . Then, normalized mutual information  $NMI(A, B)$  is defined as (2):

$$NMI(A, B) = \frac{-2 \sum_{i=1}^R \sum_{j=1}^D C_{ij} \log(C_{ij}N / C_i C_j)}{\sum_{i=1}^R C_i \log(C_i / N) + \sum_{j=1}^D C_j \log(C_j / N)} \quad (2)$$

where  $C_i$  ( $C_j$ ) is the sum of the elements of  $C$ , over row  $i$  (column  $j$ ), and  $N$  is the total number of nodes in the graph.  $NMI$  value of 1 indicates that  $A$  and  $B$  are exactly equal.

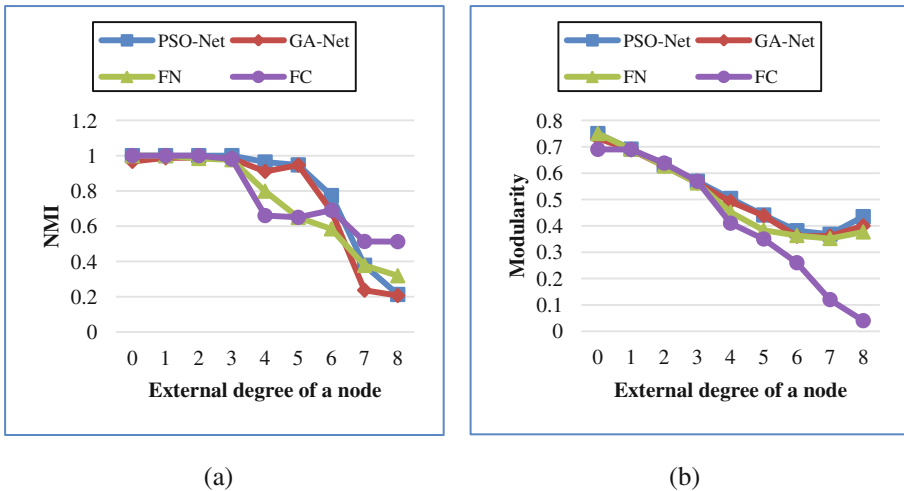
#### 4.3 Experimental Results in Synthetic Datasets

The most famous benchmark for community detection is the Girvan-Newman (GN) networks [1]. Each network has 128 nodes, divided into four communities of 32 nodes. The average degree of this type of networks is equal to 16. The nodes are connected together in a random order, but in such a way, that  $k_{in} + k_{out} = 16$ , which  $k_{in}$  and  $k_{out}$  are the internal and external degree of a node, respectively.

Increasing the value of  $k_{out}$  leads to more connections between the nodes of different communities, and therefore, the correct detection of communities becomes more difficult. Thereupon, in this case, the resulting graphs pose greater challenges to the

community mining methods. Figure 3(a) shows the average NMI value over 10 independent runs, obtained by each algorithms for different values of  $k_{out}$ . As can be seen, for the values of  $k_{out}$  less than seven, PSO-Net gets higher NMI value. When  $k_{out}$  is 7, performance of PSO-Net is worse than FC. For the  $k_{out}$  value of 8, GA-Net and PSO-Net obtain the least NMI value, respectively. It can be concluded that our approach has better performance in detecting communities of networks with more clear clusters.

Another measure that should be investigated is modularity. As can be seen in Fig. 3(b), for all values of  $k_{out}$ , the modularity for our proposed method is highest, which means that, the community structure resulted by PSO-Net is more modular than other three approaches. Similarly, in this case, the modularity results are obtained from an average of 10 runs.



**Fig. 3.** Comparison of PSO-Net, GA-Net, FN and FC in terms of (a) NMI and (b) Modularity on the Girvan-Newman benchmark.

In Table 1, the average number of clusters that each of the four algorithms returns over 10 run, is reported. As can be seen, our method for the values of  $k_{out}$  in the range of [0–4], divides the GN benchmark into 4 clusters which exactly is equal to true number of communities. For other values of  $k_{out}$ , PSO-Net detects the reasonable number of communities in comparison with other methods.

#### 4.4 Experimental Results in Real-World Datasets

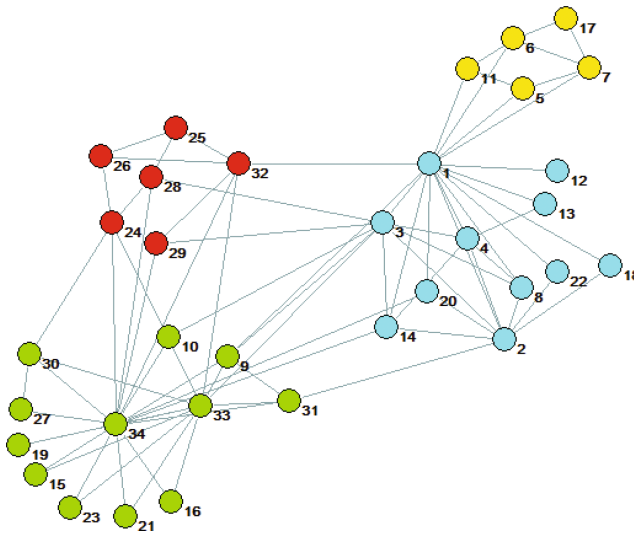
We now show the application of PSO-Net on two popular real-world networks, the *Zachary’s Karate Club*, and the *American College Football*, and compare our results with GA-Net, FN and FC methods.

Zachary’s Karate Club network, studied by Zachary, is a social network of friendships between 34 members of a karate club at a US university in 1970. During the

**Table 1.** Number of Communities detected by four methods on GN benchmark, for different values of  $k_{out}$

	0	1	2	3	4	5	6	7	8
PSO-Net	4	4	4	4	4	4.2	3.9	4.7	5.9
GA-Net	4.8	4.3	4.1	4.4	5.5	4.8	6.5	6.6	8.1
FN	4	4	4	4	3.6	3.1	3.2	3.5	3.6
FC	4	4	4	5	2	2	21	70	78

course of Zachary’s study, because of disagreements, the club divided in two groups about of the same size. And each of these two groups, are clustered in two subgroups. The community structure of this network is shown in Fig. 4.



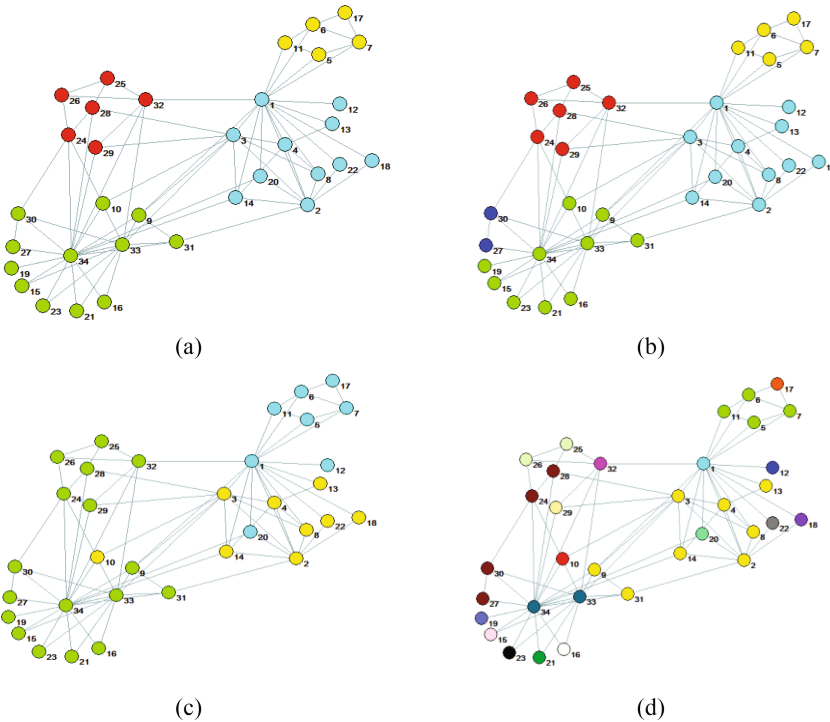
**Fig. 4.** Zachary’s karate club network

Table 2 shows the detailed comparative results of the various algorithms on the Karate network. For each algorithm, we have listed the NMI measure, modularity measure and then the number of communities. As can be seen, the average and best NMI values of PSO-Net are superior to that of other algorithms. GA-Net provides smaller standard deviation than PSO-Net, but the difference between these values, is negligible. Moreover the average and best Modularity values of our method, are higher than other algorithms. Also, standard deviation of our method for modularity is smaller than GA-Net. The column of average number of detected communities, shows that, except FC algorithm, other methods, provide the number of clusters that are near to the real one.

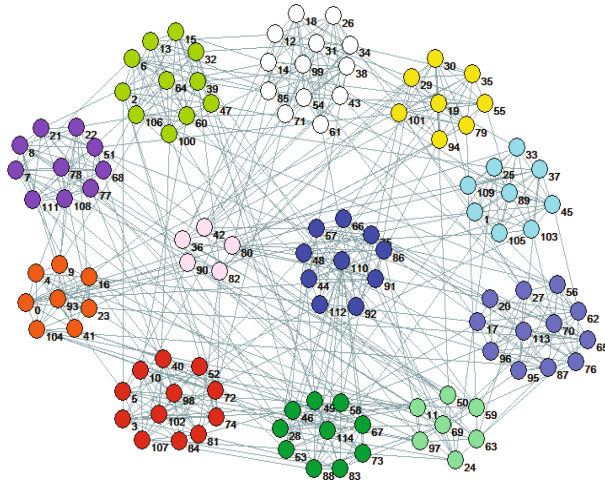
**Table 2.** Results obtained by four algorithms on Zachary’s Karate Club network.

Method	NMI				Modularity				Num. of Com.
	best	average	worst	std	best	average	worst	std	average
PSO-Net	<b>1</b>	<b>0.88</b>	0.60	0.100	<b>0.42</b>	<b>0.40</b>	0.37	<b>0.01</b>	3.7
GA-Net	0.94	0.80	<b>0.69</b>	0.096	0.40	0.37	0.29	0.03	4.2
FN	0.63	0.63	0.63	–	0.38	0.38	<b>0.38</b>	–	3
FC	0.56	0.56	0.56	–	0.13	0.13	0.13	–	19

It can be seen from Figs. 4 and 5 that, the detected community structure of our method on Zachary’s karate club network, is the real community structure. But the detected structure of other methods, are different from the true one.

**Fig. 5.** The detected communities of best result of (a) PSO-Net, (b) GA-Net, (c) FN and (d) FC on Zachary’s karate network

The American College Football network is a network with 115 nodes and 616 edges that grouped in 12 communities. The vertices represent teams and the edges indicate the season games between nodes in the year. The real communities of this network are shown in Fig. 6.



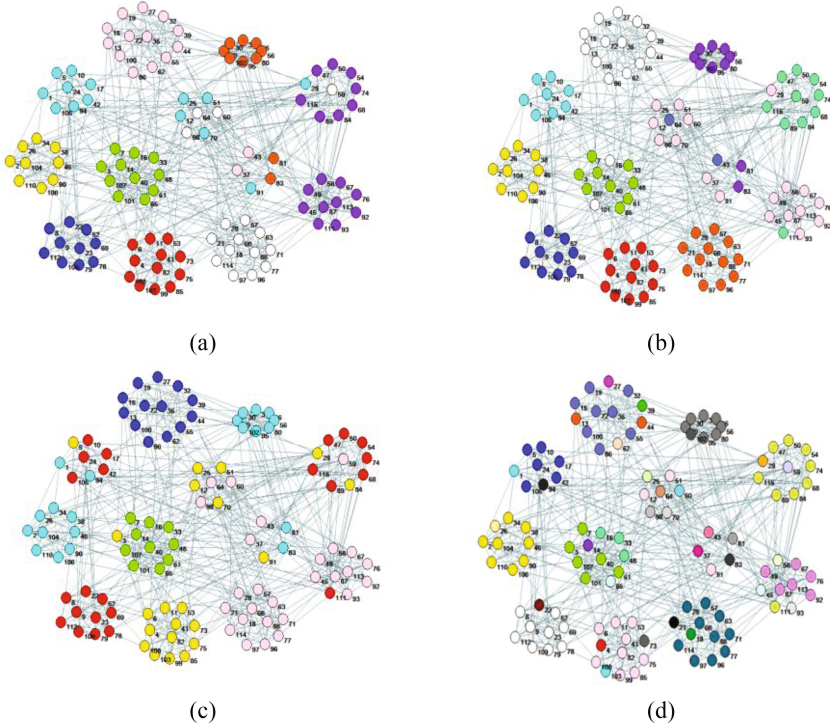
**Fig. 6.** American college football network

In Table 3, the results of four algorithms on this network are reported. As can be seen, PSO-Net has the highest average and the best NMI values after GA-Net. But standard deviation of our method is smaller than GA-Net. The modularity value for PSO-Net in three cases (best, average and worst) is the highest among all methods and the standard deviation of our method is smaller. GA-Net and PSO-Net, extract the closer number of clusters to real structure, respectively.

**Table 3.** Results obtained by the four algorithms on American College Football network.

Method	NMI				Modularity				Num. of Com.
	best	average	worst	std	best	average	worst	std	
PSO-Net	<b>0.89</b>	<b>0.82</b>	<b>0.80</b>	<b>0.033</b>	<b>0.52</b>	<b>0.52</b>	<b>0.51</b>	<b>0.003</b>	4
GA-Net	0.71	0.63	0.59	0.035	0.46	0.44	0.39	0.022	9.6
FN	0.70	0.70	0.70	–	0.49	0.49	0.49	–	4
FC	0.49	0.49	0.49	–	0.30	0.30	0.30	–	17

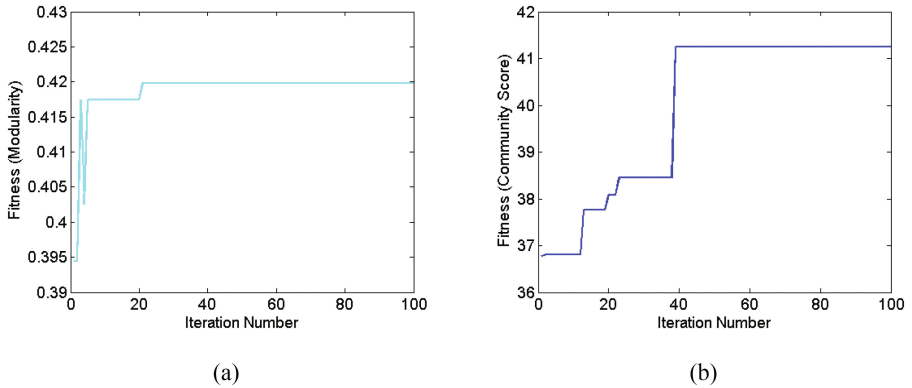
The best results of the four algorithms on football network are shown in Fig. 7. As can be seen, from Figs. 6 and 7, the community structure discovered by FC method, is very different from the true one. But, other approaches detect similar structure to real community structure on football network.



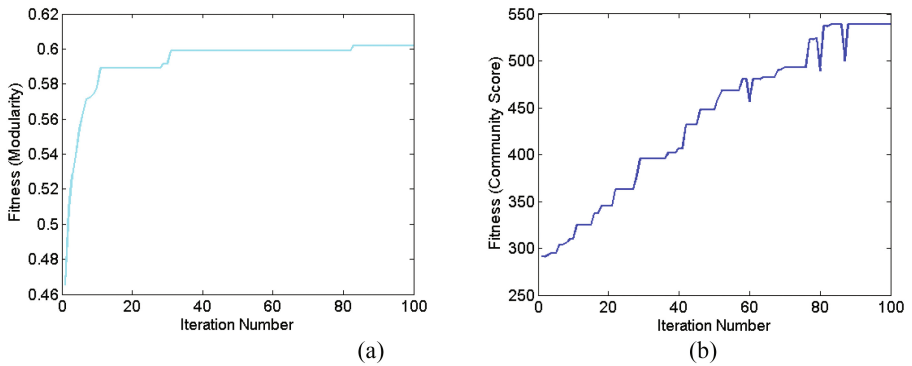
**Fig. 7.** Detected communities of best result of (a) PSO-Net, (b) GA-Net, (c) FN and (d) FC on Football network

## 5 Coverage Analysis for the Proposed Algorithm

In this Section, we investigate the convergence rate of our algorithm and another random optimization algorithm, i.e., GA-Net on real-world networks. It is worth noting that the fitness functions of two methods are different, and we just compare the convergence points in these methods. Figure 8(a) and (b) show the speed of convergence of GA-Net and PSO-Net, for karate club network, respectively. As can be seen, GA-Net in the iteration number of 39, achieves to maximum value of its objective function. However, PSO-Net converges in 21st iteration. That means, convergence rate of our method for karate network is better. It is worth mentioning that the NMI and Modularity of PSO-Net in discovering community structure of this network were largest among all methods. Figure 9(a) shows convergence rate of GA-Net for football network. As can be seen, this algorithm achieved to maximum value of fitness in 88<sup>th</sup> iteration. In Fig. 9(b), we can see that, PSO-Net converges in 83rd iteration for football network. Here, the difference is not significant.



**Fig. 8.** Comparison of convergence rate of (a) PSO-Net and (b) GA-Net in karate club network



**Fig. 9.** Comparison between convergence rate of (a) PSO-Net (b) GA-Net on American College Football network

## 6 Conclusion

In this paper, a novel community detection method based on particle swarm optimization (PSO) algorithm named PSO-Net has been proposed. We focus on the modification of the PSO. In our method, the particles for approaching to their local and the global best, take part in crossover operation with them. Then, for spreading search space, a mutation operator is performed on each particle. The algorithm takes modularity measure as its fitness function. Experiments on synthetic and real world networks showed that PSO-Net has good results in discovering communities of these networks, especially, in karate club network. Moreover, the convergence rate of PSO-Net in comparison with GA-Net is very faster. In the future, we will aim to applying multi-objective optimization to improve quality results.

## References

1. Girvan, M., Newman, M.E.: Community structure in social and biological networks. *Proc. Natl. Acad. Sci.* **99**, 7821–7826 (2002)
2. Radicchi, F., Castellano, C., Cecconi, F., Loreto, V., Parisi, D.: Defining and identifying communities in networks. *Proc. Natl. Acad. Sci. U.S.A.* **101**, 2658–2663 (2004)
3. Newman, M.E.: Fast algorithm for detecting community structure in networks. *Phys. Rev. E* **69**, 066133 (2004)
4. Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. *Phys. Rev. E* **69**, 026113 (2004)
5. Sun, P.G.: Community detection by fuzzy clustering. *Phys. A* **419**, 408–416 (2015)
6. Sun, P.G., Yang, Y.: Methods to find community based on edge centrality. *Phys. A* **392**, 1977–1988 (2013)
7. Fortunato, S.: Community detection in graphs. *Phys. Rep.* **486**, 75–174 (2010)
8. Du, H., Feldman, M.W., Li, S., Jin, X.: An algorithm for detecting community structure of social networks based on prior knowledge and modularity. *Complexity* **12**, 53–60 (2007)
9. Fortunato, S., Latora, V., Marchiori, M.: Method to find community structures based on information centrality. *Phys. Rev. E* **70**, 056104 (2004)
10. Sun, P.G.: Weighting links based on edge centrality for community detection. *Phys. A Stat. Mech. Appl.* **394**, 346–357 (2014)
11. Carrasco, J.J., Fain, D.C., Lang, K.J., Zhukov, L.: Clustering of bipartite advertiser-keyword graph
12. Amiri, B., Hossain, L., Crawford, J.W., Wigand, R.T.: Community detection in complex networks: multi-objective enhanced firefly algorithm. *Knowl. Based Syst.* **46**, 1–11 (2013)
13. Duch, J., Arenas, A.: Community detection in complex networks using extremal optimization. *Phys. Rev. E* **72**, 027104 (2005)
14. Gong, M., Cai, Q., Chen, X., Ma, L.: Complex network clustering by multiobjective discrete particle swarm optimization based on decomposition. *IEEE Trans. Evol. Comput.* **18**, 82–97 (2014)
15. Gong, M., Ma, L., Zhang, Q., Jiao, L.: Community detection in networks by using multiobjective evolutionary algorithm with decomposition. *Phys. A Stat. Mech. Appl.* **391**, 4050–4060 (2012)
16. Ji, J., Song, X., Liu, C., Zhang, X.: Ant colony clustering with fitness perception and pheromone diffusion for community detection in complex networks. *Phys. A Stat. Mech. Appl.* **392**, 3260–3272 (2013)
17. Li, S., Chen, Y., Du, H., Feldman, M.W.: A genetic algorithm with local search strategy for improved detection of community structure. *Complexity* **15**, 53–60 (2010)
18. Liu, C., Liu, J., Jiang, Z.: A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks. *IEEE Trans. Cybern.* **44**, 2274–2287 (2014)
19. Pizzuti, C.: GA-Net: a genetic algorithm for community detection in social networks. In: Rudolph, G., Jansen, T., Beume, N., Lucas, S., Poloni, C. (eds.) PPSN 2008. LNCS, vol. 5199, pp. 1081–1090. Springer, Heidelberg (2008). [https://doi.org/10.1007/978-3-540-87700-4\\_107](https://doi.org/10.1007/978-3-540-87700-4_107)
20. Pizzuti, C.: A multiobjective genetic algorithm to find communities in complex networks. *IEEE Trans. Evol. Comput.* **16**, 418–430 (2012)
21. Qu, J.: A hybrid algorithm for community detection using PSO and EO. *Adv. Inf. Sci. Serv. Sci.* **5**, 187 (2013)



22. Donetti, L., Munoz, M.A.: Detecting network communities: a new systematic and efficient algorithm. *J. Stat. Mech. Theory Exp.* **2004**, P10012 (2004)
23. Clauset, A., Newman, M.E., Moore, C.: Finding community structure in very large networks. *Phys. Rev. E* **70**, 066111 (2004)
24. Shi, J., Malik, J.: Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**, 888–905 (2000)
25. Hagen, L., Kahng, A.B.: New spectral methods for ratio cut partitioning and clustering. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **11**, 1074–1085 (1992)
26. Gog, A., Dumitrescu, D., Hirsbrunner, B.: Community detection in complex networks using collaborative evolutionary algorithms. In: Almeida e Costa, F., Rocha, L.M., Costa, E., Harvey, I., Coutinho, A. (eds.) *ECAL 2007. LNCS (LNAI)*, vol. 4648, pp. 886–894. Springer, Heidelberg (2007). [https://doi.org/10.1007/978-3-540-74913-4\\_89](https://doi.org/10.1007/978-3-540-74913-4_89)
27. Shang, R., Bai, J., Jiao, L., Jin, C.: Community detection based on modularity and an improved genetic algorithm. *Phys. A Stat. Mech. Appl.* **392**, 1215–1231 (2013)
28. Abramson, D., Abela, J.: A parallel genetic algorithm for solving the school timetabling problem (1991)
29. Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. In: *Proceedings of the Sixth International Symposium on Micro Machine and Human Science, MHS 1995, (IEEE 1995)*, pp. 39–43 (1995)
30. Scott, J.: *Social Network Analysis*. Sage (2012)
31. Freeman, L.C.: A set of measures of centrality based on betweenness. *Sociometry* **40**, 35–41 (1977)
32. Latora, V., Marchiori, M.: A measure of centrality based on the network efficiency. arxiv: con-math 0402050 (2004)
33. Brandes, U., et al.: On modularity clustering. *IEEE Trans. Knowl. Data Eng.* **20**, 172–188 (2008)
34. Liu, Y., Luo, J., Yang, H., Liu, L.: Finding closely communicating community based on ant colony clustering model. In: *2010 International Conference on Artificial Intelligence and Computational Intelligence (AICI), (IEEE 2010)*, pp. 127–131 (2010)
35. Handl, J., Knowles, J.: An evolutionary approach to multiobjective clustering. *IEEE Trans. Evol. Comput.* **11**, 56–76 (2007)
36. Danon, L., Diaz-Guilera, A., Duch, J., Arenas, A.: Comparing community structure identification. *J. Stat. Mech. Theory Exp.* **2005**, P09008 (2005)



# Mouse Tracking Measures and Movement Patterns with Application for Online Surveys

Catia Cepeda<sup>1,2</sup>(✉), Joao Rodrigues<sup>2</sup>, Maria Camila Dias<sup>2</sup>, Diogo Oliveira<sup>2</sup>,  
Dina Rindlisbacher<sup>1,3</sup>, Marcus Cheetham<sup>1,3</sup>, and Hugo Gamboa<sup>2</sup>

<sup>1</sup> Department of Internal Medicine, University Hospital Zurich, Zurich, Switzerland

<sup>2</sup> LIBPhys (Laboratory for Instrumentation, Biomedical Engineering and Radiation Physics), Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Caparica, Portugal

`c.cepada@campus.fct.unl.pt`

<sup>3</sup> University Research Priority Program “Dynamics and Healthy Aging”, University of Zurich, Zurich, Switzerland

**Abstract.** There is growing interest in the field of human-computer interaction in the use of mouse movement data to infer e.g. user’s interests, preferences and personality. Previous work has defined various patterns of mouse movement behavior. However, there is a paucity of mouse tracking measures and defined movement patterns for use in the specific context of data collection with online surveys. The present study aimed to define and visualize patterns of mouse movements while the user provided responses in a survey (with questions to be answered using a 5-point Likert response scale). The study produced a wide range of different patterns, including new patterns, and showed that these can easily be distinguished. The identified patterns may - in conjunction with machine learning algorithms - be used for further investigation toward e.g. the recognition of the user’s state of mind or for user studies.

**Keywords:** Knowledge extraction · Mouse behavior patterns  
Mouse tracking · Human-computer interaction · User · Survey

## 1 Introduction

A multitude of human factors influences human-computer interaction (HCI) (e.g., [18]). The influence on HCI of individually stable patterns of thinking, feeling and behavior is of longstanding interest (e.g., [9, 21]), as this often reflects underlying interests, preferences and personality. Making decisions is a complex cognitive and affective process [8, 13]. Understanding user behavior in the context of decision-making has increasingly attracted attention in HCI research [10, 19].

Pointer tracking refers to the recording of users’ mouse cursor positions, used, for example, to capture the mouse movement trajectories for the purpose

of further analysis. Data acquisition of mouse cursor positions has the advantage of being cheap, easy to implement and is already integrated in the use of the computer.

The present study aimed to identify patterns of mouse movements while the users give input in an online survey. These mouse movement patterns are potentially relevant as a means to understanding the user, such as in terms of the user's patterns of decision uncertainty.

Given the relative paucity of mouse tracking measures and mouse movement patterns in the literature, we present a new set of mouse behavior patterns that could potentially be combined with machine learning algorithms as a means to capturing information [14] about stable patterns of thinking, feeling and behavior of the user.

## 1.1 Related Work

Eye tracking systems are used in HCI research since mid-1970s [22]. The data structure is similar to that of mouse movements (x and y positions in screen over time). In fact, a wide range of eye movement behaviors have been associated with mouse movements behaviors. There is also multimodal data acquisition devices available, such as *Tobii* and *SensorMotoric Instruments* (SMI) systems, that allow concurrent measurement of eye and mouse movement behavior.

For instance, *Tobii* permits eye tracking and analysis of eye sampling behavior while the user observes and interacts with web pages [4]. This system also enables concurrent acquisition of video, sound, key-strokes and mouse clicks. Analyses include a range of measures such as mouse movement velocity, and can visualize results using various methods, such as heat maps. The analyses of different modalities may also be combined in order to assess, for example, the time from the first fixation to a particular target until the user clicks on the same target (or the number of clicks on the target).

SMI [2] also provides behavioral and gaze analysis software for research in the fields of reading research, psychology, cognitive neuroscience, marketing research and usability testing. While this system only processes eye and head tracking data, it has the advantage of allowing the analyzes of several subjects simultaneously. This permits analysis, for example, of the hit ratio, that is the relative number of subjects in the sample that fixated at least once on the target.

Although eye tracking systems have a comparatively long history, the field of mouse tracking had developed several interesting approaches for mouse movements analysis. This largely relates to web pages usability testing in order to improve the user experience [1, 3, 5–7, 15], but others extract data from the mouse coordinates, such as path distance, time measures and mouse clicks in order to study user's behavior rather than the web design itself.

For instance, *Revett et al.* and *Hugo et al.* [11, 23] propose the biometric identification of the user based only on mouse or pointer movements. Another approach, led by *Khan et al.* [17], related the mouse behavior patterns with personality. In *Pimenta et al.* mental fatigue has been detected by means of mouse movements [20], while *Hibbel et al.* related movements to emotions [12, 26].

Other measures and movements patterns have also been used in behavior studies. In 2006, *Arroyo et al.* described mouse behaviors in the context of web-sites, reporting user behavior that consists of a long pause next to text or a blank space, followed by a fast and direct movement towards a link [6]. *Arroyo et al.* also examined hesitation patterns and random movements, while *Huang et al.* compared clicks and hover distributions, unclicked hovers and abandonments [15].

*Seelye et al.* used the deviation of the movement in relation to a straight path and the time between the two targets to distinguish older adults with and without mild cognitive impairment (MCI). They found that more curved or looped mouse movements and less consistency over time are more closely correlated with MCI subjects [24].

*Yamauchi et al.* focused on two trajectory measures from mouse cursor to detect user emotions. They defined attraction as the area under the curve from the starting position to the end position and zigzag as the number of direction changes during the movement. A statistical model build with these trajectory measures could predict nearly 10%–20% of the variance of positive affect and attentiveness ratings [25].

*Arapakis et al.* used a large number of measures to predict user engagement, as indicated by, for example, attention and usefulness [5]. The set of measures included the most common distance and time measures and also measures related to the target, for instance, the number of movements toward and away from the target, or the number of hovers over the target compared with around the target.

More recently, *Katerina et al.* used a wide set of measures, including mouse and keyboard measures [16]. Their objective was to examine the relationship between the measures extracted from mouse and keystroke and end-user behavioral measures. Two examples of measures examined in terms of mouse movements are the number of mouse long pauses and the number of clicks in the end of direct mouse movements. From keystroke dynamics one example of a measure done was the time elapsed between key press and key release.

To the best of our knowledge, no previous studies have reported mouse movements during data collection using online surveys.

## 2 Study Design

### 2.1 Participants and Procedure

N = 119 volunteers recruited via a pool of test participants and students of the University of Zurich and of the ETH Zurich participated in this study. The participants were between 20 and 52 years old ( $M = 25.4$ ;  $SD = 5.4$ ; 18 male). All participants were native or fluent speakers of Standard German. Written informed consent was obtained before participation, according to the guidelines of the Declaration of Helsinki.

## 2.2 Data Acquisition Architecture

In this study, the data resulted from the interaction of the user with the web browser while completing an online survey, which was programmed to send the data to a server machine via AJAX, where it is finally recorded as a file in a data base.

The results of the survey are also saved on the database, although in this case via the Survey Management System using *PHP*. Therefore, if needed, these results could be accessed as well (Fig. 1).

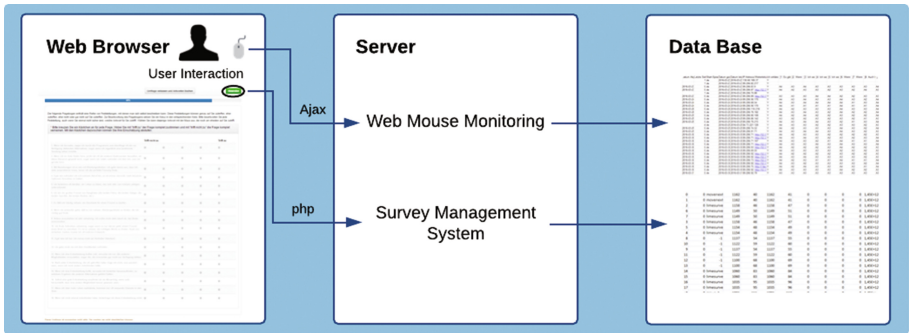


Fig. 1. Architecture.

## 2.3 Data Collection

The pointer movement is recorded by a server, which creates a report file with relevant recorded data: frame number; event type (represented by 0 when a movement is verified and 1 when the mouse button is pressed down); question number if hovered; answer number if hovered; x and y screen’s position (in pixels) and time stamp. The name of the file includes the IP address, the survey ID and the step of the questionnaire.

The online survey is constructed using a freely available software survey tool on the web. The online survey presents a sequence of statements and the answers are 5-point Likert-type scale. The results from the survey could be returned to a csv file.

## 2.4 Data Cleaning

To ensure correct formatting and processing of data from the server file, a validation procedure is applied as a first step. This validation procedure ignores data acquired with touch screen devices, reorder the data by time, join different files from the same questionnaire and detects how many samples are lost.

### 3 Behavioral Patterns Description

The data acquired with the LimeSurvey contains information about the mouse position with and without scroll in pixels. This data is first interpolated with equal time interval between samples in order to retrieve the correct information from it. With the mouse position pre-processed and the other information delivered by the LimeSurvey, several measures from temporal, spatial and contextual domains can be derived.

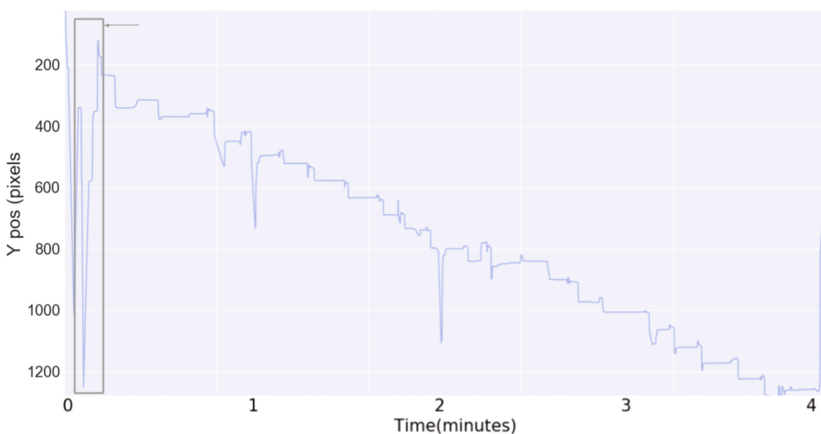
In this study, these measures are essential to compute several of the behavioral patterns described further.

#### 3.1 Overview Pattern

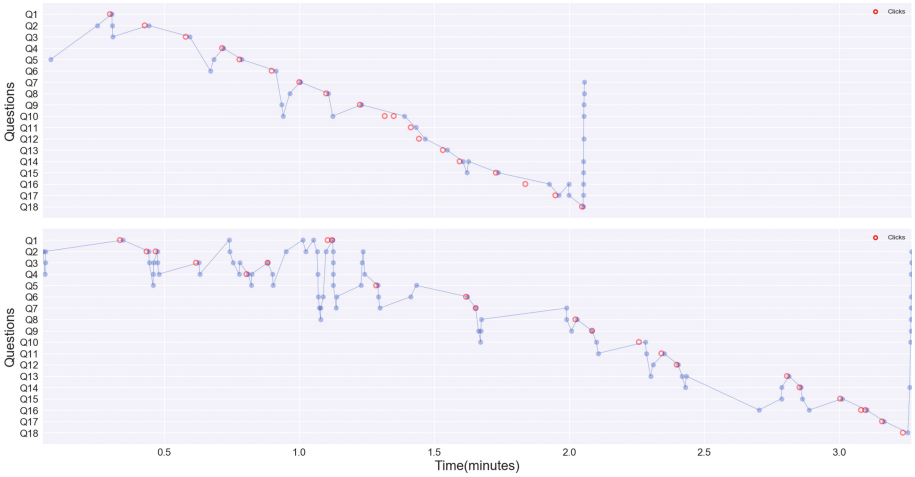
A behavior that can be found in some subjects in participating/answering surveys regards getting an overall idea of the number of questions, the length of the survey or the types of questions. This behavior is characterized by, at the beginning of the survey, scrolling the cursor over a wide area in direction to the bottom of the survey getting an overview of it. In Fig. 2 it is represented the mouse y coordinate represented over time, which makes it easy to observe this behavior. The first question are at the top of the plot (small y values) and, moving forward through the next questions, the y increases. At the beginning of the questionnaire, this subject goes to the end of the survey and then comes back to the first questions. This behavior also occurs after one minute and two minutes of interaction, but never so far as the first time.

#### 3.2 Fast Decision Pattern

While some people take a long time to answer the questions, others are very fast. It is possible to find both behaviors, that we call Fast Decision Patterns, which



**Fig. 2.** Representation of the y-axis mouse movement over time. The rectangle area corresponds to an overview pattern.



**Fig. 3.** Representation of the questions where the mouse is located over time for two different subjects. The subject at the top takes around two minutes to answer the questionnaire while the subject at the bottom almost four minutes.

are represented in Fig. 3. Both plots represent the question where the mouse is located over time and, as it is possible to observe, the subject at the top is much faster than the bottom subject, taking one and a half less minutes to answer the same questionnaire.

The work of Arroyo et al. [6] analyzed fast movements towards a target.

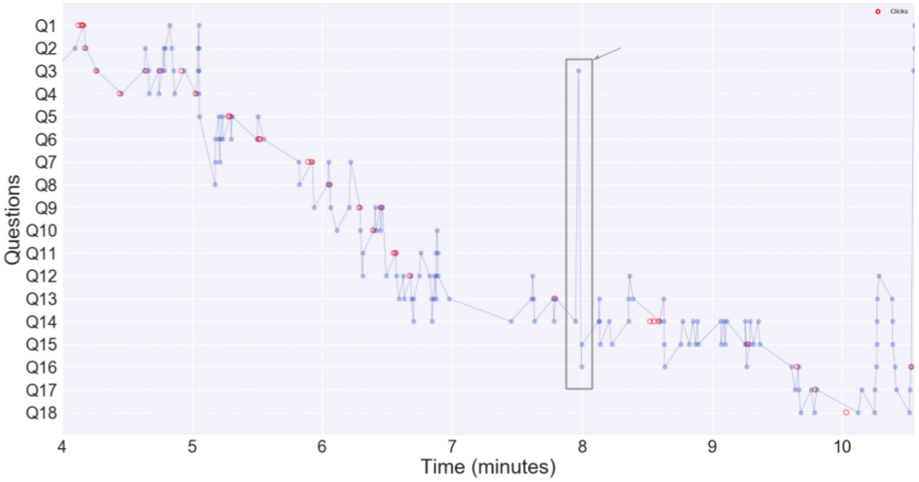
### 3.3 Revisit Pattern

A typical behavior of the subject that can be found in the survey context, is to revisit prior questions after some time of having answered. In Fig. 4 the user has revisited a prior answer (from question 14 to question 3) which was at the top of the survey. Interestingly, after answering the first time to the question 3, this subject responded to question 4 and came back to question 3, having changed three times the option previously answered. The revisit was around three minutes after these changes.

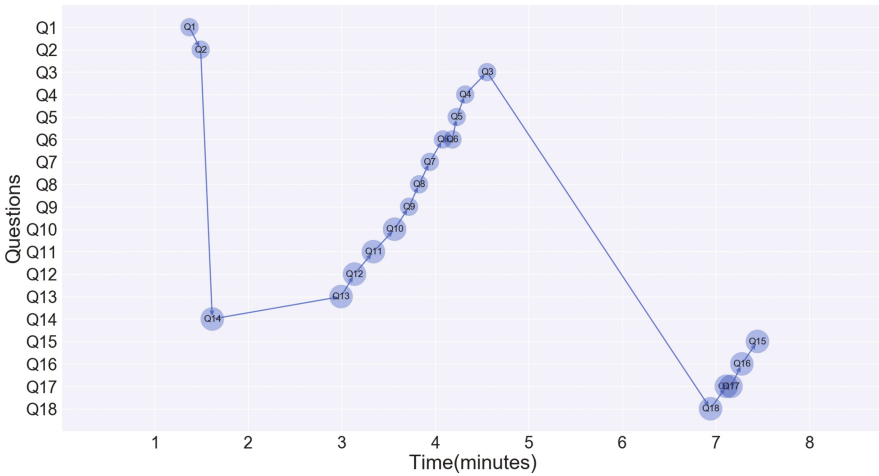
The analysis done by SMI [2] considers a similar metric with eye movements for a group evaluation: the average number of glances into the target.

### 3.4 Skips Pattern

When answering the survey some subjects would not have a linear behavior of following the natural order of questions. In fact, some subjects would skip questions and answer in an unnatural order. In Fig. 5, it is represented the questions answered over time. It is observed that the user does not take a linear approach in completing the survey, after answering question two, the subject starts to answer from question 14 to the previous questions. When the user is back to question 3, goes again to the end and answer question 18 until question 15.



**Fig. 4.** Representation of the questions where the mouse is located over time. The red circles represent the mouse clicks. The rectangle area corresponds to a revisit behavior. (Color figure online)

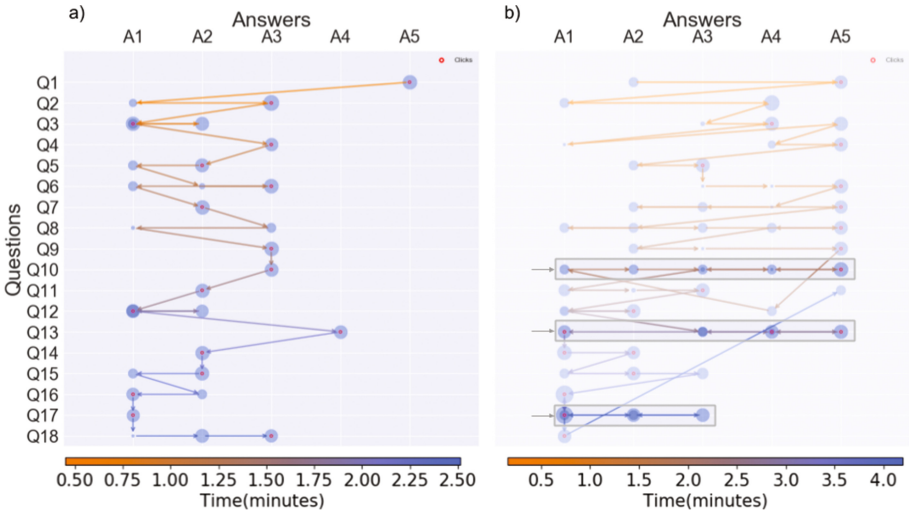


**Fig. 5.** Representation of the questions answered over time. This user is an example of skips behavior.

### 3.5 Hover Pattern

In the context of the survey, a typical behavior found on certain users is hovering multiple available options before selecting their final answer. In Fig. 6, two different users are compared in their survey completion. The flow chart indicates the way each user behave by indicating in which options they kept their mouse. Each blue circle is a selectable option to answer the corresponding question,





**Fig. 6.** Chart flow of two different users in answering the survey. The y-axis represents the question number and the x-axis is the option answer number. Clicks are depicted as red circles. The color-bar shows the flow of time used by the arrows that point the flow of the user’s behavior. (a) The user has a less representative hovering behavior. (b) The highlighted areas show the hovering behavior of the user. (Color figure online)

which the user hovered. The size of the circle is proportional to the time spent on that option.

As can be seen on Fig. 6, the user on the right (b) has more hovered areas (specially highlighted areas) than the user represented on the left (a).

Although *Tobii* [4] is an eye tracking system, it considers the number of fixations before fixating on the target, which is similar to what we are suggesting here. Previous studies also includes hover patterns in mouse movements analysis. *Katerina et al.* [16] considered the number of mouse hovers that turned into mouse clicks and *Arapakis et al.* [5] compared between hovering the area of interest in relation to other areas. Also *Huang et al.* [15] analyzed the hover distributions and clicks to verify the number of search results hovered before the user clicks.

### 3.6 Hover Reading Pattern

During the completion of questionnaires, the questions have some text in the left border which can be read in several ways. We found two distinct patterns: some people move the mouse to the text area, while reading the question, while others just move the mouse around the answers area. One example of each behavior are shown in Figs. 7 and 8, for the first it is evident that for each item the subject is hovering the text of the question before choosing an answer. That is not verified in the second, that only moves the mouse around the answers.

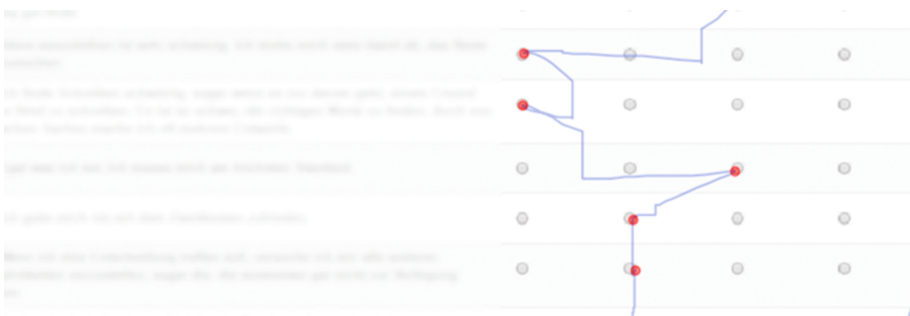
The computational process of this behavior is quite easy, the survey software has a tool in which the width of text of the question can be defined. Knowing that, the x mouse coordinates can be associated to questions or answers area.



**Fig. 7.** Representation of the mouse movements (in blue) of a subjects that moves the mouse to the question text while reading it. The mouse clicks are represented by a red circle. (Color figure online)

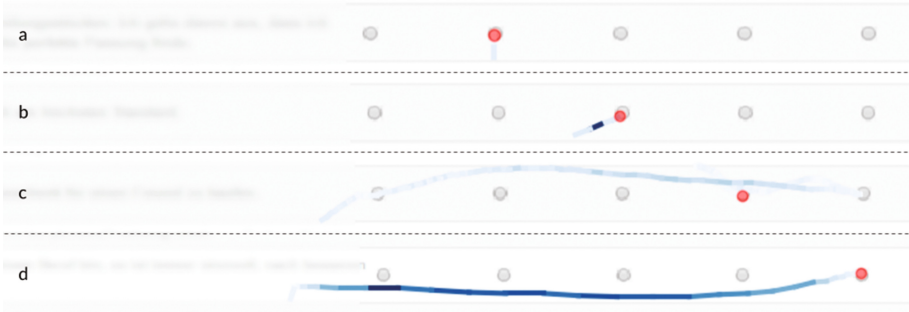
### 3.7 Inter Item Pattern

The distance and time taken between the answered choice and enter the next question could be different from person to person. The time and distance are highly correlated and define the same kind of behaviors. However, some more specific patterns can be highlighted, for instance, the subject can take more time because it was moving slower, or because it was moving a lot, even if quickly. Therefore it is important to individualize these measures. In Fig. 9 it is presented four possible behaviors. Considering that the color intensity depends on the velocity (more intense for higher speeds), the (a) and (b) present short distance inter items, being (b) much faster than (a), while (c) and (d) present



**Fig. 8.** Representation of the mouse movements (in blue) of a subjects that keeps the mouse around the answers area, even when reading the question. The mouse clicks are represented by a red circle. (Color figure online)

long distances inter item, being (d) much faster than (c). Here although (a) and (c) have very different distances, the speed of movement is similar. The same is true to (b) and (d).

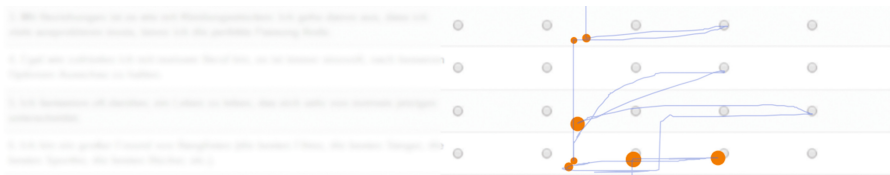


**Fig. 9.** Representation of the mouse movement in the survey context considering only the inter item interval. The color of the line corresponds to the velocity of the movement (color more intense for higher velocity). In (a) there is an example of short distance but low speed, in (b) short distance and high speed, in (c) long distance and low speed and (d) long distance and high speed. (Color figure online)

### 3.8 Long Pauses Pattern

Long pauses correspond to mouse movements at the same place (x and y coordinates) for a long period of time. This can be observed in Fig. 10 in which orange circles represent long pauses while answering the survey questions. The longer the pauses, the larger the circles.

Multiple studies considered the number and time of long pauses [6,16,24].

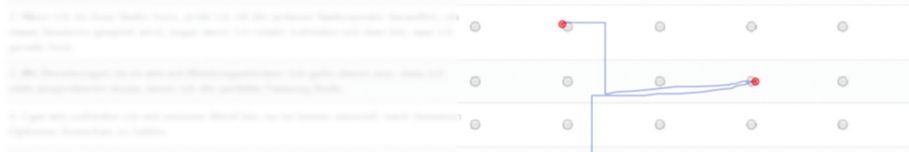


**Fig. 10.** Representation of long pauses pattern in mouse movement. In orange are presented circles that are larger according to the time paused. (Color figure online)

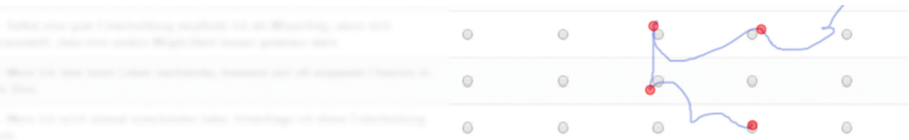
### 3.9 Straight and Curvy Pattern

Straight patterns are characterized by a direct or straight line in direction to a target. This pattern indicates that a target has been spotted and the subject decided to move the cursor towards it. The opposite behavior is the curvy pattern, characterized by more curved movements. Comparing Fig. 11 with Fig. 12 it is possible to detect a huge differences in the way they move the mouse.

The studies from *Katerina et al.* [16] and *Seelye et al.* [24] had these patterns into consideration, having compared more straight or curved movements.



**Fig. 11.** Representation of straight patterns with the mouse. The red circles correspond to mouse clicks. (Color figure online)

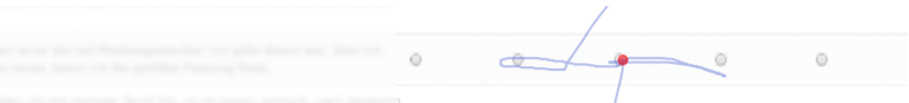


**Fig. 12.** Representation of curvy patterns with the mouse. The red circles correspond to mouse clicks. (Color figure online)

### 3.10 <-turn Pattern

While making a decision, sometimes the mouse movement nearly inverts its direction, this pattern has been called <-turn. Figure 13 presents this behavior two times during the choice of a question answer. To compute this behavior it should be detected angles close to 180 degrees in change of direction within the same item.

*Yamauchi et al.* [25] analyzed a similar pattern considering direction change.



**Fig. 13.** Representation of <-turn pattern. In blue is presented the mouse movement and in red the mouse click. (Color figure online)

### 3.11 Random Movements

While some movements are spontaneous and have an inner purpose, others might just be unconscious and have no specific intention. The latter patterns are described as random movement patterns and are characterized by a large number of movements confined in a non-interest area for a short time, as shown in Fig. 14.

This behavior was briefly described by [6], however they do not present a visualization example or a way to compute random movements.



Fig. 14. Representation of a random movement made by the mouse cursor.

### 3.12 Loop Pattern

In the category of random movements, a pattern that can be found is characterized by a turn of more than 360°, which can be defined as a loop and observed in Fig. 15. This behavior was previously considered by *Seelye et al.* [24] that calculated the number of looped mouse movements.



Fig. 15. Representation of a loop pattern.

## 4 Conclusion

This study demonstrates the use of mouse tracking measures and movement patterns in the specific context of online survey-based data collection. The survey consisted of several questions, each to be answered using a 5-point Likert response scale. Using only the mouse movements data, we show that it is possible to extract a wide range of different behaviors. The results also show the behavior patterns can easily be distinguished by mere visual inspection.

Although some of the behavioral patterns have already been reported in other studies (e.g., [6, 16, 24]), none were used in the context of surveys. Given that

this is a completely different task situation with different task requirements, the proposed patterns require a different interpretation. This work delivered also new patterns of movement that were not reported in the previous literature, contributing therefore to the current state of the art.

It is possible to group several of these patterns according to their potential explanation. There are patterns that might be associated with personality traits, decision confidence or decision difficulty, but this awaits further investigation. For example, overview, fast decision, skips, straight and curvy, inter items intervals and long pauses could indicate personal characteristics and some users would follow the questions in an orderly and sequential manner, while others would first get an overall picture of the survey questions and then answer (overview pattern). Fast decisions could be related to confidence, and decision difficulty could be associated with hover pattern and <-turn.

Concerning the hover reading pattern, the users that move the mouse to the question text while reading it are less goal-oriented than those who just move the cursor directly to the next question. Whether this is so requires further investigation. If this is the case, it is also possible the first group of users could reveal a higher correlation between mouse and eye movements.

## 5 Future Work

As a first step after this work, it would be interesting to create metrics that express each of the patterns extracted. Consistent with other studies, we will progress in order to apply machine learning techniques to infer personality and states of mind from mouse movements data.

The recognition of these patterns in more complex contexts could be applied to improve the usability of websites and create an adapted design and contents according with user preferences.

Another application area is clinical/ergonomics field, for example to recognize mental fatigue or even mental diseases by studying the cognitive state of the subject given that users' state of mind could be directly associated with a conjunction of behaviors.

## References

1. Clicktale. <https://www.clicktale.com/>
2. Eye Tracking Solutions by SMI. <https://www.smivision.com/>
3. Inspectlet - Website Heatmaps, Session Recording, Form Analytics. <https://www.inspectlet.com/>
4. Eye tracking technology for research - Tobii Pro (2015). <https://www.tobiipro.com/>
5. Arapakis, I., Leiva, L.A.: Predicting user engagement with direct displays using mouse cursor information. In: Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval - SIGIR 2016, pp. 599–608. ACM Press, New York (2016). <https://doi.org/10.1145/2911451.2911505>, <http://dl.acm.org/citation.cfm?doid=2911451.2911505>

6. Arroyo, E., Selker, T., Wei, W.: Usability tool for analysis of web designs using mouse tracks. In: CHI 2006 extended abstracts on Human factors in computing systems - CHI EA 2006, p. 484. ACM Press, New York (2006). 10.1145/1125451.1125557, <http://dl.acm.org/citation.cfm?doid=1125451.1125557>
7. Atterer, R., Wnuk, M., Schmidt, A.: Knowing the user's every move. In: Proceedings of the 15th International Conference on World Wide Web - WWW 2006, p. 203. ACM Press, New York (2006). <https://doi.org/10.1145/1135777.1135811>, <http://portal.acm.org/citation.cfm?doid=1135777.1135811>
8. Damasio, A.R.: *Descartes' error : emotion, reason, and the human brain*. New York (1994)
9. Dillon, A., Watson, C.: User analysis in HCI - The historical lessons from individual differences research. *Int. J. Hum. Comput. Stud.* **45**(6), 619–637 (1996). <https://doi.org/10.1006/ijhc.1996.0071>, <https://www.sciencedirect.com/science/article/pii/S1071581996900713>
10. Djamasbi, S., Tulu, B., Loiacono, E., Shitefleet-Smith, J.: Can a reasonable time limit improve the effective usage of a computerized decision aid? *Commun. Assoc. Inf. Syst.* **23**(22), 393–408 (2008). <http://digitalcommons.wpi.edu/uxdmrl-pubs>, <http://digitalcommons.wpi.edu/uxdmrl-pubs/27>, <http://aisel.aisnet.org/cais/vol23/iss1/22>
11. Gamboa, H.F.S.: Multi-modal behavioral biometrics based on HCI and electrophysiology, 1–216, April 2008. <http://www.lx.it.pt/~afred/pub/thesisHugoGamboa.pdf>
12. Hibbeln, M., Jenkins, J.L., Schneider, C., Valacich, J.S., Weinmann, M.: How is your user feeling? inferring emotion through human-computer interaction devices. *MIS Q.* **41**(1), 1–21 (2017). <https://doi.org/10.25300/MISQ/2017/41.1.01>
13. Hodgkinson, G.P., Bown, N.J., Maule, A.J., Glaister, K.W., Pearman, A.D.: Breaking the frame: an analysis of strategic cognition and decision making under uncertainty. *Strateg. Manag. J.* **20**(10), 977–985 (1999). [https://doi.org/10.1002/\(SICI\)1097-0266\(199910\)20:10<977::AID-SMJ58>3.0.CO;2-X](https://doi.org/10.1002/(SICI)1097-0266(199910)20:10<977::AID-SMJ58>3.0.CO;2-X)
14. Holzinger, A.: Human-Computer Interaction and Knowledge Discovery (HCI-KDD): what is the benefit of bringing those two fields to work together? In: Cuzzocrea, A., Kittl, C., Simos, D.E., Weippl, E., Xu, L. (eds.) *CD-ARES 2013*. LNCS, vol. 8127, pp. 319–328. Springer, Heidelberg (2013). [https://doi.org/10.1007/978-3-642-40511-2\\_22](https://doi.org/10.1007/978-3-642-40511-2_22)
15. Huang, J., White, R.W., Dumais, S.: No clicks, no problem: Using cursor movements to understand and improve search (2011). <https://doi.org/10.1145/1978942.1979125>
16. Katerina, T., Nicolaos, P.: Mouse behavioral patterns and keystroke dynamics in End-User Development: What can they tell us about users' behavioral attributes? *Comput. Hum. Behav.* **83**, 288–305 (2018). <https://doi.org/10.1016/j.chb.2018.02.012>, <https://www.sciencedirect.com/science/article/pii/S0747563218300700>
17. Khan, I.A., Brinkman, W.P., Fine, N., Hierons, R.M.: Measuring personality from keyboard and mouse use. In: Proceedings of the 15th European Conference on Cognitive Ergonomics the Ergonomics of Cool Interaction - ECCE 2008, p. 1 (2008). <https://doi.org/10.1145/1473018.1473066>
18. Olson, G.M., Olson, J.S.: Human-computer interaction: psychological aspects of the human use of computing. *Ann. Rev. Psychol.* **54**(1), 491–516 (2003). <https://doi.org/10.1146/annurev.psych.54.101601.145044>

19. Payne, J.W., Bettman, J.R., Johnson, E.J.: Adaptive strategy selection in decision making. *J. Exp. Psychol. Learn. Memory Cogn.* **14**(3), 534–552 (1988). <https://doi.org/10.1037/0278-7393.14.3.534>, <http://doi.apa.org/getdoi.cfm?doi=10.1037/0278-7393.14.3.534>
20. Pimenta, A., Carneiro, D., Novais, P., Neves, J.: Monitoring mental fatigue through the analysis of keyboard and mouse interaction patterns. In: Pan, J.-S., et al. (eds.) HAIS 2013. LNCS (LNAI), vol. 8073, pp. 222–231. Springer, Heidelberg (2013). [https://doi.org/10.1007/978-3-642-40846-5\\_23](https://doi.org/10.1007/978-3-642-40846-5_23)
21. Pocius, K.E.: Personality factors in human-computer interaction: a review of the literature. *Comput. Hum. Behav.* **7**(3), 103–135 (1991). [https://doi.org/10.1016/0747-5632\(91\)90002-I](https://doi.org/10.1016/0747-5632(91)90002-I), <https://www.sciencedirect.com/science/article/pii/S074756329190002I>
22. Rayner, K.: Eye movements in reading and information processing: 20 years of research. *Psychol. Bull.* **124**(3), 372–422 (1998). <http://www.ncbi.nlm.nih.gov/pubmed/9849112>
23. Revett, K., Jahankhani, H., de Magalhães, S.T., Santos, H.M.D.: A survey of user authentication based on mouse dynamics. In: Jahankhani, H., Revett, K., Palmer-Brown, D. (eds.) ICGeS 2008. CCIS, vol. 12, pp. 210–219. Springer, Heidelberg (2008). [https://doi.org/10.1007/978-3-540-69403-8\\_25](https://doi.org/10.1007/978-3-540-69403-8_25)
24. Seelye, A., Hagler, S., Mattek, N., Howieson, D.B., Wild, K., Dodge, H.H., Kaye, J.A.: Computer mouse movement patterns: a potential marker of mild cognitive impairment. *Alzheimer's & Dement. Diagn. Assess. Dis. Monit.* **1**(4), 472–480 (2015). <https://doi.org/10.1016/j.dadm.2015.09.006>, <https://www.sciencedirect.com/science/article/pii/S2352872915000792>, <http://linkinghub.elsevier.com/retrieve/pii/S2352872915000792>
25. Yamauchi, T., Xiao, K.: Reading emotion from mouse cursor motions: affective computing approach, November 2017. <https://doi.org/10.1111/cogs.12557>
26. Zimmermann, P., Guttormsen, S., Danuser, B., Gomez, P.: Affective computing—a rationale for measuring mood with mouse and keyboard. *Int. J. Occup. Saf. Ergon.* **9**(4), 539–551 (2003). <https://doi.org/10.1080/10803548.2003.11076589>





# Knowledge Compilation Techniques for Model-Based Diagnosis of Complex Active Systems

Gianfranco Lamperti<sup>1</sup>(✉), Marina Zanella<sup>1</sup>, and Xiangfu Zhao<sup>2</sup>

<sup>1</sup> Department of Information Engineering, University of Brescia, Brescia, Italy  
gianfranco.lamperti@unibs.it

<sup>2</sup> College of Mathematics, Physics and Information Engineering,  
Zhejiang Normal University, Jinhua, China

**Abstract.** According to complexity science, the essence of a complex system is the emergence of unpredictable behavior from interaction among components. Loosely inspired by this idea, a diagnosis technique of a class of discrete-event systems, called complex active systems, is presented. A complex active system is a hierarchical graph, where each node is a network of communicating automata, called an active unit. Specific interaction patterns among automata within an active unit give rise to the occurrence of emergent events, which may affect the behavior of superior active units. This results in the stratification of the behavior of the complex active system, where each different stratum corresponds to a different abstraction level of the emergent behavior. As such, emergence is a peculiar property of a complex active system. To speed up the diagnosis task, model-based knowledge is compiled offline and exploited online by the diagnosis engine. The technique is sound and complete.

**Keywords:** Knowledge compilation · Complex system  
Emergent behavior · Discrete-event system  
Active system · Model-based diagnosis · Lazy techniques

## 1 Introduction

In an interview in January 2000, the physicist Stephen Hawking was asked the following question [32]: *Some say that while the 20th century was the century of physics, we are now entering the century of biology. What do you think of this?* Professor Hawking replied: *I think the next century will be the century of complexity.* Colloquially, we use the qualifier “complex” to indicate something that is complicated in nature. In this perspective, a *complex system* is complicated in structure and behavior, such as an aircraft or a nuclear power plant. However, according to complexity science [1,2,8,24], although it is likely to be complicated, a complex system does not equate to a complicated system. In this different perspective, a complex system is composed of several individual

© IFIP International Federation for Information Processing 2018

Published by Springer Nature Switzerland AG 2018. All Rights Reserved

A. Holzinger et al. (Eds.): CD-MAKE 2018, LNCS 11015, pp. 43–64, 2018.

[https://doi.org/10.1007/978-3-319-99740-7\\_4](https://doi.org/10.1007/978-3-319-99740-7_4)

components that, once aggregated, assume collective characteristics that are not manifested, and cannot be predicted from the properties of the individual components. So, a human being is much more than the union of some 100 trillion cells that make up her body. Likewise, a cell of a human body is much more than the union of its molecules. What we think as a human being, in terms of personality and character, is in fact a collective manifestation of the different interactions among the neurons and synapses in the brain. On their turn, these are in continuous interaction with the other cells of the body, possibly with clusters of cells that constitute organs, which themselves are complex systems. Locally, each cell is characterized by its specific behavior and interaction rules, globally resulting in the collective manifestation of the human being.

Complexity science questions the traditional reductionist approach adopted in the natural sciences, namely the reduction of complex natural phenomena to several simple processes, and the application of a simple theory to each process. More specifically, the *principle of superimposition* is no longer accepted, namely that the comprehension of the whole phenomenon relies on the superimposition of its parts. Based on this principle, for instance, if you understand the theory of an elementary particle, then you will understand every natural phenomenon. Likewise, if you understand DNA, then you will comprehend all biological phenomena. By contrast, a significant aspect of complex systems is that a new collective level emerges through interactions between autonomous elements. Hence, in order to comprehend the complex system, additional knowledge is required beyond the comprehension of the single elements. More generally, a complex system is structured in a hierarchy of semi-autonomous subsystems, with the behavior of a subsystem at level  $i$  of the hierarchy being affected, although not completely determined, by the behavior of each subsystem at level  $i - 1$ . As such, the behavior of a complex system is *stratified*.

Loosely inspired by complexity science, this paper presents a method to extract knowledge - above all, about emergent behavior - from the models of individual clusters of (component) systems and to exploit this knowledge for the lazy diagnosis of a class of discrete event systems (DESs) [5], called *complex active systems* (CASs). A sort of CASs was first introduced in [20, 21]; however, the relevant model-based diagnosis task proved naive. Subsequently, a viable diagnosis technique was presented in [17, 23] and extended in [22]. To the best of our knowledge, apart from the works cited above, no approach to diagnosis of DESs (much less to diagnosis of static systems) based on the complexity paradigm has been proposed so far. Moreover, none of the complexity-inspired approaches cited above is based on knowledge compilation. Still, several works can be related to this paper in varying degree, as discussed below.

An approach is described in [14], where the notion of a *supervision pattern* is introduced for flat DESs, which allows for a flexible specification of the diagnosis problem. However, the events associated with supervision patterns are not exploited for defining any emergent behavior. Diagnosis of hierarchical finite state machines (HFSMs), which are inspired by state-charts [9], provides a sort of structural complexity. Diagnosis of HFSMs has been considered in [13, 26].

However, complexity is merely confined to structure, without emergent events or behavior stratification. An algorithm for computing minimal diagnoses of tree-structured systems is presented in [31]. Subdiagnoses are generated by traversing top-down the tree, which are eventually combined to yield the candidate diagnoses of the whole system. However, despite the fact that the considered systems are static and the diagnosis method is consistency-based, neither complexity nor emergent behavior is conceived, as the goal of the technique is efficiency of the diagnosis task by exploitation of the structure of the system. An approach to consistency-based diagnosis of static systems supported by structural abstraction (which aggregates components to represent the system at different levels of structural detail) is described in [6] as a remedy of computational complexity of model-based diagnosis. Evidence from experimental evaluation indicates that, on average, the technique performs favorably with the algorithm of Mozetič [25]. Still, no emergent behavior is conceived. A technique for consistency-based diagnosis of multiple faults in combinational circuits is presented in [30]. To scale the diagnosis to large circuits, a technique for hierarchical diagnosis is proposed. An implementation on top of the tool presented in [12], which assumes that the system model has been compiled into propositional formulas in decomposable negation normal form (DNNF), has demonstrated the effectiveness of the approach. However, neither emergent behavior nor behavior stratification is conceived, and the technique addresses static systems only. A scalable technique for diagnosability checking of DESs is proposed in [28]. In contrast with the method presented in [27], which suffers from exponential complexity, new algorithms with polynomial complexity were proposed in [15,33], called the *twin plant* method. However, the construction of a global twin plant, which corresponds to the synchronization based on observable events of two identical instances of the automaton representing the whole DES behavior, is often impractical. The method proposed in [28] exploits the distribution of a DES to build a local twin plant for each component. Then, the DES components (and their relevant local twin plants) are organized into a *jointree*, a classical tool adopted in various fields of Artificial Intelligence, including probabilistic reasoning [11,29] and constraint processing [7]. The transformation of the DES into a jointree is an artifice for reducing the complexity of the diagnosability analysis task. Neither emergent behavior nor behavior stratification is assumed for the DES, nor any knowledge extraction is performed. A variant of the decentralized/distributed approach to diagnosis of DESs is introduced in [16], with the aim of computing local diagnoses which are globally consistent. To this end, as in [28] but in the different perspective of diagnosis computation rather than diagnosability, a technique based on jointrees (called *junction trees* in the paper) is proposed. The goal is to mitigate the complexity of model-based diagnosis of DESs associated with abduction-based elicitation of system trajectories. However, the goal of [16] is the efficiency of the diagnosis task, which is performed online only, thereby without exploiting any compiled knowledge. The transformation of the DES into a junction tree is a technical stratagem for improving the decentralized/distributed approach to diagnosis of DESs. The DES is plain, with no emergent behavior.

The contribution of the present paper is threefold: (a) specification of a CAS based on active units, (b) proposal of a process for extracting knowledge from active units, and (c) specification of a diagnosis task for CASs exploiting compiled knowledge.

## 2 Complex Active Systems

A CAS can be defined bottom-up, starting from its atomic elements, the *active components* (ACs). The behavior of an AC, which is equipped with input/output *terminals*, is defined by a communicating automaton [3]. The transition function moves the AC from one state to another when a specific *input event* is *ready* at an input terminal. In so doing, the transition possibly generates a set of *output events* at several output terminals. If specified so, an AC can perform a transition without the need for a ready event; formally, the transition is triggered by the “ $\varepsilon$ ” *empty event*.<sup>1</sup> ACs communicate with one another through *links*. A link is a connection between an output terminal  $o$  of an AC  $c$  and an input terminal  $i'$  of another AC  $c'$ . When an event  $e$  is generated at  $o$  by a transition in  $c$ ,  $e$  becomes ready at terminal  $i'$  of  $c'$ . At most one link can be connected with a terminal.

When several ACs are connected by links, the resulting network is an *active system* (AS) [18,19]. A state of an AS is a pair  $(S, E)$ , where  $S$  is the array of states of the components in the AS and  $E$  is the array of (possibly empty) events that are ready at the input terminals of the components. A state of the AS is *quiescent* iff all elements in  $E$  are  $\varepsilon$  (no event is ready). A *trajectory*  $T$  of an AS is a finite sequence of transitions of ACs in the AS, namely  $T = [t_1(c_1), \dots, t_q(c_q)]$ , which moves the AS from an *initial* quiescent state to a *final* quiescent state. In an AS, a terminal that is not connected with any link is *dangling*.

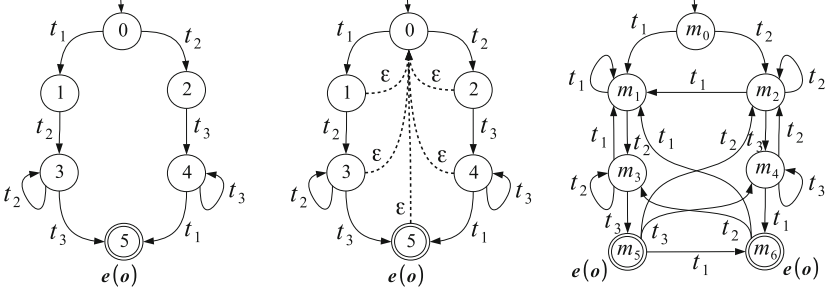
An *active unit* (AU) is an AS extended by a set of input terminals, a set of output terminals, and a set of *emergent events*, where the input terminals are the dangling input terminals of the AS. Each emergent event is a pair  $(e(o), r)$ , where  $e$  is an event generated at the output terminal  $o$  of the AU and  $r$  is a regular expression whose alphabet is a subset of the transitions of the ACs involved in the AU. The event  $e(o)$  *emerges*<sup>2</sup> from a trajectory<sup>2</sup> of the AU when a sequence of transitions in the trajectory matches  $r$ . The state of recognition of an emergent event  $(e(o), r)$  is maintained by a *matcher*, namely a deterministic finite automaton (DFA), derived from  $r$ , in which each final state corresponds to the occurrence of  $e(o)$ . Remarkably, the notion of a matcher of  $r$  differs from that of a recognizer of  $r$ , as illustrated below.

*Example 1.* Let  $(e(o), r)$  be an emergent event in an AU  $\mathcal{U}$ , with  $\Sigma = \{t_1, t_2, t_3\}$  being the alphabet of the regular expression

$$r = t_1 t_2^+ t_3 \mid t_2 t_3^+ t_1. \quad (1)$$

<sup>1</sup> Transitions triggered by empty events are typically used for modeling state changes caused by external events (e.g. a lightning may cause the reaction of a sensor in a protection system).

<sup>2</sup> The notion of a trajectory defined for an AS is also valid for the AU incorporating the AS.



**Fig. 1.** The recognizer of  $r = t_1 t_2^+ t_3 \mid t_2 t_3^+ t_1$ , where  $\Sigma = \{t_1, t_2, t_3\}$  (left); the NFA obtained by the insertion of  $\varepsilon$ -transitions (center); and the matcher  $\mu(e(o), r)$  (right).

Depicted on the left side of Fig. 1 is the *recognizer* of  $r$ , a DFA with the final state being marked by the emergent event  $e(o)$ . When the final state 5 is reached, the emergent event  $e$  is generated at the output terminal  $o$  of  $\mathcal{U}$ . Assume the following trajectory  $T$  in  $\mathcal{U}^*$ ,

$$T = [t_3, \overbrace{t_1, t_2, t_3}^{T'}, t_1, t_3]. \quad (2)$$

$T$  includes two overlapping subtrajectories matching  $r$ , namely  $T' = [t_1, t_2, t_3]$  and  $T'' = [t_2, t_3, t_1]$ , where the suffix  $[t_2, t_3]$  of  $T'$  is a prefix of  $T''$ . Hence, the emergent event  $e(o)$  is expected to occur twice in  $T$ , in correspondence of the last transition of  $T'$  and  $T''$ , respectively.

Assume further to trace the occurrences of  $e(o)$  based on the recognizer of  $r$  displayed on the left side of Fig. 1. The sequence of states of this recognizer is outlined in the second row of Table 1, where each state is reached by the corresponding transition of  $T$  listed in the first row of the table. As indicated in the third row, the emergent event  $e(o)$  is correctly generated at the fourth transition in  $T$ , which is the last transition of the subtrajectory  $T'$ . At this point, since no transition exits the final state 5, the recognizer starts again from its initial state 0 in order to keep matching. It first changes state to 1 in correspondence of  $t_1$ , and with  $t_3$  (mismatch) it returns to 0. The result is that, owing to the overlapping of  $T'$  and  $T''$ , the second  $e(o)$  is not produced.

Given the pair  $(e(o), r)$ , in order to recognize all (possibly overlapping) instances of  $r$ , we need to transform the recognizer of  $r$  into the *matcher* of  $r$  as follows:

1. An  $\varepsilon$ -transition is inserted into the recognizer from each non-initial state to the initial state;
2. The nondeterministic finite automaton (NFA) obtained in step 1 is determinized into an equivalent DFA;
3. The DFA generated in step 2 is minimized, thereby obtaining the matcher of  $r$ .

The final states of the minimized DFA are marked by the emergent event  $e(o)$ .

**Table 1.** Generation of emergent events with overlapping.

Transitions in $T$	$t_3$	$t_1$	$t_2$	$t_3$	$t_1$	$t_3$
States of the recognizer of $r$	0	1	3	5	1	0
Emergent events				$e(o)$		
States of the matcher $\mu(e(o), r)$	$m_0$	$m_1$	$m_3$	$m_5$	$m_6$	$m_0$
Emergent events				$e(o)$	$e(o)$	

*Example 2.* With reference to Example 1, consider the recognizer of the regular expression  $r$  displayed on the left side of Fig. 1. Outlined on the center is the NFA obtained by inserting five  $\varepsilon$ -transitions (dashed arrows) toward the initial state (step 1). The DFA resulting from the determinization of the NFA is displayed on the right side of the figure (step 2). Incidentally, this DFA is also minimal, thus minimization (step 3) is not applied. In conclusion, the DFA on the right side of Fig. 1 is the matcher  $\mu(e(o), r)$ , with the final states  $m_5$  and  $m_6$  being marked by  $e(o)$ . The dynamics of the matching performed by  $\mu(e(o), r)$  on the trajectory  $T$  is outlined in the last two rows of Table 1. In sharp contrast with the matching performed by the recognizer of  $r$ , which produces only one  $e(o)$ , the matcher correctly generates the emergent event twice, based on the two overlapping subtrajectories of  $T$ , namely  $T' = [t_1, t_2, t_3]$  and  $T'' = [t_2, t_3, t_1]$ . Unlike the recognizer of  $r$ , after reaching the state  $m_5$  and generating  $e(o)$ , the next transition  $t_1$  moves the matcher to the other final state  $m_6$ , thereby generating the second occurrence of  $e(o)$  correctly.

A CAS  $\mathcal{X}$  is a directed tree, where each node is an AU and each arc  $(\mathcal{U}, \mathcal{U}')$ , with  $\mathcal{U}$  being a child of  $\mathcal{U}'$  in the tree, is a set of links connecting all the output terminals of  $\mathcal{U}$  with some input terminals of  $\mathcal{U}'$ . A component/link is in  $\mathcal{X}$  iff it is in an AU of  $\mathcal{X}$ .

*Example 3.* Outlined on the left side of Fig. 2 is a CAS called  $\bar{\mathcal{X}}$ , involving the AUs  $A$ ,  $B$ , and  $C$ , where  $A$  has one input terminal,  $B$  has one input terminal and one output terminal, and  $C$  has one output terminal. Since each AU has at most one child, the hierarchy of  $\bar{\mathcal{X}}$  is linear (no branches). To avoid irrelevant details, the internal structure of the AUs is omitted.

A state of  $\mathcal{X}$  is a triple  $(S, E, M)$ , where  $S$  is the array of states of the ACs in  $\mathcal{X}$ ,  $E$  is the array of (possibly empty) events ready at the input terminals of the ACs in  $\mathcal{X}$ , and  $M$  is the array of states of the matchers of the events emerging from all AUs in  $\mathcal{X}$ . Like for ASSs, a state of  $\mathcal{X}$  is *quiescent* iff all elements in  $E$  are  $\varepsilon$ . A *trajectory* of  $\mathcal{X}$  is a finite sequence of transitions of the ACs in  $\mathcal{X}$ ,

$T = [t_1(c_1), \dots, t_q(c_q)]$ , which moves  $\mathcal{X}$  from an *initial* quiescent state to a *final* quiescent state. Given an initial state  $x_0$  of  $\mathcal{X}$ , the *space* of  $\mathcal{X}$  is a DFA

$$\mathcal{X}^* = (\Sigma, X, \tau, x_0, X_f), \quad (3)$$

where  $\Sigma$  is the alphabet, namely the set of transitions of the ACs in  $\mathcal{X}$ ,  $X$  is the set of states,  $\tau$  is the transition function, namely  $\tau : X \times \Sigma \mapsto X$ , and  $X_f \subseteq X$  is the set of final states, which are quiescent. In other words, the sequence of symbols in  $\Sigma$  (transitions of ACs) marking a path (sequence of transitions) in  $\mathcal{X}^*$  from  $x_0$  to a final state is a trajectory of  $\mathcal{X}$ . Hence,  $\mathcal{X}^*$  is a finite representation of the (possibly infinite) set of trajectories of  $\mathcal{X}$  starting at  $x_0$ .

Within a trajectory of a CAS  $\mathcal{X}$ , each transition is either *observable* or *unobservable*. The mode in which an observable transition is perceived is defined by the *viewer*  $\mathcal{V}$  of  $\mathcal{X}$ , namely a set of pairs  $(t(c), \ell)$ , where  $t(c)$  is a transition of an AC  $c$  and  $\ell$  is an *observable label*. Given a transition  $t(c)$ , if  $(t(c), \ell) \in \mathcal{V}$ , then  $t(c)$  is observable via label  $\ell$ ; otherwise,  $t(c)$  is unobservable.

Assuming that  $\mathcal{X}$  includes  $n$  AUs, namely  $\mathcal{U}_1, \dots, \mathcal{U}_n$ , the *temporal observation* of a trajectory  $T$  of  $\mathcal{X}$  based on a viewer  $\mathcal{V}$ , denoted  $T_{\mathcal{V}}$ , is an array  $(O_1, \dots, O_n)$  where,  $\forall i \in [1..n]$ ,  $O_i$  is the *observation* of  $\mathcal{U}_i$ , defined as the sequence of observable labels associated with the observable transitions in  $T$  that are relevant to the ACs in  $\mathcal{U}_i$  only:

$$O_i = [\ell \mid t(c) \in T, c \in \mathcal{U}_i, (t(c), \ell) \in \mathcal{V}]. \quad (4)$$

In other words,  $T_{\mathcal{V}}$  is the result of grouping by AUs the observable labels associated with the observable transitions in  $T$ .

Not only each transition in a trajectory  $T$  is either observable or unobservable; it also is either *normal* or *faulty*. Faulty transitions are defined by the *ruler*  $\mathcal{R}$  of  $\mathcal{X}$ , a set of pairs  $(t(c), f)$ , where  $t(c)$  is a transition of an AC  $c$  and  $f$  is a *fault*. Given a transition  $t(c)$ , if  $(t(c), f) \in \mathcal{R}$ , then  $t(c)$  is faulty via fault  $f$ ; otherwise,  $t(c)$  is normal. The *diagnosis* of a trajectory  $T$  of  $\mathcal{X}$  based on  $\mathcal{R}$ , denoted  $T_{\mathcal{R}}$ , is defined as follows<sup>3</sup>:

$$T_{\mathcal{R}} = \{f \mid t(c) \in T, (t(c), f) \in \mathcal{R}\}. \quad (5)$$

If  $T_{\mathcal{R}} \neq \emptyset$ , then  $T$  is faulty; otherwise,  $T$  is normal. Even if  $\mathcal{X}^*$  includes an infinite number of trajectories, the domain of the possible diagnoses is always finite, this being the powerset  $2^{\mathcal{F}}$ , where  $\mathcal{F}$  is the domain of faults involved in  $\mathcal{R}$ .

### 3 Diagnosis Problem

When  $\mathcal{X}$  performs an (unknown) trajectory  $T$ , what is observed is a temporal observation  $\mathcal{O} = T_{\mathcal{V}}$ , where  $\mathcal{V}$  is the viewer of  $\mathcal{X}$ . However, owing to partial unobservability, several (possibly an infinite number of) trajectories may be compatible with  $\mathcal{O}$ . Hence, several (a finite number of) *candidate diagnoses* may be

<sup>3</sup> More generally, any set of faults is called a diagnosis.

compatible with  $\mathcal{O}$ . The goal is finding all these candidates. A *diagnosis problem*  $\wp(\mathcal{X}, \mathcal{O})$  is an association between  $\mathcal{X}$  and a temporal observation  $\mathcal{O}$ . Given the viewer  $\mathcal{V}$  and the ruler  $\mathcal{R}$  of  $\mathcal{X}$ , the *solution* to  $\wp(\mathcal{X}, \mathcal{O})$  is the set of candidate diagnoses

$$\Delta(\wp(\mathcal{X}, \mathcal{O})) = \{ T_{\mathcal{R}} \mid T \in \mathcal{X}^*, T_{\mathcal{V}} = \mathcal{O} \}. \quad (6)$$

*Example 4.* Consider the CAS  $\bar{\mathcal{X}}$  in Example 3. Assume that the observable labels involved in  $\bar{\mathcal{V}}$  are  $a, b, c, d, e$ , and  $f$ ; the faults involved in the ruler  $\bar{\mathcal{R}}$  are  $p, q, v, w, x, y$ , and  $z$ ; the temporal observation is  $\bar{\mathcal{O}} = (\bar{\mathcal{O}}_A, \bar{\mathcal{O}}_B, \bar{\mathcal{O}}_C)$ , where  $\bar{\mathcal{O}}_A = [e, f]$ ,  $\bar{\mathcal{O}}_B = [c, d]$ , and  $\bar{\mathcal{O}}_C = [a, b, a]$ . We have that  $\wp(\bar{\mathcal{X}}, \bar{\mathcal{O}})$  is a diagnosis problem.

It should be clear that Eq. (6) is only a definition formalizing what the solution to a diagnosis problem is. It makes no sense to enumerate the candidate diagnoses as suggested by this definition, as the size of  $\mathcal{X}^*$  is exponential in the number of ACs and input terminals. The space  $\mathcal{X}^*$  plays only a formal role and is never materialized. Even the reconstruction of the subspace of  $\mathcal{X}^*$  that is compatible with  $\mathcal{O}$  is out of the question because, in the worst case, it suffers from the same exponential complexity of  $\mathcal{X}^*$ . So, what to do? The short answer is: focusing on the AUs rather than on the whole of  $\mathcal{X}$ . The important points are soundness and completeness: the set of diagnoses determined must coincide with the set of candidate diagnoses defined in Eq. (6).

## 4 Knowledge Compilation

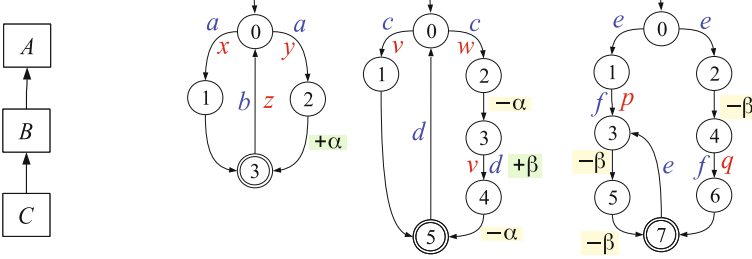
The diagnosis process involves several tasks, which are performed either *offline* or *online*, that is, *before* or *while* the CAS is being operated, respectively. The tasks performed offline are collectively called “knowledge compilation”, and the resulting data structures, “compiled knowledge”. Compiled knowledge is meant to speed up the task performed online by the diagnosis engine. In offline processing, the AUs are processed independently from one another. For each AU  $\mathcal{U}$ , three sorts of data structures are compiled in cascade: the *unit space*  $\mathcal{U}^*$ , the *unit labeling*  $\mathcal{U}_\delta$ , and the *unit knowledge*  $\mathcal{U}_\Delta$ , of which the latter is exploited online. Online processing depends on a diagnosis problem to be solved, specifically, on the observation, whereas offline processing does not.

**Definition 1 (Unit space).** *Let  $\mathcal{U}$  be an AU involving an AS  $\mathcal{A}$ . The space of  $\mathcal{U}$  is a DFA*

$$\mathcal{U}^* = (\Sigma, U, \tau, u_0, U_f), \quad (7)$$

where: the alphabet  $\Sigma$  is the set of transitions of ACs in  $\mathcal{U}$ ;  $U$  is the set of states  $(S, E, M)$ , where  $(S, E)$  is a state of  $\mathcal{A}$  and  $M$  is the array of states of the matchers of the emergent events of  $\mathcal{U}$ ;  $u_0 = (S_0, E_0, M_0)$  is the initial state, where  $(S_0, E_0)$  is a quiescent state of  $\mathcal{A}$  and  $M_0$  is the array of initial states of the matchers;  $U_f \subseteq U$  is the set of final states, where  $(S, E, M) \in U_f$  iff  $(S, E)$  is a quiescent state of  $\mathcal{A}$ ; and  $\tau : U \times \Sigma \mapsto U$  is the transition function,





**Fig. 2.** CAS  $\bar{\mathcal{X}}$  (left) and unit spaces  $C^*$ ,  $B^*$ , and  $A^*$ . (Color figure online)

where  $(S, E, M) \xrightarrow{t(c)} (S', E', M') \in \tau$  iff  $(S, E) \xrightarrow{t(c)} (S', E')$  occurs in  $\mathcal{A}$ ,  $M'$  is the result of updating  $M$  based on  $t(c)$ , and  $(S', E', M')$  is connected to a final state<sup>4</sup>.

*Example 5.* We assume that the space of each AU has been generated already, as shown next to the CAS  $\bar{\mathcal{X}}$  in Fig. 2, namely  $C^*$  (left),  $B^*$  (center), and  $A^*$  (right). In each unit space, the states are identified by numbers, the final states are double circled, and the transitions are marked by relevant information only, namely: observable label (in blue), fault (in red), occurrence of an emergent event (prefixed by the “+” plus sign), and consumption of an event emerged from a child unit (prefixed by the “−” minus sign). For instance, in  $B^*$ , the transition from 3 to 4 is observable via the label  $d$ , is faulty via the fault  $v$ , and generates the emergent event  $\beta$ . The transition from 2 to 3, which is unobservable and normal, consumes the event  $\alpha$  emerging from  $C$ , the child of  $B$ .

Since the space of an AU does not depend on other AUs, the occurrence of a transition depending on an event  $e$  emerging from a child AU in the CAS is not constrained by the actual readiness of  $e$  at one input terminal, as this information is outside the scope of the AU. Yet, the enforcement of this *interface* constraint is not neglected, but only deferred to the diagnosis engine operating online (cf. Sect. 5).

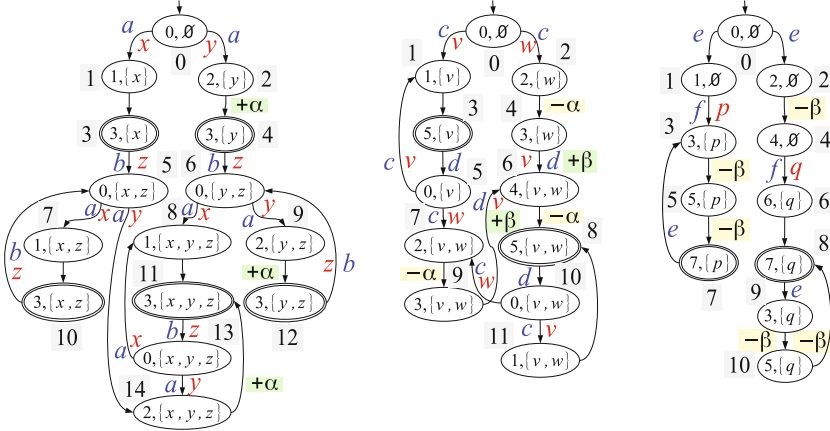
**Definition 2 (Unit labeling).** Let  $\mathcal{U}^* = (\Sigma, U, \tau, u_0, U_f)$  be the space of an AUU in a CAS  $\mathcal{X}$ , let  $\mathcal{R}$  be the ruler of  $\mathcal{X}$ , and let  $\delta$  be the domain of diagnoses in  $\mathcal{U}$ . The labeling of  $\mathcal{U}$  is a DFA

$$\mathcal{U}_\delta = (\Sigma, U', \tau', u'_0, U'_f), \quad (8)$$

where:  $U' \subseteq U \times \delta$  is the set of states;  $u'_0 = (u_0, \emptyset)$  is the initial state;  $U'_f \subseteq U'$  is the set of final states, with  $(u, \delta) \in U'_f$  if  $u \in U_f$ ;  $\tau' : U' \times \Sigma \mapsto U'$  is the transition function, with  $(u, \delta) \xrightarrow{t(c)} (u', \delta') \in \tau'$  iff  $u \xrightarrow{t(c)} u' \in \tau$  and

$$\delta' = \begin{cases} \delta \cup \{f\} & \text{if } (t(c), f) \in \mathcal{R} \\ \delta & \text{otherwise.} \end{cases}$$

<sup>4</sup> The requirement on connection means that there is a contiguous sequence of transitions from  $(S', E', M')$  to a final state in  $U_f$ .



**Fig. 3.** Unit labelings  $C_\delta$  (left),  $B_\delta$  (center), and  $A_\delta$  (right).

*Example 6.* Displayed on Fig. 3 are the unit labelings  $C_\delta$ ,  $B_\delta$ , and  $A_\delta$  derived from the unit spaces outlined in Fig. 2. To facilitate subsequent referencing, states are re-identified by numbers. Owing to the additional field  $\delta$  (set of faults), the number of states in  $\mathcal{U}_\delta$  is generally larger than the number of states in  $\mathcal{U}^*$ .

**Definition 3 (Unit knowledge).** Let  $\mathcal{U}$  be an AU in a CAS  $\mathcal{X}$ , let  $\mathcal{U}_\delta$  be a labeling of  $\mathcal{U}$ , and let  $\mathcal{R}$  be the ruler of  $\mathcal{X}$ . Let  $\mathcal{U}'_\delta$  be the NFA obtained from  $\mathcal{U}_\delta$  by substituting the alphabet symbols marking the transitions as follows. For each transition  $(u, \delta) \xrightarrow{t(c)} (u', \delta')$  in  $\mathcal{U}_\delta$ ,  $t(c)$  is replaced with a triple  $(\ell, e, E)$ , where: if  $(t(c), \ell') \in \mathcal{V}$ , then  $\ell = \ell'$ , else  $\ell = \varepsilon$ ; if  $t(c)$  consumes an event  $e'$  emerging from a child unit, then  $e = e'$ , else  $e = \varepsilon$ ;  $E$  is the (possibly empty) set of emergent events generated at  $t(c)$  by  $\mathcal{U}$ . Eventually, all triples  $(\varepsilon, \varepsilon, \emptyset)$  are replaced by  $\varepsilon$ . The unit knowledge  $\mathcal{U}_\Delta$  is the DFA obtained by the determinization of  $\mathcal{U}'_\delta$ , where each final state  $s_f$  of  $\mathcal{U}_\Delta$  is marked by the aggregation of the diagnosis sets associated with the final states of  $\mathcal{U}'_\delta$  within  $s_f$ , denoted  $\Delta(s_f)$ <sup>5</sup>.

*Example 7.* Shown in Fig. 4 are the unit knowledge  $C_\Delta$ ,  $B_\Delta$ , and  $A_\Delta$ , derived from the unit labelings displayed in Fig. 3, where  $\varepsilon$  elements in triples  $(\ell, \alpha, \beta)$  are omitted and states are re-identified by numbers. For instance, consider the final state 4 of  $C_\Delta$  (left of Fig. 4), which includes the states  $7 = (1, \{x, z\})$ ,  $10 = (3, \{x, z\})$ , and  $14 = (2, \{x, y, z\})$  of  $C_\delta$ . Since the state  $10 = (3, \{x, z\})$  in  $C_\delta$  is final (it is final in  $C'_\delta$  too), the state 4 of  $C_\Delta$  is marked by  $\{\{x, z\}\}$ .

<sup>5</sup> Based on the algorithm *Subset Construction* [10], when an NFA is determinized into an equivalent DFA, each state in the DFA is identified by a subset of the states of the NFA.

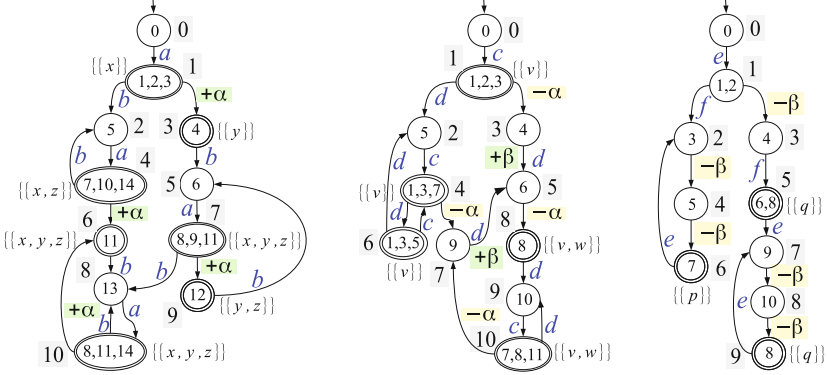


Fig. 4. Unit knowledge  $C_\Delta$  (left),  $B_\Delta$  (center), and  $A_\Delta$  (right).

## 5 Diagnosis Engine

A diagnosis problem for  $\mathcal{X}$  is solved online by the *diagnosis engine* by exploiting the knowledge of the AUs compiled offline. Each unit knowledge  $\mathcal{U}_\Delta$  is constrained by the observation of  $\mathcal{U}$  and by the events emerging from the child AUs of  $\mathcal{U}$ .

**Definition 4 (Abduction of leaf AU).** Let  $\mathcal{U}$  be an AU that is a leaf of the tree of a CAS, let  $\mathcal{U}_\Delta = (\Sigma, S, \tau, s_0, S_f)$  be the knowledge of  $\mathcal{U}$ , and let  $O = [\ell_1, \dots, \ell_n]$  be an observation of  $\mathcal{U}$ . The abduction of  $\mathcal{U}$  is a DFA

$$\mathcal{U}_O = (\Sigma, S', \tau', s'_0, S'_f),$$

where:  $S' \subseteq S \times [0..n]$  is the set of states<sup>6</sup>;  $s'_0 = (s_0, 0)$  is the initial state;  $S'_f \subseteq S'$  is the set of final states, where  $s' \in S'_f$  iff  $s' = (s, n)$ ,  $s \in S_f$ , with  $s'$  being marked by  $\Delta(s') = \Delta(s)$ ;  $\tau' : S' \times \Sigma \mapsto S'$  is the transition function, where  $(s, j) \xrightarrow{(\ell, \varepsilon, E)} (s', j')$  iff  $(s', j')$  is connected to a final state,  $s \xrightarrow{(\ell, \varepsilon, E)} s' \in \tau$ , and

$$j' = \begin{cases} j + 1 & \text{if } \ell \neq \varepsilon \text{ and } \ell_{j+1} = \ell \\ j & \text{if } \ell = \varepsilon. \end{cases} \quad (9)$$

*Example 8.* Consider the unit knowledge  $C_\Delta$  displayed on the left side of Fig. 4. Let  $O_C = [a, b, a]$  be the observation of  $C$ . The unit abduction  $C_O$  is shown on the left side of Fig. 5.

To extend Definition 4 to nonleaf AUs, we introduce the notion of a unit interface in Definition 5 (generalized in Definition 8).

<sup>6</sup> Each natural number in  $[0..n]$  is called an *index* of  $O$ .

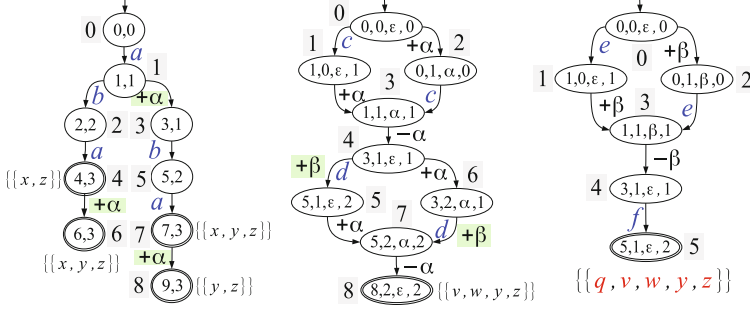


Fig. 5. Unit abductions  $C_{\mathcal{O}}$  (left),  $B_{\mathcal{O}}$  (center), and  $A_{\mathcal{O}}$  (right).

**Definition 5 (Interface of leaf AU).** Let  $\mathcal{U}_{\mathcal{O}}$  be an abduction where  $\mathcal{U}$  is a leaf AU. Let  $\mathcal{U}'_{\mathcal{O}}$  be the NFA obtained from  $\mathcal{U}_{\mathcal{O}}$  by substituting  $E$  for each transition symbol  $(\ell, \varepsilon, E)$  such that  $E \neq \emptyset$ , and  $\varepsilon$  for all other symbols. The interface  $\mathcal{U}_{\mathfrak{S}}$  of  $\mathcal{U}$  is the DFA obtained by determinization of  $\mathcal{U}'_{\mathcal{O}}$ , where each final state  $s'_f$  is marked by the union of the sets of diagnoses marking the final states of  $\mathcal{U}'_{\mathcal{O}}$  included in  $s'_f$ , denoted  $\Delta(s'_f)$ .

*Example 9.* Considering the unit abduction  $C_{\mathcal{O}}$  displayed on the left side of Fig. 5, the unit interface  $C_{\mathfrak{S}}$  is shown on the left side of Fig. 6, where all three states are final.

To extend the notion of a unit abduction to nonleaf AUs (Definition 7), we make use of the join operator defined below.

**Definition 6 (Join).** The join “ $\otimes$ ” of two sets of diagnoses,  $\Delta_1$  and  $\Delta_2$ , is the set of diagnoses defined as follows:

$$\Delta_1 \otimes \Delta_2 = \{ \delta \mid \delta = \delta_1 \cup \delta_2, \delta_1 \in \Delta_1, \delta_2 \in \Delta_2 \}. \quad (10)$$

**Definition 7 (Abduction of nonleaf AU).** Let  $\mathcal{U}$  be a nonleaf AU in  $\mathcal{X}$ , let  $\mathcal{U}_1, \dots, \mathcal{U}_k$  be the child AUs of  $\mathcal{U}$  in  $\mathcal{X}$ , let  $\mathcal{U}_{\Delta} = (\Sigma, S, \tau, s_0, S_f)$  be the knowledge of  $\mathcal{U}$ , let  $O = [\ell_1, \dots, \ell_n]$  be an observation of  $\mathcal{U}$ , let  $\mathbf{E}$  be the domain of tuples of (possibly empty) events emerging from child AUs ready at the input terminals of  $\mathcal{U}$ , let  $\mathcal{U}_{i\mathfrak{S}} = (\Sigma_i, S_i, \tau_i, s_{0i}, S_{fi})$  be the interface of  $\mathcal{U}_i$ ,  $i \in [1..k]$ , and let  $\mathbf{S} = S_1 \times \dots \times S_k$ . The abduction of  $\mathcal{U}$  is a DFA

$$\mathcal{U}_{\mathcal{O}} = (\Sigma', S', \tau', s'_0, S'_f),$$

where:  $\Sigma' = \Sigma \cup \Sigma_1 \cup \dots \cup \Sigma_k$ ;  $S' \subseteq S \times \mathbf{S} \times \mathbf{E} \times [0..n]$  is the set of states;  $s'_0 = (s_0, (s_{01}, \dots, s_{0k}), (\varepsilon, \dots, \varepsilon), 0)$  is the initial state;  $S'_f \subseteq S'$  is the set of final states, where  $s'_f \in S'_f$  iff  $s'_f = (s_f, (s_{f1}, \dots, s_{fk}), (\varepsilon, \dots, \varepsilon), n)$ ,  $s_f \in S_f$ ,  $\forall i \in [1..k]$ ,  $s_{fi} \in S_{fi}$ , and  $s'_f$  is marked by the set of diagnoses

$$\Delta(s'_f) = \Delta(s_f) \otimes \Delta(s_{f1}) \otimes \dots \otimes \Delta(s_{fk}); \quad (11)$$

$\tau' : \Sigma' \times S' \mapsto S'$  is the transition function, where

$$(s, (s_1, \dots, s_k), E, j) \xrightarrow{\sigma} (s', (s'_1, \dots, s'_k), E', j') \in \tau'$$

iff  $(s', (s'_1, \dots, s'_k), E', j')$  is connected to a final state and either of the following conditions holds:

- (a)  $s \xrightarrow{\sigma} s' \in \tau$ ,  $\sigma = (\ell, e, \bar{E})$ , either  $e = \varepsilon$  or  $e$  is ready in  $E$ ,  $E'$  equals  $E$  except that  $e$  is removed, and the index  $j'$  is defined as in Eq. (9);
- (b)  $s_i \xrightarrow{\sigma} s'_i \in \tau_i$ ,  $\sigma = E_i$ , all elements in  $E$  corresponding to the input emergent events in  $E_i$  are  $\varepsilon$ ,  $E'$  equals  $E$  except that all events in  $E_i$  are inserted into  $E'$ .

*Example 10.* Consider the unit knowledge  $B_\Delta$  displayed on the center of Fig. 4 and the unit interface  $C_{\mathfrak{S}}$  displayed on the left side of Fig. 6. Let  $\bar{O}_B = [c, d]$  be the observation of  $B$ . The unit abduction  $B_{\mathcal{O}}$  is shown on the center of Fig. 5. Each state in  $B_{\mathcal{O}}$  is marked by a quadruple  $(s_B, s_C, e, j)$ , where  $s_B$  is a state of  $B_\Delta$ ,  $s_C$  is a state of  $C_{\mathfrak{S}}$  ( $C$  is the unique child of  $B$ ),  $e$  is the event emerging from  $C$  and possibly ready (if  $e \neq \varepsilon$ ) at the input terminal of  $B$  ( $B$  includes one input terminal only), and  $j$  is an index of the observation  $\bar{O}_B$ . Let  $\bar{\Delta}$  be the set of diagnoses marking the final state  $8 = (8, 2, \varepsilon, 2)$ . Based on Eq. (11),  $\bar{\Delta}$  is generated via join  $\Delta(8) \otimes \Delta(2)$ , where  $\Delta(8)$  is the set associated with the state 8 of the unit knowledge  $B_\Delta$  (shown on the center of Fig. 4), namely  $\{\{v, w\}\}$ , and  $\Delta(2)$  is the set associated with the state 2 of the unit interface  $C_{\mathfrak{S}}$  (shown on the left side of Fig. 6), namely  $\{\{y, z\}\}$ . Hence,  $\bar{\Delta} = \{\{v, w\}\} \otimes \{\{y, z\}\} = \{\{v, w, y, z\}\}$ . The unit interface  $B_{\mathfrak{S}}$ , drawn from  $B_{\mathcal{O}}$ , is displayed on the right side of Fig. 6.

**Definition 8 (Interface of AU).** Let  $\mathcal{U}_{\mathcal{O}}$  be an abduction. Let  $\mathcal{U}'_{\mathcal{O}}$  be the NFA obtained from  $\mathcal{U}_{\mathcal{O}}$  by substituting  $E$  for each transition symbol  $(\ell, e, E)$  such that  $E \neq \emptyset$ , and  $\varepsilon$  for all other symbols. The interface  $\mathcal{U}_{\mathfrak{S}}$  of  $\mathcal{U}$  is the DFA obtained by determinization of  $\mathcal{U}'_{\mathcal{O}}$ , where each final state  $s'_f$  is marked by the union of the sets of diagnoses marking the final states of  $\mathcal{U}'_{\mathcal{O}}$  included in  $s'_f$ , denoted  $\Delta(s'_f)$ .

*Example 11.* Considering the unit abduction  $B_{\mathcal{O}}$  displayed on the center of Fig. 5, the unit interface  $B_{\mathfrak{S}}$  is shown on the right side of Fig. 6.

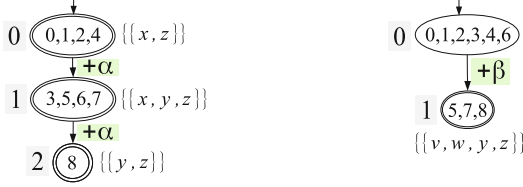
## 6 Soundness and Completeness

Once the abduction process reaches the root  $\mathcal{U}$  of the CAS  $\mathcal{X}$ , thereby generating  $\mathcal{U}_{\mathcal{O}}$ , the solution to the diagnosis problem  $\wp(\mathcal{X}, \mathcal{O})$  can be determined by collecting the diagnoses marking the final states of  $\mathcal{U}_{\mathcal{O}}$  (Proposition 1).

**Proposition 1 (Correctness).** Let  $\wp(\mathcal{X}, \mathcal{O})$  be a diagnosis problem, let  $\mathcal{U}$  be the AU at the root of  $\mathcal{X}$ , let  $\mathcal{U}_{\mathcal{O}}$  be the abduction of  $\mathcal{U}$ , with  $S_f$  being the set of final states, and let

$$\Delta(\mathcal{U}_{\mathcal{O}}) = \bigcup_{s_f \in S_f} \Delta(s_f). \quad (12)$$

We have  $\Delta(\wp(\mathcal{X}, \mathcal{O})) = \Delta(\mathcal{U}_{\mathcal{O}})$ .



**Fig. 6.** Unit interfaces  $C_{\mathfrak{S}}$  (left) and  $B_{\mathfrak{S}}$  (right).

To prove Proposition 1, further terminology is required. Let  $\bar{\mathcal{U}}$  be the CAS within  $\mathcal{X}$  rooted in  $\mathcal{U}$ . The *restriction* of  $\wp(\mathcal{X}, \mathcal{O})$  on  $\bar{\mathcal{U}}$  is a diagnosis problem  $\wp(\bar{\mathcal{U}}, \bar{\mathcal{O}})$  where the viewer  $\bar{\mathcal{V}}$  of  $\bar{\mathcal{U}}$  is the restriction of the viewer of  $\mathcal{X}$  on pairs  $(t(c), \ell)$  such that  $c$  is a component in  $\bar{\mathcal{U}}$ , the ruler  $\bar{\mathcal{R}}$  of  $\bar{\mathcal{U}}$  is the restriction of the ruler of  $\mathcal{X}$  on pairs  $(t(c), f)$  such that  $c$  is a component in  $\bar{\mathcal{U}}$ , the initial state of  $\bar{\mathcal{U}}$  is the restriction of the initial state of  $\mathcal{X}$  on the components and links in  $\bar{\mathcal{U}}$ , and  $\bar{\mathcal{O}}$  is the restriction of  $\mathcal{O}$  on observations of AUs in  $\bar{\mathcal{U}}$ . The notion of a trajectory is extended to any DFA involved in the diagnosis task, namely unit labeling, unit knowledge, unit abduction, and unit interface. A *trajectory* in any such DFA is a string of the regular language of the DFA, that is, the sequence of symbols in the alphabet which mark the sequence of transitions from the initial state to a final state of the DFA; such a final state is called the *accepting state* of the trajectory. The set of trajectories in a DFA  $\mathcal{D}$  (the regular language of  $\mathcal{D}$ ) is denoted by  $\|\mathcal{D}\|$ . For instance,  $T \in \|\mathcal{U}_{\mathcal{O}}\|$  denotes a trajectory  $T$  in the abduction  $\mathcal{U}_{\mathcal{O}}$ .

A *path*  $p$  in a DFA  $\mathcal{D}$  is the sequence of transitions yielding a trajectory  $[\sigma_1, \dots, \sigma_n]$  in  $\mathcal{D}$ , namely  $p = s_0 \xrightarrow{\sigma_1} s_1 \xrightarrow{\sigma_2} \dots \xrightarrow{\sigma_n} s_n$ . A *semipath* in  $\mathcal{D}$  is a prefix of a path in  $\mathcal{D}$ . A *subpath*  $p'$  in  $\mathcal{D}$  is a contiguous sequence of transitions within a path, from state  $s_i$  to state  $s_j$ , namely  $s_i \xrightarrow{\sigma_{i+1}} s_{i+1} \xrightarrow{\sigma_{i+2}} \dots \xrightarrow{\sigma_{j-1}} s_{j-1} \xrightarrow{\sigma_j} s_j$ , concisely denoted  $s_i \xrightarrow{\sigma} s_j$ , where  $\sigma = [\sigma_{i+1}, \dots, \sigma_{j-1}, \sigma_j]$  is the *subtrajectory* generated by  $p'$ .

The notion of an interface is extended to any trajectory. The *interface* of a trajectory  $T$ , denoted  $\mathfrak{S}(T)$ , is the sequence of nonempty sets of emergent events occurring in  $T$ . Specifically, if  $T$  is a trajectory in  $\mathcal{U}_{\mathfrak{S}}$ , then  $\mathfrak{S}(T) = T$ . If  $T$  is a trajectory in either  $\mathcal{U}_{\mathcal{O}}$  or  $\mathcal{U}_{\Delta}$ , then  $\mathfrak{S}(T) = [E \mid (\ell, e, E) \in T, E \neq \emptyset]$ . If  $T$  is a trajectory in either  $\mathcal{U}^*$  or  $\mathcal{U}_{\delta}$ , then each set  $E$  in  $\mathfrak{S}(T)$  is the nonempty set of events emerging at the occurrence of a component transition in  $T$ .

Let  $\mathcal{G} = (N, A)$  be a directed acyclic graph (DAG), where  $N$  is the set of nodes and  $A$  is the set of arcs. Let  $n$  and  $n'$  be two nodes in  $N$ . We say that  $n$  precedes  $n'$ , denoted  $n \prec n'$ , iff  $n'$  is reachable from  $n$  by a contiguous sequence of arcs. A *topological sort*  $\mathcal{S}$  in  $\mathcal{G}$  is a sequence of all nodes in  $N$  (with each node occurring once) such that, for each pair  $(n, n')$  of nodes in  $\mathcal{S}$ , if  $n \prec n'$  in  $\mathcal{G}$ , then  $n \prec n'$  in  $\mathcal{S}$ . The whole (finite) set of topological sorts in  $\mathcal{G}$  is denoted  $\|\mathcal{G}\|$ .

Based on the terminology introduced above, a sketch of the proof of Proposition 1 can be given on the ground of Lemma 11, Lemma 12, Corollary 11, and Lemma 13.

**Lemma 11 (Mapping).** *Let  $\mathcal{D} = (\Sigma, S, \tau, s_0, S_f)$  be a DFA. Let  $\Sigma'$  be an alphabet (possibly including  $\varepsilon$ ) and let  $\Sigma \mapsto \Sigma'$  be a surjective mapping from  $\Sigma$  to  $\Sigma'$ . Let  $\mathcal{N} = (\Sigma', S, \tau', s_0, S_f)$  be the NFA derived from  $\mathcal{D}$  by replacing each transition  $s \xrightarrow{\sigma} \sigma \in \tau$  with  $s \xrightarrow{\sigma'} \sigma \in \tau'$ , where  $\sigma'$  is the symbol in  $\Sigma'$  mapping the symbol  $\sigma$  in  $\Sigma$ . Let  $\mathcal{D}'$  be the DFA obtained by determinization of  $\mathcal{N}$ . If  $T'$  is a trajectory in  $\|\mathcal{D}'\|$  with accepting state  $s'_{\text{f}}$ , then, for each final state  $s_{\text{f}} \in S'_{\text{f}}$ , there is a trajectory  $T$  in  $\|\mathcal{D}\|$  with accepting state  $s_{\text{f}}$  such that  $T'$  is the mapping of  $T$  based on  $\Sigma \mapsto \Sigma'$ .*

**Proof.** By induction on  $T'$ .

(*Basis*) According to the *Subset Construction* determinization algorithm, if  $s \in s'_0$ , where  $s'_0$  is the initial state of  $\mathcal{D}'$ , then there is a path  $s_0 \xrightarrow{\varepsilon^*} s$  in  $\mathcal{N}$  where  $\varepsilon^*$  is a (possibly empty) sequence of  $\varepsilon$ . Hence, there is a path  $s_0 \xrightarrow{\sigma} s$  in  $\mathcal{D}$  such that  $\varepsilon^*$  is the mapping of  $\sigma$  based on  $\Sigma \mapsto \Sigma'$ .

(*Induction*) Assume that there is a semipath  $s'_0 \xrightarrow{\sigma'_1} s'$  in  $\mathcal{D}'$  such that for each  $s \in s'$  there is a semipath  $s_0 \xrightarrow{\sigma} s$  in  $\mathcal{D}$  where  $\sigma'$  is the mapping of  $\sigma$  based on  $\Sigma \mapsto \Sigma'$ . Consider the new semipath  $s'_0 \xrightarrow{\sigma'_1} s' \xrightarrow{\bar{\sigma}} \bar{s}'$  in  $\mathcal{D}'$ . According to *Subset Construction*, for each state  $\bar{s} \in \bar{s}'$  there is a state  $s \in s'$  such that  $s \xrightarrow{\bar{\sigma}_n} \bar{s}$  is a subpath in  $\mathcal{N}$  where  $\bar{\sigma}_n$  starts with  $\bar{\sigma}$ , followed by a (possibly empty) sequence of  $\varepsilon$ . Hence, there is a subpath  $s \xrightarrow{\bar{\sigma}_d} \bar{s}$  in  $\mathcal{D}$  such that  $\bar{\sigma}_d$  maps to  $[\bar{\sigma}]$ . Thus, by virtue of the induction hypothesis,  $\sigma \cup \bar{\sigma}_d$  maps to  $\sigma' \cup [\bar{\sigma}]$  based on  $\Sigma \mapsto \Sigma'$ .  $\square$

**Corollary 11.** *With reference to the assumptions stated in Lemma 11, if  $T'$  is generated by a path  $p' = s'_0 \xrightarrow{\sigma'_1} s'_1 \xrightarrow{\sigma'_2} s'_2 \xrightarrow{\sigma'_3} \dots \xrightarrow{\sigma'_{n-1}} s'_{n-1} \xrightarrow{\sigma'_n} s'_n$  in  $\mathcal{D}'$ , then there is a path  $p = s_0 \xrightarrow{\sigma_1} s_1 \xrightarrow{\sigma_2} s_2 \xrightarrow{\sigma_3} \dots \xrightarrow{\sigma_{n-1}} s_{n-1} \xrightarrow{\sigma_n} s_n$  in  $\mathcal{D}$  such that,  $\forall i \in [1..n]$ ,  $s_{i-1} \in s'_{i-1}$ ,  $s_i \in s'_i$ , and  $[\sigma'_i]$  is the mapping of  $\sigma_i$  based on  $\Sigma \mapsto \Sigma'$ .*

**Lemma 12 (Soundness).** *If  $s'_{\text{f}}$  is a final state in  $\mathcal{U}_{\mathcal{O}}$ ,  $\delta \in \Delta(s'_{\text{f}})$ , and  $T \in \|\mathcal{U}_{\mathcal{O}}\|$  with accepting state  $s'_{\text{f}}$ , then  $\delta \in \Delta(\wp(\bar{\mathcal{U}}, \bar{\mathcal{O}}))$ , where  $\delta = \bar{T}_{\bar{\mathcal{R}}}$ ,  $\bar{T} \in \|\bar{\mathcal{U}}^*\|$ , and  $\Im(\bar{T}) = \Im(T)$ .*

**Proof.** By bottom-up induction on the tree of  $\bar{\mathcal{U}}$ .

(*Basis*) Assume that  $\mathcal{U}$  is a leaf AU. Since  $s'_{\text{f}} = (s_{\text{f}}, n)$  is a final state in  $\mathcal{U}_{\mathcal{O}}$ ,  $\delta \in \Delta(s'_{\text{f}})$ , and  $T \in \|\mathcal{U}_{\mathcal{O}}\|$  with accepting state  $s'_{\text{f}}$ , based on Definition 4 and Lemma 11, there is  $T_{\Delta} \in \|\mathcal{U}_{\Delta}\|$  with accepting state  $s_{\text{f}}$  such that  $\delta \in \Delta(s_{\text{f}})$  and  $\Im(T_{\Delta}) = \Im(T)$ . Hence, based on Definition 3 and Lemma 11, there is  $T_{\delta} \in \|\mathcal{U}_{\delta}\|$  with accepting state  $(s, \delta)$  such that  $\Im(T_{\delta}) = \Im(T_{\Delta}) = \Im(T)$ . Therefore, based on Definition 2, there is  $\bar{T} \in \|\bar{\mathcal{U}}^*\|$  such that  $\bar{T}_{\bar{\mathcal{V}}} = \bar{\mathcal{O}}$  and  $\bar{T}_{\bar{\mathcal{R}}} = \delta$ ; in other words, according to Eq. (6),  $\delta \in \Delta(\wp(\bar{\mathcal{U}}, \bar{\mathcal{O}}))$ . Also,  $\Im(\bar{T}) = \Im(T_{\delta}) = \Im(T)$ .

(*Induction*) Assume that  $\mathcal{U}$  is the parent of the AUs  $\mathcal{U}_1, \dots, \mathcal{U}_k$ , where  $\forall i \in [1..k]$ , if  $s'_{\text{if}}$  is a final state in  $\mathcal{U}_{i\mathcal{O}}$ ,  $\delta_i \in \Delta(s'_{\text{if}})$ , and  $T_i \in \|\mathcal{U}_{i\mathcal{O}}\|$  with accepting state  $s'_{\text{if}}$ , then  $\delta_i \in \Delta(\wp(\bar{\mathcal{U}}_i, \bar{\mathcal{O}}_i))$ , where  $\delta_i = \bar{T}_i \bar{\mathcal{R}}_i$ ,  $\bar{T}_i \in \|\bar{\mathcal{U}}_i^*\|$ , and  $\Im(\bar{T}_i) = \Im(T_i)$ .

Since  $s'_{\text{f}} = (s_{\text{f}}, (s_{1\text{f}}, \dots, s_{k\text{f}}), (\varepsilon, \dots, \varepsilon), n)$  is a final state in  $\mathcal{U}_{\mathcal{O}}$ ,  $\delta \in \Delta(s'_{\text{f}})$ , and  $T \in \|\mathcal{U}_{\mathcal{O}}\|$  with accepting state  $s'_{\text{f}}$ , according to Eq. (11),  $\delta \in \Delta(s'_{\text{f}}) = \Delta(s_{\text{f}}) \circ$

$\Delta(s_{1f}) \otimes \cdots \otimes \Delta(s_{kf})$ , where,  $\forall i \in [1..k]$ ,  $s_{if}$  is final in  $\mathcal{U}_{i\mathfrak{S}}$ . Hence, based on Definition 6,  $\delta = \delta_0 \cup \delta_1 \cup \cdots \cup \delta_k$ , where  $\delta_0 \in \Delta(s_f)$  and,  $\forall i \in [1..k]$ ,  $\delta_i \in \Delta(s_{if})$ . Also, based on Definition 8 and Lemma 11,  $\forall i \in [1..k]$ , there is  $T_i \in \|\mathcal{U}_{i\mathcal{O}}\|$  with accepting state  $s'_{if} \in s_{if}$  such that  $\delta_i \in s'_{if}$  and all emergent events in  $T_i$  are consumed in  $T$ . Thus, by virtue of the induction hypothesis, we have  $\delta_i \in \Delta(\wp(\mathcal{U}_i, \bar{\mathcal{O}}_i))$ , where  $\delta_i = \bar{T}_i \bar{\mathcal{R}}_i$ ,  $\bar{T}_i \in \|\bar{\mathcal{U}}_i^*\|$ , and  $\mathfrak{S}(\bar{T}_i) = \mathfrak{S}(T_i)$ . According to Definition 7, the trajectory  $T$  is composed of triples  $(\ell, e, E)$  interspersed by elements  $E_i$ , with each  $E_i$  being a nonempty set of events emerging from  $\mathcal{U}_i$ . Note that, for each  $E_i$  in  $T$  there is one transition in  $\bar{T}_i$  generating  $E_i$ . Thus, each  $\bar{T}_i$  is composed of transitions  $t(c)$ , where  $c$  is a component in  $\bar{\mathcal{U}}_i$ , in which there are some transitions generating sets  $E_i$  of emergent events. The sequence of  $E_i$  generated in  $\bar{T}_i$  equals the subsequence of  $E_i$  in  $T$ ,  $i \in [1..k]$ . So, there is a sequential correspondence between each  $E_i$  in  $T$  and each transition in  $\bar{T}_i$  generating the set  $E_i$  of emergent events. According to Definition 7 and Corollary 11, if  $s'_0 \xrightarrow{\sigma'_1} s'_1 \xrightarrow{E_{i1}} s''_1 \xrightarrow{\sigma'_2} s'_2 \xrightarrow{E_{i2}} s''_2 \xrightarrow{\sigma'_3} \cdots \xrightarrow{\sigma'_n} s'_n$  is the path in  $\mathcal{U}_{\mathcal{O}}$  generating the trajectory  $T = \sigma'_1 \cup [E_{i1}] \cup \sigma'_2 \cup [E_{i2}] \cup \sigma'_3 \cup \cdots \cup \sigma'_n$ , then there is a path  $s_0 \xrightarrow{\sigma_1} s_1 \xrightarrow{\sigma_2} s_2 \xrightarrow{\sigma_3} \cdots \xrightarrow{\sigma_n} s_n$  in  $\mathcal{U}_{\delta}$  generating the trajectory  $T^* = \sigma_1 \cup \sigma_2 \cup \cdots \cup \sigma_n$ ,  $T^* \in \|\bar{\mathcal{U}}^*\|$ , such that  $T^*$  generates the observation of  $\mathcal{U}$  and the diagnosis  $\delta_0$ . Now, consider the DAG  $\mathcal{G}$  constructed by the following four steps, where each node is either a component transitions or a set  $E_i$  of emergent events, while arcs indicate precedence dependencies between these nodes. Step 1: in  $T$ , substitute  $\sigma_j$  for each  $\sigma'_j$ ,  $j \in [1..n]$ ; this step substitutes sequences of transitions of components in  $\mathcal{U}$  for corresponding sequences of triples  $(\ell, e, E)$  in  $T$ ; Step 2: transform each trajectory  $\bar{T}_i = [t_1, t_2, \dots, t_{n_i}]$ ,  $i \in [1..k]$ , into a connected sequence of nodes  $t_1 \rightarrow t_2 \rightarrow \cdots \rightarrow t_{n_i}$ , where arrows indicate arcs (precedence dependencies); Step 3: transform  $T$  into a connected sequence of nodes in the same way as in step 2; Step 4: for each transition  $t_i$  in  $\bar{T}_i$  generating the set  $E_i$  of emergent events, insert an arc from  $t_i$  to the corresponding  $E_i$  in  $T$ . This results in a DAG, namely a *dependency graph*  $\mathcal{G}$ , where each  $E_i$  is entered by an arc from a transition in  $\bar{T}_i$ ,  $i \in [1..k]$ . Let  $\bar{T}$  be the sequence of transitions relevant to a topological sort of  $\mathcal{G}$ , where all  $E_i$  are eventually removed. As such,  $\bar{T}$  is a sequence of transitions of components in  $\bar{\mathcal{U}}$  fulfilling the precedences imposed by  $\mathcal{G}$ . Remarkably,  $\bar{T}$  fulfills the following properties: (1)  $\bar{T} \in \|\bar{\mathcal{U}}^*\|$ , because the component transitions in each  $\bar{T}_i$  are only constrained by the emptiness of the output terminals of  $\mathcal{U}_i$ , while the component transitions in  $T$  are only constrained by the availability of the events emerging from  $\mathcal{U}_1, \dots, \mathcal{U}_k$ , which are checked by the mode in which the abduction  $\mathcal{U}_{\mathcal{O}}$  is generated; (2)  $\bar{T}_{\mathcal{Y}} = \bar{\mathcal{O}}$ , because the sequence of observable labels generated by transitions in  $T$  equals the observation of  $\mathcal{U}$  and,  $\forall i \in [1..k]$ ,  $\bar{T}_i \bar{\mathcal{V}}_i = \bar{\mathcal{O}}_i$ ; and (3) the sequence of faults generated by the the transitions in  $T$  equals  $\delta_0$  and,  $\forall i \in [1..k]$ ,  $\bar{T}_i \bar{\mathcal{R}}_i = \delta_i$ . In other words,  $\bar{T} \bar{\mathcal{R}} = \delta = \delta_0 \cup \delta_1 \cup \cdots \cup \delta_k$ ; hence,  $\delta \in \Delta(\wp(\bar{\mathcal{U}}, \bar{\mathcal{O}}))$ . Also,  $\mathfrak{S}(\bar{T}) = \mathfrak{S}(T)$ .  $\square$

**Lemma 13 (Completeness).** *If  $\delta \in \Delta(\wp(\bar{\mathcal{U}}, \bar{\mathcal{O}}))$ , where  $\delta = \bar{T} \bar{\mathcal{R}}$  and  $\bar{T} \in \|\bar{\mathcal{U}}^*\|$ , then there is  $T \in \|\mathcal{U}_{\mathcal{O}}\|$  ending in  $s'_f$  such that  $\delta \in \Delta(s'_f)$  and  $\mathfrak{S}(T) = \mathfrak{S}(\bar{T})$ .*



**Proof.** By bottom-up induction on the tree of  $\bar{\mathcal{U}}$ .

(*Basis*) Assume that  $\mathcal{U}$  is a leaf AU. Since  $\delta \in \Delta(\wp(\bar{\mathcal{U}}, \bar{\mathcal{O}}))$ , where  $\delta = \bar{T}_{\bar{\mathcal{R}}}$  and  $\bar{T} \in \|\bar{\mathcal{U}}^*\|$ , based on Definition 2, there is  $T_\delta \in \|\mathcal{U}_\delta\|$  with accepting state  $(s, \delta)$  such that  $\Im(T_\delta) = \Im(\bar{T})$ . Hence, based on Definition 3, there is  $T_\Delta \in \|\mathcal{U}_\Delta\|$  with accepting state  $s_f$  such that  $\delta \in \Delta(s_f)$  and  $\Im(T_\Delta) = \Im(T_\delta) = \Im(\bar{T})$ . Thus, based on Definition 4, there is  $T \in \|\mathcal{U}_\mathcal{O}\|$  ending in  $s'_f = (s_f, n)$  such that  $\delta \in \Delta(s'_f)$  and  $\Im(T) = \Im(T_\Delta) = \Im(\bar{T})$ .

(*Induction*) Assume that  $\mathcal{U}$  is the parent of the AUs  $\mathcal{U}_1, \dots, \mathcal{U}_k$ , where,  $\forall i \in [1..k]$ , if  $\delta_i \in \Delta(\wp(\bar{\mathcal{U}}_i, \bar{\mathcal{O}}_i))$ , where  $\delta_i = \bar{T}_{i\bar{\mathcal{R}}_i}$  and  $\bar{T}_i \in \|\bar{\mathcal{U}}_i^*\|$ , then there is  $T_i \in \|\mathcal{U}_{i\mathcal{O}}\|$  with accepting state  $s'_{if}$  such that  $\delta_i \in \Delta(s'_{if})$  and  $\Im(T_i) = \Im(\bar{T}_i)$ .

Since  $\delta \in \Delta(\wp(\bar{\mathcal{U}}, \bar{\mathcal{O}}))$ , where  $\delta = \bar{T}_{\bar{\mathcal{R}}}$  and  $\bar{T} \in \|\bar{\mathcal{U}}^*\|$ , we have  $\delta = \delta_0 \cup \delta_1 \cup \dots \cup \delta_k$ , where  $\delta_0$  includes the faults of the components in the AU  $\mathcal{U}$  and,  $\forall i \in [1..k]$ ,  $\delta_i$  includes the faults of the components in the CAS  $\bar{\mathcal{U}}_i$ . Since each  $\delta_i$  is the diagnosis of the trajectory  $\bar{T}_i$  corresponding to the restriction of  $\bar{T}$  on the transitions of the components in  $\bar{\mathcal{U}}_i$  such that  $\bar{T}_i \in \|\bar{\mathcal{U}}_i^*\|$ ,  $\bar{T}_{i\bar{\mathcal{V}}_i} = \mathcal{O}_i$ , and  $\bar{T}_{i\bar{\mathcal{R}}_i} = \delta_i$ , based on Eq. (6),  $\delta_i \in \wp(\bar{\mathcal{U}}_i, \bar{\mathcal{O}}_i)$ . Hence, by virtue of the induction hypothesis, there is  $T_i \in \|\mathcal{U}_{i\mathcal{O}}\|$  with accepting state  $s'_{if}$  such that  $\delta_i \in \Delta(s'_{if})$  and  $\Im(T_i) = \Im(\bar{T}_i)$ . Also, based on Definition 8, there is  $T'_i \in \|\mathcal{U}_{i\Im}\|$  with accepting state  $s_{if}$  such that  $\delta_i \in \Delta(s_{if})$  and  $\Im(T'_i) = \Im(T_i) = \Im(\bar{T}_i)$ . Let  $T'$  be the subtrajectory of  $\bar{T}$  including only the transitions of components in  $\mathcal{U}$ . As such,  $T' \in \|\mathcal{U}^*\|$ ,  $T'_\mathcal{V}$  equals the observation in  $\mathcal{O}$  relevant to  $\mathcal{U}$ , and  $T'_\mathcal{R} = \delta_0$ . Hence, based on Definition 2, there is  $T_\delta \in \|\mathcal{U}_\delta\|$  with accepting state  $(s, \delta_0)$  such that  $\Im(T_\delta) = \Im(T)$ . Also, based on Definition 3, there is  $T_\Delta \in \|\mathcal{U}_\Delta\|$  with accepting state  $s_f$  such that  $\delta \in \Delta(s_f)$  and  $\Im(T_\Delta) = \Im(T_\delta) = \Im(T)$ . Thus, based on Definition 7, there is  $T \in \|\mathcal{U}_\mathcal{O}\|$  with accepting state  $s'_f = (s_f, (s_{1f}, \dots, s_{kf}), (\varepsilon, \dots, \varepsilon), n)$  such that  $\delta_0 \in \Delta(s_f)$  and,  $\forall i \in [1..k]$ ,  $\delta_i \in \Delta(s_{if})$ . Since  $\delta = \delta_0 \cup \delta_1 \cup \dots \cup \delta_k$ , based on Definition 6 and according to Eq. (11),  $\delta \in \Delta(s_f) \otimes \Delta(s_{1f}) \otimes \dots \otimes \Delta(s_{kf})$ ; that is,  $\delta \in \Delta(s'_f)$ . Also,  $\Im(T) = \Im(\bar{T})$ .  $\square$

*Example 12.* Consider the unit knowledge  $A_\Delta$  displayed on the right side of Fig. 4 and the unit interface  $B_\Im$  displayed on the right side of Fig. 6. Let  $\bar{\mathcal{O}}_A = [e, f]$  be the observation of  $A$ . The unit abduction  $A_\mathcal{O}$  is shown on the right side of Fig. 5. Let  $\bar{\Delta}$  be the set of diagnoses marking the final state  $5 = (5, 1, \varepsilon, 2)$ . Based on Eq. (11),  $\bar{\Delta}$  is generated via join  $\Delta(5) \otimes \Delta(1)$ , where  $\Delta(5)$  is the set associated with the state 5 of the unit knowledge  $A_\Delta$  (on the right side of Fig. 4), namely  $\{\{q\}\}$ , and  $\Delta(1)$  is the set associated with the state 1 of the unit interface  $B_\Im$  (on the right side of Fig. 6), namely  $\{\{v, w, y, z\}\}$ . Hence,  $\bar{\Delta} = \{\{q, v, w, y, z\}\}$ . Based on Proposition 1,  $\bar{\Delta}$  is the solution to the diagnosis problem  $\wp(\bar{\mathcal{X}}, \bar{\mathcal{O}})$  defined in Example 4.

## 7 Computational Complexity

Had we adapted the diagnosis technique proposed for ASs [18, 19] to the diagnosis of CASs, we would have inherited (and even exacerbated) its poor performance (see the experimental results in [17]). In contrast with diagnosis of ASs, the

diagnosis engine described in Sect. 5 does not require the abduction of the whole system. Instead, it focuses on the abduction of each single AU based on the interface constraints coming from the child AUs. In doing so, it exploits the unit knowledge compiled offline.

We analyze the time complexity of solving a diagnosis problem  $\wp(\mathcal{X}, \mathcal{O})$  based on the (not unreasonable) assumption that the processing time is proportional to the size (number of states) of the data structures generated by the diagnosis engine. Furthermore, we make the following *bounding assumptions*:  $\mathcal{X}$  is composed of  $n$  AUs; each nonleaf AU has  $c$  child AUs, has  $h$  input terminals, and defines  $m$  emergent events; each unit knowledge (generated offline) includes  $k$  states; the length of each AU observation in  $\wp(\mathcal{X}, \mathcal{O})$  is  $o$ . We also assume that the size of the DFA obtained by the determinization of an NFA compares to the size of the NFA<sup>7</sup>. We call this the *determinization assumption*. We first consider the (upper bound of the) complexity  $\mathcal{C}$  of the abduction  $\mathcal{U}_{\mathcal{O}}$  of a leaf AU (at level zero, that is, without children). Based on Definition 4,  $\mathcal{C}_{\mathcal{U}_{\mathcal{O}}} = k \cdot o$ . In order to estimate the complexity of each unit abduction  $\mathcal{U}_{\mathcal{O}}$  at level one, where all children of  $\mathcal{U}$  are leaf AUs, two steps have to be analyzed: (1) generation of  $c$  interfaces and (2) generation of  $\mathcal{U}_{\mathcal{O}}$  based on Definition 7. As to step (1), on the ground of the determinization assumption, the number of states of the interfaces of the child AUs is  $c \cdot k \cdot o$ . Based on Definition 7, the (upper bound of the) number of states generated by step (2) for each unit abduction at level one is  $k \cdot (k \cdot o)^c \cdot m^h \cdot o = (k \cdot o)^{c+1} \cdot m^h$ . Owing to the factor  $(k \cdot o)^{c+1}$ , the contribution  $c \cdot k \cdot o$  of step (1) can be neglected; hence,

$$\mathcal{C}_{\mathcal{U}_{\mathcal{O}}} = (k \cdot o)^{c+1} \cdot m^h. \quad (13)$$

The complexity of each unit abduction  $\mathcal{U}_{\mathcal{O}}$  at the second level (where  $\mathcal{U}$  is the grandparent of leaf AUs) can be computed as before, where the size of each interface equals  $\mathcal{C}_{\mathcal{U}_{\mathcal{O}}}$  in Eq. (13), namely

$$\mathcal{C}_{\mathcal{U}_{\mathcal{O}}} = k \cdot ((k \cdot o)^{c+1} \cdot m^h)^c \cdot m^h \cdot o = (k \cdot o)^{c^2+c+1} \cdot m^{(c+1) \cdot h}. \quad (14)$$

At level  $d$  (depth of the tree) of the root AU, the complexity of the unit abduction is

$$\mathcal{C}_{\mathcal{U}_{\mathcal{O}}} = (k \cdot o)^{e^d + e^{d-1} + \dots + e^2 + e + 1} \cdot m^{(e^{d-1} + \dots + e^2 + e + 1) \cdot h}. \quad (15)$$

Since  $1 + e + e^2 + \dots + e^d = n$ , with  $n$  being the number of AUs in the CAS, the dominant factor in Eq. (15) is  $(k \cdot o)^n$ . In other words, the complexity of the unit abduction is exponential in the number of AUs in the CAS.

Finally, we consider the space complexity of knowledge compilation, which is performed offline. We assume that each AU includes  $p$  components and  $l$

<sup>7</sup> In the worst case, if  $n$  is the number of states of the NFA, then the number of states of the DFA is  $2^n$ . However, this is an extremely improbable scenario since, in practice, the size of the DFA resulting from the determinization of an NFA generated randomly typically compares with the size of the NFA (cf. Fig. 4). If the NFA includes a considerable percentage of  $\varepsilon$ -transitions, then the size of the DFA is likely to be substantially smaller than the size of the NFA [4].

links, with each component having  $s$  states and generating  $q$  different events for each connected link. Each matcher (the DFA recognizing the occurrence of an emergent event) has  $\mu$  states. The number of possible faults is  $f$ . The space complexity of the unit space  $\mathcal{U}^*$  is  $\mathcal{C}_{\mathcal{U}^*} = s^p \cdot (q+1)^\ell \cdot \mu^m$ . The space complexity of the unit labeling  $\mathcal{U}_\delta$  is

$$\mathcal{C}_{\mathcal{U}_\delta} = \mathcal{C}_{\mathcal{U}^*} \cdot 2^f = s^p \cdot (q+1)^\ell \cdot \mu^m \cdot 2^f . \quad (16)$$

According to the determinization assumption, the complexity of the unit knowledge equals the complexity of the unit labeling, namely  $\mathcal{C}_{\mathcal{U}_\Delta} = \mathcal{C}_{\mathcal{U}_\delta}$ .

## 8 Conclusion

The contributions of this paper are the specification of a class of DESs inspired by the complexity paradigm, called complex active systems, and a knowledge-compilation technique that speeds up online diagnosis for such systems. Most notably, the shift from ASs to CASs does not come with an additional cost in the diagnosis task; on the contrary, the diagnosis technique is not only sound and complete, but also viable compared to the diagnosis of ASs. In fact, since a state of the AS includes the states of *all* components and the states of *all* links, the complexity of the abduction of the whole AS in diagnosis of ASs is exponential both in the number of components *and* in the number of links.

This theoretical expectation is confirmed by the experimental results presented in [17], with the diagnosis engine being not supported by any compiled knowledge. Two diagnosis engines have been implemented, one *greedy* and one *lazy*. The greedy engine makes use of the same technique of behavior reconstruction adopted in diagnosis of ASs [19], while the lazy engine operates similarly to the technique proposed in this paper (although without any compiled knowledge). The results clearly show that the processing time of the lazy engine increases almost linearly with the size of the system, in contrast with the processing time of the greedy engine, which grows exponentially.

On the ground of these results, we expect that the technique proposed in this paper, which is essentially a lazy engine exploiting compiled knowledge, is still more efficient than the lazy engine in [17], since the low-level model-based reasoning, performed offline and incorporated in the compiled knowledge, is avoided online. Experiments to confirm this intuition will be carried out.

But, why should we model a real system as a CAS rather than a flat AS? In our opinion, the reason is twofold. First, real event-driven systems are typically organized hierarchically, at different abstraction levels. Modeling one such system as a (flat) AS may be awkward because of the mismatch between the hierarchical organization of the structure and behavior of the real system and the flat organization of the modeling AS. CASs provide a natural modeling support against such a mismatch, where emergent events are the means of communication between strata at different abstraction levels, thereby supporting behavior stratification. In short, the first benefit is ergonomics in the modeling task. Second, once the real system is modeled as a CAS, the diagnosis task provides the

sound and complete solution to a diagnosis problem more efficiently than in a (flat) AS. Since diagnosis is potentially interesting to the degree that it is accurate and viable, the second benefit is correctness and viability of the diagnosis task.

As diagnosis of CASs is still in its infancy, several research paths can be envisaged. First, the tree-based topology of the CAS can be relaxed to a directed acyclic graph (DAG), where an AU can have several parent units. Moreover, each node of the DAG can be generalized to a (possibly cyclic) network of AUs. Second, the language of the patterns defining emergent events can be extended beyond regular expressions, based on more powerful grammars, possibly enriched by semantic rules. Third, the diagnosis task, which in this paper is assumed to be *a posteriori*, that is, performed at the reception of a complete temporal observation of the CAS, can be made *reactive*, where diagnosis is performed as soon as a piece of observation is received. Finally and more challengingly, an *adaptive* CAS can be envisaged, where the behavior of components and AUs can change based on specific patterns of events, so as to convert a nonconstructive or even disruptive behavior to a more constructive behavior.

**Acknowledgments.** This work was supported in part by the Zhejiang Provincial Natural Science Foundation of China (No. LY16F020004).

## References

1. Atay, F., Jost, J.: On the emergence of complex systems on the basis of the coordination of complex behaviors of their elements: synchronization and complexity. *Complexity* **10**(1), 17–22 (2004)
2. Bossomaier, T., Green, D.: *Complex Systems*. Cambridge University Press, Cambridge (2007)
3. Brand, D., Zafripulo, P.: On communicating finite-state machines. *J. ACM* **30**(2), 323–342 (1983)
4. Brognoli, S., Lamperti, G., Scandale, M.: Incremental determinization of expanding automata. *Comput. J.* **59**(12), 1872–1899 (2016)
5. Cassandras, C., Lafortune, S.: *Introduction to Discrete Event Systems*, 2nd edn. Springer, New York (2008). <https://doi.org/10.1007/978-0-387-68612-7>
6. Chittaro, L., Ranon, R.: Hierarchical model-based diagnosis based on structural abstraction. *Artif. Intell.* **155**(1–2), 147–182 (2004)
7. Dechter, R.: *Constraint Processing*. The Morgan Kaufmann Series in Artificial Intelligence. Morgan Kaufmann Publishers Inc., San Francisco (2003)
8. Goles, E., Martinez, S. (eds.): *Complex Systems*. Springer, Dordrecht (2001). <https://doi.org/10.1007/978-94-010-0920-1>
9. Harel, D.: Statecharts: a visual formalism for complex systems. *Sci. Comput. Program.* **8**(3), 231–274 (1987)
10. Hopcroft, J., Motwani, R., Ullman, J.: *Introduction to Automata Theory, Languages, and Computation*, 3rd edn. Addison-Wesley, Reading (2006)
11. Huang, C., Darwiche, A.: Inference in belief networks: A procedural guide. *Int. J. Approx. Reason.* **15**(3), 225–263 (1996)

12. Huang, J., Darwiche, A.: On compiling system models for faster and more scalable diagnosis. In: 20th National Conference on Artificial Intelligence (AAAI 2005), Pittsburgh, PA, pp. 300–306 (2005)
13. Idghamishi, A., Zad, S.: Fault diagnosis in hierarchical discrete-event systems. In: 43rd IEEE Conference on Decision and Control, Paradise Island, Bahamas, pp. 63–68 (2004)
14. Jéron, T., Marchand, H., Pinchinat, S., Cordier, M.: Supervision patterns in discrete event systems diagnosis. In: Seventeenth International Workshop on Principles of Diagnosis (DX 2006), Peñaranda de Duero, Spain, pp. 117–124 (2006)
15. Jiang, S., Huang, Z., Chandra, V., Kumar, R.: A polynomial algorithm for testing diagnosability of discrete event systems. *IEEE Trans. Autom. Control* **46**(8), 1318–1321 (2001)
16. Kan John, P., Grastien, A.: Local consistency and junction tree for diagnosis of discrete-event systems. In: Eighteenth European Conference on Artificial Intelligence (ECAI 2008), pp. 209–213. IOS Press, Amsterdam (2008)
17. Lamperti, G., Quarenghi, G.: Intelligent monitoring of complex discrete-event systems. In: Czarnowski, I., Caballero, A.M., Howlett, R.J., Jain, L.C. (eds.) *Intelligent Decision Technologies 2016*. SIST, vol. 56, pp. 215–229. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-39630-9\\_18](https://doi.org/10.1007/978-3-319-39630-9_18)
18. Lamperti, G., Zanella, M.: *Diagnosis of Active Systems: Principles and Techniques*. Springer International Series in Engineering and Computer Science, vol. 741. Springer, Dordrecht (2003). <https://doi.org/10.1007/978-94-017-0257-7>
19. Lamperti, G., Zanella, M., Zhao, X.: *Introduction to Diagnosis of Active Systems*. Springer, Cham (2018). <https://doi.org/10.1007/978-3-319-92733-6>
20. Lamperti, G., Zhao, X.: Diagnosis of higher-order discrete-event systems. In: Cuzzocrea, A., Kittl, C., Simos, D.E., Weippl, E., Xu, L. (eds.) *CD-ARES 2013*. LNCS, vol. 8127, pp. 162–177. Springer, Heidelberg (2013). [https://doi.org/10.1007/978-3-642-40511-2\\_12](https://doi.org/10.1007/978-3-642-40511-2_12)
21. Lamperti, G., Zhao, X.: Specification and model-based diagnosis of higher-order discrete event systems. In: *IEEE International Conference on Systems, Man, and Cybernetics (SMC 2013)*, Manchester, United Kingdom, pp. 2342–2347 (2013)
22. Lamperti, G., Zhao, X.: Diagnosis of complex active systems with uncertain temporal observations. In: Buccafurri, F., Holzinger, A., Kieseberg, P., Tjoa, A.M., Weippl, E. (eds.) *CD-ARES 2016*. LNCS, vol. 9817, pp. 45–62. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-45507-5\\_4](https://doi.org/10.1007/978-3-319-45507-5_4)
23. Lamperti, G., Zhao, X.: Viable diagnosis of complex active systems. In: *IEEE International Conference on Systems, Man, and Cybernetics (SMC 2016)*, Budapest, pp. 457–462 (2016)
24. Licata, I., Sakaji, A.: *Physics of Emergence and Organization*. World Scientific, Singapore (2008)
25. Mozetič, I.: Hierarchical diagnosis. In: Hamscher, W., Console, L., de Kleer, J. (eds.) *Readings in Model-Based Diagnosis*. Morgan Kaufmann (1992)
26. Paoli, A., Lafortune, S.: Diagnosability analysis of a class of hierarchical state machines. *J. Discrete Event Dyn. Syst. Theory Appl.* **18**(3), 385–413 (2008)
27. Sampath, M., Sengupta, R., Lafortune, S., Sinnamohideen, K., Teneketzis, D.: Diagnosability of discrete-event systems. *IEEE Trans. Autom. Control* **40**(9), 1555–1575 (1995)
28. Schumann, A., Huang, J.: A scalable jointree algorithm for diagnosability. In: *Twenty-Third National Conference on Artificial Intelligence (AAAI 2008)*, Chicago, IL, pp. 535–540 (2008)

29. Shenoy, P., Shafer, G.: Propagating belief functions with local computations. *IEEE Expert* **1**(3), 43–52 (1986)
30. Siddiqi, S., Huang, J.: Hierarchical diagnosis of multiple faults. In: 20th International Joint Conference on Artificial Intelligence (IJCAI 2007), Hyderabad, India, pp. 581–586 (2007)
31. Stumptner, M., Wotawa, F.: Diagnosing tree-structured systems. *Artif. Intell.* **127**(1), 1–29 (2001)
32. West, G.: *Scale: The Universal Laws of Growth, Innovation, Sustainability, and the Pace of Life in Organisms, Cities, Economies, and Companies*. Penguin Press, New York (2017)
33. Yoo, T., Lafortune, S.: Polynomial-time verification of diagnosability of partially observed discrete-event systems. *IEEE Trans. Autom. Control* **47**(9), 1491–1495 (2002)



# Recognition of Handwritten Characters Using Google Fonts and Freeman Chain Codes

Alexiei Dingli<sup>(✉)</sup>, Mark Bugeja<sup>(✉)</sup>, Dylan Seychell<sup>(✉)</sup>, and Simon Mercieca

University of Malta, Msida, Malta

{alexiei.dingli,mark.bugeja,dylan.seychell}@um.edu.mt

**Abstract.** In this study, a unique dataset of a scanned seventeenth-century manuscript is presented which up to now has never been read or analysed. The aim of this research is to be able to transcribe this dataset into machine readable text. The approach used in this study is able to convert the document image without any prior knowledge of the text. In fact, the training set used in this study is a synthetic dataset built on the Google Fonts database. A feed forward Deep Neural Network is trained on a set of different features extracted from the Google Font character images. Well established features such as ratio of character width and height as well as pixel count and Freeman Chain Code is used, with the latter being normalised using Fast Fourier Normalisation that has yielded excellent results in other areas but never been used in Handwritten Character Recognition. In fact, the final results show that this particular Freeman Chain Code feature normalisation yielded the best results achieving an accuracy of 55.1% which is three times higher than the standard Freeman Chain Code normalisation method.

**Keywords:** Handwritten Character Recognition  
Machine learning · Deep learning

## 1 Introduction

Handwritten Character Recognition (HCR) converts a set of segmented characters into digital format. Most HCR techniques make use of Machine Learning models as well as computer vision to accomplish this task. The approach used in this study is a deep learning approach where a unique data set, built using Google Fonts is initially created. A number of different variations of the different fonts were generated in order to further increase the dataset using augmentation techniques. This step is necessary as early studies have shown that Machine Learning models tend to over-fit particular features, such as placement of the character in the image, rotation and thickness of the generated character. The final documents are then processed to extract various features. The feature set uses a variety of features that are extracted from the training set and then passed

as an input to a Deep Neural Network (DNN) model that uses a variation of hidden layers. This model in turn is able to use different variables extracted from the feature set to develop a probabilistic classifier that is able to classify various characters within the document images. The Google Fonts dataset was created in order to make use of the variations found in different fonts to mimic variations in handwritten documents. Thus, we can create a more robust model that is able to transcribe documents written in different handwriting styles that were not used to train the model. Most models make use of some of pre-labelled documents as a training set. It is very time-consuming process to manually transcribe a subset of documents, and even more time consuming to transcribe the whole set. Furthermore, it is not always the case that there are sufficient character-images in hand that can be used to train a Deep Learning Model.

### 1.1 The Manuscript

The dataset referred to as the manuscript is a 512-page document written in the 17th century. It is an unofficial history of the Knights of Malta written by Salvatore Imbroli. There is no known digital analysis of the said manuscript. Analysing it from a digital point of view requires the overcoming of a number of hurdles related to the script, ink and paper used at the time. Problems arising from the manuscript include small variations in writing style and the quality of the scans themselves. In this paper we are not addressing the challenge of segmenting the documents and evaluating the character recognition on the segmented characters. That challenge is addressed in a dissertation researched by Dylan Seychell. The final evaluation results used are a set of manually segmented characters selected from 25 randomly selected document images. The manuscript was provided by the National Library of Malta, thanks to the collaboration established by Dr. Simon Mercieca with Ms. Maroma Camilleri. The research and transcription of specimen pages from this manuscript was done by Dr. Simon Mercieca with the collaboration of students following the history course at the Faculty of Arts of the University of Malta (Fig. 1).

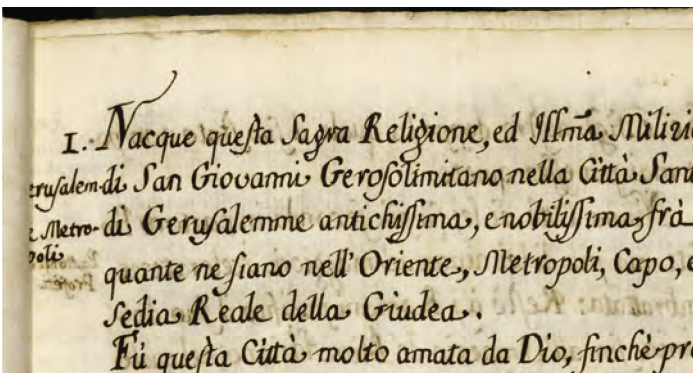


Fig. 1. Partial document image taken from the Manuscript



## 2 Background

### 2.1 Introduction

Artificial Neural Networks have been successfully used in a number of HCR models. In particular, supervised learning models have yielded excellent results. In the following section a number of models used in HCR systems are evaluated including Feed Forward Networks, Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN).

### 2.2 Feature Based Models

One of the most important components of Machine Learning is the feature set used for training. Depending on the number of features and the quality of the features, the models accuracy is increased. Moreover, the larger the dataset and more varied the dataset, the more generalized the model becomes. Some of the features used in handwritten recognition include the following.

**Diagonal Feature Extraction Scheme:** Each individual character image is resized to  $90 \times 60$  pixels it is then further divided into  $10 \times 10$  pixel bins. Features are extracted from the pixels for each bin by going through each diagonal for all available bins. A set of 54 features are extracted using this scheme [14].

**Contour Feature Scheme:** Character images are converted to a binary image. The contour of the image is extracted. Algorithms such as contour analysis can be used for this step. Each contour point is now a feature. For a given character there might be multiple contour points. When using such a scheme it is not always the case that for each image the same number of contour points are extracted. Thus, the feature set is normalized the gain the same set of features for training using Artificial Neural Networks. In addition, direction of pixels can also be used as part of the contour feature scheme feature set [9].

**Freeman Chain Codes Scheme:** Freeman chain code algorithms go through the edge of a character image and extract the direction each pixel takes with respect to the contour of the shape of the character image. This scheme has yielded excellent results with accuracy going over 90%. As in the contour feature scheme the extracted freeman chain codes need to be normalised [13].

**Curvlet Transform Based Feature Scheme:** The authors of this [18] noted the importance of handwritten text orientation written by the writer. Based on the needle shaped elements from the edge along the curves of the text. The extracted elements contain the directional sensitivity found in smooth contours. Character images are resized to a standard width and height. The Curvlet feature coefficients is extracted from each document character image. Apart from the above features there are a number of features that are considered standard in feature extraction. Including, character image width, height, centroid of the character as well as black pixel count.

### 2.3 Convolutional Neural Networks

Convolutional Neural Networks (CNN) are a class of feed forward networks. These networks have been applied and have achieved excellent results in areas of image recognition. CNN use the concept of convolution in order to create an architecture using several layers of convolution and nonlinear activation functions [10, 17]. One of the most early convolutional network was pioneered by Yann LeCun in 1988 [12]. The resulting architecture called LeNet was used for character recognition for zip codes and numerical digits. What makes CNN attractive is the fact that no features apart from the images are used when training the network. In most of the other networks explored the quality and accuracy of the neural net is depended on the features extracted. The more information can be gained from the features the better the accuracy as well as the generalisation of the model. Whilst this is desirable most of the time we do not know what features work well or what we want to extract. CNN accept images as inputs and extract features through leveled steps of convolution. Each layer in the network performs a simple computational function and the result is fed to the subsequent layer with the final result fed to the classifier. The computational process done by each layer is achieved using back-propagation. This process specifies that for each variable, the difference in the classification loss with respect to that parameter is computed and the parameters are updated with the goal of minimizing the loss function. Simard et al. developed a CNN for handwritten digit recognition [17]. Their approach maximizes on the potential of this network by augmenting the dataset. Dataset augmentation is the process of adding distortion to the dataset to improve the generalization obtained by the neural net. The MNIST dataset was used for training and testing. The dataset was split into 60000 images for training and a further 10000 for testing. The overall architecture described makes use of two subsequent convolutional layers with a kernel size of  $5 \times 5$  and two fully connected layers. The first fully connected layer uses 100 hidden units for training and the final classification layer is made up of 10 nodes representing the 10 classes used to classify digits. The authors also describe than the first fully connected layer was varied in size according to the number classification used. In the case of handwritten Japanese characters the number of hidden nodes was changed two 400 units for optimal results. By using the distorted dataset, they managed to obtain an overall error of 0.4% which was considered as state of the art at the time. Deformed training set is also used in the CNN proposed by Ciresan et al. [3]. The architecture of the model uses an input layer of images size  $29 \times 29$ , where the original images are resized from  $128 \times 128$  to  $20 \times 20$  size image and centred over a  $29 \times 29$  blank image. A convolutional layer with a  $4 \times 4$  kernel followed by a max pooling layer with kernel size of  $2 \times 2$  connected to a  $9 \times 9$  kernel convolution layer and connected to a final max pooling layer with kernel size of  $3 \times 3$ . The final fully connected layer made up of 150 nodes and a final classification layer with a varying number of nodes depending on the dataset used. 62 classes for the NIST SD dataset and 10 for the MNIST dataset. The dataset is distorted at the beginning of every epoch iteration using an elastic deformation with a variable value of  $\alpha = 6, 36$  as well as

a vertical and horizontal scaling of range from 0 to 15% and distorted by  $\pm 15\%$ . An error of 11.8% is reported when classifying uppercase and lowercase letters. The presented architecture is able to classify around 10,000 characters per second. The above implementations trained CNN to classify English handwritten text. The method proposed by Rahman et al. is applied to handwritten Bangla characters [15]. The dataset used contains 20000 handwritten characters with 400 character images representing each character of the Bangla alphabet. The input layer of the network accepts grey scale image with dimension of  $28 \times 28$  pixels. The architecture of the network used includes 2 convolutional layers using a  $5 \times 5$  kernel, 2 max pooling layers using a  $2 \times 2$  kernel a fully connected layer with 192 nodes and the final output layer made up of 50 nodes representing all the letters in the Bangla alphabet. The dataset was split into 17,500 character images for training and 2500 character images for testing. The overall accuracy of the system obtained was that of 85.36% accuracy. The authors noted that although the results were promising they did not compare to results achieved by other machine learning algorithms [15].

## 2.4 Recurrent Neural Networks

Unlike feed forward networks, RNN do not form a forward connected cycle of nodes. In fact, using this architecture the network uses the parameters learned from the node as a new recurring input. Thus, being able learn how to classify character images from unsegmented handwritten text. This is due to a property in RNN's where the output is depended on the previous output. This 'memory' like property records information on what has been computed beforehand and predicts the next output. In most Neural Network models input  $x$  passed to the network corresponds to some label  $y$  in a one to one relationship. Using RNN this behaviour changes [5]. In fact, an input could be a sequence of data which corresponds to either one output or another sequence of outputs with varying size. Ultimately the output is depended on all the history of the inputs that had been fed to the network beforehand. In most handwritten text recognition systems segmentation and transcription are two completely different components. In this paper, the authors propose a system that combines segmentation and transcription using RNN [6]. The system is built up of 7 layers. The input layer accepts images as input. The images might not necessarily be character images rather whole words or even sentences. These images are split into small zones and converted to a one-dimensional vector. The input layer is connected to a Multidimensional Long Short Term Memory layer (LSTM). A standard LSTM is made up of three distinct gates. The input gate, forge gate and the output gate that is connected to one RNN. In Multidimensional LSTM, the connections have been extended to  $n$  recurrent connections for each of the nodes previous states. The resultant output is converted back to the size of the original zone and fed to a feed forward network which uses a tanh activation function. This process is repeated another two times up to which the final Multidimensional LSTM layer converts the output to a one-dimensional vector and transcribed to the Connectionist Temporal Classification layer. The latter output layer is

specifically designed classification layer used in RNN where it transforms the output into sequence labeling and does not require pre-segmentation of labels or post processing to convert to transcription values. The training data used were a set of 1518 images with 120 distinct characters. The system achieved an accuracy of 96.75% Jameel et al., made some observations on what input can be fed to the network [8]. Instead of feeding images a set of features were extracted and used as input. Jameel et al. argue that although curves, lines and intersections are intuitive features to extract the sequence in which they appear is also very important. Thus, shadow features were extracted by computing a sequence of values that depict what happens when scanning a character image and at what time curves and other features appear in the image. The sequence was then used as a feature set to input into the RNN. A back propagation neural network was used that is, a fully connected RNN. The training set was made up of 877 character images including uppercase and lower case letters. It took from 10,000,000 to 15,000,000 steps to train the network and achieve 91.4% accuracy [8].

The literature evaluated in this study established that a set of handcrafted features might give excellent results in the area of HCR. The adopted architecture used is a simple feed forward network which uses a substantial number of neurons and hidden layers. The features established from the literature used include Geometrical properties and Freeman Chain Codes.

### 3 Methodology

Some of the most successful techniques use Artificial Neural Networks (ANN) combined with a set of handcrafted features to recognise handwritten character images. Our approach differs in two ways. Instead of a shallow artificial neural network a deep neural network is used. The authors in [4] argue that the more hidden layers are used in a neural network the more accuracy can be gained. Consequently, a more varied dataset yields better generalisation. In the following section a detailed explanation of the training set used is presented as well as justification on the number of steps used for varying the dataset.

#### 3.1 Training Set

The aim of this study is to build a system that is able to automatically analyse handwritten text from the manuscript without any prior knowledge of the text. This implies that the classification model used needs to be robust enough to convert any character image into the corresponding ASCII value. This by no means is a trivial task. Many of the approaches undertaken by the authors in [9, 13, 14, 18] use the MNIST dataset to test and train the classifier but none of these approaches take into consideration approaches which test on a different dataset. Thus, it was decided to develop a new dataset computed from a number of fonts. The font database used is the google font database found in google Google Fonts database. A dataset creation step generates a character image for every character and font found in the font database. In order to maximise

generalisation a series of distortion steps are added to the creation stage this include rotation, scaling and moving the centre point of the character inside the image frame. Once a font style is loaded a random font size is selected. The selected range of font size varied this step is used to cater for characters with different widths and heights. This is an important step as variations in handwritten text might include variations in size of the same character. Thus, for example the letter ‘a’ written by the same author might not always have the same dimension. The next step varies the position of the letter with respect to the character image. It is highly unlikely that the extracted characters in the character segmentation module are centred directly at the centre of the image. In fact, due to the scan line approach most of the time the character position varies in the character image window space. This step was therefore introduced so that the classification model does not overfit the classification label with respect to the character position and start classifying character images according to this unwanted feature. In an ideal scenario, the model trains on the shape of the character image. Due to the steps used in the feature extraction phase a number of pre-processing techniques are applied to the dataset. The images were binarised using the fast 2-D Otsu thresholding algorithm [20]. This technique converts the character image into a black and white image using a 2 dimensional histogram projected on the diagonal of the image. The image is then thinned with the Zhang Suen thinning algorithm [19] converting the stroke of the character to a pixel of depth. Thus, retaining only the skeleton shape of each character. These steps are used to optimise on the FCC extraction process. This step produced a total of 110,584  $64 \times 64$  binarised character images.

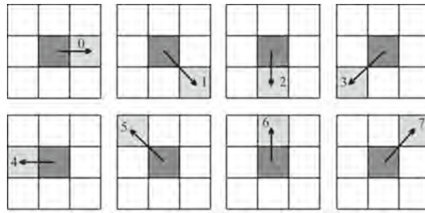
### 3.2 Feature Extraction

The second most important step in the classification module is the feature extraction. The overall accuracy of the classifier depends on the quality of the features extracted from the character images as well as the variety training set used. The following set of features compiled from the literature are used given the high accuracy reported by the authors using these features in [9, 14]. Furthermore, a number of experiments were undertaken using these features to find the best model to classify the handwritten characters. The experiments include variations in the feature schemes used as well as variation in the amount of training data and number of steps taken for the classifier to reach optimal performance. In the following section a description of the features and their corresponding extraction procedure is presented.

### 3.3 Freeman Chain Code

As mentioned throughout the paper FCC is an algorithm generally used to encode shapes in data compression. The algorithm lends itself well for edge detection. Another property of FCC is that the resulting encoding is a chain of values corresponding to the change in direction at the edge of the pixel. This is a desirable training feature as irrespective of the character position the direction

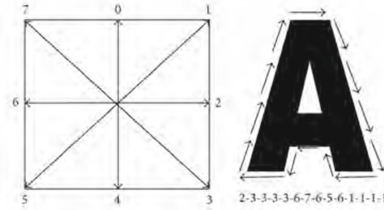
that the edge contour of a character shape has it will always remain the same. Couple with descriptive features such as the width and height of an image and the number of black pixels should lead to interesting results. In order to extract the FCC the character image is scanned pixel by pixel. Starting from the top left pixel moving towards the left direction. Once the initial transition is found, that is, finding the first black to white or white to black pixel change, then the start position is of the shape is noted. The transition value depends on whether the image background is black or white whilst the character has the opposite colour value (Fig. 2).



**Fig. 2.** 8-Connectivity direction matrix used in FCC

From the start position the method looks at the neighbour pixels and searches for pixel value of the same colour as the start position pixel. The search is done clockwise starting from the perpendicular (top centre in matrix) position. In order for the adjacent pixel to be accepted as a candidate for the chain code it needs to be classified as a border pixel. This step prevents the algorithm from going through the area of the shape rather than the edge. The algorithm repeats this process until it creates a closed loop of the shape and returns back to the start position pixel. Although in our approach we use a clockwise moving direction the algorithm would still work if the directions are captured in an anticlockwise manner. The important rule is that only one direction is take to capture all features within the dataset to keep consistency [16].

The final output of this system is two separate data objects. One data object contains the chain code and the directions of the edge pixels whilst the other object contains the set of points traversed at the border. It is important to note that at this stage depending on the character shape and size, the dimensions of the chain set and the border position set varies. In most machine learning approaches as well as in the model used for this study the model will accept a fixed set size of features. Given this, normalisation techniques are applied to the data sets so that irrespective of the output size of the dataset the dimensions are always consistent. Furthermore, 8-connectivity matrix is used for feature extraction. This is due to the fact that more shape variations can be captured using this method (Fig. 3).



**Fig. 3.** Result of Freeman Chain Code

### 3.4 Histogram Normalization

The chain code normalisation process is fairly simple. An array of 8 features is first created. For every value in the chain code set for the chain value that corresponds to a position in the array then the array value is incremented by 1. Effectively a histogram of values for the directions in the chain code is created.

### 3.5 Elliptic Fourier Feature Normalization

This normalisation technique is used to normalise the edge points of the character shape. Using elliptic Fourier features to describe a shape has been successfully applied to closed shapes in different research fields. The authors in [11] use four different co-efficient that represent each harmonic used to identify the closed shape. For a closed shape of  $k$  elements  $n$  harmonics are used. These co-efficients represent the major and minor projections of the  $x$ -axis and the  $y$ -axis. When this method is applied to the edge contour points a normalized set of 37 features is computed. Apart from the normalization advantage this method also produces a feature set that is invariant to any rotation, dilation or translation of the shape. Thus, irrespective of the shape position, stroke and rotation the same feature value is extracted each time. This property of elliptic Fourier feature normalisation is extremely desirable in the case of handwritten character recognition given that as outlined before variations in the testing set rotation or scale might vary the accuracy of the result. Such an implementation has been used and tested in different fields such as [2]. Up to the submission of this study no research has been submitted that use this normalisation for handwritten character recognition or compares the effects of using such procedure with respect to the histogram normalisation technique.

### 3.6 Geometrical Properties

The final set of features extracted from the character image describe the geometrical property of a character. Including the ratio of the width and height of a character image. The centroid of the character with respect to the image window as well as a count numbering the number of black pixels found in the character image. The latter is an important feature as through the thinning

and dilation normalisation process each character has the same stroke thickness. Thus, according the shape and size of character an estimation of the number of black pixels can be computed for different character images falling under the same class.

### 3.7 Network Architecture

The adopted machine learning model is a simple feed forward neural network. This model has shown in different implementation of handwritten text recognition to yield excellent results. Our approach aims to increase the accuracy of the system by using a deep neural network. There is no formal definition for describing a neural network as a deep neural network. The authors in [4] attempt to classify a neural network as a deep network depending on the number of hidden layers. The overall accuracy achieved by the deep neural network is depended on the number of neurons available in the hidden layer as well as the number of hidden layers. The authors in [7] argue that by increasing the number of neurons in the hidden layer higher accuracy is obtained. On the other hand, Increasing the number of neurons in the hidden layer also increase the probability that over-fitting occurs.

## 4 Evaluation and Results

In this section, an evaluation of the character recognition module is presented.

### 4.1 DNN Models

The DNN Classification models are split into two main models. Each model is trained on a different feature set. The features include:

1. Freeman Chain Code normalized using Histograms referred to as FCC-HIST.
2. Character Contours normalised using Elliptic Fourier Analysis referred to as FCC-EFT.

All of the feature sets also contain structural data that includes width and height of the characters and number of black pixels found in the character image. The datasets are split 35000 character images and 100000 character Images. The testing set is 10% of the training set as in Table 1.

The models are further divided into two architectures. In order to choose the architecture and number of neurons to use an empirical process of elimination was used. A number of models where evaluated using different configurations and trained for 10,000 steps. The models that achieved the most promising results were then chosen for further training. The final two architectures used included a shallow model with 2 layers and a deep model with 6 layers. The first architecture is a shallow model containing two hidden layers with 64 and 32 hidden neurons respectively for each hidden layer. The second architecture is a deep architecture



consisting of 6 layers with 1024, 512, 256, 128, 64, 52 hidden neurons. Finally, each model is trained for three iterations of 100,000 steps and the accuracy is calculated from the average accuracy obtained on the same configuration model. In the following sections a presentation of the results obtained for these models is presented in Table 1 as well as a discussion on the results obtained.

**Table 1.** Results obtained when evaluating the models

Model used	Accuracy (%) with training set of 35,000	Accuracy (%) with training set of 110,000
FCC_HIST 2 Layers	5.2	9.4
FCC_HIST 6 Layers	12.6	17.8
FCC_EFT 2 Layers	25.3	37.7
FCC_EFT 6 Layers	42.8	55.1

The overall accuracy of the system when classifying the handwritten character's segmented from the Manuscript is quite low. The best model was able to properly classify only 55.1% on the characters. A number of factors have contributed to this accuracy value. On inspection of the data although the characters were processed in the same manner as the testing set slight variations in the characters resulted in broken features or inconclusive feature extraction values. These include for example a chain code with only 3 values which is impossible as this would mean that the edge pixels of the character are made up of three directions only. In fact, when analysing the data, the character's which had a complete closing loop obtained better classification results with respect to the other characters with an increase in accuracy of 13%. Furthermore, characters such as the letter 's' was completely misclassified. This is mostly due to the way the letters were written in the manuscript. Figure 4 shows the letter 's' in various variations written in the manuscript.



**Fig. 4.** Comparison of letter 'f' (a) and 's' (b) found in the Manuscript

Even in today's writing style the character could easily be confused with an 'f'. Another reason which contributes to the low accuracy obtained in the final

testing phase of this study is that the model might have over trained on the testing set. More distortion and more character images would effectively result in better classifications. Moreover, other feature selection techniques need to be added such as zoning [14] and curvelet scheme [9].

Other approaches that might yield better results using different neural networks such as CNN and RNN. The latter were not chosen due to the requirement when used with handwritten text to have an initial labeled testing set. CNN were used at the start of the experimentation phase during the course of the research. The reason that the implementation was rejected is that although the training and testing results obtained were substantially better than the final model's when tested on the actual Manuscript the classification accuracy was a bit better than random distribution.

## 5 Conclusion

The primary objective of this study is the creation of a database of characters that can be used as a training set for the Manuscript. The database was created and distortion effects were applied to the images to try and improve accuracy of the final text recognition component. Although the aim is met there is a lot more that needs to be done to consider such an objective complete. When developing the database, a lot of issues such as fonts that were too 'fancy' to be used for training as well as problems with characters not having a shape that can be mapped along the edge as a complete loop resulted in a number of issues with the final accuracy of the system. A lot of manual intervention was needed to clean out the database. The final result was a dataset of 110,254 character images. Tests involving different number of machine learning model's need to be computed in order to completely measure how effective the database is for handwritten text classification and this includes evaluation on other datasets such as the MNIST dataset. A large part of this study was spent researching on different implementations of machine learning and deep learning models for HCR. There is a lot of research in the area and the state of the art results are impressive to say the least. Most of the research is done on datasets that have been already identified and labeled such as the MNIST dataset and builds on a lot of models that have yielded excellent results, but, have not been implemented upon real world data. The approach chosen was built on some of the models discussed in the background [9, 13]. These model's yielded excellent results using FCC feature extraction schemes coupled with contour analysis schemes and normalisation techniques using histograms. Our approach on the other hand uses elliptic Fourier features to normalise the contour features of a character. The latter normalisation technique is used in other fields but as of the time of the writing for this dissertation never featured in handwritten text recognition modules. The model adopting this scheme yielded the best results in combination with a deep learning architecture compromised of 6 hidden layers and 2036 hidden units. The best model obtained 55.1% accuracy after 100,000 epochs on the training set. A change in the configuration of the network might have yield better

results. The results obtained by the 2-layer Deep Neural Network was slightly less accurate that might prove the theory presented by Ba et al. [1] that suggest that shallow networks are able to achieve the result as deep networks. The issue mostly lies with the current implementations used for shallow networks. Thus, with other classification models the model results might have yielded an improved accuracy. On the other hand, Goodfellow et al. [4] provide empirical results that prove that the deeper a model's architecture the better accuracy is yielded over time. More experiments on the configuration of the network might yield better results. Another approach that can be researched on is using RNN. This approach was rejected in the dissertation as it required a number of labeled documents matching the manuscript to build a supervised model. On the other hand, a similar approach undertaken in the study using the Google Font database can be used. Instead of a number of characters a document is created using text found in Italian literature. A number of different fonts are applied on the text and distortion applied on the text lines to create a more natural handwritten text effect. The model is then trained on these set of already labeled documents and evaluated on the system. The overall aim of creating a system that goes through the motions of converting a set of unseen handwritten character images to an ASCII representation has been met the results show that using FTT increases recognition accuracy by 37.3%.

## References

1. Ba, J., Caruana, R.: Do deep nets really need to be deep? In: *Advances in Neural Information Processing Systems*, pp. 2654–2662 (2014)
2. Ballaro, B., Reas, P., Tegolo, D.: Elliptical fourier descriptors for shape retrieval in biological images. In: *International Conference on Electronics, Control & Signal Processing*. SG (2002)
3. Ciresan, D.C., Meier, U., Gambardella, L.M., Schmidhuber, J.: Convolutional neural network committees for handwritten character classification. In: *2011 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1135–1139. IEEE (2011)
4. Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y.: *Deep Learning*, vol. 1. MIT Press, Cambridge (2016)
5. Graves, A., Jaitly, N.: Towards end-to-end speech recognition with recurrent neural networks. In: *International Conference on Machine Learning*, pp. 1764–1772 (2014)
6. Graves, A., Schmidhuber, J.: Offline handwriting recognition with multidimensional recurrent neural networks. In: Koller, D., Schuurmans, D., Bengio, Y., Bottou, L. (eds.) *Advances in Neural Information Processing Systems 21*, pp. 545–552. Curran Associates, Inc. (2009). <http://papers.nips.cc/paper/3449-offline-handwriting-recognition-with-multidimensional-recurrent-neural-networks.pdf>
7. Hornik, K.: Approximation capabilities of multilayer feedforward networks. *Neural Netw.* **4**(2), 251–257 (1991)
8. Jameel, A.: Experiments with various recurrent neural network architectures for handwritten character recognition. In: *Proceedings of the Sixth International Conference on Tools with Artificial Intelligence*, pp. 548–554. IEEE (1994)
9. Kimura, F., Shridhar, M.: Handwritten numerical recognition based on multiple algorithms. *Pattern Recogn.* **24**(10), 969–983 (1991)

10. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*, pp. 1097–1105 (2012)
11. Kuhl, F.P., Giardina, C.R.: Elliptic fourier features of a closed contour. *Comput. Graph. Image Process.* **18**(3), 236–258 (1982)
12. LeCun, Y., Touresky, D., Hinton, G., Sejnowski, T.: A theoretical framework for back-propagation. In: *Proceedings of the 1988 Connectionist Models Summer School*, CMU, Pittsburgh, PA, pp. 21–28. Morgan Kaufmann (1988)
13. Nasien, D., Omar, F.S., Azmi, A.N., Yulianti, D.: Freeman chain code route length optimization using meta-heuristic techniques for handwritten character recognition (2015)
14. Pradeep, J., Srinivasan, E., Himavathi, S.: Diagonal feature extraction based handwritten character system using neural network. *Int. J. Comput. Appl.* **8**(9), 17–22 (2010). (0975–8887)
15. Rahman, M.M., Akhand, M., Islam, S., Shill, P.C., Rahman, M.H.: Bangla handwritten character recognition using convolutional neural network. *Int. J. Image Graph. Signal Process.* **7**(8), 42 (2015)
16. Shahab, W., Al-Otum, H., Al-Ghoul, F.: A modified 2D chain code algorithm for object segmentation and contour tracing. *Int. Arab J. Inf. Technol.* **6**(3), 250–257 (2009)
17. Simard, P.Y., Steinkraus, D., Platt, J.C., et al.: Best practices for convolutional neural networks applied to visual document analysis. In: *ICDAR*, vol. 3, pp. 958–962 (2003)
18. Singh, B., Mittal, A., Ansari, M., Ghosh, D.: Handwritten Devanagari word recognition: a curvelet transform based approach. *Int. J. Comput. Sci. Eng.* **3**(4), 1658–1665 (2011)
19. Zhang, T., Suen, C.Y.: A fast parallel algorithm for thinning digital patterns. *Commun. ACM* **27**(3), 236–239 (1984)
20. Zhu, N., Wang, G., Yang, G., Dai, W.: A fast 2D Otsu thresholding algorithm based on improved histogram. In: *Chinese Conference on Pattern Recognition, CCPR 2009*, pp. 1–5. IEEE (2009)



# An Efficient Approach for Extraction Positive and Negative Association Rules from Big Data

Bemarisika Parfait<sup>1,2(✉)</sup>, Ramanantsoa Harrimann<sup>1</sup>, and Totohasina André<sup>1</sup>

<sup>1</sup> Laboratoire de Mathématiques et d'Informatique,  
ENSET, Université d'Antsiranana, Antsiranana, Madagascar  
bemarisikap7@yahoo.fr, ramana\_riri@yahoo.fr, andre.totohasina@gmail.com

<sup>2</sup> Laboratoire d'Informatique et de Mathématiques, EA2525,  
Université de La Réunion, Saint-Denis, France

**Abstract.** Mining association rules is an significant research area in Knowledge Extraction. Although the negative association rules have notable advantages, but they are less explored in comparaison with the positive association rules. In this paper, we propose a new approach allowing the mining of positive and negative rules. We define an efficient method of support counting, called *reduction-access-database*. Moreover, all the frequent itemsets can be obtained in a single scan over the whole database. As for the generating of interesting association rules, we introduce a new efficient technique, called *reduction-rules-space*. Therefore, only half of the candidate rules have to be studied. Some experiments will be conducted into such reference databases to complete our study.

**Keywords:** Big Data · Extraction association rules  
Reduction-access-database · Reduction-rules-space

## 1 Introduction and Motivations

Since Agrawal's work [1], the extraction of association rules has been on of the most popular techniques for in Knowledge Extraction. An association rule is an implication of the form "if **Condition** then **Result**". Association rules may be used for store layout, target marketing, organize promotions of the supermarket, etc. In the literature, there exist two types of association rules: positive and negative rules. An association rule is said to be positive when it considers the presence of variables. It is negative when it considers the absence of these same variables. Although the negative rules have obvious advantages [6, 10], they remain less explored in comparaison with positive rules. One of the major disadvantages lies in their difficult extraction, this type increases the exponential costs. Besides, the current approaches [10, 11, 15, 16, 18] are limited on the Apriori's data structure and support-confidence pair. While, this data structure

imposes the repetitive accesses over the database, which can be costly. In addition, the support-confidence pair is questionable: (i) finding frequent itemsets is very complex in large databases and/or for low minimum support threshold; (ii) the number of rules that can be reduced nevertheless remains high that many prove uninteresting. In order to exceed these notable limits, we propose an efficient approach for mining positive and negative association rules using a new pair, *support- $M_{GK}$* . We introduce a new economical technique of support counting, called *reduction-access-database*, based on the new data structure MATRIXSUPPORT and generator concepts. Therefore, a simple pass allows us to extract all the frequent itemsets over the whole database. As for association rules generating, we introduce an efficient method, called *reduction-rules-space*, partitioning the search space rules. Therefore, only half of the candidate rules are to study. Based on these optimizations, we also propose ERAPN algorithm, less consumer in memory. We present the experimental evaluation conducted with databases from the literature by showing performances compared to semantically close approach such that RAPN algorithm [14] and Wu's algorithm [19].

The rest of this paper is organized as follows. Section 2 introduces the formal concepts. Section 3 details our approach. Section 4 summarizes our experimental results. Section 5 reviews the related work. A conclusion is given in Sect. 6.

## 2 Preliminaries Concepts

This section describes association rules terminology (Subsect. 2.1) and limits of the support-confidence pair (Subsect. 2.2).

### 2.1 Association Rules and Terminology

A transactional context is a triple  $\mathcal{B} = (\mathcal{T}, \mathcal{I}, \mathcal{R})$ , where  $\mathcal{T}, \mathcal{I}$  and  $\mathcal{R}$  are finite and not empty sets. An element of  $\mathcal{I}$  is called item (or attribute). A set of items, called an itemset. An element of  $\mathcal{T}$  is called transaction (or object) represented by a TID-Transaction Identifier, and  $\mathcal{R}$  is a binary relationship between  $\mathcal{T}$  and  $\mathcal{I}$ . So,  $|\mathcal{T}|$  and  $|\mathcal{I}|$  denotes the total number of transactions and items respectively. The table below represents an example. Given  $X, Y \subseteq \mathcal{I}$ ,  $\neg X = \overline{X} = \mathcal{I} \setminus X = \{t \in \mathcal{T} \mid \exists i \in X : (i, t) \notin \mathcal{R}\}$  is called the logical negation of  $X$ . For example, with the Table 1, we have  $AB = \{3, 5\}$ , so  $\overline{AB} = \{1, 2, 4, 6\}$ . A  $k$ -itemset is an itemset of length  $k$ . We will use the correspondances (Galois connections [12])  $g(\mathcal{I}) = \{t \in \mathcal{T} \mid \forall i \in \mathcal{I}, i\mathcal{R}t\}$  and  $f(\mathcal{T}) = \{i \in \mathcal{I} \mid \forall t \in \mathcal{T}, i\mathcal{R}t\}$ . The function  $g$  is antimonotony: for all  $X, Y \subseteq \mathcal{I}$ , if  $X \subseteq Y$  then  $g(Y) \subseteq g(X)$ . It is clear that if  $X \subseteq Y$  then  $supp(X) \geq supp(Y)$ . The applications  $\gamma = fog$  and  $\gamma' = gof$  are Galois closure operators. An itemset  $X$  is closed if  $X = \gamma(X)$ .

A positive rule is an implication of the form  $X \rightarrow Y$ . It is called negative rule which consider the absence of the item, i.e.,  $X \rightarrow \overline{Y}$ ,  $\overline{X} \rightarrow Y$  and  $\overline{X} \rightarrow \overline{Y}$ , where  $X \cap Y = \emptyset$ .  $X$  is called the premise and  $Y$  the conclusion. To determine an association rule interesting, two measures are used, support and confidence [1]. The support of  $X$  is the number of transactions that contain  $X$ , defined

**Table 1.** Example of the transactional context  $\mathcal{B}$

TID	Items	Positive and negative items	Equivalent binary
1	ACD	A¬BCD¬E	10110
2	BCE	¬ABC¬DE	01101
3	ABCE	ABC¬DE	11101
4	BE	¬AB¬C¬DE	01001
5	ABCE	ABC¬DE	11101
6	BCE	¬ABC¬DE	01101

as  $supp(X) = \frac{|\{t \in \mathcal{T} | X \subseteq t\}|}{|\mathcal{T}|} = \frac{|g(X)|}{|\mathcal{T}|}$ . Denoting by  $P$  the intuitive probability measure defined on  $(\mathcal{T}, \mathcal{P}(\mathcal{T}))$  by  $P(Z) = \frac{|Z|}{|\mathcal{T}|}$  for  $Z \subseteq \mathcal{T}$ , the support of  $X$  can be written in terms of  $P$  as  $supp(X) = P(X)$ . The item  $X$  is said to be frequent if its support exceeds a minimum support threshold value,  $minsup \in [0, 1]$ , i.e.  $supp(X) \geq minsup$ . The support and confidence of  $X \rightarrow Y$  are defined as  $supp(X \cup Y) = \frac{|g(X \cup Y)|}{|\mathcal{T}|} = \frac{|g(X) \cup g(Y)|}{|\mathcal{T}|}$  and  $conf(X \rightarrow Y) = P(Y|X) = \frac{supp(X \cup Y)}{supp(X)}$ , respectively. Thereafter, we will omit the sign union and sometimes write  $\overline{XY}$  instead of  $X \cup Y$ . According to Morgan, we obtain, for all  $X, Y \subseteq \mathcal{I}$ ,  $supp(\overline{X}) = 1 - supp(X)$ ,  $supp(X\overline{Y}) = supp(X) - supp(XY)$ ,  $supp(\overline{X}Y) = supp(Y) - supp(XY)$  and  $supp(\overline{X}\overline{Y}) = 1 - supp(X) - supp(Y) + supp(XY)$ .

### 2.2 Limit of Support-Confidence Pair

Despite its notable contribution, this pair support-confidence easily selects uninteresting association rules (independence stochastic between two itemsets (For all  $X, Y \subseteq \mathcal{I}, P(Y|X) = P(Y)$ ), or dependence negative ( $P(Y|X) < P(Y)$ )). The examples in Table 2 illustrate this. In this case, the first four columns give the characteristics of the purchase of products A and B, the last four indicate those of the purchase of coffee and tea. We obtain  $supp(A \cup B) = 0.72$  and  $P(B|A) = 0.9$ . These reasonably high values lead us to believe that the persons buying A also buy B. However, we find that the confidence is equal to the probability of the conclusion regardless of the premise (i.e.  $P(B|A) = P(B)$ ), it is a stochastic independence between A and B. The rule  $A \rightarrow B$  that seemed interesting is therefore misleading. On the other hand, we obtain  $supp(tea \cup coffee) = 0.2$ ,

**Table 2.** Limit of the couple support-confidence

	A	¬A	∑		coffee	¬coffee	∑
B	72	18	90	tea	20	5	25
¬B	8	2	10	¬tea	70	5	75
∑	80	20	100	∑	90	10	100

which assumes that tea favors coffee. However, the share of people buying coffee regardless of whether they also buy tea is higher, it is a negative dependence between tea and coffee. The rule  $\text{tea} \rightarrow \text{coffee}$  that seemed interesting is therefore misleading. That’s why the support-confidence couple sometimes extracts uninteresting rules. The use of other more effective measures is imperative.

### 3 Mining Positive and Negative Association Rules

In this section, we introduce our approach for mining positive and negative association rules. It describes in a double problematic: finding frequent itemsets and generating potential valid association rules based on the previously extracted frequent itemsets. The first problem is often complex (in the worst case, it reaches  $2^{|\mathcal{I}|}$ ) and dramatic when one considers the negative items. With the small database from Table 1, we have 1024 different items instead of 32 positive. The second problem is also complex (for an  $m$ -itemset, we have  $5^m - 2(3^m) + 1$  instead of  $3^m - 2^{m+1} + 1$ ). From Table 1, we have 2640 different rules instead of 180 classical rules. In these dimensions, it is necessary to select only a part. In [5, 8], we have initiated the solution, which will be refined in Subsects. 3.1 (*reduction-access-database* method) and 3.2 (*reduction-rules-space* method).

#### 3.1 Mining Frequent Itemsets: Reduction-Access-Database

This is based on two steps: finding (in a single scans) frequent 1 and 2-itemsets, and frequent  $k$ -itemsets ( $k \geq 3$ ). After that first step, frequent 2-itemsets are used to generate candidate 3-itemsets. The process continues until no more candidate can be generated. Given a *minsup*, finding the set of frequent itemsets  $\mathcal{F}$ , defined:

$$\mathcal{F} = \{X \subseteq \mathcal{I} | X \neq \emptyset \wedge \text{supp}(X) \geq \text{minsup}\}. \tag{1}$$

As noted, mining frequent itemsets is very complex. The worst case concerns the small itemsets (1 and 2-itemsets). To answer this, we develop a new data structure MATRIXSUPPORT. The following Table 3 describes its formalism on the small database from Table 1. It is a projection of database  $\mathcal{B}$  in relation to its attributes. The idea is to acquire data as the structure develops and store it. To each attribute corresponds a cell of the matrix to which we associate the absolute

**Table 3.** Formalism of the MATRICESUPPORT in dataset  $\mathcal{B}$

TID	Attributes
1	ACD
2	BCE
3	ABCE
4	BE
5	ABCE
6	BCE

Scan the database  $\mathcal{B}$

MATRICESUPPORT					
$i \setminus j$	A	B	C	D	E
A	3	2	3	1	2
B	-	5	4	0	5
C	-	-	5	1	4
D	-	-	-	1	0
E	-	-	-	-	5



support, noted  $|v_{ij}|$ , expressing the number of times the item  $v_j$  appears with the item  $v_i$ , where  $i$  (resp.  $j$ ) denotes the  $i$ -th line (resp.  $j$ -th column) of table. This is then used to identify the relative support, defined by:

$$\text{supp}(v_{ij}) = |v_{ij}|/|\mathcal{B}|. \quad (2)$$

For example, in Table 3,  $\text{supp}(v_{11}) = \text{supp}(A) = |v_{11}|/6 = 3/6$ ,  $\text{supp}(v_{12}) = \text{supp}(AB) = |v_{12}|/6 = 2/6$  and  $\text{supp}(v_{23}) = \text{supp}(BC) = |v_{23}|/6 = 4/6$ . For this technique, the supports of the small itemsets are retrievable in a simple pass over the whole dataset  $\mathcal{B}$ . As we mentioned, the generation of candidate  $k$ -itemsets is obtained from the frequent  $(k-1)$ -itemsets. To this end, the support of the candidate will be calculated as follows using the generator concept. An itemset  $X$  is a generator if it's minimal (of set inclusion) in its equivalence class. Its equivalence class is given by  $[X] = \{X' \subseteq \mathcal{I} | \gamma(X') = \gamma(X)\}$ . Note that the computational cost of closures is very exponential. However, the following lemma exploits the monotony of support upon set inclusion.

**Lemma 1.**  $\forall X, Y \in \mathcal{I}$ , if  $X \subseteq Y$  and  $\text{supp}(X) = \text{supp}(Y)$ , then  $\gamma(X) = \gamma(Y)$ .

This Lemma 1 indicate that an itemset  $X$  is generator if it has no proper subset with the same support. For example, from the Table 3,  $\text{supp}(AB) = \text{supp}(ABC) = \text{supp}(ABE) = \text{supp}(ABCE) = 2/6$ , we have  $\gamma(AB) = \gamma(ABC) = \gamma(ABE) = \gamma(ABCE)$ . Because,  $AB$  is minimal, then it is generator. If the candidate is a not generator, it will be calculated using the following Proposition 1.

**Proposition 1.** For all  $X$  non generator,  $\text{supp}(X) = \min\{\text{supp}(X') | X' \subset X\}$ .

*Proof.* Let  $\mathcal{I}$  be a set items. Let  $X$  and  $X_1$  be two itemsets on  $\mathcal{I}$  such that  $X_1 \subseteq X$ . Due to the monotonicity of support, we have  $\text{supp}(X) \leq \text{supp}(X_1)$ . In addition (by assumption),  $X$  is not generator, it exists  $X' \subseteq X$  on  $\mathcal{I}$  such that  $\text{supp}(X') = \text{supp}(X)$ . However,  $\text{supp}(X_1)$  is minimal in  $\mathcal{I}$ , so  $\text{supp}(X_1) < \text{supp}(X')$ . Finally,  $\text{supp}(X) = \text{supp}(X_1) = \min\{\text{supp}(X') | X' \subset X\}$ .  $\square$

The support of a non generator  $k$  size is exactly the smallest support of its  $(k-1)$ -subsets. For example, from the Table 3, we have  $\text{supp}(AC) = \text{supp}(A) = 3/6$ , therefore  $AC$  and its superset  $ABC$  are not generators itemsets. However, the superset of its subset  $ABC$  is then obtained by  $\text{supp}(ABC) = \min\{2/6, 3/6, 4/6\} = 2/6$ . The following properties generalizes this observation.

*Property 1.* Given  $X \subseteq \mathcal{I}$ , if  $X$  is a generator, then  $\forall Y \subseteq X$ ,  $Y$  is a generator, whereas if  $X$  is not a generator,  $\forall Z \supseteq X$ ,  $Z$  is not a generator.

**Theorem 1.** Any subset of a generator itemset must also be a generator. Any superset of a nongenerator itemset must also be nongenerator.

*Proof.* Let  $X$  and  $Z$  be two itemsets on  $\mathcal{I}$  satisfy  $X \subseteq Z$ . It exists an itemset  $Y \subseteq \mathcal{I}$  ( $Y \neq \emptyset$ ), disjoint of  $X$  such that  $Z = X \cup Y$ . If  $X$  is assumed to be a non

generator itemset, then it admits a proper subset  $T$  ( $T \neq \emptyset$ ) that is equivalent to it  $T \subseteq X$  and  $T \approx X$ , giving  $T \cup Y \approx X \cup Y$ . By hypothesis,  $X \cap Y = \emptyset$ , so  $T \cup Y \subseteq X \cup Y$ , The itemset  $Z$  is equivalent to a proper subset  $T \cup Y$ , so it is not generator. The contrapose gives the result.  $\square$

This theorem is central in search space of frequent itemsets, no pass is done if a candidate is not generator. Only the generator are generated from database.

### 3.2 Generating Association Rules: Reduction-Rules-Space

The most common framework in the association rules generation is the support-confidence pair. As we already mentioned (see Subsect. 2.2), this pair allow the pruning of many associations that are discovered in data, there are cases when many uninteresting may be produced. As such, we use the new pair support- $M_{GK}$ . The next paragraph introduces the new measure,  $M_{GK}$  [13,17,19].

Given  $X, Y \subseteq \mathcal{I}$ , such that  $X \cap Y = \emptyset$ ,  $M_{GK}$  of  $X \rightarrow Y$  is defined by:

$$M_{GK}(X \rightarrow Y) = \begin{cases} \frac{P(Y|X)-P(Y)}{1-P(Y)}, & \text{if } X \text{ favors } Y, P(Y) \neq 1 \\ \frac{P(Y|X)-P(Y)}{P(Y)}, & \text{if } X \text{ disfavors } Y, P(Y) \neq 0. \end{cases} \quad (3)$$

In equation 3,  $X$  favors (resp. disfavors)  $Y$  indicate  $P(Y|X) > P(Y)$  (resp.  $P(Y|X) < P(Y)$ ). In our approach, an association rule  $X \rightarrow Y$  is positive exact if  $M_{GK}(X \rightarrow Y) = 1$ , else, it is positive approximate rule. The following definition defines the interesting and uninteresting rules.

**Definition 1.** *Given  $X, Y \subseteq \mathcal{I}$ , an association rule  $X \rightarrow Y$  is interesting if  $P(Y|X) > P(Y)$ , it's not interesting if  $P(Y|X) \leq P(Y)$ .*

The range of values for  $M_{GK}$  varies in  $[-1, 1]$ . Two zones are present: *attractive zone* and *repulsive zone*. The first is a zone that ranges from independence ( $P(Y|X) = P(Y)$ ) to logical implication ( $P(Y|X) = 1$ ). The second is a zone that ranges from incompatibility ( $P(Y|X) < P(Y)$ ) to independence. If  $M_{GK}(X \rightarrow Y) = 1$ , then  $X$  and  $Y$  are strongly correlated, which denotes the logical implication between  $X$  and  $Y$ . Moreover, the rule  $X \rightarrow Y$  is exact. Similarly, if  $M_{GK}(X \rightarrow Y) = -1$ , then  $X$  and  $Y$  are incompatible. This corresponds to the repulsion limit between  $X$  and  $Y$ . If  $M_{GK}(X \rightarrow Y) = 0$ , then  $X$  and  $Y$  are stochastically independant, moreover, the rule  $X \rightarrow Y$  is not interesting. If  $-1 \leq M_{GK}(X \rightarrow Y) < 0$ ,  $Y$  is negatively dependent on  $X$ . Similarly, if  $0 < M_{GK}(X \rightarrow Y) \leq 1$ , then  $Y$  is positively dependent on  $X$ .

Let  $minsup \in [0, 1]$  and  $minmgk \in [0, 1]$  be two minimum thresholds of support and  $M_{GK}$ , respectively. The rule  $X \rightarrow Y$  is said to be valid according to our approach if its support  $supp(X \cup Y)$  is frequent and  $M_{GK}(X \rightarrow Y) \geq minmgk$ . The set of all valid association rules from  $\mathcal{B}$  is denoted  $\mathcal{E}_{RAPN}$ , formally:

$$\mathcal{E}_{RAPN} = \{X, Y \in \mathcal{I} | supp(X \cup Y) \geq minsup \ \& \ M_{GK}(X \rightarrow Y) \geq minmgk\}. \quad (4)$$

For the sake of comprehension, we apply this model on a same example in Table 1. The minimum support (resp. *minmgk*) is equal to 0.1 (resp. 0.8). Because,  $M_{GK}(A \rightarrow B) = 0 < 0.8$ , then  $A$  and  $B$  are stochastically independent, the association rule  $A \rightarrow B$  is invalid. Moreover, it is not added in  $\mathcal{E}_{RAPN}$ . But,  $supp(\overline{A} \cup B) = supp(B) - supp(A \cup B) = 0.90 - 0.72 = 0.18 > 0.1$  and  $M_{GK}(\overline{A} \rightarrow B) = 0.88 > 0.8$ , the rule  $\overline{A} \rightarrow B$  is valid, it added in  $\mathcal{E}_{RAPN}$ . On the other hand, one has  $M_{GK}(\text{tea} \rightarrow \text{coffee}) < 0$ , coffee is negatively dependant on tea. This is a situation we should consider the negative association rules.

In the following paragraph, we present our strategies for elimination of uninteresting association rules from  $\mathcal{B}$ . We show that only half candidates are to study by using the new technique, *reduction-rules-space*. Indeed, we are interested in partitioning the search space as shown in the following Proposition 2.

**Proposition 2.** *For all  $X, Y \in \mathcal{I}$ , (1)  $X \text{ fav } Y \Leftrightarrow Y \text{ fav } X \Leftrightarrow \overline{X} \text{ fav } \overline{Y} \Leftrightarrow \overline{Y} \text{ fav } \overline{X}$ . (2)  $X \text{ disfav } Y \Leftrightarrow X \text{ fav } \overline{Y} \Leftrightarrow \overline{Y} \text{ fav } X \Leftrightarrow Y \text{ fav } \overline{X} \Leftrightarrow \overline{X} \text{ fav } Y$ .*

*Proof.* Let  $X$  and  $Y$  be items of  $\mathcal{I}$ . We first prove, (a)  $X$  favors  $Y \Leftrightarrow Y$  favors  $X$ , (b)  $X$  favors  $Y \Leftrightarrow \overline{X}$  favors  $\overline{Y}$  and (c)  $X$  favors  $Y \Leftrightarrow \overline{Y}$  favors  $\overline{X}$ . In second time, (a)  $X$  disfavors  $Y \Leftrightarrow X$  favors  $\overline{Y}$ , (b)  $X$  disfavors  $Y \Leftrightarrow \overline{Y}$  favors  $X$ , (c)  $X$  disfavors  $Y \Leftrightarrow Y$  favors  $\overline{X}$  and (d)  $X$  disfavors  $Y \Leftrightarrow \overline{X}$  favors  $Y$ .  
 1(a)  $X$  favors  $Y \Leftrightarrow P(Y|X) > P(Y) \Leftrightarrow \frac{supp(X \cup Y)}{supp(X)} > supp(Y) \Leftrightarrow \frac{supp(X \cup Y)}{supp(Y)} > supp(X) \Leftrightarrow P(X|Y) > P(X) \Leftrightarrow Y$  favors  $X$ . (b)  $X$  favors  $Y \Leftrightarrow supp(X \cup Y) > supp(X)supp(Y) \Leftrightarrow 1 - supp(X) - supp(Y) + supp(X \vee Y) > 1 - supp(X) - supp(Y) + supp(X)supp(Y) \Leftrightarrow 1 - supp(X \wedge Y) > (1 - supp(X))(1 - supp(Y)) \Leftrightarrow supp(\overline{X} \wedge \overline{Y}) > supp(\overline{X})supp(\overline{Y}) \Leftrightarrow \frac{supp(\overline{X} \vee \overline{Y})}{supp(\overline{X})} > supp(\overline{Y}) \Leftrightarrow P(\overline{Y}|\overline{X}) > P(\overline{Y}) \Leftrightarrow \overline{X}$  favors  $\overline{Y}$ . (c)  $X$  favors  $Y \Leftrightarrow supp(\overline{X} \vee \overline{Y}) > supp(\overline{X})supp(\overline{Y})$  implies  $\frac{supp(\overline{X} \vee \overline{Y})}{supp(\overline{Y})} > supp(\overline{X}) \Leftrightarrow P(\overline{X}|\overline{Y}) > P(\overline{X}) \Leftrightarrow \overline{Y}$  favors  $\overline{X}$ . 2(a)  $X$  disfavors  $Y \Leftrightarrow P(Y|X) < P(Y) \Leftrightarrow 1 - P(Y|X) > 1 - P(Y) \Leftrightarrow P(\overline{Y}|X) > P(\overline{Y}) \Leftrightarrow X$  favors  $\overline{Y}$ . (b)  $X$  disfavors  $Y \Leftrightarrow P(\overline{Y}|X) > P(\overline{Y}) \Leftrightarrow \frac{supp(X \cup \overline{Y})}{supp(\overline{Y})} > supp(X) \Leftrightarrow P(X|\overline{Y}) > P(X) \Leftrightarrow \overline{Y}$  favors  $X$ . (c)  $X$  disfavors  $Y \Leftrightarrow Y$  disfavors  $X \Leftrightarrow P(X|Y) < P(X) \Leftrightarrow P(\overline{X}|Y) > P(\overline{X}) \Leftrightarrow Y$  favors  $\overline{X}$ . (d)  $X$  disfavors  $Y \Leftrightarrow P(\overline{X}|Y) > P(\overline{X}) \Leftrightarrow P(Y|\overline{X}) > P(Y) \Leftrightarrow \overline{X}$  favors  $Y$ .  $\square$

This is if  $X$  favors  $Y$  ( $P(Y|X) > P(Y)$ ), then only  $X \rightarrow Y$ ,  $Y \rightarrow X$ ,  $\overline{X} \rightarrow \overline{Y}$  and  $\overline{Y} \rightarrow \overline{X}$  are to be studied. If  $X$  disfavors  $Y$  ( $P(Y|X) < P(Y)$ ), then only  $X \rightarrow \overline{Y}$ ,  $\overline{X} \rightarrow Y$ ,  $\overline{Y} \rightarrow X$  and  $Y \rightarrow \overline{X}$  are to be studied. That's why our method studies only half of the candidates. The following proposition describes the independence between a pair of two variables.

**Proposition 3.** *Given  $X$  and  $Y$  two itemsets of  $\mathcal{I}$ , if  $(X, Y)$  is a stochastically independent pair, so are pairs  $(\overline{X}, Y)$ ,  $(X, \overline{Y})$ ,  $(\overline{X}, \overline{Y})$ .*

*Proof.*  $P(\overline{X})P(Y) - P(\overline{X} \wedge Y) = (1 - P(X))P(Y) - (P(Y) - P(X \cap Y)) = P(X \wedge Y) - P(X)P(Y)$ . So, if  $P(X \wedge Y) = P(X)P(Y)$ , then  $P(\overline{X})P(Y) = P(\overline{X} \wedge Y)$ . Since  $X$  and  $Y$  play symmetric roles, we have the same result for  $(X, \overline{Y})$ , then replacing  $Y$  with  $\overline{Y}$ , for  $(\overline{X}, \overline{Y})$ .  $\square$

This Proposition 3 is ideal, no association rule can be interesting if  $(X, Y)$  is stochastically independent. We continue our analysis by studying the candidate rules over the attractive class. To do this, we introduce the Proposition 4 in order to pruning certain positive and negative association rules.

**Proposition 4.** *For all  $X$  and  $Y$  of  $\mathcal{I}$  such that  $X$  favors  $Y$  and  $X \subseteq Y$ , we have (1)  $M_{GK}(X \rightarrow Y) \leq M_{GK}(Y \rightarrow X)$ , (2)  $M_{GK}(X \rightarrow Y) = M_{GK}(\overline{Y} \rightarrow \overline{X})$ , (3)  $M_{GK}(Y \rightarrow X) = M_{GK}(\overline{X} \rightarrow \overline{Y})$ , (4)  $M_{GK}(X \rightarrow Y) \leq M_{GK}(\overline{X} \rightarrow \overline{Y})$ .*

*Proof.* Let  $X$  and  $Y$  be items of  $\mathcal{I}$ . (1) According to Proposition 2 (1),  $X$  favors  $Y \Leftrightarrow Y$  favors  $X$ , it gives  $M_{GK}(Y \rightarrow X) = \frac{P(X|Y)-P(X)}{1-P(X)} = \frac{P(X)[P(Y|X)-P(Y)]}{P(\overline{X})P(Y)} = \frac{P(X)P(\overline{Y})}{P(\overline{X})P(Y)} \frac{P(Y|X)-P(Y)}{1-P(Y)} = \frac{P(X)}{P(\overline{X})} \frac{P(\overline{Y})}{P(Y)} M_{GK}(X \rightarrow Y)$ . Because  $X$  favors  $Y$  and  $X \subseteq Y$ , we have  $P(X) \geq P(Y) \Leftrightarrow P(\overline{X}) \leq P(\overline{Y}) \Leftrightarrow P(X)P(\overline{Y}) \geq P(\overline{X})P(Y)$ , where  $M_{GK}(X \rightarrow Y) \leq M_{GK}(Y \rightarrow X)$ . (2)  $M_{GK}(X \rightarrow Y) = \frac{P(Y|X)-P(Y)}{1-P(Y)} = \frac{-P(X|\overline{Y})+P(X)}{P(X)} = \frac{P(\overline{X}|\overline{Y})-P(\overline{X})}{1-P(\overline{X})} = M_{GK}(\overline{Y} \rightarrow \overline{X})$ . From this property,  $M_{GK}$  is implicative. So, the property (3) is immediate, it derives from this implicative character of the  $M_{GK}$ . The property remains to be shown (4). Indeed, according to Proposition 2 (2), we have  $M_{GK}(X \rightarrow Y) \leq M_{GK}(Y \rightarrow X) = M_{GK}(\overline{X} \rightarrow \overline{Y})$ , which gives us  $M_{GK}(X \rightarrow Y) \leq M_{GK}(\overline{X} \rightarrow \overline{Y})$ .  $\square$

In this Proposition 4, the properties (1), (2), (3) and (4) guarantee that if  $X \rightarrow Y$  is valid, then  $Y \rightarrow X$ ,  $\overline{X} \rightarrow \overline{Y}$  and  $\overline{Y} \rightarrow \overline{X}$  will also be the same because  $M_{GK}$  of the rule  $X \rightarrow Y$  is less than or equal to those of  $Y \rightarrow X$ ,  $\overline{X} \rightarrow \overline{Y}$  and  $\overline{Y} \rightarrow \overline{X}$ . The set of valid rules of the class is thus derived from the only rule  $X \rightarrow Y$ . This will significantly limit the research space. The following Proposition 5 is introduced to loosen certain rules of the repulsive class.

**Proposition 5.** *For all  $X$  and  $Y$  of  $\mathcal{I}$ , such that  $X$  disfavors  $Y$  and  $X \subseteq Y$ , we have (1)  $M_{GK}(X \rightarrow \overline{Y}) = M_{GK}(Y \rightarrow \overline{X})$ , (2)  $M_{GK}(\overline{X} \rightarrow Y) = M_{GK}(\overline{Y} \rightarrow X)$ , and (3)  $M_{GK}(X \rightarrow \overline{Y}) \leq M_{GK}(\overline{X} \rightarrow Y)$ .*

*Proof.* Let  $X$  and  $Y$  of  $\mathcal{I}$ . According to Proposition 2 (2), we have  $X$  disfavors  $Y \Leftrightarrow X$  favors  $\overline{Y} \Leftrightarrow \overline{Y}$  favors  $X \Leftrightarrow Y$  favors  $\overline{X} \Leftrightarrow \overline{X}$  favors  $Y$ . Thus, due to the implicative character of  $M_{GK}$ , the properties (1) and (2) are then immediate. It remains to show (3). As  $X$  favors  $\overline{Y}$ , we get  $M_{GK}(X \rightarrow \overline{Y}) = \frac{P(\overline{Y}|X)-P(\overline{Y})}{1-P(\overline{Y})} = \frac{P(\overline{X})P(\overline{Y})}{P(X)P(Y)} \frac{P(Y|\overline{X})-P(Y)}{1-P(Y)} = \frac{P(\overline{X}P(\overline{Y}))}{P(X)P(Y)} M_{GK}(\overline{X} \rightarrow Y)$ . By hypothesis,  $X$  disfavors  $Y$  and  $X \subseteq Y$ , we have  $P(X) \geq P(Y) \Leftrightarrow P(\overline{X}) \leq P(\overline{Y})$  implique  $P(\overline{X})P(\overline{Y}) \leq P(X)P(Y)$ , finally  $M_{GK}(X \rightarrow \overline{Y}) \leq M_{GK}(\overline{X} \rightarrow Y)$ .  $\square$

The properties (1), (2) and (3) of this Proposition 5 indicate that if  $X \rightarrow \overline{Y}$  is valid, then  $\overline{X} \rightarrow Y$ ,  $Y \rightarrow \overline{X}$  and  $\overline{Y} \rightarrow X$  will be valid, because this  $M_{GK}$  is less than or equal to those of  $\overline{X} \rightarrow Y$ ,  $Y \rightarrow \overline{X}$  and  $\overline{Y} \rightarrow X$ . Only  $X \rightarrow \overline{Y}$  will make it possible to deduce the interest of the class.

**Proposition 6.** *For all  $X$  and  $Y$  of  $\mathcal{I}$ ,  $M_{GK}(X \rightarrow \overline{Y}) = -M_{GK}(X \rightarrow Y)$ .*

*Proof.* We show in two cases: (1)  $X$  favors  $Y$  and (2)  $X$  disfavors  $Y$ . (1)  $X$  favors  $Y \Leftrightarrow X$  favors  $\bar{Y} \Leftrightarrow X$  disfavors  $\bar{Y}$  implies  $M_{GK}(X \rightarrow \bar{Y}) = \frac{P(\bar{Y}|X) - P(\bar{Y})}{P(\bar{Y})} = -\frac{P(Y|X) - P(Y)}{1 - P(Y)} = -M_{GK}(X \rightarrow Y)$ . (2)  $X$  disfavors  $Y \Leftrightarrow X$  favors  $\bar{Y}$ , this gives  $M_{GK}(X \rightarrow \bar{Y}) = \frac{P(\bar{Y}|X) - P(\bar{Y})}{1 - P(\bar{Y})} = -\frac{P(Y|X) - P(Y)}{P(Y)} = -M_{GK}(X \rightarrow Y)$ .  $\square$

The next result of the following proposition makes it possible to characterize the exact negative association rules according to support- $M_{GK}$  pair.

**Proposition 7.** *Let  $X, Y$  and  $Z$  be three itemsets disjoint 2 to 2, if  $XZ \rightarrow Y$  (resp.  $XZ \rightarrow \bar{Y}$ ) is an exact rule, so is rule  $X \rightarrow Y$  (resp.  $X \rightarrow \bar{Y}$ ).*

*Proof.*  $M_{GK}(XZ \rightarrow Y) = 1 \Leftrightarrow \frac{P(Y|XZ) - P(Y)}{1 - P(Y)} = 1 \Leftrightarrow \frac{\text{supp}((X \cup Z) \cup Y)}{\text{supp}(X \cup Z)} = 1$ . Since  $X, Y$  and  $Z$  are disjoint 2 to 2,  $\frac{\text{supp}(X \cup Y)}{\text{supp}(X)} = 1 \Leftrightarrow \frac{P(Y|X) - P(Y)}{1 - P(Y)} = 1 \Leftrightarrow M_{GK}(X \rightarrow Y) = 1$ . Replacing  $Y$  with  $\bar{Y}$  for  $XZ \rightarrow \bar{Y}$  and  $X \rightarrow \bar{Y}$ .  $\square$

The Corollary 1 is the consequence of the Proposition 7.

**Corollary 1.** *Let  $X$  and  $Y$  be two itemsets on  $\mathcal{I}$ , for all  $Z \subseteq \mathcal{I}$  such that  $Z \subset X$ , if  $M_{GK}(X \rightarrow \bar{Y}) = 1$ , then  $M_{GK}(Z \rightarrow \bar{Y}) = 1$ .*

*Proof.* For all  $Z$ , such that  $Z \subset X$ , we have  $\text{supp}(X) > 0$ . Therefore, by Proposition 7, we have  $M_{GK}(Z \rightarrow \bar{Y}) = 1$ .  $\square$

**Proposition 8.** *Let  $X, Y, T$  and  $Z$  be four itemsets of  $\mathcal{I}$ , such that  $X$  favors  $Y$  and  $Z$  favors  $T$ , and  $X \cap Y = Z \cap T = \emptyset$ , and  $X \subset Z \subseteq \gamma(X)$ , and  $Y \subset T \subseteq \gamma(Y)$ . Then,  $\text{supp}(X \cup Y) = \text{supp}(Z \cup T)$  and  $M_{GK}(X \rightarrow Y) = M_{GK}(Z \rightarrow T)$ .*

*Proof.*  $\forall X, Y, T, Z \subseteq \mathcal{I}$ ,  $\text{supp}(X \cup Y) = \frac{|g(X \cup Y)|}{|\mathcal{I}|} = \frac{|g(X) \cap g(Y)|}{|\mathcal{I}|}$  and  $\text{supp}(Z \cup T) = \frac{|g(Z \cup T)|}{|\mathcal{I}|} = \frac{|g(Z) \cap g(T)|}{|\mathcal{I}|}$ . Because  $X \subset Z \subseteq \gamma(X)$  and  $Y \subset T \subseteq \gamma(Y)$ , we have  $\text{supp}(X) = \text{supp}(Z)$  and  $\text{supp}(Y) = \text{supp}(T)$ . It causes  $g(X) = g(Z)$  and  $g(Y) = g(T)$  implies  $\text{supp}(X \cup Y) = \text{supp}(Z \cup T)$ . As  $\text{supp}(X) = \text{supp}(Z)$  and  $\text{supp}(Y) = \text{supp}(T)$ , we have  $P(Y|X) = P(T|Z) \Leftrightarrow P(Y|X) - P(Y) = P(T|Z) - P(Y) \Leftrightarrow \frac{P(Y|X) - P(Y)}{1 - P(Y)} = \frac{P(T|Z) - P(T)}{1 - P(T)} \Leftrightarrow M_{GK}(X \rightarrow Y) = M_{GK}(Z \rightarrow T)$ .  $\square$

**Proposition 9.** *Let  $X, Y, T, Z \subseteq \mathcal{I}$ , such that  $X$  disfavors  $Y$  and  $Z$  disfavors  $T$ , and  $X \cap Y = Z \cap T = \emptyset$ , and  $X \subset Z \subseteq \gamma(X)$ , and  $Y \subset T \subseteq \gamma(Y)$ . Then,  $\text{supp}(X \cup Y) = \text{supp}(Z \cup T)$  and  $M_{GK}(X \rightarrow \bar{Y}) = M_{GK}(Z \rightarrow \bar{T})$ .*

*Proof.*  $\forall X, Y, T, Z \subseteq \mathcal{I}$ ,  $\text{supp}(X \cup Y) = \frac{|g(X \cup Y)|}{|\mathcal{I}|}$  and  $\text{supp}(Z \cup T) = \frac{|g(Z \cup T)|}{|\mathcal{I}|}$ . Because  $X \subset Z \subseteq \gamma(X)$  and  $Y \subset T \subseteq \gamma(Y)$ , we have  $\text{supp}(X) = \text{supp}(Z)$  and  $\text{supp}(Y) = \text{supp}(T)$ . It causes  $g(X) = g(Z)$  and  $g(Y) = g(T)$  implies  $\text{supp}(X \cup Y) = \text{supp}(Z \cup T)$ . Because  $\text{supp}(X) = \text{supp}(Z)$  and  $\text{supp}(Y) = \text{supp}(T)$ , we have  $P(Y|X) = P(T|Z) \Leftrightarrow P(\bar{Y}|X) = P(\bar{T}|Z) \Leftrightarrow P(\bar{Y}|X) - P(\bar{Y}) = P(\bar{T}|Z) - P(\bar{Y}) \Leftrightarrow \frac{P(\bar{Y}|X) - P(\bar{Y})}{1 - P(\bar{Y})} = \frac{P(\bar{T}|Z) - P(\bar{T})}{1 - P(\bar{T})} \Leftrightarrow M_{GK}(X \rightarrow \bar{Y}) = M_{GK}(Z \rightarrow \bar{T})$ .  $\square$

**Proposition 10.** *Let  $X, Y, T, Z \subseteq \mathcal{I}$ , such that  $X$  disfavors  $Y$  and  $Z$  disfavors  $T$ , and  $X \cap Y = Z \cap T = \emptyset$ , and  $X \subset Z \subseteq \gamma(X)$ , and  $Y \subset T \subseteq \gamma(Y)$ . Then,  $\text{supp}(X \cup Y) = \text{supp}(Z \cup T)$  and  $M_{GK}(\bar{X} \rightarrow Y) = M_{GK}(\bar{Z} \rightarrow T)$ .*

*Proof.*  $\forall X, Y, T, Z \subseteq \mathcal{I}$ ,  $\text{supp}(X \cup Y) = \frac{|g(X \cup Y)|}{|T|}$  and  $\text{supp}(Z \cup T) = \frac{|g(Z \cup T)|}{|T|}$ . Because  $X \subset Z \subseteq \gamma(X)$  and  $Y \subset T \subseteq \gamma(Y)$ , we have  $\text{supp}(X) = \text{supp}(Z)$  and  $\text{supp}(Y) = \text{supp}(T)$ . It causes  $g(X) = g(Z)$  and  $g(Y) = g(T)$  implies  $\text{supp}(X \cup Y) = \text{supp}(Z \cup T)$ . Because  $\text{supp}(X) = \text{supp}(Z)$  and  $\text{supp}(Y) = \text{supp}(T)$ , we have  $P(Y|X) = P(T|Z) \Leftrightarrow P(Y|\bar{X}) = P(T|\bar{Z}) \Leftrightarrow P(Y|\bar{X}) - P(T) = P(T|\bar{Z}) - P(T) \Leftrightarrow \frac{P(Y|\bar{X}) - P(Y)}{1 - P(Y)} = \frac{P(T|\bar{Z}) - P(T)}{1 - P(T)} \Leftrightarrow M_{GK}(\bar{X} \rightarrow Y) = M_{GK}(\bar{Z} \rightarrow T)$ .  $\square$

The following Subsection summarizes these different optimizations via the Algorithm 1 and the Algorithm 3.

### 3.3 Our Algorithm

As we mentioned, our approach describes in a double problematic: mining frequent itemsets (Algorithm 1) and generation of potential valid positive and negative association rules (Algorithm 3). The Algorithm 1 takes as argument a context  $\mathcal{B}$ , a *minsup*. It returns a set  $\mathcal{F}$  of frequent itemsets, where  $\mathcal{C}_k$  denotes the set of candidate  $k$ -itemsets, and  $\mathcal{CGM}_k$  the set of generator  $k$ -itemsets. The database  $\mathcal{B}$  is built in line 1. Next,  $\mathcal{F}_1$  and  $\mathcal{F}_2$  are generated in a single pass (Algorithm 1 lines 2 and 3). The EOMF-GEN function (Algorithm 2) is called to generate candidates (Algorithm 1 line 5). It takes as argument  $\mathcal{F}_{k-1}$ , and returns a superset  $\mathcal{C}_k$ . The initialization of  $\mathcal{C}_k$  to the empty set is done in line 1 (Algorithm 2). A join between the elements of  $\mathcal{F}_{k-1}$  is then made (Algorithm 2 lines 2 to 6). Indeed, two  $p$  and  $q$  items of  $\mathcal{F}_{k-1}$  form a  $c$  if, and only if they contain common  $(k-2)$ -itemsets. For example, joining  $ABC$  and  $ABD$  gives

---

#### Algorithm 1. EOMF: Frequent Itemset Mining

---

**Require:** A dataset  $\mathcal{B} = (\mathcal{T}, \mathcal{I}, \mathcal{R})$ , a minimum support *minsup*.

**Ensure:** All frequent itemsets  $\mathcal{F}$ .

```

1: MATRICESUPPORT  $\leftarrow$  Scan( $\mathcal{B}$ ); //Scan dataset  $\mathcal{B}$ 
2:  $\mathcal{F}_1 \leftarrow \{c_1 \in \text{MATRICESUPPORT} \mid \text{supp}(c_1) \geq \text{minsup}\}$ ; //Generate 1-itemsets
3:  $\mathcal{F}_2 \leftarrow \{c_2 \in \text{MATRICESUPPORT} \mid \text{supp}(c_2) \geq \text{minsup}\}$ ; //Generate 2-itemsets
4: for ( $k = 3; \mathcal{F}_{k-1} \neq \emptyset; k++$ ) do
5:    $\mathcal{C}_k \leftarrow$  EOMF-GEN( $\mathcal{F}_{k-1}$ ); //New candidate (see algorithm 2)
6:   for all (transaction  $t \in \mathcal{T}$ ) do
7:      $\mathcal{C}_t \leftarrow$  subset( $\mathcal{C}_k, t$ ) or  $\mathcal{C}_t = \{c \in \mathcal{C}_k \mid c \subseteq t\}$  //Select candidate in  $t$ 
8:     for all (candidate  $c \in \mathcal{C}_t$ ) do
9:       if ( $c \in \mathcal{CGM}_k$ ) then
10:         $\text{supp}(c) = |\{t \in \mathcal{T} \mid c \subseteq t\}|/|\mathcal{T}|$ ;
11:       else
12:         $\text{supp}(c) = \min\{\text{supp}(c') \mid c' \subset c\}$ ;
13:       end if
14:        $\text{supp}(c)++$ ;
15:     end for
16:   end for
17:    $\mathcal{F}_k \leftarrow \{c \in \mathcal{C}_k \mid \text{supp}(c) \geq \text{minsup}\}$ ; //Generate frequent itemsets
18: end for
19: return  $\mathcal{F} = \bigcup_k \mathcal{F}_k$ 

```

---

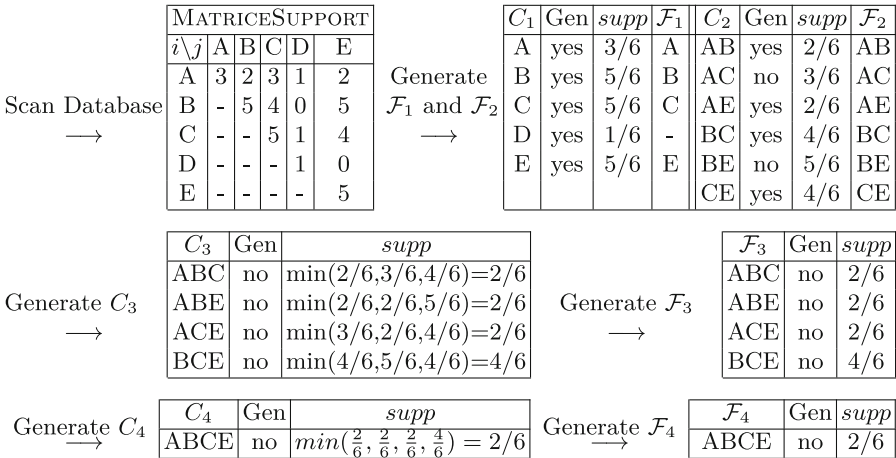
**Algorithm 2.** EOMF-GEN Procedure**Require:** A set  $\mathcal{F}_{k-1}$  of frequent  $(k-1)$ -itemset**Ensure:** A set  $C_k$  of candidate  $k$ -itemset

```

1:  $C_k \leftarrow \emptyset$ 
2: for all itemset  $p \in \mathcal{F}_{k-1}$  do
3:   for all itemset  $q \in \mathcal{F}_{k-1}$  do
4:     if ( $p[1] = q[1], \dots, p[k-2] = q[k-2], p[k-1] < q[k-1]$ ) then
5:        $c \leftarrow p \cup q(k-1)$ ; //Generate candidate
6:     end if
7:     for all  $((k-1)$ -subset  $s$  of  $c$ ) do
8:       if ( $s \in \mathcal{F}_{k-1}$ ) then
9:          $C_k \leftarrow C_k \cup \{c\}$ ;
10:      end if
11:    end for
12:  end for
13: end for
14: return  $C_k$ 

```

$ABCD$ . However, joining  $ABC$  and  $CDE$  does not work because they do not contain common 2-itemsets. Once  $C_k$  has been established, it researches among the elements of  $C_k$ . If this is the case, it calculates the support in two cases (Algorithm 1 lines 9 to 13): if  $c$  is generator, an access to the database is made to know its support (Algorithm 1 line 10), otherwise, it is derived from its subsets without going through the database (Algorithm 1 line 12). The support is then increased (Algorithm 1 line 14). And, only frequent itemsets are retained in  $\mathcal{F}_k$  (Algorithm 1 line 17). For the sake of comprehension, we apply this Algorithm 1 on a small database  $\mathcal{B}$ , shown in Table 1. The minimum support is equal to  $2/6$ , where Gen. designates a generator itemset. Results are shown in Fig. 1. After reading the dataset  $\mathcal{B}$ ,  $D$  is not frequent, its support is smaller than the  $minsup$ . It is pruned from the next step. The other elements are kept to generate  $C_2$ . These elements are frequent, its gives  $F_2$ . Then,  $C_3$  is generated. We have

**Fig. 1.** Example of the Algorithm 1,  $minsup = 2/6$

$supp(AC) = supp(A)$  and  $supp(BE) = supp(B) = supp(E)$ ,  $AC$  and  $BE$  are not generators. No candidate of  $C_3$  is then generator, i.e. no access over the  $\mathcal{B}$ . Also in the last step, the support of  $ABCE$  is equal to  $ABC$  (or  $ABE$ , or  $ACE$ ). From this example, our approach does it in a single pass to the database, this is not the case for the existing ones, they do it in 4 passes.

The following Algorithm 3 embodies the different optimizations we have defined in above Subsect. 3.2. The Algorithm 3 takes as argument a set  $\mathcal{F}$ , thresholds  $minsup$  and  $minmgk$ , and returns a set  $\mathcal{E}_{RAPN}$ . It is initialized by the empty set in line 1. Next, for each itemset of  $\mathcal{F}$  set, the set  $\mathcal{A}$  is generated (line 3). For each subset  $X_{k-1}$  of  $\mathcal{A}$  (line 4), the algorithm proceeds in two recursive steps. The first consists in generating attractive class rules using the single rule  $X \rightarrow Y$  (Algorithm 3 lines 6 to 11). Indeed, if  $supp(X \rightarrow Y) \geq minsup$  and  $M_{GK}(X \rightarrow Y) \geq minmgk$ , then the  $\mathcal{E}_{RAPN}$  set is updated by adding  $X \rightarrow Y$ ,  $Y \rightarrow X$ ,  $\bar{Y} \rightarrow \bar{X}$  and  $\bar{X} \rightarrow \bar{Y}$  (Algorithm 3 line 9). The second step consists in generating repulsive class rules by studying only  $X \rightarrow \bar{Y}$  (Algorithm 3 lines 11 to 16). The Algorithm 3 is updated by adding  $X \rightarrow \bar{Y}$ ,  $Y \rightarrow \bar{X}$ ,  $\bar{Y} \rightarrow X$  and  $\bar{X} \rightarrow Y$  (Algorithm 3 line 14). ERAPN returns the  $\mathcal{E}_{RAPN}$  set (Algorithm 3 line 19).

---

### Algorithm 3. Association Rules Generation

---

**Require:** A set  $\mathcal{F}$  of frequent itemsets, a  $minsup$  and  $minmgk$ .

**Ensure:** A set  $\mathcal{E}_{RAPN}$  of valid positive and negative rules.

```

1:  $\mathcal{E}_{RAPN} = \emptyset$ ;
2: for all ( $k$ -itemset  $X_k$  of  $\mathcal{F}$ ,  $k \geq 2$ ) do
3:    $\mathcal{A} = \{(k-1)\text{-itemset} \mid X_{k-1} \subset X_k\}$ 
4:   for all ( $X_{k-1} \in \mathcal{A}$ ) do
5:      $X = X_{k-1}$ ;  $Y = X_k \setminus X_{k-1}$ ;
6:     if ( $P(Y|X) > P(Y)$ ) then
7:        $supp(X \cup Y) = \frac{|g(X \cup Y)|}{|\mathcal{F}|}$ ;  $M_{GK}(X \rightarrow Y) = \frac{P(Y|X) - P(Y)}{1 - P(Y)}$ ;
8:       if ( $supp(X \cup Y) \geq minsup$  &  $M_{GK}(X \rightarrow Y) \geq minmgk$ ) then
9:          $\mathcal{E}_{RAPN} \leftarrow \mathcal{E}_{RAPN} \cup \{X \rightarrow Y, Y \rightarrow X, \bar{Y} \rightarrow \bar{X}, \bar{X} \rightarrow \bar{Y}\}$ ;
10:      end if
11:     else
12:        $supp(X \cup \bar{Y}) = \frac{|g(X \cup \bar{Y})|}{|\mathcal{F}|}$ ;  $M_{GK}(X \rightarrow \bar{Y}) = \frac{P(\bar{Y}|X) - P(\bar{Y})}{1 - P(\bar{Y})}$ ;
13:       if ( $supp(X \cup \bar{Y}) \geq minsup$  &  $M_{GK}(X \rightarrow \bar{Y}) \geq minmgk$ ) then
14:          $\mathcal{E}_{RAPN} \leftarrow \mathcal{E}_{RAPN} \cup \{X \rightarrow \bar{Y}, Y \rightarrow \bar{X}, \bar{Y} \rightarrow X, \bar{X} \rightarrow Y\}$ ;
15:       end if
16:     end if
17:   end for
18: end for
19: return  $\mathcal{E}_{RAPN}$ 

```

---

**Example Illustrate of Algorithm 3.** Indeed, we consider the frequent itemset  $ABC \subseteq \mathcal{F}$  (cf. Fig. 1). We will study a total  $|\mathcal{E}_{RAPN}(ABC)| = 72$  rules, where 12 positive rules and 60 negative rules. First, we start to study the positive rules. There are 6 possible rules:  $A \rightarrow BC$ ,  $B \rightarrow AC$ ,  $C \rightarrow AB$ ,  $AB \rightarrow C$ ,  $AC \rightarrow B$  and  $BC \rightarrow A$ . Since  $ABC$  is frequent, then its subsets  $A$ ,  $B$  and  $C$  are also frequent, which gives the other candidates  $A \rightarrow B$ ,  $A \rightarrow C$ ,  $B \rightarrow C$ ,  $B \rightarrow A$ ,  $C \rightarrow A$  and  $C \rightarrow B$ . Indeed, we will study first  $A \rightarrow B$ ,  $A \rightarrow C$  and  $B \rightarrow C$ .



**Table 4.** Generation of positive association rules,  $minsup = 0.1$  and  $minmgk = 0.6$

$X \rightarrow Y$	$P(X)$	$P(Y)$	$supp(X \cup Y)$	$P(Y X) - P(Y)$	$1 - P(Y)$	$M_{GK}(X \rightarrow Y)$
$A \rightarrow B$	0.50	0.83	0.33	0.17	-0.17	-1
$A \rightarrow C$	0.50	0.83	0.50	0.17	0.17	1
$B \rightarrow C$	0.83	0.83	0.67	0.17	-0.03	-0.2
$B \rightarrow A$	0.83	0.50	0.33	0.50	-0.10	-0.2
$C \rightarrow A$	0.83	0.50	0.50	0.50	0.10	0.2
$C \rightarrow B$	0.83	0.83	0.67	0.17	-0.03	-0.2

Given  $minsup = 0.1$  and  $minmgk = 0.6$ . Results are shown in Table 4 below. Because  $M_{GK}(B \rightarrow C) = M_{GK}(B \rightarrow A) = M_{GK}(C \rightarrow A) = -0.2 < 0.6$  and  $M_{GK}(C \rightarrow B) = 0.2 < 0.6$ , then  $B \rightarrow C$ ,  $B \rightarrow A$ ,  $C \rightarrow A$  and  $C \rightarrow B$  are not valid. So, by Propositions 7 and 8,  $A \rightarrow BC$ ,  $BC \rightarrow A$ ,  $B \rightarrow AC$  and  $C \rightarrow AB$  are also invalid. Since  $M_{GK}(A \rightarrow B) = -1 < 0$ , then  $A \rightarrow B$  is invalid.

Here, we derive the valid positive and negative traditional. Because,  $supp(A\bar{B}) = supp(A) - supp(AB) = 0.17 > 0.1$ , and, by Proposition 6,  $M_{GK}(A \rightarrow \bar{B}) = -M_{GK}(A \rightarrow B) = 1 > 0.6$ . Therefore,  $A \rightarrow \bar{B}$  is exact negative rule. Since  $supp(AC) = 0.50 > 0.1$  and  $M_{GK}(A \rightarrow C) = 1 > 0.6$ ,  $A \rightarrow C$  is exact positive rule. Because,  $A \rightarrow C$  is exact, by Proposition 7,  $AB \rightarrow C$  is also exact rule. Because  $A \rightarrow \bar{B}$  is exact negative, by Proposition 9,  $AC \rightarrow \bar{B}$  is also exact negative. Because  $\bar{A} \rightarrow B$  is exact negative, by Corollary 1,  $\bar{A}\bar{B} \rightarrow B$  is also exact negative. Because  $\bar{B} \rightarrow A$  is exact negative, by Proposition 10,  $\bar{B} \rightarrow AC$  is also exact negative. Therefore, because  $\bar{B} \rightarrow AC$  is exact negative, by Proposition 7,  $\bar{B} \rightarrow C$  is also exact negative. Results are shown in following Table 5. In this small example, our approach restores nine valid positive and negative association rules in total, including two positive rules of type  $X \rightarrow Y$ , two negative rules of type  $X \rightarrow \bar{Y}$  and five negative rules of type  $\bar{X} \rightarrow Y$ .

**Table 5.** Potential valid association rules according to support- $M_{GK}$

$X \rightarrow Y$	$P(X)$	$P(Y)$	$supp(X \cup Y)$	$P(Y X) - P(Y)$	$1 - P(Y)$	$M_{GK}(X \rightarrow Y)$
$A \rightarrow C$	0.50	0.83	0.50	0.17	0.17	1
$AB \rightarrow C$	0.33	0.83	0.33	0.17	0.17	1
$A \rightarrow \bar{B}$	0.50	0.83	0.33	0.17	0.17	1
$AC \rightarrow \bar{B}$	0.50	0.83	0.33	0.17	0.17	1
$\bar{A} \rightarrow B$	0.50	0.83	0.50	0.17	0.17	1
$\bar{A}\bar{C} \rightarrow B$	0.50	0.83	0.50	0.17	0.17	1
$\bar{B} \rightarrow A$	0.17	0.50	0.17	0.50	0.50	1
$\bar{B} \rightarrow C$	0.17	0.83	0.17	0.17	0.17	1
$\bar{B} \rightarrow AC$	0.17	0.50	0.17	0.50	0.50	1

**Complexity of ERAPN Algorithm.** There are three lenses: *average*, *best* and *worst case*. The first model evaluates the average time, which proves to be very difficult and leaves the framework of this work. The second model estimates the minimal time, which also leaves the framework of this work. We are interested in the last one, because we want to evaluate the costs of calls of the most expensive operations. In what follows, we present the study of the complexity of our ERAPN algorithm. This is calculated for each of the two constituting steps: frequent itemsets mining and finding positive and negative association rules.

*Complexity of Frequent Itemsets Mining (Algorithm 1):* The Algorithm 1 takes as input the transaction context  $\mathcal{B} = (\mathcal{T}, \mathcal{I}, \mathcal{R})$ . Let  $n = |\mathcal{T}|$  and  $m = |\mathcal{I}|$ . There is worst case if the candidate are generators (i.e.  $2^{\mathcal{I}}$ ). The time complexity of support counting for 1 and 2-itemsets is  $\mathcal{O}(m \times n)$  (line 1). The instructions for lines 2–3 are  $\mathcal{O}(2)$ . The cost of finding longest frequent itemsets (i.e. all itemsets of sizes  $\geq 3$ ) (lines 4–16) is equal to the sum of the following costs. EOMF-GEN: there are  $(2^m - m - 1)$  candidates to generate. Thus, the cost of this procedure is  $\mathcal{O}(2^m - m)$  (lines 2–13 in the Algorithm 2). The cost of support counting of longest candidates is  $\mathcal{O}(n(2^m - m))$  (lines 6–16). The time complexity of space frequent itemsets is  $\mathcal{O}(2^m - m)$  (line 17). The global complexity of this Algorithm 1 is therefore  $\mathcal{O}(mn + 2^m - m + n(2^m - m) + 2^m - m) = \mathcal{O}(n2^m)$ .

*Complexity of Rule Generation (Algorithm 3):* The algorithm takes as input a set of frequent itemsets  $\mathcal{F}$ , which is obtained from a context  $\mathcal{B}$ . Its global complexity is linear in  $|\mathcal{F}|$ , which takes  $\mathcal{O}(2^{-1}|\mathcal{F}|(5^m - 2(3^m)))$ . This complexity is obtained by the following instructions. The “for” loop (line 2), which runs through all of the  $\mathcal{F}$  itemsets, is done in  $\mathcal{O}(|\mathcal{F}|)$  at worst. The second “for” loop (line 4) is  $\mathcal{O}(|\mathcal{A}|/2)$  at worst, because only half of the candidate rules that are traversed in our approach to test their eligibility (instructions 6 to 16). It is carried out in two identical tests (lines 8 and 13). For each of the tests, the possible number of rules generated, at a  $m$ -itemset, is equal to  $2^{2m} - 2^{m+1}$ . Which gives  $C_m^{m-1}(2^{2(m-1)} - 2^m)$  for a  $(m - 1)$ -itemset,  $C_m^{m-2}(2^{2(m-2)} - 2^{(m-1)})$  for a  $(m - 2)$ -itemset, and so an. In sum, we have  $\mathcal{O}(|\mathcal{A}|) = \sum_{k=2}^m C_m^k (2^{2k} - 2^{k+1}) = \sum_{k=2}^m C_m^k 4^k - 2 \sum_{k=2}^m C_m^k 2^k = [\sum_{k=0}^m C_m^k 4^k - (1 + 4m)] - 2 [\sum_{k=0}^m C_m^k 2^k - (1 + 2m)]$ . Now, for all  $x$  of  $\mathbb{R}$ ,  $\sum_{k=0}^m C_m^k x^k = (1 + x)^m$ , so  $\mathcal{O}(|\mathcal{A}|) = \mathcal{O}(5^m - 2(3^m) + 1) = \mathcal{O}(5^m - 2(3^m))$ . Finally, the overall time complexity is  $\mathcal{O}(|\mathcal{F}||\mathcal{A}|/2) = \mathcal{O}(2^{-1}|\mathcal{F}|(5^m - 2(3^m)))$ .

In the worst case, the total complexity of the ERAPN algorithm is of the order of  $\mathcal{O}(2^{-1}|\mathcal{F}|(5^m - 2(3^m))) + \mathcal{O}(n2^m) = \mathcal{O}(2^{-1}|\mathcal{F}|(5^m - 2(3^m)) + n2^m)$ .

## 4 Experimental Results

This section presents the experimental study conducted in order to evaluate the performances of our algorithm. The latter is implemented in  $R$  and tested on PC Core i3 and 4 GB of RAM running under Windows system. We compare the results with those of Wu and RAPN, conducted out on four databases from UCI,

such as **Adult**, **German**, **Income** and **Iris**. For each algorithm, we have chosen the same thresholds to avoid biasing the results. The following Table 6 reports the characteristics of datasets, and the number of positive and negative rules by varying the minimum thresholds *minsup* and *minmgk*. Indeed, the first three columns indicate the data characteristics in question, the last fifteen columns present the different results, where the column labelled “++” corresponds to the type  $X \rightarrow Y$ , column “-+” to  $\bar{X} \rightarrow Y$ , column “+-” to  $X \rightarrow \bar{Y}$ , and “--” to  $X \rightarrow \bar{Y}$ . The behaviour of algorithms varies according to data characteristics. The large database is much more time-consuming to run. In other words, the number of rules increases as thresholds decrease. Except for dense databases (**Adult** and **German**) and relatively low thresholds (*minsup* = 1% et *minmgk* = 60%), the number of rules (see Table 6) in Wu is 100581 and 89378 for RAPN. They are relatively large, due to the strong contribution of positive rules of type  $X \rightarrow Y$  and negative rules of type  $X \rightarrow \bar{Y}$  (see Table 6), than for ERAPN (28784 rules). The rules of type  $\bar{X} \rightarrow Y$  and  $\bar{X} \rightarrow \bar{Y}$  remain reasonable for each algorithm. On less dense databases (**Income** and **Iris**), these algorithm gives the reasonable number of rules. Note that RAPN, for Iris data, does not extract the type  $\bar{X} \rightarrow \bar{Y}$  (see Table 6) for *minsup* (resp. *minmgk*) over 3% (resp. 80%). Figure 2 below shows the response times by varying the *minupp* and keeping *minmgk* = 60%. They also increase when thresholds are lowered. The execution time of ERAPN is faster than that of Wu and RAPN. ERAPN gained 7 more times the best response in the worst cases. These different performances can be explained as follows. RAPN and Wu are limited on classical data structure, which requires repetitive access over the whole database. Wu has the lowest performance. One of the main reasons lies in the pruning technique. In this case, the *interest* measure does not have effective properties for frequent itemsets mining. In addition, the search space of valid association rules can be covered exhaustively. To this, our algorithm introduces the different optimizations. Therefore, the all frequent itemsets can be traversed only once. Moreover, the search space of rules is only half full. In all cases, our model remains the most selective and concise.

**Table 6.** Characteristics of datasets and results extracts

Database	T	Z	<i>minsup</i>	<i>minmgk</i>	Wu					RAPN					ERAPN				
					++	-+	+-	--	Σ	++	-+	+-	--	Σ	++	-+	+-	--	Σ
Adult	48842	115	1%	60%	97956	625	1215	785	100581	87800	542	615	421	89378	27500	422	510	352	28784
			2%	70%	55925	453	852	556	57786	53950	323	503	344	55120	25536	354	385	225	26500
			3%	80%	38750	345	412	310	39817	22033	156	254	145	22588	18523	124	154	124	18925
German	1000	71	1%	60%	51478	456	1148	555	53637	41235	401	565	412	42613	26456	340	380	245	27421
			2%	70%	39683	352	744	456	41235	38555	234	425	384	39598	18800	220	203	156	19379
			3%	80%	9835	144	545	321	10845	18500	75	325	232	19132	12157	95	85	55	15392
Income	6876	50	1%	60%	2800	227	527	254	3808	2130	350	385	286	3151	1552	95	103	84	1834
			2%	70%	2200	127	327	213	2867	2054	214	330	185	2783	1433	55	63	65	1616
			3%	80%	1325	87	252	121	1785	1212	65	156	45	1478	923	35	30	17	1005
Iris	150	15	1%	60%	2437	159	196	160	2952	1954	10	60	24	2048	1500	150	165	124	1939
			2%	70%	2000	159	145	120	2424	1323	10	55	20	1407	1122	75	85	65	1347
			3%	80%	1200	159	59	45	1463	1056	25	30	-	1111	965	14	22	18	1019

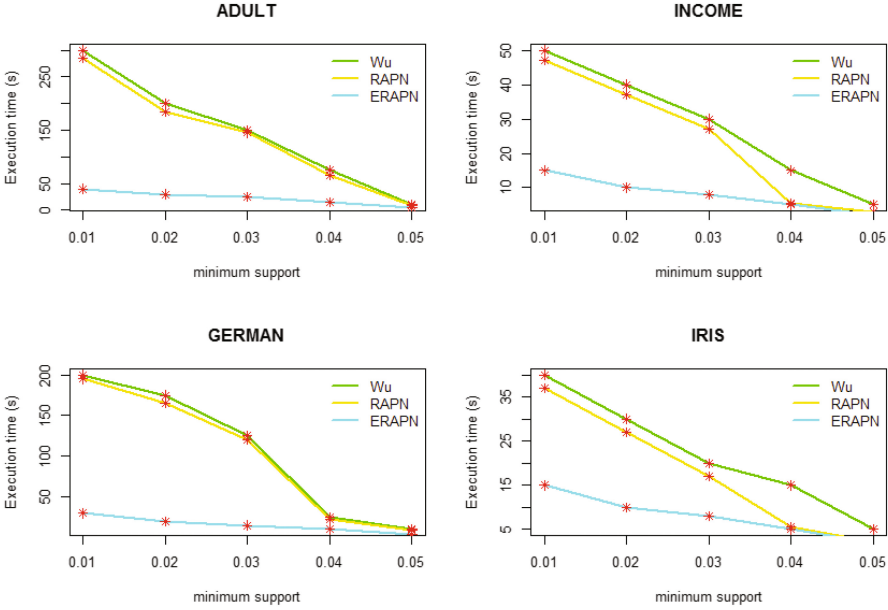


Fig. 2. Performances for each algorithm according to *minsup*

## 5 Related Work

Association rules mining is an active topic in Big Data. Apriori algorithm [2] is the first model that deals with this topic. On the other hand, it scans database multiple times as long as large frequent itemsets are generated. Apriori TID algorithm [2] generates candidate itemset before database is scanned with the help of Apriori-Gen function. Database is scanned only first time to count support, rather than scanning database it scans candidate itemsets. Despite their notable contributions, Apriori and Apriori TID [2] algorithms are limited on a single type of classical (or positive) association rules. The negative association rules has not been studied. To this, outreach works has been proposed.

Brin et al. [10] propose a model generating negative association rules by using the  $\chi^2$  measure. It is a first time in the literature the notion of negative relationships. The statistical chi-square is used to verify the independence between two variables. It's also used to determine the nature of the relationship, and a correlation metric. Although effective, the model suffers the problem of space memory due to the chi-square  $\chi^2$  was used. In [15], the authors present an approach to mine strong negative rules. They combine positive frequent itemsets with domain knowledge in the form of a taxonomy to mine negative association rules. However, as mentioned in many works [3,14], their approach is hard to generalized since it is domain dependant and requires a predefined taxonomy. Boulicaut et al. [9] present an approach using constraints to generate the association of the form  $X \wedge Y \rightarrow \bar{Z}$  or  $\bar{X} \wedge Y \rightarrow Z$  with negations using closed itemsets.

Despite its notable contribution, this method is limited of this form. Wu et al. [19] propose an approach for generating both positive and negative association rules. They add on top of the support-confidence framework other two measures, called *interest* and *CPIR* for a better pruning of the frequent itemset and frequent association rules, respectively. One of the key problems lies in pruning: no optimized techniques are used, and the search space can be exhaustively explored due to the measure *interest* was used. In [3], the authors propose an approach for mining positive and negative association rules. They add on top of the support-confidence framework another measure, called *Correlation coefficient*. Nevertheless, it requires to challenging problem of finding the frequent association rules, their strategy for search space is not optimized, which can be costly. In [16], the authors propose a new algorithm SRM (substitution rules mining) for mining only negative association of the type  $X \rightarrow \bar{Y}$ . Although effective, SRM algorithm is limited of this only type. In [11], the authors propose the PNAR algorithm. Although obtaining notable contributions, PNAR suffers the high volume of results, due to support-confidence pair was used. Wilhelmiina proposes the Kingfisher algorithm [18] using the Fisher test. A notable limitation of this model lies in the computation of *p-value* imposing exhaustive passes over the whole database, which gives the high computational time. Guillaume and Papon [14] propose RAPN algorithm based on support-confidence pair and other measure,  $M_G$  ( $M_{GK}$  [13] modified). Although effective, RAPN suffers relatively the high computational cost on the search space frequent itemsets.

Note that the major handicap of these works stems mainly from the computational costs for frequent itemsets mining (repetitive passes over the whole database) and association rules mining (exhaustive passes over the search space).

Recently, we proposed a new algorithm, EOMF [5], allowing the extraction of frequent itemsets. Therefore, a single pass over the database will extract all frequent itemsets, which significantly reduces the costs of calculation. As for association rules mining, we introduced in [7,8] a new approach allowing the extraction of positive and negative association rules using a new pair, support- $M_{GK}$ . As a result, only half of all candidate rules are studied, which also reduces the search space significantly. In this paper, we combine our works [5,7,8]. Ameliorations have been made, especially in terms of accuracy and simplicity. In [5], the path of frequent itemsets space has been quite heavy: the non-generator itemsets are implicitly taken twice for each calculation step, which can be costly. This gap has been corrected in the current work. We introduced a new strategy of the search space via a notable property (cf. Property 1) exploiting the monotony concepts of the generator itemsets, which consequently reduces the cost. Therefore, improvements have been added in Algorithm 1 (lines 4 to 16), which makes the approach robust. In [7,8], we used the parameter  $vc_\alpha(r) = \sqrt{\frac{1}{n} \frac{n-n_X}{n_X} \frac{n_Y}{n-n_Y} \chi^2(\alpha)}$ , to prune association rules. Nevertheless, this parameter presents a notable limit. It requires the exhaustive paths over the whole database to know its values for each candidate association rule, i.e. for a  $m$ -itemset, computable in  $\mathcal{O}(2^m)$  on its contingency table, it gives  $4C_m^k 2^k$ , traditionally cost, for all  $k$ . This parameter is not very selective, its sometimes eliminates the interesting rules (or robust), but

considers the uninteresting rules (far from the logical implication), because, of its critical value. In this paper, we try to close this limit using a simple parameter,  $minmgk \in [0, 1]$ , that does not require access to the database. In addition, we introduced effective properties for the search space (cf. Propositions 7 to 10).

## 6 Conclusion

In this paper, we have studied the problems of positive and negative association rules for Big Data. Further optimizations have been defined. Experiments conducted on reference databases, compared to RAPN and Wu algorithms, have emphasized the efficiency of our approach. A study on the extraction of disjunctions/conjunctions association rules has not been initially developed, which gives leads to explore from a methodological and algorithmic point of view.

## References

1. Agrawal, R., Imielinski, T., Swami, A.N.: Mining association rules between sets of items in large databases. In: Proceedings of ACM SIGMOD, pp. 207–216 (1993)
2. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. In: Proceedings of 20th VLDB Conference, Santiago Chile, pp. 487–499 (1994)
3. Antonie, M.-L., Zaïane, O.R.: Mining positive and negative association rules: an approach for confined rules. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.) PKDD 2004. LNCS (LNAI), vol. 3202, pp. 27–38. Springer, Heidelberg (2004). [https://doi.org/10.1007/978-3-540-30116-5\\_6](https://doi.org/10.1007/978-3-540-30116-5_6)
4. Bastide, Y., Taouil, R., Pasquier, N., Stumme, G., Lakhal, L.: PASCAL: un algorithme d'extraction des motifs fréquents. Tech. Sces. Info. **21**, 65–95 (2002)
5. Bemarisika, P., Totohasina, A.: Eomf, un algorithme d'extraction optimisée des motifs fréquents. In: Proceedings of AAFD & SFC, Marrakech Maroc, pp. 198–203 (2016)
6. Bemarisika, P.: Extraction de règles d'association selon le couple support- $M_{GK}$ : Graphes implicatifs et Application en didactique des mathématiques. Université d'Antananarivo, Madagascar (2016)
7. Bemarisika, P., Totohasina, A.: Optimisation de l'extraction des règles d'association positives et négatives. In: Actes des 24èmes Rencontres de la Société Francophone de Classification, Lyon 1, France, pp. 25–28 (2017)
8. Bemarisika, P., Totohasina, A.: Optimized mining of potential positive and negative association rules. In: Bellatreche, L., Chakravarthy, S. (eds.) DaWaK 2017. LNCS, vol. 10440, pp. 424–432. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-64283-3\\_31](https://doi.org/10.1007/978-3-319-64283-3_31)
9. Boulicaut, J.-F., Bykowski, A., Jeud, B.: Towards the tractable discovery of association rules with negations. In: Larsen, H.L., Andreasen, T., Christiansen, H., Kacprzyk, J., Zadrozny, S. (eds.) FQAS 2000. Advances in Soft Computing, vol. 7. Springer, Heidelberg (2001). [https://doi.org/10.1007/978-3-7908-1834-5\\_39](https://doi.org/10.1007/978-3-7908-1834-5_39)
10. Brin, S., Motwani, R., Silverstein, C.: Bayond market baskets: generalizing association rules to correlation. In: Proceedings of the ACM SIGMOD, pp. 265–276 (1997)
11. Cornelis, C., Yan, P., Zhang, X., Chen, G.: Mining positive and negative association rules from large databases. Proc. IEEE, 613–618 (2006)

12. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer, Heidelberg (1999). <https://doi.org/10.1007/978-3-642-59830-2>
13. Guillaume, S.: Traitement de données volumineuses: Mesure et algorithmes d'extraction de règles d'association. Ph.D. thesis, Université de Nantes (2000)
14. Guillaume, S., Papon, P.-A.: Extraction optimisée de règles d'association positives et négatives (RAPN). In: Actes de la 13e Conference International Franco. EGC, pp. 157–168 (2013)
15. Savasere, A., Omiecinski, E., Navathe, S.: Mining for strong negative associations in a large database of customer transactions. In: Proceedings of ICDE, pp. 494–502 (1998)
16. Teng, W.-G., Hsieh, M.-J., Chen, M.-S.: A statistical framework for mining substitution rules. *Knowl. Inf. Syst.* **7**, 158–178 (2005)
17. Totohasina, A., Ralambondrainy H.: ION, a pertinent new measure for mining information from many types of data. In: IEEE, SITIS, pp. 202–207 (2005)
18. Hämmäläinen, W.: Kingfisher: an efficient algorithm for searching for both positive and negative dependence rules with statistical significance measures. *Knowl. Inf. Syst.* **32**, 383–414 (2012)
19. Wu, X., Zhang, C., Zhang, S.: Efficient mining of both positive and negative association rules. *ACM Trans. Inf. Syst.* **3**, 381–405 (2004)



# Field-Reliability Predictions Based on Statistical System Lifecycle Models

Lukas Felsberger<sup>1,2(✉)</sup>, Dieter Kranzlmüller<sup>1</sup>, and Benjamin Todd<sup>2</sup>

<sup>1</sup> Institut für Informatik, Ludwig Maximilians Universität Muenchen, Oettingenstr. 67, 80538 München, Germany

<sup>2</sup> CERN, Route de Meyrin, 1211 Genève, Switzerland  
lukas.felsberger@cern.ch

**Abstract.** Reliability measures the ability of a system to provide its intended level of service. It is influenced by many factors throughout a system lifecycle. A detailed understanding of their impact often remains elusive since these factors cannot be studied independently. Formulating reliability studies as a Bayesian regression problem allows to simultaneously assess their impact and to identify a predictive model of reliability metrics.

The proposed method is applied to currently operational particle accelerator equipment at CERN. Relevant metrics were gathered by combining data from various organizational databases. To obtain predictive models, different supervised machine learning algorithms were applied and compared in terms of their prediction error and reliability. Results show that the identified models accurately predict the mean-time-between-failure of devices – an important reliability metric for repairable systems - and reveal factors which lead to increased dependability. These results provide valuable inputs for early development stages of highly dependable equipment for future particle accelerators.

**Keywords:** Reliability prediction · System lifecycle  
Bayesian learning

## 1 Introduction

Reliability measures the ability of a system to perform as expected during its intended lifetime. The field-reliability of complex repairable systems is a result of all actions during all stages of its system lifecycle. These stages are (1) conceptual design, (2) detailed design and testing, (3) manufacturing, (4) installation, (5) operation and maintenance, and (6) phase-out and disposal. At each stage an interplay of complex technical, organizational, and human processes leads to a more or less desirable outcome in terms of system reliability.

---

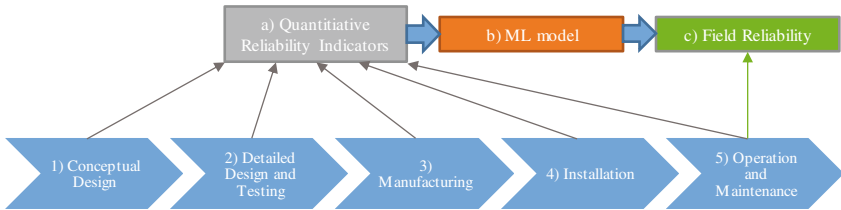
This work has been sponsored by the Wolfgang Gentner Programme of the German Federal Ministry of Education and Research (grant no. 05E12CHA).



An assessment of all stages and processes is not feasible, since models capturing the interactions between all relevant processes in system development do not exist. Therefore, most common reliability methods focus on certain stages and aspects during a system lifecycle, which can be modeled and understood - we provide an overview in Sect. 2. However, such methods struggle to quantify the overall uncertainty of reliability predictions in a systematic way since relevant contributions during a system lifecycle might have been disregarded and are not straight-forward to include.

Instead of focusing on models for certain stages and aspects of a system we propose to learn a statistical model of the whole product lifecycle to predict the observed field-reliability with machine learning techniques as depicted in Fig. 1. For a set of existing comparable systems with known field-reliability so-called *quantitative reliability indicators* are gathered. Using the reliability indicators as input variables and the field-reliability metric as target variables, a statistical reliability model is learned by a supervised machine learning algorithm.

The learned model will always be an approximation of the true underlying system lifecycle processes. The lost accuracy due to the statistical model and the limited granularity of the reliability indicators can be quantified by Bayesian methods. Thereby, the overall predictive certainty can be quantified in an efficient way based on the available data.



**Fig. 1.** Illustration of the proposed approach. The achieved field-reliability (c) can be seen as the result of relevant processes during the whole product lifecycle (1–5). It is not feasible to capture and model all of the relevant processes. Instead, it is proposed to learn a reduced-order statistical lifecycle model (b) with machine-learning algorithms based on *quantitative reliability indicators* (a).

We demonstrate that the learned models accurately predict reliability metrics even with a limited set of reliability indicators (as is the case at early stages of a system’s lifecycle). Compared to traditional reliability assessment methods, this leads to a reduced workload for reliability predictions and to a systematic quantification of uncertainties. Furthermore by an appropriate choice of reliability indicators and machine learning algorithms one can study the influence of each individual reliability indicator. This information assists engineers in design decisions for highly reliable systems.

The rest of the paper is structured as follows: In Sect. 2, we present related methods to reliability predictions. In Sect. 3, we explain the methodology of our approach and in Sect. 4 we apply it to a use-case.

## 2 Literature Review

A general review of the challenges in reliability studies is given in [23]. The author of [23] concludes that the two major challenges in reliability studies are complexity and uncertainty. Reliability studies must consider technical, organizational and human factors each of which influences the field-reliability of systems. In the following paragraph a selection of reliability prediction methods to tackle these problems is given.

*Reliability Engineering Methods.* Scientific literature on reliability engineering prediction methods of electronic systems is numerous. An attempt to classify and evaluate the existing methods is given in the IEEE standard 1413 [6, 19] and its successors. In this standard they have been classified as based on

- handbooks,
- stress and damage models (often referred to as physics-of-failure based), and
- field-data.

Most methods are based on early designs of the considered system and the selected components.

A common criticism for handbook based models is that they do not consider interactions of components but only single-component faults. However, faults due to single-component failures are not dominant [1, 5, 7, 14, 18]. As a result the actual field-reliabilities can deviate from the predicted ones by orders of magnitude [12]. The author of [5] argues that some methods should not be used to predict the field-reliabilities but rather as part of a review process at a stage when limited information on the final design is available.

Stress- and damage models are in general more accurate than handbook-based methods. However, the development of such methods requires more effort [18].

Instead of assessing the system on the component level, some approaches use a top-down approach in which the field-reliability of new systems is estimated from field-data of similar systems in operation [9, 11].

*Reliability Program Assessment.* A different approach to evaluate the field-reliability of systems is taken in [16]. The likelihood of achieving the required field-reliability is estimated by a review of the design processes. Each system is assigned a score depending on its design processes and it is shown that this score correlates with the probability of fulfilling field-reliability requirements. Thereby organizational aspects of reliability are taken into account.

*Organizational and Human Reliability Analysis.* In the review article [23] Section 3.1.3 is dedicated to non-technical factors in reliability studies since its contribution to the field-reliability can be significant.

In our work we propose to infer the most relevant processes or factors in a system lifecycle from the field-reliability data of a set of systems. This includes organizational and human reliability factors. The method can be applied at any stage of a system lifecycle to guide engineering decisions.

### 3 Methodology

In this section we define the relevant terms, explain the methods used and describe the general methodology.

#### 3.1 Definitions

*System Reliability.* It is generally defined as the ability of a system to provide its intended level of services for a specified time  $t$ . For a constant failure rate and repairable systems, it is usually measured as availability  $A$ , which is defined by

$$A = \frac{MTBF}{MTBF + MTTR} \quad (1)$$

with  $MTBF$  being the mean-time-between-failure and  $MTTR$  being the mean-time-to-repair. The  $MTBF$  is being calculated as

$$MTBF = \frac{t_{operation}}{n_{faults}} \quad (2)$$

with  $t_{operation}$  being the cumulative operational time of the considered devices and  $n_{faults}$  being the total number of faults within the operational time. The  $MTTR$  can be evaluated by

$$MTTR = \frac{t_{inrepair}}{n_{faults}} \quad (3)$$

with  $t_{inrepair}$  being the total time a system is in repair and  $n_{faults}$  the total number of faults during the operational time. The un-availability  $U_A$  is given by  $U_A = 1 - A$ .

*System Lifecycle.* It is the overall process describing the lifetime of a system. It is a concept from systems engineering to address all stages of a product from its beginning to end. Here these stages shall be divided into (1) conceptual design, (2) detailed design and testing, (3) manufacturing, (4) installation, (5) operation and maintenance, (6) and phase-out and disposal.<sup>1</sup>

*System Definition.* This discussion is focused on repairable electronic systems. A more precise definition will be given for the use-case in Sect. 4.<sup>2</sup>

---

<sup>1</sup> Depending on the system under study the definitions of the stages may change. The proposed methodology is not restricted to this specific choice of stages.

<sup>2</sup> There is no implicit restriction for the proposed method to electronic repairable systems. It can also be used for non-repairable systems and for mechanic, electric, electronic, or software systems. However, the definitions of the fault metrics must be adapted.

### 3.2 Method

The central assumption is that the observed field-reliability is the outcome of all technical, organizational and human processes during all stages of a system's lifecycle. It is unfeasible to model all these interactions due to their complexity and non-linearity. Therefore, we restrict ourselves to learning statistical models of the observed field-reliability of comparable systems based on reliability indicators collected throughout the system lifecycle. Modern machine learning algorithms are capable of learning accurate predictive models of field-reliability based on the relevant reliability indicators. The loss of information due to the limited availability of data and the intrinsic uncertainty of the problem can be assessed by using Bayesian machine learning methods.

**Lifecycle Analysis by Machine Learning.** To arrive at a firm mathematical description of the proposed method, let us hypothesize the existence of a deterministic model  $\mathbf{F} : \mathcal{Z} \mapsto \mathcal{Y}$  to determine any field-reliability metric  $\mathbf{Y} \in \mathcal{Y}$  from all relevant input variables  $\mathbf{Z} \in \mathcal{Z}$  in the form of

$$\mathbf{Y} = \mathbf{F}(\mathbf{Z}). \quad (4)$$

This would be a model to quantify the contribution of all relevant processes towards the field-reliability during the whole system lifecycle. Since it is not possible to derive such a formula or to gather all relevant inputs, we try to approximate the true field-reliability metrics  $\mathbf{Y}$  by a reduced model

$$\mathbf{Y} \approx \mathbf{y} = f(\mathbf{x}), \quad (5)$$

with  $\mathbf{x} \in \mathcal{X}$ ,  $\dim(\mathcal{X}) \ll \dim(\mathcal{Z})$  being the set of collected reliability indicators and  $f : \mathbf{x} \mapsto \mathbf{y}$ ,  $\mathbf{y} \in \mathcal{Y}$  being an approximate model. When supplied with pairs of input and output data  $\mathcal{D} = \{(\mathbf{x}_1, \mathbf{Y}_1), \dots, (\mathbf{x}_N, \mathbf{Y}_N)\}$ , a statistical learning algorithm can learn such a model by minimizing a certain loss function  $l : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$ . This is essentially a regression problem which can be studied with a vast range of learning algorithms.

There are three additional requirements, which render algorithms fitter for the intended purpose. Firstly, to quantify the uncertainty of the predictions of the reliability metrics, probabilistic models shall be learned

$$p(\mathbf{Y}|\mathbf{x}). \quad (6)$$

Our method is based on an arbitrary non-linear mapping from reliability indicators to features  $\Phi : \mathcal{X} \mapsto \mathbb{R}^n$ . Since it is of interest which features are relevant, secondly, algorithms of parametric form will yield that additional information,

$$p(\mathbf{Y}|\mathbf{w} \cdot \Phi(\mathbf{x})), \quad (7)$$

with  $\mathbf{w} \in \mathbb{R}^n$  being a weight vector indicating the relevance of each feature. Thirdly, methods learning sparse models based on fewer features are preferred

from a practical point of view, since they require a reduced data collection effort for predicting field-reliability. A general justification of such methods on philosophical grounds is given by Occam's razor [8].

We present concrete algorithms fulfilling these criteria in Sect. 4. Even though the outlined requirements are not mandatory, they facilitate the data collection and model assessment process by providing additional feedback.

**Data Collection, Model Selection and Reliability Prediction.** The collection of data and the training and selection of a model should be seen as an integrated process. The problem domain and a-priori available expert knowledge allows to draw guidelines for the data collection. We present these guidelines in the paragraphs below. After that, we show how to learn a predictive model with the collected data and how further refinements of the data collection are assisted by properly selected learning algorithms.

*Collection of Training Systems.* Since the method is based on the field-reliability of existing comparable systems, the choice of the collected systems will have an influence on the accuracy of the predictions for future systems. Two general recommendations can be given for this selection:

- Only systems which have been in use for a significant exploitation period with accurately monitored reliability metrics shall be used.
- The choice of systems for which a field-reliability model is learned shall include systems which are comparable to the system for which a field-reliability shall be predicted. In reliability studies, comparable systems are similar in terms of technical, organizational, and human factors throughout their lifecycle.

*Collection of Reliability Indicators.* The choice of these indicators largely influences the quality of the models in terms of their accuracy and interpretability. The following statements can be made:

- Based on expert knowledge, recommendations can be given for indicators which carry important reliability information; e.g. operational conditions such as load, temperature or humidity can contribute significantly to the failure rate. Systems, which are mass-produced, will achieve different field-reliabilities than prototypes. Different organizational structures or project management strategies influence the reliability of a final system. [17, 22].
- In engineering practice the collection of data is facing practical limitations due to time or other restrictions. Therefore, a natural choice is to begin to collect the indicators based on a trade-off between collection effort and expected information content. For the use-case in Sect. 4, we show that accurate predictions can be obtained from a very limited set of meta-variables as reliability indicators. Furthermore, one always needs to consider the availability of the indicators for the systems in the data-set.<sup>3</sup>

---

<sup>3</sup> If availability indicators are unavailable for some of the selected systems, supervised learning techniques for incomplete data-sets can be employed.

*Collection of System Reliability Metrics.* The choice of reliability metrics is usually given by the system under study. For our choice of system and assuming a constant failure rate,<sup>4</sup> these are given by *MTBF* and *MTTR*. Based on these other metrics can be derived.

*Model Selection and Validation.* Using the collected data, one is able to compile a data-set  $\mathcal{D} = \{(\mathbf{x}_1, \mathbf{Y}_1), \dots, (\mathbf{x}_N, \mathbf{Y}_N)\}$  for which  $\mathbf{x}_i$  and  $\mathbf{Y}_i$  are the collected reliability indicators and the field-reliability metrics for system  $i$ , respectively. A reliability model shall be learned with this data-set. We use a general model selection and assessment approach as is e.g. discussed in Chapter 7 of [10] with minor modifications due to the particularities of the problem setting.

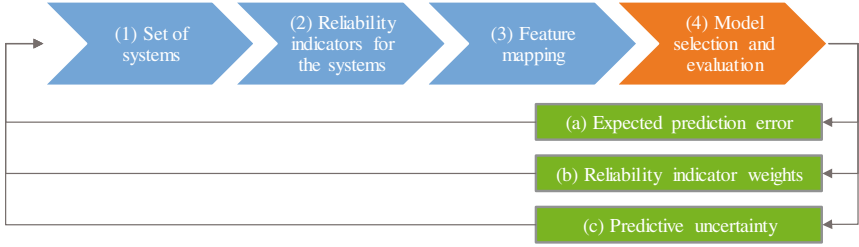
The first step is to split the data-set into a training data-set  $\mathcal{D}_{train}$  and a test data-set  $\mathcal{D}_{test}$ . This splitting is not performed arbitrarily. Instead the training data-set shall contain systems with an age higher than a certain threshold age  $a_s$  and the test data-set shall exclusively contain systems younger than the threshold age. Thereby, we test the approach for its applicability to future systems. For the model selection and assessment the training data-set will be used exclusively.<sup>5</sup> With a five-fold cross-validation method we compare different learning algorithms in terms of their applicability to the problem setting and their prediction errors. For algorithms which additionally require the tuning of hyperparameters, we used so-called nested cross-validation in which the hyperparameters are optimized in a five-fold inner cross-validated grid-search nested within each of the five outer cross-validation folds [4]. The expected mean and variance of the cross-validation error  $Err_{CV}$  is reported for each of the evaluated models. It serves as an estimate for the expected generalization error [10].

The confidence of the predictions and the relevance of the selected reliability indicators can be studied with a learning algorithm which satisfies Eqs. 6 and 7. Investigating the identified model parameters and predictions obtained by such an algorithm for one or several cross-validation folds gives this additional information. The confidence or uncertainty of the predictions provides feedback on the amount and quality of the collected data. The weight vector  $\mathbf{w}$  indicates the relevance of the features and the reliability indicators. Depending on the complexity of the mapping  $\Phi : \mathcal{X} \mapsto \mathbb{R}^n$  from the reliability indicators to the features we can identify the most important reliability indicators. Using this information and expert knowledge, we can refine our data-set (choice of systems and reliability indicators) and feature mapping  $\Phi$  to obtain more precise models. This idea is illustrated in Fig. 2

*Obtaining Reliability Predictions.* Once satisfying models in terms of their predictive errors and interpretability are found with the procedure described above,

<sup>4</sup> This assumption can be relaxed by e.g. predicting a parameterized failure rate distribution over time. Then, instead of *MTBF* and *MTTR* the reliability metrics are the parameters of the distribution. This requires a different data collection and can be considered for future work.

<sup>5</sup> Using the test data-set would lead to an over-fitting of the models and an underestimation of the generalization error.



**Fig. 2.** Illustration of the iterative data collection and reliability prediction process. The choice of (1) systems, (2) reliability indicators and (3) feature mappings influences the quality of the predictive model (4). The learning algorithm provides feed-back in the form of an expected prediction error (a), relevance weights for the reliability indicators (b) and uncertainty bounds for the field-reliability predictions (c).

they are tested with the full data-set. Since the data-set is split by the age of the systems, this testing simulates a prediction scenario - we identify a model based on data of systems in the past and evaluate its applicability to future systems.

The predictive models are now trained with the whole training data-set.<sup>6</sup> Based on the input values of the test data-set  $\mathbf{x}_{\text{test}}$  the models can predict the expected field-reliability  $\mathbf{y}_{\text{test}}$ . As the prediction is simulated, we know the observed field-reliabilities  $\mathbf{Y}_{\text{test}}$  and can compare these to the predicted ones to obtain the test error  $Err_{\text{test}}$ .<sup>7</sup> When the test error is of the order of the expected generalization error  $Err_{CV}$  obtained during model selection and validation one can conclude that the model is capable of predicting the field-reliability for new systems.

The overall data collection, model selection and reliability prediction process is summarized in the pseudo-algorithm below. The use-case in Sect. 4 follows the presented procedure closely.

*Pseudoalgorithm illustrating the overall model selection and reliability prediction process:*

1.  $\mathcal{D} = \{(\mathbf{x}_1, \mathbf{Y}_1), \dots, (\mathbf{x}_N, \mathbf{Y}_N)\} \leftarrow$  Initial data collection.
2. Sort  $\mathcal{D}$  by system age.
3. Split  $\mathcal{D}$  in  $\mathcal{D}_{\text{train}}$  and  $\mathcal{D}_{\text{test}}$  with  $a_{\text{test}} < a_s \leq a_{\text{train}}$ .
4. **While** satisfying predictive model has not been found **do**:
  - (a) Shuffle  $\mathcal{D}_{\text{train}}$  randomly.
  - (b) Evaluate  $Err_{CV}$  by (nested) CV.
  - (c) Evaluate parameter weights  $\mathbf{w}$  and predictive uncertainty for one fold.
  - (d) **If** Model has large  $Err_{CV}$  or predictive uncertainty **then**

<sup>6</sup> Again, hyperparameters are optimized by a cross-validated grid search over a hyperparameter grid.

<sup>7</sup> We note that in a realistic application scenario the true observed field-reliabilities are not available. However, the available data can always be split by system age to test the generalizability of the identified models to newer systems.

- Change set of systems, reliability indicators, or feature mapping.
- (e) **Else** jump to 5.
- 5. Train predictive model with  $\mathcal{D}_{train}$ .
- 6. Evaluate  $Err_{test}$  and compare with  $Err_{CV}$ .
- 7. Evaluate parameter weights  $\mathbf{w}$  and predictive distributions.

## 4 Use-Case

This section describes how the proposed method was used to learn a model for the expected field-reliability of accelerator power converters. The system of interest, the collected data and features, the used learning algorithms and the results are discussed.

**System Definition.** The considered systems are magnet power converters at the CERN particle accelerator facilities. A power converter is a device to transform electrical energy. The conversion is in terms of voltage, current and waveform. Magnet power converters control the flow of current through particle accelerator magnets. In order to achieve precise magnetic fields these converters generally need to control the output current very precisely.

### Dataset, Reliability Metrics and Reliability Indicators

*Set of Systems.* At CERN there are currently more than 6000 power converters of approximately 600 different types in use. Their field-reliability is continuously tracked by a centralized computerized maintenance management system (CMMS). After removal of converter types with a cumulative operational time  $t_{operation}$  of less than ten years and cleaning the data, approximately 300 power converter types remained for reliability analysis. Table 1 gives an overview of minimal and maximal characteristic attributes of power converters in the dataset. Considering the vast range of converter types one would not expect a global model to accurately predict the field-reliability. Therefore, both local- and global-models were trained.

**Table 1.** Illustration of characteristic power converter attributes of the studied dataset.

	Power [W]	Current [A]	Voltage [U]	Age [yrs]	MTBF [hrs]
Minimum	$10^{-6}$	$10^{-4}$	$10^{-3}$	2.2	$10^3$
Maximum	$10^8$	$4 \cdot 10^4$	$10^5$	49.7	$6 \cdot 10^5$



*Reliability Indicators for the Systems.* The initial choice of reliability indicators depends on

- the system development stage at which the prediction shall be carried out,
- recommendations from system experts,
- the time or effort which can be attributed to the data collection.

The following list shows the collected reliability indicators. The selection is based on recommendations from CERN engineers in charge of the complete lifecycle of the studied systems. Naturally, the selection is also limited by the availability of data:

- I: Rated current of the converter. Depending on the rated current different converter technologies have to be used. One major stress effect of high currents in terms of reliability is an increased heat load which requires a proper heat management [13, 17].
- U: Rated voltage of the converter. Higher voltages require the appropriate electrical insulation and can cause failure mechanisms such as arcing or corona discharge [13, 17].
- P: Rated power of the converter. Similarly to currents increased power leads to increased heat loads.
- Quantity: This refers to the quantity of each type of power converter that is used at CERN. The quantity of a power converter is not related to a physical wear-out mechanism. However, throughout the lifecycle converter types produced and operated in large quantities are treated differently than power converters of small quantities both in terms of technical and organizational matters.
- Avg. Age: The average age of converters for each converter type. Depending on the maintenance strategy a decreasing or constant availability as a function of the age is expected, respectively.
- Cum. Age: The cumulative age of converters for each converter type. A dependency of the availability on the cumulative age could indicate both a organizational learning curve in terms of a more efficient maintenance and a degradation with age of the converters.
- Pol 0–9: The polarity of the converter. This indicates the operating modes, technology and complexity of the converter.<sup>8</sup>
- Acc. 1–9: The accelerator in which the converter type is used. Depending on the accelerator the converter type is exposed to different operating conditions<sup>9</sup> and operation modes.

---

<sup>8</sup> The discrete set of polarities is given by: (1) Unipolar, (2) Bipolar Switch Mechanic, (3) Bipolar I - Unipolar U - 2 Quadrants, (4) Unipolar I Bipolar U 2 Quadrants, (5) Bipolar Pulse-Width-Modulation, (6) Bipolar Relay, (7) Bipolar Electronic I/U, (8) Bipolar Anti-Parallel 4 Quadrants, (9) Bipolar I-circulation 4 Quadrants and, (0) un-specified or other Polarity.

<sup>9</sup> E.g. the radiation levels differ on the kind of accelerator. However, there is also different operation conditions within each of the accelerators.

- in Acc.: The number of different particle accelerators in which each power converter is used.

We probed different indicators for their information content by appropriate Bayesian learning methods. The required learning algorithms are introduced later in this section.

*Reliability Metrics for the Systems.* The studied field-reliability metrics are *MTBF* and *MTTR* as defined in Sect. 3.<sup>10</sup> These are directly computed in the CMMS with the necessary variables for power converter type  $i$  which are defined as follows:

- $t_{operation,i}$ : Cumulative time in operation of all converters of converter type  $i$ . Note that commissioning and testing times are not counted towards operation time.
- $n_{faults,i}$ : Cumulative number of faults of all converters of converter type  $i$  during the operational time  $t_{operation,i}$ . Note that only internal faults of the system which require an external action to alleviate the problem are included. Internal faults which are automatically resolved or are very short and faults due to external reasons are not included. This ensures that a model for the reliability of the considered systems itself is learned and not of its surroundings.
- $t_{inrepair,i}$ : Cumulative time in repair of all converters of converter type  $i$  during the operational time  $t_{operation,i}$ . The repair time starts by a request from the system operators to the system experts and ends when the problem was resolved and the system can continue to operate.

**Algorithms.** By formulating the reliability prediction problem as a supervised machine learning problem we can choose from a range of existing learning algorithms to generate the desired statistical model for predictive purposes. Since the uncertainty in the field-reliability predictions shall be quantified (i.e. finding a model as presented in Eq. 6), the choice of algorithms is narrowed down. Furthermore, sparse parametric models (as in Eq. 7) are preferred since they potentially require fewer reliability indicators to be collected and - more importantly - since they allow an estimation of the relevance of the choice of reliability indicators and the generated features.

A summary of the chosen algorithms is given in Table 2. Note that the scikit-learn python implementations of the algorithms were used [20]. A detailed description of each algorithm can be found on their website and in their user-guide [3]. Since the algorithms are standard implementations, only references to detailed documentation are given:

- ARD - Automatic Relevance Determination Regression: Sparse Bayesian regression technique as described in [2] - Chapter 7.2.1. The implementation is taken from [3] - Chapter 1.1.10.2.

<sup>10</sup> Note that the Availability  $A$  and Un-Availability  $U_A$  can be directly obtained from the *MTBF* and the *MTTR*.

**Table 2.** Summary of learning algorithms.

	UQ (6)	Feature weights (7)	Sparsity	Global/Local
ARD	Yes	Yes	Yes	Global
BAR	Yes	Yes	Balanced	Global
GP	Yes	No	No	Local
ENCV	No	Yes	Yes	Global
SVR	No	Only for linear kernel	no	Local

- BAR - Bayesian Ridge Regression: A Bayesian regression method as introduced in [15]. It is similar to the ARD Regression but fewer parameters have to be determined from the data. The implementation is taken from [3] - Chapter 1.1.10.1.
- GP - Gaussian Process Regression. A kernel-trick based Bayesian Regression technique. The implementation is described in [21] - Algorithm 2.1 and was taken from [3] - Chapter 1.7.1. The kernel is based on a combination of a radial-basis-function kernel and a white-kernel. The kernel parameters were optimized in the learning process.
- EN: Elastic Net Regression. The implementation is taken from [3] - Chapter 1.1.5 - which includes a description of the algorithm. Hyperparameters were optimized in a cross-validated grid-search.
- SVR - Support Vector Machine Regression: A kernel-trick based regression method. A description is given in [3] - Chapter 1.4.2. Linear basis functions were used and the hyperparameters were optimized by a cross-validated grid-search.

**Model Selection and Validation.** This section closely follows the procedure presented in Sect. 3. The data-set  $\mathcal{D}$  was compiled from the data collection described above including 281 collected systems, nine reliability indicators and two field-reliability metrics. To simulate a prediction scenario the whole data-set of 281 different converter types was split into a training set  $\mathcal{D}_{train}$  with 210 converter types which are at least fifteen years old and a test set with 71 converter types which are less than fifteen years old.<sup>11</sup>

For the model selection and validation we restricted ourselves to the training data which we shuffled randomly. A scaling operator re-scaled the features or inputs  $\mathbf{x}_{train}$  to zero mean and unit variance. The same scaling operator was later applied to the features in the test data-set  $\mathbf{x}_{test}$ . Furthermore, the logarithms of the reliability metrics  $\log(\mathbf{Y})$  were taken instead of their nominal value for the full data-set.

<sup>11</sup> In other words we pretended to be in 2003 and tried to predict the field-reliability of power converters between 2003 and 2018.

Based on the introduced (nested) cross-validation we compared the following different choices of the set of systems, reliability indicators and feature mappings for all the introduced algorithms:

- Choice of systems: We trained models with the complete set of power converter types and with a random sub-selection of only 42 converter types.
- Choice of reliability indicators: We trained models with the complete set of reliability indicators and a set in which the quantity of converters per type was removed.
- Choice of feature mapping: Based on the reliability indicators, following features were generated:
  - Based on the numeric indicators  $\mathbf{x}_{\text{num}}$  linear features and logarithmic features were chosen -  $\bar{\Phi}(\mathbf{x}_{\text{num}}) = [\mathbf{x}_{\text{num}}, \log(\mathbf{x}_{\text{num}})]^T$ .
  - The categorical indicators  $\mathbf{x}_{\text{cat}}$  were split into binary features, whereas the number of binary variables corresponds to the number of categories per categorical variable.

A feature vector of 34 dimensions was obtained by combining all features. This was the first choice for the feature mapping and we refer to it as first-order feature mapping.

The second choice of feature mapping accounts for second-order interactions of the numeric variables and we refer to it as second-order feature mapping:

$$\bar{\Phi}(\mathbf{x}_{\text{num}}) = \left[ \mathbf{x}_{\text{num}}, \log(\mathbf{x}_{\text{num}}), [\mathbf{x}_{\text{num}}, \log(\mathbf{x}_{\text{num}})] \cdot [\mathbf{x}_{\text{num}}, \log(\mathbf{x}_{\text{num}})]^T \right]^T. \quad (8)$$

By this more complex mapping we obtain 629 features. One could expect that a more accurate model can be learned when including second-order interactions which is balanced by a lack of interpretability of the individual feature weights.

In the following we report the results of our model selection procedure. For each algorithm the cross-validation error  $Err_{CV}$ <sup>12</sup> is reported and the feature weights  $\mathbf{w}$  of the learned models and the obtained predictions are plotted for the last cross-validation fold.<sup>13</sup> All results are provided in terms of the two chosen reliability metrics *MTBF* and *MTTR*.

*Reference Configuration.* The first configuration we studied is based on the complete set of power converters, the complete set of reliability indicators and the first-order feature mapping. The cross-validation errors  $Err_{CV}$  are given in Table 3a for the *MTBF* and in Table 4a for the *MTTR*. As the values for the

<sup>12</sup> Note that the mean-squared-error was used throughout.

<sup>13</sup> Note that only predictions obtained by the BAR algorithm are illustrated due to space limitations. It assigns relevance weights to the feature functions and it quantifies uncertainties of both the field-reliability predictions and the feature function weights. Therefore, it is suited to study the earlier mentioned additional information provided by algorithms which satisfy Eqs. 6 and 7.

**Table 3.** Obtained mean-squared-errors for the  $\log(MTBF)$  - (a)  $Err_{CV}$  for the reference model, (b)  $Err_{CV}$  for a reduced set of systems, (c)  $Err_{CV}$  for a reduced set of reliability indicators, (d)  $Err_{CV}$  for non-linear numeric feature mappings, and (e)  $Err_{test}$  for the predictions of the test data-set. Comparison of (a) and (e) indicates if the method can be extended to future converter types.

	ARD	BAR	GP	EN	SVR
$Err_{CV}$ (a)	0.39±0.15	0.35±0.13	0.37±0.14	<b>0.34 ± 0.12</b>	0.46±0.16
$Err_{CV}$ (b)	<b>0.90 ± 0.79</b>	0.82±0.73	0.81±0.74	0.65±0.49	<b>0.64 ± 0.50</b>
$Err_{CV}$ (c)	1.03±0.24	<b>1.00 ± 0.19</b>	<b>1.00 ± 0.19</b>	1.01±0.22	1.02±0.24
$Err_{CV}$ (d)	0.59±0.23	0.37±0.05	0.38±0.05	<b>0.32 ± 0.05</b>	0.48±0.12
$Err_{test}$ (e)	<b>0.30</b>	0.33	0.32	<b>0.30</b>	0.38

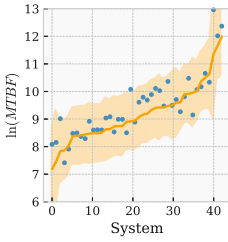
**Table 4.** Obtained mean-squared-errors for the  $\log(MTTR)$  - (a)  $Err_{CV}$  for the reference model, (b)  $Err_{CV}$  for a reduced set of systems, (c)  $Err_{CV}$  for a reduced set of reliability indicators, (d)  $Err_{CV}$  for non-linear numeric feature mappings, and (e)  $Err_{test}$  for the predictions of the test data-set.

	ARD	BAR	GP	EN	SVR
$Err_{CV}$ (a)	0.23±0.05	0.22±0.004	0.22±0.04	0.22±0.04	0.23±0.05
$Err_{CV}$ (b)	0.32±0.17	0.24±0.11	0.24±0.12	0.23±0.09	0.25±0.17
$Err_{CV}$ (c)	0.30±0.16	0.23±0.06	0.23±0.06	.28±0.11	0.29±0.16
$Err_{CV}$ (d)	3.12±4.83	0.23±0.02	0.23±0.03	0.22±0.02	0.34±0.06
$Err_{test}$ (e)	0.38	0.35	0.35	0.35	0.36

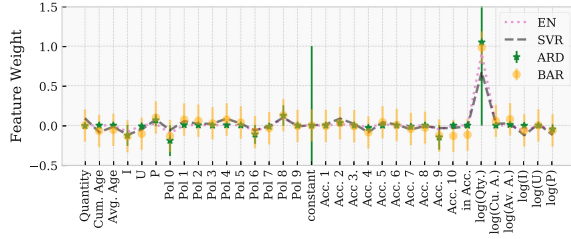
target variables were not scaled to unit-variance but simply by a logarithmic function, the values from the  $MTBF$  table cannot be compared with values of the  $MTTR$  table. Only values within a table are comparable. We noted that all algorithms yielded models with comparable cross-validation errors.

The obtained parameter weights  $\mathbf{w}$  for the last cross-validation fold are shown in Fig. 3b for the  $MTBF$  and in Fig. 3d for the  $MTTR$ . All algorithms identified similar models. For the  $MTBF$  the dominant parameter was the logarithm of the quantity of converters per type  $\log(Qty)$  for all models and the rated power  $P$  was dominant for the  $MTTR$ . From the predictions obtained with the BAR algorithm for the last fold, Fig. 3a for the  $MTBF$  and Fig. 3c for the  $MTTR$ , we noted that the model for the  $MTTR$  did not identify a significant variation whereas the  $MTBF$  was predicted properly. We concluded that a precise model for the  $MTBF$  had been learned with the collected data, the selected feature mappings, and algorithms. For the  $MTTR$  no such model could be identified and a further refinement would be necessary.

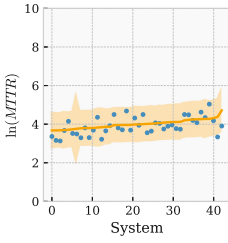
In the following we present variations of the reference configuration in terms of selected systems, reliability indicators and feature mappings.



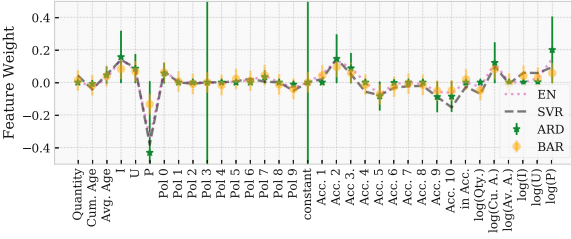
(a)



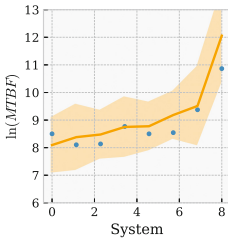
(b)



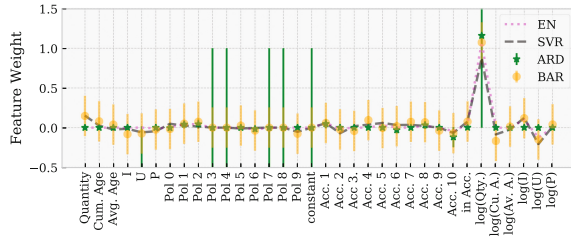
(c)



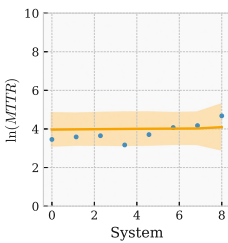
(d)



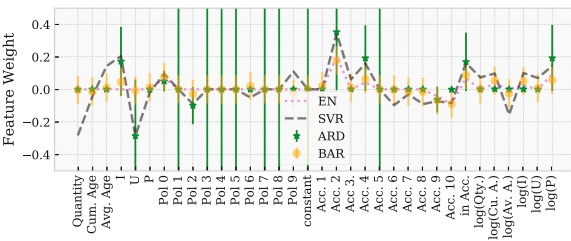
(e)



(f)



(g)



(h)

**Fig. 3.** (a), (c), (e), (g): Prediction of the  $\log(MTBF)/\log(MTTR)$  for the last fold of the cross-validation procedure. The orange line depicts the mean of the predictive distribution and the orange shaded area the 95% confidence intervals. The blue dots mark the actual observed field-reliabilities. Note that the different converter types were ordered by the mean of the predictive distribution for illustration purposes. (b), (d), (f), (h): Estimated feature weights for the parametric models. Figures (a), (b), (c), (d) are for the reference configuration and figures (e), (f), (g), (h) for a reduced set of data items in the learning data.

*Reduced Set of Training Systems.* The second configuration is similar to the reference configuration except for using a random sub-selection of only 42 converter types in the training data-set. This illustrates the dependence of the confidence levels of the identified feature weights  $\mathbf{w}$  and the predictions for the Bayesian algorithms (ARD and BAR) on the amount of training data.

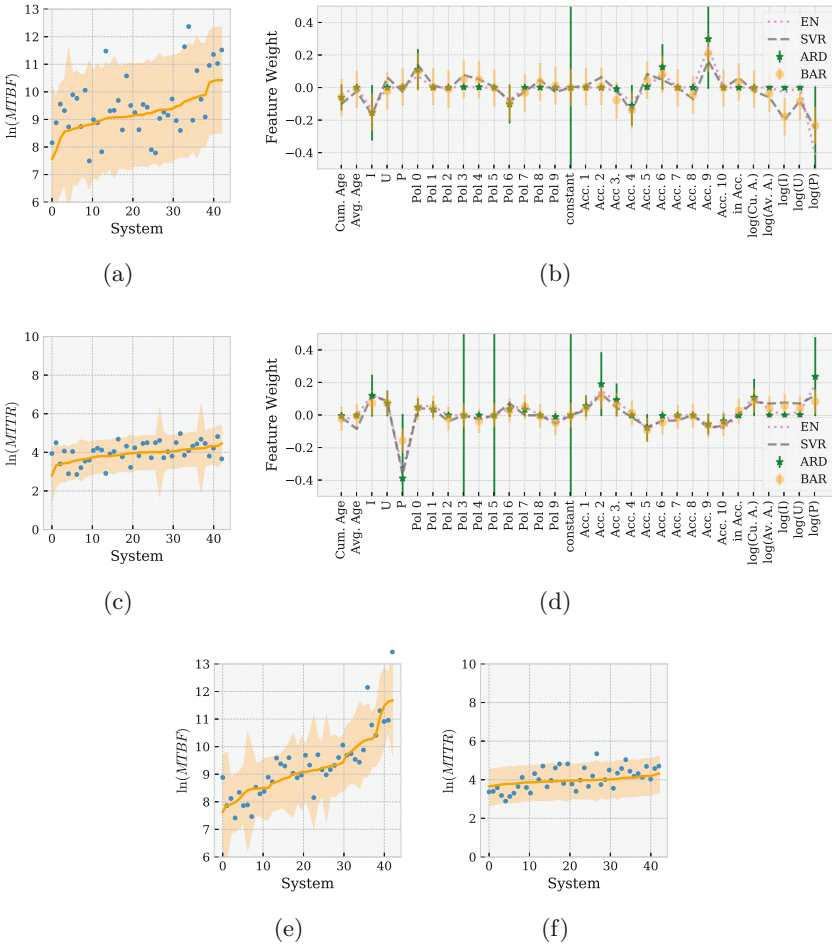
The cross-validation errors  $Err_{CV}$  in Table 3b for the *MTBF* and Table 4b for the *MTTR* were larger than those of the reference configuration. The obtained parameter weights  $\mathbf{w}$  for the last cross-validation fold in Fig. 3f (*MTBF*) and in Fig. 3h (*MTTR*) deviated slightly in absolute terms and largely in terms of their confidence levels from the reference configuration for the ARD and BAR algorithm. The predictive uncertainties of the BAR algorithm in Fig. 3e (*MTBF*) and Fig. 3g (*MTTR*) increased only slightly in comparison with the reference configuration. Again, no predictive model of the *MTTR* could be identified. We concluded that a reduced set of training data manifests itself in increased uncertainties in parameters or predictions.

*Reduced Set of Reliability Indicators.* The third configuration resembles the reference configuration except for the removal of the variable indicating the quantity of systems per converter type. This variable had been identified as the single most important reliability indicator for *MTBF* predictions.

The cross-validation errors  $Err_{CV}$  in Table 3c for the *MTBF* were much larger than those of the reference configuration and slightly larger for the *MTTR* models (Table 4c). The obtained parameter weights  $\mathbf{w}$  for the last cross-validation fold in Fig. 4b for the *MTBF* models were totally different than the reference configuration. The weights for the *MTTR* models (Fig. 4d) were similar to the reference configuration. The predictive uncertainties of the BAR algorithm in Fig. 3e increased drastically for the *MTBF* and only slightly for the *MTTR* (Fig. 3g) in comparison with the reference configuration. This is consistent with our expectation, since we removed the most important reliability indicator for the *MTBF* models. This time no proper predictive model of either the *MTTR* or the *MTBF* could be identified. We concluded that the choice of reliability indicators has a strong influence on the quality of the models.

*Second-Order Feature Mapping.* In the fourth configuration the second-order feature mapping replaces the first-order mapping of the reference configuration. The cross-validation errors  $Err_{CV}$  in Table 3d for the *MTBF* and Table 4d for the *MTTR* were of the same order as those of the reference configuration except for the model learned with the ARD algorithm. The 629 obtained parameter weights  $\mathbf{w}$  were not illustrated. The predictions of the BAR algorithm in Fig. 4e (*MTBF*) and in Fig. 4f (*MTTR*) were comparable with the reference configuration. No model could be identified for the *MTTR*. We concluded that the extended feature mapping does not improve the predictive errors and complicates the interpretation of the models.

**Prediction.** The reference configuration was used for the prediction scenario as it had shown to be interpretable and predicted the *MTBF* in the model

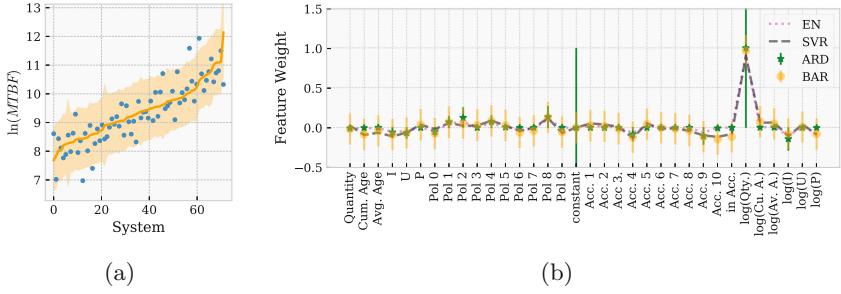


**Fig. 4.** (a), (c), (e), (f): Prediction of the  $\log(MTBF)/\log(MTTR)$  for the last fold of the cross-validation procedure. The orange line depicts the mean and the orange shaded area the 95% confidence intervals. The blue dots mark the actual observed field reliabilities. Note that the different converter types were ordered by the mean of the predictive distribution. (b), (d): Estimated feature weights for the parametric models. Figures (a), (b), (c), (d) are for the configuration with a reduced set of reliability indicators and figures (e), (f) for the second-order feature mapping. Note that the illustrations of the 629 second-order feature weights were omitted.

selection procedure properly. The prediction of the  $MTTR$  was not carried out since no suitable model had been identified. The predictive models were learned with the full training data-set and their predictions were evaluated with the test data-set.<sup>14</sup> Due to the splitting of the training and test data by the age of the systems this simulated a prediction scenario.

<sup>14</sup> Hyperparameters were optimized by cross-validation over a parameter grid.





**Fig. 5.** (a): Predictions of the  $\log(MTBF)$  with the final models for the test data-set. The orange line depicts the mean and the orange shaded area the 95% confidence intervals. The blue dots mark the actual observed field-reliabilities. Note that the different converter types were ordered by the mean of the predictive distribution. (b): Estimated feature weights for the predictive models.

The test errors  $Err_{test}$  in Table 3e were of the same order as the cross-validation errors  $Err_{CV}$  of the reference configuration (3a). We concluded that the learned models generalize to newer power converters. The feature weights (5b) and the predictions (5a) were consistent with our expectations and demonstrate that we could predict the  $MTBF$  accurately.

**Discussion.** One of the major insights created by applying the methods to the use-case is that the field-reliabilities are strongly dependent on the quantity of converters per converter type. This fact can lead to an increased reliability for future systems. However, explanations for this dependence are plentiful and a more detailed analysis will have to be carried out.

The method is capable of learning more detailed statistical models for the whole lifecycle of systems. This requires to collect more reliability indicators than were available in this work. However, the purpose of this work was to illustrate that even with very coarse high-level data a good predictive model can be trained. The selected Bayesian algorithms which learn sparse parametric models were especially fit for this purpose. It has to be pointed out that the approach is empirical and that causal relationships have to be identified or confirmed by further studies or expert judgment.

## 5 Conclusion and Outlook

An approach was presented to predict the field-reliability of complex electronic systems at an early development stage based on a statistical lifecycle model learned from data collected for similar operational systems. It was demonstrated that the field-reliability can be predicted accurately based on very few reliability indicators. Compared to existing methods this implies a reduced data collection effort and an integrated quantification of predictive uncertainty based on

the granularity of the available information and the implicit randomness of the investigated processes. The results of such a study uncover reliability relevant factors which lead to improved system designs at very early stages of design.

Sparse Bayesian Regression methods are the key to efficiently learn accurate models. The confidence in field-reliability predictions is automatically quantified with respect to the available data and the randomness inherent in the problem. Future research can focus on more detailed and potentially incomplete data-sets. Based on that, further relevant processes for the field-reliability of systems may be uncovered.

## References

1. Barnard, R.: What is wrong with reliability engineering? In: INCOSE International Symposium, vol. 18, pp. 357–365. Wiley Online Library (2008)
2. Bishop, C.M.: Pattern recognition and machine learning (Information Science and Statistics). Springer, New York (2006)
3. Blondel, M., et al.: Scikit-learn user guide (2018). [http://scikit-learn.org/stable/user\\_guide.html](http://scikit-learn.org/stable/user_guide.html)
4. Cawley, G.C., Talbot, N.L.: On over-fitting in model selection and subsequent selection bias in performance evaluation. *J. Mach. Learn. Res.* **11**, 2079–2107 (2010)
5. Denson, W.: The history of reliability prediction. *IEEE Trans. Reliab.* **47**(3), 321–328 (1998)
6. Elerath, J.G., Pecht, M.: IEEE 1413: a standard for reliability predictions. *IEEE Trans. Reliab.* **61**(1), 125–129 (2012)
7. Foucher, B., Boullie, J., Meslet, B., Das, D.: A review of reliability prediction methods for electronic devices. *Microelectron. Reliab.* **42**(8), 1155–1162 (2002)
8. Gauch, H.G.: *Scientific Method in Practice*. Cambridge University Press, Cambridge (2003)
9. Gullo, L.: In-service reliability assessment and top-down approach provides alternative reliability prediction method. In: *Proceedings of Reliability and Maintainability Symposium, Annual*, pp. 365–377. IEEE (1999)
10. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning*. Springer, New York (2001)
11. Johnson, B.G., Gullo, L.: Improvements in reliability assessment and prediction methodology. In: *Proceedings of Reliability and Maintainability Symposium, Annual*, pp. 181–187. IEEE (2000)
12. Jones, J., Hayes, J.: A comparison of electronic-reliability prediction models. *IEEE Trans. Reliab.* **48**(2), 127–134 (1999)
13. Kapur, K.C., Pecht, M.: *Reliability Engineering*. Wiley, Hoboken (2014)
14. Leonard, C.T., Pecht, M.: How failure prediction methodology affects electronic equipment design. *Qual. Reliab. Eng. Int.* **6**(4), 243–249 (1990)
15. MacKay, D.J.: Bayesian interpolation. *Neural Comput.* **4**(3), 415–447 (1992)
16. Miller, R., Green, J., Herrmann, D., Heer, D.: Assess your program for probability of success using the reliability scorecard tool. In: *Annual Symposium-RAMS on Reliability and Maintainability*, pp. 641–646. IEEE (2004)
17. O’Connor, P., Kleyner, A.: *Practical Reliability Engineering*. Wiley, Chichester (2012)
18. Pandian, G.P., Diganta, D., Chuan, L., Enrico, Z., Pecht, M.: A critique of reliability prediction techniques for avionics applications. *Chin. J. Aeronaut.* (2017)

19. Pecht, M.G., Das, D., Ramakrishnan, A.: The IEEE standards on reliability program and reliability prediction methods for electronic equipment. *Microelectron. Reliab.* **42**(9–11), 1259–1266 (2002)
20. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011)
21. Williams, C.K., Rasmussen, C.E.: *Gaussian processes for machine learning*. The MIT Press, Massachusetts (2006)
22. Womack, J.P., Womack, J.P., Jones, D.T., Roos, D.: *Machine that Changed the World*. Simon and Schuster, New York (1990)
23. Zio, E.: Reliability engineering: old problems and new challenges. *Reliab. Eng. Syst. Saf.* **94**(2), 125–141 (2009)



# Building a Knowledge Based Summarization System for Text Data Mining

Andrey Timofeyev<sup>(✉)</sup> and Ben Choi

Computer Science, Louisiana Tech University, Ruston, USA  
andtimo@latech.edu, pro@benchoi.org

**Abstract.** This paper provides details on building a knowledge based automatic summarization system for mining text data. The knowledge based system mines text data on documents and webpages to create abstractive summaries by generalizing new concepts, deriving main topics, and creating new sentences. The knowledge based system makes use of the domain knowledge provided by Cyc development platform that consists of the world's largest knowledge base and one of the most powerful inference engines. The system extracts syntactic structures and semantic features by employing natural language processing techniques and Cyc knowledge base and reasoning engine. The system creates a summary of the given documents in three stages: knowledge acquisition, knowledge discovery, and knowledge representation for human readers. The knowledge acquisition derives syntactic structure of each sentence in the documents and maps their words and their syntactic relationships into Cyc knowledge base. The knowledge discovery abstracts novel concepts and derives main topics of the documents by exploring the ontology of the mapped concepts and by clustering the concepts. The knowledge representation creates new English sentences to summarize the documents. This system has been implemented and integrated with Cyc knowledge based system. The implementation encodes a process consisting seven stages: syntactic analysis, mapping words to Cyc, concept propagation, concept weights and relations accumulation, topic derivation, subject identification, and new sentence generation. The implementation has been tested on various documents and webpages. The test performance data suggests that such a system could benefit from running on parallel and distributed computing platforms. The test results showed that the system is capable of creating new sentences that include abstracted concepts not explicitly mentioned in the original documents and that contain information synthesized from different parts of the documents to compose a summary.

**Keywords:** Data mining · Text summarization · Artificial intelligence  
Knowledge extraction · Knowledge-based systems

## 1 Introduction

In this paper, we describe the implementation details of the automatic summarization system reported in [1]. The system mines text data on documents and webpages and uses knowledge base and inference engine to produce an abstractive summary. It generates summaries by composing new sentences based on the semantics derived from the text. The system combines syntactic structures and semantic features to provide

summaries that contains information synthesized from various parts of the document. It is built on Cyc development platform that consists of the world's largest knowledge ontology and one of the most powerful inference engines that allow information comprehension and generalization [2]. In addition, the Cyc knowledge ontology provides the domain knowledge for the subject matter discussed in the documents.

Abstractive document summarization is a task that is still considered complex for a human and especially for a machine. When human experts perform document summarization they tend to use their domain expertise about subject matter to merge information from various parts of the document and synthesize novel information, which was not explicitly mentioned in the text [3]. Our proposed system aims to follow similar approach. It generalizes new abstract concepts based on the knowledge derived from the text. It automatically detects main topics described in the text. Moreover, it composes new English sentences for some of the most significant concepts. The created sentences form an abstractive summary, combining concepts from different parts of the input text.

Our text data mining system is domain independent and unsupervised, being limited only by the common sense ontology provided by the Cyc development platform. The system conducts summarization process in three steps: knowledge acquisition, knowledge discovery, and knowledge representation.

The knowledge acquisition step derives syntactic structure of each sentence of the input document and maps words and their relations into Cyc knowledge base. Next, the knowledge discovery step generalizes concepts upward in the Cyc ontology and detects main topics covered in the text. Finally, the knowledge representation step composes new sentences for some of the most significant concepts defined in main topics. The syntactic structure of the newly created sentences follows an enhanced subject-predicate-object model, where adjective and adverb modifiers are used to produce more complex and informative sentences.

The system was implemented as a pipelined and modular data mining framework. Such system design allows comprehensible data flow, convenient maintenance and implementation of additional functionality as needed. The system was tested on various documents and webpages. The test results show that the system is capable of identifying key concepts and discovering main topics comprised in the original text, generalizing new concept not explicitly mentioned in the text and creating new sentences that contain information synthesized from various parts of the text. The newly created sentences have complex syntactic structures that enhance subject-predicate-object triplets with adjective and adverb modifiers. For example, the sentence "Colored grapefruit being sweet edible fruit" was automatically generated by the system analyzing encyclopedia articles describing grapefruits. Here, the subject concept "grapefruit" is modified by the adjective concept "colored" that was not explicitly mentioned in the text and the object concept "edible fruit" is modified by the adjective concept "sweet". The modifiers are chosen based on the weight of the syntactic relation.

The rest of the paper is organized as follows. Section 2 outlines related work undertaken in automatic text summarization area. Section 3 gives a brief overview of the summarization process steps performed by the system. Section 4 covers system implementation details. Section 5 provides thorough description of the system modules. Section 6 presents testing results. Section 7 discusses conclusions and directions of future work.

## 2 Related Work

Automatic text summarization seeks to compose a concise and coherent version of the original text preserving the most important information. Computational community has studied automatic text summarization problem since late 1950s [4]. Studies in this area are generally divided into two main approaches – extractive and abstractive. Extractive text summarization aims to select the most important sentences from original text to form a summary. Such methods vary by different intermediate representations of the candidate sentences and different sentence scoring schemes [5]. Summaries created by extractive approach are highly relevant to the original text, but do not convey any new information. Most prominent methods in extractive text summarization use term frequency versus inverse document frequency (TF-IDF) metric [6, 7] and lexical chains for sentence representation [8, 9]. Statistical methods based on Latent Semantic Analysis (LSA), Bayesian topic modelling, Hidden Markov Model (HMM) and Conditional random field (CRF) derive underlying topics and use them as features for sentence selection [10, 11]. Despite significant advancements in the extractive text summarization, such approaches are not capable of semantic understanding and limited to the shallow knowledge contained in the text.

In contrast, abstractive text summarization aims to incorporate the meaning of the words and phrases and generalize knowledge not explicitly mentioned in the original text to form a summary. Phrase selection and merging methods in abstractive summarization aim to solve the problem of combining information from multiple sentences. Such methods construct clusters of phrases and then merge only informative ones to form summary sentences [12]. Graph transformation approaches convert original text into a form of semantic graph representation and then combine or reduce such representation with an aim of creating an abstractive summary [13, 14]. Summaries constructed by described methods consist of sentences not used in the original text, combining information from different parts, but such sentences do not convey new knowledge.

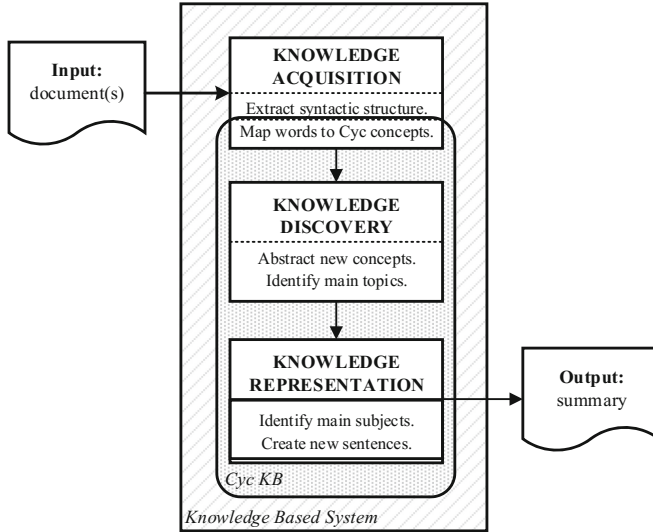
Several approaches attempt to incorporate semantic knowledge base into automatic text summarization by using WordNet lexical database [8, 15, 16]. Major drawback of WordNet system is the lack of domain-specific and common sense knowledge. Unlike Cyc, WordNet does not have reasoning engine and natural language generation capabilities.

Recent rapid development of deep learning contributes to the automatic text summarization, improving state-of-the-art performance. Deep learning methods applied to both extractive [17] and abstractive [18] summarization show promising results, but such approaches require vast amount of training data and powerful computational resources.

Our system is similar to the one proposed in [19]. In this work, the structure of created sentences has simple subject-predicate-object pattern and new sentences are only created for clusters of compatible sentences found in the original text.

### 3 Overview of the Summarization Process

Our system conducts summarization process in three steps: knowledge acquisition, knowledge discovery, and knowledge representation. Summarization process workflow is illustrated in Fig. 1.

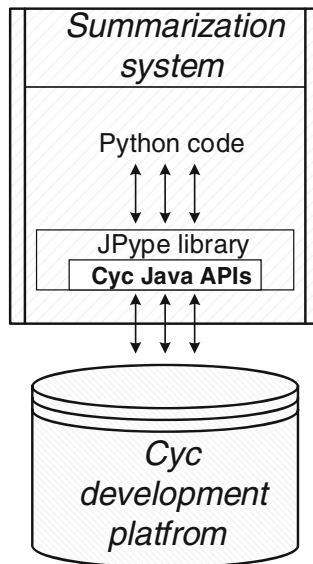


**Fig. 1.** System's workflow diagram.

The knowledge acquisition step consists of two parts. It first takes text documents as an input and derives their syntactic structures. Then it maps each word in the text to its corresponding Cyc concept and assigns word's weight and derived syntactic relations to that concept. The knowledge discovery step is responsible for abstracting new concepts that are not explicitly mentioned in the text. During this process, system derives ancestor concept for each mapped Cyc concept, assigns ancestor-descendant relation and adds scaled descendant concept weight and descendant concept associations to the ancestor concept. In addition, the system identifies main topics comprised in the text by clustering mapped Cyc concepts. During the knowledge representation step, the system first identifies most informative subject concepts in each of the discovered main topics and then composes English sentences for each identified subject. This process ensures that the summary sentences are composed using information synthesized from different parts of the text while preserving coherence to the main topics.

## 4 Details of the System's Implementation

We chose Python as the implementation language to develop our system because of the advanced Natural Language Processing tools and libraries it supplies. Our system uses Cyc knowledge base and inference engine as a backbone for the semantic analysis. Cyc development platform supports communications with the knowledge base and utilization of the inference engine through the application programming interfaces (APIs) implemented in Java. We utilize Java-Python wrapper supported by JPytype library to allow our system using Cyc Java API packages. JPytype library is essentially an interface at a basic level of virtual machines [20]. It requires starting Java Virtual Machine with a path to the appropriate jar files before Java methods and classes can be accessible within Python code. Communication between our system and Cyc development platform is illustrated in Fig. 2. To the best of our knowledge, our developed system is the first Python-based system that allows communication with Cyc development platform.



**Fig. 2.** Communication between summarization system and Cyc development platform.

We have designed our system as a modular and pipelined data mining framework. Modularity provides the ability to conveniently maintain parts of the system and to add new functionality as needed. Pipelined design allows comprehensible data flow between different modules.

The system consists of seven modules:

1. Syntactic analysis;
2. Mapping words to Cyc KB;
3. Concepts propagation;



4. Concepts' weights and relations accumulation;
5. Topics derivation;
6. Subjects identification;
7. New sentences generation.

Modules 1 and 2 together constitute the knowledge acquisition step of the summarization process. Modules 3, 4 and 5 together make up the knowledge discovery step of the summarization process. Modules 6 and 7 together form knowledge representation step of the summarization process. System modules are illustrated in Fig. 3.

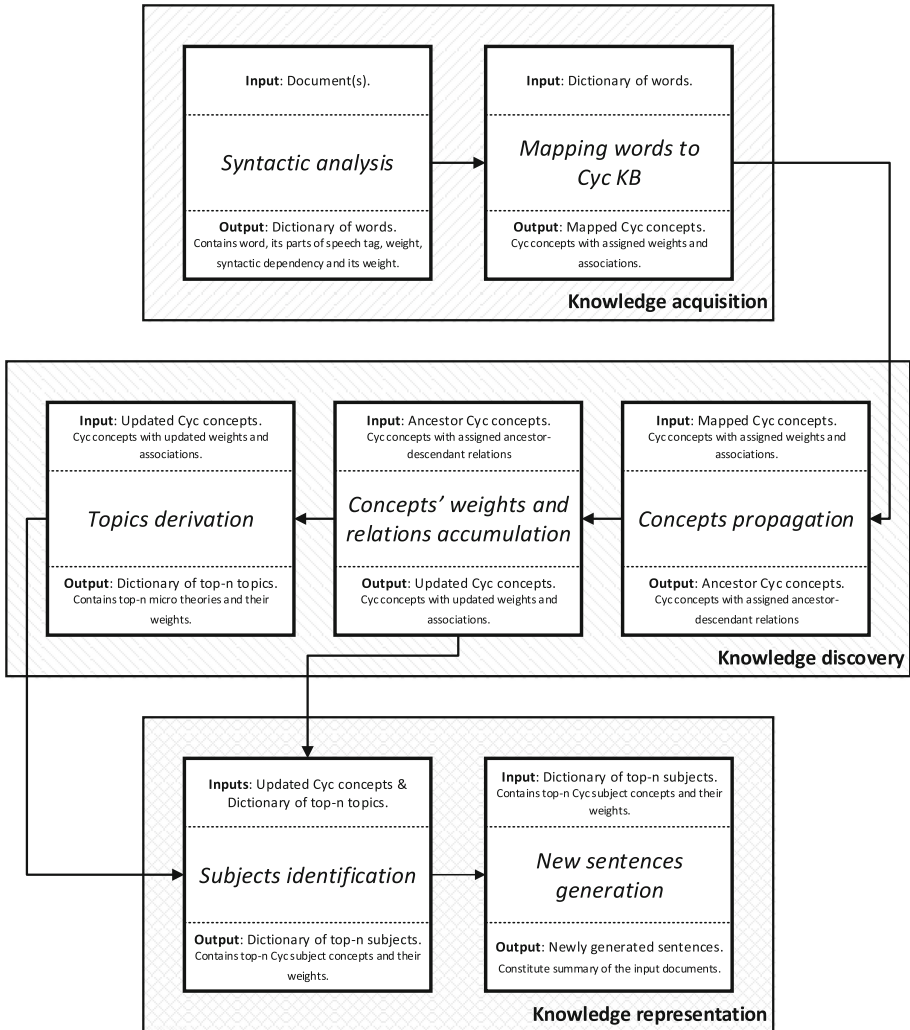
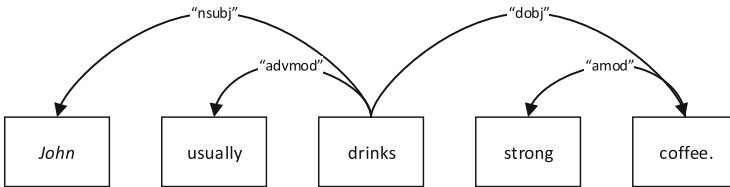


Fig. 3. Modular design of the system.

## 5 Description of the System’s Modules

### 5.1 “Syntactic Analysis” Module

The first module in the system is the “Syntactic analysis” module. The role of this module is essentially a data preprocessing. The module takes documents as an input and transforms them into syntactic representations. It first performs text normalization by lemmatizing each word in each sentence. Then it derives part of speech tags, parses syntactic dependencies and counts word’s weights. The syntactic dependencies are recorded in the following format: (“word” “type” “head”), where “word” is the dependent element, “type” is the type of the dependency, and “head” is the leading element. For example, applying syntactic parser on the following sentence: “John usually drinks strong coffee” produces the following syntactic dependencies between words: (“John” “nsubj” “drinks”), (“coffee” “dobj” “drinks”), (“usually” “advmod” “drinks”), (“strong” “amod” “coffee”). Syntactic dependencies of the example sentence are illustrated in Fig. 4.



**Fig. 4.** Illustration of the syntactic dependencies of a sample sentence.

The “Syntactic analysis” module is implemented using SpaCy – Python library for advanced natural language processing. SpaCy library is the fastest in the world with the accuracy within one percent of the current state of the art systems for part of speech tagging and syntactic dependencies analysis [21]. The “Syntactic analysis” module operates outside of the Cyc development platform. The output of the module is a dictionary that contains words, their part of speech tags, weights and syntactic dependencies. This dictionary serves as an input for “Mapping words to Cyc KB” module.

### 5.2 “Mapping Words to Cyc KB” Module

The “Mapping words to Cyc KB” module takes dictionary of words, derived by the “Syntactic analysis” module, as an input. This module finds an appropriate Cyc concept for each word in the dictionary, and assigns word’s weight and syntactic dependency associations to Cyc concept. It starts by mapping each word to the corresponding Cyc concept (1). Next, it assigns word’s weight to Cyc concept (2). Then it maps the syntactic dependency head to the appropriate Cyc concept. Finally, it assigns the syntactic dependency association and its weight to the Cyc concept (3). Table 1 provides the description of Cyc commands used to implement each step.

**Table 1.** Description of Cyc commands used by “Mapping words to Cyc KB” module.

Step	Cyc command	Description
1	(#\$and (\$denotation ?Word ?POS ?Num ?Concept) (\$wordForms ?Word ?WordForm “word”) (\$genls ?POS ?POSTag))	Command uses built-in “#\$denotation” Cyc predicate to relate a “word”, its part of speech tag (?POS), and a sense number (?Num) to concept (?Concept). It also uses “#\$wordForms” and “#\$genls” predicates to accommodate for all variations of word’s lexical forms
2	(#\$conceptWeight ?Concept ?Weight)	Command uses user-defined “#\$conceptWeight” Cyc predicate that assigns the weight (?Weight) to the concept (?Concept)
3	(#\$conceptAssociation ?Concept ?Type ?HeadConcept ?Weight)	Command uses user-defined “#\$conceptAssociation” Cyc predicate that assigns a specific type (?Type) of a syntactic dependency association, the leading element (?HeadConcept) and the weight (?Weight) to the concept (?Concept)

This module communicates with Cyc development platform and updates weight and syntactic dependency relations of Cyc concepts. The output of the module are mapped Cyc concepts with assigned weights and syntactic dependency relations. The mapped Cyc concepts serve as an input for “Concepts propagation” module. “Syntactic analysis” and “Mapping words to Cyc KB” modules together constitute the knowledge acquisition step of the summarization process.

### 5.3 “Concepts Propagation” Module

The “Concepts propagation” module takes Cyc concepts, mapped by “Mapping words to Cyc KB” module, as an input and finds their closest ancestor concepts. This module performs generalization and abstraction of new concepts that have not been mentioned in the text explicitly. It starts by querying Cyc knowledge base for all the concepts that have assigned weight (1). Then it finds an ancestor concept for each concept derived by the query (2). Next, it records the number of ancestor’s descendant concepts and their weight (3). Finally, it assigns ancestor-descendant relation between ancestor and descendant concepts (4). Table 2 provides the description of Cyc commands used to implement each step.

This module communicates with Cyc development platform to derive all mapped Cyc concepts, find closest ancestor concepts and update ancestor concepts’ relations. The output of the module are ancestor Cyc concepts with assigned descendant concepts’ weights and counts and ancestor-descendant relations. The ancestor Cyc concepts are used by “Concepts’ weights and relations accumulation” module.

**Table 2.** Description of Cyc commands used by “Concepts propagation” module.

Step	Cyc command	Description
1	(#\$conceptWeight ?Concept ?Weight)	Command uses user-defined “#\$conceptWeight” Cyc predicate to retrieve concepts (?Concept) that have assigned weights (?Weight)
2	(#\$min-genls ?Concept)	Command uses built-in “min-genls” Cyc predicate to retrieve the closest ancestor concept for the given concept (?Concept)
3	(#\$conceptDescendants ?Concept ?Weight ?Count)	Command uses user-defined “#\$conceptDescendants” Cyc predicate to record the number of descendants (?Count) and their weight (?Weight) to the ancestor concept (?Concept)
4	(#\$conceptAncestorOf ?Concept ?Descendant)	Command uses user-defined “#\$conceptAncestorOf” predicate to assign ancestor-descendant relation between the ancestor concept (?Concept) and the descendant concept (?Descendant)

#### 5.4 “Concepts’ Weights and Relations Accumulation” Module

The “Concepts’ weights and relations accumulation” module takes ancestor Cyc concepts as an input and adds descendants’ accumulated weight and relations to ancestor concepts if the calculated descendant-ratio is higher than the threshold. The descendant-ratio is the number of mapped descendant concepts divided by the number of all descendant concepts of an ancestor concept. This module starts by querying Cyc knowledge base for all ancestor concepts (1). Then it calculates the descendant ratio for each ancestor concept (2.1, 2.2). Next, it adds propagated descendants’ weight (3) and descendants’ associations with their propagated weights (4) to ancestor concepts if the descendant-ratio is higher than the defined threshold. Table 3 provides the description of Cyc commands used to implement each step.

This module communicates with Cyc development platform to derive all ancestor Cyc concepts, find the number of ancestor’s mapped descendants, find the number of all ancestor’s descendants and update ancestor’s weight and relations. The output of the module are the Cyc concepts with updated weights and syntactic dependency associations. Updated Cyc concepts are used by the “Topics derivation” and the “Subjects identification” modules.

#### 5.5 “Topics Derivation” Module

The “Topics derivation” module takes updated Cyc concepts as an input and derives defining micro theory for each concept. Micro theories with the highest weights represent the main topics of the document. This module first derives defining micro theory for each Cyc concept that have assigned weight (1). Then it counts the weights of derived micro theories based on their frequencies and picks up top-n with the highest weights. Table 4 provides the description of Cyc command used to implement defining micro theory derivation.

**Table 3.** Description of Cyc commands used by “Concepts’ weights and relations accumulation” module.

Step	Cyc command	Description
1	(#\$conceptDescendants ?Concept ?Weight ?Count)	Command uses user-defined “#\$conceptDescendants” Cyc predicate to retrieve all concepts (?Concept) that have descendants
2.1	(#\$conceptAncestorOf ?AncConcept ?MappedDesc)	Command uses user-defined “#\$conceptAncestorOf” predicate to retrieve mapped descendant concepts (?MappedDesc) of the given ancestor concept (?AncConcept)
2.2	(#\$genls ?AncConcept ?DescConcept)	Command uses built-in “#\$genls” Cyc predicate to retrieve all descendant concepts (?DescConcept) of the given ancestor concept (?AncConcept)
3	(#\$conceptWeight ?AncConcept ?DescWeight)	Command uses user-defined “#\$conceptWeight” Cyc predicate to assigns the descendant concepts’ propagated weight (?DescWeight) to the ancestor concept
4	(and (#\$conceptAncestorOf ?AncConcept ?DescConcept) (#\$conceptAssociation ?DescConcept ?Type ?HeadConcept ?Weight))	Command uses user-defined “#\$conceptAncestorOf” and “#\$conceptAssociation” Cyc predicates to assign descendant’s association (?DescConcept) and its propagated weight (?Weight) to the ancestor concept (?AncConcept)

**Table 4.** Description of Cyc command used by “Topics derivation” module

Step	Cyc command	Description
1	(#\$and (#\$conceptWeight ?Concept ?Weight) (#\$definingMt ?Concept ?MicroTheory))	Command uses user-defined “#\$conceptWeight” Cyc predicate and built-in “#\$definingMt” Cyc predicate to derive defining micro theory (?MicroTheory) for each concept (?Concept) that have assigned weight (?Weight)

This module communicates with Cyc development platform to derive defining micro theory for each mapped Cyc concept. Calculation of the derived micro theories’ weights is handled outside of the Cyc development platform. The output of the module is the micro theories dictionary that contains top-n micro theories with highest weights. This dictionary serves as an input for the “Subjects identification” module. The “Concepts propagation”, the “Concepts’ weights and relations accumulation” and the “Topics derivation” modules together constitute knowledge discovery step of the summarization process.

## 5.6 “Subjects Identification” Module

The “Subjects identification” module uses updated Cyc concepts and the dictionary of top-n micro theories as an input to derive most informative subject concepts based on a subjectivity rank. Subjectivity ranks is the product of the concept’s weight and the concept’s subjectivity ratio. Subjectivity ratio is the number of concept’s syntactic dependency associations labelled as “subject” relations divided by the total number of concept’s syntactic dependency associations. Subjectivity rank allows identifying concepts with the strongest subject roles in the documents. The module start by querying Cyc knowledge base for all mapped Cyc concepts for each micro theory in top-n micro theories dictionary (1). Then it calculates subjectivity ratio and subjectivity rank for each derived Cyc concept (2.1, 2.2). Finally, it picks top-n subject concepts with the highest subjectivity rank. Table 5 provides the description of Cyc commands used to implement each step.

**Table 5.** Description of Cyc commands used by “Subjects identification” module.

Step	Cyc command	Description
1	(#\$and (\$definingMt ?Concept ?MicroTheory) (\$conceptWeight ?Concept ?Weight))	Command uses built-in “#\$definingMt” Cyc predicate and user-defined “conceptWeight” Cyc predicate to derive concepts (?Concept) that have assigned weight (?Weight) for each micro theory (?MicroTheory) in micro theories dictionary
2.1	(#\$conceptAssociation ?Concept “nsubj” ?HeadConcept ?Weight)	Command uses user-defined “#\$conceptAssociation” Cyc predicate with “nsubj” parameter to derive the concept’s (?Concept) syntactic dependency associations labelled as “subject” relations
2.2	(#\$conceptAssociation ?Concept ?Type ?HeadConcept ?Weight)	Command uses user-defined “#\$conceptAssociation” Cyc predicate with no parameter specified (?Type) to derive all concept’s (?Concept) syntactic dependency associations

This module communicates with Cyc development platform to derive mapped Cyc concepts for each defining micro theory in the input dictionary and to find the number of the concept’s syntactic dependency associations labelled as “subject” relation and the number of all syntactic dependency associations of the concept. Calculations of the subjectivity ratio and the subjectivity rank are handled outside of the Cyc development platform. The output of the module is the dictionary that contains top-n subjects with the highest subjectivity rank. This dictionary serves as an input for the “New sentence generation” module.

### 5.7 “New Sentences Generation” Module

The “New sentences generation” module takes the dictionary of top-n most informative subjects as an input and produces new sentences for each of the subject to form a summary of the input documents. The module starts by deriving a natural language representation of each subject Cyc concept in the dictionary (1). Then it picks the adjective Cyc concept modifier with the highest subject-adjective syntactic dependency association weight (2) and derives its natural language representation. Next, it picks top-n predicate Cyc concepts with the highest subject-predicate syntactic dependency association weights (3) and derives their natural language representations. Then it picks the adverb Cyc concept modifier with the highest predicate-adverb syntactic dependency association weight (4) and derives its natural language representation. Next, it picks top-n object Cyc concepts with the highest product of subject-object and predicate-object syntactic dependency association weights (5.1, 5.2) and derives their

**Table 6.** Description of Cyc commands used by “New sentence generation” module.

Step	Cyc command	Description
1	(#\$generate-phrase ?Concept)	Command uses built-in “#\$generate-phrase” Cyc predicate to retrieve corresponding natural language representation for a Cyc concept (?Concept)
2	(#\$conceptAssociation ?Concept “amod” ?HeadConcept ?Weight)	Command uses user-defined “#\$conceptAssociation” Cyc predicate with “amod” parameter to derive Cyc concept (?Concept) associations labelled as adjective modifier syntactic dependency relation
3	(#\$conceptAssociation ?Concept “pred” ?HeadConcept ?Weight)	Command uses user-defined “#\$conceptAssociation” Cyc predicate with “pred” parameter to derive Cyc concept (?Concept) associations labelled as predicate syntactic dependency relation
4	(#\$conceptAssociation ?Concept “advmod” ?HeadConcept ?Weight)	Command uses user-defined “#\$conceptAssociation” Cyc predicate with “advmod” parameter to derive Cyc concept (?Concept) associations labelled as adverb modifier syntactic dependency relation
5.1	(#\$conceptAssociation ?Concept “obj” ?HeadConcept ?Weight)	Command uses user-defined “#\$conceptAssociation” Cyc predicate with “obj” parameter to derive Cyc concept (?Concept) associations labelled as object syntactic dependency relation
5.2	(#\$conceptAssociation ?Concept “subj-obj” ?HeadConcept ?Weight)	Command uses user-defined “#\$conceptAssociation” Cyc predicate with “subj-obj” parameter to derive Cyc concept (?Concept) associations labelled as subject-object syntactic dependency relation

natural language representations. Then, it picks the adjective Cyc concept modifier with the highest object-adjective syntactic dependency association weight and derives its natural language representation. Finally, it composes the new sentence using subject, subject-adjective, predicate, predicate-adverb, object and object-adjective natural language representations. Table 6 provides the description of Cyc commands used to implement each step.

This module communicates with Cyc development platform to derive appropriate Cyc concepts for each sentence element based on the weights of their syntactic dependency associations and derive their natural language representation. New sentences are composed outside of the Cyc development platform and serve as an output for the module and the whole summarization system. The “Subjects identification” and the “New sentences generation” modules together constitute the knowledge representation step of the summarization process.

## 6 Testing and Results

We have tested our system on various encyclopedia articles describing concepts from different domain. First, we conducted an experiment using multiple articles about grapefruits. In this experiment, we increased the number of analyzed articles on each run of the system, starting with a single article. Figure 5 illustrates new sentences created by the system. These results show the progression of sentence structure from simple subject-predicate-object triplet to more complex structure enhanced by the adjective and adverb modifiers when more articles were processed by the system.

<p>“Grapefruit being fruit.” (a)</p> <p>“Grapefruit being colored edible fruit.” (b)</p> <p>“Colored grapefruit being sweet edible fruit.” (c)</p>
----------------------------------------------------------------------------------------------------------------------------------------------------

**Fig. 5.** Test results of new sentences created for multiple articles about grapefruit; (a) – single article, (b) – two articles, (c) – three articles.

Next, we applied our system on five encyclopedia articles describing different types of felines, including cats, tigers, cougars, jaguars and lions. Figure 6 shows main topics and concepts extracted from the text and newly created sentences.

These results show that the system is able to abstract new concepts and create new sentences that contain information synthesized from different parts of the documents. Concepts like “canis”, “mammal meat” and “felis” were derived by the generalization process and were not explicitly mentioned in the original documents. Our system yields better results compared to the reported in [19]. New sentences created by the system have structure that is more complex and contain information fused from various parts of the text. More testing results are reported in [1].



<p><u>Topics (micro theories):</u></p> <ul style="list-style-type: none"> <li>• #<math>\\$</math>BiologyMt</li> <li>• #<math>\\$</math>BiologyVocabularyMt</li> <li>• #<math>\\$</math>HumanSocialLifeMt</li> </ul> <p><u>Concepts:</u></p> <ul style="list-style-type: none"> <li>• #<math>\\$</math>Cat</li> <li>• #<math>\\$</math>DomesticCat</li> <li>• #<math>\\$</math>FelisGenus</li> <li>• #<math>\\$</math>FelidaeFamily</li> <li>• #<math>\\$</math>Animal</li> </ul>	<p><u>Sentences:</u></p> <p><i>"Cat usually being native animal."</i></p> <p><i>"Big felis usually being natural predatory animal."</i></p> <p><i>"Big felis usually being exotic animal."</i></p> <p><i>"Big felis often using killing method."</i></p> <p><i>"Big felis often using marking."</i></p> <p><i>"Male feline often killing prey."</i></p> <p><i>"Male feline living historical mountain range."</i></p>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Fig. 6.** Test results of new sentences, concepts and main topics for encyclopedia articles about felines.

## 7 Conclusions and Future Work

In this paper, we described an implementation of the knowledge based automatic summarization system that creates an abstractive summary of the text. This task is still challenging for machines, because in order to create such summary, the information from the input text has to be aggregated and synthesized, drawing knowledge that is more general. This is not feasible without using the semantics and having domain knowledge. To have such capabilities, our implemented system uses Cyc knowledge base and its reasoning engine. Utilizing semantic features and syntactic structure of the text shows great potential in creating abstractive summaries. We have implemented and tested our proposed system. The results show that the system is able to abstract new concepts not mentioned in the text, identify main topics and create new sentences using information from different parts of the text.

We outline several directions for the future improvements of the system. The first direction is to improve the domain knowledge representation, since the semantic knowledge and reasoning are only limited by Cyc knowledge base. Ideally, the system would be able to use the whole World Wide Web as a domain knowledge, but this possesses challenges like information inconsistency and sense disambiguation. The second direction is to improve the structure of the created sentences. We use subject-predicate-object triplets extended by adjective and adverb modifiers. Such structure can be improved by using more advanced syntactic representation of the sentence, e.g. graph representation. Finally, some of the created sentences are not conceptually connected to each other. Analyzing the relations between concepts on the document level will help in creating sentences that will be linked to each other conceptually.

## References

1. Timofeyev, A., Choi, B.: Knowledge based automatic summarization. In: Proceedings of the 9th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3 K 2017), pp. 350–356. SCITEPRESS (2017). <https://doi.org/10.5220/0006580303500356>
2. Cycorp: Cycorp Making Solutions Better. <http://www.cyc.com>
3. Cheung, J., Penn, G.: Towards robust abstractive multi-document summarization: a caseframe analysis of centrality and domain. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pp. 1233–1242. Association for Computational Linguistics (2013)
4. Luhn, H.: The automatic creation of literature abstracts. *IBM J. Res. Dev.* **2**, 159–165 (1958). <https://doi.org/10.1147/rd.22.0159>
5. Nenkova, A., McKeown, K.: A survey of text summarization techniques. In: Charu, A., Zhai, C. (ed.) *Mining Text Data*, pp. 43–76. Springer, Boston (2012). [https://doi.org/10.1007/978-1-4614-3223-4\\_3](https://doi.org/10.1007/978-1-4614-3223-4_3)
6. Hovy, E., Chin-Yew, L.: Automated text summarization and the SUMMARIST system. In: Proceedings of a workshop held at Baltimore, Maryland, October 13–15, 1998, pp. 197–214. Association for Computational Linguistics (1998). <https://doi.org/10.3115/1119089.1119121>
7. Radev, D., Jing, H., Styś, M., Tam, D.: Centroid-based summarization of multiple documents. *Inf. Process. Manag.* **40**, 919–938 (2004). <https://doi.org/10.3115/1117575.1117578>
8. Barzilay, R., Elhadad, M.: Using lexical chains for text summarization. *Adv. Autom. Text summ.* 111–121 (1999). <https://doi.org/10.7916/d85b09vz>
9. Ye, S., Chua, T., Kan, M., Qiu, L.: Document concept lattice for text understanding and summarization. *Inf. Process. Manag.* **43**, 1643–1662 (2007). <https://doi.org/10.1016/j.ipm.2007.03.010>
10. Gong, Y., Liu, X.: Generic text summarization using relevance measure and latent semantic analysis. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 19–25. ACM (2001). <https://doi.org/10.1145/383952.383955>
11. Shen, D., Sun, J., Li, H., Yang, Q., Chen, Z.: Document summarization using conditional random fields. In: Proceedings of International Joint Conference on Artificial Intelligence, pp. 2862–2867. IJCAI (2007)
12. Bing, L., Li, P., Liao, Y., Lam, W., Guo, W., Passonneau, R.: Abstractive multi-document summarization via phrase selection and merging. In: Proceedings of the ACL-IJCNLP, pp. 1587–1597. Association for Computational Linguistics (2015)
13. Ganesan, K., Zhai, C., Han, J.: Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In: Proceedings of the 23rd International Conference on Computational Linguistics, pp. 340–348. Association for Computational Linguistics (2010)
14. Moawad, I., Aref, M.: Semantic graph reduction approach for abstractive text summarization. In: Seventh International Conference Computer Engineering & Systems (ICCES), pp. 132–138. IEEE (2012). <https://doi.org/10.1109/iccес.2012.6408498>
15. Bellare, K., Das Sharma, A., Loiwal, N., Mehta, V., Ramakrishnan, G., Bhattacharyya, P.: Generic text summarization using WordNet. In: Language Resources and Evaluation Conference LREC, pp. 691–694 (2004)

16. Pal, A., Saha, D.: An approach to automatic text summarization using WordNet. In: IEEE International Advance Computing Conference (IACC), pp. 1169–1173. IEEE (2014). <https://doi.org/10.1109/iadcc.2014.6779492>
17. Nallapati, R., Zhai, F., Zhou, B.: SummaRuNNer: a recurrent neural network based sequence model for extractive summarization of documents. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI 2017), pp. 3075–3081. AAAI (2017)
18. Rush, A.M., Chopra, S., Wetson, J.: A neural attention model for abstractive sentence summarization. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing EMNLP, pp. 379–389 (2015). <https://doi.org/10.18653/v1/d15-1044>
19. Choi, B., Huang, X.: Creating new sentences to summarize documents. In: The 10th IASTED International Conference on Artificial Intelligence and Application (AIA 2010), pp. 458–463. IASTED (2010)
20. JPyype: Java to Python integration. <http://jpyype.sourceforge.net>
21. Honnibal, M., Johnson, M.: An improved non-monotonic transition system for dependency parsing. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing EMNLP, pp. 1373–1378 (2015). <https://doi.org/10.18653/v1/d15-1162>



# Spanish Twitter Data Used as a Source of Information About Consumer Food Choice

Luis G. Moreno-Sandoval<sup>1,2</sup>(✉), Carolina Sánchez-Barriga<sup>1,2</sup>,  
Katherine Espíndola Buitrago<sup>1,2</sup>, Alexandra Pomares-Quimbaya<sup>2</sup>,  
and Juan Carlos García<sup>2</sup>

<sup>1</sup> Colombian Center of Excellence and Appropriation on Big Data  
and Data Analytics (CAOBA), Bogotá, Colombia  
morenoluis@javeriana.edu.co

<sup>2</sup> Pontificia Universidad Javeriana, Bogotá, Colombia

**Abstract.** Food related consumer behavior is a topic of major interest to areas such as health and marketing. Social media offers a scenario in which people share information about preferences, interests and motivations about eating habits and food products that have not been explored as appropriate. In this work we present an algorithm to exploit the potential of Twitter as a data gathering platform to provide insight about behavior of consumers, by linking the food-related content, including emoji's expressed by Twitter users, to their demographic profile (age, gender, socioeconomic level). We further link this data to dietary choices expressed in different moments of their daily life. We found out that including Spanish Twitter data analysis, like the one presented in this work, into marketing researchers tools, could be very useful to advance in customer-centric strategies.

**Keywords:** Social networks · Consumer behavior · Twitter · Food analysis  
Social media

## 1 Introduction

### 1.1 Motivation

Twitter data have become a source of insights to study consumer behavior in different contexts and domains. Regarding food consumer behavior, the need for using Twitter raises in some limitations of common practices due to the strong influence of contextual variables and the subjective charge that consumers' responses can have when looking for socially desirable or over-rationalized answers [1].

However, Twitter offers an option to those limitations, since it becomes a natural consumption setting which provides access to consumer information spontaneously. According to Vidal et al., [1], Twitter is used to present daily information including consumption routines and comments; and, since eating and drinking are some of the most common human activities, tweets can be a data source for food-related consumer behavior insights.

According to Nielsen [2], Twitter is well positioned to study eating situations since it provides researchers the opportunity to retrieve spontaneous data, generated in real-life settings. In this sense, it is possible to collect data in any situation, considering that consumers are increasingly using smartphones to access social media [2].

## 1.2 The Potential of Twitter in Marketing Research

Evolution of digital trends such as social networks, mobile technologies, cloud computing or Big Data provide a huge source of information about consumer behavior, needs and requirements [3]. Therefore, companies can offer completely new services, participate interactively with customers and provide a completely different work environment, which is the reason why digital technology plays a critical role in consumer research strategies.

The proposal of Uhl et al., [4], argues that using a Customer Centralization strategy on organizations allows to position and to achieve economic success on organizations. From the perspective of the customer's life cycle, the Customer Centralization strategy focuses on four phases: information, supply, purchase and after-sales. In a transversal way, the customer experience goes through every phase. It is defined as a managerial philosophy that involves a complete alignment of the company with existing and potential relationships with its customers; it makes customer the focus of all commercial considerations. The central principle is to increase the value of the company and the customer itself through the systematic management of these relationships.

In order to improve company's customer knowledge and considering Twitter as an invaluable source of information about customer profile, interests and behaviors, this paper proposes an algorithm able to analyze food mention behavior in social networks from different points of view. The algorithm is able to identify the context in which a customer mentions food related words and characterizes the situation in which it was posted. In addition, it considers not only narrative texts, but also hashtags and user mention. The proposed algorithm demonstrates improvement on current approaches employed in food-related studies in social media.

This paper is organized as follows: In the Sect. 2, recent studies that use twitter information to perform analysis in specific areas of knowledge such as food-consumption, tobacco consumption and healthcare are presented. Section 3 describes the proposed algorithm. First, it describes the food extraction processes used for the construction of the knowledge base that supports the algorithm, and then explains the proposed algorithm for food detection. Section 4 presents the main results obtained in the case study and concludes with Sect. 5 presenting our main contributions and future work.

## 2 Related Works

Recent food-related studies have focused on problems, topics and consumer behavior research in public health. Vidal et al., [1] found that *“people tended to mainly tweet about eating situations when they enjoyed them, due to the foods they consumed, the place in which they were and/or the people they shared the meal with.”*

Abbar et al., [5] found that foods mentioned in daily tweets of users are predictive of the national obesity and diabetes statistics, showing how the calories tweeted are linked to user interests and demographic indicators, and that users sharing more friends are more likely to display a similar interest towards food. This work includes demographic indicators correlated with food-related information. The studies from Abbar et al. enriched data using a variety of sources, which allowed considering nutritional value of the foods mentioned in tweets, demographic characteristics of the users who tweet them, their interests, and the social network they belonged to.

In a recent study, Prier et al., [6] used LDA to find topics related to tobacco consumption, such as addiction recovery, other drug use, and anti-smoking campaigns. Finally, Dredze et al., [7] applied a Food Topic Aspect Model on tweets, to find out mentions of various aliments; The results suggest that chronic health behaviors, such as tobacco use, can be identified and measured, however, this does not apply to other short-term health events, such as outbreaks of disease. Also, it is found that the demographics of Twitter users can affect this type of studies, leaving the debate open.

Users of online social networks (OSN) reveal a lot about themselves; however, depending on their privacy concerns, they also choose not to share details that seem sensitive to them, reconfiguring access to their information in the OSN [11]. Many applications on Facebook that are well-known for being able to use them, request a lot of information from the user [12]. On the contrary, the proposal presented in this article is based exclusively on the publicly available information of users of social networks and, in that sense, does not violate any agreement on the use of data applicable in America and Europe. In addition, demographic data derived from OSN users, and employed for the food-consumption analysis, is the result of a previous project that demonstrates and validates the potential of twitter public publications to infer valuable information about its users [13, 14].

### 3 Consumer Food Choice Identification

In order to explore Twitter data, we used *bag of words* [8, 9, 13] as a method to understand the tweet content related to food consumption. This method uses an initial food knowledge base with 1128 words, generated by an automatic domain constructor [16]. In the first approach of the analysis, we found out that a large portion of the tweets that include food words are not referring to actual food consumption, this is one of the most important challenges on the algorithm. Most of them refer to popular sayings that include food words like:

*“amigo el ratón del queso (friend the mouse of the **cheese**)”.*

*“cuentas claras y el **chocolate** espeso (bills clear and **chocolate** thick)”.*

*“sartén por el **mango** (taking the frying pan by the handle)”.*

*“al **pan**, **pan** y al **vino**, **vino** (the **bread** is **bread** and the **wine** is **wine**)”.*

Some tweets had another type of expressions widely used in other contexts, that includes food words; for example, the word **jam** in the Colombian political context is associated with corruption issues and is widely used in social networks, for example:

“Desastroso es un gobierno lleno de *mermelada* y clientelismo (a government full of *jam* and clientelism is disastrous)”.

Additionally, this tweet understanding also shows that many users refer to specific products associated with popular brands, without using the food word, such as “*Pony Malta (soda)*”, “*Coca – Cola (soda)*”, “*Galletas Oreo (cookies)*”. In the same way, other users use hashtags like “*#almuerzo (#lunch)* and *#aguacate (#avocado)*” or mentions (or usernames) such as “*@baileysoriginal* and *@BogotaBeerCo*” to make a reference to products or places of consumption. Finally, we also concluded that users refer to food consumption with emojis as shown on Fig. 1.



Fig. 1. Emoji use referring food consumption.

Taking into account these insights, we had two main challenges: create a knowledge base that can be used to analyze food mention behavior in narrative texts from social networks and propose a **new food mention identification algorithm** that recognizes the context of food-related words using different aspects of the publication to disambiguate it, like hashtags, user mentions, emojis and food n-grams with  $n > 1$  as well as non-food n-grams.

### 3.1 Knowledge Base Generation

In this section we present the result of the knowledge base generation, which is composed of 11 lists, namely:

- *Emoji list*: this list was constructed using as primary source the 11th version of the Unicode emoji characters and sequences from Unicode standard. This list has a total of 2620 emojis.
- *Food emoji list*: Felbo et al., [10] used the emoji prediction to find topics related to feeling in different domains. In our proposal we use a subset of the 95 emojis in the *emoji list*, named as the *food emoji list*. An extract of this list is presented in Fig. 2, where both the emoji and its meaning in English and Spanish, can be seen.
- *What list*: to construct this list, the initial auto generated food list (see Sect. 3) was manually reviewed, generating a new list which considers only unigrams, used on the food consumption context. It contains 776 words including their stem.
- *Where list*: this list allows identifying places, locations or spatial situations associated with food consumption. It contains 128 words.
- *Who list*: this list enables to identify people with whom food is shared using relationships and professions. It contains 112 words.
- *When list*: this list has moments, occasions and temporary situations in which people consume food. It contains 27 words.

Emoji	Food	Spanish word
	Avocado	Aguacate
	Bread	Pan
	Chicken	Pollo
	Burger	Hamburguesa
	Ice cream	Helado

**Fig. 2.** Food emoji list sample.

- Food stop word list: it contains 178 popular sayings or expressions (non-food n-grams) frequently used on Twitter with food words, which do not correspond to the food consumption context. This list aims to be a filter to discard tweets.
- Food list: this list is composed of 441 n-grams with  $n > 1$ .
- Food user mention list: this list details 95 Twitter usernames associated with products, brands and places.
- Food entity list: food brand list with 812 elements.
- Food hashtag list: includes 450 hashtags related to food, that are typically used to refer to specific products or places.

According to the beforehand described lists, in the following section we present the proposed algorithm to identify food mentions in Twitter text.

### 3.2 Modelling

The proposed algorithm focuses on determining whether the tweet can be related to food context by text or by entity. In order to accomplish this, the main input of the algorithm is the tweet preprocessed text, which contains tokens, their stem and recognized entities. The algorithm is shown on Fig. 3 and explained next:

First, if the tweet only contains non-food n-gram, it is discarded and the algorithm finishes. Otherwise, the algorithm tries to determine a food context relationship by text or by entity:

- **By Text**: for each token in a tweet, the algorithm checks if its stem belongs to the *what list*, if so, it validates the token only if it is a noun. If there are no more tokens to check, the algorithm determines a food context relationship by text.
- **By Entity**: the algorithm determines a food context relationship by entity, if the tweet contains food context n-grams, brands, hashtags, mentions or emojis using the recognized entities from the preprocessed text.

Consequently, the algorithm assures a food context relationship only if, on the previous steps, at least one relationship or food mention was determined. In that case, the algorithm tries to identify context characteristics like places, people, or moments, using the *where*, *who* and *when lists*. Otherwise, the tweet is discarded. As a result, the algorithm stores five types of elements: (i) food n-grams, product or brand; (ii) place,



whose identification is made through words, hashtags or mentions; (iii) people, with whom the food is consumed; (iv) moment of consumption (time of day, consumption time, day of the week, among others) and finally, (v) tweet publication time.

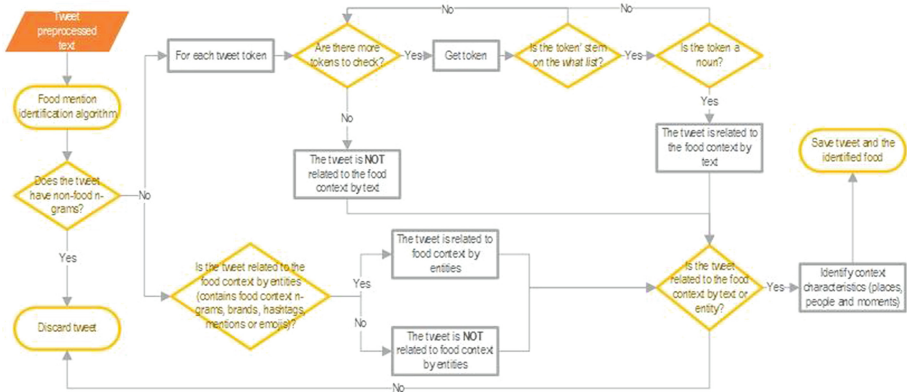


Fig. 3. Proposed food mention identification algorithm

### 3.3 Evaluation

To evaluate the proposed algorithm, an ETL (Extract, Transform, Load) system was designed and implemented using Big Data technologies. As shown in Fig. 4, Twitter is used as data source, which is extracted using its public API<sup>1</sup> implementation, in Python<sup>2</sup>.

The extracted data are stored in a MongoDB<sup>3</sup> database, in the *Staging area*. Then, in the *transformation stage*, four steps take place:

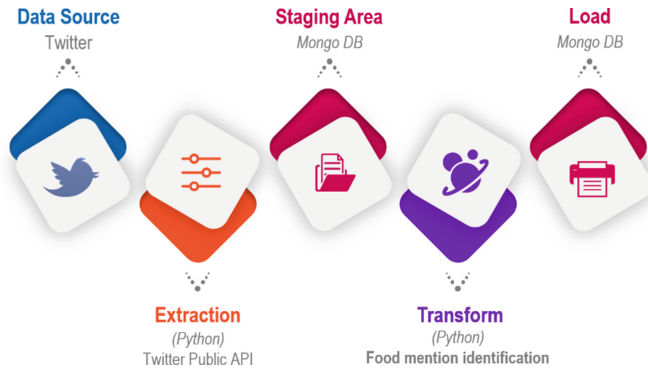
1. *Data cleaning*: in this step, data is selected from the *Staging area* to be prepared, only if their geographic location corresponds to Colombia and their language to Spanish.
2. *Text preprocessing*: in this step, Twitter text is tokenized, tagged, parsed, stemmed and lemmatized using the spaCy's<sup>4</sup> Spanish processing pipeline. Additionally, the structured text (user mentions, hashtags and emojis - using the *emoji list* described on Sect. 3.1.) is extracted and labeled accordingly.

<sup>1</sup> About Twitter's APIs, <https://help.twitter.com/en/rules-and-policies/twitter-api>, last access date: 3<sup>rd</sup> May 2018, we also provide companies, developers, and users with programmatic access to Twitter data through our APIs.

<sup>2</sup> Python, <https://www.python.org/>, last access date: 3<sup>rd</sup> May 2018, *Python is a programming language that lets you work quickly and integrate systems more effectively.*

<sup>3</sup> MongoDB, <https://www.mongodb.com/>, last access date: 3<sup>rd</sup> May 2018, *building on the Best of Relational with the Innovations of NoSQL.*

<sup>4</sup> spaCy, <https://spacy.io/>, last access date: 18<sup>th</sup> April 2018, *spaCy v2.0 Features new neural models for tagging, parsing and entity recognition.*



**Fig. 4.** ETL implementation design.

3. *Named Entity Recognition (NER)*: here, the n-grams from the following lists are recognized and labeled as entities within the text: *food list*, *food stop word list*, *food user mention list*, *food hashtag list* and *food entity list*.
4. *Food mention identification*: in this last step, our algorithm (see Sect. 3.2) is used to identify whether or not a tweet contains a food mention.

Finally, if the tweet contains a food mention, it is uploaded to Mongo DB in the *load stage*. It is worth mentioning that this exercise is part of a project named Digital Segmentation System from CAOBA [13], which, based on the information from Twitter, generates an approximation to the characteristics of the users who create the publications. These characteristics range from sociodemographic aspects, to socio-graphic attributes related to emotions, interests, and polarity. These variables will be considered in the next section.

Taking into account the ETL system, a case study was constructed with 1.3 million tweets extracted during fifteen days within the same month. In this period, our proposed algorithm identified 11,691 tweets that mentioned food, corresponding to 2% of the extracted tweets. A sample of the results obtained from the algorithm were manually evaluated to identify if the original tweet is actually related to the food context; as a result, a **precision** of 70% was obtained.

## 4 Results

The loaded tweets were classified depending on the type of the mentioned words. Our method manages to identify 1,310 different words, where 59 are mentioned 100 or more times. Table 1 shows the 20 most frequent words according to the time of day (breakfast: 5 am–9 am, lunch: 11 am–2 pm, snack: 10 am/3 pm–5 pm and dinner: 6 pm–9 pm).

In general, the word “cerveza” (beer) is the most frequent, almost at any time of the day; however, there is a group of words showing the consistency with a Colombian dietary routine to be mentioned at a specific time of day, such is the case of *bebida caliente* (hot drink), *pan* (bread), *queso* (cheese), *arepa* (white corn cake) and

**Table 1.** Number of tweets according to the type of content entity. All words mark with \* cannot be translated to English.

Breakfast		Lunch		Snack		Dinner	
Word	%	Word	%	Word	%	Words	%
cafe (coffee)	15,6%	cerveza (beer)	19,6%	cerveza (beer)	19,7%	cerveza (beer)	21,5%
cerveza (beer)	14,7%	torta (cake)	13,0%	torta (cake)	10,6%	comer (eat)	12,6%
torta (cake)	13,6%	almorzar (have lunch)	11,9%	comer (eat)	9,5%	cafe (coffee)	7,1%
comer (eat)	9,3%	comer (eat)	9,8%	cafe (coffee)	8,6%	pizza (pizza)	6,5%
bebida caliente (hot drink)	8,8%	cafe (coffee)	8,4%	aguacate (avocado)	6,6%	aguacate (avocado)	6,5%
pan (bread)	4,6%	pizza (pizza)	3,1%	vino (wine)	4,5%	torta (cake)	6,2%
tinto (black coffee)	4,5%	pollo (chicken)	3,1%	almorzar (have lunch)	3,7%	hamburguesa (burger)	4,2%
arepa (*)	2,8%	pan (bread)	3,0%	pan (bread)	3,7%	pan (bread)	3,8%
carne (meat)	2,7%	tragar (swallow)	2,9%	pizza (pizza)	3,7%	tragar (swallow)	3,6%
tragar (swallow)	2,5%	carne (meat)	2,8%	coctel (coctel)	3,6%	jugar (play)	3,1%
chocolate (chocolate)	2,4%	coctel (coctel)	2,5%	tragar (swallow)	3,3%	coctel (coctel)	3,1%
almorzar (have lunch)	2,3%	arroz (rice)	2,5%	pinchar (*)	3,2%	chocolate (chocolate)	3,0%
pizza (pizza)	2,1%	hamburguesa (burger)	2,4%	chocolate (chocolate)	2,9%	pinchar (*)	2,8%
queso (cheese)	2,1%	chocolate (chocolate)	2,4%	bebida caliente (hot drink)	2,8%	queso (chese)	2,5%
coctel (coctel)	2,1%	vino (wine)	2,4%	hamburguesa (burger)	2,6%	vino (wine)	2,5%
empanada (*)	2,1%	mango (mango)	2,3%	pollo (chicken)	2,5%	pana (friend)	2,5%
caballo (horse)	2,1%	pana (friend)	2,3%	carne (meat)	2,3%	empanada (*)	2,5%
aguacate (avocado)	1,9%	bebida caliente (hot drink)	2,0%	empanada (*)	2,2%	pollo (chicken)	2,4%
papaya (papaya)	1,9%	queso (cheese)	1,9%	queso (cheese)	2,1%	arepa (*)	1,9%
hamburguesa (burger)	1,8%	arepa (*)	1,8%	tinto (black coffee)	1,9%	carne (meat)	1,9%

*chocolate(chocolate)* at breakfast; or *arroz (rice)*, *pollo (chicken)*, *pizza y carne (pizza and meat)* at lunchtime.

According to the previously established classification, it was found that, 33.108 times, a word was identified as food, product or brand; 1.324 times as a place, 1.426 times as a companion and 2.726 times as a consumption occasion. For the three last classifications, the most frequently mentioned words are presented in Fig. 5.

Additionally, it is possible to know the behavior of users according to the day time in which they publish. Figure 6 shows a tendency to publish more tweets around noon and between 18–21 h.

In relation to the sociographic variables, emotion is detected for 40% of the tweets and polarity for 86.1%. When performing the individual analysis of the most frequent words and their relationship with the emotion of the tweet, it is observed (see Table 2)



Fig. 5. Words cloud for where, who and when

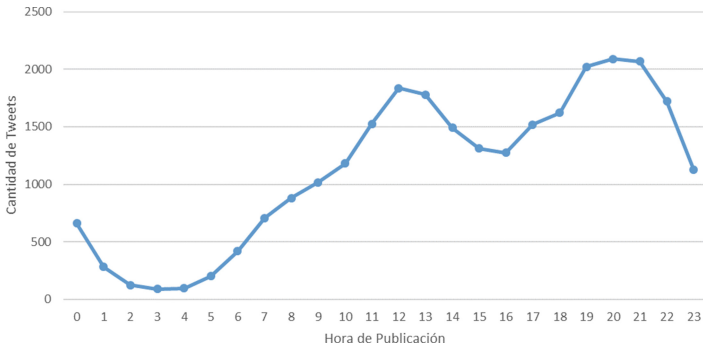


Fig. 6. Publication Frequency Distribution of tweets according to publication time

Table 2. Most frequent words according to emotions. All words mark with \* cannot be translated to English.

Words/Emotion	Happiness 😄	Anger 😡	Fear 😨	Repulsion 🤢	Surprise 😲	Sadness 😞	Total
cerveza (beer)	87,2%	2,5%	1,3%	1,1%	1,4%	6,5%	100,0%
torta (cake)	97,9%	0,2%	0,2%	0,0%	0,5%	1,2%	100,0%
comida (dinner)	57,7%	6,3%	2,8%	4,8%	4,4%	23,9%	100,0%
café (coffee)	72,6%	4,5%	2,9%	2,1%	3,2%	14,7%	100,0%
pizza (pizza)	65,8%	5,2%	3,6%	2,4%	2,7%	20,3%	100,0%
almorzar (lunch)	60,9%	7,6%	4,5%	1,4%	1,7%	23,9%	100,0%
coctel (cocktail)	93,2%	1,1%	2,2%	1,4%	0,4%	1,8%	100,0%
bebida caliente (hot drink)	91,1%	0,4%	1,9%	1,9%	0,7%	4,1%	100,0%
vino (wine)	93,6%	0,8%	1,1%	0,8%	0,4%	3,4%	100,0%
trago (*)	63,6%	5,7%	2,7%	2,7%	2,3%	23,0%	100,0%
chocolate (chocolate)	68,6%	5,7%	2,9%	1,4%	2,9%	18,6%	100,0%
aguacate (avocado)	92,9%	1,9%	1,0%	1,0%	0,0%	3,3%	100,0%
pollo (chicken)	65,3%	6,0%	3,3%	2,7%	1,3%	21,3%	100,0%
queso (cheese)	72,2%	3,5%	5,6%	2,1%	2,1%	14,6%	100,0%
hamburguesa (burger)	75,9%	3,0%	1,5%	3,0%	0,0%	16,5%	100,0%

that words like cake, cocktail, hot drink, wine and avocado, have a high participation with tweets related to joy. On the other hand, words like dinner, pizza and chicken, have shares in sadness exceeding 20% of the tweets.

#### 4.1 Characterizing Users by Age and Gender

The following table (see Table 3) shows the grouping by type of food and age groups of users who mention them, considering about 50% of the most mentioned words. It is observed that there is a trend in the 35-year old population, and more towards the mention of healthier foods, such as meat, cheese, chicken or the so-called “Natural Food” which in this text refers to as the usual or homemade food: rice, pasta, potato. The tendency to mention alcoholic beverages is strong in the Colombian tweets; however, it is much more pronounced in the population under 35 years.

**Table 3.** Food group versus age range.

Food group/age range	13–24	25–34	35 and more
Bebida alcohólica (alcoholic beverage)	36,6%	35,0%	26,4%
Bebidas (drinks)	13,0%	12,6%	16,5%
Carne, queso, pollo, queso (Meat, cheese, chicken, cheese)	9,5%	8,1%	10,0%
Comida natural (natural food)	15,9%	17,3%	18,5%
Comida rápida (fast food)	7,2%	5,1%	6,2%
Frutas, verduras (fruits, vegetables)	1,8%	1,1%	4,8%
Helados, postres (ice cream, desserts)	1,1%	0,9%	1,1%
Panes, tortas, arepas (breads, cakes, arepas)	14,9%	20,0%	16,5%
<b>Total General</b>	<b>100,0%</b>	<b>100,0%</b>	<b>100,0%</b>

When the differences at the gender level are observed, there is a tendency of men towards the mention of alcoholic beverages, such as wine, drink and beer; whereas in women, terms such as desserts, sweets, milkshakes and chocolates, are more frequent. That is, a pronounced tendency was found in women towards mentioning sweets; nevertheless, the mention of fruits by them, is also evident. These results are presented in the word clouds of Fig. 7.

The differences in the mentions of alcoholic beverages are also perceived at a socio-economic level, representing the greatest differences; such as fondness for beer into lower socio-economical levels, and the opposite for cocktail, a more expensive drink. Coffee and hot drinks also predominant in high strata. The following table (see Table 4) presents the words showing the greatest differences between strata.



**Fig. 7.** Food related words by gender (men and women)

**Table 4.** Word distribution by socioeconomic level. All words mark with \* cannot be translated to English.

Word/Socioeconomic level	High	Medium	Low
Cerveza (beer)	25,21%	32,09%	34,75%
Trago (shot)	3,55%	5,81%	9,62%
Pan (bread)	7,11%	4,89%	6,53%
Pizza (pizza)	5,72%	6,62%	8,32%
Empanada (*)	0,23%	0,11%	0,16%
Aguacate (avocado)	0,78%	8,05%	2,77%
Pollo (chicken)	5,03%	3,32%	6,20%
Pasta (pasta)	2,72%	1,94%	3,92%
Arroz (rice)	3,19%	2,60%	4,08%
Tinto (black coffee)	0,88%	0,62%	0,98%
Arepa (*)	3,83%	2,55%	1,79%
Coctel (cocktail)	5,12%	4,59%	2,45%
Carne (meat)	5,54%	3,09%	2,61%
Jugo (juice)	3,60%	3,43%	6,04%
Café (coffee)	3,37%	1,81%	1,96%
Bebida caliente (hot drink)	8,59%	3,39%	1,14%
Torta (cake)	15,51%	15,10%	6,69%
<b>Total</b>	<b>100,0%</b>	<b>100,0%</b>	<b>100,0%</b>

## 5 Discussion and Future Work

Food preferences expressed in social networks as Twitter become a valuable source of information for making decisions about consumer centralization strategies. Therefore, knowing tendencies of publishing, interests and behaviors based on comments published by users allows identifying pertinence and strength of marketing strategies.

Through a case study, it is shown that Twitter information provides some elements that allow a global analysis of preferences in foods, products or brands, based on the mentions made by users in the network. Despite founding just a 2% of messages related

to food, there is a significant number of users, which would significantly exceed approximations made by other methodologies, such as specialized surveys. The advantage of having continuous information collection also enables a significant increase in the volume of users that can be identified over time, as well as the identification of patterns or changes in behavior, constituting a very relevant aspect.

One of the most valuable elements of the exercise is the generation of the knowledge base, which must be adjusted to ensure that the products of interest (including those of competitors) are at the base; in turn, more specific relationships can be established about users' opinions or emotions about them. An advantage of the way in which the algorithm was implemented is the possibility of making these adjustments without major difficulties. This would create new sources of unstructured open data, allowing other systems to feed from their knowledge bases, such as systems of health, marketing or others [15].

Despite the remarkable advantages, it is important to note that the algorithm's accuracy is 70%, a value associated mainly with trying to build a knowledge base for such a broad domain. This behavior affects the results, generating erroneous interpretations; however, as mentioned before, if this algorithm was applied to a more specific domain, its performance would increase.

To estimate the magnitude of interpretation errors, it will be necessary to deepen in a content analysis where the intentionality is verified directly in the tweet texts. This kind of analysis requires a huge amount of time for its completion, which exceeds the initial objectives and scope of this research. However, it is proposed as a future work.

**Acknowledgements.** This research was carried out by the Center of Excellence and Appropriation in Big Data and Data Analytics (CAOBA). It is being led by the Pontificia Universidad Javeriana Colombia and it was funded by the Ministry of Information Technologies and Telecommunications of the Republic of Colombia (MinTIC) through the Colombian Administrative Department of Science, Technology and Innovation (COLCIENCIAS) within contract No. FP44842- anex46-2015.

## References

1. Vidal, L., Ares, G., Machín, L., Jaeger, S.R.: Using Twitter data for food-related consumer research: a case study on “what people say when tweeting about different eating situations”. *Food Qual. Prefer.* **45**, 58–69 (2015)
2. Nielsen Company: Advertising and audiences: State of the media, May 2014. [http://www.nielsen.com/content/dam/niensenglobal/jp/docs/report/2014/Nielsen\\_Advertising\\_and\\_AudiencesReport-FINAL.pdf](http://www.nielsen.com/content/dam/niensenglobal/jp/docs/report/2014/Nielsen_Advertising_and_AudiencesReport-FINAL.pdf)
3. Janasz, T., Koschmider, A., Born, M., Uhl, A.: Digital capability framework: a toolset to become a digital enterprise. In: *Digital Enterprise Transformation*, pp. 51–84. Routledge, Abingdon (2016)
4. Uhl, A., MacGillavry, K., Diallo, A.: Digital transformation at DHL freight: the case of a global logistics provider. In: *Digital Enterprise Transformation*, pp. 287–302. Routledge, Abingdon (2016)

5. Abbar, S., Mejova, Y., Weber, I.: You Tweet what you eat: studying food consumption through Twitter. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, pp. 3197–3206. ACM, April 2015
6. Prier, Kyle W., Smith, Matthew S., Giraud-Carrier, C., Hanson, Carl L.: Identifying health-related topics on Twitter. In: Salerno, J., Yang, S.J., Nau, D., Chai, S.-K. (eds.) SBP 2011. LNCS, vol. 6589, pp. 18–25. Springer, Heidelberg (2011). [https://doi.org/10.1007/978-3-642-19656-0\\_4](https://doi.org/10.1007/978-3-642-19656-0_4)
7. Dredze, M., Paul, M.J., Bergsma, S., Tran, H.: Carmen: a Twitter geolocation system with applications to public health. In: AAAI Workshop on Expanding the Boundaries of Health Informatics Using AI (HIAI), vol. 23, p. 45, June 2013
8. Moreno-Sandoval L., et al.: CSL: a combined Spanish Lexicon - resource for polarity classification and sentiment analysis. In: Proceedings of the 19th International Conference on Enterprise Information Systems, vol. 1, ICEIS, pp. 288–295 (2017). ISBN 978-989-758-247-9. <https://doi.org/10.5220/0006336402880295>
9. Moreno-Sandoval L., Mendoza-Molina J., Puertas E., Duque-Marín A., Pomares-Quimbaya A., Alvarado-Valencia J.: Age classification from Spanish Tweets - the variable age analyzed by using linear classifiers. In: Proceedings of the 20th International Conference on Enterprise Information Systems, vol. 1, pp. 275–281. ICEIS (2018). ISBN 978-989-758-298-1. <https://doi.org/10.5220/0006811102750281>
10. Felbo, B., Mislove, A., Sogaard, A., Rahwan, I., Lehmann, S.: Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm (2017). arXiv preprint, [arXiv:1708.00524](https://arxiv.org/abs/1708.00524)
11. Quercia, D., Kosinski, M., Stillwell, D., Crowcroft, J.: Our Twitter profiles, our selves: predicting personality with Twitter. In: Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third International Conference on Social Computing (SocialCom), pp. 180–185. IEEE, October 2011
12. Krishnamurthy, B., Wills, C.E.: Characterizing privacy in online social networks. In: Proceedings of the first workshop on Online social networks, pp. 37–42. ACM, August 2008
13. Vargas-Cruz, J., Pomares-Quimbaya, A., Alvarado-Valencia, J., Quintero-Cadavid, J., Palacio-Correa, J.: Desarrollo de un Sistema de Segmentación y Perfilamiento Digital. *Procesamiento Del Lenguaje Natural*, 59, 163–166 (2017). Recuperado de <http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/5511>
14. Petz, G., Karpowicz, M., Fürschuß, H., Auinger, A., Střiteský, V., Holzinger, A.: Reprint of: computational approaches for mining user’s opinions on the Web 2.0. *Inf. Process. Manag.* **51**(4), 510–519 (2015)
15. Calero Valdez, A., Ziefle, M., Verbert, K., Felfernig, A., Holzinger, A.: Recommender systems for health informatics: state-of-the-art and future perspectives. In: Holzinger, A. (ed.) *Machine Learning for Health Informatics*. LNCS (LNAI), vol. 9605, pp. 391–414. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-50478-0\\_20](https://doi.org/10.1007/978-3-319-50478-0_20)
16. Puertas Del Castillo, E., Alvarado Valencia, J., Pomares Quimbaya, A.: Constructor automático de modelos de dominios sin corpus preexistente. *Procesamiento Del Lenguaje Natural*, 59, 129–132 (2017). Recuperado de <http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/5503>





# Feedback Matters! Predicting the Appreciation of Online Articles A *Data-Driven Approach*

Catherine Sotirakou<sup>1</sup>(✉), Panagiotis Germanakos<sup>2</sup>,  
Andreas Holzinger<sup>3</sup>, and Constantinos Mourlas<sup>1</sup>

<sup>1</sup> Faculty of Communication and Media Studies,  
National and Kapodistrian University of Athens,  
Sofokleous 1, 10559 Athens, Greece  
{katerinasot,mourlas}@media.uoa.gr

<sup>2</sup> Products & Innovation, SAP SE,  
Dietmar-Hopp-Allee 16, 69190 Walldorf, Germany  
panagiotis.germanakos@sap.com

<sup>3</sup> Institute of Medical Informatics, Statistics and Documentation (IMI),  
Medical University Graz, Auenbruggerplatz 2, 8036 Graz, Austria  
andreas.holzinger@medunigraz.at

**Abstract.** The current era of advanced computational mobile systems, continuous connectivity and multi-variate data has led to the deployment of rich information settings that generate constant and close to real-time feedback. Journalists and authors of articles in the area of Data Journalism have only recently acknowledged the influence that the audience reactions and opinions can bring to effective writing, so to be widely appreciated. Such feedback may be obtained using specific metrics that describe the user behavior during the interaction process like shares, comments, likes, claps, recommendations, or even with the use of specialized mechanisms like mood meters that display certain emotions of readers they experience while reading a story. However, which characteristics can reveal an article's character or type in relation to the collected data and the audience reflection to the benefit of the author? In this paper, we investigate the relationships between the characteristics of an article like structure, style of speech, sentiment, author's popularity, and its success (number of claps) by employing natural language processing techniques. We highlight the emotions and polarity communicated by an article liable to increase the prediction regarding its acceptability by the audience.

**Keywords:** Data journalism · Natural Language Processing · Sentiment Emotions · News articles · Computer-assisted content analysis  
Machine learning

## 1 Introduction

In recent years, the advancement of computational systems and devices, along with the explosive growth and availability of open data have led to computational journalism's growth. Now, data journalists use in their news preparation and writing, software and technologies which are found in the cross-borders of three different research disciplines:

Computer Science, Social Sciences, and Media & Communications. According to [12] science and journalism were two fields that coexisted for the first time in Philipp Meyer's book [16] *Precision Journalism*, where he explained that the implementation of scientific methods used in social science experiments when used in reporting, produce very good results in both the news-gathering process and the final story. While computation has long been assisting journalists in different phases of the news production process, and what is known as computer-assisted reporting has been practiced since the 1960s [10], the scientific investigation of it has been more rigorously undertaken the last decade. According to an extensive literature review from Ausserhofer [4], a significant research output on data journalism and related fields has been produced only since 2010, with a prior small number of attempts to be regarded as isolated. Researchers and journalists usually study ways and explore methods to (a) locate and extract useful data (like information-rich textual data on the internet), (b) analyze and understand the meaning of massive datasets (i.e., trying to draw connections between the sender and the recipient of an email using email meta-characteristics), and (c) acknowledge more thoroughly the user feedback (Twitter data analysis) in their analysis and articles composition.

In our research, we mainly focus on the third research stream which refers to the *what* and *how* to incorporate the audience feedback in judging both the quality and the character of an article from authors' point of view facilitating their writing. The central motivational factor is the mind shift which has been observed lately regarding both the online news consumption and journalists' tendency to learn more about their audiences [1, 23]. Authors and editors used to ignore the audience feedback [5, 14] relying mainly on personal predictions what they thought the audience desires might be or as Tandoc [32] suggests "they substituted their own preferences for those of their actual audience". According to the literature one of the reasons for this reluctant attitude towards the audience feedback was the fact that journalists should follow clear journalistic norms and remain autonomous without being affected by external preferences that might erode journalistic quality [5]. However, the explosion of big data and information technologies revolutionized the way that readers provide feedback to journalists (reviews, letters to the editor). Today, newsrooms have Web analytics tools that provide insights such as clicks, page views, viewing time, conversion funnels analysis and user flows that are only some of the audience metrics that help authors monitor how people interact with the online content on their newsroom's website. Accordingly, Natural Language Processing (NLP) techniques are widely applied in sentiment analysis so to determine which features users are particularly in favor of or to trace content that needs to be removed (in case of negative feedback).

Our main concern is not simply to employ data-driven techniques predicting the probability for an article to gain more likes, claps, or recommendations but rather to go a step further and embrace a more human-centered approach, to understand the characteristics that make a great article based on the motivation and the scope that the journalist wants to communicate. This information will help on enriching an *a priori* knowledge that could be utilized as one more dimension and guideline, contributing to the success of the potential next article. For the scope of this paper, we highlight the emotions and polarity that an article communicates in relation to other well-recognized characteristics of article's style and author's reputation. We tackle the problem of

finding the interesting relations between the characteristics of an article namely the (i) article's structure, (ii) style of speech, (iii) sentiment (iv) author's popularity, and its success (number of claps), assuming that for online articles posted on a blogging platform it is possible to establish a predictive link between several content-based facets and success. We use several NLP and machine learning methods for cleaning, filtering, and processing text data like measuring term frequency and lexicon based approaches, as well as for checking the importance of each factor, and also to evaluate the predictive power of our model. Our preliminary results (from a sample of 3000 articles extracted from the Medium online publishing platform in 2017) has shown that indeed characteristics of the user, emotions expressed in the text, personal tone of speech and the use of uncommon words influence the prediction of the results regarding the level of acceptability by the audience.

## 2 Related Work

In this section, we describe the distinctive attributes of the text as determinants of articles' quality and acceptability by the broader audience. We emphasize on previous researches that consider sentiment, structure and writing style since those factors have been used to verify our experiments. Journalists write stories hoping that their writing will arouse emotions in audiences, sometimes positive like hope, joy or trust and others negative such as fear and anger depending on the given coverage. The emotional state of the online reader is a key factor for authors to understand the audiences' reactions to the news stories [6], e.g. know how the reader would respond to their article, if they manage to express the desired emotion correctly on their writing, etc. The digital age offers a constantly increasing number of sentiment-rich resources like news articles available on the Web while technology nowadays allows readers to leave feedback and show their appreciation easily. Such an appreciation is usually demonstrated through various actions like sharing e.g., an article or post of interest with the community or with the targeted audience, ranking the content based on the quality or interest it might present or following the owner of the content.

According to literature, emotional online news content especially awe-inspiring content is more likely to become viral [6]. Different NLP methods that are trying to capture emotional categories, sentiment polarity, and emotional dimensions [24] have been proposed by researchers to extract the emotions expressed in the texts. Intelligent natural language processing tasks are required to be more sophisticated to improve their accuracy, thus a variety of sentiment and emotion lexica and corpora [11, 21, 31] have been created that can identify a plethora of emotions instead of just suggesting whether they express positive, negative or neutral sentiment. Indicators of collective user behavior and opinion are increasingly common features of online news stories and may include information about how the story made the readers feel. Typical example of such features is Facebook's set of emoticons called "Reactions", that urges users to express their experienced emotions about a post by using a button that includes five different emotional states: Love, Haha, Wow, Sad, Angry [8, 33].

Other approaches have focused on determining what are the characteristics of a good structure for a news article regarding text coherence. As Louis and Nenkova [18]

have previously categorized in their work, there are three ways of testing text coherence “by exploring systematic lexical patterns, entity coreference and discourse relations from large collections of texts”. In terms of measuring the quality of a given article as a whole, recent work in the field has decomposed news articles into a set of simpler characteristics that reveal different linguistic and narrative aspects of online news [2]. Arapakis and his colleagues after discussing with several journalists, editors and computational linguists, proposed “a multidimensional representation of quality”, taken from the editor’s perspective that groups the attributes of a news story into five categories: Readability, informativeness, style, topic, and sentiment, with each category having several sub-categories such as fluency and conciseness for the readability and subjectivity, sentimentality, and polarity for sentiment. Their findings suggest that the journalists’ perception of a well-written story correlates positively with fluency, richness (feature from the category style) and completeness (feature from the category informativeness), while aspects like subjectivity and polarity proved to be weakly correlated. Therefore, this particular work suggests that sentiment is not of great importance when it comes to article quality, whereas text comprehension and writing style seem to be determinants of quality.

To examine the potential effects of emotions, writing style and readability on article quality prediction in the journalism domain, researchers Louis and Nenkova [19], investigated science articles from the New York Times, by separating them into two categories “very good”, in which articles from the authors whose writing appeared in “The Best American Science Writing” anthology series were included, and the “typical” category that included all the remaining articles. Their experiments showed that “excellent authors associated with greater degree of sentiment, and deeper study of the research problem” as well as usage of what is called beautiful language, meaning the unusual phrasing. This approach is similar to ours, however, this study explored only science articles from the New York Times, which we already know that are examples of good quality journalism. In our approach, we consider articles from a broad spectrum, written not only by professional journalists but mostly by amateur writers.

The scientific community also has been experimenting with predictive analytics [7, 20, 30]. In their work McKeown et al. [20] present a system that predicts the future impact of a scientific concept, based on the information available from recently published research articles, while Sawyer et al. [30] suggest that award-winning academic papers use simple phrasing. However, a successful news story would be described by alternative characteristics in contrast to good academic writing. In another work, Ashok, Feng and Choi [3] used statistical models to predict the success of a novel based on its stylistic elements. Their work examined a collection of books and movie scripts of different genres, providing insights into the writing style, such as lexical and syntactic rules, sentiment, connotation, and distribution of word categories and constituents commonly shared among the best sellers. The results suggest that successful writing includes more complex styling features like prepositions, pronouns, and adjectives while unsuccessful novels have more verbs, adverbs, and foreign words, thus having higher readability. Moreover, this work found that the writing style of successful novels is similar to news articles.

Still, very little research is available on preprocessing noisy texts, which is usually done, particularly by large companies (e.g. Facebook) manually to identify and correct

spelling errors, or other noise (whitespaces, boundaries, punctuation) [35]. This field is a topic of research for quite a long time, but the effects of cleaning text passages is still rarely described [36]. The problem of noise is underestimated within the machine learning community, but has serious consequences for language identification, tokenization, POS tagging and named entity recognition.

Finally, there are many works today that have applied in different cases for textual analysis factors like animate and personal pronouns, use of visual words, people-oriented content, use of beautiful language, sub-genres, sentiment, and the depth of research description [19] lexical choices such as thinking and action verbs [3]. Readability is also a significant factor, in their work [29] explored features like word and sentence length, cohesion scores and syntactic estimates of complexity. Their results showed that increased use of verb phrases provides better readability. In our current work we use those factors in combination since we believe can determine the quality and acceptance of an article regarding structure, style, author’s popularity and emotionality.

### 3 Method and Dataset

We employ a data-driven approach aiming to respond to two typical related research questions: (a) To what extent the character of an article can reveal the reaction of the readers, producing more or fewer claps? and (b) would it be of importance to predict the character of an article to the benefit of the journalist? We investigate textual data from online articles on the Web so to help the authors to adjust their writing style, gaining more acceptance from their audience. We extract content-based features from the articles on the online publishing platform, Medium<sup>1</sup>, based on relevant metrics concerning those proposed in the reviewed literature, such as the use of beautiful language, the tone of speech, polarity and sentiment, and genre.

Before we begin to delve into the effect of the different characteristics of an article to its success, we first need to extract features using text analysis techniques related to the four different aspects of an article that we suggest, namely the (i) article’s structure, (ii) style of speech, (iii) sentiment (iv) author’s popularity. We use several NLP methods for cleaning, filtering, and processing text data like measuring term frequency and lexicon based approaches. After defining the above categories we need to study their importance on the article’s success, applying machine learning algorithms. A random forest classifier is used to evaluate the significance of the features above in the success of an article as well as to create a decision tree able to predict whether the claps count of a given article will be high, medium or low. We run experiments on a mixed-topic dataset of over 3 thousand articles published at 2017 and downloaded at the beginning of 2018. The articles had a large distribution of claps ranging from 84 to 157 K claps.

As we mentioned earlier, for this research we collected data from an online publishing platform called “Medium”, that hosts a variety of articles and publications produced by either amateur or professional writers, journalists, bloggers, companies, and range from short to long articles, with topics that cover a variety of topics such as

---

<sup>1</sup> <https://medium.com>.

science, education, politics, well-being etc. Medium provides to the reader an automatically calculated display of the reading time on every article so they can know how much is required of them to read through an entire story. Like votes on Digg<sup>2</sup> stories, “claps” (formerly called “Recommend”) on Medium represent whether readers liked the story or not and would recommend it to other users on the platform (a user can clap more than once). In the world of Medium, the success of an article is measured regarding claps count, which is the number of times the readers have been clapped.

In this paper, we suggest that a more comprehensive understanding of the content of an article should comply with four directions, as content specification elements. Below we describe in more detail what each one represents along with the method and resources used.

### 3.1 Content Structure

Good structure is a key to higher quality articles [2, 3]. In our model, we use five characteristics of structure: genre, title words, reading time (since it is predetermined by the Medium), the proportion of images in the text and proportion of bullet points in the article.

*Genre:* The topic of the article reflect certain characteristics of its nature and it has been greatly investigated in previous work [2, 3, 19]

*Title:* The title of an article is a significant aspect of every story and is also a clickbait strategy that is being used a lot in journalism [13, 34]

*Reading time:* every article posted on Medium has its length measured automatically, by the time it would take the user to read it.

*Images:* The use of a great number of beautiful images is one of the best practices used in social media marketing.

*Bullet points:* According to research the extensive use of the so-called listicle; a mixture of ‘list’ and ‘article’ is an interesting phenomenon and its power lies “not only in the power of the format in and by itself but also in ‘shareable factors’ that are related to the individual listicle” [26].

### 3.2 Style

There are multiple aspects that characterize the style of speech, and stylistic variation reflects largely the meaning that is communicated [27]. Many researchers have examined this dimensions of writing style by measuring features like formal language [17, 27, 28], attractiveness and richness [2], and use of beautiful language that refers to words with lowest frequency or with highest perplexity under the phoneme and letter models [19]. Except for the above style-related dimensions we propose the personal tone of speech, typically examined in communication research and more specifically in political speeches.

---

<sup>2</sup> <http://digg.com>.

*Tone of speech:* Communication researchers have studied the self-referential or self-reflexive messages that the media and advertising companies use to attract the audience’s attention [25]. Politicians like Barack Obama also have recognized the persuasive power of personal pronouns and they use it strategically to their rhetoric [22]. Moreover, personal pronouns are an indicator of the existence of people in the story and prior works have experimented with human-related stories [19] as a factor of success. In our study, we investigate the use of personal pronouns (I, you, we etc.), reflexive pronouns like myself, yourself, and possessive pronouns (mine, yours, etc.).

*Beautiful phrasing:* Using beautiful phrases and creative words can amuse the audience and according to findings of Louis and Nenkova [19] they are discovered in high-quality articles. We apply Term Frequency Inverse Document Frequency (TF-IDF) to determine what words in the corpus of articles might be rarer and thus favorable to use in our model.

### 3.3 Sentiment

Emotions expressed in an article can motivate people to leave feedback, share and rate the content [3, 6, 19]. Theories in the nature of emotion, like Plutchik’s model, suggest the existence of basic emotions such as joy, sadness, anger, fear, trust, surprise, disgust, and anticipation [24]. For this study we used the resource EmoLex11, from the NRC suite of lexica, that is based on Plutchik theory, to extract the sentiment and polarity expressed in the articles. This word-emotion association lexicon [21] contains 14,182 words labeled according to emotions and also includes annotations for negative and positive sentiments. We investigate four affect-related features: the density of both positive and negative emotions, the density of each emotion (joy, sadness, anger, fear, trust, surprise, disgust, and anticipation), emotion expressed in the title, and polarity. We compute the counts of emotion words, each normalized by the total number of article words, and the total count of all the emotion words both negative and positive.

### 3.4 Author’s Popularity

Popularity indicates the total followers -potential readers- of an author on Medium and usually is used in social media network analysis, where both features like followers and following (users that a given user is following) play a crucial role on the user’s position in the social network.

In preparation for the analysis, we further “cleaned” the dataset of 3030 articles by removing all the NaN values, html code, foreign languages and end up with 2990 useful articles. Furthermore, we stemmed the texts and dropped all the stop words and non-standard words and characters, such as punctuations, spaces, special characters and urls.

Note that our feature computation step is not tuned for the quality prediction task in any way. Rather we aim to represent each facet as accurately as possible. Ideally we would require manual annotations for each facet (visual, sentiment nature etc.) to achieve this goal. At this time, we simply check some chosen features’ values on a random collection of snippets from our corpus and check if they behave as intended without resorting to these annotations.

## 4 Analysis and Results

We started by converting the numerical variable that represents the number of claps into categorical one, having three values representing low, medium and high acceptance. Suitable python libraries were imported to automatically convert the numerical variables that represent claps in the three different groups, and thus get a quick acceptance segmentation by binning the numerical variable in groups.

Our next objective becomes now to develop a news classifier and study the effect of the selection of our predictor variables on the performance of the acceptance prediction model. The *randomForest* package from *sklearn* library in Python was used to create a classifier of our articles and additionally to measure the importance of the predictor variables. The importance of a variable is computed internally during the construction of the decision trees by checking the increase of prediction error when data for that variable is permuted while all others are left unchanged. In the case of Random Forest, to determine the importance of the predictor variables, we calculated the Gini index for each of them. After that, we ordered from highest to lowest rate and kept the 15 most important variables out of 44.

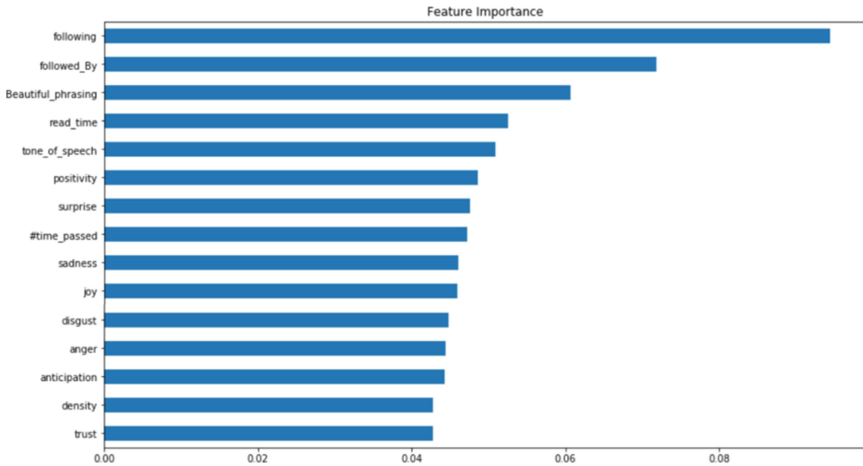
The table above (see Table 1) presents the ordered list of the importance of the variables of the selected categories, such as authors' popularity, beautiful phrasing etc., starting from the most important variables that affect the acceptance of an article. A graphical representation is also depicted in Fig. 1.

**Table 1.** Ordered list of the importance of selected features affecting the acceptance of an article.

Feature	Importance
Following	0.094
Followers	0.071
Beautiful phrasing (rare words)	0.060
Read time (length)	0.052
Tone of speech (self-reference)	0.052
Positivity	0.048
Surprise	0.047
Time passed (since publication date)	0.047
Sadness	0.046
Joy	0.046
Disgust	0.044
Anger	0.044
Anticipation	0.044
Density	0.042
Trust	0.042

For prediction purposes we split the dataset into two sets, one containing 80% of the total articles that was used as the training set and the other 20% was used to test the classifier so that the model can be trained and tested on different data. We run the





**Fig. 1.** Graphical representation of the importance of the features.

model on the train data to construct the confusion matrix and compare the predictions with the observations in the validation data set, which is as we said different from the one used to build the model.

The labels above represent the low, medium and high numbers of claps according to the acceptance segmentation process we discussed earlier by binning the numerical variable into three groups. Thus, bin 0 represents the low acceptance of the users while bin 2 represents the group of articles that have a high number of claps. The bin 1 represents the intermediate state (see Table 2).

Studying the confusion matrix (see Fig. 2) we can see that the predictive power of our model is mainly focused on the:

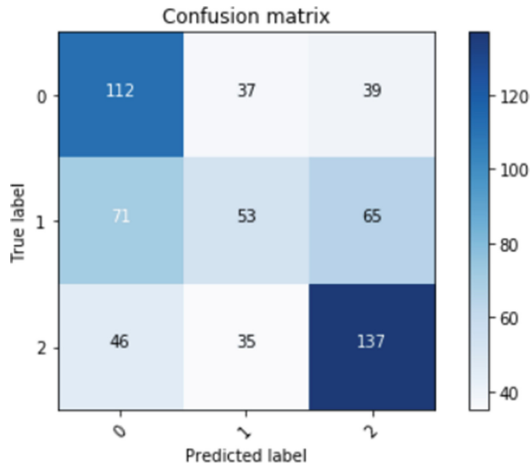
- Label 0 representing articles having a very low number of claps (between 87 and 1500) and,
- Label 2 representing the famous articles (claps between 5400 and 55000).

The model seems to have a poor performance on the intermediate state for articles where the number of claps is between 1500 and 5400. The predictive power for the 0-label and 2-label is 63%.

Our experiments revealed several interesting insights. Our empirical results indicate that the following relationships between Medium users are the most important factors in predicting the audience's positive feedback, thus the author's readership depends

**Table 2.** The three groups (high, medium, low) of clap numbers.

Bucket	Start	End
0	87.0	1500.0
1	1500.0	5400.0
2	5400.0	55000.0



**Fig. 2.** Graphical representation of the confusion matrix.

highly on his online followers. This is a logical consequence because a greater number of followers results in a larger potential audience that is likely to leave positive feedback, especially if they already like the author’s previous work.

We observed that our results are in line with the related literature, such as the power of self-reference in communicating a message and the fact that good writers are logophiles and prefer to use uncommon and long words in their appealing writing. Our findings suggest that making unusual word choices in writing might result in better feedback and accord with previous studies that proved the adults’ preference in unfamiliar words to the everyday ones [9].

News articles are not only anonymously edited information but are essentially narrated, and according to Grunwald [15], they are “constructed by a personally involved, individual journalist performing a role as an engaged narrator using a variation of communication acts” hoping to achieve a reliable and interesting deliverance of the message. Likewise, in our experiments, the use of personal tone of speech is highly correlated with success.

Another finding was that the only structure-based feature that correlates with success is article’s length, while aspects such as genre, the number of title words, images, and bullet points present a poor correlation with positive feedback.

Moreover, our hypothesis that sentiment is of great importance is proven by our model, with aspects such as positivity and the emotion of surprise performing better than anger and disgust. Also, emotional density is an important feature for predicting the article’s success in contrast to the aspect of polarity.

Finally, it is widely agreed that time factors like publishing date can change the audience appreciation depending on the platform’s popularity over a certain period. We ran experiments where we incorporated the time factors into the model to accurately capture the user clapping over time, and the results were surprisingly accurate. It seems that platform’s popularity in time is particularly influential factor in the accuracy of a model and will be further investigated in the future.

## 5 Conclusions

Nowadays, we are living in an era of rich information settings that generate constant and real-time feedback. In the area of Data Journalism, we observe a mind shift of journalists and authors towards extracting and learning more about their audiences. On the other hand, users-readers need to interact with a vast amount of contents residing in a variety of resources so to find what they are looking for. Especially, in the case of online articles' consumption such a reality makes them more selective and critical on which ones to read and stories to appreciate. This necessitates the consideration of the audience reflection and opinions in the composition of articles if we expect that they will meet their purpose and will be successful. Therefore, the main challenge is how can we figure out the character of the article based on the readers' feedback? Predicting the reaction of the audience toward a story has become of considerable importance not only for researchers and scientists but also for media organizations and digital managers that heavily invest in software to gain insights into readers behavior.

In this paper, we propose a model to discover the characteristics of online articles that result in greater audience acceptance. Our model relates to four different dimensions of articles' characteristics namely, article's structure, style of speech, sentiment and author's popularity. We applied several NLP and machine learning algorithms to extract a consensus, in terms of features' importance and prediction on claps, about the data derived from the online publishing platform "Medium". From our experiments we can formulate a preliminary understanding that several attributes characterize online article's success. Our findings demonstrate that indeed characteristics of the user, emotions expressed in the text, personal tone of speech and the use of uncommon words are highly correlated with influence of acceptance by the audience.

In the future, we plan to focus on the development of more articles' characteristics and examine inherent correlations aiming at optimizing and further improving the prediction of our model. We also plan to adopt a human-centred viewpoint on the interpretation of the features, and their subsequent relationships, in an attempt to identify a weighted impact on the final result. Such a finding might trigger a deeper understanding on the requirements of a successful article based on the reactions of the audience so to be used as guidelines for the journalists and authors facilitating their success.

**Acknowledgements.** This work is supported by the General Secretariat for Research and Technology (GSRT) and the Hellenic Foundation for Research and Innovation (H.F.R.I.) in the context of the action '1st Proclamation of Scholarships from ELIDEK for PhD Candidates'.

## References





1. Anderson, C.W.: Between creative and quantified audiences: web metrics and changing patterns of newswork in local US newsrooms. *Journalism* **12**(5), 550–566 (2011)
2. Arapakis, I., Peleja, F., Berkant, B., Magalhaes, J.: Linguistic benchmarks of online news article quality. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), vol. 1, pp. 1893–1902 (2016)

3. Ashok, V.G., Feng, S., Choi, Y.: Success with style: using writing style to predict the success of novels. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1753–1764 (2013)
4. Ausserhofer, J., Gutounig, R., Oppermann, M., Matiasek, S., Goldgruber, E.: The datafication of data journalism scholarship: focal points, methods, and research propositions for the investigation of data-intensive newswork. *Journalism* (2017) <https://doi.org/10.1177/1464884917700667>
5. Beam, R.A.: How newspapers use readership research. *Newsp. Res. J.* **16**(2), 28–38 (1995)
6. Berger, J., Milkman, K.L.: What makes online content viral? *J. Mark. Res.* **49**(2), 192–205 (2012)
7. Bergsma, S., Post, M., Yarowsky, D.: Stylometric analysis of scientific articles. In: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 327–337. Association for Computational Linguistics (2012)
8. Chaykowski, K.: Facebook No Longer Just Has A ‘Like’ Button, Thanks To Global Launch Of Emoji ‘Reactions’, *Forbes* article (2016). <https://www.forbes.com/sites/kathleenchaykowski/2016/02/24/facebook-no-longer-just-has-a-like-button-thanks-to-global-launch-of-emoji-reactions/#29919a54692d>. Accessed 04 Mar 2018
9. Colman, A.M., Walley, M., Sluckin, W.: Preferences for common words, uncommon words and non-words by children and young adults. *Br. J. Psychol.* **66**(4), 481–486 (1975)
10. Cox, M.: The development of computer-assisted reporting. Informe presentado en Association for Education in Journalism and Mass Communication. Universidad de Carolina del Norte, Chapel Hill (2000)
11. de Albornoz, J.C., Plaza, L., Gervás, P.: SentiSense: an easily scalable concept-based affective lexicon for sentiment analysis. In: LREC, pp. 3562–3567 (2012)
12. Flew, T., Spurgeon, C., Daniel, A., Swift, A.: The promise of computational journalism. *Journal. Pract.* **6**(2), 157–171 (2012)
13. Frampton, B.: Clickbait: The changing face of online journalism. *BBC News*, 14 September 2015
14. Gans, H.J.: *Deciding what’s news: A study of CBS evening news, NBC nightly news, Newsweek, and Time*. Northwestern University Press (1979)
15. Grunwald, E.: Narrative norms in written news. *Nord. Rev.* **26**(1), 63–79 (2005)
16. Meyer, P.: *Precision Journalism: A Reporter’s Introduction to Social Science Methods*. Indiana University Press, Bloomington (1973)
17. Lahiri, S., Mitra, P., Lu, X.: Informality judgment at sentence level and experiments with formality score. In: Gelbukh, A. (ed.) *CICLing 2011*. LNCS, vol. 6609, pp. 446–457. Springer, Heidelberg (2011). [https://doi.org/10.1007/978-3-642-19437-5\\_37](https://doi.org/10.1007/978-3-642-19437-5_37)
18. Louis, A., Nenkova, A.: A corpus of science journalism for analyzing writing quality. *Dialogue Discourse* **4**(2), 87–117 (2013)
19. Louis, A., Nenkova, A.: What makes writing great? first experiments on article quality prediction in the science journalism domain. *Trans. Assoc. Comput. Linguist.* **1**, 341–352 (2013)
20. McKeown, K., et al.: Predicting the impact of scientific concepts using full-text features. *J. Assoc. Inf. Sci. Technol.* **67**(11), 2684–2696 (2016)
21. Mohammad, S.M., Turney, P.D.: Crowdsourcing a word–emotion association lexicon. *Comput. Intell.* **29**(3), 436–465 (2013)
22. Nakagwe, L.: The persuasive power of personal pronouns in Barack Obama’s rhetoric (2012)
23. Napoli, P.M.: *Audience Evolution: New Technologies and the Transformation of Media Audiences*. Columbia University Press, New York (2011)

24. Nissim, M., Patti, V.: Semantic aspects in sentiment analysis. In: *Sentiment Analysis in Social Networks*, pp. 31–48 (2017)
25. Noth, W.: Self-reference in the media: the semiotic framework. In: *Self-Reference in the Media*, pp. 3–30. Mouton de Gruyter, New York (2007)
26. Okrent, A.: The listicle as literary form. *Univ. Chic. Mag.* **106**(3), 52–53 (2014)
27. Pavlick, E., Nenkova, A.: Inducing lexical style properties for paraphrase and genre differentiation. In: *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 218–224 (2015)
28. Pavlick, E., Tetreault, J.: An empirical analysis of formality in online communication. *Trans. Assoc. Comput. Linguist.* **4**(1), 61–74 (2016)
29. Pitler, E., Nenkova, A.: Revisiting readability: a unified framework for predicting text quality. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 186–195. Association for Computational Linguistics (2008)
30. Sawyer, A.G., Laran, J., Xu, J.: The readability of marketing journals: are award-winning articles better written? *J. Mark.* **72**(1), 108–117 (2008)
31. Strapparava, C., Mihalcea, R.: Semeval-2007 task 14: affective text. In: *Proceedings of the 4th International Workshop on Semantic Evaluations*, pp. 70–74. Association for Computational Linguistics (2007)
32. Tandoc Jr., E.C.: Journalism is twerking? how web analytics is changing the process of gatekeeping. *New Media Soc.* **16**(4), 559–575 (2014)
33. Vaiciukynaite, E., Massara, F., Gatautis, R.: An investigation on consumer sociability behaviour on Facebook. *Eng. Econ.* **28**(4), 467–474 (2017)
34. Zheng, H.-T., Yao, X., Jiang, Y., Xia, S.-T., Xiao, X.: Boost clickbait detection based on user behavior analysis. In: Chen, L., Jensen, C.S., Shahabi, C., Yang, X., Lian, X. (eds.) *APWeb-WAIM 2017*. LNCS, vol. 10367, pp. 73–80. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-63564-4\\_6](https://doi.org/10.1007/978-3-319-63564-4_6)
35. Petz, G., Karpowicz, M., Fuerschuss, H., Auinger, A., Stritesky, V., Holzinger, A.: Computational approaches for mining user’s opinions on the Web 2.0. *Inf. Process. Manag.* **51**(4), 510–519 (2015). <https://doi.org/10.1016/j.ipm.2014.07.011>
36. Petz, G., et al.: On text preprocessing for opinion mining outside of laboratory environments. In: Huang, R., Ghorbani, A.A., Pasi, G., Yamaguchi, T., Yen, N.Y., Jin, Beijing (eds.) *AMT 2012*. LNCS, vol. 7669, pp. 618–629. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-35236-2\\_62](https://doi.org/10.1007/978-3-642-35236-2_62)



# Creative Intelligence – Automating Car Design Studio with Generative Adversarial Networks (GAN)

Sreedhar Radhakrishnan<sup>(✉)</sup> , Varun Bharadwaj<sup>(✉)</sup> ,  
Varun Manjunath<sup>(✉)</sup> , and Ramamoorthy Srinath 

PES University, Bengaluru 560085, India  
sreedhar1895@gmail.com, varunbharadwaj1995@gmail.com,  
varunmanjunath2012@gmail.com,  
ramamoorthysrinath@gmail.com

**Abstract.** In this paper, we propose and implement a system based on Generative Adversarial Networks (GANs), to create novel car designs from a minimal design studio sketch. A key component of our architecture is a novel convolutional filter layer, that produces sketches similar to those drawn by designers during rapid prototyping. The sketches produced are more aesthetic than the ones from standard edge detection filters or gradient operations. In addition, we show that our system is able to generate hitherto unseen perspectives of a car, given a sketch of the car at just a single viewing angle. For extensive training, testing and validation of our system, we have developed a comprehensive, paired dataset of around 100,000 car images (with transparent backgrounds) and their respective sketches. Our work augments human intelligence and creativity using machine learning and deep neural networks. Our system has the significant benefit of reducing the cycle time in the sketch-to-image process which has largely been considered a creative domain. This is achieved by learning to interpret a preliminary sketch drawn by a designer, to generate novel visual designs in a matter of seconds, which may otherwise require considerable time and effort. While the system enhances the productivity of the designer, the machine learning enhanced design visualizations can cut costs during the product prototyping stage. Our system exhibits good impactful potential for the automobile industry and can be easily adapted to industries which require creative intelligence.

**Keywords:** Computational creativity · Generative Adversarial Networks  
Automobile design · Deep learning · Computer vision · Sketching filter

## 1 Introduction

‘A picture is worth a thousand words’, is an idiom that resonates with visual designers across industries. It is extremely relevant today, considering the scale at which product design has engendered innovation in the last decade. One of the most important phases in product design and manufacturing is prototype development. These visualizations are the best way to communicate ideas beyond language barriers. Developing multiple

novel prototypes for stakeholders is an arduous task and one that requires an amalgamation of creativity and tenacity. A designer first channels his creativity in the form of a ‘napkin sketch’, which is a rudimentary sketch that captures the thought process prior to developing prototypes. Although developing a napkin sketch may not take very long, developing multiple novel prototypes is a very time-consuming task. In today’s fast-paced environment, every minute saved would result in reduced cost for an organization. One such industry where innovation in product design results in high quality products and revenue is the automobile industry. Creating a palette of designs before zeroing in on the right design prior to manufacturing is a creative and expensive bottleneck in the production process.

In this paper, we seek to aid artists and car designers by breaking the creative bottleneck during car sketch-to-design translation. Empirical studies and extensive research such as ‘Role of sketching in conceptual design of car styling’ [1] have shown the primary importance of car sketches in both cost reduction and expediting the idea-to-product process.

We propose a system that uses GANs to augment the creativity of designers by generating multiple novel car designs in a matter of seconds. By leveraging semantic information of a given studio sketch, our system also generates car designs in multiple hitherto unseen perspectives. In addition, we show interesting results where a model trained only on car images shows serendipitous and visually appealing results for sketches of bikes and even non-automobile entities such as trees.

## 1.1 Generative Adversarial Networks

The domain of deep learning for computer vision has witnessed rapid progress since 1998, where the work on ‘Gradient-Based Learning Applied to Document Recognition’ [2] discussed Convolutional Neural Networks (CNN). AlexNet [3], that won the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC) further led to prodigious amount of research in CNN. AlexNet architecture, marked the first time when a CNN achieved a top five test error rate (rate at which, given an image, the model does not output the correct label with its top five predictions) of 15.4%.

Generative Adversarial Networks (GANs) [4], a class of machine learning algorithms used in unsupervised learning, introduced in 2014 have shown immense potential in areas of domain adaptation and image translation. GANs introduced a powerful concept where there are two adversarial models, a generator and a discriminator, with the models being trained using a min-max method. In other words, the generator produces images by capturing a data distribution while trying to minimize the classification accuracy of the discriminator. Essentially, it is the generator’s motive to deceive the discriminator while the discriminator tries to maximize its classification accuracy. The discriminator’s role is to classify an input image as an original image or one that is not (generated from the generator). Mathematically, the value function for a discriminator and a generator is given by a simple formulation of cross-entropy and expected values.

The adversarial networks are trained until the generator can satisfactorily produce outputs that the discriminator classifies as real. Although stability and stopping criterion were major concerns, GANs fostered research in the areas of image translation, domain adaptation and various forms of generative tasks.

There has been a lot of work on improving the stability of GANs such as Deep Convolutional Generative Adversarial Networks (DCGANs) [5] and Wasserstein GAN (WGAN) [6]. Coupled Generative Adversarial Networks (CoGAN) [7] introduced an architecture where two GANs are used to learn a joint distribution of multi-domain images. This is achieved through weight sharing in both the generator and the discriminator in the layers where high level features are extracted. Conditional Adversarial Networks [8] not only learn a mapping from input to output domain but also learn a loss function to train this mapping. Conditional Adversarial Networks, thus expect a paired dataset for training. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks (Cycle GAN) [9] employs an inverse mapping such that the pairing is obtained only if an image  $X_i$  in domain  $X$  produces an image  $Y_i$  in domain  $Y$  such that translation of  $Y_i$  from domain  $Y$  back to domain  $X$  produces  $X_i$ . This mapping is useful in cases when a paired dataset is not available.

While Conditional Adversarial Networks perform well in image translation tasks, the architecture expects a paired dataset, which in most cases either does not exist or is difficult to acquire. Hence, we developed a paired dataset of around 100,000 car images (with a transparent background) along with their respective sketches. The dataset will not only help us leverage Conditional GANs in our system but will also be an aid for researchers worldwide supporting further advancement and innovation in related fields. Researchers can contact the authors of this paper to obtain the dataset. The dataset will also be open-sourced for public domain access.

## 1.2 Related Work

Computational creativity and its related fields have attracted a lot of attention in the research community. There has been work in developing a standard pipeline for sketch based modelling [10] which encompasses methods and techniques to enable users to interact with a computer program through sketching. The three main stages are input, filter and interpret. Later, image acquisition and visual rules are used for preventing over-sketching and similar defects. Although the above paper does not use machine learning for modelling input sketches, the goal of improving human productivity remains the same.

Work on design generation in fashion and clothing industry, such as Visually-Aware Fashion Recommendation and Design with Generative Image Models [11] use feature representations from CNNs and Siamese neural networks to make a fashion recommendation system. Further, in unison with GANs, the same system is used in a generative manner, where new designs are fabricated conditioned on the given user profile and product category. However, it is not a sketch-based modelling system.

There has also been prior work in interpreting human sketches for image retrieval in large databases [12]. Humans can visualize and draw objects with a certain degree of accuracy. Thus, sketch modelling based image retrieval systems have the potential to perform better than traditional retrieval systems. While our work focuses on deep learning based rapid prototype generation from a sketch, human sketch based information retrieval provides an intuitive interface for fast image retrieval reflecting human imagination.

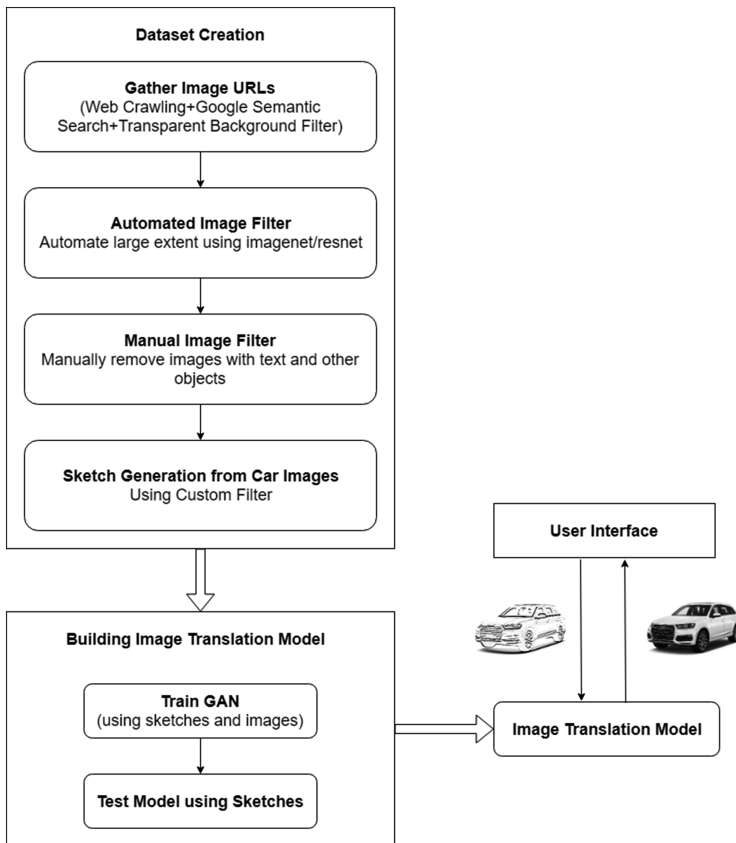


A recent work presents an implementation where GANs are used in a user interactive program to generate realistic images constrained by a natural image manifold [13]. The object can be distorted and reshaped and the GANs automatically adjust the output keeping all edits as realistic as possible. The manifold of natural images is used as a constraint on the output of various image manipulation operations, this makes sure the result always lies in the learned manifold.

In our paper, we utilize GANs for learning the distribution across the training samples and creating new samples in the specific domain of automobile design, especially cars. For the training of conditional GANs, we fabricate artificial sketches using a novel convolutional-morphological filter.

## 2 Proposed System

Our system, as shown in Fig. 1, essentially has three phases: dataset creation, machine learning model training and developing a user interface for the designer. Within the dataset creation phase, there are two core components, first being, the development of a



**Fig. 1.** Proposed GAN based car sketch image translation system.

comprehensive dataset of around 100,000 car images and the second being the generation of sketches using a new filter that we have engineered for the task.

In the model training phase, we have trained 12 machine learning models through which our system generates novel car designs covering various colour combinations and unseen car angles. The system also has a web based interface where the designer can upload sketches, select colours and perspectives and view the design visualizations.

### 3 Dataset

The construction of a comprehensive dataset involved three main steps:

1. Identifying and mining transparent car images.
2. Machine learning based removal of images with abnormal artifacts (undesirable objects or symbols). This was implemented using ImageNet [14] and ResNet50 [15]



**Fig. 2.** Snapshot of sample images in the dataset.

for image recognition and for classifying and filtering out images with entities that are not cars.

3. The remaining images with abnormal artifacts in the dataset were eliminated through manual inspection, which required extensive effort. About 60% of the time in the dataset collection phase was spent on manual inspection.

The multiple colours, angles and consistent transparent backgrounds across all car images in the dataset ensured that the dataset formed the base of the system. The system expects high quality, clean data with no background inconsistencies or noise for ensuring successful training of the machine learning models. The developed dataset satisfies all the above conditions and thus forms a substrate for subsequent steps in developing our system. The quality and variety of images satisfies our requirements to get paired datasets of same and different perspectives of cars for training in the next phase of the pipeline. A snapshot of the car image dataset is shown in Fig. 2.

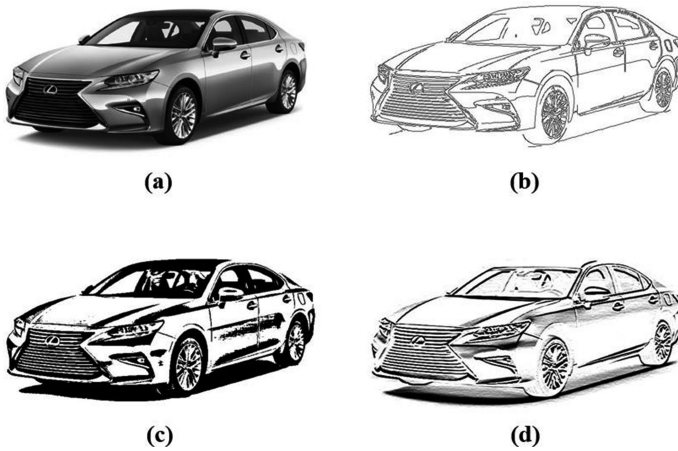
## 4 The BiSECT Sketching Filter

Generally, if an outline is required, the standard approaches are gradient operators or edge filters. Many famous algorithms appear in standard image processing and computer vision literature. Notable ones which produce appreciable results are the Marr-Hildreth filter, Laplacian operator, Difference of Gradients (DoG), Sobel filter, Scharr filter, Canny edge operator [16] and the Extended Difference of Gaussians (XDoG) [17]. Each of these have their own locality, sensitivity and orientation based advantages and disadvantages. However, for our use case, none of these standard techniques produced outputs that capture the artistic nuances of a human sketch.

Hence, our goal was to create a filter that produces sketches similar to those drawn by humans for training the system with a paired dataset. The filters were applied on car images such as Fig. 3(a) and the outputs were analyzed. As shown in Fig. 3(b), while the Canny edge detector is able to identify edges, it lacks the stroke detail of sketches. The output from XDoG, on the other hand, as shown in Fig. 3(c), captures too many details, beyond what we expect in a simple sketch. The output of XDoG looks synthetic and does not have the sketch texture one would expect from a pencil drawn sketch. There has been prior work on simulating sketches computationally such as ‘Automatic Generation of Pencil-sketch Like Drawings’ [18], using gradient estimation on smoothed images.

The sketching filter we propose is shown to perform well for car images, leading to a ‘napkin sketch’ kind of effect as shown in Fig. 3(d). These sketches are similar to the preliminary drawings by designers or artists in the field of design and manufacturing.

We propose a new convolutional filter, the “BiSECT” sketching filter, which is an abbreviation for Bidirectional Sobel Enhanced with Closed Thresholds, which conveys the inner working of the filter concisely. The filter can be thought of as a “bisecting” operation that extracts the sketch artifacts from a given image.



**Fig. 3.** Outputs of various filters. (a) shows the input image. (b), (c) and (d) show the outputs of the Canny, xDoG and BiSECT operators respectively.

#### 4.1 Formulation of the BiSECT Sketching Filter

The BiSECT sketching filter was developed using a combination of existing standard convolutional kernels and morphological operations. A level of smoothing Gaussians and pixel-wise logical operations are used for combining and retaining the appropriate edges across different intermediate outputs. The filter was engineered as follows:

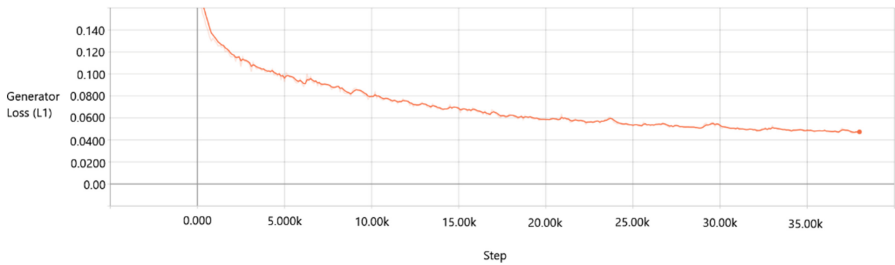
1. We use a combination of Sobel family operators because a single Sobel-Feldman convolutional filter restricts itself to finding edges in a particular orientation.
2. Combining the Sobel filter family outputs using a cascade of bitwise-OR and bitwise-AND operations allowed retaining important edges and having finer control over the final sketch.
3. After this series of convolution operations, we use a threshold layer to discard all the unnecessary levels of gray all over the intermediate outputs. We explored Otsu's thresholding and adaptive thresholding techniques. After some experimentation, an appropriate global thresholding was satisfactory.
4. Finally, a 'morphological close' operation is performed on the thresholded output to coalesce all slightly disconnected true edges with structural elements of sizes  $(8*1)$  and  $(1*8)$  for both vertical and horizontal axes.

After aggregating and auditing our image dataset, each image from the 100,000 and odd car images was paired with its corresponding output from the above filter. This forms a standard paired dataset, which can be fed into conditional adversarial networks.

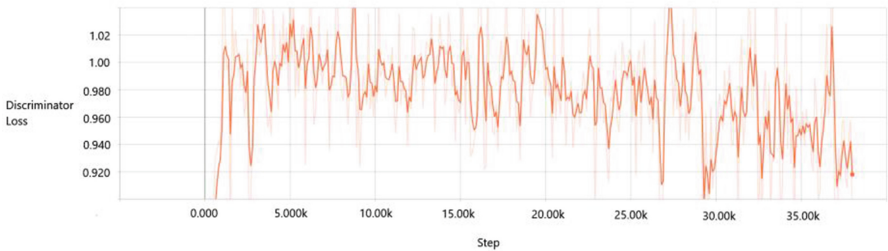
## 5 Experiments and Results

Experiments were run to train the system to generate novel designs by learning a pixel-level semantic mapping from sketches. The goal was to train the system to learn from minimal amount of data. This is because we wanted to adapt the system to the real world wherein the system will be continuously subjected to a plethora of sketches with variations.

The system learns the mapping from minimal data and generates high quality designs on significant amount of test data making the system robust and reliable. In the experiments below, we configured the model to train for 200 epochs on 200 paired images. The graphs in Figs. 4 and 5 show the generator and discriminator losses against the number of steps in training of the GANs.



**Fig. 4.** Generator Training Loss Visualization

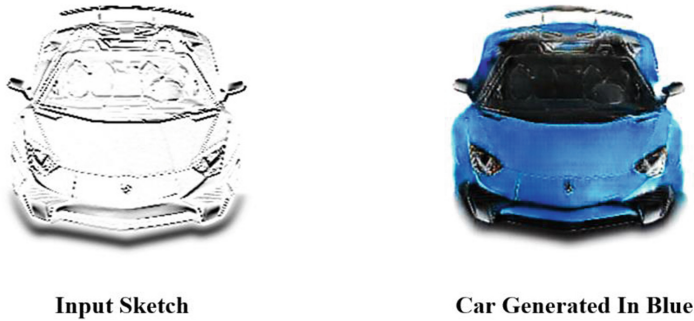


**Fig. 5.** Discriminator Training Loss Visualization

### 5.1 Single Colour Car Generation

We trained the system to generate designs in red, blue, white, and black as well as grayscale. We can observe that the system is able to generate state of the art car prototypes from preliminary sketches. The system correctly identifies regions in the sketch that need to be filled with a respective colour and texture. The system consistently is able to identify and augment even subtle details such as headlamp and tyre correctly.

Each of the input sketches visualize a diverse range of car models, shapes and sizes. The models correctly interpret the sketch, such as in Fig. 6 where the system has correctly distinguished between the skin and the seat of the convertible itself (Fig. 7).



**Fig. 6.** Translation of sketch to a blue prototype (Color figure online)



**Fig. 7.** Translation of sketch to a black prototype



**Fig. 8.** Dual colour output: Red and Yellow (Color figure online)

## 5.2 Mixed Colour Car Generation

In addition, we trained the system to generate prototypes consisting of an amalgamation of multiple colours. The combinations we trained the system on were red and yellow, black and white, blue and gray as well as a four colour mix consisting of black, blue, reddish-brown and gray (Figs. 8 and 9).

**Input Sketch****Car Generated In Blue and Gray****Fig. 9.** Dual colour output: Blue and Gray (Color figure online)

### 5.3 Multiple Design Visualizations

Our system enables designers to visualize multiple prototypes for a given sketch. Viewing multiple prototypes in juxtaposition allows the designer to make faster decisions regarding the final design. In addition, the system enables the designer to visualize or develop multiple prototypes quickly resulting in significant saving in cost, time and effort.

In the example shown in Fig. 10, we have generated three different car prototypes using grayscale, single color and multiple color combination models from a single sketch. While all the three models largely resemble each other structurally, each prototype has its own aesthetic appeal.

**Car Generated In Blue****Car Generated in Grayscale****Car Generated in Black, Blue, Gray and Brown****Fig. 10.** Outputs from multiple models

With minimal training, the system is able to embellish the sketch to produce prototypes in multiple colours across various shapes and sizes, as shown in the figures in this section.

## 6 Artist Sketch Experiment

We obtained a shaded pencil sketch drawn by an artist. We used the GrabCut algorithm [19] to segment out the design. We then applied the “BiSECT” filter and tested the result on our machine learning models. The initial pencil shaded sketch image took considerable amount of time and our system interpreted the sketch and generated multiple prototypes in less than a minute.

In this particular case, the artist spent 8 h spread over a period of 5 days in creating the initial sketch on paper. The sketch in Fig. 11 is evidently more detailed than a napkin sketch. Our system is capable of processing sketches that do not require extensive amount of time and effort to draw. Thus, with a preliminary sketch such as those shown in Sect. 5, our system will generate multiple designs and the artist need not spend a long time to add extensive details.



**Pencil Shaded Sketch**



**Generated Blue Prototype**



**Generated Black Prototype**

**Fig. 11.** Testing on an artist’s sketch



## 7 Generating Hitherto Unseen Perspectives of a Car

We have made some interesting headway in using the system to predict and generate how a car will look from various hitherto unseen angles and perspectives. This is done purely based on the input sketch drawn by the designer in a single angle. We thus trained the system to learn a pixel level mapping across different angles.

While the generated prototypes did have some artifacts, particularly due to lack of paired images across multiple angles, the semantic information of the shape was largely captured by the models. We are currently working on improving the accuracy of the models and generating an interactive 3D prototype of a car purely from the single angle 2D sketch input by the designer.

We developed models that visualize the car's left angular front view, left view, left angular rear view and front view. By flipping these views, we obtained right angular

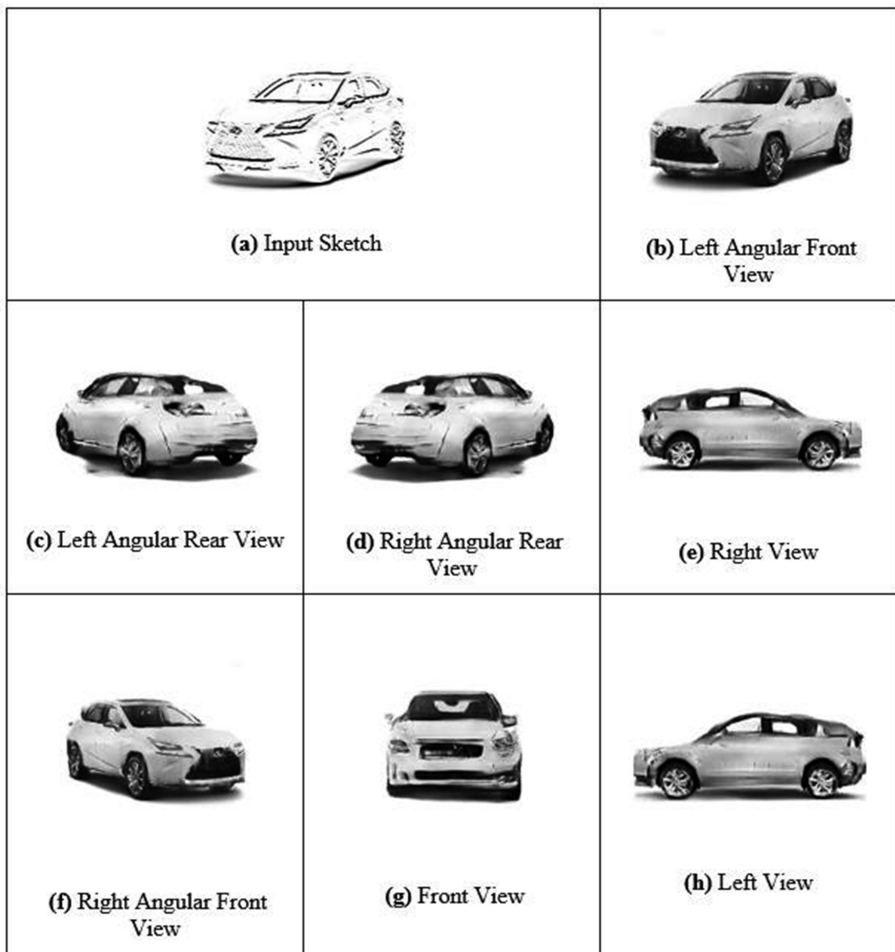


Fig. 12. Generated unseen car perspective visualizations from a single angle sketch

rear view, right view and right angular front view. The input sketch and the generated views and perspectives are shown in Fig. 12.

The rear views in particular have noticeable artifacts. However, the system was able to estimate the probable shape of the car with no information about the same in the input sketch.

## 8 Experiments on Unseen Domains

We tested our system on sketches from completely unseen domains. We first tested the system on a bike sketch (Fig. 13), as it still falls under the automobile family. The generated image suggests that the system can produce satisfactory results for other types of vehicles as well. We then tested the system on a completely unrelated domain by feeding it a sketch of a tree (Fig. 14), and the result resembles a tree in autumn. The fact that our model was not exposed to either of these domains during training phase shows that our system can easily be adapted for most industries that require creativity and design capabilities.



Input Bike Sketch



Generated Output

**Fig. 13.** Output of grayscale model on bike sketch



Input Sketch



Generated Output

**Fig. 14.** Output of multi-colour model on tree sketch

## 9 Conclusion

In this paper, we have made three major contributions. First, a Generative Adversarial Network based system that converts car sketches to multiple prototypes having different colours and perspectives. Second, a novel convolutional sketching filter that produces sketches similar to those drawn by designers during rapid prototyping and third, a comprehensive paired dataset of about 100,000 car images (with transparent background) and their respective sketches.

We also showed certain interesting test results wherein the system was easily able to adapt to other domains that require creative intelligence and design capabilities. In most cases, human thought process is based on a recent set of prior learnings, experiences and observations. Using the proposed system to generate multiple novel combinatorial designs, the creativity of designers is augmented beyond standard creative bottlenecks.

## 10 Learnings and Future Enhancements

Our system qualitatively derives car designs from sketches by augmenting human creativity and intelligence using the fields of digital image processing, computer vision and (deep) machine learning. We believe, some of the possible future enhancements of our work can include:

- 3D rendering of a car from a studio sketch given a specific perspective of the car. In Sect. 7, we showed interesting results of the system generating visual prototypes in multiple angles from a single angle input sketch. We can extend this concept by feeding the multiple generated 2D perspectives of a car to a 3D image stitching algorithm to produce a 3D rendered car from a single angle 2D car sketch.
- Producing augmented reality, virtual reality and mixed reality experiences of the generated car prototype. We can extend 3D rendering of a sketch to producing immersive experiences. Essentially, the designer will input a 2D sketch of a system and (contingent on AR/VR hardware availability) the designer can view how the generated car will look like upon production using virtual reality or other similar visualization forms.
- Generating multiple perspectives using two sketches instead of a single sketch. Currently the back side of the car is approximated as no data is available in the input sketch. To overcome this drawback, sketches of left angular front view and left angular rear view can be used together to provide almost all the required details needed to generate all perspectives of a car.
- Making the system flexible by enabling finer control of the desired generated prototype. Currently our system allows the user to choose from a certain set of colours or colour combinations. We can enhance this feature by providing the artist more granular control by allowing selection of a particular colour for a given car section. As an example, this will allow the artist to choose a different colour shade for the door and the front of the car.

- Using a crowdsourcing based approach to obtain evaluation metrics for the car prototypes obtained from the various models. Preferably, obtain metrics from a group of artists/experts and use these metrics as a means for improving certain models.

## References

1. Bouchard, C., Aoussat, A., Duchamp, R.: Role of sketching in conceptual design of car styling. *J. Des. Res.* **5**(1), 116–148 (2006)
2. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. In: *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324 (1998)
3. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: *Proceedings of the 25th International Conference on Advances in Neural Information Processing Systems*, vol. 1, pp. 1097–1105 (2012)
4. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: *Proceedings of the 27th International Conference on Advances in Neural Information Processing Systems*, pp. 2672–2680 (2014)
5. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434* (2015)
6. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein GAN. *arXiv preprint arXiv:1701.07875* (2017)
7. Liu, M.-Y., Tuzel, O.: Coupled generative adversarial networks. In: *Proceedings of the 29th International Conference on Advances in Neural Information Processing Systems*, pp. 469–477 (2016)
8. Isola, P., Zhu, J.-Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004v2* (2017)
9. Zhu, J.-Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint arXiv:1703.10593v4* (2018)
10. Olsen, L., Samavati, F.F., Sousa, M.C., Joaquim, A.: Sketch based modeling: a survey. *Comput. Graph.* **33**(1), 85–103 (2009)
11. Kang, W.C., Fang, C., Wang, Z., McAuley, J.: Visually-aware fashion recommendation and design with generative image models, *arXiv:1711.02231v1* (2017)
12. Parui, S., Mittal, A.: Similarity-invariant sketch-based image retrieval in large databases. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8694, pp. 398–414. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-10599-4\\_26](https://doi.org/10.1007/978-3-319-10599-4_26)
13. Zhu, J.-Y., Krähenbühl, P., Shechtman, E., Efros, A.A.: Generative visual manipulation on the natural image manifold. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *ECCV 2016*. LNCS, vol. 9909, pp. 597–613. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46454-1\\_36](https://doi.org/10.1007/978-3-319-46454-1_36). arXiv:1609.03552v2
14. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database, *Computer Vision and Pattern Recognition*, Department of Computer Science, Princeton University, USA (2009)
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Computer Vision and Pattern Recognition*, *arXiv:1512.03385* (2015)
16. Canny, J.: A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **Pami-8**(6), 184–203 (1987)

17. Winnemöllera, H., Kyprianidis, J.E., Olsen, S.C.: XDoG: an eXtended difference-of-Gaussians compendium including advanced image stylization. *Comput. Graph.* **36**(6), 720–753 (2012)
18. Zhou, J., Li, B.: Automatic generation of pencil-sketch like drawings from personal photos. In: *IEEE International Conference on Multimedia and Expo, ICME, Amsterdam, Netherlands* (2005)
19. Rother, C., Kolmogorov, V., Blake, A.: “GrabCut” — interactive foreground extraction using iterated graph cuts. *SIGGRAPH ACM Trans. Graph.* **23**, 309–314 (2014)

**MAKE-Text**



# A Combined CNN and LSTM Model for Arabic Sentiment Analysis

Abdulaziz M. Alayba<sup>(✉)</sup>, Vasile Palade, Matthew England, and Rahat Iqbal

School of Computing, Electronics and Mathematics, Faculty of Engineering,  
Environment and Computing, Coventry University, Coventry, UK  
Alaybaa@uni.coventry.ac.uk,  
{Vasile.Palade,Matthew.England,R.Iqbal}@coventry.ac.uk

**Abstract.** Deep neural networks have shown good data modelling capabilities when dealing with challenging and large datasets from a wide range of application areas. Convolutional Neural Networks (CNNs) offer advantages in selecting good features and Long Short-Term Memory (LSTM) networks have proven good abilities of learning sequential data. Both approaches have been reported to provide improved results in areas such image processing, voice recognition, language translation and other Natural Language Processing (NLP) tasks. Sentiment classification for short text messages from Twitter is a challenging task, and the complexity increases for Arabic language sentiment classification tasks because Arabic is a rich language in morphology. In addition, the availability of accurate pre-processing tools for Arabic is another current limitation, along with limited research available in this area. In this paper, we investigate the benefits of integrating CNNs and LSTMs and report obtained improved accuracy for Arabic sentiment analysis on different datasets. Additionally, we seek to consider the morphological diversity of particular Arabic words by using different sentiment classification levels.

**Keywords:** Arabic sentiment classification · CNN · LSTM  
Natural Language Processing(NLP)

## 1 Introduction

In the past decade, social media networks have become a valuable resource for data of different types, such as texts, photos, videos, voices, GPS reading, etc. The explosion of data we experience today in many areas has led researchers in data science to develop new machine learning approaches. There were improvements in different areas, such as: Neural Networks, Deep Learning, Natural Language Processing (NLP), Computer Vision, Geolocation Detection, etc. Sentiment Analysis is one of the topics that attracted much attention from NLP and machine learning researchers. Sentiment analysis deals with the texts or the reviews of people that include opinions, sentiments, attitudes, emotions, statements about products, services, foods, films, etc. [1].

There is a certain sequence of steps to perform supervised learning for sentiment analysis, i.e., converting the text to numeric data and mapping with labels, performing feature extraction/selection to train some classifiers using a training dataset and then estimate the error on the test dataset. Sentiment analysis has various analytic levels that are: document level, sentence level, aspect level [2,3], word level, character level [4] and sub-word level [5]. Deep neural networks have shown good performance in this area in [6–8].

We have also obtained good results on using deep neural networks for sentiment analysis on our own dataset, an Arabic Health Services dataset, reported in [9,10]. We have obtained an accuracy between 0.85 and 0.91 for the main dataset in [9] using SVM, Naïve Bayes, Logistic Regression and CNNs. Also, using merged lexicon with CNNs and pre-trained Arabic word embedding, the accuracy for the main dataset was improved to 0.92, and for a Sub-dataset (as described in [10]) the obtained accuracy was between 0.87 and 0.95.

The sentiment analysis approach in this paper is a combination of two deep neural networks, i.e., a Convolutional Neural Network (CNN) and a Long Short Term Memory (LSTM) network. Kim [6] defined CNNs to have convolving filters over each input layer in order to generate the best features. CNNs have shown improvements in computer vision, natural language processing and other tasks. Athiwaratkun and Kang [11] confirmed that the CNN is a powerful tool to select features in order to improve the prediction accuracy. Gers et al. [12] showed the capabilities of LSTMs in learning data series by considering the previous outputs.

This paper first presents some background on deep neural networks and Arabic sentiment classification in Sect. 2. Section 3 describes the Arabic sentiment datasets we use. Section 4 illustrates the architecture of the proposed merged CNN-LSTMs Arabic sentiment analysis model. The results of the sentiment classification using the model will be presented in Sect. 5, which will be compared with other results. Section 6 concludes the study and the experiments, and outlines the future work.

## 2 Background and Related Work

Deep neural network models have had great success in machine learning, particularly in various tasks of NLP. For example, automatic summarization [13], question answering [14], machine translation [15], words and phrases distributed representations [16], sentiment analysis [6] and other tasks. Kim [6] proposed a deep learning model for sentiment analysis using CNNs with different convolutional filter sizes. Wang et al. [17] applied an attention-based LSTMs model for aspect-level sentiment analysis.

Arabic sentiment analysis has become a research area of interest in recent years. Abdul-Mageed et al. [18] studied the effect at sentence level on the subjectivity and sentiment classification for Modern Standard Arabic language (MSA) using an SVM classifier. Shoukry and Rafea [19] applied SVM and Naïve Bayes at sentence level for sentiment classification using 1000 tweets. Abdulla et al. [20] compared corpus-based and lexicon-based approaches for sentiment analysis.



Abdulla et al. [21] addressed the challenges of lexicon construction and sentiment analysis. Badaro et al [22] created a large Arabic sentiment lexicon using English-based linking to the ESWN lexicon and WordNet approach. Duwairi et al. [23] collected over 300,000 Arabic tweets and labeled over 25,000 tweets using crowdsourcing. Al Sallab et al. [24] employed three deep learning methods for Arabic sentiment classification. Ibrahim et al. [25] showed sentiment classifications for MSA and the Egyptian dialect using different types of text data such as tweets, product reviews, etc. Dahou et al. [26] reported on the usage of Arabic pre-trained word representation with CNN increased sentiment classification performance. Tartir and Abdul-Nabi [27] concluded that a semantic approach leads to good sentiment classification results even when the dataset size is small. El-Beltagy et al. [28] enhanced the performance of a sentiment classification using a particular set of features.

### 3 Datasets

There is a lack of Arabic sentiment datasets in comparison to English. In this paper, four datasets (where one is a subset of another) will be used in the experiments. Each used only two sentiment classes, i.e., Positive and Negative sentiment.

#### 3.1 Arabic Health Services Dataset (Main-AHS and Sub-AHS)

This is our own Arabic sentiment analysis dataset collected from Twitter. It was first presented in [9] and it has two classes (positive and negative). The dataset contains 2026 tweets and it is an unbalanced dataset that has 1398 negative tweets and 628 positive tweets. We call this dataset **Main-AHS**, and we selected a subset of this dataset, called **Sub-AHS**, which was introduced in [10]. The **Sub-AHS** dataset contains 1732 tweets, with 502 positive tweets and 1230 negative tweets.

#### 3.2 Twitter Data Set (Ar-Twitter)

The authors of [20] have manually built a labeled sentiment analysis dataset from Twitter using a crawler. The dataset contains 2000 tweets with two classes (Positive and Negative) and each class contains 1000 tweets. The dataset covered several topics in Arabic such as politics, communities and arts. There are some tweets in the available online dataset are missing and, hence, the used size of the dataset in our experiments is 975 negative tweets and 1000 positive tweets.

#### 3.3 Arabic Sentiment Tweets Dataset (ASTD)

The authors of [29] presented a sentiment analysis dataset from Twitter that contains over 54,000 Arabic tweets. It has four classes (objective, subjective positive, subjective negative, and subjective mixed). However, in this paper only two classes (positive and negative) will be used and the numbers of negative and positive tweets are 1684 and 795 respectively, giving a total of 2479 tweets.

## 4 CNN-LSTM Arabic Sentiment Analysis Model

The fundamental architecture of the proposed model is shown in Fig. 1 and it outlines the combination of the two neural networks: CNN and LSTM. There are no accurate tools for preprocessing Arabic text, especially non Standard Arabic text like most of the tweets. There are many forms for a single word in Arabic, for example Arabic words are different based on gender, the tenses of the verbs, the speaker voices, etc. [31]. Table 1 shows several examples of a single Arabic verb (and it has more other forms), the pronunciation of the word as Buckwalter translation [30] and the description of the verb's type.

**Table 1.** Some examples of multiple forms of a single Arabic verb

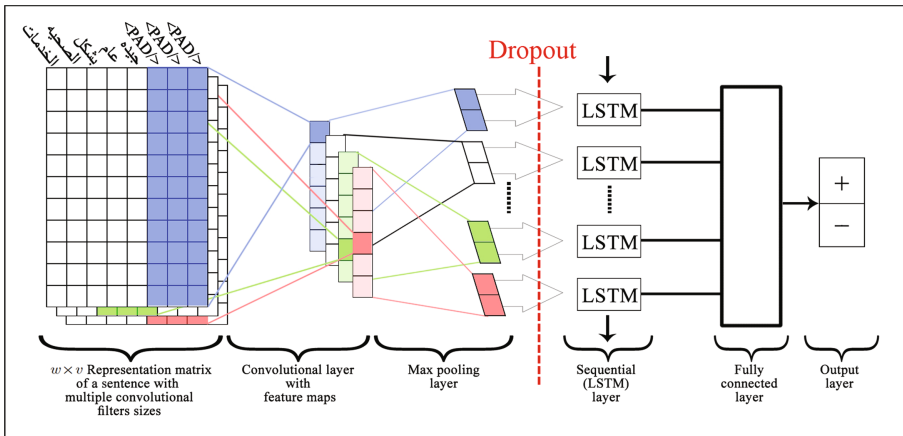
Arabic word	Buckwalter Arabic Encoding	Word type
فعل	fEl	Masculine Verb - past tense for singular
فعلت	fElIt	Feminine Verb - past tense for singular
يفعل	yfEl	Masculine Verb - present tense for singular
تفعل	tfEl	Feminine Verb - present tense for singular
يفعلان	yfElAn	Masculine Verb - present tense for dual
تفعلان	tfElAn	Feminine Verb - present tense for dual
يفعلون	yfElwn	Masculine Verb - present tense for plural
يفعلن	yfEln	Feminine Verb - present tense for plural

There will be three different levels of sentiment analysis for each proposed dataset. The reason of using different levels is to try to expand the number of features in short tweets and to deal with many forms of a single word in Arabic. This is an example tweet «الخدمات الصحية بشكل عام جيدة.» and the English translation of this tweet is 'Health services are generally good'. The levels are as follows.

**Character Level (Char-level)**, by converting the sentence into characters instead of words such as [ 'ا', 'ل', 'خ', 'د', 'م', 'ا', 'ت', 'ا', 'ل', 'ص', 'ح', 'ي', 'ن', 'ي', 'د', 'ي', 'ج', 'م', 'ا', 'ع', 'ل', 'ك', 'ش', 'ب', 'ه', 'ه', 'ر', 'e', 'a', 'l', 't', 'h', 's', 'e', 'r', 'v', 'i', 'c', 'e', 's', 'a', 'r', 'e', 'g', 'e', 'n', 'e', 'r', 'a', 'l', 'l', 'y', 'g', 'o', 'o', 'd' ]. The (Char-level) for the English example is [ 'H', 'e', 'a', 'l', 't', 'h', 's', 'e', 'r', 'v', 'i', 'c', 'e', 's', 'a', 'r', 'e', 'g', 'e', 'n', 'e', 'r', 'a', 'l', 'l', 'y', 'g', 'o', 'o', 'd' ]. At this level, the number of features is increased, such as in the above example, the number of characters is 24 for the Arabic example, and each letter represents one feature.

The second level is **Character  $N$ -Gram Level (Ch5gram-level)**: where we measure the length of all the words in each dataset and we calculate the average length of words (which is five characters for all the different datasets). Then, we split any word that, has more than the average number into several sub-words. Whereas, any word that consist of the same average number of characters or less will be kept as it is. The average word's length for each dataset is five characters and a 5-gram example is [الصحيه', 'الصحي', 'خدمات', 'نخدما', 'الخدم', 'جيده', 'عام', 'بشكل']. The (Ch5gram-level) for the English example is ['Healt', 'ealth', 'servi', 'ervic', 'rvic', 'vices', 'are', 'gener', 'enera', 'neral', 'erall', 'rally', 'good']. This level can be useful in order to deal with many forms of Arabic words, especially for words with more than five letters. Also, the number of the features is expanded in this level too. The third level is **Word Level (Word-level)**, where the sentence is divided into words using the space as splitter, such as [جيده', 'عام', 'بشكل', 'الصحيه', 'الخدمات].

The (Word-level) for the English example is ['Health', 'services', 'are', 'generally', 'good']. This level is the most commonly chosen option in the field of sentiment analysis.



**Fig. 1.** A combined CNN-LSTM model architecture for sentiment analysis, with an example of Arabic tweet using Word-Level.

The input data layer is represented as a fixed-dimension matrix of different vector embeddings based on different sentiment analysis levels. Each sentiment analysis level has different tokens, for example, in the Char-level the token is a single character. In the Ch5gram-level, the token is a whole word if the length of the word is five characters or less. Also, the token for any words that has more than five letters is split into five gram character like in the Ch5 Gram-level example from above. In the Word-level, the tokens are based on the words in each tweet. Each token is represented as a fixed-size vector in the input matrix. Then,

the multiple convolutional filters slide over the matrix to produce a new feature map and the filters have various different sizes to generate different features. The Max-pooling layer is to calculate the maximum value as a corresponding feature to a specific filter. The output vectors of the Max-pooling layer become inputs to the LSTM networks to measure the long-term dependencies of feature sequences. The output vectors of the LSTMs are concatenated and an activation function is applied to generate the final output: either positive or negative.

#### 4.1 Input Layer

This is the first layer in the model and it represents each tweet as a row of vectors. Each vector represents a token based on the the sentiment analysis level used. Each different level has a different token to be embedded, such as in the Char-level, each character in the tweet will be represented into a specific vector with a fixed size of 100. Each word in the tweet, which is one token in the Word-level is embedded into a vector with length of 100 and that is the same with each token in the Ch5gram-level. This layer is a matrix of size  $w \times v$ , where  $v$  is the length of the vector and  $w$  is the number of tokens in the tweets. The value of  $w$  is the maximum length of a tweet. Any tweet that contains less than the maximum number of tokens in the tweet will be padded with  $\langle Pad \rangle$  to have the same length with the maximum tweet length. For instance, the maximum length of tweets with the character level in the Main-AHS dataset is 241 tokens and any tweets that have less than the maximum number will be padded to 241 to get the same length. Each matrix in the Character level in the Main-AHS dataset has the size of  $241 \times 100$ .

#### 4.2 Convolutional Layer

Each input layer contains a sequence of vectors and it is scanned using a fixed size of filter. For example, we used the filter size 3 for Word-level to extract the 3-gram features of words. Also, we used the filter size 20 in the Char-level and the filter size 10 in the Ch5gram-level. The filter strides or shifts only one column and one row over the matrix. Each filter detects multiple features in a tweet using the *ReLU* [32] activation function, in order to represent them in the feature map.

#### 4.3 Max-Pooling Layer

After the Convolutional layer, the Max-pooling layer minimizes and down-samples the features in the feature map. The *max* operation or function is the most commonly used technique for this layer and it is used in this experiment. The reason of selecting the highest value is to capture the most important feature and reduce the computation in the advanced layers. Then the dropout technique is applied to reduce overfitting with the dropout value is 0.5.

#### 4.4 LSTM Layer

One of the advantages of the LSTMs is the ability of capturing the sequential data by considering the previous data. This layer takes the output vectors from the dropout layer as inputs. This layer has a set number of units or cells and the input of each cell is the output from the dropout layer. The final output of this layer have the same number of units in the network.

#### 4.5 Fully Connected Layer

The outputs from LSTMs are merged and combined in one matrix and then passed to a fully connected layer. The array is converted into a single output in the range between 0 and 1 using the fully connected layer, in order to be finally classified using *sigmoid* function [33].

### 5 Experiments and Results

These experiments aimed to utilize a very deep learning model using a combination of CNN and LSTM. The learning performance of the model will be measured using the accuracy of the classifier [34].

$$Acc = \frac{(TP + TN)}{(TP + TN + FP + FN)}. \quad (1)$$

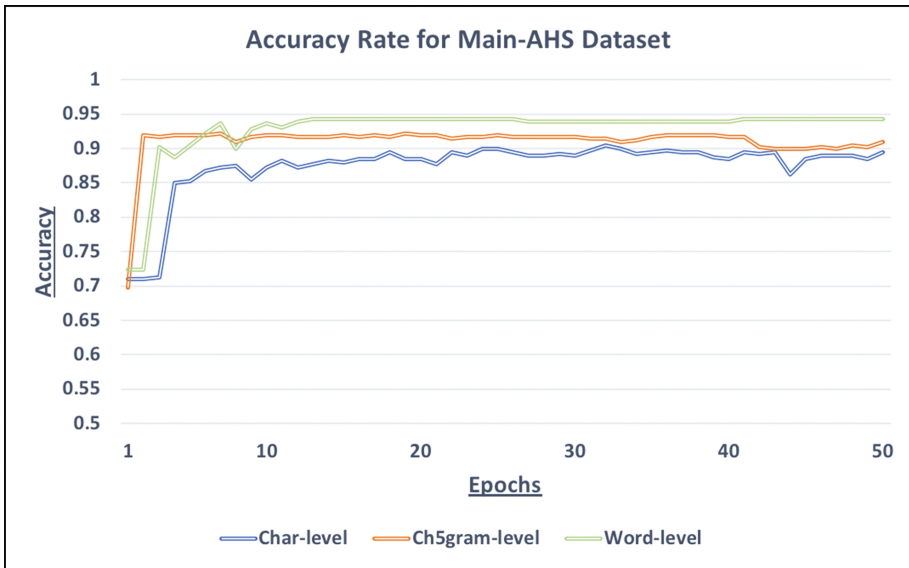
Here,  $TP$  is the number of tweets that are positive and predicted correctly as positive,  $TN$  is the number of tweets that are negative and predicted correctly as negative,  $FP$  is the number of tweets that are negative but predicted incorrectly as positive, and  $FN$  is the number of tweets that are positive but predicted incorrectly as negative.

**Table 2.** Accuracy comparison of the proposed method with different sentiment levels and other models for the same datasets.

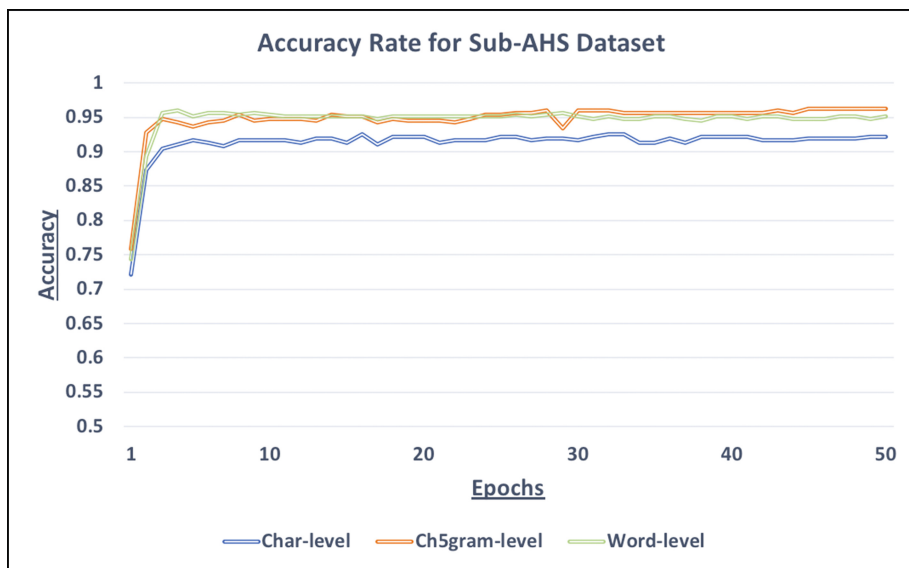
Sentiment level	Main-AHS	Sub-AHS	Ar-Twitter	ASTD
Char-level	0.8941	0.9164	0.8131	0.7419
Ch5gram-level	0.9163	<b>0.9568</b>	0.8283	<u>0.7762</u>
Word-level	<b>0.9424</b>	0.9510	<b>0.8810</b>	0.7641
<i>Alayba et al.</i> [10]	0.92	0.95		
<i>Dahou et al.</i> [26]			85.01	<b>79.07</b>
<i>Abdulla et al.</i> [20]			87.20	

All the experiments using different datasets and sentiment analysis levels use the same size of the training and test datasets. The size of the training set is 80% of the whole dataset, and the test set contains the remaining 20%

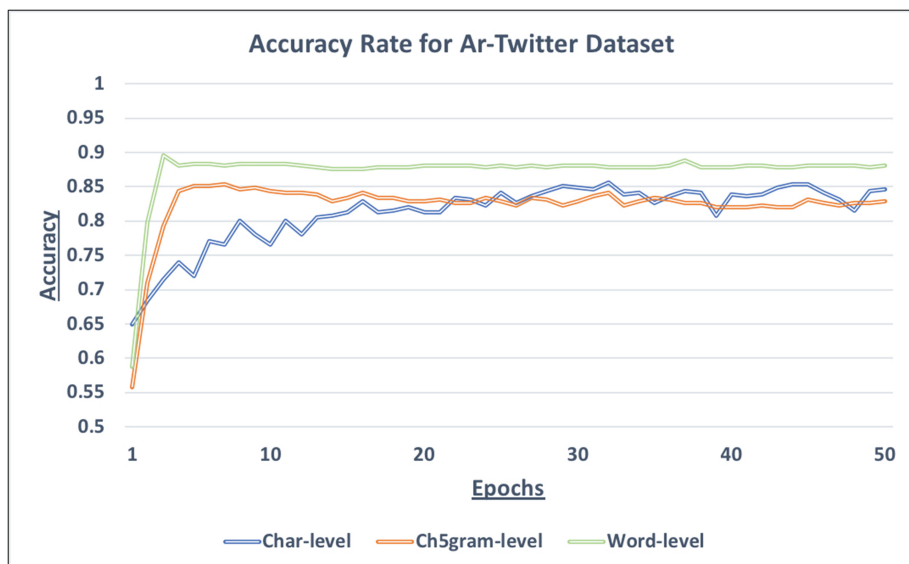
of the dataset. The model is trained using the training set and then the test set is used to measure the performance of the model. The number of epochs is 50 for all the experiments. Table 2 shows the accuracy results in the 50 epochs for the four datasets using different sentiment levels. The best accuracy results for the three different used levels are identified by underlining the best results. Also, Table 2 compares the results of our model with the results published in other papers. It is clear from Table 2 that the proposed model improved the performance of sentiment classification in three datasets: Main-AHS, Sub-AHS, and Ar-Twitter, but it is lower than [26] for the ASTD dataset model (by only a small margin). Figures 2, 3, 4 and 5 illustrate the accuracies on different datasets over 50 epochs. Each line represents different sentiment analysis level. Char-level generally has the lowest accuracy results in the different datasets compared with the other levels, but for Ar-Twitter, it is better than the accuracy obtained on the Ch5gram-level after 23 epochs. Word-level achieves the best accuracy results for the Main-AHS and Ar-Twitter datasets and it has similar results with Ch5gram-level for the Sub-AHS.



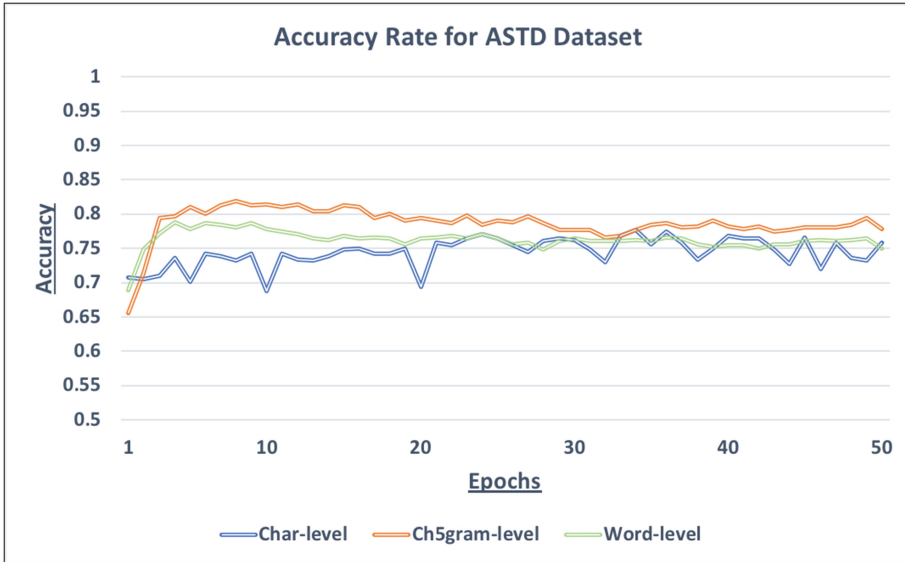
**Fig. 2.** Accuracy on the test set for Main-AHS dataset using different sentiment analysis levels.



**Fig. 3.** Accuracy on the test set for Sub-AHS dataset using different sentiment analysis levels.



**Fig. 4.** Accuracy on the test set for Ar-Twitter dataset using different sentiment analysis levels.



**Fig. 5.** Accuracy on the test set for ASTD dataset using different sentiment analysis levels.

## 6 Conclusions and Future Work

This paper investigated the benefits of combining CNNs and LSTMs networks in an Arabic sentiment classification task. It also explored the effectiveness of using different levels of sentiment analysis because of the complexities of morphology and orthography in Arabic. We used character level to increase the number of features for each tweet, as we are dealing with short messages, which was not an ideal option for our model. However, using Word-level and Ch5gram-level have shown better sentiment classification results.

This approach has improved the sentiment classification accuracy for our Arabic Health Services (AHS) dataset to reach 0.9424 for the Main-AHS dataset, and 0.9568 for the Sub-AHS dataset, compared to our previous results in [10] which were 0.92 for the Main-AHS dataset and 0.95 for the Sub-AHS dataset. Future work will use some pre-trained word representation models, such as word2vec [16], GloVe [35], and Fasttext [36] for the embedding layer.

## References

1. Liu, B.: Sentiment Analysis and Opinion Mining. Synthesis Lectures on Human Language Technologies. Morgan & Claypool, San Rafael (2012)
2. Balaji, P., Nagaraju, O., Haritha, D.: Levels of sentiment analysis and its challenges: a literature review. In: International Conference on Big Data Analytics and Computational Intelligence (ICBDAC) 2017, Chirala, India, vol. 6, pp. 436–439 (2017)



3. Feldman, R.: Techniques and applications for sentiment analysis. *Commun. ACM* **56**(4), 82–89 (2013)
4. Lakomkin, E., Bothe, C., Wermter, S.: GradAscent at EmoInt-2017: character and word level recurrent neural network models for tweet emotion intensity detection. In: Editor, F., Editor, S. (eds.) *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis 2017*, pp. 169–174. ACL, Copenhagen (2017)
5. Joshi, A., Prabhu, A., Shrivastava, M., Varma, V.: Towards sub-word level compositions for sentiment analysis of Hindi-English code mixed text. In: *Proceedings of COLING 2016, The 26th International Conference on Computational Linguistics: Technical Papers 2016*, pp. 2482–2491. The COLING 2016 Organizing Committee, Osaka (2016)
6. Kim, Y.: Convolutional neural networks for sentence classification. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751. ACL, Doha (2014)
7. Yin, W., Schütze, H.: Multichannel variable-size convolution for sentence classification. In: *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pp. 204–214. ACL, Beijing (2015)
8. Shin, B., Lee, T., Choi, J.D.: Lexicon integrated CNN models with attention for sentiment analysis. In: *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pp. 149–158. ACL, Copenhagen (2017)
9. Alayba, A.M., Palade, V., England, M., Iqbal, R.: Arabic language sentiment analysis on health services. In: *2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR)*, pp. 114–118. IEEE, Nancy (2017)
10. Alayba, A.M., Palade, V., England, M., Iqbal, R.: Improving sentiment analysis in Arabic using word representation. In: *2018 2nd International Workshop on Arabic and Derived Script Analysis and Recognition (ASAR)*, pp. 13–18. IEEE, London (2018)
11. Athiwaratkun, B., Kang, K.: Feature Representation in Convolutional Neural Networks. arXiv preprint [arXiv:1507.02313](https://arxiv.org/abs/1507.02313) (2015)
12. Gers, F.A., Eck, D., Schmidhuber, J.: Applying LSTM to time series predictable through time-window approaches. In: Tagliaferri, R., Marinaro, M. (eds.) *Neural Nets WIRN Vietri-01. Perspectives in Neural Computing 2002*, vol. 9999, pp. 193–200. Springer, London (2002). [https://doi.org/10.1007/978-1-4471-0219-9\\_20](https://doi.org/10.1007/978-1-4471-0219-9_20)
13. Yousefi-Azar, M., Hamey, L.: Text summarization using unsupervised deep learning. *Expert Syst. Appl.* **68**, 93–105 (2017)
14. Yih, S.W., He, X., Meek, C.: Semantic parsing for single-relation question answering. In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, (Vol. 2: Short Papers)*, pp. 643–648. ACL, Baltimore (2014)
15. Auli, M.W., Galley, M., Quirk, C., Zweig, G.: Joint language and translation modeling with recurrent neural networks. In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1044–1054. ACL, Seattle (2013)
16. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: *Proceedings of the 26th International Conference on Neural Information Processing Systems NIPS 2013*, vol. 2, pp. 3111–3119. Curran Associates Inc., Lake Tahoe (2013)
17. Wang, Y., Huang, M., Zhao, L., Zhu, X.: Attention-based LSTM for aspect-level sentiment classification. In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing 2016*, pp. 606–615. ACL, Austin (2016)

18. Abdul-Mageed, M., Diab, M.T., Korayem, M.: Subjectivity and sentiment analysis of modern standard Arabic. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies HLT 2011: Short Papers - Volume 2, pp. 587–591. ACL, Stroudsburg (2011)
19. Shoukry, A., Rafea, A.: Sentence-level Arabic sentiment analysis. In: 2012 International Conference on Collaboration Technologies and Systems (CTS), pp. 546–550. IEEE, Denver (2012)
20. Abdulla, N.A., Ahmed, N.A., Shehab, M.A., Al-Ayyoub, M.: Arabic sentiment analysis: lexicon-based and corpus-based. In: 2013 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT), pp. 1–6. IEEE, Amman (2013)
21. Abdulla, N., Majdalawi, R., Mohammed, S., Al-Ayyoub, M., Al-Kabi, M.: Automatic lexicon construction for Arabic sentiment analysis. In: 2014 International Conference on Future Internet of Things and Cloud, pp. 547–552. IEEE, Barcelona (2014)
22. Badaro, G., Baly, R., Hajj, H., Habash, N., El-Hajj, W.: A large scale Arabic Sentiment Lexicon for Arabic opinion mining. In: Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP), pp. 165–173. ACL, Doha (2014)
23. Duwairi, R.M., Marji, R., Sha’ban, N., Rushaidat, S.: Sentiment analysis in Arabic tweets. In: 2014 5th International Conference on Information and Communication Systems (ICICS), pp. 1–6. IEEE, Irbid (2014)
24. Al Sallab, A., Hajj, H., Badaro, G., Baly, B., El Haj, W., Shaban, K.B.: Deep learning models for sentiment analysis in Arabic. In: Proceedings of the Second Workshop on Arabic Natural Language Processing, pp. 9–17. ACL, Beijing (2015)
25. Ibrahim, H.F., Abdou, S.M., Gheith, M.: Sentiment analysis for modern standard Arabic and colloquial. *Int. J. Nat. Lang. Comput. (IJNLC)* **4**(2), 95–109 (2015)
26. Dahou, A., Xiong, S., Zhou, J., Haddoud, M.H., Duan, P.: Word embeddings and convolutional neural network for Arabic sentiment classification. In: Proceedings of COLING 2016, The 26th International Conference on Computational Linguistics: Technical Papers, pp. 2418–2427. The COLING 2016 Organizing Committee, Osaka (2016)
27. Tartir, S., Abdul-Nabi, I.: Semantic sentiment analysis in Arabic social media. *J. King Saud Univ. Comput. Inf. Sci.* **29**(2), 229–233 (2017)
28. El-Beltagy, S.R., Khalil, T., Halaby, A., Hammad, M.: Combining lexical features and a supervised learning approach for Arabic sentiment analysis. In: Gelbukh, A. (ed.) *CICLing 2016*. LNCS, vol. 9624, pp. 307–319. Springer, Cham (2018). [https://doi.org/10.1007/978-3-319-75487-1\\_24](https://doi.org/10.1007/978-3-319-75487-1_24)
29. Nabil, M., Aly, M., Atiya, A.: ASTD: Arabic sentiment tweets dataset. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 2515–2519. ACL, Lisbon (2015)
30. Smrř, O.: Encode Arabic Online Interface. <http://quest.ms.mff.cuni.cz/cgi-bin/encode/index.fcgi>. Accessed 18 June 2018
31. Bahloul, M.: *Structure and Function of the Arabic Verb*. Routledge, London (2008)
32. Keras. <https://keras.io>. Accessed 15 Apr 2018
33. Han, J., Moraga, C.: The influence of the sigmoid function parameters on the speed of backpropagation learning. In: Mira, J., Sandoval, F. (eds.) *IWANN 1995*. LNCS, vol. 930, pp. 195–201. Springer, Heidelberg (1995). [https://doi.org/10.1007/3-540-59497-3\\_175](https://doi.org/10.1007/3-540-59497-3_175)
34. Manning, C., Raghavan, P., Schütze, H.: *Introduction to Information Retrieval*, 1st edn. Cambridge University Press, New York (2008)

35. Pennington, J., Socher, R., Manning, C.D.: GloVe: global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543. ACL, Doha (2014)
36. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. *Trans. Assoc. Comput. Linguist.* **5**, 135–146 (2017)



# Between the Lines: Machine Learning for Prediction of Psychological Traits - A Survey

Dirk Johannßen<sup>(✉)</sup> and Chris Biemann

LT Group, MIN Faculty, Department of Computer Science, Universität Hamburg,  
Vogt-Kölln-Straße 30, 22527 Hamburg, Germany  
{johannssen,biemann}@informatik.uni-hamburg.de  
<http://lt.informatik.uni-hamburg.de/>

**Abstract.** A connection between language and psychology of natural language processing for predicting psychological traits (NLPpsych) is apparent and holds great potential for accessing the psyche, understand cognitive processes and detect mental health conditions. However, results of works in this field that we call NLPpsych could be further improved and is sparse and fragmented, even though approaches and findings often are alike. This survey collects such research and summarizes approaches, data sources, utilized tools and methods, as well as findings. Approaches of included work can roughly be divided into two main strands: word-list-based inquiries and data-driven research. Some findings show that the change of language can indicate the course of mental health diseases, subsequent academic success can be predicted by the use of function words and dream narratives show highly complex cognitive processes – to name but a few. By surveying results of included work, we draw the ‘bigger picture’ that in order to grasp someone’s psyche, it is more important to research *how* people express themselves rather than *what* they say, which surfaces in function words. Furthermore, often research unawarely induce biases that worsen results, thus leading to the conclusion that future research should rather focus on data-driven approaches rather than hand-crafted attempts.

**Keywords:** Computational psychology · Machine learning · Survey  
Natural language processing

## 1 Introduction

One rather newly opened application field for natural language processing (NLP), is NLP for predicting psychological traits, which we call NLPpsych. Due to computer systems in clinical psychology, massive amounts of textual interactions in social networks, as well as an uprising of blogs and online communities,

the availability of massive amounts of data has catalyzed research of psychological phenomena such as mental diseases, connections between intelligence and use of language or a data-driven understanding of dream language – to name but a few – with NLP methods.

Possible applications range from detecting and monitoring a course of mental health illnesses by analyzing language [1], finding more objective measures and language clues on subsequent academic success of college applicants [2] or discovering that dream narratives show highly complex cognitive processes [3]. Promising possible scenarios for future work could explore connections of personality traits or characteristics with subsequent development or research on current emotional landscapes of people by their use of language.

Even though the potential is high, the sub-field of natural language processing in psychology we call NLPsych henceforth is a rather fragmented field. Results of included works vary in accuracy and often show room for improvement by using either best-practice methods, by shifting the research focus onto e.g. function words, by using data-driven methods or by combining established approaches in order to perform better. This survey provides an overview over some of the recent approaches, utilized data sources and methods, as well as findings and promising pointers. Furthermore, the aim is to hypothesize about possible connections of different findings in order to draw a ‘big picture’ from those findings.

## 1.1 Research Questions

Even though the most broadly employed research questions target mental health due to the high relevance of findings in clinical psychology, this survey ought to have a broader understanding. Thus the following research questions, which have been derived from included work, approach mental changes, cognitive performance and emotions. Three exemplary works, that address similar questions, are mentioned as well. Important ethical considerations are out of scope of this paper (see e.g. dedicated workshops<sup>1</sup>).

**Research Question (i)** Does a change of the cognitive apparatus also change the use of language and if so, in what way? (e.g. [3–5])

**Research Question (ii)** Does the use of language correspond to cognitive performance and if so, which aspects of language are indicators? (e.g. [2,6,7])

**Research Question (iii)** Is a current mood or emotion detectable by the use of language besides explicit descriptions of the current mental state? (e.g. [1,7,8])

## 1.2 Structure of This Paper

Firstly, this survey aims to grant an overview of some popular problem domains with an idea of employed approaches and data sources in Sect. 2 and the development of broadly utilized tools in Sect. 3. Widely employed measurements of different categories will be discussed (Sect. 4). Section 5 describes utilized methods and tools, and is divided into two strands: on the one hand, data-driven

<sup>1</sup> <http://www.ethicsinnlp.org/>.

approaches that use supervised machine learning and on the other hand, statistics on statistics on manually defined features and so-called ‘inquiries’, i.e. counts over word lists and linguistic properties. Secondly, this work surveys parallels and connections of important findings that are utilized in order to conclude a ‘bigger picture’ in Sect. 6.

### 1.3 Target Audience

This paper targets an audience that is both familiar with basics of natural language processing and psychology, with some experience in machine learning (ML), Deep Learning (DL) or the use of standard tools for those fields, even though some explanations will be provided.

### 1.4 Criteria for Inclusion

Criteria for the inclusion of surveyed work can be divided into three aspects: Firstly, often cited work and very influential findings were included. Secondly, the origin of included work such as well established associations, authors or journals. And lastly, the soundness of the content with this survey’s focus in terms of methodology or topic. Only if a work suits at least one – if not all – of those aspects, said work has been included in this survey. E.g. work published by the Association for Computational Linguistics (ACL)<sup>2</sup>, which targets NLP, was considered. Well established journals of different scientific fields such as e.g. Nature, which dedicates itself to natural science, were considered. Search queries included ‘lexical database’, ‘psychometric’, ‘dream language’, ‘psychology’, ‘mental’, ‘cognitive’ or ‘text’. Soundness was considered in terms of topic (e.g. subsequent academic success, mental health prediction or dream language), as well as innovative and novel approaches.

## 2 Popular Problem Domains, Approaches and Data Sources

This section presents popular NLPpsych problem domains and is ordered by descending popularity. Approaches will be briefly explained.

### 2.1 Mental Health

Mental health is the most common problem domain for approaches that use NLP to characterize psychological traits as some of the following works demonstrate.

**Depression Detection Systems.** Morales *et al.* [9] summarized different depression detection systems in their survey and show an emerging field of research that has matured. Those depression detection systems often are linked to language and therefore have experienced gaining popularity among NLP in

<sup>2</sup> <https://www.aclweb.org/portal/>.

clinical psychology. Morales *et al.* [9] described and analyzed utilized data sources as well. *The Distress Analysis Interview Corpus (DAIC)*<sup>3</sup> offers audio and video recordings of clinical interviews along with written transcripts on depressions and thus is less suitable for textual approaches that solemnly focus on textual data but can be promising when visual and speech processing are included. The *DementiaBank* database offers different multi media entries on the topic of clinical dementia research from 1983 to 1988. *The ReachOut Triage Shared Task* dataset from the SemEval 2004 Task 7 consists of more than 64,000 written forum posts and was fully labeled for containing signs of depression. Lastly, *Crisis Text Line*<sup>4</sup> is a support service, which can be freely used by mentally troubled individuals in order to correspond textually with professionally trained counselors. The collected and anonymized data can be utilized for research.

**Suicide Attempts.** In their more recent work, Coppersmith *et al.* [10] investigated mental health indirectly by analyzing social media behavior prior to suicide attempts on Twitter. *Twitter*<sup>5</sup> is a social network, news- and micro blogging service and allows registered users to post so-called tweets, which were allowed to be 140 characters in length before November 2017 and 280 characters after said date. As before in [11], the Twitter users under observation had publicly self-reported their condition or attempt.

**Crisis.** Besides depression, anxiety or suicide attempts, there are more general crises as well, which Kshirsagar *et al.* [12] detect and attempt to explain. For their work they used a specialized social network named Koko and used a combination of neural and non-neural techniques in order to build classification models. *Koko*<sup>6</sup> is an anonymous emotional peer-to-peer support network, used by Kshirsagar *et al.* [12]. The dataset originated from a clinical study at the MIT and can be implemented as chatbot service. It offers 106,000 labeled posts, with and some without crisis. A test set of 1,242 posts included 200 crisis labeled entries, i.e.  $\sim 16\%$ .

*Reddit*<sup>7</sup> is a community for social news rather than plain text posts and offers many so-called sub-reddits, which are sub-forums dedicated to certain, well defined topics. Those sub-reddits allow for researchers to purposefully collect data. Shen *et al.* [13] detected anxiety on Reddit by using depression lexicons for their research and training Support Vector Machine (SVM, Cortes *et al.* [14]) classifiers, as well as Latent Dirichlet Allocation (LDA, Blei *et al.* [15]) for topic modeling (for LDA see Sect. 3). Those lexicons offer broad terms that can be combined with e.g. Language Inquire and Word Count (LIWC, Pennebaker *et al.* [16]) features in order to identify different conditions in order to be able to distinguish those mental health issues. Shen *et al.* [13] used an API offered by Reddit in order to access sub-reddits such as r/anxiety or r/panicparty.

<sup>3</sup> <http://dcapswoz.ict.usc.edu/>.

<sup>4</sup> <https://www.crisistextline.org/>.

<sup>5</sup> <https://twitter.com/>.

<sup>6</sup> <https://itskoko.com/>.

<sup>7</sup> <https://www.reddit.com/>.

**Dementia.** In their recent work, Masrani *et al.* [17] used six different blogs to detect dementia by using different classification approaches. Especially the lexical diversity of language was the most promising feature, among others.

**Multiple Mental Health Conditions.** Coppersmith *et al.* [11] researched the detection of a broad range of mental health conditions on Twitter. Coppersmith *et al.* [11] targeted the well discriminability of language characteristics of the following conditions: attention deficit hyperactivity disorder (ADHD), anxiety, bipolar disorder, borderline syndrome, depression, eating disorders, obsessive-compulsive disorder (OCD), post traumatic stress disorder (PTSD), schizophrenia and seasonal affective disorder (SAD) – all of which were self-reported by Twitter users.

## 2.2 Dreams Language in Dream Narratives

**Dream Language.** Niederhoffer *et al.* [3] researched the general language of dreams from a data-driven perspective. Their main targets are linguistic styles, differences between waking narratives and dream narratives, as well as the emotional content of dreams. In order to achieve this, they used a community named DreamsCloud. *DreamsCloud*<sup>8</sup> is a social network community dedicated to sharing dreams in a narrative way, which also offers the use of data for research purposes. There are social functions such as ‘liking’ a dream narrative or commenting on it, as Niederhoffer *et al.* [3] describe in their work. There are more than 119,000 dream narratives from 74,000 users, which makes this network one of the largest of its kind. Since DreamsCloud is highly specialized, issues such as relevance or authenticity are less crucial as they would be on social networks like Facebook<sup>9</sup>.

**LIWC and Personality Traits.** Hawkins *et al.* [18] layed their focus on LIWC characteristics especially and a correlation with the personality of a dreamer. Data was collected by clinical studies in which Hawkins *et al.* [18] gathered dream reports from voluntary participants. Their work is more thorough in terms of length, depth and rate of conducted experiments on LIWC features. Dreams could be distinguished from waking narratives, but – as of said study – correlations with personality traits could not be found.

## 2.3 Mental Changes

As we will be showing in Sect. 6, mental changes and mental health problems are seemingly connected. However, natural changes such as growth or life-changing experiences can alter the use of language as well.

**Data Generation and Life-Changing Events.** Oak *et al.* [19] pointed out that the availability of data in the clinical psychology often is a difficulty for researchers. The application scenario chosen for a study on data generation for

<sup>8</sup> <https://www.dreamscloud.com/>.

<sup>9</sup> <http://www.facebook.com>.



clinical psychology are life-changing events. Oak *et al.* [19] aimed to use NLP for tweet generation. The BLEU score measures n-gram precision, which can be important for next character- or next word predictions, as well as for classification tasks. Another use case of this measure is the quality of machine translations. Oak *et al.* [19] use the BLEU score to evaluate the quality of their n-grams for language production of their data generation approach of life-changing events. Even though the generated data would not be appropriate to be used for e.g. classification tasks, Oak *et al.* [19] nonetheless proposed useful application scenarios such as virtual group therapies. 43% of human annotators thought the generated data to be written by real Twitter users.

**Changing Language Over the Course of Mental Illnesses.** A study by Reece *et al.* [1] revealed that language can be a key for detecting and monitoring the whole process from onsetting mental illnesses to a peak and a decline as therapy shows positive effects on patients. Participants involved in the study had to prove their medical diagnosis and supply their Twitter history. Different techniques were used to survey language changes. MTurk was used for labeling their data. Reece *et al.* [1] were able to show a correlation between language changes and the course of a mental disease. Furthermore, their model achieved high accuracy in classifying mental diseases throughout the course of illness.

**Language Decline Through Dementia and Alzheimer's.** It is known that cognitive capabilities decline during the course of the illness dementia. Masrani *et al.* [17] were able to show that language declines as well. Lancashire *et al.* [20] researched the possibility of approaching Alzheimer's of the writer Agatha Christie by analyzing novels written at different life stages from age 34 to 82. The first 50,000 words of included novels were inquired with a tool named TACT, which operates comparable to LIWC (shown in Sect. 3) and showed a decline in language complexity and diversity. During their research, Masrani *et al.* [17] detected dementia by including blogs from medically diagnosed bloggers with and without dementia. Self-reported mental conditions, as it is often used for research of social networks, are at risk of being incorrect (e.g. pranks, exaggeration or inexperience).

**Development.** Goodman *et al.* [8] showed that the acquisition and comprehension of words and lexical categories during the process of growth correspond with frequencies of parental usage, depending on the age of a child. Whilst the acquisition of lexical categories and comprehension of words correlates with the frequency of word usage of parents later on in life, simple nouns are acquired earlier. Thus, whether words were more comprehensible was dependent on known categories and a matter of similarity by the children.

## 2.4 Motivation and Emotion

Emotions and motivations are less common problem domains. Some approaches aim to detect general emotions, further researchers focus on strong emotions such as hate speech, others try to provide valuable resources or access to data.

**Distant Emotion Detection.** In order to better understand the emotionality of written content, Pool *et al.* [21] used emotional reactions of Facebook users as labels for classification. *Facebook* offers insightful social measurements such as richer reactions on posts (called *emoticons*) or numbers as friends, even though most available data is rather general.

**Hate Speech.** Serrà *et al.* [4] approached the question of emotional social network posts by surveying the characteristics of hate speech. In order to tackle hate speech usually containing a lot of neologism, spelling mistakes and out-of-vocabulary words (OOV), Serrà *et al.* [4] constructed a two-tier classification that firstly predicts next characters and secondly measures distances between expectation and reality. Other works on hate speech include [22–24].

**Motivational Dataset.** Since data sources for some sub-domains such as motivation are sparse, Pérez-Rosas *et al.* [25] created a novel contributing a motivational interviewing (MI) dataset by including 22,719 utterances from 227 distinct sessions, conducted by 10 counselors. *Amazon mechanical turk* (MTurk) is a crowdsourcing service. Research can define manual tasks and define quality criteria. Pérez-Rosas *et al.* [25] used MTurk for labeling their short texts by crowdsourcers. They achieved a high Intraclass Correlation Coefficient (ICC) of up to 0.95. MI is a technique in which the topic ‘change’ is the main object of study. Thus, as described in Subsect. 6.3, this dataset could also contribute to early mental disease detection. MI is mainly used for treating drug abuse, behavioral issues, anxiety or depressions.

**Emotions.** Pool *et al.* [21] summarized in their section on emotional datasets some highly specialized databases on emotions, which the authors analyzed thoroughly. *The International Survey on Emotion Antecedents and Reactions* (ISEAR)<sup>10</sup> dataset offers 7,665 labeled sentences from 3,000 respondents on the emotions of joy, fear, anger, sadness, disgust, shame and guilt. Different cultural backgrounds are included. *The Fairy Tales*<sup>11</sup> dataset includes the emotional categories angry, disgusted, fearful, happy, sad, surprised and has 1,000 sentences from fairy tales as the data basis. Since fairy tales usually are written with the intention to trigger certain emotions of readers or listeners, this dataset promises potential for researchers. *The Affective Text*<sup>12</sup> dataset covers news sites such as Google news, NYT, BBC, CNN and was composed for the SemEval 2007 Task 14. It offers a database with 250 annotated headlines on emotions including anger, disgust, fear, joy, sadness and surprise.

## 2.5 Academic Success

Few researchers in NLPsych have approached a connection between language and academic success. Some challenges are lack of data and heavy biases as

<sup>10</sup> <http://emotion-research.net/toolbox/toolboxdatabase.2006-10-13.2581092615>.

<sup>11</sup> <https://github.com/bogdanneacsu/tts-master/tree/master/fairytales>.

<sup>12</sup> <http://web.eecs.umich.edu/~mihalcea/downloads/AffectiveText.SemEval.2007.tar.gz>.

some might assume that an eloquent vocabulary, few spelling mistakes or a sophisticated use of grammar indicate a cognitive skilled writer. Pennebaker *et al.* [2] approached the subject in a data-driven fashion and therefore less biased. Data was collected by accessing more than 50,000 admission essays from more than 25,000 applicants. The college admission essays could be labeled with later academic success indicators such as grades. The study showed that rather small words such as function words correlate with subsequent success, even across different majors and fields of study. Function words (also called closed class words) are e.g. pronouns, conjunctions or auxiliary words, which tendentially are not open for expansion, whilst open class words such as e.g. nouns can be added during productive language evolution.

### 3 Tools

In this section we discuss some broadly used tools for accessing mainly written psychological data. Included frameworks are limited to the programming language Python, since it is well established – especially for scientific computing – and included works mostly use libraries and frameworks designed for Python.

#### 3.1 Word Lists

**LIWC.** The Language Inquiry and Word Count (LIWC) was developed by Pennebaker *et al.* [16] for the English language and has been transferred to other language such as e.g. German by Wolf *et al.* [26]. The tool was psychometrically validated and can be considered a standard in the field. LIWC stands for a tool that operates with recorded dictionaries of word lists and a vector of approximately 96 metrics (depending on the version and language) such as number of pronouns or number of words associated with familiarity to be counted in input texts.

**CELEX**<sup>13</sup> is a lexical inquiry database, that was developed by Baayen *et al.* [27] and later on enhanced to a CELEX release 2. The database contains 52,446 lemmas and 160,594 word forms in English and a number of those in Dutch and German as well. It is regularly used by researchers such as Fine *et al.* [28] did in order to research possible induced biases in corpora, which used CELEX for predicting human language by measuring proportions of written and spoken English based on CELEX entries.

Kshirsagar *et al.* [12] used the Affective Norms for English Words (ANEW), which is an inquiry tool such as LIWC, as well as labMT, used by Reece *et al.* [1], which is a word list score for sentiment analysis.

#### 3.2 Corpus-Induced Models

**LDA.** Blei *et al.* [15] developed a broadly used generative probabilistic model called Latent Dirichlet Allocation (LDA) that is able to collect text through a three-layered Bayesian model that builds models on the basis of underlying topics.

<sup>13</sup> <https://catalog ldc.upenn.edu/ldc96114>.

**SRILM.** The SRI Language modeling toolkit (SRILM), produced by Stolcke [29] is a software package that consists of C++ libraries, other programs and scripts that combine functionality for processing, as well as producing mainly speech recognition and other applications such as text. Oak *et al.* [19] used SRILM for 4-gram modeling for language generation of life-changing events.

### 3.3 Frameworks

**NLTK.** The Natural Language Toolkit (NLTK) is a library for Python that offers functionality for language processing, e.g. tokenization or part-of-speech (POS) tagging. It is used on a general basis. E.g. Shen *et al.* [13] use NLTK for POS tagging and collocation.

**Scikit-Learn.** The tool of choice of Pool *et al.* [21] and Shen *et al.* [13] was scikit-learn [30], a freely available and open sourced library for Python. Since scikit-learn is designed to be compatible with other numerical libraries, it can be considered one of the main libraries for machine learning in the field of natural language processing.

### 3.4 Further Tools

Further tools that are being used in included work in some places are the cross-linguistic lexical norms database (CLEX) [31] for evaluating and comparing early child language, the Berlin affective word list reloaded (BAWL-R) [32] that is based on the previous version of BAWL for researching affective words in the German language and lastly HMMlearn, used by Reece *et al.* [1], which is a Python library for Hidden Markov Models (HMM).

## 4 Psychometric Measures

When conducting research on NLPsych, the selection of psychometric measurements are crucial for evaluating given data on psychological effects or to detecting the presence of target conditions before a classification task can be set up. Therefore, in this section we describe broadly used psychometric measurements of included work. Psychometrics can be understood as a discipline of psychology – usually found in clinical psychology – that focus on ‘testing and measuring mental and psychology ability, efficiency potentials and functions’ [33].

There are measures for machine learning as well such as e.g. the accuracy score, which will not be covered due to broadly available standard literature on this matter.

### 4.1 Questionnaires

**BDI and HAM-D.** The Beck Depression Inventory (BDI) and HAM-D are used and described by Morales *et al.* [9] and Reece *et al.* [1] for measuring the severity

of depressions. The HAM-D is a questionnaire that is clinically administrated and consists of 21 questions, whilst the BDI is a questionnaire that consists of the same 21 questions, but is self-reported.

**CES-D.** The Center for Epidemiological Studies Depression Scale (CES-D) is a questionnaire for participants to keep track on their depression level and has been used in their work by Reece *et al.* [1].

## 4.2 Wordlist Measurements

**MITI.** The Motivational Interviewing Integrity Treatment score (MITI) measures how well or poorly a clinician is using MI (motivational interviewing), as Pérez-Rosas *et al.* [25] described in their work. The Processes related to ‘change talk’, thus the topical focus, is the crucial part of this measurement. Global counts and behavior counts distinguish the impact on this measure. Words that encode the MITI level are e.g. ‘focus’, ‘change’, ‘planning’ or ‘engagement’.

**CDI.** The Categorical Dynamic Index (CDI), used by Niederhoffer *et al.* [3], Jørgensen *et al.* [31], as well as Pennebaker *et al.* [2], is described as a bipolar continuum, applicable on any text, that measures the extend of how categorical or dynamic thinking is. Since those two dimensions are said to distinguish between cognitive styles of thinking, it therefore can reveal e.g. whether or not dreamers are the main character of their own dream [3]. The CDI can be measured by inquiring language with tools such as e.g. LIWC and weighting the categories.

## 5 Broadly Used Research Methods

Since there are two main approaches for performing NLPsych – data-driven approaches and manual approaches from clinical psychology – this section will be divided into those two strands, beginning with feature approaches and ending with data-driven machine learning approaches. Within those strands, the methods are ordered by their complexity.

### 5.1 A General Setup of NLPsych

Even though there are detailed differences between approaches of included works, there is a basic schema in the way NLPsych is set up. Figure 1 illustrates a classification setup. Firstly, after having collected data, pieces of information are read and function as input. Different measures or techniques can be applied to the data by an annotator to assign labels to the input. Whether or not annotation takes place, depends on the task and origin of the data.

Secondly, after separating training, test, and sometimes development sets, features get extracted from those data items, e.g. LIWC category counts, the ANEW sadness score or POS tags. A feature extractor computes a nominal or numerical feature vector, which will be described in Subsect. 5.3.

Thirdly, depending on the approach, this feature vector is directly used in rule based models such as e.g. defined LIWC scores that correlate with dream aspects, as Niederhoffer *et al.* [3] did. A different approach uses the feature vector on a machine learning algorithm in order to compute a classifier model, that thereafter can be used to classify new instances of information, as Reece *et al.* [1] demonstrated in their work.

Finally, for both of the approaches, the accuracy of the classification task is determined and researchers analyze and discuss the consequences of their findings, as well as use the models for classification tasks.

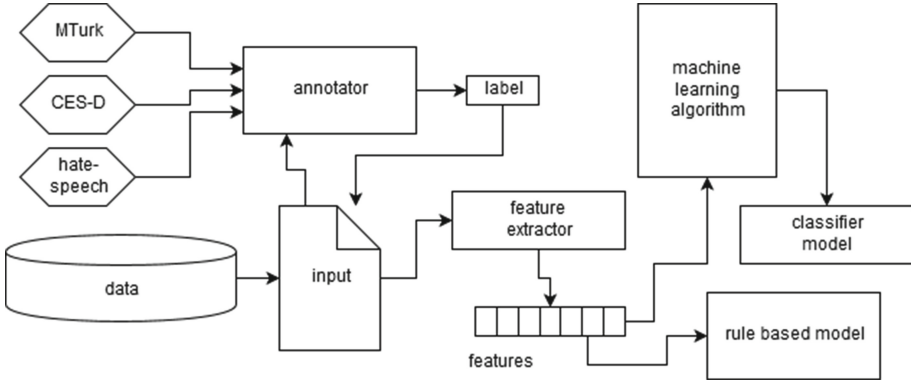


Fig. 1. A general setup for classification tasks in NLPsych

## 5.2 Supervised Machine Learning Approaches

**SVM.** Support Vector Machines (SVM) are a type of machine learning algorithm that measure distances of instances to so-called support vectors that map said examples in order to form a dividing gap. This gap separates said examples into categories to perform classification or regression tasks. This broadly utilized standard method has been used by e.g. Pool *et al.* [21] for BOW models via scikit-learn.

**HMM.** Reece *et al.* [1] used Hidden Markov Models (HMM), which are probabilistic models for modeling unseen events, as well as word shift graphs that visualize changes in the use of language [1].

**RNN.** The Recurrent Neural Networks (RNN) are an architecture of deep neural networks that differ from feed forward neural networks by having time-delayed connections to cells of the same layer and thus possesses a so-called memory. RNNs require for the input to be numeric feature vectors. Words or sentences typically get transformed by the use of embedding methods (e.g. [34]) into numerical representations. Some authors that use RNNs are Cho *et al.* [35] who used encoder and decoder in order to maximize the conditional probabilities of representations. Kshirsagar *et al.* [12] used RNNs for word embeddings and Serrà *et al.* [4] trained character based language models with RNNs.

**LSTM.** A Long short-term memory neural network (LSTM, Hochreiter *et al.* [36]) is a type of RNN in which three gates (input, forget and output) in an inner, so-called memory cell, are employed to be able to learn the amount of retained memory depending on the input and the inner state. LSTMs are capable of saving information over arbitrary steps, thus enabling them to *remember* a short past for sophisticated reasoning. LSTMs nowadays are the method of choice for classification on sequences and can be considered as established standard. Long short-term memory neural networks often are used when calculation power, as well as big amounts of data are available and a memory is needed to train precise models. The latter often is the case when working with psychological data. E.g. Oak *et al.* [19] used an LSTM for training language models for language production of life-changing events.

### 5.3 Features for Characterizing Text

Features serve as characteristics of texts and are always computable for every text, e.g. the average rate of words per sentence. Some of said features are numerical, some are nominal. Those features usually are stored in a feature vector that serves as input for classifiers but can be used directly, e.g. in order to perform statistics on them and to draw conclusions. Not every presented feature is being used as such. On the one hand, LIWC, tagging and BOWs are used as characteristics of text and thus are classically used as features. On the other hand, LDA targets the data collection process and n-grams, CLMs, as well as next character predictions can be utilized for modeling.

**LIWC.** In Sect. 3 the LIWC is described as a set of categories for which word lists were collected. The core dictionary and tool with its capability of calculating a feature vector for language modeling is well established and can be categorized as method of choice in psychological language inquiry. The way LIWC is used, is very common. However, researchers usually focus on some selected aspects of the feature vector in order to grasp psychological effects. Coppersmith *et al.* [11] used LIWC for differentiating the use of language of healthy people versus people with mental conditions and diseases. Hawkins *et al.* [18] and Niederhoffer *et al.* [3] researched the language landscape of dream narratives. Scores, such as the LIWC sadness score were the basis of the work of Homan *et al.* [5] on depression symptoms. Morales *et al.* [9] also surveyed the broad use of LIWC in depression detection systems. Pennebaker *et al.* [2], which partly developed LIWC used the tool to research word usage in connection with college admission essays. Reece *et al.* [1] captured the general mood of participants by using LIWC and Shen *et al.* [13] surveyed the language of a crisis with LIWC.

**LDA.** Latent Dirichlet Allocation (LDA) is a probabilistic model for collecting text corpora on the basis of underlying topics in a three layered bayesian model, as described in Sect. 3. Some researchers that used the LDA are Niederhoffer *et al.* [3] for topic modeling in order to explore the main themes of given texts and Shen *et al.* [13] which used LDA to predict membership of classes by a given topic.

**BOW.** Bag of words (BOW – sometimes called vector space models) are models that intentionally dismiss information of the order of text segments or tokens and thus e.g. grammar by only taking into account presence resp. absence of word types in a text. Usually, BOW models are used for document representation where neither the order nor grammar of tokens are crucial but rather their frequency. Shen *et al.* [13] use so-called continuous bag-of-word models (CBOW, [34]) with a window size of 5 in order to create word embeddings. Homan *et al.* [5], Kshirsagar *et al.* [12] and Serrà *et al.* [4] use BOW for embedding purposes. *Tf-idf* is a measure for relevance that quantifies the term frequency (tf) inverse document frequencies (idf) by using said BOW models [12].

**Part of Speech Tagging (POS).** POS is the approach to assign lexical information to segmented or tokenized parts of a text. Those tags can be used as labels and hence be used as additional information for e.g. classification tasks. Some authors that used tagging were Masrani *et al.* [17] and Reece *et al.* [1].

**N-grams.** A continuous sequence of  $n$  tokens of a text is called  $n$ -gram. The higher the chosen  $n$ , the more precise language models on the basis of  $n$ -grams can be used for e.g. classification or language production while training becomes more excessive with higher  $n$ . Some of the authors that use either word-based  $n$ -grams or character based  $n$ -grams are Kshirsagar *et al.* [12], Homan *et al.* [5], Oak *et al.* [19], Reece *et al.* [1] and Shen *et al.* [13].

**CLM.** A Character  $n$ -gram Language Model (CLM) is closely related to  $n$ -grams and is a term for language models that use  $n$ -gram frequencies of letters for probabilistic modeling, used by Coppersmith *et al.* [11] as model that models emotions on the basis of character sequences.

**Next character prediction** is the prediction of words of characters on the basis of probabilistic language models, which have been used by Serrà *et al.* [4] for determining the soundness of an expectable use of language with actual language usage in order to detect hate-speech.

## 6 Findings from Included Works

In the following, we will mainly focus on firstly some important findings of the included work for the research questions, and secondly on granting a ‘big picture’ of a possible general connection between language and cognitive processes. An overview of the problem domains (without the approaches and data sources, as they are task specific), tools, psychometric measures and research methods can be found in Table 1.

### 6.1 Language and Emotions

**Hate speech** detection has been a popular task ever since the recent discussion of verbal abuse on social networks has dominated some headlines [4]. Hate speech is especially prone to neologism, out-of-vocabulary words (OOV) and a



**Table 1.** Overview of included works.

1st Author	Problem	Data sources	Tools	Measures	Method
Morales [9]	Depression	Multiple	i.a. LIWC	BDI, HAM-D	Manual
Copper. [10]	Suicide	Twitter			Manual
Copper. [11]	Multiple	Twitter	LIWC		CLM
Kshirs. [12]	Crisis	Koko	ANEW		RNN
Shen [13]	Anxiety	Reddit	NLTK, LIWC		SVM
Masrani [17]	Dementia	Blogs		TF-IDF, SUBTL	LR, NN
Hawkins [18]	Dreamers	Participants	LIWC		Manual
Niederhof. [3]	Dreams	DreamsCloud	LIWC	CDI	LDA
Oak [19]	Events	Twitter	SRILM	BLEU	n-grams
Reece [1]	Mental cond.	Twitter	MTurk, LIWC	BDI, CES-D	HMM
Goodman [8]	Development	Participants			RSA
Pool [21]	Emotions	ISEAR	Scikit-learn		BOW
S�erra [4]	Hate speech	Social networks	MTurk		RNN
Perez-R. [25]	Motivation	Participants	MTurk	ICC	Manual
Penneb. [2]	Acad. success	Participants	LIWC		Manual

lot of noise in the form of spelling and grammar mistakes. Furthermore, a known vocabulary of words that can be considered part of hate speech gets outdated rapidly. Serr a *et al.* [4] proposed a promising two-tier approach by training next character prediction models for each class as well as training a neural network classifier that takes said class models as input in order to measure the distance of expectation and reality. They achieved an accuracy of 0.951. Thus, in order to detect hate speech, it is more important to focus on *how* people alter their use of language rather than to focus on the particular words.

**Dreams.** Niederhoffer *et al.* [3] researched dream language by analyzing the content with an LDA topic model [15], categorizing emotions by the emotional classification model [10] and linguistic style via LIWC [16]. Dreams can be described as narratives, that predominantly describe past events in a first person point of view via first person pronouns with a particular attention to people, locations, sensations (e.g. hearing, seeing, the perceptual process of feeling). Since those dream narrations often exceed observations that are explainable by the dreamers (e.g. different physical laws of the observable world), complex cognitive processes can be assumed. Due to lexical categories revealing those connections, it can be concluded that it is more important *how* people express their dreams, rather than *what* they state.

**Distress.** In order to detect distress on Twitter, Homan *et al.* [5] asserted the so-called sadness score from LIWC together with keywords and could show a

direct link to the distress and anxiety of Twitter users. Homan *et al.* [5] also analyzed the importance of expert annotators and showed that their classifier, trained with expert annotator labels, achieved an F-score of 0.64. This direct link adds to the impression, that the way people express themselves is connected to cognitive processes.

All of the above mentioned findings and their direct conclusions lead to an answer of the **research question (i)** on the connection between emotions and language, which can be reacted upon with approval.

## 6.2 Cognitive Performance and Language

Works on subsequent academic success often induce strong biases such as the intuition that spelling mistakes indicate cognitive performance. The study of language and context that has been undertaken by Pennebaker *et al.* [2] indirectly tackles those biases, as the study targets a connection between the use of language and subsequent academic success by investigating college proposal essays with LIWC. Pennebaker *et al.* [2] could show that cognitive potential was not connected to *what* applicants expressed but rather to *how* they expressed themselves in terms of closed class words such as pronouns, articles, prepositions, conjunctions, auxiliary verbs or negation. However, correlations over four years of college measured each year, ranged from  $r = 0.18$  to  $r = 0.2$ , which are significant, but not very high. The second **research question (ii)** targets a connection between cognitive performance and the use of language. Closed class words such as function words have shown a connection with subsequent academic success. Therefore, the research question can be confirmed, that cognitive performance can be connected with the use of language.

## 6.3 Changing Language and Cognitive Processes

As Goodman *et al.* [7] pointed out, many phenomena in natural language processing such as implication, vagueness, non-literal language are difficult to detect. Some aspects of the use of language even stay unnoticed by speakers themselves: at times the use of language on social media platforms indicate early staged physical or mental health conditions, which even holds true when the speaker is not yet aware of the health decline [1] him- or herself, which induces the importance for early detection via use of language. By using aspects of informed speakers and game theory, Goodman *et al.* [7] achieved a correlation of  $r = 0.87$ .

Reece *et al.* [1] were able to detect an early onset of dementia through tweets (Twitter posts) up to nine months before the official diagnosis of participants has been made ( $F1 = 0.651$ ). Moreover, the word shift graphs of Hidden Markov Models (HMM) on time series in a sliding window could show a course of disease from early changes in language to stronger changes and a diagnosis until a normalization of language use as the condition was treated. This change has been detected by the labMT happiness score, which is a sentiment measurement

tool for psychologically depicted scores on a dictionary, similar to LIWC. Thus, the connection of mental changes and the use of language, subject to **research question (iii)** can be confirmed as well.

## 7 Conclusion

Across most studies of included works, there are two main conclusion.

**Reduction of Bias.** It has shown that function words can be the key factors of grasping the psyche of humans by surveying their use of language. Fine *et al.* [28] showed in their work that some corpora unknowingly induced strong biases that alter the objectivity of said corpora – e.g. the corpus of Google for n-grams over-predicts how fast technological terms are understood by humans. Researchers tend to resort to strong biases when designing e.g. data collection for corpora or classification tasks, since experiences seemingly foretell e.g. cognitive performance with biased measures such as the usage of a complex grammar, eloquence or of making few spelling mistakes – as explained by Pennebaker *et al.* [2] –, thus leading us to the following, second conclusion:

**Focus on *how* People Express Themselves, Rather Than *What* They Express.** The three research questions and their answers have lead to a hypothesis based on findings of included works: in order to grasp the psyche by the use of language, it is more important to survey *how* people express themselves rather than which words are actually used. Most important findings when looking at NLPsych have in common, that a possible key for accessing the psyche lies in small words such as function words or with dictionaries developed by psychologists that focus on cognitive associations with words rather than the lexical meaning of words, such as LIWC, labMT or ANEW. Furthermore, function words are more accessible, easier to measure and easier to count than e.g. complex grammar. Thus leading us to the conclusion that in order to access the psyche of humans through written texts, the most promising approaches are data driven, aware of possible biases and focus on function words rather than a content-based representation.

## 8 Future Work

This section discusses some possible next steps for research in NLPsych.

**A Connection of Scientific Fields and Sub-fields.** As shown in Sect. 2, natural language processing in the sub-field of psychology is mainly about the study of language in clinical psychology and thus connected to mental conditions and diseases. Findings from other application areas such as dream language or the connection of language and academic success as indicators for cognitive performance could be valuable if connected to, or if used in other domains.

**Researchers Should Rely More on Best Practice Approaches.** Some included work such as Reece *et al.* [1] demonstrate the advantages the sub-field

can experience if state of the art methods are used and connected in order to access the full potential of natural language. As Morales *et al.* [9] pointed out, it is promising to enhance promising research approaches with state of the art and best practice methods, as well as a connection to other sub-fields for future development of a natural language processing.

**Use Established Psychometrics Combined with NLPsych.** Whilst the already mentioned perceptions for future work – the connection of sub-fields and the usage of best practice approaches – are rather natural and known by many researchers, one possible research gap of NLPsych, the operant motive test (OMT) – developed by Scheffer *et al.* [37] –, illustrates the potential that NLPsych holds. The OMT is a well established psychometrical test that asserts the fundamental motives of humans by letting participants freely associate usually blurred images. Said images show scenarios in which labeled persons interact with each other. Participants are asked to answer questions on those images.

Since trained psychologists do not solemnly rely on provided word lists but rather develop an intuition for encoding the OMT – nonetheless showing high cross-observer agreement – that enables them to access the psyche, there has yet to be a method to be developed for this intuition by using best practice approaches and connecting scientific fields. This way, artificial intelligence might become even better at ‘reading between the lines’.

## References

1. Reece, A.G., Reagan, A.J., Lix, K.L.M., Dodds, P.S., Danforth, C.M., Langer, E.J.: Forecasting the onset and course of mental illness with twitter data. *Nat. Sci. Rep.* **7**(1), 13006 (2017). <https://doi.org/10.1038/s41598-017-12961-9>. ISSN: 2045-2322
2. Pennebaker, J.W., Chung, C.K., Frazee, J., Lavergne, G.M., Beaver, D.I.: When small words foretell academic success: the case of college admissions essays. *PLOS ONE* **9**(12), e115844 (2014). <https://doi.org/10.1371/journal.pone.0115844>. ISSN: 1932-6203
3. Niederhoffer, K., Schler, J., Crutchley, P., Loveys, K., Coppersmith, G.: In your wildest dreams: the language and psychological features of dreams. In: Proceedings of the Fourth Workshop on Computational Linguistics and Clinical Psychology - From Linguistic Signal to Clinical Reality, pp. 13–25. Association for Computational Linguistics, Vancouver (2017). <http://www.aclweb.org/anthology/W17-3102>
4. Serrà, J., Leontiadis, I., Spathis, D., Stringhini, G., Blackburn, J., Vakali, A.: Class-based prediction errors to detect hate speech with out-of-vocabulary words. In: Proceedings of the First Workshop on Abusive Language Online, pp. 36–40. Association for Computational Linguistics, Vancouver, August 2017. <http://www.aclweb.org/anthology/W17-3005>
5. Homan, C., Johar, R., Liu, T., Lytle, M., Silenzio, V., Alm, C.O.: Toward macro-insights for suicide prevention: analyzing fine-grained distress at scale. In: Proceedings of the Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, pp. 107–117. Association for Computational Linguistics, Baltimore (2014). <http://www.aclweb.org/anthology/W14-3213>

6. Tomasello, M.: *The New Psychology of Language: Cognitive and Functional Approaches to Language Structure*, 2nd edn, p. 376. Psychology Press, Mahwah (2002). ISBN: 978-1-317-69352-9
7. Goodman, N.D., Frank, M.C.: Pragmatic language interpretation as probabilistic inference. *Trends Cogn. Sci.* **20**(11), 818–829 (2016)
8. Goodman, J.C., Dale, P.S., Li, P.: Does frequency count? Parental input and the acquisition of vocabulary. *J. Child Lang.* **35**(3), 515–531 (2008). <https://doi.org/10.1017/S0305000907008641>. Accessed 21 Mar 2018. ISSN: 1469-7602, 0305-0009
9. Morales, M., Scherer, S., Levitan, R.: A cross-modal review of indicators for depression detection systems. In: *Proceedings of the Fourth Workshop on Computational Linguistics and Clinical Psychology - From Linguistic Signal to Clinical Reality*, pp. 1–12. Association for Computational Linguistics Vancouver (2017). <http://www.aclweb.org/anthology/W17-3101>
10. Coppersmith, G., Ngo, K., Leary, R., Wood, A.: Exploratory analysis of social media prior to a suicide attempt. Presented at the CLPsych, San Diego, CA, USA, pp. 106–117 (2016). <https://doi.org/10.18653/v1/W16-0311>
11. Coppersmith, G., Dredze, M., Harman, C., Hollingshead, K.: From ADHD to SAD: analyzing the language of mental health on twitter through self-reported diagnoses - semantic scholar. Presented at the CLPsych@HLTNAACL, Denver, CO, USA (2015). [www.aclweb.org/anthology/W15-1201](http://www.aclweb.org/anthology/W15-1201)
12. Kshirsagar, R., Morris, R., Bowman, S.: Detecting and explaining crisis. In: *Proceedings of the Fourth Workshop on Computational Linguistics and Clinical Psychology - From Linguistic Signal to Clinical Reality*, pp. 66–73. Association for Computational Linguistics, Vancouver (2017). <http://www.aclweb.org/anthology/W17-3108>
13. Shen, J.H., Rudzicz, F.: Detecting anxiety on Reddit. In: *Proceedings of the Fourth Workshop on Computational Linguistics and Clinical Psychology - From Linguistic Signal to Clinical Reality*, pp. 58–65. Association for Computational Linguistics, Vancouver (2017). <http://www.aclweb.org/anthology/W17-3107>
14. Cortes, C., Vapnik, V.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995). <https://doi.org/10.1023/A:1022627411411>. ISSN: 0885-6125
15. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *J. Mach. Learn. Res.* **3**, 993–1022 (2003). <http://dl.acm.org/citation.cfm?id=944919.944937>. ISSN: 1532-4435
16. Pennebaker, J., Chung, C., Ireland, M., Gonzales, A., Booth, R.J.: *The development and psychometric properties of LIWC 2007. Software Manual*, Austin, TX, USA (2007)
17. Masrani, V., Murray, G., Field, T., Carenini, G.: Detecting Dementia through retrospective analysis of routine blog posts by bloggers with Dementia. In: *BIONLP 2017*, pp. 232–237. Association for Computational Linguistics, Vancouver (2017). <http://www.aclweb.org/anthology/W17-2329>
18. Hawkins, R., Boyd, R.: Such stuff as dreams are made on: dream language, LIWC norms, and personality correlates. *Dreaming* **27** (2017). <https://doi.org/10.1037/drm0000049>
19. Oak, M., et al.: Generating clinically relevant texts: A case study on lifechanging events. In: *Proceedings of the Third Workshop on Computational Linguistics and Clinical Psychology*, pp. 85–94 (2016). <https://doi.org/10.18653/v1/W16-0309>
20. Lancashire, I., Hirst, G.: Vocabulary changes in Agatha Christie’s mysteries as an indication of Dementia: a case study. In: *Cognitive Aging: Research and Practice*, pp. 8–10. ser. *Cognitive Aging: Research and Practice*, Toronto (2009)

21. Pool, C., Nissim, M.: Distant supervision for emotion detection using Facebook reactions. In: Proceedings of the Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media (PEOPLES), Osaka, Japan, pp. 30–39 (2016). <https://aclanthology.info/papers/W16-4304/w16-4304>
22. Benikova, D., Wojatzki, M., Zesch, T.: What does this imply? Examining the impact of implicitness on the perception of hate speech. In: Rehm, G., Declerck, T. (eds.) GSCL 2017. LNCS (LNAI), vol. 10713, pp. 171–179. Springer, Cham (2018). [https://doi.org/10.1007/978-3-319-73706-5\\_14](https://doi.org/10.1007/978-3-319-73706-5_14). ISBN: 978-3-319-73706-5
23. Warner, W., Hirschberg, J.: Detecting hate speech on the World Wide Web. In: Proceedings of the Second Workshop on Language in Social Media, pp. 19–26. Association for Computational Linguistics, Montreal (2012). <http://dl.acm.org/citation.cfm?id=2390374.2390377>
24. Schmidt, A., Wiegand, M.: A survey on hate speech detection using natural language processing. In: Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media, pp. 1–10. Association for Computational Linguistics, Valencia (2017). <https://doi.org/10.18653/v1/W17-1101>
25. Pérez-Rosas, V., Mihalcea, R., Resnicow, K., Singh, S., An, L.: Building a motivational interviewing dataset. In: Proceedings of the Third Workshop on Computational Linguistics and Clinical Psychology, pp. 42–51 (2016). <https://doi.org/10.18653/v1/W16-0305>
26. Wolf, M., Horn, A., Mehl, M., Haug, S., Pennebaker, J., Kordy, H.: Computergestützte quantitative Textanalyse: Äquivalenz und Robustheit der deutschen Version des Linguistic Inquiry and Word Count. *Diagnostica* **54**, 85–98 (2008). <https://doi.org/10.1026/0012-1924.54.2.85>
27. Baayen, R., Piepenbrock, R., Rijn, H.: The CELEX lexical data base [CD-ROM Manual]. Linguistic Data Consortium, University of Pennsylvania, Pennsylvania (1993)
28. Fine, A., Frank, A.F., Jaeger, T.F., Van Durme, B.: Biases in predicting the human language model. In: Proceedings of the Conference on 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, Baltimore, MD, USA, vol. 2, pp. 7–12 (2014). <https://doi.org/10.3115/v1/P14-2002>
29. Stolcke, A.: SRILM - an extensible language modeling toolkit. In: Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002), Denver, CO, USA, vol. 2 (2004)
30. F. Pedregosa, et al.: Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011). <http://dl.acm.org/citation.cfm?id=1953048.2078195>. Accessed 24 April 2018. ISSN: 1532-4435
31. Jørgensen, R.N., Dale, P.S., Bleses, D., Fenson, L.: CLEX: across-linguistic lexical norms database\*. *J. Child Lang.* **37**(2), 419–428 (2010). <https://doi.org/10.1017/S0305000909009544>. ISSN: 1469-7602, 0305-0009
32. Vö, M., Conrad, M., Kuchinke, L., Urton, K., Hofmann, M., Jacobs, A.: The Berlin affective word list reloaded (BAWL-r). *Behav. Res. Methods* **41**, 534–538 (2009). <https://doi.org/10.3758/BRM.41.2.534>
33. Psychometric Inc: Toole, M.O. (ed.) The Free Dictionary (2018). <https://medical-dictionary.thefreedictionary.com/psychometric>
34. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Proceedings of Workshop at ICLR, Scottsdale, AZ, USA (2013)

35. Cho, K., et al.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: [ARXIV:1406.1078](#) [cs, stat], pp. 1724–1734. Association for Computational Linguistics, Doha (2014). [arXiv:1406.1078](#). <http://www.aclweb.org/anthology/D14-1179>
36. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997). <https://doi.org/10.1162/neco.1997.9.8.1735>. ISSN: 0899-7667
37. Scheffer, D., Kuhl, J.: Der Operante Motiv-Test (OMT): Ein neuer Ansatz zur Messung impliziter Motive. In: Stiensmeier, J., Rheinberg, F. (eds.) *Tests und Trends, Jahrbuch der psychologischen Diagnostik*, vol. N.F.2., pp. 129–150. Hogrefe Verlag, Göttingen (2003)



# LawStats – Large-Scale German Court Decision Evaluation Using Web Service Classifiers

Eugen Ruppert<sup>1(✉)</sup>, Dirk Hartung<sup>2</sup>, Phillip Sittig<sup>1</sup>, Tjorben Gschwander<sup>1</sup>,  
Lennart Rönneburg<sup>1</sup>, Tobias Killing<sup>1</sup>, and Chris Biemann<sup>1</sup>

<sup>1</sup> LT Group, MIN Faculty, Department of Computer Science, Universität Hamburg,  
Hamburg, Germany

{ruppert,5sittig,4gschwan,6roenneb,6killing,  
biemann}@informatik.uni-hamburg.de  
<https://lt.informatik.uni-hamburg.de/>

<sup>2</sup> Bucerius Law School, Hamburg, Germany  
[dirk.hartung@law-school.de](mailto:dirk.hartung@law-school.de)  
<https://www.law-school.de>

**Abstract.** *LawStats* provides quantitative insights into court decisions from the Bundesgerichtshof - Federal Court of Justice (BGH), the Federal Court of Justice in Germany. Using Watson Web Services and approaches from Sentiment Analysis (SA), we can automatically classify the revision outcome and offer statistics on judges, senates, previous instances etc. via faceted search. These statistics are accessible through a open web interface to aid law professionals. With a clear focus on interpretability, users can not only explore statistics, but can also understand, which sentences in the decision are responsible for the machine's decision; links to the original texts provide more context. This is the first large-scale application of Machine Learning (ML) based Natural Language Processing (NLP) for German in the analysis of ordinary court decisions in Germany that we are aware of. We have analyzed over 50,000 court decisions and extracted the outcomes and relevant entities. The modular architecture of the application allows continuous improvements of the ML model as more annotations become available over time. The tool can provide a critical foundation for further quantitative research in the legal domain and can be used as a proof-of-concept for similar efforts.

**Keywords:** Law domain · Web APIs · Text classification  
Cognitive services · Faceted search

## 1 Introduction

Legal professionals have become accustomed to the use of digital media and tools in their practice and Natural Language Processing (NLP) and Information Extraction (IE) generally offer a lot of potential benefits for many domains.



However, their application to the legal domain is extremely limited to date. The legal profession needs exact and correct decisions. Thus, many struggle to accept Machine Learning (ML) techniques with a reported performance below 100%. In digital systems, rule-based IE is still dominant as it offers a high precision. But while rule-based systems allow you to get detailed insights in a document collection, a meaningful understanding on document level is hardly achievable without ML methods. In addition, law is traditionally regarded as a normative and consensus-based science and only recently quantitative analysis and empirical methodology have become popular [5]. An aspiring school of thought classifies law as a complex adaptive system [13] and therefore deems technology absolutely necessary in order to tackle this complexity [14].

The project *LawStats* is the result of a collaboration between the Language Technology group at the University of Hamburg with the Bucerius Law School. Combining an entity extraction model trained by law students using the IBM Watson Knowledge Studio<sup>1</sup>, and a tool for Aspect-Based Sentiment Analysis (ABSA) [15], the *LawStats* application analyzes court decisions and offers a faceted search interface to aid law practitioners. Users can explore the court decision database from the Bundesgerichtshof - Federal Court of Justice (BGH). The web application offers facets for searching by judges, senates and lower courts like higher regional courts or district courts as well as by period of time. The user has the option to sort and search in all categories to look up information about court decisions and their components in a court decision database containing currently more than 50,000 court decisions. Users can upload and analyze additional court decision files to enlarge the database and test the application's analytical performance.

## 2 Related Work

The application of NLP tools and analysis on legal problems is a rather young area of research in Germany.<sup>2</sup> In the U.S., empirical and NLP-based analysis of court decisions has led to impressive results such as predictive modeling of Supreme Court decisions [8]. In Germany however, analysis of court decisions has so far been limited to special jurisdictions<sup>3</sup>, albeit with impressive results if ML techniques were used [20]. Our procedure is not aimed at court decision predictions and thereby differs from *Waltl's* approach [20]. We are also not using any pre-existing meta-data but extract all of the entities from the document text using our ML model and the outcome classification is solely based on a text classifier. In these regards our approach substantially differs from previous academic ventures in both method and mere size of the corpus.

Corpus Linguistic approaches to law studies exist as well [19], most notably the *JuReKo* corpus [4], and enable statistical analysis and evaluation [9]. These works are related to our paper as they use NLP techniques to analyze court

<sup>1</sup> <https://www.ibm.com/watson/services/knowledge-studio/>.

<sup>2</sup> For an introduction to computer assisted, linguistic research in law, see [18].

<sup>3</sup> For Labour Law, see [17].

decisions, they differ, however, substantially from our work as for them ML techniques have not played an important role so far.

IE of document collections is often performed in journalism.<sup>4</sup> Journalists search for Named Entities (NEs) and their relations in a corpus. Then, faceted search [16] is used instead of a simple keyword search, as it is more effective for professionals. Even though these frameworks offer impressive visualizations [3, 22], they cannot be used for document classification, as it would require training for particular domains. With a set law domain and expert annotators, we are able to perform polarity classification as well. Since the task is similar to SA [2, 11], we utilize a system originally developed for Sentiment Analysis (SA) and re-train it on our dataset annotations.

The presented system aids a Human in the Loop (HiL) working style, which is required for domains with (1) a lot of textual data and (2) the need for explainable ML classifications [6]. Professionals explore pre-annotated data using a faceted search interface and can add annotations. E.g., HiL is being employed in the biomedical domain [21], which needs an entity-centric access (bottom-up). In our use-case, we are concentrating on revision outcomes (top-down classification), that can be explored by different meta information.

### 3 Document Processing

#### 3.1 Pre-processing

We perform two pre-processing steps to enhance the quality of the training data, which translates into better performance of the resulting ML models. In a normalization step, we replace abbreviations and inconsistently formatted expressions with a standardized form to reduce sparsity in the model. This step is necessary as annotator time is limited and we are striving for a high recall in IE. Note that we perform preprocessing on the annotation set as well as on every other document that later enters the system to ensure consistency.

The second preprocessing step is to replace all dots that are not full stops. Since the Watson Knowledge Studio (WKS) has difficulties with German sentence splitting and over-segments document on abbreviation dots (such as *bzw.*), we use our own sentence splitter and replace all non-sentence-end dots with underscores. This is necessary since we heavily build on the notion of a sentence in our setup and annotation in WKS is currently only possible within sentence boundaries.

#### 3.2 Information Extraction

We extract and store the information from the BGH decisions. First, we analyze the document to determine the decision outcome, i.e. whether the revision was successful or rejected. Here, we make use of the particular structure of court decisions, as the operative provisions of decision are typically set at the beginning

<sup>4</sup> See, e.g. *Overview* (<https://www.overviewdocs.com/>) or *New/s/leak* [22].

of the document. To determine the verdict decision, we classify the first ten sentences of a decision and use the one with the highest confidence score as the indicator for the outcome.

Additionally, we extract the entities *Gericht (court)*, *Richter (judge)*, *Aktenzeichen (docket number)* and dates from the text. These are sorted to determine the relevant docket number, the correct procedural process and the temporal sequence. To determine the facets for the search interface, we combine these with the decision outcome. WKS is used to train the NE extraction model and to generate the training data for the outcome model in one annotation step (see Sect. 5.1).

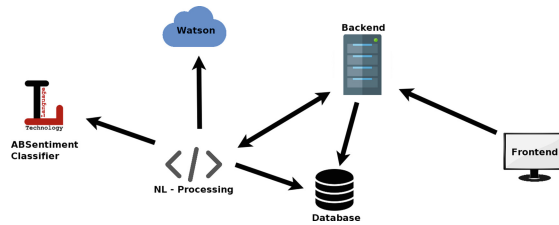


Fig. 1. System architecture of *LawStats*

## 4 System Architecture

**Overview.** The architecture of the application (Fig. 1) consists of a front-end website and a back-end web server using Spring Boot and Spring Data. Document storage is performed by an Apache Solr instance. For text analysis, we use a Java API<sup>5</sup> to send and receive data from Watson Natural Language Understanding (NLU) in order to extract relevant entities from the court decisions. The outcome of the decisions is determined by a text classifier (see Sect. 5.2 for details).

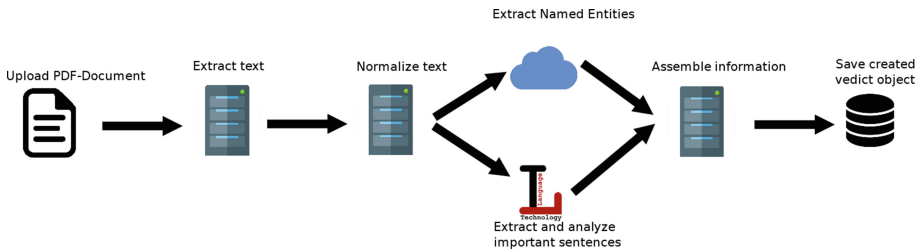


Fig. 2. Data flow pipeline

<sup>5</sup> <https://www.ibm.com/watson/developercloud/natural-language-understanding/>.

**Data Flow.** The data flow is presented in Fig. 2. Once the PDF verdict document is uploaded, the document text is extracted and a normalization of sentence boundaries and dates is performed. Then, we send the document to the Watson NLU API, while at the same time analyzing the verdict decision. After analysis, the verdict document is constructed from both sources and stored in the Solr index.

## 5 Machine Learning

### 5.1 Named Entity Recognition (NER)

For NER, we use the Watson NLU API with a custom model. Internally, WKS employs the Statistical Information and Relation Extraction (SIRE)<sup>6</sup> classifier for sequential annotation and extraction of entities. It works in a similar way to a standard Conditional Random Field (CRF) [10] by employing symbolic feature combinations.

**Annotation: Watson Knowledge Studio.** We define two different entity sets to be annotated by our team of seven domain experts. The first set contains all entities listed above (see Sect. 3.2). As courts and judges are finite and docket numbers and dates follow a definable pattern, we use dictionaries and regular expressions for pre-annotation. Our annotators have to correct false positives, limit annotations to relevant entities (i.e. not all courts, but only those, who were part of the procedural process) and annotate irregular mentions. Only annotations remaining after this manual step were used for training. Further, annotators identify the phrases used to indicate the outcome of the case. This task is done without pre-annotation. Our annotators could perform both tasks – correction and annotation – in one single pass.

In total, 1850 court decisions were annotated. The decisions were randomly sampled on the corpus. We set the Inter Annotator Agreement (IAA) threshold at 0.8<sup>7</sup> and have 20% of all documents annotated by at least two different annotators. Before training the entity extraction model and deploying it to NLU, we remove the phrase-based outcome expressions from the training data to avoid confusing the sequential classifier.

**Evaluation.** We use the WKS performance tool with a training set of 1260 documents, a dev set of 414 documents and a test set of 126 documents. The results in Table 1 show that the results are generally reliable with the exception of the extraction of court names, as their recall is only at 0.68. Since the document text is normalized, dates and docket numbers can be identified with a pattern feature extractor. Additionally, they mostly occur in very confined contexts, where

<sup>6</sup> <https://www.ibm.com/blogs/insights-on-business/government/relationship-extraction/>.

<sup>7</sup> Only documents with high-quality annotations are chosen for ML training.

**Table 1.** Evaluation of entity recognition

Entity	Precision	Recall	F <sub>1</sub>
Docket number	0.99	0.97	0.98
Date	0.99	0.91	0.95
Court	0.92	0.68	0.78
Judge	0.95	0.95	0.95

they are preceded by a few different keywords (e.g. “AktENZEICHEN”). Further investigation has shown that courts appear in two entirely different functions in the decisions: as the deciding court (our target) and as lists of courts involved in previous relevant jurisprudence. This problem could be solved by limiting the IE to particular sections of the decisions such as the beginning or the very end.

## 5.2 Revision Outcome Classification

To evaluate the revision outcome of a court decision, we classify single sentences into the classes “Revisionserfolg” (revision successful), “Revisionsmisserfolg” (revision not successful) and “irrelevant”. As described in Sect. 3.2, we take the first ten sentences of a decision, classify them independently and use the classification with the highest confidence score as the evaluation of the whole document. Here, we use an open-source text classification framework for German [15]<sup>8</sup>.

**Annotation and Training.** Training data is obtained from the WKS annotations. We extract the annotated sentences as well as a random set of irrelevant sentences and train a multi-class SVM [1] classifier. For the feature set, we compute TF-IDF (Term Frequency Inverse Document Frequency) scores and word embeddings [12] on an in-domain revision corpus. The corpus contains all BGH court decisions available online. We build a feature vector based on the TF-IDF values and concatenate it with the averaged word vectors in a sentence. Furthermore, we induce features on the training data. We obtain a list of 30 highest-scoring (TF-IDF) terms per label (positive, negative, irrelevant) and add the relative frequencies of these terms to the feature vector. The training data consists of 2,200 labeled sentences. We use a balanced ratio of sentences for the two classes of successful/non-successful revision and use twice as much of irrelevant sentences for training. For testing, we use 550 sentences.

**Evaluation** A simple baseline of choosing the majority class (irrelevant) scores 0.46 F<sub>1</sub>. When we train the classifier on a standard out-domain feature set<sup>9</sup>, we reach 0.70 F-score. By pre-training the TF-IDF vectors and the word2vec model on the in-domain collection of revision decisions, we reach a score of 0.91

<sup>8</sup> <https://github.com/uhh-It/LT-ABSA>.

<sup>9</sup> A news corpus is used for TF-IDF estimation, off-the-shelf German embeddings.

**Table 2.** Sentence-level evaluation of revision outcomes

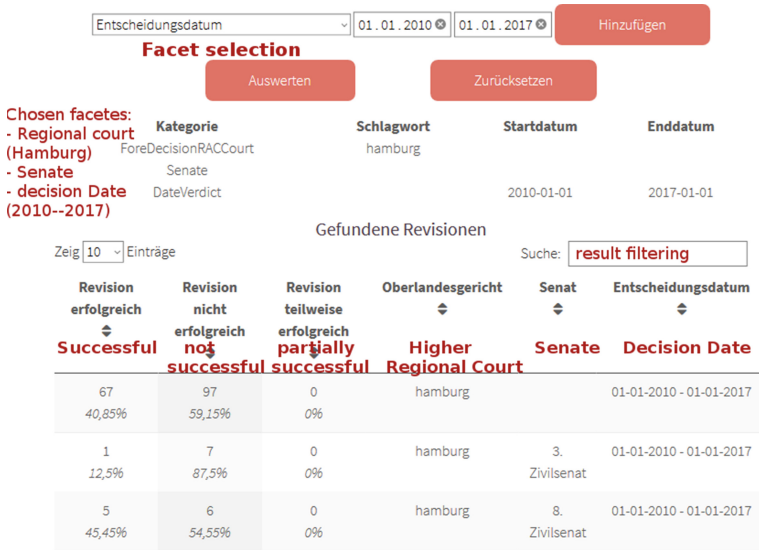
Classifier	Precision	Recall	F-score
Majority class baseline	.46	.46	.46
LT-ABSA out-domain	.71	.70	.70
LT-ABSA in-domain	.91	.91	.91

(see Table 2). Error analysis shows that the major factor limiting the performance is the strong similarity between sentences indicating a successful and an unsuccessful revision. In most documents, the long sentences follow a rigid structure. Variation in the expression of the final outcome requires additional training data. Especially the edge cases (partially successful) show a lot of variation in the verdict. Since we classify on document-level, we have performed a document-level evaluation of the revision outcomes to verify that our sentence extraction approach works as expected. Two expert annotators annotated 100 documents each. The possible error cases were wrong polarity (successful/not successful) and when the classifier picked an irrelevant sentence as the decision-bearing sentence. Results are presented in Table 3.

**Table 3.** Document-level evaluation of revision outcomes

Annotation set	Correct	Wrong	Irrelevant
Set 1	85	12	3
Set 2	88	11	1
Overall percentage	.87	.12	.02

With a precision of 0.87, we obtain a comparable performance as on the sentence level. Furthermore, the document selection features the same distribution as the training set (Fischer’s test  $p < 0.0001$ ), making it a representative sample of the complete collection. Error analysis of the incorrectly classified documents shows an even distribution. 12 documents were wrongly classified as “not successful” versus 11 “successful”. About a quarter of the wrongly classified documents are partly misclassified. E.g. the revision was partially successful but classified as “not successful”. For training, we had added the partly successful class to the positive “successful” class. In about 2% of the documents, the wrong sentence is selected by the classifier. These sentences are often short phrases containing judge names; the classifier learned their co-occurrence in training data. This could be alleviated by masking entities in this classification task.



**Fig. 3.** Faceted search showing revision outcomes of the Higher Regional Court (OLG) Hamburg, faceted by the deciding senate; time range 2010–2017.

## 6 User Interface

The publicly available web application can be divided into two main components: a web page where the user is able to upload his locally saved revisions and inspect them, and a section to filter and examine the existing database of revisions. On the upload page, external PDF revisions can be uploaded and analyzed. After the file has been analyzed, the result is added to the database and the user is redirected to a result page.

The application allows faceted search on metadata and automatically extracted information in the document collection. The user can search for judges, senates, the corresponding “Oberlandesgericht” (higher regional court equiv.), “Landesgericht” (state court equiv.), or “Amtsgericht” (district court equiv.) decisions as well as the docket number (see Fig. 3). To assess diachronic developments of revision outcomes, users can search for a timespan in which the revisions or their respective previous court decisions were decided. To enable exploratory searches and comparison of verdict decisions by facet, facetes can be selected without query terms. Then, the application returns e.g. revision outcome statistics for all judges, courts, etc. Combinations of fields can be used here as well. The results page contains all extracted information about a decision such as courts, judges, etc. of the verdict file. Additionally, the page contains the classified revision outcome, the confidence score, and the sentence that determined the evaluation.

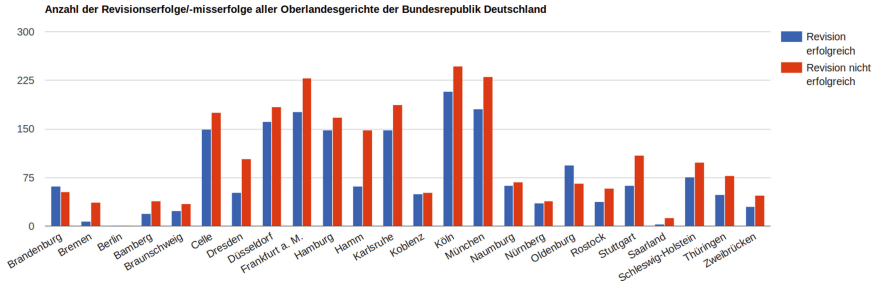


Fig. 4. Statistics, overview of successful vs. unsuccessful revisions per originating court.

## 7 Conclusion

In this application paper, we have presented an application to access a large-scale corpus of BGH decisions, which is explorable by law professionals and publicly available online.<sup>10</sup> We have demonstrated that the most interesting part of a decision – the result – can be quite reliably determined using ML techniques for very large corpora of decisions. In future work, we would like to more tightly integrate the loop of annotation, model update and classification in order to enable a setup, in which the model can continuously improve on the basis of user’s corrections in a human-in-the-loop setup. We plan to verify that the results are solid and helpful and ensure that our system aids professionals by reliable classification [7] (Fig. 4).

While techniques in this work are rather standard, the value of this work lies in enabling a new field of application: an immense genuine added value from this application could be created with a thorough statistical analysis of factors correlating with success in front of the Federal Supreme Court. For this purpose, the quality of the entity extraction and the classification ought to be improved by different approaches and additional training. But even already now, the data set compiled using this application can be structured and analyzed profoundly by interdisciplinary teams. Both the confirmation of known influences like procedure types and yet unknown factors, e.g. duration of proceedings or geographical origin of the cases, would be an interesting starting point for substantial unprecedented large-scale legal analysis.

**Acknowledgements.** We would like to thank Ming-Hao Bobby Wu and Tim Fischer for their help in document conversion. We also would like to thank International Business Machines Corporation (IBM) for their ongoing support of the project.

<sup>10</sup> <http://ltdemos.informatik.uni-hamburg.de/lawstats/> – Source code is available under a permissive license at <https://github.com/Kirikaku/LawStats>.



## References

1. Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, COLT 1992, pp. 144–152. ACM, New York (1992)
2. Cambria, E., Schuller, B., Xia, Y., Havasi, C.: New avenues in opinion mining and sentiment analysis. *IEEE Intell. Syst.* **28**(2), 15–21 (2013)
3. Dörk, M., Riche, N.H., Ramos, G., Dumais, S.: Pivotpaths: strolling through faceted information spaces. *IEEE Trans. Vis. Comput. Graph.* **18**(12), 2709–2718 (2012)
4. Gauer, I., Hamann, H., Vogel, F.: Das juristische Referenzkorpus (JuReko) - Computergestützte Rechtslinguistik als empirischer Beitrag zu Gesetzgebung und Justiz. In: DHd 2016: Modellierung - Vernetzung - Visualisierung, Leipzig, Germany, pp. 129–131 (2016)
5. Hamann, H.: Evidenzbasierte Jurisprudenz: Methoden empirischer Forschung und ihr Erkenntniswert für das Recht am Beispiel des Gesellschaftsrechts. Mohr Siebeck, Heidelberg (2014)
6. Holzinger, A., Biemann, C., Pattichis, C.S., Kell, D.B.: What do we need to build explainable AI systems for the medical domain? *CoRR* abs/1712.09923 (2017)
7. Holzinger, K., Mak, K., Kieseberg, P., Holzinger, A.: Can we trust machine learning results? artificial intelligence in safety-critical decision support. *ERCIM News* **112**(1), 42–43 (2018)
8. Katz, D.M., Bommarito, M.J., Blackman, J.: A general approach for predicting the behavior of the Supreme Court of the United States. *PLoS ONE* **12**(4) (2017)
9. Kuhn, F.: Zugänge zur Rechtssemantik, chap. Inhaltliche Erschließung von Rechtsdokumenten auf Grundlage von Automaten. Walter de Gruyter, Berlin/New York (2015)
10. Lafferty, J., McCallum, A., Pereira, F.C.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of the ICML 2001, Williamstown, MA, USA, pp. 282–289 (2001)
11. Liu, B.: Sentiment analysis and subjectivity. In: Handbook of Natural Language Processing, vol. 2, pp. 627–666 (2010)
12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Workshop at International Conference on Learning Representations (ICLR), Scottsdale, AZ, USA, pp. 1310–1318 (2013)
13. Ruhl, J.B.: Law’s complexity - a primer. *Georgia State Univ. Law Rev.* **24**(4) (2008). <http://ssrn.com/abstract=1153514>
14. Ruhl, J.B., Katz, D.M., Bommarito, M.J.: Harnessing legal complexity. *Sci. Mag.* **355**(6332), 1377–1378 (2017). <https://doi.org/10.1126/science.aag3013>
15. Ruppert, E., Kumar, A., Biemann, C.: LT-ABSA: an extensible open-source system for document-level and aspect-based sentiment analysis. In: Proceedings of the GSCL GermEval Shared Task on Aspect-based Sentiment in Social Media Customer Feedback, Berlin, Germany, pp. 55–60 (2017)
16. Tunkelang, D.: Faceted search. *Synth. Lect. Inf. Concepts Retrieval Serv.* **1**(1), 1–80 (2009)
17. Vogel, F., Christensen, R., Pötters, S.: Richterrecht der Arbeit - empirisch untersucht. Möglichkeiten und Grenzen computergestützter Textanalyse am Beispiel des Arbeitnehmerbegriffs. Duncker & Humblot, Berlin (2015)
18. Vogel, F., Hamann, H., Gauer, J.: Computer-assisted legal linguistics: corpus analysis as a new tool for legal studies (2017). <https://doi.org/10.1111/lsi.12305>

19. Vogel, F.: The pragmatic turn in law. Inference and Interpretation, chap. Calculating legal meanings? Drawbacks and opportunities of corpus assisted legal linguistics to make the law (more) explicit. Mouton de Gruyter, New York, Boston (2017)
20. Watl, B., Bonczek, G., Scepankova, E., Landthaler, J., Matthes, F.: Predicting the outcome of appeal decisions in germany's tax law. In: International Federation for Information Processing (IFIP): Policy Modeling and Policy Informatics. St. Petersburg, Russia (2017)
21. Yimam, S.M., Remus, S., Panchenko, A., Holzinger, A., Biemann, C.: Entity-centric information access with the human-in-the-loop for the biomedical domains. In: Biomedical NLP Workshop Associated with RANLP 2017, Varna, Bulgaria, pp. 42–48 (2016)
22. Yimam, S., et al.: new/s/leak - information extraction and visualization for an investigative data journalists. In: ACL 2016 Demo Session, Berlin, Germany, pp. 163–168 (2016)



# Clinical Text Mining for Context Sequences Identification

Svetla Boytcheva<sup>(✉)</sup> 

Institute of Information and Communication Technologies, Bulgarian Academy of Sciences, Sofia, Bulgaria  
svetla.boytcheva@gmail.com

**Abstract.** This paper presents an approach based on sequence mining for identification of context models of diseases described by different medical specialists in clinical text. Clinical narratives contain rich medical terminology, specific abbreviations, and various numerical values. Usually raw clinical texts contain too many typos. Due to the telegraphic style of the text and incomplete sentences, the general part of speech taggers and syntax parsers are not efficient in text processing of non-English clinical text. The proposed approach is language independent. Thus, the method is suitable for processing clinical texts in low resource languages. The experiments are done on pseudonimized outpatient records in Bulgarian language produced by four different specialists for the same cohort of patients suffering from similar disorders. The results show that from the clinical documents can be identified the specialty of the physician. Even the close vocabulary is used in the patient status description there are slight differences in the language used by different physicians. The depth and the details of the description allow to determine different aspects and to identify the focus in the text. The proposed data driven approach will help for automatic clinical text classification depending on the specialty of the physician who wrote the document. The experimental results show high precision and recall in classification task for all classes of specialist represented in the dataset. The comparison of the proposed method with bag of words method show some improvement of the results in document classification task.

**Keywords:** Data mining · Text mining · Health informatics

## 1 Motivation

Healthcare is data intensive domain. Large amount of patient data are generated on daily base. However, more than 80% of this information is stored in non structured format - as clinical texts. Usually clinical narratives contain description with sentences in telegraphic style, non-unified abbreviation, many typos, lack of punctuation, concatenated words, etc. It is not straightforward how patient data can be extracted in structured format from such messy data. Natural language

processing (NLP) of non-English clinical text is quite challenging task due to lack of resources and NLP tools [11]. There are still non existing translations of SNOMED<sup>1</sup>, Medical Subject Headings (MeSH)<sup>2</sup> and Unified Medical Language System (UMLS)<sup>3</sup> for the majority of languages.

Clinical texts contain complex descriptions of events. Investigating the cumulative result of all events over the patient status require more detailed study of different ways of their description. All physicians use common vocabulary and terminology to describe organs and systems during the human body observation but tend to use different description depending on their specialty. Analyzing complex relations between clinical events will help to prove different hypothesis in healthcare and automatically to generate context models for patient status associated to diagnoses. This is very important in epidemiology and will help monitoring some chronic diseases' complications on different stages of their development. The chronic disease with highest prevalence are cardiovascular diseases, cancer, chronic respiratory diseases and diabetes<sup>4</sup>. The complications of these chronic diseases develop over time and they are with high socioeconomic impact and the main reason for over than 70% of mortality cases. In this paper are presented some results for processing data of patients with Diabetes Mellitus type 2 (T2DM), Schizophrenia (SCH) and Chronic Obstructive Pulmonary Disease (COPD).

We show that data mining and text mining are efficient techniques for identification of complex relations in clinical text.

The main goal of this research is to examine differences and specificity in patient status description produced by different medical specialists. The proposed data-driven approach is used for automatic generation of context models for patient status associated with some chronic diseases. The approach is language independent. An application of the context sequences in used for clinical text classification depending on the specialty of the physician who wrote the document.

The paper is structured as follows: Sect. 2 briefly overviews the research in the area; Sect. 3 describes the data collections of clinical text used in the experiments; Sect. 4 presents the theoretical background and formal presentation of the problem; Sect. 5 describes in details the proposed data mining method for context models generation from clinical text; Sect. 6 shows experimental results and discusses the method application in clinical texts classification; Sect. 7 contains the conclusion and sketches some plans for future work.

## 2 Related Work

Data mining methods are widely used in clinical data analyses both for structured data and free text [17]. There are two types of frequent patterns mining –

<sup>1</sup> SNOMED, <https://www.snomed.org/>.

<sup>2</sup> Medical Subject Headings – MESH, <https://www.nlm.nih.gov/mesh/>.

<sup>3</sup> UMLS, <https://www.nlm.nih.gov/research/umls/>.

<sup>4</sup> World Health Organization (WHO) fact sheets, <http://www.who.int/mediacentre/factsheets/fs355/en/>.

frequent itemsets patterns mining (FPM) and frequent sequence mining (FSM). In the first approach the order of items does not matter, and in the second one the order does matter.

In context modeling task there is some research for other domains. Ziemiński [19] proposes a method that initially generates context models from small collections of data and later summarizes them in more general models. Rabatel et al. [14] describes a method for mining sequential patterns in marketing domain taking into account not only the transactions that have been made but also various attributes associated with customers, like age, gender and etc. They initially uses classical data mining method for structured data and later is added context information exploring the attributes with hierarchical organization.

Context models in FPM are usually based on some ontologies. Huang et al. [7] present two semantics-driven FPM algorithms for adverse drug effects prevention and prediction by processing Electronic Health Records (EHR) The first algorithm is based on EHR domain ontologies and semantic data annotation with metadata. The second algorithm uses semantic hypergraph-based k-itemset generation. Jensen et al. [8] describe a method for free text in Electronic Health Records (EHR) processing in Norwegian language. They are using NOR-MeSH for estimation of disease trajectories of the cancer patients.

One of the major problems with clinical data repositories is that they contain in-complete data about the patient history. Another problem is that the raw data are too noisy and needs significant efforts for preprocessing and cleaning. The timestamps of the events are uncertain, because the physicians don't know the exact occurrence time of some events. There can be a significant gap between the onset of some dis-eases and the first record for diagnosis in EHR made by the physician. Thus a FPM method for dealing with temporal uncertainty was proposed by Ge et al. [4]. It is hard to select representative small collections of clinical narratives, because there is a huge diversity of patient status descriptions. Some approaches use frequent pattern's mining (FPM) considering the text as bag-of-words and losing all grammatical information.

The majority of FSM and FPM applications in Health informatics are for patterns identification in structured data. Wright et al. [16] present a method for prediction of the next prescribed drug in patient treatment. They use CSPADE algorithm for FSM of diabetes medication prescriptions. Patniak et al. [12] present mining system called EMRView for identifying and visualizing partial order information from EHR, more particularly ICD-10 codes.

But there are also applications of FSM for textual data. Plantevit et al. [13] present a method for FSM for Biomedical named entity recognition task.

There are developed a variety of techniques for FPM and FSM task solution. Some of them are temporal abstraction approach for medical temporal patterns discovery, one-sided constitutional nonnegative matrix factorization, and symbolic aggregate approximation [15].

Healthcare is considered as data-intensive domain and as such faces the challenges of big data processing problems. Krumholz [10] discusses the potential

and importance of harnessing big data in healthcare for prediction, prevention and improvement of healthcare decision making.

In the classification task there are used successfully many artificial intelligence (AI) approaches [9] with high accuracy: neural networks, naive Bayes classifiers, support vector machines, etc. The main reason for choosing FSM method is than in healthcare data processing the most important feature of the used method is the result to be explainable, e.i. so called “Explainable AI” [5]. This will make the decision making process more transparent.

### 3 Materials

For experiments is used a data collections of outpatient records (ORs) from Bulgarian National Diabetes Register [2].

They are generated from a data repository of about 262 million pseudonimized outpatient records (ORs) submitted to the Bulgarian National Health Insurance Fund (NHIF) in period 2010–2016 for more than 5 million citizens yearly. The NHIF collects for reimbursement purpose all ORs produced by General Practitioners and the Specialists from Ambulatory Care for every patient clinical visit. The NHIF collects for reimbursement purpose all ORs produced by General Practitioners and the Specialists from Ambulatory Care for every patient clinical visit. The collections used for experiments contain ORs produced by the following specialists: Otolaryngology (S14), Pulmology (S19), Endocrinology (S05), and General Practitioners (S00).

**Table 1.** Fields with free text in ORs that supply data for data mining components

XML field	Content
Anamnesis	Disease history, previous treatments, family history, risk factors
Status	Patient state, height, weight, BMI, blood pressure etc.
Clinical tests	Values of clinical examinations and lab data listed in arbitrary order
Prescribed treatment	Codes of drugs reimbursed by NHIF, free text descriptions of other drugs and dietary recommendations

ORs are stored in the repository as semi-structured files with predefined XML-format. Structured information describe the necessary data for health management like visit date and time; pseudonimized personal data and visit-related information, demographic data (age, gender, and demographic region), etc. All diagnoses are presented by ICD–10<sup>5</sup> codes and the name according to the standard nomenclature. The most important information concerning patient status and case history is provided like free text.

<sup>5</sup> <http://apps.who.int/classifications/icd10/browse/2016/en>.

For all experiments are used raw ORs, without any preprocessing due to the lack of resources and annotated corpora. The text style for unstructured information is telegraphic. Usually with no punctuation and a lot of noise (some words are concatenated; there are many typos, syntax errors, etc.). The Bulgarian ORs contain medical terminology both in Latin and Bulgarian. Some of the Latin terminology is also used with Cyrillic transcription.

The most important information concerning patient status and case history is provided like free text. ORs contain paragraphs of unstructured text provided as separate XML tags (see Table 1): “Anamnesis”, “Status”, “Clinical tests”, and “Prescribed treatment”.

## 4 Theoretical Background

Let’s consider each patient clinic visits (i.e. OR) as a single event. For the collection  $S$  we extract the set of all different events. Let  $E = \{e_1, e_2, \dots, e_k\}$  be the set of all possible patient events. The vocabulary  $W = \{w_1, w_2, \dots, w_n\}$ , used in all events  $E$  in  $S$  will be called *items*, where  $e_i \subseteq W$ ,  $1 \leq i \leq N$ . Lets  $P = \{p_1, p_2, \dots, p_N\}$  be the set of all different patient identifiers in  $S$ . The associated unique transaction identifiers (tids) shall be called *pids* (*patient identifiers*).

Let each sentence in a clinical text  $e_1$  is splitted on a sequence of tokens  $X \subseteq W$ .  $X$  is called an *itemset*. For each itemset  $X$  is generated a vector (sequence)  $v = \langle v_1, v_2, \dots, v_m \rangle$ , where  $v_i \in W$ ,  $1 \leq i \leq N$ . The length of a sequence  $v$  is  $m$  (the number of tokens), denoted  $len(v) = m$ . We denote  $\emptyset$  the empty sequence (with length zero, i.e.  $len(\emptyset) = 0$ ).

Let  $D \subseteq P \times E$  be the set of all sequences in collection in the format  $\langle pid, sequence \rangle$ . We will call  $D$  *database*.

Let  $p = \langle p_1, p_2, \dots, p_m \rangle$  and  $q = \langle q_1, q_2, \dots, q_t \rangle$  be two sequences over  $W$ . We say that  $q$  is *subsequence of*  $p$  denoted by  $q \subseteq p$ , if there exists one-to-one mapping:  $\theta: [1, t] \rightarrow [1, m]$ , such that  $q_i = p_{\theta(i)}$  and for any two positions  $i$ , and  $j$  in  $q$ ,  $i < j \Rightarrow \theta(i) < \theta(j)$ .

Each sequential pattern is a sequence. A sequence  $A = X_1, X_2, \dots, X_m$ , where  $X_1, X_2, \dots, X_m$  are itemsets is said to *occur in another sequence*  $B = Y_1, Y_2, \dots, Y_t$ , where  $Y_1, Y_2, \dots, Y_t$  are itemsets, if and only if there exist integers  $1 \leq i_1 < i_2 \dots < i_k \leq t$  such that  $X_1 \subseteq Y_{i_1}, X_2 \subseteq Y_{i_2}, \dots, X_m \subseteq Y_{i_m}$ .

Let  $D$  is a database and  $Q \subseteq E$  is a sequential pattern. The *support* of a sequential pattern  $Q$ , denoted  $support(Q)$  is the number of sequences where the pattern occurs divided by the total number of sequences in the database.

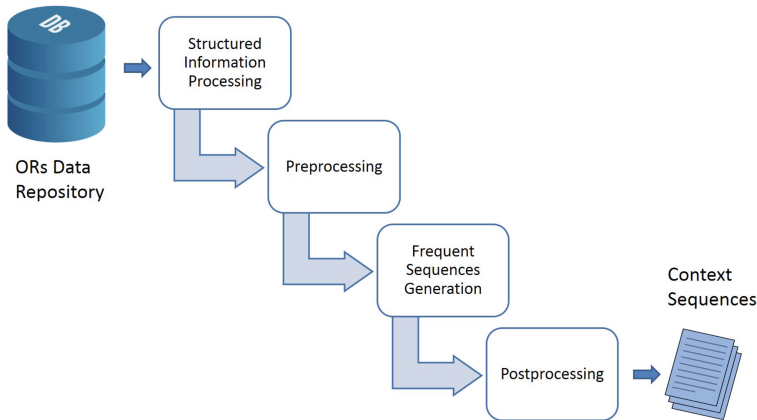
We define minimal support threshold *minsup* – a real number in the range  $[0, 1]$ . A frequent sequential pattern is a sequential pattern having a support no less than *minsup*.

In our task we are looking only for frequent sequential pattern for given *minsup*.

## 5 Method

Initially we generate collections  $S_1, S_2, \dots, S_r$  of ORs from the repository, using the structured information data for specialists who wrote them. We define vocabularies  $W_1, W_2, \dots, W_r$ .

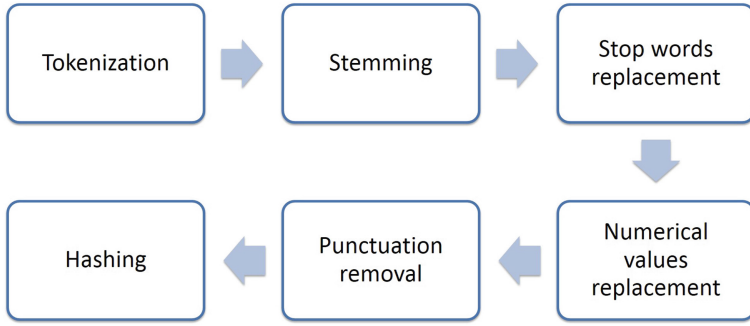
The collections processing is organized as pipeline (see Fig. 1). The first step is to split each collection on two subsets – one that contain only “Anamnesis” for patients ( $SA_i$ ) and the other  $SH_i$  – for their “Status”. Each of these subsets will be processed independently. We define for all collections vocabularies  $WA_1, WA_2, \dots, WA_r$  and  $WH_1, WH_2, \dots, WH_r$  for each of these subsets correspondingly.



**Fig. 1.** Pipeline for automatic context sequences generation

The next step converts free text from ORs into database (see Fig. 2). After tokenization is applied stemming. All stop words are replaced by terminal symbol *STOP*. ORs contain many numerical values, like clinical test results, vitals (Body Mass Index, Riva Roci – blood pressure), etc. Numerical values are replaced by terminal symbol *NUM*. The sentences have mainly telegraphic style, or the information is described as sequence of phrases separated by semicolon. We consider those phrases as *sentences*. Sentence splitting is applied to construct sequences of itemsets for each document. In this process all additional punctuation is removed. To separate the sentences is used negative number -1, and -2 is used to denote the end of the text. The last stage of the preprocessing is hashing, which purpose is to speed-up the process of frequent sequence mining. In the hashing phase each word is replaced by unique numerical ID.



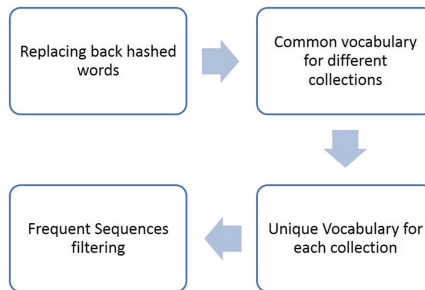


**Fig. 2.** Pipeline for preprocessing of free-text in outpatient record

For frequent sequence mining is used algorithm CM-SPAM [3], more efficient variation of SPAM algorithm [1], that is considered as one of the fastest algorithms for sequential mining. CM-SPAM is even faster than SPAM, but more important is that CM-SPAM is more efficient for low values of *minsup*. This is important, because in clinical text some cases are not so frequent, because the prevalence of the diseases is usually lower in comparison with other domains. The *minsup* values for clinical data are usually in the range [0.01,0.1].

The last step is the postprocessing phase (see Fig. 3) that starts with replacing back the hashed words. Then we identify unique vocabulary for each collection:

$$\begin{aligned}
 WAU_i &= WA_i - \bigcup_{j \neq i} WA_j - \bigcup_j WH_j \\
 WHU_i &= WH_i - \bigcup_{j \neq i} WH_j - \bigcup_j WA_j
 \end{aligned}$$



**Fig. 3.** Pipeline for postprocessing of the generated frequent sequences

Let  $FA_1, FA_2, \dots, FA_r$  and  $FH_1, FH_2, \dots, FH_r$  are the frequent sequences generated on step 3. We need to filter all sequences that occur in any sequence of the other sets or a frequent sequence from other collection occur in them.

$$\begin{aligned}
 FFA_i &= \{Z | Z \in FA_i \wedge \exists j \neq i \ Y \in FA_j \vee Y \in FH_j \\
 &\quad \text{such that } Y \subseteq Z \vee Z \subseteq Y \wedge \exists X \in FA_i \ X \subseteq Z\} \\
 FFH_i &= \{Z | Z \in FH_i \wedge \exists j \neq i \ Y \in FA_j \vee Y \in FH_j \\
 &\quad \text{such that } Y \subseteq Z \vee Z \subseteq Y \wedge \exists X \in FH_i \ X \subseteq Z\}
 \end{aligned}$$

The so filtered frequent sequences sets together with unique words form the specific terminology and sub-language used by different specialist in patient disease history and status description.

## 6 Experiments and Results

For a cohort of 300 patients suffering from T2DM and COPD are extracted ORs for all their clinical visits in 3 year period (2012–2014) to different specialists: Otolaryngology (S14), Pulmology (S19), Endocrinology (S05), and General Practitioners (S00). After preprocessing of ORs in all collections are separately extracted Anamnesis and Status descriptions for each patient (Tables 2 and 3).

The minsup value were set as relative minsup function of the ration between the number of patients and ORs. It is approximately 0.02% for the smallest set SA14, 0.03% for SA05 and SA19 and 0.1% for the largest set SA00. This is a rather small minsup value that will guarantee coverage even for more rare cases but with sufficient support. For Status subset the minsup value were set in similar range – 0.05% for SH14 and SH19, 0.08% for SH05 and 0.09% for SH00.

All subsets are processed with CM-SPAM for frequent sequences mining. In addition the algorithm dEclat [18] for frequent itemsets mining was applied. The frequent itemsets were filtered with similar method as frequent sequences (see Tables 2 and 3). For experiments are used Java implementations of the algorithms from SPFM (Open-Source Data Mining Library) <sup>6</sup>.

The datasets for Anamnesis are sparse, because they contain descriptions of different patient diseases history, complaints, and risk factors. Thus the diversity of explanations causes lower number of generated frequent sequences and higher number of unique vocabulary (see Fig. 4 and Table 2). The unique vocabulary contain different complaints and many informal words for their explanation. Although the set SA00 is larger than the other sets for this set are generated lower number of frequent sequences. This set corresponds to the ORs written by general practitioners, who usually observe larger set of diseases than other specialists. The set SA05 contains more consistent information about the T2DM complaints only.

<sup>6</sup> SPFM, <http://www.philippe-fournier-viger.com/spmf/index.php?link=algorithms.php>.

**Table 2.** Frequent sequences in Anamnesis section

	SA00	SA05	SA19	SA14
ORs	11,345	798	532	156
Patients	294	195	70	173
Items/Vocabulary	4,337	1,767	1,527	447
minsup	0.1 (131)	0.03 (24)	0.02(11)	0.03(5)
Frequent Sequences	1,713	23,677	8,250	6,643
Filtered Sequences	1,358	23,327	7,932	6,527
Frequent Itemsets	80	1,815	477	200
Filtered Itemsets	37	1,732	396	178
Unique words	2,923	747	628	144

**Table 3.** Frequent sequences in Status section

	SH00	SH05	SH19	SH14
ORs	11,345	798	532	156
Patients	294	195	70	173
Items/Vocabulary	3,412	1,131	700	627
minsup	0.09 (1,022)	0.08 (64)	0.05 (27)	0.05 (8)
Frequent Sequences	107,267	27,949	26,341	345
Filtered Sequences	106,634	27,185	25,670	321
Frequent Itemsets	31,902	7,176	2,224	30
Filtered Itemsets	30,462	5,551	1,467	22
Unique words	2,422	391	195	346

In contrast the datasets for Status are dense, because they contain predefined set of organs and systems status description. The Status explanation usually contains phrases rather than sentences. Each phrase describes single organ/system and its current condition. The similarity between Status explanations causes significant growth of the number of generated frequent sequences and lower number of unique vocabulary (see Fig. 5 and Table 3). Although the higher number of the generated frequent sequences during the filtering process they shrink faster, because contain similar subsequences. The unique vocabulary contains specific terminology for some organs and systems that are in main focus and interest for the physician that makes the medical examination. The result set of context sequence contain only specific sub-language used from specialists in their area.

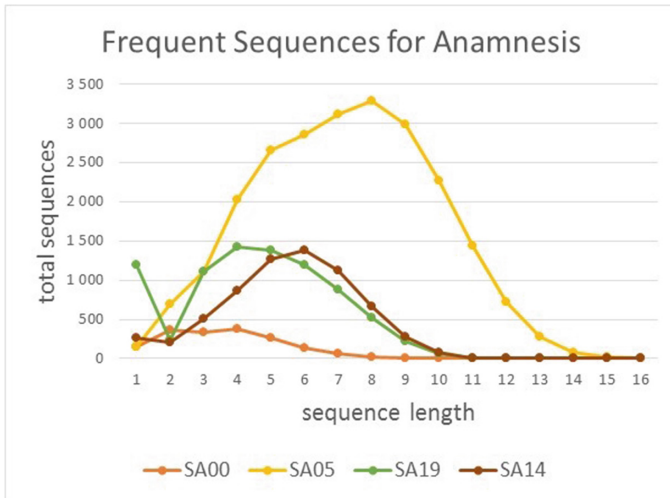


Fig. 4. Generated frequent sequences for Anamnesis section grouped by length

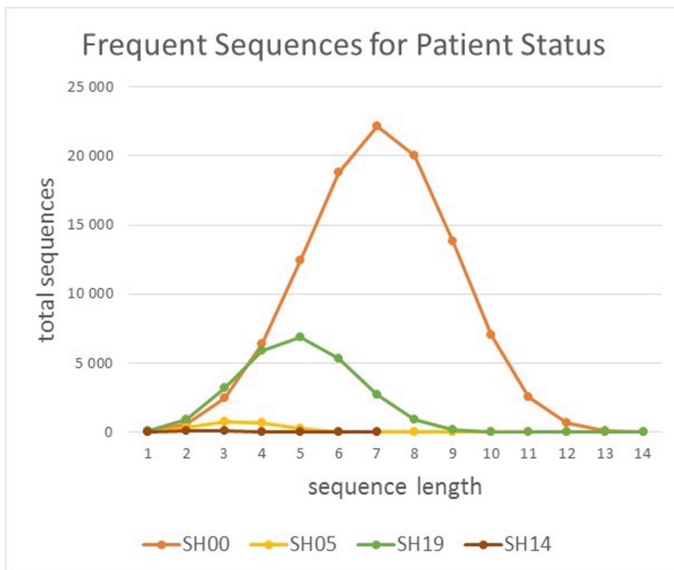


Fig. 5. Generated frequent sequences for Status section grouped by length

The extracted frequent sequences and frequent itemsets are used for multi class text classification. Experiments are provided by non-exhaustive cross-validation (5 iterations on sets in ratio 7:1 training to test). For comparison of the obtained results is used bag of words (BOW) method by applying frequent itemsests generated by dEclat algorithm.

The classification is based on unique vocabulary used for classes and on the filtered sequences and frequent itemsets from all classes that match the text. As golden standard in the evaluation are used specialty codes from ORs structured data.

Six types of experiments are performed. In the first task are used subsets for Anamnesis section for all four specialty classes 00, 05, 14 and 19. The evaluation results (Table 4) for F1 measure ( $F1 = 2 * Precision * Recall / (Precision + Recall)$ ) show that context sequences method outperforms BOW method for all classes, except class 19 for Anamnesis subsets. The evaluation for Status section classification is just the opposite (Table 5). BOW method shows better results than context sequences. The main reason is that Status section is written in telegraphic style with phrases rather than full sentences. Usually Status section contains sequence of attribute-value (*A-V*) pairs - anatomical organ/system and its status/condition.

**Table 4.** Evaluation of rules for Anamnesis for S00, S05, S14 and S19

	Context sequences				BOW			
	SA00	SA05	SA14	SA19	SA00	SA05	SA14	SA19
Precision	0.9986	0.3674	0.8707	0.6130	0.9997	0.1872	0.7785	0.7804
Recall	0.8574	0.9848	0.8205	0.9568	0.6944	0.9975	0.7436	0.8684
F1	0.9226	0.5351	0.8449	0.7473	0.8195	0.3152	0.7607	0.8221

General practitioners used in ORs terminology and phrases that can be found in ORs for all specialties. Thus the class 00 is not disjoint with classes 05, 14 and 19. Class 00 is one of the main reasons for misclassification. Another experiment was performed with “pure” classes – including only 05, 14, and 19 (Table 6). The F1-measure values show better performance in classification task for Anamnesis for all classes, in compassion with BOW method. For Status tast the results for both methods are comparable (Table 7).

Finally the classification of both sections – Anamnesis and Status is used for classification of the outpatient record as a whole document. The evaluation results (Table 8) show that results for context sequences drop down and BOW method performance is better. After eliminating the noisy set S00 - the result (Table 9) for context sequences method significantly improve and outperform BOW method for all three classes 05, 14 and 19.

**Table 5.** Evaluation of rules for Status for S00, S05, S14 and S19

	Context sequences				BOW			
	SH00	SH05	SH14	SH19	SH00	SH05	SH14	SH19
Precision	0.9990	0.5551	0.9560	0.2420	0.9750	0.6105	1.0000	0.8954
Recall	0.8108	0.9010	0.9744	0.9925	0.9653	0.7995	0.9487	0.6891
F1	0.8951	0.6870	0.9651	0.3891	0.9701	0.6923	0.9737	0.7788

**Table 6.** Evaluation of rules for Anamnesis for S05, S14 and S19

	Context sequences			BOW		
	SA05	SA14	SA19	SA05	SA14	SA19
Precision	0.9581	1.0000	0.9714	0.8803	1.0000	0.9935
Recall	0.9873	0.8312	0.9751	0.9987	0.7436	0.8701
F1	0.9725	0.9078	0.9733	0.9358	0.8529	0.9277

**Table 7.** Evaluation of rules for Status for S05, S14 and S19

	Context Sequences			BOW		
	SH05	SH14	SH19	SH05	SH14	SH19
Precision	0.9902	1.0000	0.8829	0.9251	1.0000	1.0000
Recall	0.9103	0.9744	0.9944	1.0000	0.9610	0.8910
F1	0.9486	0.9870	0.9353	0.9611	0.9801	0.9424

**Table 8.** Evaluation of rules for ORs for S00, S05, S14 and S19

	Context sequences				BOW			
	S00	S05	S14	S19	S00	S05	S14	S19
Precision	0.9999	0.6958	0.9750	0.6251	0.9979	0.8356	0.9935	0.9618
Recall	0.9428	0.9975	1.0000	1.0000	0.9871	1.0000	0.9809	0.9097
F1	0.9705	0.8198	0.9873	0.7693	0.9924	0.9105	0.9872	0.9351

**Table 9.** Evaluation of rules for ORs for S05, S14 and S19

	Context sequences			BOW		
	S05	S14	S19	S05	S14	S19
Precision	1.0000	1.0000	0.9981	0.9645	1.0000	1.0000
Recall	0.9987	1.0000	1.0000	1.0000	0.9872	0.9492
F1	0.9994	1.0000	0.9991	0.9819	0.9935	0.9739

## 7 Conclusion and Further Work

The proposed data-driven method is based on data mining techniques for context sequences identification in clinical text depending on medical specialty of the doctor. The method is language independent and can be used for low resource languages. The huge number of generated frequent sequences is reduced during the filtering process. The experimental results show that context sequences methods outperforms BOW method for sparse datasets in classification task.

Using “human-in-the-loop” [6] approach some further analyses of the significance for the domain of the generated frequent sequences and the misclassified documents will be beneficial. The space of clinical events is too complex. Thus “human-in-the-loop” can be applied also for subclustering task by using patient age, gender and demographic information. Reducing the dimensionality will help to determine different context sequences depending on the patient phenotype.

As further work can be mentioned also the task for context sequences similarities measuring. It can be used to identify synonyms and semantically close phrases.

**Acknowledgments.** This research is partially supported by the grant Specialized Data Mining Methods Based on Semantic Attributes (IZIDA), funded by the Bulgarian National Science Fund in 2017–2019. The author acknowledges the support of Medical University - Sofia, the Bulgarian Ministry of Health and the Bulgarian National Health Insurance Fund.

## References

1. Ayres, J., Flannick, J., Gehrke, J., Yiu, T.: Sequential pattern mining using a bitmap representation. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 429–435. ACM (2002)
2. Boytcheva, S., Angelova, G., Angelov, Z., Tcharaktchiev, D.: Integrating data analysis tools for better treatment of diabetic patients. In: CEUR Workshop Proceedings, vol. 2022, pp. 229–236 (2017)
3. Fournier-Viger, P., Gomariz, A., Campos, M., Thomas, R.: Fast vertical mining of sequential patterns using co-occurrence information. In: Tseng, V.S., Ho, T.B., Zhou, Z.-H., Chen, A.L.P., Kao, H.-Y. (eds.) PAKDD 2014. LNCS (LNAI), vol. 8443, pp. 40–52. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-06608-0\\_4](https://doi.org/10.1007/978-3-319-06608-0_4)
4. Ge, J., Xia, Y., Wang, J., Nadungodage, C.H., Prabhakar, S.: Sequential pattern mining in databases with temporal uncertainty. *Knowl. Inf. Syst.* **51**(3), 821–850 (2017). <https://doi.org/10.1007/s10115-016-0977-1>
5. Gunning, D.: Explainable artificial intelligence (xai). Defense Advanced Research Projects Agency (DARPA), nd Web (2017)
6. Holzinger, A.: Interactive machine learning for health informatics: when do we need the human-in-the-loop? *Brain Inform.* **3**(2), 119–131 (2016)
7. Huang, J., Huan, J., Tropsha, A., Dang, J., Zhang, H., Xiong, M.: Semantics-driven frequent data pattern mining on electronic health records for effective adverse drug event monitoring. In: 2013 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 608–611. IEEE (2013)

8. Jensen, K., et al.: Analysis of free text in electronic health records for identification of cancer patient trajectories. *Sci. Reports* **7**, 46226 (2017)
9. Jindal, R., Malhotra, R., Jain, A.: Techniques for text classification: literature review and current trends. *Webology* **12**(2), 1 (2015)
10. Krumholz, H.M.: Big data and new knowledge in medicine: the thinking, training, and tools needed for a learning health system. *Health Aff.* **33**(7), 1163–1170 (2014)
11. Névóel, A., Dalianis, H., Velupillai, S., Savova, G., Zweigenbaum, P.: Clinical natural language processing in languages other than english: opportunities and challenges. *J. Biomed. Semant.* **9**(1), 12 (2018)
12. Patnaik, D., Butler, P., Ramakrishnan, N., Parida, L., Keller, B.J., Hanauer, D.A.: Experiences with mining temporal event sequences from electronic medical records: initial successes and some challenges. In: *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 360–368. ACM (2011)
13. Plantevit, M., Charnois, T., Klema, J., Rigotti, C., Crémilleux, B.: Combining sequence and itemset mining to discover named entities in biomedical texts: a new type of pattern. *Int. J. Data Min. Model. Manag.* **1**(2), 119–148 (2009)
14. Rabatel, J., Bringay, S., Poncelet, P.: Mining sequential patterns: a context-aware approach. In: Guillet, F., Pinaud, B., Venturini, G., Zighed, D. (eds.) *Advances in Knowledge Discovery and Management. SCI*, vol. 471, pp. 23–41. Springer, Heidelberg (2013)
15. Wang, F., Lee, N., Hu, J., Sun, J., Ebadollahi, S.: Towards heterogeneous temporal clinical event pattern discovery: a convolutional approach. In: *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 453–461. ACM (2012)
16. Wright, A.P., Wright, A.T., McCoy, A.B., Sittig, D.F.: The use of sequential pattern mining to predict next prescribed medications. *J. Biomed. Inform.* **53**, 73–80 (2015)
17. Yadav, P., Steinbach, M., Kumar, V., Simon, G.: Mining electronic health records (ehrs): a survey. *ACM Comput. Surv. (CSUR)* **50**(6), 85 (2018)
18. Zaki, M.J., Gouda, K.: Fast vertical mining using diffsets. In: *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 326–335. ACM (2003)
19. Ziemiński, R.Z.: Accuracy of generalized context patterns in the context based sequential patterns mining. *Control Cybern.* **40**, 585–603 (2011)



# **MAKE-Smart Factory**



# A Multi-device Assistive System for Industrial Maintenance Operations

Mario Heinz<sup>1</sup>(✉), Hitesh Dhiman<sup>1</sup>, and Carsten Röcker<sup>1,2</sup>

<sup>1</sup> University of Applied Science Ostwestfalen-Lippe, 32657 Lemgo, Germany  
{mario.heinz,hitesh.dhiman,carsten.roecker}@hs-owl.de

<sup>2</sup> Fraunhofer IOSB-INA, 32657 Lemgo, Germany

**Abstract.** Recent advances in the field of industrial digitization and automation lead to an increasing need for assistance systems to support workers in various fields of activity, such as assembly, logistics and maintenance. Current assistance systems for the maintenance area are usually based on a single visualization technology. However, in our view, this is not practicable in terms of real activities, as these operations involve various subtasks for which different interaction concepts would be advantageous. Therefore, in this paper, we propose a concept for a multi-device assistive system, which combines multiple devices to provide workers with relevant information over different subtasks of a maintenance operation and present our first prototype for such a system.

**Keywords:** Human-machine interaction · Assistive systems · Augmented reality · Smart factory

## 1 Introduction

Nowadays, we experience a trend towards the development of digital assistive systems to support workers in industrial environments. This trend is caused by the ongoing digitization and automation and the growing complexity and heterogeneity of manufacturing processes and production plants [1]. In this context, digital assistance systems are intended to reduce the cognitive load of workers to make and keep complex manufacturing systems controllable. Current assistance systems for industrial applications thereby cover various fields of activity, such as assembly, maintenance, logistic and training [17]. However, the number of industrial assistance systems in the field of maintenance, logistics or training is comparatively lower compared to the number of systems in the field of assembly. In the last few years, scientific developments in this context are increasingly relying on augmented reality (AR) devices such as tablet-PCs, in-situ projections or head-mounted displays (HMDs), which are able to enrich the user's field of view with digital information and virtual objects [2, 3]. Additionally, recent advantages in the field of depth sensor technologies open up various possibilities of creating context-aware systems and interaction methods [4]. While stationary assistance systems for assembly activities are relatively widespread, the number of systems to support maintenance activities and repair tasks proves to be comparatively low. A possible reason could be the fact that maintenance and repair tasks require both a mobile approach and a specific adaptation to changing environments and machine

types, while assembly systems are usually limited to a specific workplace and thus to a fixed environment.

Therefore, in this paper, we will present a concept for a multi-device assistive system to support users on maintenance tasks in the industrial environment. In Sect. 2, we will take a look at current state-of-the-art assistive systems for maintenance applications in industrial environments. In Sect. 3, we will introduce our concept for a multi-device assistive system. In Sect. 4 we present the status of our current prototypical implementation of multi-device assistive system. In Sect. 5 we finally provide a conclusion and an outlook on future research activities in the context of this paper.

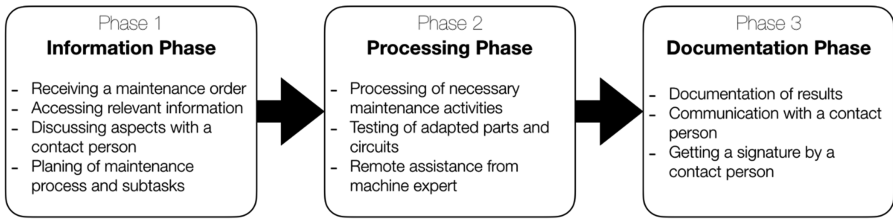
## 2 Related Work

In contrast to assistance systems for assembly tasks, the development of assistance systems for maintenance operations requires a considerably higher variability. This is justified by the fact, that maintenance tasks represent mobile activities which are carried out in different environments and at different production plants. As shown in Fig. 1, maintenance or repair processes can generally be divided into three sections: the information phase (Phase 1), the processing phase (Phase 2) and the documentation phase (Phase 3). While the information phase concerns the collection of information to a related task, the processing phase represents the actual execution of the maintenance activity or repair process at a production system. The documentation phase is finally used to document the results of a maintenance or repair process to finish the overall operation.

Visualization technologies used in current assistive systems, such as tablet PCs, in-situ projections and HMDs have different advantages and disadvantages regarding their application for activities and situations in industrial environments.

In this context, previous studies like presented by Funk et al. [5] and Büttner et al. [18] towards the evaluation of different types of devices for the implementation of assistance systems for assembly tasks revealed, that in-situ projections can offer a better support compared to HMDs and Tablet-PCs. However, since these systems usually follow a stationary design based on assembly tables with fixed dimensions, they are not really applicable for mobile scenarios such as maintenance operations which are very likely to be performed at different production systems in different locations. In addition, mobile systems based on in-situ projections currently exist only in the form of niche developments such as the projector helmet presented by Funk et al. [6], the *TeleAdvisor* introduced by Gurevich et al. [7] or the semi portable *MagicMirror* system introduced by Fiorentino et al. [8]. Therefore, mobile assistive systems have to be build up on one of the other visualization technologies such as HMDs and tablet PCs.

Zheng et al. [9] and Aromaa et al. [2] evaluated the efficiency of paper based instructions and different devices like tablet PCs and HMDs to assist workers during maintenance operations. The results of these studies show that HMDs do not have significant advantages over other devices, such as tablet PCs, in terms of the completion time and the number of errors in the processing of a maintenance task. But we have to point out, that these studies only focus on the performance of a maintenance task itself, and do not evaluate the gathering of information about the machine nor the



**Fig. 1.** Overview of the different phases of a maintenance or repair process and the related activities.

phase of documentation where the result of the operation is recorded. These phases, however, require a distinct kind of information presentation and interaction design to provide complex information. At the same time, the system must be able to provide a common viewing and processing of information by several persons in order to allow a coordination between the worker and a contact person.

In this sense, devices such as smartphones, tablet PCs, or laptops are very likely to provide an adequate way to view information about the details of a maintenance task and represent an ideal tool for documenting the results of such processes. But on the other hand, these systems do not allow to work hands-free without switching the attention between the display of the device and the location of interest. Compared with these systems, AR-based HMDs prove to be more advantageous for the processing phase because of their possibility to work free handed. But they are not efficient in viewing and processing complex data and do not allow to share information between multiple users.

In addition, the acceptance of users in relation to the technologies used is also to be observed, as this has a significant influence on the usability and user experience of the entire system. While touch-based systems are now widely used in everyday life as well as in the industrial environment, the proportion of HMDs used in the industry is still rather low to non-existent. But in the future, this circumstance might change since current developments in this area are subject to rapid progress.

### 3 A Concept for Multi-device Assistive Systems

For the implementation of an assistance system based on a combination of different devices and technologies, there are various requirements regarding the communication, the handling and processing of data streams, the handling of different devices and users as well as the synchronization between different devices. Figure 2 shows a general concept for the implementation of such a system including multiple devices, a local server, a logistic software system, a production plant and external sensor systems. In this concept, the server acts as a communicator between the digital infrastructure of a production facility and the different devices. It handles incoming and outgoing data streams, controls device communications, holds relevant information about different tasks and related media content as well as it allocates devices to specific users.

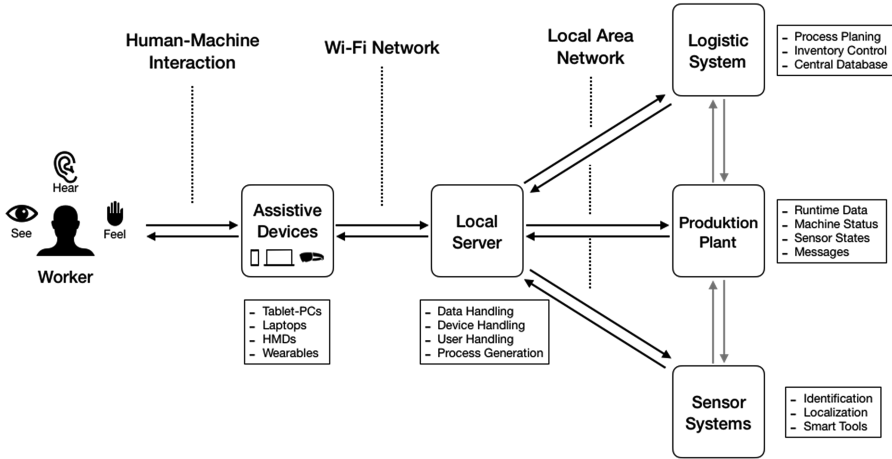


Fig. 2. Overview on the concept of a multi-device assistive system.

### 3.1 Assistive Devices

During the last years, a lot of devices based on numerous technologies have been developed and evaluated to provide users with information over different sensory channels. Today, mobile devices like smartphones, Tablet-PCs or Laptops are the most frequently used devices for mobile applications. But the latest developments in the area of AR-based HMDs show great potential for future applications in industrial environments. In contrast to previous assistance systems, which are usually limited to a single device type, our approach follows a combination of different systems to their respective advantages and disadvantages in terms of different situations and activities.

As stated in Sect. 2, there are currently no existing devices that can perform interactive in-situ projections in mobile applications. Therefore, the choice of applicable devices falls primarily on the use of Tablet PCs, laptops and HMDs supported by various wearables like smartwatches or other tactile wearable devices like work gloves [10, 11], bracelets [12, 13] or shoes [14, 15].

### 3.2 Device Communication

In order to integrate different types of devices into the overall system, a unified communication has to be implemented, which enables the development of programs for different operating systems and device types. The basis for this is a special data structure, which allows a transfer of the different content and media formats and on the other a communication protocol which of as many potentially usable devices is implemented. A protocol which is implemented in most devices with a wired or wireless network adapter is the Transmission Control Protocol/Internet Protocol (TCP/IP) which allows two devices to communicate via a serial connection over a specific port. The common object-based data structures for the transmission between the server and the devices are primarily the Extensible Markup Language (XML) format or JavaScript Object Notation

(JSON) format, as they are supported by various programming languages and systems [16]. Since the text-based instructions of current assistance systems are usually extended by different media formats, such as pictures or videos, it is also necessary to transfer these files from the server system to the different devices. A transmission of these files via a JSON or XML structure would be inefficient and time-consuming because each file would have to be transformed into textual information for the transfer. A potential solution for the deploying of various media files via a network connection could be the implement of a Representational State Transfer-API (REST-API). This software, implemented on the server side, allows to make files and further information accessible to other devices via a specific network address.

### 3.3 User Management

Since it is possible that several workers use the same equipment or several maintenance activities are carried out at the same time, the server system must be able to allocate devices and maintenance processes to a specific user. This is realized by an individual user ID which is transmitted during communication between the server and the individual devices in order to identify the current user.

### 3.4 Device Synchronisation

To allow users to switch between different devices during a maintenance process without performing further adjustments requires an efficient synchronization mechanism. The synchronization is performed by the server, which holds the information about which devices are used by a single user. At each interaction on one of the devices, a message is send to the server to ask for the data to be shown in a next or previous step. The server then sends this new data package to each device with the matching user ID.

## 4 Prototypical Implementation

Our prototypical implementation of a multi-device assistive system aims to support users on maintenance operations and repair tasks at a laundry folding machine. This machine is normally used in industrial laundries to fold large amounts of hotel linen and towels. The assistive system thereby provides workers with step-by-step instructions and further information about the machine and the environment. The system is located in the SmartFactoryOWL, a demonstrator facility for industrial research projects in the scope of digitization and automation in Lemgo, Germany [19].

The final assistive system will consist of an ordinary mini-PC, acting as a local server-system, an AR-based HMD (Microsoft HoloLens<sup>1</sup>), a tablet-PC, a smartphone and custom wearable devices which are used to support users during different phases of a maintenance operation. The server provides a REST API for reading media content,

---

<sup>1</sup> <https://www.microsoft.com/en-gb/hololens>.

as well as a TCP/IP socket connection for transmitting control information and text-based instructions, as well as the links for the associated media contents. By transmitting just the links for media files on the REST server, the usage of this content lies in the hand of the visualization software of the different devices. For the transmission of control information and text-based data, we used a specific JSON structure to provide step-by-step instructions as well as further information about the design and position of virtual objects, relevant machine data or the position of the user (see Fig. 3).

```

Step {
  „TutorialID“ : „“,          # ID of the operation
  „StepNumber“ : „“,         # index of step in tutorial
  „TotalSteps“ : „“,        # number of steps in tutorial
  „Type“ : „“,              # type of the step ()
  „Title“ : „“,             # title of the step
  „Instruction_1“ : „“,     # main instruction (text)
  „Instruction_2“ : „“,     # additional instruction (text)
  „Media_1“ : „“,          # link to image or video
  „Media_2“ : „“,          # link to image or video
  „VirtualObjects“ : [],    # list of virtual objects and their locations for this step
  „MachineStatus“ : „“,    # status information of the related plant
  „Location“ : []          # three-dimensional position of the user
}

```

**Fig. 3.** Example of the data structure for a step as part of a step-by-step instruction for augmented reality devices.

The server has access to the internal database of the folding machine and can thus collect detailed information about the machine status and messages such as warnings or errors. In addition, the server is also able to read the states of the light sensors inside the machine in order to check the proper operation of a folding process. These two datasets are then used to choose the related set of instructions to perform a maintenance or repair process.

The software for the tablet-PC was developed as a 2D application for Windows-based operating systems. In this way, it works on Windows-based tablet-PCs as well as laptops or stationary PCs. It allows to visualize step-by-step instructions and different media files and is also able to display runtime data and sensor information from the folding machine.

The software for the AR-based HMD was developed via the Unity Game Development Platform<sup>2</sup> and presents step-by-step instructions for maintenance activities such as the cleaning of light sensors at the folding machine (Fig. 4). The system initially scans an QR-Code to set the global origin for its coordinate system. In this way, the application can be used at any machine of the same type as long as the QR-Code is placed at the same spot. Afterwards, the user can choose an instruction from a menu. The step-by-step process of the application consists of a main window, which is placed on the center above the machine. This window shows textual information about the

<sup>2</sup> <https://unity3d.com>.

current step of a tutorial extended by an image of the related part on the machine and allows the user to switch between the different steps. In order to guide the user to the right place, virtual objects such as animated arrows or highlighted planes are placed on relevant parts of the machine. When the user walks around the machine, the main windows will further automatically adjust its orientation to face the position of the user.



**Fig. 4.** Example views from the AR-based HMD software.

## 5 Conclusion and Future Work

In this paper, we presented a general concept for an assistive system to support users on various phases of maintenance operations at industrial production systems. Contrary to previous publications, our Concept thereby follows a multi-device approach to take advantage of different device technologies in different situations. Our prototypical implementation of such a system is still under development, but it already shows potential to be a good solution to be also applied to multiple machines.

We consider our concept and the prototypical implementation as a first step in order to provide a basis for a framework to integrate mobile assistance systems in the industrial environment. In the future, the existing system will be continuously improved and extended by various systems and information sources in order to provide additional information. Furthermore, we will evaluate different combinations of devices and technologies as well as different visualization methods with regard to their usability, user experience and productivity by performing extensive user studies and questionnaires.

**Acknowledgements.** This work is funded by the German Federal Ministry of Education and Research (BMBF) for project ADIMA under grant number 13FH019PX5.

## References

1. Radziwon, A., Bilberg, A., Bogers, M., Madsen, E.S.: The smart factory: exploring adaptive and flexible manufacturing solutions. *Procedia Eng.* **69**, 1184–1190 (2014)
2. Aromaa, S., Aaltonen, I., Kaasinen, E., Elo, J., Parkkinen, I.: Use of wearable and augmented reality technologies in industrial maintenance work. In: *Proceedings of the 20th International Academic Mindtrek Conference*, pp. 235–242. ACM (2016)



3. Fite-Georgel, P.: Is there a reality in industrial augmented reality? In: 2011 10th IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. 201–210. IEEE (2011)
4. Izadi, S., et al.: KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera. In: Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, pp. 559–568. ACM (2011)
5. Funk, M., Kosch, T., Schmidt, A.: Interactive worker assistance: comparing the effects of in-situ projection, head-mounted displays, tablet, and paper instructions. In: Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 934–939. ACM (2016)
6. Funk, M., Mayer, S., Nistor, M., Schmidt, A.: Mobile in-situ pick-by-vision: order picking support using a projector helmet. In: Proceedings of the 9th ACM International Conference on Pervasive Technologies Related to Assistive Environments, p. 45. ACM (2016)
7. Gurevich, P., Lanir, J., Cohen, B., Stone, R.: TeleAdvisor: a versatile augmented reality tool for remote assistance. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 619–622. ACM (2012)
8. Fiorentino, M., Radkowski, R., Boccaccio, A., Uva, A.E.: Magic mirror interface for augmented reality maintenance: an automotive case study. In: Proceedings of the International Working Conference on Advanced Visual Interface, pp. 160–167. ACM (2016)
9. Zheng, X.S., Foucault, C., Matos da Silva, P., Dasari, S., Yang, T., Goose, S.: Eye-wearable technology for machine maintenance: effects of display position and hands-free operation. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, pp. 2125–2134. ACM (2015)
10. Hsieh, Y.-T., Jylhä, A., Jacucci, G.: Pointing and selecting with tactile glove in 3D environment. In: Jacucci, G., Gamberini, L., Freeman, J., Spagnolli, A. (eds.) *Symbiotic 2014*. LNCS, vol. 8820, pp. 133–137. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-13500-7\\_12](https://doi.org/10.1007/978-3-319-13500-7_12)
11. Moy, G., Wagner, C., Fearing, R.S.: A compliant tactile display for teletaction. In: Proceedings 2000, IEEE International Conference on Robotics and Automation, ICRA 2000, vol. 4, pp. 3409–3415. IEEE (2000)
12. Matscheko, M., Ferscha, A., Riener, A., Lehner, M.: Tactor placement in wrist worn wearables. In: 2010 International Symposium on Wearable Computers (ISWC), pp. 1–8. IEEE (2010)
13. Brock, A., Kammoun, S., Macé, M., Jouffrais, C.: Using wrist vibrations to guide hand movement and whole body navigation. *i-com* **13**(3), 19–28 (2014)
14. Fu, X., Li, D.: Haptic shoes: representing information by vibration. In: Proceedings of the 2005 Asia-Pacific Symposium on Information Visualisation, vol. 45, pp. 47–50. Australian Computer Society, Inc (2005)
15. Xu, Q., Gan, T., Chia, S.C., Li, L., Lim, J.H., Kyaw, P.K.: Design and evaluation of vibrating footwear for navigation assistance to visually impaired people. In: 2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp. 305–310. IEEE (2016)
16. Nurseitov, N., Paulson, M., Reynolds, R., Izurieta, C.: Comparison of JSON and XML data interchange formats: a case study. *Caine* **9**, 157–162 (2009)
17. Büttner, S., et al.: The design space of augmented and virtual reality applications for assistive environments in manufacturing: a visual approach. In: Proceedings of the 10th International Conference on Pervasive Technologies Related to Assistive Environments, pp. 433–440. ACM (2017)

18. Büttner, S., Funk, M., Sand, O., Röcker, C.: Using head-mounted displays and in-situ projection for assistive systems: a comparison. In: Proceedings of the 9th ACM International Conference on Pervasive Technologies Related to Assistive Environments, p. 44. ACM (2016)
19. Büttner, S., Mucha, H., Robert, S., Hellweg, F., Röcker, C.: HCI in der SmartFactoryOWL– Angewandte Forschung & Entwicklung. Mensch und Computer 2017-Workshopband (2017)



# Feedback Presentation for Workers in Industrial Environments – Challenges and Opportunities

Mario Heinz<sup>1(✉)</sup> and Carsten Röcker<sup>1,2</sup>

<sup>1</sup> University of Applied Science Ostwestfalen-Lippe, 32657 Lemgo, Germany  
{mario.heinz, carsten.roecker}@hs-owl.de

<sup>2</sup> Fraunhofer IOSB-INA, 32657 Lemgo, Germany

**Abstract.** On the long term, the current wave of digitization and automation in the industrial environment will result in a progressively higher complexity and heterogeneity in the industrial environment. In this context, a growing need arises for the development of digital assistance systems to support workers in various fields of activities. Current systems are generally limited to visualizations and visual feedback. Therefore, in the scope of this paper, we take a look at the major challenges and opportunities for the integration of multimodal feedback systems in today's and future industrial environments. It shows that the integration of multimodal feedback is subject to a complex combination of technical, user-centric and legal aspects.

**Keywords:** Human-machine-interaction · Multimodal feedback  
Assistive systems · Augmented-reality · Smart factory

## 1 Introduction

Today's industrial landscape is characterized by a mixture of analogue and digital production facilities. As a result of the advancing digitization and automation in the industrial sector, this situation will change significantly in the upcoming years and the number of intelligent production facilities, so-called smart factories, will steadily increase. These smart environments will be characterized by complex digital and automated production systems, robots, autonomous transport vehicles, sensor systems and a high number of other digital devices [1, 2]. At the same time, these changes will also shift the role of the worker in the industrial environment and most of the workers will primarily be employed in the field of monitoring, maintenance and logistics rather than in the area of assembly [8, 25]. The activities in the field of assembly, on the other hand, will in future be limited to specific tasks that cannot be automated due to the emerging high product diversity and short production cycles. In order to facilitate the completion of these activities and to ensure the safety of people in these dynamic, highly automated areas, extensive and individually customizable assistance systems will be required. These systems in form of digital equipped workplaces or various mobile devices will have to provide workers with relevant information about their surroundings or ongoing and upcoming tasks. At the same time, these systems must

also inform their users about relevant situations within their environment as well as possible dangers such as autonomous vehicles or robots around their workplaces. For this reason, an integration of adequate feedback methods will be necessary for the implementation of future assistance systems which aim to extend and enrich the interaction between a user and a digital system by providing information over different sensory channels [5].

But, despite the opportunities to improve the support for workers, an extensive integration of feedback systems also raises numerous challenges regarding technical, user-related and legal aspects. Thus, adequate feedback presentations require a deep integration of the related systems into the infrastructure of a production facility to get access to relevant information about ongoing processes and production systems. In addition, these systems will have to meet certain requirements regarding their usability and user experience, as well as data security and privacy regulations in the workplace.

Therefore, as part of this paper, we want to introduce and discuss key challenges and opportunities for the integration of multimodal feedback systems for industrial applications. In the second part we will first look at the current state of research in the field of feedback technologies and adaptive assistance systems. Then, in part three and four, we will discuss the major challenges and opportunities for the application of multimodal feedback systems in the industrial environment. Finally, in part five and six, we will discuss our findings and provide an outlook towards future research activities in this context.

## 2 Related Work

### 2.1 Feedback Modalities and Feedback Devices

Feedback represents an essential component of human learning and behavior. Generally, according to the *principle of actio et reactio*, it can be seen as the reaction of an environment to an action performed by an individual within it [6]. The positive or negative evaluation of this reaction has an impact on subsequent actions and can influence future behavior in similar situations. Feedback is perceived through a variety of different sensations via the various sensory channels of the human body such as seeing, hearing, feeling, smelling, tasting as well as kinesthetics [5]. These modalities are used to create an internal representation of the environment and to build or expand knowledge about the interaction between different entities and actions. Feedback is thereby provided via a single sensory channel or through a combination of different channels, often referred to as multimodal feedback [7]. Multimodal actuations are generally offering a more natural and trusted perception as long as the provided sensations are corresponding to plausible procedures [8, 9]. But various studies also revealed that a combination of a primary and a supporting feedback dimension is often more effective than a combination of three or more feedback dimensions [8–12].

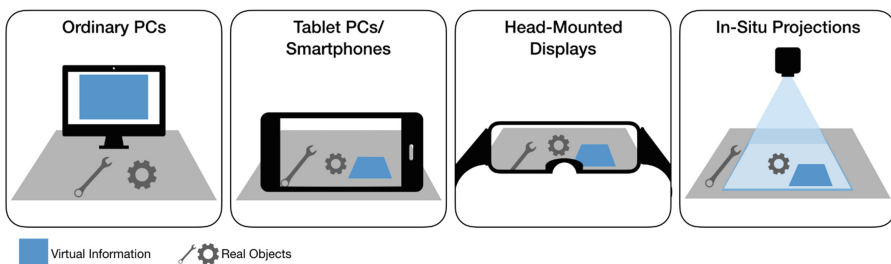
Regarding digital devices, visual feedback is carried out via different forms of light sources, stationary and mobile displays, digital projections and head-mounted displays (HMDs). The provided information ranges from simple status lights or color changes over symbols to images, text and animations. Auditory feedback, on the other hand, is

presented over speakers or headphones and ranges from simple acoustic signals over signal patterns to spoken language. Haptic feedback, in this sense, is divided into tactile feedback and kinesthetic feedback. Tactile feedback addresses the human sense of touch and is usually presented through vibrations, while kinesthetic perception refers to the posture and movement of the joints as well as the perception of external forces that are performed against the body [7]. Especially the development of haptic and tactile systems has increased considerably in recent years [13, 14]. Currently, an important aspect concerns the integration of tactile actuators into various garments, such as work gloves [15, 16], bracelets [17, 18] or shoes [19, 20] to provide workers with additional tactile information.

## 2.2 Assistive Systems for Industrial Applications

In recent years, the increasing digitization and automation has triggered a trend towards the development of assistive systems to support workers on various activities in the industrial environment. These assistance systems provide their users with step-by-step instructions for daily tasks but may also display further information such as machine-related or process-related data as well as warnings about faulty actions or potential dangers in the environment. The application of these systems ranges from assembly tasks over maintenance operations to activities in the field of logistics. The devices used for this purpose include normal PCs or mobile devices such as smartphones and tablet PCs.

But, due to the ongoing developments in the field of augmented reality (AR) as well as in the field of mobile devices and wearables, current assistance systems progressively focus on the implementation of augmented reality scenarios. These systems use in-situ projections, special AR tablet PCs or AR HMDs to project digital information directly into the field of view of a user. Figure 1 shows an overview of different visualization technologies for today's and future assistive systems.



**Fig. 1.** Overview of visualization technologies for today's and future assistive systems.

Stationary assistive systems for assembly tasks like the assembly tables presented by Funk et al. [21] and Büttner et al. [22] use a combination of in-situ projections and a recognition of hand movements via deep cameras. Thus, the system allows to project information directly into the working area. The interaction with the system is implemented via hand-tracking based on the integrated depth camera, e.g. via virtual buttons.

Assistive systems for maintenance tasks such as the systems presented by Zheng et al. [23] and Aromaa et al. [24], on the other hand, are usually based on AR tablet-PCs, AR HMDs and wearables in order to allow the implementation of mobile maintenance scenarios.

However, regarding the provision of feedback, stationary and mobile assistive systems for industrial applications are generally restricted to a presentation of information via the visual channel. The extension of these systems by additional feedback modalities is still ongoing research: Funk et al. [8] and Kosch et al. [12], for example, prototypically extended an assembly workplace with devices for the presentation of auditory and tactile error feedback in order to evaluate the effectiveness and user experience of the different modalities.

### **3 Challenges for Feedback Presentation in Industrial Environments**

Various interdisciplinary challenges arise for the integration and application of multimodal feedback systems in the industrial environment. This includes both technical and user related as well as legal aspects and addresses different research fields such as human-machine interaction, industrial communication, machine learning, artificial intelligence, sensor technologies, workplace privacy and data security.

#### **3.1 Integration of Feedback Systems in the Industrial Infrastructure**

From a technological point of view, a major challenge concerns the general integration of feedback systems in industrial environments. To provide workers with adequate feedback, these systems must be able to collect and process information about the environment. In this context, today's industrial facilities are already equipped with certain kinds of sensory systems to collect information about air pressure, power consumption, the localization of materials, vehicles and employees or other parameters. In future industrial facilities, however, the number of sensory systems needs to increase significantly in order to create decent virtual representations of the environment including ongoing processes, power management, material flows, errors, and other relevant information. Due to the high dynamics and complexity of future production plants, various sensors as well as powerful algorithms for scene analysis and processing are required in order to develop context-aware processes and workflows [25]. Figure 2 shows an abstract overview for the integration of feedback systems in industrial environments. Feedback devices as part of assistance systems for various activities are both connected via the digital system of the respective workplace as well as via the central server system. This enables feedback to be applied in relation to local and production-wide processes.

Developments in the field of industrial communication technology are currently splitting into various research tracks which follow the implementation of industrial network infrastructures using different technologies. The solutions range from local server systems to cloud-based systems and various intermediate solutions which hold relevant data locally and, if necessary, retrieve additional data from a cloud service [26, 27].

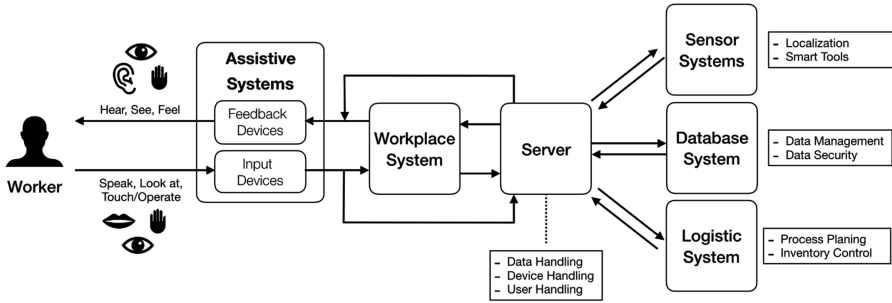


Fig. 2. Overview of the general integration for feedback devices in an industrial environment

The implementation of the communication channels, however, is currently frequently carried out via a combination of wired and radio-based networks to connect the growing number of stationary and mobile digital systems in today’s industrial environments. But due to the progressive development of mobile systems in the research area of the Internet of Things, which aims to digitalize and interconnect various systems in order to generate a virtual representation of ongoing processes and connections, there is a growing need for a wireless network solution for industrial environments that provides the required high bandwidth and low latency [28–30]. In this context, the 5g technology, also known as the Tactile Internet, which is based on mobile communication technologies, aims to integrate industrial plants and other digital devices in industrial environments into a nationwide internet ready network [26, 31].

### 3.2 Data Processing

Because of the complexity of industrial networks, the collection, processing and analysis of the high number of heterogeneous data streams, generated by numerous sensors, production systems and other entities requires efficient algorithms and concepts [32]. In this field, scientists are increasingly relying on the use of developments in the field of machine learning and artificial intelligence in order to be able to identify different relationships and situations [33, 34]. This contextual sensitivity, in turn, makes it possible to draw conclusions about the correct or incorrect execution of actions by individual workers to initiate an adequate response.

Regarding the presentation of multimodal feedback, humans are sensory impressions are subject to certain temporal limits during which time a reaction is perceived as natural. This also represents a strong temporal limitation for the processing and presentation of relevant feedback information. For the auditory system, this limit is about 100 ms, while the visual system is limited to 10 ms and the haptic system to 1 ms [35, 36]. While the performance of auditory and visual feedback information is already achieved via today’s network structures, the realization of the transmission of haptic information is still a current research subject, which is currently challenged by researchers in the scope of the development of 5G technologies, also referred to as Tactile or Haptic Internet [31, 36]. In general, it must be ensured that the processing and transmission of feedback information corresponds to the temporal limits of human information processing.

### 3.3 External Influences

Another key challenge for the integration of feedback systems in the industrial environment concerns external influences that could affect both workers and sensory systems [37, 38]. This primarily includes changing volume and light conditions as well as various forms of vibrations caused by machines or tools. These influences are very likely to have a negative impact on the performance of feedback. For example, a purely visual feedback could be disturbed by light influences such as apertures or particularly bright ambient lights. The performance of auditory feedback such as alert tones or spoken text could also be disturbed by a high ambient volume. Furthermore, tactile feedback could be overlaid by vibrations generated by production systems or work tools. External influences, however, may not only affect the provision of feedback directly, but could also influence sensory systems in the surrounding which in turn are highly relevant for the presentation of feedback. The application of feedback systems therefore requires a detailed analysis of the working environment and the tools needed to carry out the respective activity.

### 3.4 User Acceptance

A further user-related challenge concerns the acceptance of workers towards modern technologies, especially with regard to interaction and feedback devices. While younger generations are generally more familiar with the functionality and application of modern interaction and feedback systems, there are some high dislikes on the part of older generations regarding these kind of systems [39, 40]. Röcker [41] identified different societal and technological as well as privacy-related concerns towards the usage of new technologies in future work environments. Furthermore, Holzinger et al. [42] introduced a “previous exposure to technology” factor which has general influence on the acceptance of software applications. Another influence on the acceptance of digital systems stems from the increasing use of methods from machine learning and artificial intelligence (AI). Since these methods have to be regarded as black boxes, it is nearly impossible to understand their internal behavior. This raises the general question if we can trust results from machine learning [43] and how we can build explainable AI systems [44].

### 3.5 Data Security and Workplace Privacy

Another challenge directly related with user acceptance concerns the implementation of directives on data protection and workplace privacy in industrial environments. This especially includes personal data from employees as well as data collected by assistive systems, interaction and feedback devices. These systems are able to capture complex information about the performance and location of workers and are therefore often viewed as a potential way to monitor employees which would negatively affect their workplace privacy [45–47]. The same also applies to the previously described sensory systems for the localization of different entities in an industrial environment [48, 49]. However, personal data is required in this context in order to ensure the adaptiveness of the assistance systems and thus also the individual adaptation of the presentation of



feedback information. According to Sack and Röcker [50] knowledge about technical processes is influenced by age and technology experience while the knowledge over technical processes is not related with attitudes like security or privacy. Thus, the confidence in technology and the reduction of privacy concerns has to be build up by the designers and developers of future assistive systems and feedback devices [51].

### 3.6 Selection of Feedback Devices and Feedback Presentations

A further challenge which is highly related to user acceptance and workplace privacy arises through the selection of suitable feedback systems for the application in industrial environments. These systems can generally be categorized in portable and stationary devices. While portable systems are able to provide a location-independent presentation of feedback modalities, they open up a greater potential for long-term monitoring in terms of acceptance and workplace privacy (Sect. 3.4). In contrast, stationary feedback systems integrated in the working environment are limited to certain areas and may therefore provide a lower sense of permanent monitoring. But, compared to mobile systems, stationary systems are not able to provide feedback to users outside their workspace. Furthermore, a common use of stationary feedback systems by several people also has to be considered as critical, because provided information is likely to be misinterpreted by another user.

Another aspect concerns the potential overlay or attenuation of sensory sensations caused by feedback systems. For example, data gloves can provide a much more detailed feedback in the execution of manual activities. But overlaying the skin with one or more layers of fabric may lead to a lowering of the sensation of the haptic receptors. Moreover, the construction of some portable systems can lead to a restriction of freedom of movement.

In addition to the selection of applicable feedback systems, the choice of an adequate presentation of feedback information is also of high importance. In this context, Funk et al. [52] evaluated different visualization techniques to support an assembly task for impaired workers. In general, due to the growing number of devices in the area of interaction technologies, it will be necessary to develop new technology-specific concepts for the presentation of feedback over visual, acoustic and tactile channels. Furthermore, the acceptance towards a system is also dependent on its proper functionality. With regard to feedback systems, the focus here is primarily on fulfilling the temporal limitation of the human information processing of various information channels described in Sect. 3.2.

### 3.7 Cognitive Workload

Another user-related challenge concerns the increasing cognitive load generated by a huge amount of digital information passed to workers in modern industrial environments [53, 54]. To reduce the cognitive load of workers, intelligent filter routines are necessary, which analyze the existing data streams based on different parameters such as the experience level, the current activity, the position and the surroundings of a worker in order to select individual relevant information. In this context, a broad examination towards the individual perception of cognitive stress will be necessary.

Thereby, particularly HMDs are known to cause headaches and dizziness during prolonged use [55, 56]. Potential reasons for this are likely to be found in the limited ergonomics of these systems and the extensive presentation of additional visual information which leads to a constant change of the visual focus between digital information and the real world. Furthermore, it is important to evaluate how a long-term presentation of multimodal feedback sensations through digital Feedback devices affects the cognitive workload.

## **4 Opportunities for Feedback Presentation in Industrial Environments**

Despite the challenges and problems presented in Sect. 3, the integration of multimodal feedback technologies opens up numerous opportunities to assist workers by carrying out their activities and to improve the overall productivity and security. Furthermore, these systems can also be used to extend and enrich interaction concepts of augmented-reality technologies or to support even workers with certain cognitive or motoric disabilities, blindness or deafness on daily tasks.

### **4.1 Assistive Systems**

In the first place, the use of effective multimodal feedback methods in future industrial environments offers the possibility to extend existing assistance systems, which aim to adequately support workers by providing step-by-step instructions and further information about upcoming tasks and activities as well as warnings about errors or critical situations. In addition, the comprehensive integration of these systems into the digital infrastructure of industrial production environments can provide a more comprehensive feedback that exceeds the limits of the immediate environment of the workplace.

As described in Sect. 2, current assistance systems are usually limited to visual feedback presentations to assist workers. This in turn may contribute to a reduced usability and user experience. The presentation of feedback over multiple channels could thereby create a more natural and trusted loop of interaction between the system and the worker. In this context, the evaluations of various feedback modalities for the support of an assembly operation presented by Funk et al [8] and Kosch et al [12] are just to be seen as the beginning of an extensive evaluation process to identify adequate feedback devices and presentations for different activities in the industrial environment.

During the last years, developments in the field of assistive systems are increasingly relying on AR technologies. But recently developed devices such as AR tablet-PCs, AR HMDs or in-situ projectors still suffer from limited multimodal feedback implementations. Thereby, a presentation of multimodal feedback information in an AR scenario could create a much more natural and trustful relation between virtual and real objects [9]. In this way, for example, gesture-based interaction could be enhanced with tactile or auditory impressions. However, the use of such systems, in particular in the form of HMDs, will also require an extensive evaluation of the respective activity, the environment as well as the behavior and the cognitive burden of the user in order to create an adequate feedback presentation.

## 4.2 Inclusion of People with Disabilities

In addition to increasing the productivity and safety of healthy workers in different fields of activity, the use of multimodal feedback systems as part of assistive systems also offers the opportunity to especially support people with certain disabilities on different activities. During the years, several assistive technologies have been developed and evaluated to assist people with motoric or cognitive disabilities as well as blindness or deafness in private life as well as at the workplace [57]. Regarding the industrial environment, developments of assistive technologies for people with disabilities mainly focus on assembly operations supported via in-situ projections and motion tracking [58]. In this context, Korn [59] evaluated the application of gamification elements during an assembly task to support cognitively impaired people on an assembly task. Furthermore, Kosch et al. [12] further compared visual, auditory and tactile feedback methods to support impaired workers at an assembly task and Funk et al. [52] also evaluated different visualization techniques to support an assembly task for impaired workers. The results of these studies show that people with different disabilities can benefit from the provision of additional feedback information to perform operations that are normally too complex with respect to their performance index. In general, assistive systems can represent a long-term opportunity for a greater autonomy and an increased self-esteem for people with disabilities, and also a possibility for full-fledged occupational participation in the first labour market [57].

But especially the development of systems for people with disabilities requires a comprehensive evaluation of technological, social and legal aspects. Therefore, regarding impaired workers, feedback systems should fulfill specific guidelines and regulations like the German Federal Ordinance on Barrier-Free Information Technology [60] which holds detailed information about how to implement visualizations and other feedback modalities in a barrier-free way.

## 5 Conclusion

In this paper, we examined major challenges and opportunities for the presentation of multimodal feedback in today's and future industrial environments. We have shown that the emerging challenges are of a highly interdisciplinary nature, addressing fields like human-machine interaction, industrial communication, machine learning, artificial intelligence, sensor technologies, workplace privacy, data security and occupational science (Sect. 3). But the key challenges such as the choice of feedback devices and feedback presentations, the technology acceptance of workers or the privacy at the workplace can primarily be seen as highly user-related.

As mentioned in Sect. 1, in the upcoming years the role of workers in the industrial environment will shift from assembly activities to controlling and logistical tasks as well as maintenance operations. Since these tasks mainly represent mobile activities, mobile assistance systems and feedback systems will be increasingly needed to support workers in changing environments and at different production facilities. This, in turn, requires a distinct way to capture changing working environments in mobile scenarios.

However, prior to the development of these assistive systems, it is essential to carry out a detailed analysis of the environment and the respective activity as well as of the devices involved in order to be able to perform a selection of the appropriate feedback modalities and devices. As discussed in Sect. 4.2, an additional aspect also emerges through the development of systems for people with disabilities. In this context, additional analyses for the determination of individual needs will be required to provide adequate support for workers with different disabilities.

Technical aspects, on the other hand, are focused on the integration of feedback technologies into the digital infrastructure of today's and future industrial environments. Current research thereby offers several solutions for an extensive integration of feedback systems (Sect. 3.1). While the 5G technology offers an interconnection between a high number of digital devices in a nationwide network, also a basic combination of wired and wireless networks with the required low latency and high bandwidth would be possible.

In addition to the various challenges, the possibilities for the integration of feedback systems presented in Sect. 4 prove to be highly relevant. The development and extension of assistance systems through various feedback modalities can contribute to higher productivity and higher safety of workers in future industrial environments. Especially, in the context of augmented reality devices as well as for the support of workers with certain disabilities, multimodal feedback could be highly beneficial.

Taking into account the potentially prevailing environmental influences in industrial environments such as changing lights and noise as well as vibrations generated by machines or work tools, effective combinations of different feedback modalities for certain working environments and fields of activity are required (Sect. 3.3). But, since the user-centered provision of auditory feedback in certain areas would only be feasible with headphones or target-oriented loudspeakers, which could potentially cause excessive attenuation of ambient noise or a limitation of privacy [8], developers may prefer combinations of visual and tactile feedback systems. In some cases, however, it may be necessary to further supplement the used portable feedback systems by using stationary interface-specific feedback systems in order to improve the overall usability and user experience.

In general, especially due to the rapid development of new interaction systems, we see high needs for research activities regarding the selection of appropriate feedback systems and the associated presentation methods to extend assistive systems for industrial environments.

## 6 Outlook

Regarding the integration of multimodal feedback systems, our future research will primarily be focused on questions concerning the selection of suitable devices and presentation techniques for the support of stationary and mobile activities in different occupational fields in industrial environments. This also includes the evaluation of various commercial and in-house development systems within the framework of comprehensive user studies and surveys on the effectiveness, acceptance and usability of various device combinations. Due to the changing role of workers in the industrial

environment, our research will be oriented towards the development of systems for mobile scenarios based on augmented reality technologies. In close cooperation with research colleagues from the fields of industrial communication technology, machine learning and artificial intelligence as well as the field of occupational sciences, we want to discuss necessary aspects of the requirements regarding the network infrastructure, the data processing and the usability and user experience for integrating feedback systems into industrial environments in order to examine potential solutions.

**Acknowledgments.** This work is funded by the German Federal Ministry of Education and Research (BMBF) for project ADIMA under grant number 13FH019PX5.

## References

1. Fellmann, M., Robert, S., Büttner, S., Mucha, H., Röcker, C.: Towards a framework for assistance systems to support work processes in smart factories. In: Holzinger, A., Kieseberg, P., Tjoa, A.M., Weippl, E. (eds.) CD-MAKE 2017. LNCS, vol. 10410, pp. 59–68. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-66808-6\\_5](https://doi.org/10.1007/978-3-319-66808-6_5)
2. Radziwon, A., Bilberg, A., Bogers, M., Madsen, E.S.: The smart factory: exploring adaptive and flexible manufacturing solutions. *Procedia Eng.* **69**, 1184–1190 (2014)
3. Brettel, M., Friederichsen, N., Keller, M., Rosenberg, M.: How virtualization, decentralization and network building change the manufacturing landscape: an Industry 4.0 perspective. *Int. J. Mech. Ind. Sci. Eng.* **8**(1), 37–44 (2014)
4. Gorecky, D., Schmitt, M., Loskyll, M., Zühlke, D.: Human-machine-interaction in the Industry 4.0 era. In: 2014 12th IEEE International Conference on Industrial Informatics (INDIN), pp. 289–294. IEEE (2014)
5. Salvendy, G.: *Handbook of Human Factors and Ergonomics*. Wiley, Hoboken (2012)
6. Aström, K.J., Murray, R.M.: *Feedback Systems: An Introduction for Scientists And Engineers*. Princeton University Press, Princeton (2010)
7. Rodrigues, J., Cardoso, P., Monteiro, J., Figueiredo, M. (eds.): *Handbook of Research on Human-computer Interfaces, Developments, and Applications*. IGI Global, Hershey (2016)
8. Funk, M., Heusler, J., Akcay, E., Weiland, K., Schmidt, A.: Haptic, auditory, or visual?: towards optimal error feedback at manual assembly workplaces. In: *Proceedings of the 9th ACM International Conference on Pervasive Technologies Related to Assistive Environments*, p. 43. ACM (2016)
9. Sigrist, R., Rauter, G., Riener, R., Wolf, P.: Augmented visual, auditory, haptic, and multimodal feedback in motor learning: a review. *Psychon. Bull. Rev.* **20**(1), 21–53 (2013)
10. Vitense, H.S., Jacko, J.A., Emery, V.K.: Multimodal feedback: an assessment of performance and mental workload. *Ergonomics* **46**(1–3), 68–87 (2003)
11. Hecht, D., Reiner, M.: Sensory dominance in combinations of audio, visual and haptic stimuli. *Exp. Brain Res.* **193**(2), 307–314 (2009)
12. Kosch, T., Kettner, R., Funk, M., Schmidt, A.: Comparing tactile, auditory, and visual assembly error-feedback for workers with cognitive impairments. In: *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 53–60. ACM (2016)
13. Benali-Khoudja, M., Hafez, M., Alexandre, J.M., Kheddar, A.: Tactile interfaces: a state-of-the-art survey. In: *International Symposium on Robotics*, vol. 31, pp. 23–26 (2004)

14. Stone, R.J.: Haptic feedback: a brief history from telepresence to virtual reality. In: Brewster, S., Murray-Smith, R. (eds.) *Haptic HCI 2000*. LNCS, vol. 2058, pp. 1–16. Springer, Heidelberg (2001). [https://doi.org/10.1007/3-540-44589-7\\_1](https://doi.org/10.1007/3-540-44589-7_1)
15. Hsieh, Y.-T., Jylhä, A., Jacucci, G.: Pointing and selecting with tactile glove in 3D environment. In: Jacucci, G., Gamberini, L., Freeman, J., Spagnolli, A. (eds.) *Symbiotic 2014*. LNCS, vol. 8820, pp. 133–137. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-13500-7\\_12](https://doi.org/10.1007/978-3-319-13500-7_12)
16. Moy, G., Wagner, C., Fearing, R.S.: A compliant tactile display for teletaction. In: *Proceedings of the 2000 IEEE International Conference on Robotics and Automation, ICRA 2000*, vol. 4, pp. 3409–3415. IEEE (2000)
17. Matscheko, M., Ferscha, A., Riener, A., Lehner, M.: Tactor placement in wrist worn wearables. In: *2010 International Symposium on Wearable Computers (ISWC)*, pp. 1–8. IEEE (2010)
18. Brock, A., Kammoun, S., Macé, M., Jouffrais, C.: Using wrist vibrations to guide hand movement and whole body navigation. *i-com* **13**(3), 19–28 (2014)
19. Fu, X., Li, D.: Haptic shoes: representing information by vibration. In: *Proceedings of the 2005 Asia-Pacific Symposium on Information Visualisation*, vol. 45, pp. 47–50. Australian Computer Society, Inc (2005)
20. Xu, Q., Gan, T., Chia, S.C., Li, L., Lim, J.H., Kyaw, P.K.: Design and evaluation of vibrating footwear for navigation assistance to visually impaired people. In: *2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pp. 305–310. IEEE (2016)
21. Funk, M., Korn, O., Schmidt, A.: Assistive augmentation at the manual assembly workplace using in-situ projection. In: *Proceeding of the CHI Workshop on Assistive Augmentation*, p. 11 (2014)
22. Büttner, S., Sand, O., Röcker, C.: Extending the design space in industrial manufacturing through mobile projection. In: *Proceedings of the 17th International Conference on Human-computer Interaction with Mobile Devices and Services Adjunct*, pp. 1130–1133. ACM (2015)
23. Zheng, X.S., Foucault, C., Matos da Silva, P., Dasari, S., Yang, T., Goose, S.: Eye-wearable technology for machine maintenance: effects of display position and hands-free operation. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 2125–2134. ACM (2015)
24. Aromaa, S., Aaltonen, I., Kaasinen, E., Elo, J., Parkkinen, I.: Use of wearable and augmented reality technologies in industrial maintenance work. In: *Proceedings of the 20th International Academic Mindtrek Conference*, pp. 235–242. ACM (2016)
25. Wieland, M., Kopp, O., Nicklas, D., Leymann, F.: Towards context-aware workflows. In: *CAiSE07 Proceedings of the Workshops and Doctoral Consortium*, vol. 2, no. 25, p. 78 (2007)
26. Wollschlaeger, M., Sauter, T., Jasperneite, J.: The future of industrial communication: automation networks in the era of the Internet of Things and Industry 4.0. *IEEE Ind. Electr. Mag.* **11**(1), 17–27 (2017)
27. Ehrlich, M., Wisniewski, L., Jasperneite, J.: State of the art and future applications of industrial wireless sensor networks. In: Jasperneite, J., Lohweg, W. (eds.) *Kommunikation und Bildverarbeitung in der Automation*. TA, pp. 28–39. Springer, Heidelberg (2018). [https://doi.org/10.1007/978-3-662-55232-2\\_3](https://doi.org/10.1007/978-3-662-55232-2_3)
28. Da Xu, L., He, W., Li, S.: Internet of Things in industries: a survey. *IEEE Trans. Ind. Inf.* **10**(4), 2233–2243 (2014)

29. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of Things (IoT): a vision, architectural elements, and future directions. *Future Gener. Comput. Syst.* **29**(7), 1645–1660 (2013)
30. Lee, J., Bagheri, B., Jin, C.: Introduction to cyber manufacturing. *Manuf. Lett.* **8**, 11–15 (2016)
31. Maier, M., Chowdhury, M., Rimal, B.P., Van, D.P.: The tactile internet: vision, recent progress, and open challenges. *IEEE Commun. Mag.* **54**(5), 138–145 (2016)
32. Gellersen, H.W., Schmidt, A., Beigl, M.: Multi-sensor context-awareness in mobile devices and smart artifacts. *Mob. Netw. Appl.* **7**(5), 341–351 (2002)
33. Henningsen, S., Dietzel, S., Scheuermann, B.: Misbehavior detection in industrial wireless networks: challenges and directions. *Mob. Netw. Appl.* 1–7 (2018)
34. Meshram, A., Haas, C.: Anomaly detection in industrial networks using machine learning: a roadmap. In: Beyerer, J., Niggemann, O., Kühnert, C. (eds.) *Machine Learning for Cyber Physical Systems*. TA, pp. 65–72. Springer, Heidelberg (2017). [https://doi.org/10.1007/978-3-662-53806-7\\_8](https://doi.org/10.1007/978-3-662-53806-7_8)
35. Fettweis, G., Alamouti, S.: 5G: Personal mobile internet beyond what cellular did to telephony. *IEEE Commun. Mag.* **52**(2), 140–145 (2014)
36. Simsek, M., Aijaz, A., Dohler, M., Sachs, J., Fettweis, G.: 5G-enabled tactile internet. *IEEE J. Sel. Areas Commun.* **34**(3), 460–473 (2016)
37. Gregori, F., Papetti, A., Pandolfi, M., Peruzzini, M., Germani, M.: Digital manufacturing systems: a framework to improve social sustainability of a production site. *Procedia CIRP* **63**, 436–442 (2017)
38. Hygge, S., Knez, I.: Effects of noise, heat and indoor lighting on cognitive performance and self-reported affect. *J. Environ. Psychol.* **21**(3), 291–299 (2001)
39. Renaud, K., Van Biljon, J.: Predicting technology acceptance and adoption by the elderly: a qualitative study. In: *Proceedings of the 2008 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries: Riding The Wave of Technology*, pp. 210–219. ACM (2008)
40. Davis, F.D.: User acceptance of information technology: system characteristics, user perceptions and behavioral impacts. *Int. J. Man-Mach. Stud.* **38**(3), 475–487 (1993)
41. Röcker, C.: Social and technological concerns associated with the usage of ubiquitous computing technologies. *Issues Inf. Syst.* **11**(1), 61–68 (2010)
42. Holzinger, A., Searle, G., Wernbacher, M.: The effect of previous exposure to technology on acceptance and its importance in usability and accessibility engineering. *Univ. Access Inf. Soc.* **10**(3), 245–260 (2011)
43. Holzinger, K., Mak, K., Kieseberg, P., Holzinger, A.: Can we trust machine learning results? Artificial intelligence in safety-critical decision support. *ERCIM NEWS* **112**, 42–43 (2018)
44. Holzinger, A., Biemann, C., Pattichis, C.S., Kell, D.B.: What do we need to build explainable AI systems for the medical domain? (2017). *arXiv preprint arXiv:1712.09923*
45. Levinson, A.R.: Industrial justice: privacy protection for the employed. *Cornell J. Law Public Policy* **18**, 609 (2008)
46. Kovach, D., Kenneth, A., Jordan, J., Tansey, K., Framiñan, E.: The balance between employee privacy and employer interests. *Bus. Soc. Rev.* **105**(2), 289–298 (2000)
47. Nord, G.D., McCubbins, T.F., Nord, J.H.: E-monitoring in the workplace: privacy, legislation, and surveillance software. *Commun. ACM* **49**(8), 72–77 (2006)
48. Kaupins, G., Minch, R.: Legal and ethical implications of employee location monitoring. In: *Proceedings of the 38th Annual Hawaii International Conference on System Sciences, HICSS 2005*, p. 133a. IEEE (2005)
49. Liu, H., Darabi, H., Banerjee, P., Liu, J.: Survey of wireless indoor positioning techniques and systems. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **37**(6), 1067–1080 (2007)

50. Sack, O., Röcker, C.: Privacy and security in technology-enhanced environments: exploring users' knowledge about technological processes of diverse user groups. *Univ. J. Psychol.* **1**(2), 72–83 (2013)
51. Röcker, C., Feith, A.: Revisiting privacy in smart spaces: social and architectural aspects of privacy in technology-enhanced environments. In: *Proceedings of the International Symposium on Computing, Communication and Control (ISCCC 2009)*, pp. 201–205 (2009)
52. Funk, M., Bächler, A., Bächler, L., Korn, O., Krieger, C., Heidenreich, T., Schmidt, A.: Comparing projected in-situ feedback at the manual assembly workplace with impaired workers. In: *Proceedings of the 8th ACM International Conference on Pervasive Technologies Related to Assistive Environments*, p. 1. ACM (2015)
53. Falzon, P.: Ergonomics, knowledge development and the design of enabling environments. In: *Humanizing Work and Work Environment Conference*, pp. 10–12 (2005)
54. Villani, V., Sabattini, L., Czerniak, J.N., Mertens, A., Vogel-Heuser, B., Fantuzzi, C.: Towards modern inclusive factories: a methodology for the development of smart adaptive human-machine interfaces (2017). *arXiv preprint* [arXiv:1706.08467](https://arxiv.org/abs/1706.08467)
55. Nee, A.Y., Ong, S.K., Chryssolouris, G., Mourtzis, D.: Augmented reality applications in design and manufacturing. *CIRP Ann.-Manuf. Technol.* **61**(2), 657–679 (2012)
56. Lauber, F., Butz, A.: Are HMDs the better HUDs? In: *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 267–268. IEEE (2013)
57. Cook, A.M., Polgar, J.M.: *Assistive Technologies-E-Book: Principles and Practice*. Elsevier Health Sciences, New York City (2014)
58. Korn, O., Funk, M., Schmidt, A.: Assistive systems for the workplace: towards context-aware assistance. In: *Assistive Technologies for Physical and Cognitive Disabilities*, pp. 121–133 (2015)
59. Korn, O.: Industrial playgrounds: how gamification helps to enrich work for elderly or impaired persons in production. In: *Proceedings of the 4th ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, pp. 313–316. ACM (2012)
60. [http://www.gesetze-im-internet.de/bitv\\_2\\_0/BJNR184300011.html](http://www.gesetze-im-internet.de/bitv_2_0/BJNR184300011.html)



# **MAKE-Topology**



# On a New Method to Build Group Equivariant Operators by Means of Permutants

Francesco Camporesi, Patrizio Frosini, and Nicola Quercioli<sup>(✉)</sup>

Department of Mathematics, University of Bologna, Bologna, Italy  
francesco.camporesi@studio.unibo.it,  
{patrizio.frosini,nicola.quercioli2}@unibo.it

**Abstract.** The use of group equivariant operators is becoming more and more important in machine learning and topological data analysis. In this paper we introduce a new method to build  $G$ -equivariant non-expansive operators from a set  $\Phi$  of bounded and continuous functions  $\varphi : X \rightarrow \mathbb{R}$  to  $\Phi$  itself, where  $X$  is a topological space and  $G$  is a subgroup of the group of all self-homeomorphisms of  $X$ .

**Keywords:** Natural pseudo-distance · Filtering function  
Group action · Group equivariant non-expansive operator  
Persistent homology group · Topological data analysis

## 1 Introduction

In the last years the problem of data analysis has assumed a more and more relevant role in science, and many researchers have started to become interested in it from several different points of view. Some geometrical techniques have given their contribute to this topic, and persistent homology has proven itself quite efficient both for qualitative and topological comparison of data [5]. In particular, topological data analysis (TDA) has revealed important in managing the huge amount of data that surrounds us in the most varied contexts [3]. The use of TDA is based on the fact that in several practical situations the measurements of interest can be expressed by continuous  $\mathbb{R}^m$ -valued functions defined on a topological space, as happens for the weight of a physical body or a biomedical image [2]. However, for the sake of simplicity, in this work we will focus on real-valued functions. The continuity of the considered functions enables us to apply persistent homology, a theory that studies the birth and the death of  $k$ -dimensional holes when we move along the filtration defined by the sublevel sets of a continuous function from a topological space  $X$  to the real numbers. Interestingly, this procedure is invariant with respect to all homeomorphisms of  $X$ , that is if  $g \in \text{Homeo}(X)$ , then  $\varphi$  and  $\varphi \circ g$  induce on  $X$  two filtrations which have exactly the same topological properties under the point of view of

persistent homology. For further and more detailed information about persistent homology, we refer the reader to [6].

The importance of group equivariance in machine learning is well-known (cf., e.g., [1, 4, 10, 11]). The study of group equivariant non-expansive operators (GENEOs) proposed in this work could be a first step in the path to establishing a link between persistence theory and machine learning. The ground idea is that the observer influences in a direct way the act of measurement, and that our analysis should be mainly focused on a good approximation of the observer rather than on a precise description of the data [7]. GENEOs reflect the way the information is processed by the observer, and hence they enclose the invariance the observer is interested in. In some sense, we could say that an observer can be seen as a collection of group equivariant non-expansive operators acting on suitable spaces of data. The choice of the invariance group  $G$  is a key point in this model. For example, in character recognition the invariance group should not contain reflections with respect to a vertical axis, since the symbols ‘p’ and ‘q’ should not be considered equal to each other, while this fact does not hold for the comparison of medieval rose windows.

The use of invariance groups leads us to rely on the concept of *natural pseudo-distance*. Let us consider a set  $\Phi$  of continuous  $\mathbb{R}$ -valued functions defined on a topological space  $X$  and a subgroup  $G$  of the group  $\text{Homeo}(X)$  of all self-homeomorphisms of  $X$ . We assume that the group  $G$  acts on  $\Phi$  by composition on the right. Now we can define the *natural pseudo-distance*  $d_G$  on  $\Phi$  by setting  $d_G(\varphi_1, \varphi_2) = \inf_{g \in G} \|\varphi_1 - \varphi_2 \circ g\|_\infty$ , where  $\|\cdot\|_\infty$  denotes the sup-norm. Although the natural pseudo-distance reflects our intent to find the best correspondence between two functions of  $\Phi$ , unfortunately it leads to some practical limitations since it is difficult to compute, even when the group  $G$  has good properties.

However, the theory of group equivariant non-expansive operators makes available a method for the approximation of the natural pseudo-distance (cf. Theorem 1 in this paper). Moreover, in [8, 9] it has been proven that under suitable hypotheses the space  $\mathcal{F}(\Phi, G)$  of all GENEOs benefits from good computational properties, such as compactness and convexity. In order to proceed in the research about this space of operators, we devote this paper to introducing a new method to construct GENEOs by means of particular subsets of  $\text{Homeo}(X)$ , called *permutants*. We underline that in our method we can treat the group of invariance as a variable. This is important because the change of the observer generally corresponds to a change of the invariance we want to analyze.

Our work is organized as follows. In Sect. 2 we start explaining the mathematical setting where our research will take place. In Sect. 3 we introduce our new method for the construction of group equivariant non-expansive operators. In particular, we show how specific subsets of  $\text{Homeo}(X)$  called permutants can help us in this procedure. In Sect. 4 we illustrate our method by giving two examples. Finally, in Sect. 5 we explore the limits of our approach by proving a result about permutants.

## 2 Our Mathematical Model

In this section we recall the mathematical model illustrated in [8]. Let us consider a (non-empty) topological space  $X$ , and the topological space  $C_b^0(X, \mathbb{R})$  of the continuous bounded functions from  $X$  to  $\mathbb{R}$ , endowed with the topology induced by the sup-norm  $\|\cdot\|_\infty$ . Let  $\Phi$  be a topological subspace of  $C_b^0(X, \mathbb{R})$ , whose elements represent our data. The functions in  $\Phi$  will be called *admissible filtering functions* on the space  $X$ . We are interested in analyzing  $\Phi$  by applying the invariance with respect to a subgroup  $G$  of the group  $\text{Homeo}(X)$  of all self-homeomorphisms of  $X$ . The group  $G$  is used to act on  $\Phi$  by composition on the right, i.e. we assume that for every  $\varphi \in \Phi$  and every  $g \in G$  the map  $\varphi \circ g$  is still in  $\Phi$ . In other words, we consider the functions  $\varphi, \varphi \circ g \in \Phi$  equivalent to each other for every  $g \in G$ .

A pseudo-metric that can be used to compare functions in this mathematical setting is the *natural pseudo-distance*  $d_G$ .

**Definition 1.** We set  $d_G(\varphi_1, \varphi_2) := \inf_{g \in G} \max_{x \in X} |\varphi_1(x) - \varphi_2(g(x))|$  for every  $\varphi_1, \varphi_2 \in \Phi$ . The function  $d_G$  is called the natural pseudo-distance associated with the group  $G$  acting on  $\Phi$ .

The previous pseudo-metric can be seen as the ground truth for the comparison of functions in  $\Phi$  with respect to the action of the group  $G$ . Unfortunately,  $d_G$  is usually difficult to compute. However, a method to study the natural pseudo-distance via *G-equivariant non-expansive operators* is available.

**Definition 2.** A *G-equivariant non-expansive operator (GENEO)* for the pair  $(\Phi, G)$  is a function

$$F : \Phi \longrightarrow \Phi$$

that satisfies the following properties:

1.  $F(\varphi \circ g) = F(\varphi) \circ g, \quad \forall \varphi \in \Phi, \quad \forall g \in G;$
2.  $\|F(\varphi_1) - F(\varphi_2)\|_\infty \leq \|\varphi_1 - \varphi_2\|_\infty, \quad \forall \varphi_1, \varphi_2 \in \Phi.$

The first property represents our request of equivariance with respect to the action of  $G$ , while the second one highlights the non-expansivity of the operator, since we require a control on the norm. We define  $\mathcal{F}(\Phi, G)$  to be the set of all  $G$ -equivariant non-expansive operators for  $(\Phi, G)$ . Obviously  $\mathcal{F}(\Phi, G)$  is not empty because it contains at least the identity operator.

*Remark 1.* The non-expansivity property implies that the operators in  $\mathcal{F}(\Phi, G)$  are 1-Lipschitz and hence continuous. We highlight that GENEOs are not required to be linear, even though all the GENEOs exposed in this paper have this property.

The following key property holds, provided that  $X$  has nontrivial homology in degree  $k$  and  $\Phi$  contains all the constant functions  $c$  from  $X$  to  $\mathbb{R}$  such that there exists  $\varphi \in \Phi$  with  $c \leq \|\varphi\|_\infty$  [8].

**Theorem 1.** *If  $\mathcal{F}$  is the set of all  $G$ -equivariant non-expansive operators for the pair  $(\Phi, G)$ , then  $d_G(\varphi_1, \varphi_2) = \sup_{F \in \mathcal{F}} d_{\text{match}}(r_k(F(\varphi_1)), r_k(F(\varphi_2)))$ , where  $r_k(\varphi)$  denotes the  $k$ -th persistent Betti number function with respect to the function  $\varphi : X \rightarrow \mathbb{R}$  and  $d_{\text{match}}$  is the classical matching distance.*

Theorem 1 represents a strong link between persistent homology and the natural pseudo-distance via GENEOS. It establishes a method to compute  $d_G$  by means of  $G$ -equivariant non-expansive operators. As a consequence, the construction of GENEOS is an important step in the computation of the natural pseudo-distance. This fact justifies the interest for the result proven in Sect. 3.

### 3 A Method to Build GENEOS by Means of Permutants

In this section we introduce a new method for the construction of GENEOS, exploiting the concept of permutant. Let  $G$  be a subgroup of  $\text{Homeo}(X)$ . We consider the conjugation map

$$\begin{aligned} \alpha_g : \text{Homeo}(X) &\rightarrow \text{Homeo}(X) \\ f &\mapsto g \circ f \circ g^{-1} \end{aligned}$$

where  $g$  is an element of  $G$ .

**Definition 3.** *A non-empty finite subset  $H$  of  $\text{Homeo}(X)$  is said to be a permutant for  $G$  if  $\alpha_g(H) \subseteq H$  for every  $g \in G$ .*

*Remark 2.* The condition  $\alpha_g(H) \subseteq H$ , the finiteness of  $H$  and the injectivity of  $\alpha_g$  imply that  $\alpha_g$  is a permutation of the set  $H$  for every  $g \in G$ . Moreover, it is important to note that  $H$  is required neither to be a subset of the invariance group  $G$ , nor a subgroup of  $\text{Homeo}(X)$ .

*Remark 3.* If  $H$  and  $K$  are two permutants for  $G$ , then also the union  $H \cup K$  and the intersection  $H \cap K$  are two permutants for  $G$  (provided that  $H \cap K \neq \emptyset$ ).

If  $H = \{h_1, \dots, h_n\}$  is a permutant for  $G$  and  $\bar{a} \in \mathbb{R}$  with  $n|\bar{a}| \leq 1$ , we can consider the operator  $F_{\bar{a}, H} : C_b^0(X, \mathbb{R}) \rightarrow C_b^0(X, \mathbb{R})$  defined by setting

$$F_{\bar{a}, H}(\varphi) := \bar{a} \sum_{i=1}^n (\varphi \circ h_i).$$

The following statement holds.

**Proposition 1.** *If  $F_{\bar{a}, H}(\Phi) \subseteq \Phi$  then  $F_{\bar{a}, H}$  is a GENEOS for  $(\Phi, G)$ .*

*Proof.* First of all we prove that  $F_{\bar{a}, H}$  is  $G$ -equivariant. Let  $\tilde{\alpha}_g : \{1, \dots, n\} \rightarrow \{1, \dots, n\}$  be an index permutation such that  $\tilde{\alpha}_g(i)$  is the index of the image of  $h_i$  through the conjugacy action of  $g$ , i.e.

$$\alpha_g(h_i) = g \circ h_i \circ g^{-1} = h_{\tilde{\alpha}_g(i)}, \quad \forall i \in \{1, \dots, n\}.$$

We obtain that

$$g \circ h_i = h_{\tilde{\alpha}_g(i)} \circ g.$$

Exploiting this relation we obtain that

$$\begin{aligned} F_{\bar{a},H}(\varphi \circ g) &= \bar{a}(\varphi \circ g \circ h_1 + \dots + \varphi \circ g \circ h_n) \\ &= \bar{a}(\varphi \circ h_{\tilde{\alpha}_g(1)} \circ g + \dots + \varphi \circ h_{\tilde{\alpha}_g(n)} \circ g) \\ &= \bar{a}(\varphi \circ h_{\tilde{\alpha}_g(1)} + \dots + \varphi \circ h_{\tilde{\alpha}_g(n)}) \circ g. \end{aligned}$$

Since  $\{h_{\tilde{\alpha}_g(1)}, \dots, h_{\tilde{\alpha}_g(n)}\} = \{h_1, \dots, h_n\}$ , we get

$$F_{\bar{a},H}(\varphi \circ g) = F_{\bar{a},H}(\varphi) \circ g, \quad \forall \varphi \in \Phi, \quad \forall g \in G.$$

It remains to show that  $F_{\bar{a},H}$  is non-expansive:

$$\begin{aligned} \|F_{\bar{a},H}(\varphi_1) - F_{\bar{a},H}(\varphi_2)\|_\infty &= \left\| \bar{a} \sum_{i=1}^n (\varphi_1 \circ h_i) - \bar{a} \sum_{i=1}^n (\varphi_2 \circ h_i) \right\|_\infty \\ &= |\bar{a}| \left\| \sum_{i=1}^n (\varphi_1 \circ h_i - \varphi_2 \circ h_i) \right\|_\infty \\ &\leq |\bar{a}| \sum_{i=1}^n \|\varphi_1 \circ h_i - \varphi_2 \circ h_i\|_\infty \\ &= |\bar{a}| \sum_{i=1}^n \|\varphi_1 - \varphi_2\|_\infty \\ &= n|\bar{a}| \|\varphi_1 - \varphi_2\|_\infty \\ &\leq \|\varphi_1 - \varphi_2\|_\infty \end{aligned}$$

for every  $\varphi_1, \varphi_2 \in \Phi$ .

*Remark 4.* Obviously  $H = \{id\} \subseteq \text{Homeo}(X)$  is a permutant for every subgroup  $G$  of  $\text{Homeo}(X)$ , but the use of Proposition 1 for this trivial permutant leads to the trivial operator given by a multiple of the identity operator on  $\Phi$ .

*Remark 5.* If the group  $G$  is Abelian, every finite subset of  $G$  is a permutant for  $G$ , since the conjugacy action is just the identity. Hence in this setting, for any chosen finite subset  $H = \{g_1, \dots, g_n\}$  of  $G$  and any real number  $\bar{a}$ , such that  $n|\bar{a}| \leq 1$ ,  $F_{\bar{a},H}(\varphi) = \bar{a}(\varphi \circ g_1 + \dots + \varphi \circ g_n)$  is a  $G$ -equivariant non-expansive operator for  $(\Phi, G)$ , provided that  $F_{\bar{a},H}$  preserves  $\Phi$ .

*Remark 6.* The operator  $F_{\bar{a},H} : \Phi \rightarrow \Phi$  introduced in Proposition 1 is linear, provided that  $\Phi$  is linearly closed. Indeed, assume that a permutant  $H = \{h_1, \dots, h_n\}$  for  $G$  and a real number  $\bar{a}$  such that  $n|\bar{a}| \leq 1$  are given.

Let us consider the associated operator  $F_{\bar{a},H}(\varphi) = \bar{a} \sum_{i=1}^n (\varphi \circ h_i)$ , and assume that  $F_{\bar{a},H}(\Phi) \subseteq \Phi$ . If  $\lambda_1, \lambda_2 \in \mathbb{R}$  and  $\varphi_1, \varphi_2 \in \Phi$ , we have

$$\begin{aligned} F_{\bar{a},H}(\lambda_1\varphi_1 + \lambda_2\varphi_2) &= \bar{a} \sum_{i=1}^n ((\lambda_1\varphi_1 + \lambda_2\varphi_2) \circ h_i) \\ &= \bar{a} \sum_{i=1}^n (\lambda_1(\varphi_1 \circ h_i) + \lambda_2(\varphi_2 \circ h_i)) \\ &= \bar{a} \sum_{i=1}^n \lambda_1(\varphi_1 \circ h_i) + \bar{a} \sum_{i=1}^n \lambda_2(\varphi_2 \circ h_i) \\ &= \lambda_1 \left[ \bar{a} \sum_{i=1}^n (\varphi_1 \circ h_i) \right] + \lambda_2 \left[ \bar{a} \sum_{i=1}^n (\varphi_2 \circ h_i) \right] \\ &= \lambda_1 F_{\bar{a},H}(\varphi_1) + \lambda_2 F_{\bar{a},H}(\varphi_2). \end{aligned}$$

### 4 Examples

In this section we give two examples illustrating our method to build GENEOS.

*Example 1.* Let  $X = \mathbb{R}$  and  $\Phi \subseteq C_b^0(X, \mathbb{R})$ . We consider the group  $G$  of all isometries of the real line, i.e. homeomorphisms of  $\mathbb{R}$  of the form

$$g(x) = ax + b, \quad a, b \in \mathbb{R}, \quad a = \pm 1.$$

We also consider a translation  $h(x) = x + t$  and its inverse transformation  $h^{-1}(x) = x - t$ , for some nonzero  $t \in \mathbb{R}$ . If  $g$  preserves the orientation, i.e.  $a = 1$ , the conjugation by  $g$  acts on  $H := \{h, h^{-1}\}$  as the identity, while for  $a = -1$  this conjugation exchanges the elements of  $H$ . We can conclude that  $H$  is a permutant for  $G$ . Therefore, Proposition 1 guarantees that the operator  $F_{\frac{1}{2},H}(\varphi) = \frac{1}{2}(\varphi \circ h + \varphi \circ h^{-1})$  is a GENEIO for  $(\Phi, G)$ , provided that  $F_{\frac{1}{2},H}(\Phi) \subseteq \Phi$ . We observe that the permutant used in this example is a subset but not a subgroup of  $\text{Homeo}(X)$ .

*Example 2.* Let  $X = \{(x, y) \in \mathbb{R}^2 : x^2 + y^2 = 1\}$  and assume that  $\Phi$  is the set of 1-Lipschitzian functions from  $X$  to  $[0, 1]$ . Let  $G$  and  $H$  be the group generated by reflection with respect to the line  $x = 0$  and the group generated by the rotation  $\rho$  of  $\pi/2$  around the point  $(0, 0)$ , respectively. It is easy to check that  $H = \{id, \rho, \rho^2, \rho^3\}$  is a permutant for  $G$  and  $F_{\frac{1}{4},H}(\Phi) \subseteq \Phi$ . Therefore, Proposition 1 guarantees that the operator  $F_{\frac{1}{4},H}(\varphi) = \frac{1}{4}(\varphi + \varphi \circ \rho + \varphi \circ \rho^2 + \varphi \circ \rho^3)$  is a GENEIO for  $(\Phi, G)$ . We observe that the permutant used in this example is a subgroup of  $\text{Homeo}(X)$  but not a subgroup of  $G$ .

## 5 A Result Concerning Permutants

When  $H$  contains only the identical homeomorphism, the operator  $F_{\bar{a},H}$  is trivial, since it is the multiple by the constant  $\bar{a}$  of the identical operator. This section highlights that in some cases this situation cannot be avoided, since non-trivial permutants for  $G$  are not available. In order to illustrate this problem, we need to introduce the concept of *versatile* group.

**Definition 4.** Let  $G$  be a group that acts on a set  $X$ . We say that  $G$  is versatile if for every triple  $(x, y, z) \in X^3$ , with  $x \neq z$ , and for every finite subset  $S$  of  $X$ , at least one element  $g \in G$  exists such that (1)  $g(x) = y$  and (2)  $g(z) \notin S$ .

**Proposition 2.** Let  $X$  be a topological space and assume that  $H = \{h_1, \dots, h_n\}$  is a permutant for a subgroup  $G$  of  $\text{Homeo}(X)$ . If  $G$  is versatile, then  $H = \{id\}$ .

*Proof.* It is sufficient to prove that if  $H$  contains an element  $h \neq id$ , then  $G$  is not versatile. We can assume that  $h \equiv h_1$ . Since  $h_1$  is different from the identity, a point  $\bar{x} \in X$  exists such that  $h_1(\bar{x}) \neq \bar{x}$ . Let us consider the triple  $(h_1(\bar{x}), \bar{x}, \bar{x})$  and the set  $S = \{h_1^{-1}(\bar{x}), \dots, h_n^{-1}(\bar{x})\}$ . Suppose that  $g \in G$  satisfies Property (1) with respect to the previous triple, that is  $g(h_1(\bar{x})) = \bar{x}$ . Since the conjugacy action of  $g$  on  $H$  is a permutation, we can find an element  $h_2 \in H$  such that  $h_2 = g \circ h_1 \circ g^{-1}$ , so that  $h_2(g(\bar{x})) = g(h_1(\bar{x})) = \bar{x}$  and hence  $g(\bar{x}) = h_2^{-1}(\bar{x}) \in S$ . Therefore,  $g$  does not satisfy Property (2), for  $z = \bar{x}$ . Hence we can conclude that no  $g \in G$  exists verifying both Properties (1) and (2), i.e.  $G$  is not versatile.

*Remark 7.* Definition 4 immediately implies that if  $G, G'$  are two subgroups of  $\text{Homeo}(X)$ ,  $G \subseteq G'$  and  $G$  is versatile, then also the group  $G'$  is versatile. For example, it is easy to prove that the group  $G$  of the isometries of the real plane is versatile. It follows that every group  $G'$  of self-homeomorphisms of  $\mathbb{R}^2$  containing the isometries of the real plane is versatile. As a consequence of Proposition 2, every permutant for  $G'$  is trivial.

## 6 Conclusions

In this paper we have illustrated a new method for the construction of group equivariant non-expansive operators by means of permutants, exploiting the algebraic properties of the invariance group. The procedure enables us to manage in a quite simple way Abelian groups, but our examples show that we can find permutants, and hence GENEOS, even in a non-commutative setting. The main goal of our study is to expand our knowledge about the topological space  $\mathcal{F}(\Phi, G)$ , possibly reaching a good approximation of this space and, consequently, a good approximation of the pseudo-natural distance  $d_G$  by means of Theorem 1. The more operators we know, the more information we get about the structure of  $\mathcal{F}(\Phi, G)$ , and this fact justifies the search for new methods to build GENEOS. Many questions remain open. In particular, a deeper study of the concept of permutant seems necessary, establishing conditions for the existence of non-trivial



permutants and introducing constructive methods to build them. Furthermore, an extension of our approach to operators from a pair  $(\Phi, G)$  to a different pair  $(\Psi, H)$  seems worth of further investigation. Finally, we should check if the idea described in this paper about getting GENEOS by a finite average based on the use of permutants could be generalized to “infinite averages” (and hence integrals) based on “infinite permutants”.

**Acknowledgment.** The authors thank Marian Mrozek for his suggestions and advice. The research described in this article has been partially supported by GNSAGA-INdAM (Italy).

## References

1. Anselmi, F., Rosasco, L., Poggio, T.: On invariance and selectivity in representation learning. *Inf. Infer.: J. IMA* **5**(2), 134–158 (2016)
2. Biasotti, S.: Describing shapes by geometrical-topological properties of real functions. *ACM Comput. Surv.* **40**(4), 12:1–12:87 (2008)
3. Carlsson, G.: Topology and data. *Bull. Am. Math. Soc. (N.S.)* **46**(2), 255–308 (2009)
4. Cohen, T., Welling, M.: Group equivariant convolutional networks. In: *Proceedings of the 33rd International Conference on Machine Learning*, PMLR, vol. 48, pp. 2990–2999 (2016)
5. Edelsbrunner, H., Morozov, D.: Persistent homology: theory and practice. In: *European Congress of Mathematics*, pp. 31–50 (2013)
6. Edelsbrunner, H., Harer, J.L.: Persistent homology—a survey. *Contemp. Math.* **453**, 257–282 (2008)
7. Frosini, P.: Towards an observer-oriented theory of shape comparison. In: Ferreira, A., Giachetti, A., Giorgi, D. (eds.) *Proceedings of the 8th Eurographics Workshop on 3D Object Retrieval*, Lisbon, Portugal, pp. 5–8 (2016)
8. Frosini, P., Jabłoński, G.: Combining persistent homology and invariance groups for shape comparison. *Discret. Comput. Geom.* **55**(2), 373–409 (2016)
9. Frosini, P., Quercioli, N.: Some remarks on the algebraic properties of group invariant operators in persistent homology. In: Holzinger, A., Kieseberg, P., Tjoa, A.M., Weippl, E. (eds.) *CD-MAKE 2017*. LNCS, vol. 10410, pp. 14–24. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-66808-6\\_2](https://doi.org/10.1007/978-3-319-66808-6_2)
10. Marcos, D., Volpi, M., Komodakis, N., Tuia, D.: Rotation equivariant vector field networks. In: *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5058–5067 (2017)
11. Masci, J., Boscaini, D., Bronstein, M.M., Vandergheynst, P.: Geodesic convolutional neural networks on Riemannian manifolds. In: *Proceedings of the 2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pp. 832–840. IEEE Computer Society (2015)



# Topological Characteristics of Digital Models of Geological Core

Rustem R. Gilmanov<sup>1(✉)</sup>, Alexander V. Kalyuzhnyuk<sup>2(✉)</sup>,  
Iskander A. Taimanov<sup>3,4(✉)</sup>, and Andrey A. Yakovlev<sup>1(✉)</sup>

<sup>1</sup> OOO “Gazpromneft NTC”, 190000 St. Petersburg, Russia  
{Gilmanov.RR, Yakovlev.AA, Ale}@gazpromneft-ntc.ru

<sup>2</sup> Peter the Great St. Petersburg Polytechnic University,  
195251 St. Petersburg, Russia

Kalyuzhnyuk.AV@gazprom-neft.ru

<sup>3</sup> Chebyshev Laboratory at St. Petersburg State University,  
199178 St. Petersburg, Russia

<sup>4</sup> Sobolev Institute of Mathematics, 630090 Novosibirsk, Russia  
taimanov@math.nsc.ru

**Abstract.** We discuss the possibility of applying stochastic approaches to core modeling by using tools of topology. The study demonstrates the prospects of applying topological characteristics for the description of the core and the search for its analogs. Moreover application of topological characteristics (for example, in conjunction with machine learning methods) in the long term will make it possible to obtain petrophysical properties of the core samples without carrying out expensive and long-term filtration experiments.

**Keywords:** Topological characteristics · Betti numbers · Digital core  
Geological modeling

Nowadays the process of oil fields exploitation needs a continuous information support. This is especially necessary in the context of the emphasis shift in the development, planning and monitoring of oil and gas fields on very highly dissected and low-permeability reservoirs.

More accurate estimation of economic efficiency and optimal placement of production wells are possible if we have geological picture of the oil field. This means that mathematical measure of the geological modeling of such objects is needed.

These estimation processes include a geological and hydrodynamic modeling. Adjustment of models includes:

- static well data – logging curves, core analysis, drilling data etc.;
- dynamic data – well flow, bottomhole pressure etc.

In practice, the adjustment of models occurs iteratively – by the numerical simulation of series of direct resource-intensive tasks. The speed of such adjustment depends on geological model quality. To measure quality of such model, a mathematical method is needed to describe its “heterogeneity” and “internal complexity”. It is necessary to find a proper cell size, variogram radius, experimental data correlation etc.

Similar problems also exist in the digital modeling of core samples – rock samples extracted from the well. In modern practice, there are some tools for measuring of the heterogeneity of such models [1]:

- construction of dissect's maps;
- spectral modeling of logging curves.

However, these methods do not provide numerical characteristics (metric, measure) of a constructed 3-D model. In [2, 3] it was proposed to consider topological characteristics of these models. Here we use this approach to study digital models of geological core sample. The topological characteristics of core samples are compared with the topological characteristics of geological models.

## 1 Topological Characteristics of Three-Dimensional Digital Solid Body Models

Topological characteristics of three-dimensional digital solid body models in the study are defined with the same mathematical apparatus as in [3]. These models represent ordered sets of elementary cubes – cubic complexes. The solid bodies are not topologically equivalent if their Betti numbers  $b_0$ ,  $b_1$  and  $b_2$  are different. The meanings of these topological characteristics are follows:

- $b_0$  is the number of connected components;
- $b_1$  is the number of handles;
- $b_2$  is the number of holes (cavities).

In this paper, two elementary cubes are considered to belong to the same linear component only in case of intersecting each other by a joint face. Intersecting by a joint vertex or edge does not make them belong to the same linear component.

If we start from a solid cube, remove  $k$  holes from its interior and attach  $l$  handles to the cube we obtain the body  $X$  for which  $b_0 = 1$ ,  $b_1 = l$ ,  $b_2 = k$ .

Calculation of the Betti numbers for these cubic complexes was implemented via the numerical algorithm from [3].

## 2 Digital Core Model Analysis

### 2.1 Digital Core Model Description

The digital core model is a model obtained as a result of computer tomography.

Method is based on computer-processed combinations of many X-ray measurements taken from various angles to produce cross-sectional images of a scanned object, allowing to see inside the object without destruction [4]. Today, computed tomography is an evolving method for studying the petrographic properties of rocks. The X-ray tomography method allows to solve a huge number of geological problems, such as modeling cavities (fracture, caverns, pores), calculating the porosity, studying rock

heterogeneity, analysis of reservoir properties, measurement as core's volumes as all its voids and solids.

X-ray tomography has been used in oil industry since 80-s years [5]. The first studies has been conducted in Australia, USA, and Great Britain [6].

The result of the X-ray tomography of a core sample is a set of grayscale snapshots representing corresponding virtual sections [7]. Each snapshot point shows its radiodensity. A combination of these grayscale snapshots is used then to generate 3-dimensional radiodensity distribution of the sample in the volume [8].

## 2.2 Digital Core Model Creation Process

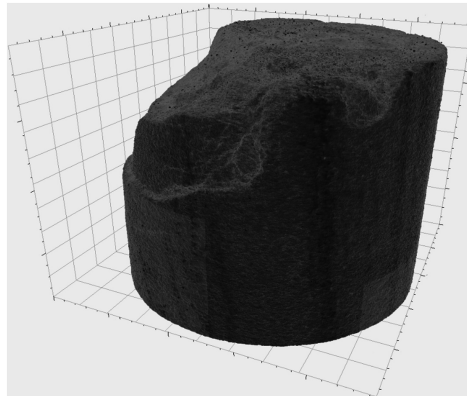
A core sample with a diameter of 46 mm and 7.8 mm high was used in the study.

Tomography snapshots are taken with 0.025 mm interval for inline and crossline sections and with 0.004 mm interval for vertical sections.

After the tomography scan, the result was stored in the SEG-Y format, commonly used to encode the results of seismic studies. This format is convenient for further analysis, because it permits to import the results in the form of voxels: a three-dimensional array, where each element corresponds to the value of the radiodensity (see Hounsfield scale [9]).

The value of radiodensity was taken in conventional units of radiodensity - certain number given by the device for computed tomography.

The core sample was imported into the Petrel (Schlumberger's oil engineering software package) (see Fig. 1). SEG-Y format is also useful to show virtual cross-sections of a digital via Petrel (see Fig. 2).

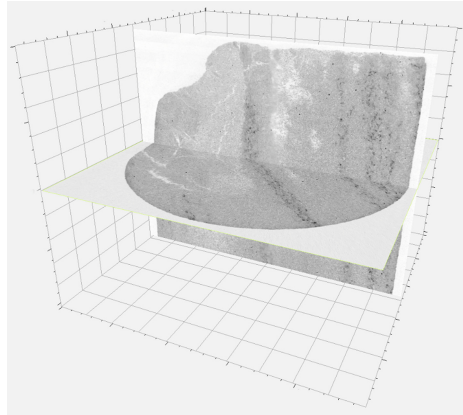


**Fig. 1.** Terrigenous core sample – 3D view of the sample (vertical axe is scaled).

## 2.3 Digital Core Model Processing

Three 3D cube-samples with the size of  $100 \times 100 \times 100$  voxels are cut from the core model (Fig. 2). The principle of “indicator formalism” is applied to the radiodensity values contained in voxels of cut cubes [10]. The range of voxel values is linearly mapped

to the segment [0, 1], so that the smallest of the radiodensity values is mapped to 0, the largest value is set to 1. These values are hereinafter referred to as “normalized radiodensity”.

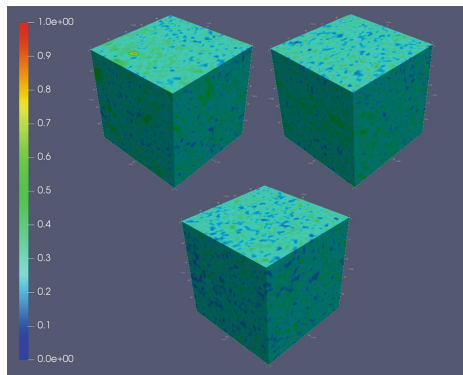


**Fig. 2.** Terrigenous core sample inline cross-section (vertical axe is scaled).

These cube-samples are transformed to a.grdecl format to evaluate their topological characteristics and to a.vtk format to visualize them using the Paraview software.

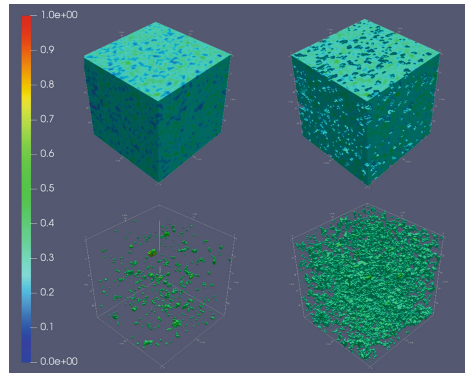
To calculate the model’s topological characteristics, it is necessary to classify voxels - to divide them according to the “skeleton of the rock” - “void”. Voxels, marked as “skeleton of the rock”, are then considered equivalent to elementary cubes from item 1.

The built-in material classifier of the tomography apparatus is not used in the study. For our purpose, we used an excursion parameter  $\alpha$ , the value of which varies from 0 to 1 [3] (Fig. 3).



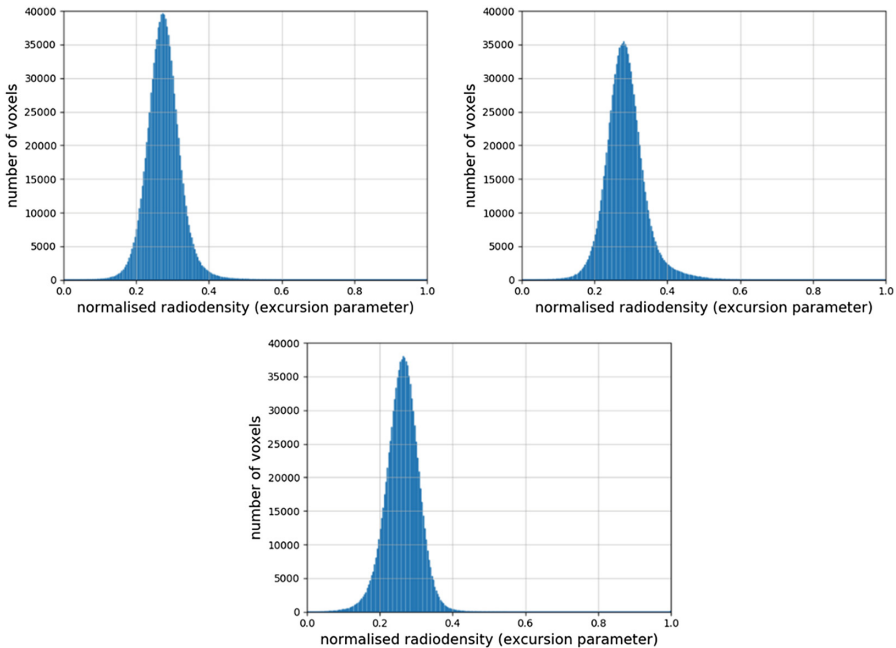
**Fig. 3.** Voxel cubes cut from the digital core model to calculating topological invariants (clockwise - 1st, 2nd and 3rd cube).

The application of the excursion parameter  $\alpha$  to the voxel model generates a cubic complex, where an elementary cube is a voxel whose normalized radiodensity value is greater than  $\alpha$  (Fig. 4).



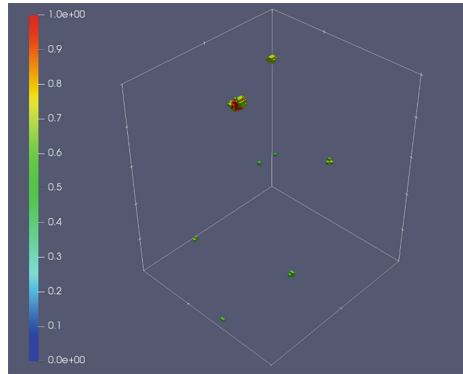
**Fig. 4.** An example of a core model separation of the “rock skeleton” - “void” for the first voxel cube cut from the digital core model (clockwise, the excursion parameter is 0, 0.2, 0.3, 0.4).

Histograms of the normalized radiodensity distribution for core samples show that the number of voxels with values greater than 0.6 for all samples is smaller than other values. It was expected that the topological characteristics obtained with these values of the excursion parameter will be negligibly small (Fig. 5).



**Fig. 5.** Histograms of the normalized radiodensity distribution for core samples (clockwise - 1st, 2nd and 3rd cube).

That is because if the excursion parameter is greater than 0.6, voxels marked as “rock skeleton” belong to local consolidations with the highest density values (Fig. 6).



**Fig. 6.** Local consolidations inside the digital core sample.

Also these histograms show the difference between second cube dispersion:

$$D_{2nd\ cube} = 2.86 * 10^{-3} \tag{1}$$

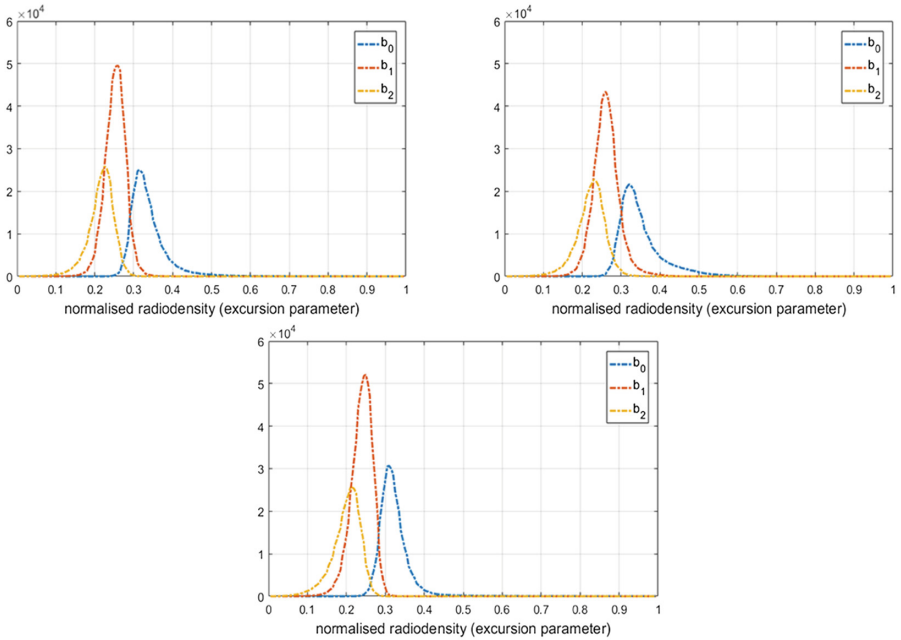
and first, and third cube dispersion:

$$D_{1st\ cube} = D_{3rd\ cube} = 2.15 * 10^{-3} \tag{2}$$

### 2.4 Topological Characteristics of Digital Core Models Evaluation

Topological characteristics of three-dimensional digital core models in the paper are evaluated similarly to topological characteristics of digital geological models in [3]. As expected from histograms (Fig. 5), the topological characteristics barely change if the excursion parameter is greater than 0.6. That is because the topological characteristics are close to constant while the excursion parameter is increasing (Fig. 7).

The calculated Betti numbers allow to consider digital core models as realizations of random fields – the same way as stochastic digital geological models [3]. It is shown on a 3-D plot of topological characteristics which axes are  $b_0$ ,  $b_1$ , and  $b_2$ , and the parameter is the excursion (Fig. 8). Shape of the obtained “curves” for the first and the third cube is similar to a “curve”, obtained for the Gaussian variogram realization of a geological random field (Fig. 9).



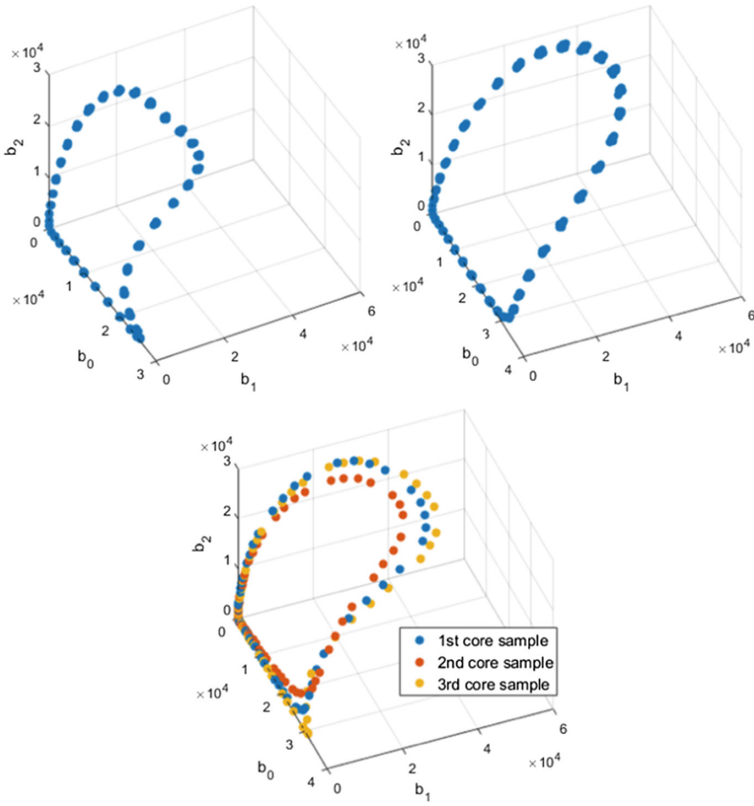
**Fig. 7.** Relationship between the Betti numbers and the excursion parameter (clockwise - 1st, 2nd and 3rd cube).

Analogously for the second cube - shape of the obtained “curve” is similar to a “curve”, obtained for the exponential variogram realization of a geological random field (Fig. 9).

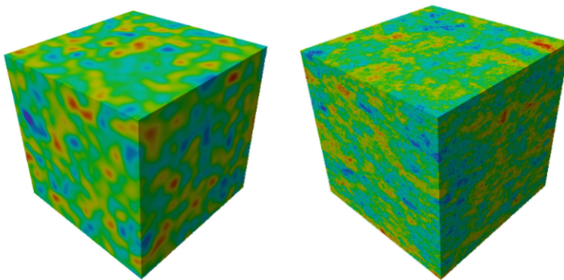
In both cases variogram radiuses are little in comparison of linear size of a model: variogram radius does not exceed 2% of linear size of a model. It shows a high compartmentalization of these models.

The differences obtained in these “curves” shape indicate an internal core heterogeneity. This difference in the inner complexity of the second cube from the first and third can be caused by an acid treatment carried out on this core sample.





**Fig. 8.** Topological characteristics in the Betti number axes (clockwise – digital geological model with a Gaussian variogram, digital geological model with an exponential variogram, digital core).



**Fig. 9.** Realization of a geological random field with a Gaussian (left) and exponential (right) variograms.

### 3 Conclusion

Analysis of topological characteristics of digital core models allows to:

- find regions of internal core heterogeneity;
- consider digital core models as realizations of digital geological stochastic models. That is how a developed geostatistics methodology can be used in digital core analysis.

The inner complexity and the Betti numbers dependencies for digital core models are going to be researched with a larger amount of cut cubes. The assessment by a specialist is required in terms of their inner structure - porosity, the presence of cracks and caverns.

### References

1. Hasanov, M.M., Belozerov, B.V., Bochkov, A.S., Fuks, O.M., Tengeli, D.I.: Automation of lithological-facies analysis on the basis of spectral theory. Publishing House "Neftyanoe Khozyaystvo" (Oil Ind.) **12**, 48–51 (2015). (in Russian)
2. Bazaikin, Y.V., Baikov, V.A., Taimanov, I.A., Yakovlev, A.A.: Numerical analysis of topological characteristics of three-dimensional geological models of oil and gas fields. *Math. Model.* **25**(10), 19–31 (2013). (in Russian)
3. Baikov, V.A., Gilmanov, R.R., Taimanov, I.A., Yakovlev, A.A.: Topological characteristics of oil and gas reservoirs and their applications. In: Holzinger, A., Goebel, R., Ferri, M., Palade, V. (eds.) *Towards Integrative Machine Learning and Knowledge Extraction*. LNCS (LNAI), vol. 10344, pp. 182–193. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-69775-8\\_11](https://doi.org/10.1007/978-3-319-69775-8_11)
4. Krivoshchekov, S.N., Kochnev, A.A.: Determination of reservoir properties of reservoir rocks using X-ray imaging core. *Master's J.* **1**, 120–128 (2014). (Russian)
5. Wellington, S.L., Vinegar, H.J.: X-ray computerized tomography. *J. Petrol. Technol.* **39**, 885–898 (1987)
6. Yakushina, O.A., Ozhogina, E.G., Hozyainov, M.S.: X-ray computational microtomography – non-destructive method of structural and phase analysis. *Comput. World (Mir izmereniy)* **10**, 12–17 (2003). (in Russian)
7. Ivanov, M.K., Burlin, U.K., Kalmykov, G.A., Karniushina, E.E., Korobova, N.I.: *Petrophysical Methods of Core Material Research*. MSU Publishing, Moscow (2008)
8. Zhukovskaia, E.A., Lopushniak, U.M.: The use of X-ray tomography in the study of terrigenous and carbonate reservoirs. *Nauchno-technicheskiy vestnik OAO "NK Rosneft"* **1**, 1–25 (2008). (in Russian)
9. De Vos, W., Casselman, J., Swennen, G.R.J.: Cone-beam computerized tomography (CBCT) imaging of the oral and maxillofacial region: a systematic re-view of the literature. *Int. J. Oral Maxillofac. Surg.* **38**(6), 609–625 (2009)
10. Baikov, V.A., Bakirov, N.K., Yakovlev, A.A.: *Mathematical geology. I. Introduction to geostatistics*. Izhevsk Institute of Computer Sciences, Izhevsk (2012). (in Russian)



# Shortened Persistent Homology for a Biomedical Retrieval System with Relevance Feedback

Alessia Angeli<sup>1</sup>, Massimo Ferri<sup>2(✉)</sup>, Eleonora Monti<sup>1</sup>, and Ivan Tomba<sup>3</sup>

<sup>1</sup> Department of Mathematics, University Bologna, Bologna, Italy  
{alessia.angeli,eleonora.monti5}@studio.unibo.it

<sup>2</sup> Department of Mathematics and ARCES, University Bologna, Bologna, Italy  
massimo.ferri@unibo.it

<sup>3</sup> 2R&D Department, CA-MI S.r.l., Via Ugo La Malfa 13, Pilastro di Langhirano, PR, Italy  
tomba.ivan@gmail.com

**Abstract.** This is the report of a preliminary study, in which a new coding of persistence diagrams and two relevance feedback methods, designed for use with persistent homology, are combined. The coding consists in substituting persistence diagrams with complex polynomials; these are “shortened”, in the sense that only the first few coefficients are used. The relevance feedback methods play on the user’s feedback for changing the impact of the different filtering functions in determining the output.

**Keywords:** Persistence diagram · Elementary symmetric function  
Projected gradient

## 1 Introduction

The interaction between a medical doctor and a smart machine must respect at least two requirements: Fast action and good integration with the human operator. As far as mere morphology is concerned, deep learning has already reached performances comparable with the ones of a dermatologist [6]. In real practice, the number of (hidden or evident, formal or intuitive) parameters in a diagnostic task is so high, that a good synthesis in short times is, at least for the moment, the field of a human expert; the machine can anyway offer a reliable, stable, powerful assistance. It is necessary that the medical doctor understands what’s going on in his/her interaction with the machine: This is a primary issue in the design of “explainable” AI systems [15]. A way out of the “black box” frustration is to accept a feedback from the user, so that the system adapts more and more to his/her viewpoint.

---

Article written within the activity of INdAM-GNSAGA.

This is the case of a system currently developed by Ca-Mi srl, an Italian company producing biomedical devices, in collaboration with the Universities of Bologna and Parma and with the Romagna Institute for Study and Cure of Tumors (IRST). The machine acquires the image of a dermatological lesion and retrieves a set of most similar images out of a database with sure diagnoses. The similarity is assessed by a relatively recent geometric-topological technique: Persistent homology. This tool is very effective above all on data of natural origin [8], but the main tool for classification and (dis)similarity assessment—the bottleneck distance between persistence diagrams—is computationally heavy. We are then experimenting a different coding of the same information contained in a persistence diagram, through a complex polynomial; the coding is then “shortened” in that we just use the first few coefficients, so that comparison and search becomes much faster; it appears that these first coefficients contain most of the relevant information.

While the first experiments are very promising [10], there is still a wide gap between what the system and the doctor see as “similar”. Therefore it is an active area of research, to study a relevance feedback method for drawing the machine’s formalization of similarity near the doctor’s skilled view.

In this paper we present a preliminary study on a small public database ( $PH^2$ ) of nevi and melanomas; our goal is to combine two relevance feedback methods expressly designed for persistent homology, with a new coding of one of its main tools, persistence diagrams.

Content Based Image retrieval (CBIR) is a ripe and challenging research area [11]. Apart from annotation-based systems, much of the success of CBIR is tied with the use of histograms (of colours, directions, etc.). Topological descriptors have entered the game but are not yet fully employed [19]. On the other hand, we are aware of the incredible flourishing of Deep Learning in the areas of image understanding and of medical diagnosis; our interest in the techniques proposed here depends on the needs to work with a rather limited database, and to be able to control step-by-step how the system adapts to the user. Still we plan, as a future step, to combine persistent homology with Deep Learning—as several researchers already do—by feeding a neural net with persistence diagrams. There are at least two reasons for wishing such a development: The importance of the user’s viewpoint (or taste, or goal) formalized by the choice of filtering functions; and the fact that whatever the kind of input, persistence diagrams have always the same structure, so that a learning network trained on persistence diagrams becomes immediately a much more versatile tool. A step in this direction has already been done [14].

## 2 Persistent Homology

Persistent homology is a branch of computational topology, of remarkable success in shape analysis and pattern recognition. Its key idea is to analyze data through *filtering functions*, i.e. continuous functions  $f$  defined on a suitable topological space  $X$  with values e.g. in  $\mathbb{R}$  (but sometimes in  $\mathbb{R}^n$  or in a circle). Given a pair

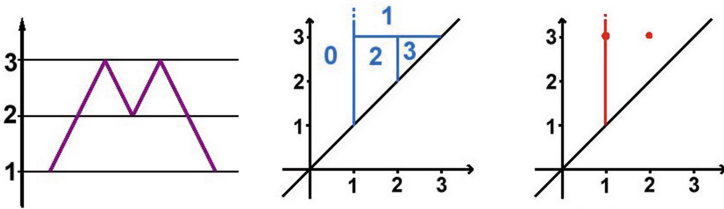
$(X, f)$ , with  $f : X \rightarrow \mathbb{R}$  continuous, for each  $u \in \mathbb{R}$  the *sublevel set*  $X_u$  is the set of elements of  $X$  whose value through  $f$  is less than or equal to  $u$ .

For each  $X_u$  one can compute the *homology modules*  $H_r(X_u)$ , vector spaces which summarize the presence of “voids” of dimension  $r$ , with  $r \in \mathbb{N}$  (connected components in the case  $r = 0$ ) and their relations. As there exist various homology theories, some additional hypotheses might be requested on  $f$  depending on the choice of the homology.

Of course, if  $u < v$  then  $X_u \subseteq X_v$ . There corresponds a linear map  $\iota_{(u,v)}^r : H_r(X_u) \rightarrow H_r(X_v)$ . On  $\Delta^+ = \{(u, v) \in \mathbb{R}^2 \mid u < v\}$  we can then define the *r-Persistent Betti Number* (*r-PBN*) function

$$\beta_{(X,f)}^r : \Delta^+ \rightarrow \mathbb{Z} \\ (u, v) \mapsto \dim \text{Im}(\iota_{(u,v)}^r)$$

All information carried by *r-PBN*'s is condensed in some points (dubbed *proper cornerpoints*) and some half-lines (*cornerlines*); cornerlines are actually thought of as *cornerpoints at infinity*. Cornerpoints (proper and at infinity) build what is called the *persistence diagram* relative to dimension  $r$ . Figure 1 shows a letter “M” as space  $X$ , ordinate as function  $f$  on the left, its 0-PBN function at the center and the corresponding persistence diagram on the right.



**Fig. 1.** Letter M, its 0-PBN function and the corresponding persistence diagram, relative to filtering function ordinate.

*Remark 1.* The theory also contemplates a *multiplicity* for cornerpoints (proper and at infinity); multiplicity higher than one is generally due to symmetries. We don’t care about it in the present research, since all cornerpoints in our experiments have multiplicity one, as usual in diagrams coming from natural images.

Classification and retrieval of persistence diagrams (and consequently of the object they represent) is usually performed by the following distance, where persistence diagrams are completed by all points on the “diagonal”  $\Delta = \{(u, v) \in \mathbb{R} \mid u = v\}$ .

**Definition 1.** Bottleneck (or matching) distance.

Let  $\mathcal{D}_k$  and  $\mathcal{D}'_k$  be two persistence diagrams with a finite number of cornerpoints, the bottleneck distance  $d_B(\mathcal{D}_k, \mathcal{D}'_k)$  is defined as

$$d_B(\mathcal{D}_k, \mathcal{D}'_k) = \min_{\sigma} \max_{P \in \mathcal{D}_k} \hat{d}(P, \sigma(P))$$

where  $\sigma$  varies among all the bijections between  $\mathcal{D}_k$  and  $\mathcal{D}'_k$  and

$$\hat{d}((u, v), (u', v')) = \min \left\{ \max \{|u - u'|, |v - v'|\}, \max \left\{ \frac{v - u}{2}, \frac{v' - u'}{2} \right\} \right\}$$

given  $(u, v) \in \mathcal{D}_k$  and  $(u', v') \in \mathcal{D}'_k$ .

For homology theory one can consult any text on algebraic topology, e.g. [13]. For persistent homology, two good references are [4, 5].

### 3 Symmetric Functions of Warped Persistence Diagrams

There are two main difficulties in comparing persistence diagrams. One is the fact that the diagonal  $\Delta$  has a special role: For a persistence diagram, the diagonal  $\Delta$  is a sort of “blown up” point, in the sense that points close to it are seen as close to each other by the bottleneck distance; moreover cornerpoints close to  $\Delta$  generally represent noise, and it would be desirable to diminish their contribution. A second difficulty consists in the coding of a set of points in the plane, in a form suited for comparison and processing: E.g. an intuitive coding like making a  $2M$  vector out of a set of  $M$  points is highly unstable and is not the solution to the problem of avoiding the permutations required by the bottleneck distance; a distance computed component by component might be very far from the optimal one. In [1]—which also contains a comparison with the bottleneck distance—we have tried to overcome both difficulties.

Following [3, 9], we have faced the first problem by two different transformations  $T$  (designed by Barbara Di Fabio) and  $R$  which “warp” the plane, so that all  $\Delta$  is sent to  $(0, 0)$ , seen here as the complex number zero (Fig. 2):

$$T : \bar{\Delta}^+ \rightarrow \mathbb{C}, \quad T(u, v) = \frac{v-u}{2}(\cos(\alpha) - \sin(\alpha) + i(\cos(\alpha) + \sin(\alpha)))$$

where  $\alpha = \sqrt{u^2 + v^2}$ .

$$R : \bar{\Delta}^+ \rightarrow \mathbb{C}, \quad R(u, v) = \frac{v-u}{\sqrt{2}}(\cos(\theta) + i \sin(\theta))$$

where  $\theta = \pi(u + v)$ .

The main ideas behind  $T$  and  $R$  are: For the reasons mentioned at the beginning of this section, cornerpoints close to  $\Delta$  ought to be considered close to each other. Therefore  $T$  and  $R$  (and other maps under study) take  $\Delta$  to zero. Moreover they wrap the strip adjacent to  $\Delta$  around zero itself, so that the contributions of noise cornerpoints should balance away in Viète’s symmetric functions. We

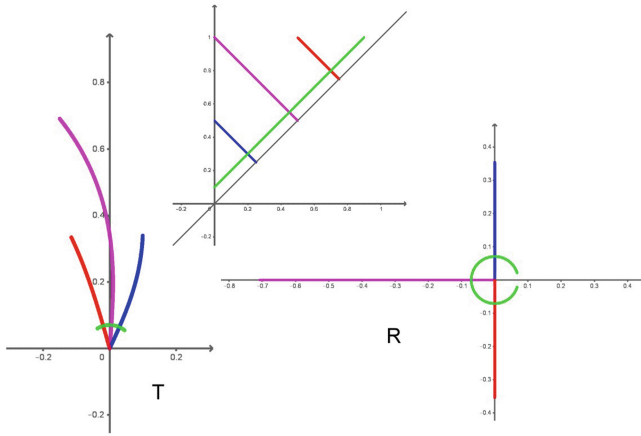


Fig. 2. The action of transformations  $T$  (left) and  $R$  (right) on some segments (above).

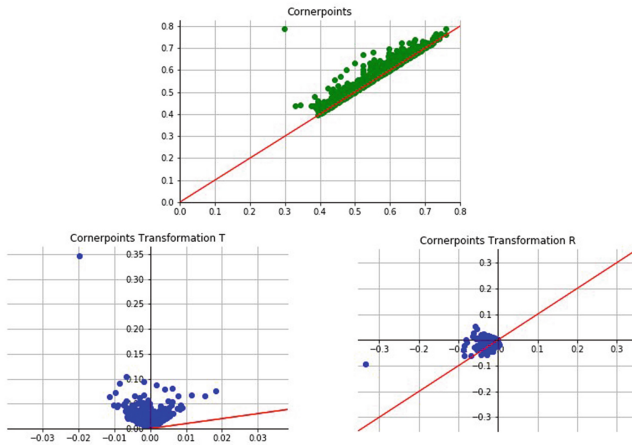


Fig. 3. A persistence diagram (above) and its images through  $T$  (left) and  $R$ .

are still looking for the best wrapping function. Both  $T$  and  $R$  are continuous maps. See Fig. 3 for a persistent diagram and its two images through  $T$  and  $R$ .

As for the second problem, following an idea of Claudia Landi [3, 9] we decided to encode each (transformed) persistence diagram  $\mathcal{D}$  as the polynomial having the complex numbers, images of the cornerpoints, as roots. We can then design distances built on the pairs of coefficients of equal position in the polynomials representing two diagrams, avoiding a combinatorial explosion. Actually, for a given diagram we form the vector having as components the elementary symmetric functions of the transformed cornerpoints (which, through Viète's formulas, equal the coefficients of the polynomial up to the sign) [16, Sect. IV.8]. So, the first component of the vector is the sum of all those numbers, the second one is the sum of all pairwise products, and so on. In order to take also cornerpoints at infinity into account, we have performed the following substitution. Given a cornerpoint at infinity (i.e. a cornerline) with abscissa  $w$  of a persistence diagram  $\mathcal{D}$ , we substitute it with the point

$$(w, \max\{v \mid (u, v) \text{ is a proper cornerpoint of } \mathcal{D}\})$$

*Remark 2.* Since cornerpoints near  $\Delta$  generally represent noise, the two transformations were designed to wrap them around zero, so that the symmetric functions be scarcely affected by them. To this goal, transformation  $R$  fits better in our case, since our filtering functions (hence the cornerpoint coordinates) are bounded between 0 and 1.

For a fixed filtering function, for a fixed transformation ( $T$  or  $R$ ), the same elementary symmetric function computed on two persistence diagrams may show a difference of several orders of magnitude, when there are many cornerpoints. This might consistently alter the comparisons, so we actually formed, for each persistence diagram  $\mathcal{D}$ , the complex vector  $a_{\mathcal{D}}$  whose  $i$ -th component  $a_{\mathcal{D}}(i)$  is the  $i$ -th root of the  $i$ -th elementary symmetric function of the transformed cornerpoints, divided by the number of cornerpoints. As an example, these are the (approximated) real parts of the first 10 symmetric functions for images IMD251 and IMD423, filtering function "Light intensity 2", transformation  $T$ :

IMD251:

(2E-1, -5.9E-1, 8.3E-2, 6.8E-2, -3.4E-2, 7.6E-3, -1E-3, 8.4E-5, -1E-6, -8.5E-7)

IMD423:

(7.1, -1.9E+2, -1.4E+3, 2.3E+3, 4.1E+4, 6E+4, -3.5E+5, -1.1E+6, 6.3E+5, 7.2E+6)

which, by taking the  $i$ -th root of the  $i$ th symmetric function, become:

IMD251:

(2E-1, -7.7E-1, 4.4E-1, 5.1E-1, -5.1E-1, 4.4E-1, -3.7E-1, 3.1E-1, -2.2E-1, -2.6E-1)

IMD423:

(7.1, -1.4E+1, -1.1E+1, 6.9, 8.4, 6.2, -6.2, -5.7, 4.4, 4.9)

and division by the number of cornerpoints (here 257 and 1408 respectively) yields:



IMD251:

(7.6E-4, -3E-3, 1.6E-3, 2E-3, -2E-3, 1.7E-3, -1.4E-3, 1.2E-3, -8.3E-4, -9.6E-4)

IMD423:

(5.1E-3, -9.7E-3, -8E-3, 4.9E-3, 5.9E-3, 4.4E-3, -4.4E-3, -4.1E-3, 3.1E-3, 3.4E-3)

whose differences finally make sense.

The advantage of using the vectors  $a_{\mathcal{D}}$  instead of the original diagrams is that one can directly design a distance between complex vectors component-by-component (e.g. the  $L^1$  distance we used), instead of considering all bijections between cornerpoint sets. Still, the computation of all the elementary symmetric functions would be too time consuming, so we performed in [1] some experiments by reducing the computation to the first  $k$  components,  $k \in \{5, 10, 20, 50\}$ , a small number compared with the hundreds of cornerpoints commonly found in the persistence diagrams of the examined images. Classification and retrieval of dermatological images using such “shortened” vectors was quite satisfactory with a dramatic time reduction.

## 4 Modifying Distances

One of the major advantages of persistent homology is its modularity: By changing filtering function we change point of view on the shape of the objects under study and on the comparison criteria. Therefore the same data can be transformed into several pairs  $(X, f)$ , and for each pair we obtain a distance reflecting the features of  $X$  captured by filtering function  $f$ . Both for classification and for retrieval, we need a single distance, so we have to blend the distances we have into one. There are two rather natural choices for that: either the maximum or the arithmetic average.

Maximum is the initial choice of [12]. An ongoing research, by some of the authors of the present paper, prefers the average instead [17]; this agrees with the idea of cooperation of the different filtering functions (like in [2, 7]), versus one of them prevailing. We make the “neutral” choice of equal weights in the starting average for initializing the subsequent optimization process. As hinted in the Introduction, the research is aimed at enhancing a device of acquisition and retrieval of dermatological images. The concept of “similarity” is (and has to remain!) highly subjective in the medical domain: We want to adapt the system to the physician, not the other way around. This can be done by modifying the weights of the different distances when building a single distance, to approximate the (pseudo)distance  $\delta$  representing the dissimilarity as perceived by the user.

Our setting is: We have a set  $X = \{x_1, \dots, x_N\}$  of objects (in our case dermatological images) and  $J$  descriptors  $d^{(1)}, \dots, d^{(J)}$  which give rise to an initial distance  $D^{IN}$  between the objects in  $X$ . In our study,  $D^{IN}$  is one of these two:

$$D^{MAX} = \max\{d^{(1)}, \dots, d^{(J)}\}, \quad D^{AVG} = \frac{d^{(1)} + \dots + d^{(J)}}{J}$$

Given a query  $q$  (i.e. an acquired image), the system retrieves the  $L$  objects closest to  $q$  with respect to  $D^{IN}$ , i.e. an  $L$ -tuple  $X_q = (x_{i_1}, \dots, x_{i_L})$  such that  $D^{IN}(q, x_{i_1}) \leq \dots \leq D^{IN}(q, x_{i_L})$ . The user is shown these objects and expresses his/her relevance feedback by assessing the perceived dissimilarities as numbers  $\delta(q, x_{i_1}), \dots, \delta(q, x_{i_L})$ .

While the Multilevel Relevance Feedback (MLRF) method proposed in [12] starts from  $D^{MAX}$ , then rescales the distances  $d^{(i)}$  and takes the maximum, our Least Squares Relevance Feedback (LSRF) scheme start from  $D^{AVG}$  and computes a new distance  $D^{OUT}$  as

$$D^{OUT} = \sum_{j=1}^J \lambda_j d^{(j)}, \quad \lambda_j \geq 0$$

by minimizing the objective function

$$g(\lambda) = \|\mathbf{d}\lambda - \delta\|_2^2$$

i.e. by looking for  $\lambda = \operatorname{argmin} \|\mathbf{d}\lambda - \delta\|_2^2$ , where the  $t$ -th row of matrix  $\mathbf{d}$  is formed by the distances  $d^{(j)}(q, x_{i_t})$ ,  $\lambda$  is the column matrix of the  $\lambda_j$  and  $\delta$  is the column matrix formed by  $\delta(q, x_{i_t})$  for  $t = 1, \dots, L$  and  $j = 1, \dots, J$ .

Since this minimization problem might have multiple solutions, the vector of weights  $\lambda_j$  in  $D^{OUT}$  is obtained by iterating the Projected Gradient method.  $D^{OUT}$  is the best possible distance approximating the user's similarity distance  $\delta$  from the given data in a least-square sense. More details on this procedure will be given in an article to come.

## 5 Experimental Results

As a preliminary study, we have tried to combine the modularity of persistent homology with the fast computation of the short vectors of symmetric functions of the transformed cornerpoints, with the adapting weights of relevance feedback.

We experimented with a small public database,  $PH^2$  [18], containing 8-bit RGB,  $768 \times 560$  pixels images of 80 common nevi, 80 atypical nevi, and 40 melanomas. We used 19 distances: 8 coming from simple morphological parameters, 11 coming from as many filtering functions; see Table 1 for a description of these features [1, 10]. It should be mentioned that the 11 distances coming from persistent homology already yield very good results, but the simple (and very fast computable) ones, obtained from morphological parameters, refine the general performance. For each image, we built the 11 persistence diagrams, performed one of the transformations  $T$  and  $R$  (see Sect. 3), and computed the 11 corresponding vectors, limited to length  $k = 20$ .

In normal operation, the user will give his/her feedback as follows. For  $j = 1, \dots, L$ , the user is asked to assign a similarity judgement (where 0 stands for "dissimilar" and 1 for "similar") to the pair  $(q, x_{i_j})$  with respect to three different aspects of the skin lesion (boundary/shape, colours, texture). This results in a

**Table 1.** Features.

Persistence features	Morphological features
Light intensity	Colour histogram
Blue	Form factor
Green	Haralick's circularity
Red	Asymmetry
Excess blue	Ellipticity
Excess green	Eccentricity
Excess red	Diameter
Light intensity 2	Colour entropy
Boundary light intensity	
Boundary	
Boundary 2	

similarity vote  $v_{i_j}$  in the range  $\{0, 1, 2, 3\}$  which is transformed into  $\delta(q, x_{i_j})$  by the following formula:

$$\delta(q, x_{i_j}) = \max \left\{ 0, D^{IN}(q, x_{i_1}) + \frac{4(3-v_{i_j})-1}{10} (D^{IN}(q, x_{i_1}) - D^{IN}(q, x_{i_L})) \right\}$$

The formula assigns the values of  $\delta$  in such a way that if the similarity vote  $v_{i_j}$  is maximal (3/3), then the corresponding  $\delta$  is lower than the lowest value of the distances of the database images from the query and, conversely, if  $v_{i_j}$  is minimal, then  $\delta$  is higher than the distance of the image which is farthest from it.

We finally retrieve again the  $L$  closest objects according to the modified distance (see Sect. 4).

**Table 2.** Sums of scores with the two considered methods of relevance feedback.

	With $D^{MAX}$	MLRF	Difference	With $D^{AVG}$	LSRF	Difference
Transf. $T$	1694	1776	82	1729	1856	127
Transf. $R$	1869	1762	73	1735	1857	122

In the present research  $L = 10$  and the retrieval is performed by the “leave one out” scheme. Not having a real relevance feedback by a physician, we assign a retrieved image the maximal similarity vote  $v = 3$  if it shares the same histological diagnosis of the query, and the minimal similarity vote  $v = 0$  otherwise.

Assessment of the retrieval is performed by counting how many, of the retrieved lesions, have the same diagnosis of the query before and after the feedback. This is summarized in Table 2 by summing these scores for all images of the database as queries, starting from  $D^{MAX}$  and then applying the MLRF scheme, starting from  $D^{AVG}$  and applying our LSRF method.

As we can see, transformations  $R$  and  $T$  yield similar results.  $D^{AVG}$  performs better than  $D^{MAX}$  and LSRF produces a better improvement than MLRF.

## 6 Conclusions

We have compared—on a small public database of nevi and melanomas—two relevance feedback methods, conceived for a retrieval system based on persistent homology, combined with a computation reduction based on elementary symmetric functions of warped persistence diagrams. A weighted average of distances, with weights obtained by a standard optimization method, seems to perform better. The experiment will be extended to larger databases and with real feedback from expert dermatologists.

**Acknowledgment.** We wish to thank all the Reviewers for the very detailed and useful comments.

## References


1. Angeli, A., Ferri, M., Tomba, I.: Symmetric functions for fast image retrieval with persistent homology (2018, preprint)
2. Brucale, A., et al.: Image retrieval through abstract shape indication. In: Proceedings of the IAPR Workshop MVA 2000, Tokyo, 28–30 November, pp. 367–370 (2000)
3. Di Fabio, B., Ferri, M.: Comparing persistence diagrams through complex vectors. In: Murino, V., Puppo, E. (eds.) ICIAP 2015. LNCS, vol. 9279, pp. 294–305. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-23231-7\\_27](https://doi.org/10.1007/978-3-319-23231-7_27)
4. Edelsbrunner, H., Harer, J.: Persistent homology—a survey. In: Surveys on Discrete and Computational Geometry, Contemporary Mathematics, vol. 453, pp. 257–282. American Mathematical Society, Providence (2008)
5. Edelsbrunner, H., Harer, J.: Computational Topology: An Introduction. American Mathematical Society, Providence (2009)
6. Esteva, A., et al.: Dermatologist-level classification of skin cancer with deep neural networks. *Nature* **542**(7639), 115 (2017)
7. Ferri, M.: Graphic-based concept retrieval. In: Cuzzocrea, A., Kittl, C., Simos, D.E., Weippl, E., Xu, L. (eds.) CD-ARES 2013. LNCS, vol. 8127, pp. 460–468. Springer, Heidelberg (2013). [https://doi.org/10.1007/978-3-642-40511-2\\_33](https://doi.org/10.1007/978-3-642-40511-2_33)
8. Ferri, M.: Persistent topology for natural data analysis — a survey. In: Holzinger, A., Goebel, R., Ferri, M., Palade, V. (eds.) Towards Integrative Machine Learning and Knowledge Extraction. LNCS (LNAI), vol. 10344, pp. 117–133. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-69775-8\\_6](https://doi.org/10.1007/978-3-319-69775-8_6)
9. Ferri, M., Landi, C.: Representing size functions by complex polynomials. *Proc. Math. Met. in Pattern Recogn.* **9**, 16–19 (1999)
10. Ferri, M., Tomba, I., Visotti, A., Stanganelli, I.: A feasibility study for a persistent homology-based k-nearest neighbor search algorithm in melanoma detection. *J. Math. Imaging Vis.* **57**(3), 324–339 (2017)

11. Ghosh, N., Agrawal, S., Motwani, M.: A survey of feature extraction for content-based image retrieval system. In: Tiwari, B., Tiwari, V., Das, K.C., Mishra, D.K., Bansal, J.C. (eds.) *Proceedings of International Conference on Recent Advancement on Computer and Communication*. LNNS, vol. 34, pp. 305–313. Springer, Singapore (2018). [https://doi.org/10.1007/978-981-10-8198-9\\_32](https://doi.org/10.1007/978-981-10-8198-9_32)
12. Giorgi, D., Frosini, P., Spagnuolo, M., Falcidieno, B.: 3D relevance feedback via multilevel relevance judgements. *The Vis. Comput.* **26**(10), 1321–1338 (2010)
13. Hatcher, A.: *Algebraic Topology*, vol. 606 (2002)
14. Hofer, C., Kwitt, R., Niethammer, M., Uhl, A.: Deep learning with topological signatures. In: *Advances in Neural Information Processing Systems*, pp. 1633–1643 (2017)
15. Holzinger, A., Biemann, C., Pattichis, C.S., Kell, D.B.: What do we need to build explainable AI systems for the medical domain? arXiv preprint [arXiv:1712.09923](https://arxiv.org/abs/1712.09923) (2017)
16. Lang, S.: *Undergraduate Algebra*. Springer, New York (2005). <https://doi.org/10.1007/0-387-27475-8>
17. Magi, S., et al.: Relevance Feedback in a Dermatology Application (2018, submitted)
18. Mendonça, T., Ferreira, P.M., Marques, J.S., Marcal, A.R., Rozeira, J.: Ph 2-a dermoscopic image database for research and benchmarking. In: 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 5437–5440. IEEE (2013)
19. Zeppelzauer, M., Zieliński, B., Juda, M., Seidl, M.: A study on topological descriptors for the analysis of 3D surface texture. *Comput. Vis. Image Underst.* (2017)

# **MAKE Explainable AI**



# Explainable AI: The New 42?

Randy Goebel<sup>1</sup>(✉), Ajay Chander<sup>2</sup>, Katharina Holzinger<sup>3</sup>, Freddy Lecue<sup>4,5</sup>,  
Zeynep Akata<sup>6,7</sup>, Simone Stumpf<sup>8</sup>, Peter Kieseberg<sup>3,9</sup>,  
and Andreas Holzinger<sup>10,11</sup> 

- <sup>1</sup> Alberta Machine Intelligence Institute, University of Alberta, Edmonton, Canada  
rgoebel@ualberta.ca
- <sup>2</sup> Fujitsu Labs of America, Sunnyvale, USA  
achander@us.fujitsu.com
- <sup>3</sup> SBA-Research, Vienna, Austria  
kholzinger@sba-research.org
- <sup>4</sup> INRIA, Sophia Antipolis, France  
freddy.lecue@inria.fr
- <sup>5</sup> Accenture Labs, Dublin, Ireland
- <sup>6</sup> Amsterdam Machine Learning Lab,  
University of Amsterdam, Amsterdam, The Netherlands
- <sup>7</sup> Max Planck Institute for Informatics, Saarbruecken, Germany  
z.akata@uva.nl
- <sup>8</sup> City, University of London, London, UK  
Simone.Stumpf.1@city.ac.uk
- <sup>9</sup> University of Applied Sciences St. Pölten, St. Pölten, Austria  
Peter.Kieseberg@fhstp.ac.at
- <sup>10</sup> Holzinger Group HCI-KDD, Institute for Medical Informatics, Statistics and  
Documentation, Medical University Graz, Graz, Austria
- <sup>11</sup> Institute of Interactive Systems and Data Science,  
Graz University of Technology, Graz, Austria  
a.holzinger@hci-kdd.org

**Abstract.** Explainable AI is not a new field. Since at least the early exploitation of C.S. Pierce's abductive reasoning in expert systems of the 1980s, there were reasoning architectures to support an explanation function for complex AI systems, including applications in medical diagnosis, complex multi-component design, and reasoning about the real world. So explainability is at least as old as early AI, and a natural consequence of the design of AI systems. While early expert systems consisted of handcrafted knowledge bases that enabled reasoning over narrowly well-defined domains (e.g., INTERNIST, MYCIN), such systems had no learning capabilities and had only primitive uncertainty handling. But the evolution of formal reasoning architectures to incorporate principled probabilistic reasoning helped address the capture and use of uncertain knowledge.

There has been recent and relatively rapid success of AI/machine learning solutions arises from neural network architectures. A new generation of neural methods now scale to exploit the practical applicability

of statistical and algebraic learning approaches in arbitrarily high dimensional spaces. But despite their huge successes, largely in problems which can be cast as classification problems, their effectiveness is still limited by their un-debuggability, and their inability to “explain” their decisions in a human understandable and reconstructable way. So while AlphaGo or DeepStack can crush the best humans at Go or Poker, neither program has any internal model of its task; its representations defy interpretation by humans, there is no mechanism to explain their actions and behaviour, and furthermore, there is no obvious instructional value . . . the high performance systems can not help humans improve.

Even when we understand the underlying mathematical scaffolding of current machine learning architectures, it is often impossible to get insight into the internal working of the models; we need explicit modeling and reasoning tools to explain how and why a result was achieved. We also know that a significant challenge for future AI is contextual adaptation, i.e., systems that incrementally help to construct explanatory models for solving real-world problems. Here it would be beneficial not to exclude human expertise, but to augment human intelligence with artificial intelligence.

**Keywords:** Artificial intelligence · Machine learning · Explainability  
Explainable AI

## 1 Introduction

Artificial intelligence (AI) and machine learning (ML) have recently been highly successful in many practical applications (e.g., speech recognition, face recognition, autonomous driving, recommender systems, image classification, natural language processing, automated diagnosis, . . . ), particularly when components of those practical problems can be articulated as data classification problems. Deep learning approaches, including the more sophisticated reinforcement learning architectures, exceed human performance in many areas [6, 17, 18, 24].

However, an enormous problem is that deep learning methods turn out to be uninterpretable “black boxes,” which create serious challenges, including that of interpreting a predictive result when it may be confirmed as incorrect. For example, consider Fig. 1, which presents an example from the Nature review by LeCun, Bengio, and Hinton [15]. The figure incorrectly labels an image of a dog lying on a floor and half hidden under a bed as “A dog sitting on a hardwood floor.” To be sure, the coverage of their image classification/prediction model is impressive, as is the learned coupling of language labels. But the reality is that the dog is *not* sitting.

The first problem is the naive but popular remedy about how to debug the predictive classifier to correct the error: augment the original labeled training set with more carefully crafted inputs to distinguish, say, a sitting from a laying dog might improve the incorrect output. This may or may not correct the problem, and doesn’t address the resource challenge of recreating the original learned model.





A **dog** is standing on a hardwood floor.

**Fig. 1.** Segment of an example from LeCun, Bengio, Hinton, Science [15]

The transparency challenge gets much more complex when the output predictions are not obviously wrong. Consider medical or legal reasoning, where one typically seeks not just an answer or output (e.g., a diagnostic prediction of prostate cancer would require some kind of explanation or structuring of evidence used to support such a prediction). In short, false positives can be disastrous.

Briefly, the representational and computational challenge is about *how* to construct more explicit models of what is learned, in order to support explicit computation that produces a model-based explanation of a predicted output.

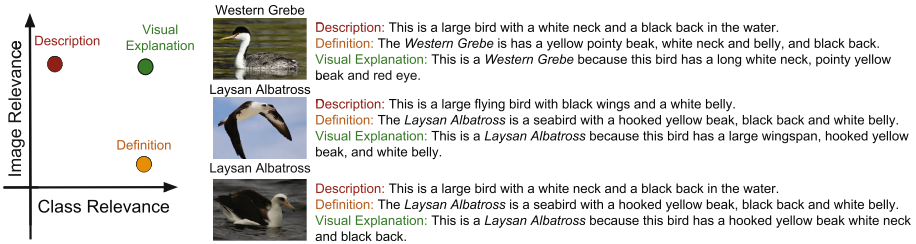
However, this is one of the historical challenges of AI: what are appropriate representations of knowledge that demonstrate some veracity with the domain being captured? What reasoning mechanisms offer the basis for conveying a computed inference in terms of that model?

The reality of practical applications of AI and ML in sensitive areas (such as the medical domain) reveals an inability of deep learned systems to communicate effectively with their users. So emerges the urgent need to make results and machine decisions transparent, understandable and explainable [9–11]. The big advantage of such systems would include not only explainability, but deeper understanding and replicability [8]. Most of all, this would increase acceptance and trust, which is mandatory in safety-critical systems [12], and desirable in many applications (e.g., in medical robotics [19], Ambient Assisted Living [23], Enterprise decision making [4], etc.). First steps have been taken towards making these systems understandable to their users, by providing textual and visual explanations [13,22] (see Figs. 2 and 3).

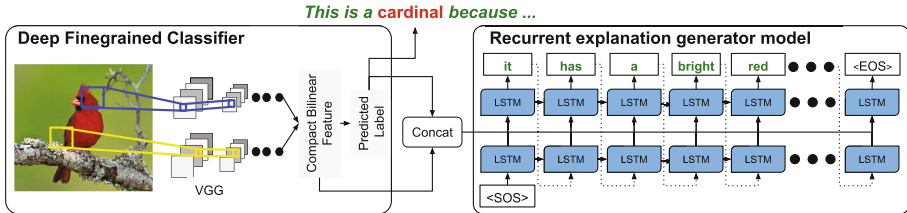
## 2 Current State-of-the-Art

Explaining decisions is an integral part of human communication, understanding, and learning, and humans naturally provide both deictic (pointing) and textual modalities in a typical explanation. The challenge is to build deep learning models that are also able to explain their decisions with similar fluency in both visual and textual modalities (see Fig. 2). Previous machine learning methods for explanation were able to provide a text-only explanation conditioned on

an image in context of a task, or were able to visualize active intermediate units in a deep network performing a task, but were unable to provide explanatory text grounded in an image.



**Fig. 2.** The goal is to generate *explanations* that are both image relevant and class relevant. In contrast, *descriptions* are image relevant, but not necessarily class relevant, and *definitions* are class relevant but not necessarily image relevant.

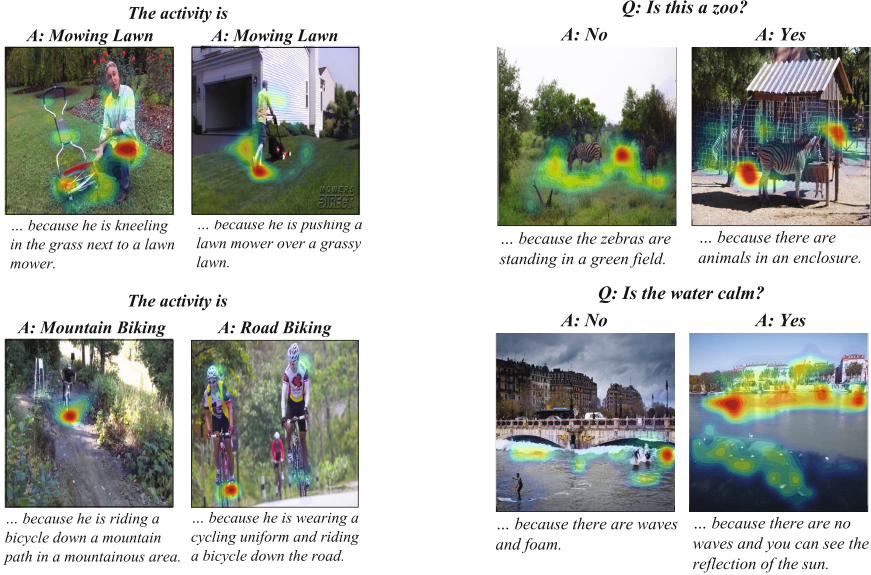


**Fig. 3.** A joint classification and explanation model [7]. Visual features are extracted using a fine-grained classifier before sentence generation; unlike other sentence generation models, condition sentence generation on the predicted class label. A discriminative loss function encourages generated sentences to include class specific attributes.

Existing approaches for deep visual recognition are generally opaque and do not output any justification text; contemporary vision-language models can describe image content but fail to take into account class-discriminative image aspects which justify visual predictions.

Hendriks et al. [7] propose a new model (see Fig. 3) that focuses on the discriminating properties of the visible object, jointly predicts a class label, and explains why the predicted label is appropriate for the image. The idea relies on a loss function based on sampling and reinforcement learning, which learns to generate sentences that realize a global sentence property, such as class specificity. This produces a fine-grained bird species classification dataset, and shows that an ability to generate explanations which are not only consistent with an image but also more discriminative than descriptions produced by existing captioning methods.

Although, deep models that are both effective and explainable are desirable in many settings, prior explainable models have been unimodal, offering either



**Fig. 4.** Left: ACT-X qualitative results: For each image the PJ-X model provides an answer and a justification, and points to the evidence for that justification. Right: VQA-X qualitative results: For each image the PJ-X model provides an answer and a justification, and points to the evidence for that justification.

image-based visualization of attention weights or text-based generation of post-hoc justifications. Park et al. [21] propose a multimodal approach to explanation, and argue that the two modalities provide complementary explanatory strengths.

Two new datasets are created to define and evaluate this task, and use a model which can provide joint textual rationale generation and attention visualization (see Fig. 4). These datasets define visual and textual justifications of a classification decision for activity recognition tasks (ACT-X) and for visual question answering tasks (VQA-X). They quantitatively show that training with the textual explanations not only yields better textual justification models, but also better localizes the evidence that supports the decision.

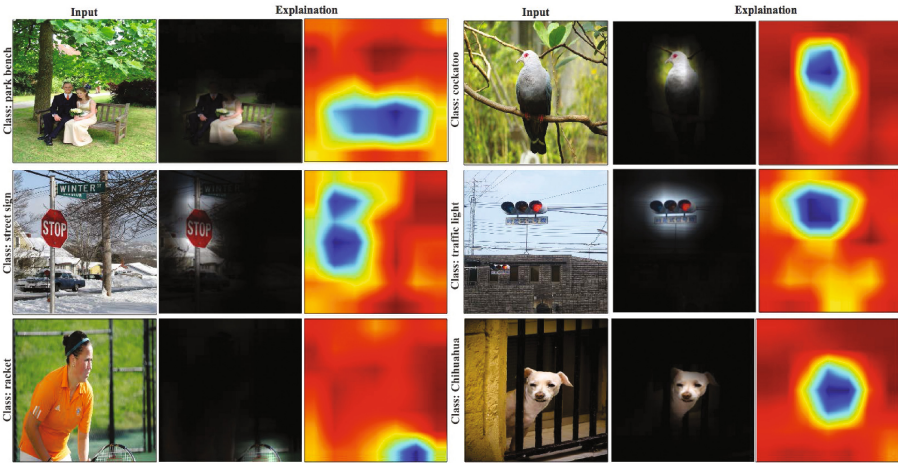
Qualitative cases also show both where visual explanation is more insightful than textual explanation, and vice versa, supporting the hypothesis that multimodal explanation models offer significant benefits over unimodal approaches. This model identifies visual evidence important for understanding each human activity. For example to classify “mowing lawn” in the top row of Fig. 4 the model focuses both on the person, who is on the grass, as well as the lawn mower. This model can also differentiate between similar activities based on the context, e.g. “mountain biking” or “road biking.”

Similarly, when asked “Is this a zoo?” the explanation model is able to discuss what the concept of “zoo” represents, i.e., “animals in an enclosure.” When

determining whether the water is calm, which requires attention to specific image regions, the textual justification discusses foam on the waves.

Visually, this attention model is able to point to important visual evidence. For example in the top row of Fig. 2, for the question “Is this a zoo?” the visual explanation focuses on the field in one case, and on the fence in another.

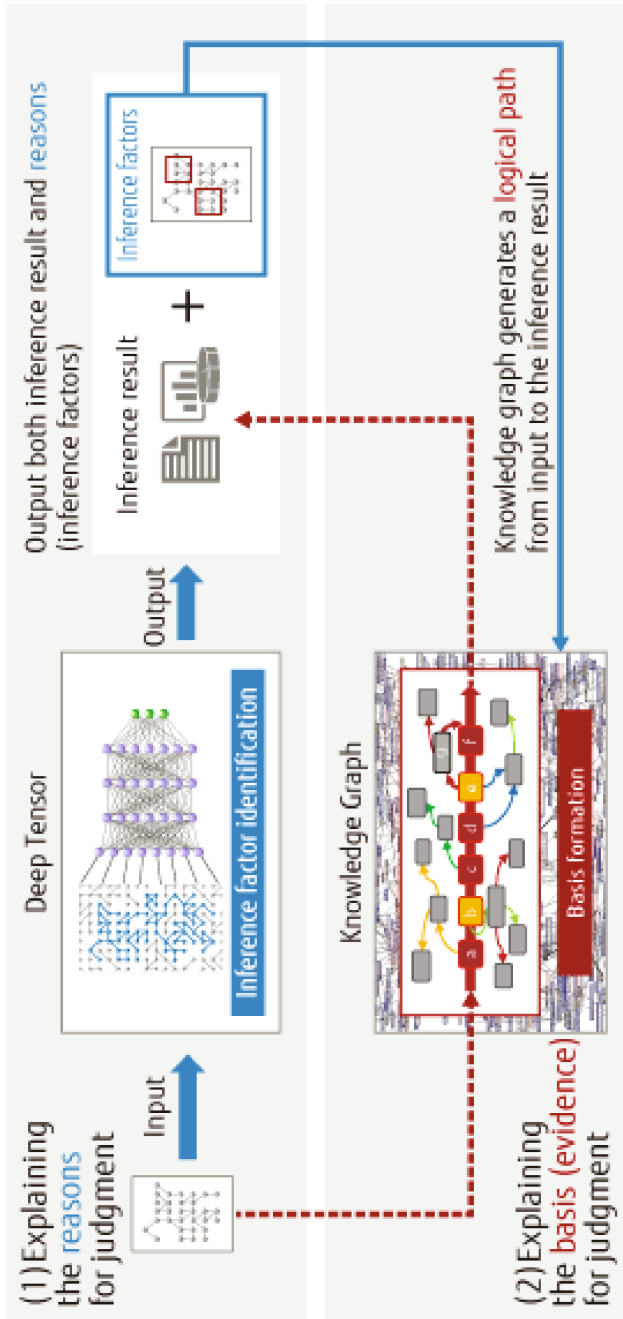
There are also other approaches to explanation that formulate heuristics for creating what have been called “Deep Visual Explanation” [1]. For example, in the application to debugging image classification learned models, we can create a heat map filter to explain where in an image a classification decision was made. There are an arbitrary number of methods to identify differences in learned variable distributions to create such maps; one such is to compute a Kullback-Leibler (KL) divergence gradient, experiments with which are described in [2], and illustrated in (see Fig. 5). In that figure, the divergence for each input image and the standard VGG image classification predictor is rendered as a heat map, to provide a visual explanation of which portion of an image was used in the classification.



**Fig. 5.** Explaining the decisions made by the VGG-16 (park bench, street sign, racket, cockatoo, traffic light and chihuahua), our approach highlights the most discriminative region in the image.

### 3 Conclusion and Future Outlook

We may think of an explanation in general as a filter on facts in a context [3]. An effective explanation helps the explainer cross a *cognitive valley*, allowing them to update their understanding and beliefs [4]. AI is becoming an increasingly ubiquitous co-pilot for human decision making. So AI learning systems will require



### Explainable AI with Deep Tensor and Knowledge Graph

Fig. 6. Explainable AI with Deep Tensor and a knowledge graph

explicit attention to the construction of problem domain models and companion reasoning mechanisms which support general explainability.

Figure 6 provides one example of how we might bridge the gaps between digital inference and human understanding. Deep Tensor [16] is a deep neural network that is especially suited to datasets with meaningful graph-like properties. The domains of biology, chemistry, medicine, and drug design offer many such datasets where the interactions between various entities (mutations, genes, drugs, disease) can be encoded using graphs. Let's consider a Deep Tensor network that learns to identify biological interaction paths that lead to disease. As part of this process, the network identifies *inference factors* that significantly influenced the final classification result. These influence factors are then used to filter a knowledge graph constructed from publicly available medical research corpora. In addition, the resulting interaction paths are further constrained by known logical constraints of the domain, biology in this case. As a result, the classification result is presented (explained) to the human user as an annotated interaction path, with annotations on each edge linking to specific medical texts that provide supporting evidence.

Explanation in AI systems is considered to be critical across all areas where machine learning is used. There are examples which combine multiple architectures, e.g., combining logic-based system with classic stochastic systems to derive human-understandable semantic explanations [14]. Another example is in the case of transfer learning [20], where learning complex behaviours from small volumes of data is also in strong needs of explanation of efficient, robust and scalable transferability [5].

**Acknowledgements.** The authors thanks their colleagues from local and international institutions for their valuable feedback, remarks and critics on this introduction to the MAKE-Explainable-AI workshop.


## References

1. Babiker, H.K.B., Goebel, R.: An introduction to deep visual explanation. In: NIPS 2017 - Workshop Interpreting, Explaining and Visualizing Deep Learning (2017)
2. Babiker, H.K.B., Goebel, R.: Using KL-divergence to focus deep visual explanation. CoRR, abs/1711.06431 (2017)
3. Chander, A., Srinivasan, R.: Evaluating explanations. In: Joint Proceedings of the IFIP Cross-Domain Conference for Machine Learning and Knowledge Extraction (IFIP CD-MAKE 2018) (2018)
4. Chander, A., Srinivasan, R., Chelian, S., Wang, J., Uchino, K.: Working with beliefs: AI transparency in the enterprise. In: Joint Proceedings of the ACM IUI 2018 Workshops Co-located with the 23rd ACM Conference on Intelligent User Interfaces (ACM IUI 2018) (2018)
5. Chen, J., Lecue, F., Pan, J.Z., Horrocks, I., Chen, H.: Transfer learning explanation with ontologies. In: Principles of Knowledge Representation and Reasoning: Proceedings of the Eleventh International Conference, KR 2018, 30 October–2 November 2018, Tempe, Arizona (USA) (2018, to appear)

6. Esteva, A., et al.: Dermatologist-level classification of skin cancer with deep neural networks. *Nature* **542**(7639), 115–118 (2017)
7. Hendricks, L.A., et al.: Generating visual explanations. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *ECCV 2016*. LNCS, vol. 9908, pp. 3–19. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46493-0\\_1](https://doi.org/10.1007/978-3-319-46493-0_1)
8. Holzinger, A., Biemann, C., Pattichis, C.S., Kell, D.B.: What do we need to build explainable AI systems for the medical domain? [arXiv:1712.09923](https://arxiv.org/abs/1712.09923) (2017)
9. Holzinger, A., et al.: Towards the augmented pathologist: challenges of explainable-AI in digital pathology. [arXiv:1712.06657](https://arxiv.org/abs/1712.06657) (2017)
10. Holzinger, A., et al.: Towards interactive Machine Learning (iML): applying ant colony algorithms to solve the traveling salesman problem with the human-in-the-loop approach. In: Buccafurri, F., Holzinger, A., Kieseberg, P., Tjoa, A.M., Weippl, E. (eds.) *CD-ARES 2016*. LNCS, vol. 9817, pp. 81–95. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-45507-5\\_6](https://doi.org/10.1007/978-3-319-45507-5_6)
11. Holzinger, A., et al.: A glass-box interactive machine learning approach for solving NP-hard problems with the human-in-the-loop. [arXiv:1708.01104](https://arxiv.org/abs/1708.01104) (2017)
12. Holzinger, K., Mak, K., Kieseberg, P., Holzinger, A.: Can we trust machine learning results? Artificial intelligence in safety-critical decision support. *ERCIM News* **112**(1), 42–43 (2018)
13. Kulesza, T., Burnett, M., Wong, W.-K., Stumpf, S.: Principles of explanatory debugging to personalize interactive machine learning. In: *Proceedings of the 20th International Conference on Intelligent User Interfaces*, pp. 126–137. ACM (2015)
14. Lécué, F., Wu, J.: Semantic explanations of predictions. *CoRR*, abs/1805.10587 (2018)
15. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* **521**, 436 (2015)
16. Maruhashi, K., et al.: Learning multi-way relations via tensor decomposition with neural networks. In: *The Thirty-Second AAAI Conference on Artificial Intelligence AAAI-18*, pp. 3770–3777 (2018)
17. Mnih, V., et al.: Human-level control through deep reinforcement learning. *Nature* **518**(7540), 529–533 (2015)
18. Moravčík, M.: Deepstack: expert-level artificial intelligence in heads-up no-limit poker. *Science* **356**(6337), 508–513 (2017)
19. O’Sullivan, S., et al.: Machine learning enhanced virtual autopsy. *Autopsy Case Rep.* **7**(4), 3–7 (2017)
20. Pan, S.J., Yang, Q.: A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **22**(10), 1345–1359 (2010)
21. Park, D.H., et al.: Multimodal explanations: justifying decisions and pointing to the evidence. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2018)
22. Ribeiro, M.T., Singh, S., Guestrin, C.: Why should i trust you?: Explaining the predictions of any classifier. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144. ACM (2016)
23. Singh, D., et al.: Human Activity recognition using recurrent neural networks. In: Holzinger, A., Kieseberg, P., Tjoa, A.M., Weippl, E. (eds.) *CD-MAKE 2017*. LNCS, vol. 10410, pp. 267–274. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-66808-6\\_18](https://doi.org/10.1007/978-3-319-66808-6_18)
24. Taigman, Y., Yang, M., Ranzato, M.A., Wolf, L.: Deepface: Closing the gap to human-level performance in face verification, pp. 1701–1708 (2014)



# A Rule Extraction Study Based on a Convolutional Neural Network

Guido Bologna<sup>1,2</sup>(✉) 

<sup>1</sup> University of Applied Sciences and Arts of Western Switzerland, Rue de la Prairie 4, 1202 Geneva, Switzerland

<sup>2</sup> University of Geneva, Route de Drize 7, 1227 Carouge, Switzerland  
Guido.Bologna@hesge.ch, Guido.Bologna@unige.ch

**Abstract.** Convolutional Neural Networks (CNNs) lack an explanation capability in the form of propositional rules. In this work we define a simple CNN architecture having a unique convolutional layer, then a Max-Pool layer followed by a full connected layer. Rule extraction is performed after the Max-Pool layer with the use of the Discretized Interpretable Multi Layer Perceptron (DIMLP). The antecedents of the extracted rules represent responses of convolutional filters, which are difficult to understand. However, we show in a sentiment analysis problem that from these “meaningless” values it is possible to obtain rules that represent relevant words in the antecedents. The experiments illustrate several examples of rules that represent n-grams.

**Keywords:** Convolutional Neural Networks · Rule extraction  
Sentiment analysis

## 1 Introduction

A natural way to explain neural network responses is by means of propositional rules [8]. Andrews et al. introduced a taxonomy to characterize all rule extraction techniques [1]. Diederich and Dillon presented a rule extraction study on the classification of four emotions from SVMs [6]. Extracting rules in Sentiment Analysis (SA) is very challenging with thousands of words represented in the inputs [6]. As a consequence Diederich and Dillon restricted their study to only 200 inputs and 914 samples. Bologna and Hayashi used the Quantized Support Vector Machine to generate rules explaining Tweets’ sentiment polarities [4]. In this work we propose a model to generate rules from a Convolutional Neural Network architecture (CNN) that is trained with a dataset related to SA. To the best of our knowledge this problem has not been tackled.

Since Convolutional Neural Networks (CNNs) started to be broadly used less than ten years ago, very few works have proposed to determine the acquired knowledge in these models. Several authors proposed to interpret CNNs in the neighborhood of the instances. As an example, Ribeiro et al. presented LIME



whose purpose is to learn an interpretable model in the local region close to an input instance [13]. Koh et al. determined the training instances that are the most important for the prediction [10]. Finally, Zhou et al. presented CAM whose purpose is to determine discriminative image regions using the average pooling of CNNs [14].

This work illustrates rule extraction from a CNN architecture shaped by an input layer representing sentences of word embeddings [11], a convolutional layer with filters, and max-pooling layer followed by a fully connected layer. After the training phase, the fully connected layer is replaced by a *Discretized Interpretable Multi Layer Perceptron* (DIMLP) [2] that allows us to extract rules. This subnetwork approximates the fully connected part of the original CNN to any desired precision. Nevertheless, the antecedents of the extracted rules represent maximal responses of convolutional filters, which can not be understood directly. Thankfully, it is possible to go back to the words that are relevant in the decision-making process. Specifically, these words are structured into n-grams, that depend on the size of the convolutional filters. In the following sections we first describe the used model, then we present the experiments, followed by the conclusion.

## 2 The Interpretable CNN Architecture

The CNN architecture used here is similar to those typically used in SA problems [9]. Before rule extraction, we train a CNN having only a fully connected layer between the last two layers of neurons. Then, we perform rule extraction by replacing these last two layers of neurons by a special subnetwork that can be translated into symbolic rules [2]. Hence, after training the modified CNN for rule extraction has two components:

- a CNN without the fully connected layer;
- a DIMLP with only a hidden layer that approximates the CNN fully connected layer.

Rules generated from the DIMLP component are related to filter thresholds in the convolutional layer of the CNN. From these values relevant words represented in the input layer can be determined. More details are given in the following paragraphs.

### 2.1 The CNN Subnetwork

Our CNN subnetwork is quite similar to that proposed in [5,9]. A smaller representation of the adopted CNN architecture is described in Fig. 1. The inputs are words represented by word embeddings of dimensionality  $d = 300$ , calculated with *word2vec* [11]. Here, the maximal number of words in a sentence is equal to  $s = 59$ , thus the size of an instance in the left part of Fig. 1 is equal to  $17700 (= 59 * 300)$ . Then, an instance is convolved with filters of different size. Specifically, for each filter we define a filtering matrix of size  $f \cdot d$ , with  $f = 1, 2, 3$ ,

corresponding to the number of words on which it operates. After convolution we apply a *Relu* function ( $\text{Relu}(x) = \text{Max}(0, x)$ ), which is non-linear. For each filter, the result of the convolution provides a vector of size  $s - f + 1$ . Afterward, the max-pooling operator is applied and the maximal value is retained, independently of where it is located in a sentence. All the max-pooling units are concatenated (right most layer in Fig. 1) and then follows a fully connected layer. In the output layer, a Softmax activation function is used. Specifically, for  $N$   $a_i$  scalars it calculates an N-dimensional vector with values between 0 and 1. It is given as:

$$S_i = \frac{\exp(a_i)}{\sum_{k=1}^N \exp(a_k)}; \quad \forall i \in 1 \dots N. \quad (1)$$

We use the Lasagne Library Package [7] to train our CNNs. The training phase is achieved with the Cross-Entropy loss function. After learning the fully connected layer, which is not represented in Fig. 1 is replaced by a DIMLP subnetwork having two layers of weights, with the second being equal to the fully connected layer of the CNN.

## 2.2 The DIMLP Subnetwork

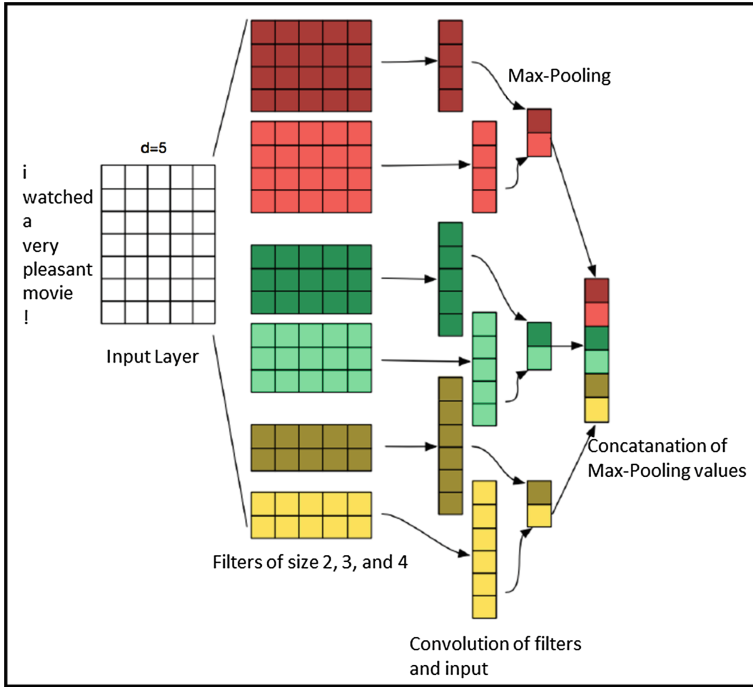
DIMLP differs from standard Multi Layer Perceptrons in the connectivity between the input layer and the first hidden layer. Specifically, any hidden neuron receives only a connection from an input neuron and the bias neuron, as shown in Fig. 2. After the first hidden layer, neurons are fully connected. Note that very often DIMLPs are defined with two hidden layers; the number of neurons in the first hidden layer being equal to the number of input neurons. The key idea behind rule extraction from DIMLPs is the precise localization of axis-parallel discriminative hyperplanes. In other words, the input space is split into hyper-rectangles representing propositional rules. Specifically, the first hidden layer creates for each input variable a number of axis-parallel hyperplanes that are effective or not, depending on the weight values of the neurons above the first hidden layer. More details on the rule extraction algorithm can be found in [3].

The activation function in the output layer of a standard DIMLP [2] is a sigmoid function given as

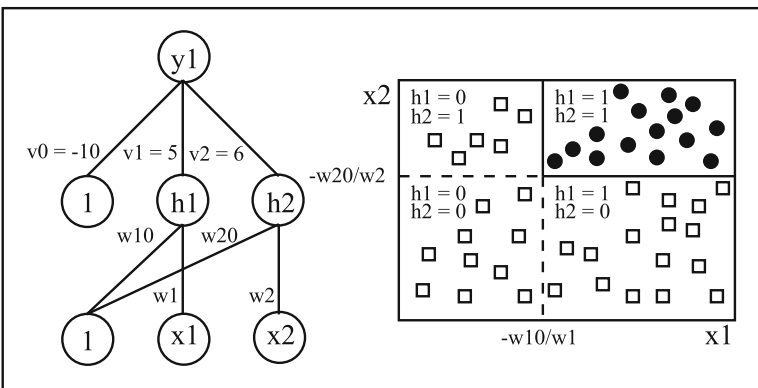
$$\sigma(x) = \frac{1}{1 + \exp(-x)}. \quad (2)$$

However, here since the CNN is trained with a Softmax function in the output layer, we replace the sigmoid by it. In the first hidden layer the activation function is a staircase function  $S(x)$  with  $Q$  stairs that approximate the Identity function  $I(x) = x; \forall x \in (R_{min}..R_{max})$ , with two constants  $R_{min}$  and  $R_{max}$ .

$$S(x) = R_{min}, \text{ if } x \leq R_{min}; \quad (3)$$



**Fig. 1.** The CNN subnetwork has an input layer with data provided as word embeddings; these word vectors are represented horizontally, while sentences are represented vertically. Then follows a convolutional layer with filters of different size. The convolution of the input with filters provides one-dimensional vectors (represented vertically). Finally, the “max” function reduces the size of these vectors that are finally concatenated (last layer on the right). The fully connected layer is not represented.



**Fig. 2.** A DIMLP network creating two discriminative hyperplanes. The activation function of neurons  $h_1$  and  $h_2$  is a step function, while for output neuron  $y_1$  it is a sigmoid.

$R_{min}$  represents the abscissa of the first stair. By default  $R_{min} = 0$ .

$$S(x) = R_{max}, \text{ if } x \geq R_{max}; \quad (4)$$

$R_{max}$  represents the abscissa of the last stair. By default  $R_{max} = 1$ . Between  $R_{min}$  and  $R_{max}$   $S(x)$  is given as

$$S(x) = I\left(R_{min} + \left[ q \cdot \frac{x - R_{min}}{R_{max} - R_{min}} \right] \left( \frac{R_{max} - R_{min}}{q} \right) \right). \quad (5)$$

Square brackets indicate the integer part function and  $q = 1, \dots, Q$ . The step function  $t(x)$  is a particular case of the staircase function with only one step:

$$t(x) = \begin{cases} 1 & \text{if } x > 0; \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

The approximation of the Identity function by a staircase function depends on the number of stairs. The larger the number of stairs the better the approximation. Note that the step/staircase activation function makes it possible to precisely locate possible discriminative hyperplanes.

As an example, in Fig. 2 assuming two different classes, the first is being selected when  $y_1 > \sigma(0) = 0.5$  (black circle) and the second with  $y_1 \leq \sigma(0) = 0.5$  (white squares). Hence, two possible hyperplane splits are located in  $-w_{10}/w_1$  and  $-w_{20}/w_2$ , respectively. As a result, the extracted unordered rules are:

- $(x_1 < -w_{10}/w_1) \rightarrow \text{square}$
- $(x_2 < -w_{20}/w_2) \rightarrow \text{square}$
- $(x_1 \geq -w_{10}/w_1) \text{ and } (x_2 \geq -w_{20}/w_2) \rightarrow \text{circle}$

### 2.3 From Meaningless Rules to Meaningful Rules

Each antecedent extracted from the max-pooling-layer is given as  $a < t$ , or  $a \geq t$ ;  $t$  being a filter threshold involving in the input layer a number of activated words. Specifically, these words correspond to bigrams and trigrams when the filter size in the convolutional layer is equal to two or three, respectively. In practice, filters of size equal to one are convolved with all possible single words. Then with the obtained values, we retain all the single words that make true a rule antecedent related to the max-pool-layer. This is repeated for filters of size two with respect to bigrams and so on with other filter sizes.

Generally, the condition expressed in a rule antecedent is true with more than one n-gram; thus a rule antecedent which is true for the DIMLP subnetwork implies a disjunction of one or more n-grams represented in the input layer (one or more n-grams connected by a logical or). Nevertheless, with the use of the ‘‘Max’’ function a unique n-gram becomes dominant (the one with the highest activation) and cancels the others.

Disjunctions of n-grams related to a rule antecedent extracted from the max-pooling-layer involving thousands of words could be considered too numerous.

However, in practice these words are not necessarily encountered, especially for rules activated by a small number of examples. These words will be useful for determining possible contradictions, such as the simultaneous presence of words/n-grams that are clearly of positive polarity and others of negative polarity.

### 3 Experiments

In the experiments we use a well-known binary classification problem describing movie reviews with Tweets [12].<sup>1</sup> The number of examples is equal to 10662, with half of them belonging to the positive class. Note that the total number of single words in the dataset is equal to 21426, while the total number of bigrams is equal to 111590. Words are coded into vectors of word embeddings of size 300 [11]. Two CNN architectures were defined; one with 50 filters of size one and two ( $f = 1, 2$ ) and another with 40 filters of size one, two and three. Hence, the last layer in Fig. 1 has 100 and 120 neurons, respectively. For deep learning training we use Lasagne libraries, version 0.2 [7]. The loss function is the categorical cross-entropy and the training parameters are:

- learning rate: 0.02;
- momentum: 0.9;
- dropout = 0.2;

From the whole dataset we selected the first 10% of the samples of each class as a testing set and the rest as training examples. Moreover, a subset of the training set representing 10% of it was used as a tuning set for early-stopping. Table 1 shows the results for the first CNN architecture. The first row of this Table is related to the original CNN, while the other rows provide results of the approximations obtained with the CNN-DIMLP combination by varying the number of stairs in the staircase activation function. Columns from left to right designate:

- train accuracy;
- predictive accuracy on the testing set;
- fidelity, which is the degree of matching between rules and the model;
- predictive accuracy of the rules;
- predictive accuracy of the rules when rules and model agree;
- number of extracted rules and total number of rule antecedents.

Note that these rules involve filter responses in the antecedents from which n-grams are determined (cf. Sect. 2.3). Table 2 presents the results obtained with the second CNN architecture.

The best predictive accuracy of rules was obtained with the second architecture, which takes into account trigrams. Note however that accuracy performance in this work is not a priority, since our purpose is to demonstrate how to generate

<sup>1</sup> Available at: <http://www.cs.cornell.edu/people/pabo/movie-review-data/>.

**Table 1.** Results obtained with the first CNN architecture including filters of size one and two, involving unigrams and bigrams. In the DIMLP subnetwork the number of stairs of the staircase activation function is varied from 20 to 200.

	Train Acc.	Test Acc.	Fidelity	Test Acc. (r1)	Test Acc. (r2)	#Rules/#Ant
CNN	<b>81.5</b>	<b>74.6</b>	–	–	–	–
CNN (q=20)	81.2	74.0	93.6	72.9	75.0	747/6159
CNN (q=50)	<b>81.5</b>	74.0	94.8	<b>74.3</b>	<b>75.5</b>	636/5730
CNN (q=100)	81.4	74.1	95.3	73.0	74.7	669/5619
CNN (q=200)	81.4	74.5	<b>95.8</b>	73.4	75.0	562/ <b>5131</b>

**Table 2.** Results obtained with the second CNN architecture including filters of size one, two and three, involving unigrams, bigrams and trigrams.

	Train Acc.	Test Acc.	Fidelity	Test Acc. (r1)	Test Acc. (r2)	#Rules/#Ant
CNN	<b>83.0</b>	<b>75.4</b>	–	–	–	–
CNN (q=20)	<b>83.0</b>	75.0	94.6	73.4	75.6	637/5633
CNN (q=50)	82.9	75.3	95.2	72.8	75.3	559/5065
CNN (q=100)	82.9	<b>75.4</b>	<b>95.5</b>	<b>75.2</b>	<b>76.5</b>	568/4885
CNN (q=200)	<b>83.0</b>	75.3	94.0	73.5	75.9	555/ <b>4798</b>

meaningful propositional rules from CNNs. Overall, fidelity on the testing set is above 90%, meaning that rules explain CNN responses in a large majority cases.

The antecedents of rules extracted from the DIMLP subnetwork involve long lists of n-grams. Here we illustrate several rules extracted from the first architecture with  $q = 200$  (the one with the highest fidelity). Rules are ranked according to their support with respect to the training set, which is the proportion of covered samples. Note that rules are not disjointed, which means that a sample can activate more than a rule. The first rule has a support of 765 training samples and 76 testing samples; it is given as:

$$- (f_{51} < 0.285) (f_{65} < 0.09) (f_{70} < 0.275) (f_{74} \geq 0.185) (f_{87} < 0.305) (f_{99} < 0.06) \text{ Class} = \text{POSITIVE}$$

Here,  $f$  designates maximal values of filters with respect to the max-pooling layer. Indexes between one and 50 are related to single words, while those between 51 and 100 correspond to bigrams. The accuracy of this rule is 93.9% on the training set and 88.1% on the testing set. Antecedent  $f_{74} \geq 0.185$  involves the presence of at least one bigram in a list of 1218 possible bigrams. Generally, the possible bigrams are not necessarily encountered in the tweets activating a rule. By negating all other antecedents of this rule we obtain a list of 12730 bigrams that are required to be absent.

We illustrate a number of sentences with possible bigrams including the dominant bigram, represented in bold and related to the rule shown above. Note that three consecutive words in bold represent two consecutive bigrams:

1. offers **that rare** combination of entertainment **and education**.
2. **a thoughtful**, provocative, insistently humanizing film.

3. **a masterful** film from a master filmmaker, **unique** in its deceptive grimness, **compelling** in its fatalist worldview.
4. cantet **perfectly captures** the hotel lobbies, two-lane highways, and roadside cafes that permeate vincent's days.
5. the film makes **a strong** case for the importance of the musicians in creating the motown sound.
6. **a compelling coming-of-age** drama about the arduous journey of **a sensitive** young girl through a series of foster homes and **a fierce** struggle to pull free from her dangerous and domineering mother's hold over her.
7. this delicately observed story, **deeply felt and masterfully** stylized, is **a triumph** for its maverick director.
8. **a portrait** of alienation **so perfect**, it will certainly succeed in alienating most viewers.
9. it's sincere to a fault, but, unfortunately, not **very compelling** or much fun.

The first eight tweets are correctly classified and we can clearly recognize words of positive polarity. The two tweets at the end of the list are classified as positive, whereas their class is negative. The last tweet is wrongly classified, because "not" before "very compelling" has been ignored. Regarding the ninth tweet, "so perfect" contributes without any doubt to a positive polarity; then, at the end of the sentence "alienating" contributes to its true negative classification, but it is not considered at all by the rule.

Rule number 17 is given as:

$$- (f_7 < 0.315) (f_{39} < 0.295) (f_{54} \geq 0.05) (f_{65} < 0.12) (f_{75} < 0.25) (f_{83} < 0.13) (f_{95} \geq 0.135) (f_{96} \geq 0.16) \text{ Class} = \text{POSITIVE}$$

It has a support of 404 samples in the training set and 47 samples in the testing set with an accuracy of 91.6% and 83.0%, respectively. The presence of one or more bigrams is imposed by  $f_{54}$ ,  $f_{95}$ , and  $f_{96}$ ; it contains 8266 elements. Ninety-six single words must be absent; they are related to  $f_7$ , and  $f_{39}$ . Moreover, 6307 mandatory absent bigrams depends on  $f_{65}$ ,  $f_{75}$ , and  $f_{83}$ . Ten tweets with their possible bigrams including the dominant bigram are shown here; the last two tweets are wrongly classified:

1. **an utterly compelling** 'who wrote it' in which the reputation of **the most** famous author who ever lived comes into question.
2. between the drama of cube? s personal revelations regarding what the shop means in the big picture, iconic characters gambol fluidly through the story, with **charming results**.
3. it's **an old** story, but **a lively** script, sharp acting and **partially animated** interludes make just **a kiss** seem **minty fresh**.
4. this is simply **the most fun** you'll ever have with **a documentary!**
5. one of **the best**, most **understated performances** of [jack nicholson's] career.
6. tadpole is **a sophisticated, funny and good-natured** treat, slight but **a pleasure**.
7. **a smart, sweet and playful romantic** comedy.
8. mr . parker has **brilliantly updated** his source and **grasped its** essence, composing **a sorrowful and hilarious tone** poem about alienated labor, or **an absurdist** workplace sitcom.

9. **beautifully filmed and well acted.** . . . but admittedly problematic in its narrative specifics.
10. one of the oddest **and most** inexplicable sequels in movie history.

The following list illustrate negative tweets correctly classified by a rule reaching an accuracy of 100% on the testing set:

1. it's an 88-min highlight reel that's **86 min too long.**
2. such an incomprehensible mess that it **feels less like bad cinema than like being** stuck in a dark pit having a nightmare **about bad** cinema.
3. during the **tuxedo's 90 min of** screen time, **there isn't** one true 'chan moment'.
4. **the script becomes** lifeless and falls apart like a cheap lawn chair.
5. **the script falls** back on **too many** tried-and-true shenanigans that hardly distinguish it from **the next teen** comedy.
6. a close-to-solid espionage thriller with the misfortune of being released a **few decades too late.**

Finally, we show another example of negative tweets correctly classified by another rule yielding predictive accuracy equal to 100%:

1. maybe leblanc thought, "hey, the movie about the baseball-playing monkey **was worse.**"
2. a muddled limp biscuit of a movie, a **vampire soap opera that doesn't make** much **sense even** on its **own terms.**
3. the **script becomes lifeless** and falls apart like a cheap lawn chair.
4. a baffling subplot involving smuggling drugs inside danish cows **falls flat**, and if you're going to alter the bard's ending, you'd **better have** a good alternative.
5. given the **fact that virtually no one** is bound to show up at theatres for it, the project **should have been** made for the tube.
6. jonathan parker's bartleby **should have been** the be-all-end-all of the modern-office anomie films.

## 4 Conclusion

We presented a model that allowed us to extract rules of high fidelity from a typical trained CNN architecture for Sentiment Analysis. Rule extraction was first applied to the layer before the output layer and then relevant words were determined from convolutional filters thresholds. Generated rules are described by disjunctions of n-grams that must be present or conjunctions of n-grams that must be absent. Moreover, extracted n-grams do not depend on particular positions in sentences. In the experiments, several examples of tweets with discriminatory bigrams that explained CNN responses were illustrated. These discriminatory words are important, as they can be used to understand how correct/wrong classifications are obtained by the classifier.



## References

1. Andrews, R., Diederich, J., Tickle, A.B.: Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowl.-based Syst.* **8**(6), 373–389 (1995)
2. Bologna, G.: Rule extraction from a multilayer perceptron with staircase activation functions. In: *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, 2000, IJCNN 2000*, vol. 3, pp. 419–424. IEEE (2000)
3. Bologna, G.: A model for single and multiple knowledge based networks. *Artif. Intell. Med.* **28**(2), 141–163 (2003)
4. Bologna, G., Hayashi, Y.: A rule extraction study from svm on sentiment analysis. *Big Data Cogn. Comput.* **2**(1), 6 (2018)
5. Cliche, M.: Bb.twtr at semeval-2017 task 4: Twitter sentiment analysis with cnns and lstms. arXiv preprint [arXiv:1704.06125](https://arxiv.org/abs/1704.06125) (2017)
6. Diederich, J., Dillon, D.: Sentiment recognition by rule extraction from support vector machines. In: *CGAT 09 Proceedings of Computer Games, Multimedia and Allied Technology 09. Global Science and Technology Forum* (2009)
7. Dieleman, S., et al.: Lasagne: First release, August 2015. <https://doi.org/10.5281/zenodo.27878>
8. Holzinger, A., Biemann, C., Pattichis, C.S., Kell, D.B.: What do we need to build explainable AI systems for the medical domain? arXiv preprint [arXiv:1712.09923](https://arxiv.org/abs/1712.09923) (2017)
9. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint [arXiv:1408.5882](https://arxiv.org/abs/1408.5882) (2014)
10. Koh, P.W., Liang, P.: Understanding black-box predictions via influence functions. arXiv preprint [arXiv:1703.04730](https://arxiv.org/abs/1703.04730) (2017)
11. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint [arXiv:1301.3781](https://arxiv.org/abs/1301.3781) (2013)
12. Pang, B., Lee, L.: A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In: *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*, p. 271. Association for Computational Linguistics (2004)
13. Ribeiro, M.T., Singh, S., Guestrin, C.: Why should i trust you?: Explaining the predictions of any classifier. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144. ACM (2016)
14. Zhou, B., Bau, D., Oliva, A., Torralba, A.: Interpreting deep visual representations via network dissection. arXiv preprint [arXiv:1711.05611](https://arxiv.org/abs/1711.05611) (2017)



# Evaluating Explanations by Cognitive Value

Ajay Chander<sup>(✉)</sup> and Ramya Srinivasan

Fujitsu Laboratories of America, Sunnyvale, CA 94085, USA  
ajay.chander@gmail.com

**Abstract.** The transparent AI initiative has ignited several academic and industrial endeavors and produced some impressive technologies and results thus far. Many state-of-the-art methods provide explanations that mostly target the needs of AI engineers. However, there is very little work on providing explanations that support the needs of business owners, software developers, and consumers who all play significant roles in the service development and use cycle. By considering the overall context in which an explanation is presented, including the role played by the human-in-the-loop, we can hope to craft effective explanations. In this paper, we introduce the notion of the “cognitive value” of an explanation and describe its role in providing effective explanations within a given context. Specifically, we consider the scenario of a business owner seeking to improve sales of their product, and compare explanations provided by some existing interpretable machine learning algorithms (random forests, scalable Bayesian Rules, causal models) in terms of the cognitive value they offer to the business owner. We hope that our work will foster future research in the field of transparent AI to incorporate the cognitive value of explanations in crafting and evaluating explanations.

**Keywords:** Explanations · AI · Cognitive value · Business owner  
Causal modeling

## 1 Introduction

Consumers, policymakers, and technologists are becoming increasingly concerned about AI as a ‘black-box’ technology. In order to engender trust in the user and facilitate comfortable interactions, it has become increasingly important to create AI systems that can explain their decisions to their users. Across a variety of fields, from healthcare to education to law enforcement and policy making, there exists a need for explaining the decisions of AI systems. In response to this, both the scientific and industrial communities have shown a growing interest in making AI technologies more transparent. The new European General Data Protection Regulation, the U.S. Defense Advanced Research Projects Agency’s XAI program [1], and institutional initiatives to

ensure the safe development of AI such as those of the Future of Life Institute, are a few of the many business, research, and regulatory incentives being created to make AI systems more transparent.

Many state-of-the-art methods provide explanations that mostly target the needs of AI engineers [10,11,13]. In other words, explanations assume some domain knowledge, or are generated for people with domain expertise. As the use of AI becomes widespread, there is an increasing need for creating AI systems that can explain their decisions to a large community of users who are not necessarily domain experts. These users could include software engineers, business owners, and end-users. By considering the overall context in which an explanation is presented, including the role played by the human-in-the-loop, we can hope to craft effective explanations.

### 1.1 Cognitive Value of an Explanation

The role of explanations and the way they should be structured is not new and dates back to the time of Aristotle [25]. The authors in [25] highlight the functions of explanations. They mention that explanations should accommodate novel information in the context of prior beliefs, and do so in a way that fosters generalization. Furthermore, researchers have also studied if certain structures of an explanation are inherently more appealing than others [26]. The authors in [23] state that explanations are social in that they are meant to transfer knowledge, presented as part of a conversation or interaction and are thus presented relative to the explainer’s beliefs about the user’s (i.e., explainee’s) beliefs.

We posit that *an explanation is a filter on facts*, and is presented and consumed as part of a larger context. Here, fundamental aspects of the context include: the entity presenting the explanation (“explainer”), the entity consuming the explanation (“explainee”), the content of the explanation itself, where the explanation is being presented, amongst others.

Let’s first understand the role of the explainee as it is the most crucial element of an explanation’s context. As discussed earlier, a wide variety of users are now interested in understanding the decisions of AI systems. There are at least four distinct kinds of users [3,4].

- *AI Engineers*: These are generally people who have knowledge about the mathematical theories and principles of various AI models. These people are interested in explanations of a functional nature, e.g. the effects of various hyperparameters on the performance of the network or methods that can be used for model debugging.
- *Software Developers and/or Integrators*: These are application builders who make software solutions. These users often make use of off-the-shelf AI modules, and integrate them with various software components. Developers are interested in explanation methods that allow them to seamlessly integrate various AI module into the use cases of their interest.
- *Business Owners*: These people are usually stakeholders who own the service and are interested in commercialization. The owner is concerned with

explainability aspects that can elucidate ways in which the application can be improved to increase financial gains, to justify predictions in order to aid in product design and people management, etc.

- *End-Users*: These are consumers of the AI service. These people are interested in understanding why certain recommendations were made, how they can use the information provided by the AI, how the recommendations will benefit them, etc.

As described above, users expect certain “cognitive values” from the explanations of AI systems. The term cognitive value can be best explained via examples. Some users may primarily expect explanations to account for personal values (e.g., privacy, safety, etc.) in the recommendations made by AI systems. In this case, the cognitive value of the explanation is to *engender trust* in the user. Some other users may largely expect explanations to be elucidating functional aspects of the AI models such as accuracy, speed and robustness; here the cognitive value of explanation is in aiding *troubleshooting and/or design*. Some users may expect explanations to help them understand the AI’s recommendation and aid them in analysis; in this case the cognitive value of explanation is in *educating* the user and help them take an appropriate *action*. Based on the task, any of the aforementioned cognitive values may be important to any of the user-types described. There could be many more cognitive values, but we believe that *trust, troubleshooting, design, education and action* are the most important cognitive values.

Thus, it becomes important to evaluate explanations based on their cognitive value in a given context. As an example, consider a business executive who wants to understand how to improve sales of the company. So, the operational goals of the explanation is largely in aiding *action* (i.e., the AI should help the business executive in specifying the steps that need to be taken in order to improve sales) and in *education* (i.e., the AI should inform the executive of the factors that determine sales, etc.). Consider some hypothetical explanations generated by an AI system as listed below.

- Factors X and Y are the most important factors in determining sales.
- Factors X and Y are the most important factors in determining sales, whenever  $X > 5$  and  $Y < 4$ , the sales is 90%.
- Factors X and Y are the most important factors responsible for sales in the past. Changing X to X+10 will improve the sales by 5%.

At a high-level, all of the aforementioned explanations look reasonable. Let us delve a little deeper. Suppose X was the amount of the product and Y was the location of the sale. Now, in explanation 2, the phrase “ $Y < 4$ ” does not convey a semantic meaning to the business owner. To the AI engineer, it may be still meaningful as the model might have mapped various locations to numbers. However, the business owner is not aware about this encoding. Even if she was made aware of what the individual numbers denoted (such as if the location is NYC, Tokyo, or Hamburg), as the number of such choices increases, the cognitive burden on the business owner increases and does not aid in educating him/her

or aiding in their action of how they can improve sales. Although explanation 1 provides semantically relevant information, it does not help the business owner in providing actionable insights in improving the sales. Explanation 3 not only educates the business owner in terms of the most important factors for improving sales, but more importantly also aids in action by suggesting *how* the sales can be improved.

The *contributions* of the paper are as follows: First, to the best of our knowledge, our work is the first to introduce the notion of “cognitive value” of an explanation and elaborate on the role of cognitive values in providing explanations to various kinds of users. Second, we compare three state-of-the-art explanation methods namely Scalable Bayesian Rule Lists [7], Random Forests, and Causal models [5] in terms of their cognitive value to the business owner. In particular, through a case study of a car dealer who is wanting to improve car sales, we show how causal models designed for explaining issues concerning fairness and discrimination [5] can be modified to provide explanations of cognitive value to this car dealer. Third, we discuss the merits and shortcomings of each of the aforementioned methods. We hope that our work will foster future research in the field of transparent AI to incorporate the cognitive value of explanations in evaluating the AI-generated explanations.

The rest of the paper is organized as follows. An overview of related work is provided in Sect. 2. The case study and the dataset is described in Sect. 3. Section 4 provides background on causal models, scalable bayesian rule lists and random forest algorithms. It also includes a description of how the causal model proposed in [5] for detecting discrimination can be leveraged to provide explanations of cognitive value. The types of explanations obtained from the three models are summarized in Sect. 5. A discussion of the relative merits and shortcomings of the explanations obtained by each of the aforementioned methods is also provided in Sect. 5. Conclusions are provided in Sect. 6.

## 2 Related Work

The new European General Data Protection Regulation (GDPR and ISO/IEC 27001) and the U.S. Defense Advanced Research Projects Agency’s XAI program [1] are probably the most important initiatives towards transparent AI. As a result, several academic as well as industrial groups are looking to address issues concerning AI transparency. Subsequently, a series of workshops, industrial meetings and discussion panels related to the area have taken place contributing to some impressive results.

Most of the work in the area is oriented towards the AI engineer and is technical. For example, in [10], the authors highlight the regions in an image that were most important to the AI in classifying it. However, such explanations are not useful to an end-user in either understanding the AI’s decision or in debugging the model [14]. In [19], the authors discuss the main factors used by the AI system in arriving at a certain decision and also discuss how changing a factor changes the decision. This kind of explanation helps in debugging for

the AI engineers. Researchers are also expanding the scope of explanations to AI agents by proposing frameworks wherein an AI agent explains its behavior to its supervisor [27]. The authors in [28] propose a model agnostic explanation framework and has been instrumental in several subsequent research efforts. There are several other impressive works across various fields catered towards helping the AI engineers [11, 13, 15, 16, 29–32, 38, 40]. A nice summary concerning explainability from an AI engineer’s perspective is provided in [22, 34].

More recently, there have been efforts in understanding the human interpretability of AI systems. The authors in [24] provide a taxonomy for human interpretability of AI systems. A nice non-AI engineer perspective regarding explanations of AI system is provided in [23]. The authors in [17] studied how explanations are related to user trust. They conducted a user study on healthcare professionals in AI-assisted clinical decision systems to validate their hypotheses. A nice perspective of user-centered explanations is provided in [18]. The author emphasizes the need for persuasive explanations. The authors in [21, 36] explore the notion of interactivity from the lens of the user. With growing interest in the concept of interpretability, various measures for quantifying interpretability have also been proposed in [39–41].

The closest to our work is perhaps [20] wherein the authors discuss how humans understand explanations from machine learning systems through a user-study. The metrics used to measure human interpretability are those concerning explanation length, number of concepts used in the explanation, and the number of repeated terms. Interpretability is measured in terms of the time to response and the accuracy of the response. Our measure is on the cognitive value an explanation offers as opposed to time to response or other such quantitative measures.

### 3 Case Study and Dataset

Our focus is on non-AI engineers. As a case study, we consider a scenario involving a business owner. Specifically, we consider a car dealer who wants to improve the sales of the cars. Thus, this user will benefit from knowing the steps that need to be taken in order to increase the sales of the cars. Thus, the cognitive value an explanation offers in this scenario should be in guiding towards an appropriate action and justifying the same.

We consider the car evaluation dataset [28, 42, 43] for our analysis, obtained from the UCI Machine learning repository. Although relatively an old dataset, it is appropriate for the problem at hand. The dataset is a collection of six attributes of cars as listed in Table 1. In the original dataset, the output attributes are “acceptable”, “unacceptable”, “good”, and “very good”. For the specific case study considered, we map acceptance to sales. For evaluation purposes, we map probability of acceptability to probability of selling the car and probability of unacceptability to probability of not being able to sell the car. There are 1728 instances in the dataset. The car dealer is interested in knowing what changes need to be done i.e., what factors of the cars need to be changed in order to improve sales.

**Table 1.** Description of input variables in the car evaluation dataset.

Input variable	Values
Buying price	vhigh, high, med, low
Price of the maintenance	vhigh, high, med, low
Number of doors	2, 3, 4, 5 more
Persons capacity in terms of persons to carry	2, 4, more
Size of luggage boot	small, med, big
Estimated safety of the car	low, med, high

## 4 Background

We consider three state-of-the-art algorithms for comparing the cognitive value they offer in the context of the aforementioned case study. We consider Random forests [8] as this model is one of the earliest interpretable models. We also consider two recent models scalable Bayesian rules [7] proposed in 2017, and causal models [5] proposed in 2018. For completeness, we provide some basic background about these models in the context of interpretability.

### 4.1 Random Forests

Random forests are a class of ensemble learning methods for classification and regression tasks [8]. The training algorithm for random forests applies the general technique of bootstrap aggregating, or bagging, to tree learners. Given a training set  $X = x_1, \dots, x_n$  with labels  $Y = y_1, \dots, y_n$ , bagging repeatedly ( $B$  times) selects a random sample with replacement of the training set and fits trees to these samples, i.e.,

For  $b = 1, \dots, B$ :

1. Sample, with replacement,  $n$  training examples from  $X, Y$ ; call these  $X_b, Y_b$ .
2. Train a classification or regression tree  $f_b$  on  $X_b, Y_b$ .

After training, predictions for unseen samples  $x$ 's can be made by averaging the predictions from all the individual regression trees on  $x$ 's or by taking a majority vote in the case of classification trees.

When considering a decision tree, for each decision that a tree (or a forest) makes there is a path (or paths) from the root of the tree to the leaf, consisting of a series of decisions, guarded by a particular feature, each of which contribute to the final predictions. The decision function returns the probability value at the leaf nodes of the tree and the importance of individual input variables can be captured in terms of various metrics such as the Gini impurity.

### 4.2 Scalable Bayesian Rule Lists (SBRL)

SBRLs are a competitor for decision tree and rule learning algorithms in terms of accuracy, interpretability, and computational speed [7]. Decision tree algorithms

are constructed using greedy splitting from the top down. They also use greedy pruning of nodes. They do not globally optimize any function, instead they are composed entirely of local optimization heuristics. If the algorithm makes a mistake in the splitting near the top of the tree, it is difficult to undo it, and consequently the trees become long and uninterpretable, unless they are heavily pruned, in which case accuracy suffers [7]. SBRLs overcome these shortcomings of decision trees.

Bayesian Rule Lists is an associative classification method, in the sense that the antecedents are first mined from the database, and then the set of rules and their order are learned. The rule mining step is fast, and there are fast parallel implementations available. The training set is  $(x_i, y_i)_i^n$ , where the  $x_i \in X$  encode features, and  $y_i$  are labels, which are generally either 0 or 1. The antecedents are conditions on the  $x$  that are either true or false. For instance, an antecedent could be: if  $x$  is a patient, antecedent  $a_j$  is true when the value of  $x$  is greater than 60 years and  $x$  has diabetes, otherwise false. Scalable Bayesian Rule Lists maximizes the posterior distribution of the Bayesian Rule Lists algorithm by using a Markov Chain Monte Carlo method. We refer interested readers to [7] for greater details related to the working of the algorithm.

### 4.3 Causal Models

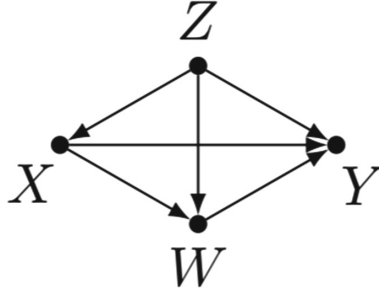
Causal models are amenable towards providing explanations as they naturally uncover the cause-effect relationship [37]. Before describing how causal models can be used to elicit explanations, we list some basic definitions used.

**Terminologies:** A structural causal model (SCM)  $M$  is a tuple  $h = [U, V, F, P(U)_i]$  where:  $U$  is a set of exogenous (unobserved) variables, which are determined by factors outside of the model;  $V$  is a set  $V_1, \dots, V_n$  of endogenous (observed) variables that are determined by variables in the model;  $F$  is a set of structural functions  $f_1, \dots, f_n$  where each  $f_i$  is a process by which  $V_i$  is assigned a value  $v_i$ ;  $P(u)$  is a distribution over the exogenous variables  $U$  [5].

Each SCM  $M$  is associated with a causal diagram  $G$ , which is a directed acyclic graph where nodes correspond to the endogenous variables ( $V$ ) and the directed edges denote the functional relationships. An intervention, denoted by  $do(X = x)$  [9], represents a model manipulation where the values of a set of variables  $X$  are set fixed to  $x$  regardless of how their values are ordinarily determined ( $f_x$ ). The counterfactual distribution is represented by  $P(Y_{X=x} = y)$  denotes the causal effect of the intervention  $do(X = x)$  on the outcome  $Y$ , where the counterfactual variable  $Y_{X=x}$  ( $Y_x$ , for short) denotes the potential response of  $Y$  to intervention  $do(X = x)$ . We will consistently use the abbreviation  $P(y_x)$  for the probabilities  $P(Y_{X=x} = y)$ , so does  $P(y|x) = P(Y = y|X = x)$ .

For our analysis, we consider a standard model provided in [5] as depicted in Fig. 1. We wish to determine the effect of  $X$  on  $Y$  (say  $X = \text{safety}$  and  $Y = \text{car sales}$ ). In this context,  $X$  would be the input factor and  $Y$  would be the output factor. There could be other factors  $Z$  and  $W$  affecting sales as shown in Fig. 1.





**Fig. 1.** Structural Causal Model considered for our analysis.

Here the factor  $Z$  is a common cause and is often referred to as a confounder. The factor  $W$  is called a mediator, because  $X$  could have a causal effect on  $Y$  through  $W$ .

There are three types of causal effects defined with respect to Fig. 1. The direct effect is modeled by the direct causal path  $X \rightarrow Y$  in Fig. 1. Indirect effect is modeled by the path  $X \rightarrow W \rightarrow Y$  and spurious effect is jointly modeled by the paths  $Z \rightarrow X$  and  $Z \rightarrow Y$ .

For each SCM, one can obtain the direct, indirect and spurious effects of  $X$  on  $Y$ . In particular, the authors in [5] define the concepts of counterfactual direct effect, counterfactual indirect effect and counterfactual spurious effects in order to estimate the discover discrimination and argue that by disentangling each of the causal effects, it can be ascertained whether there was genuine discrimination or not. The direct, (D.E.) indirect (I.E.) and spurious (S.E.) causal effects of changing the various factors on the output can be obtained from the following equations as provided in [5]. For more elaborate details, we refer readers to [5].

$$D.E_{x_0, x_1}(y|x) = \sum_{z, w} ((P(y|x_1, w, z) - P(y|x_0, w, z))P(w|x_0, z)P(z|x)) \quad (1)$$

$$I.E_{x_0, x_1}(y|x) = \sum_{z, w} (P(y|x_0, w, z)(P(w|x_1, z) - P(w|x_0, z))P(z|x)) \quad (2)$$

$$S.E_{x_0, x_1}(y) = \sum_{z, w} (P(y|x_0, w, z)P(w|x_0, z)(P(z|x_1) - P(z|x_0))) \quad (3)$$

#### 4.4 Adaptation of Causal Models to Provide Cognitive Value

Although the purpose of the authors of [5] was to explain discrimination, it is straightforward to extend this to obtain explanations that can offer cognitive values. Below, we describe the steps that need to be followed to obtain explanations of cognitive value.

- 
- 
- 1 Estimate the counterfactual direct effects for all possible combinations of SCMs for a given input  $X$  and output  $Y$ .
  - 2 Repeat Step 1 for all possible choice of input factors  $X$ .
  - 3 For each choice of input factor, generate textual statements highlighting the differential probability in output (e.g. differential probability in selling car) for change in the value of the input factor (e.g. changing the safety of the car from low to high).
  - 4 The factors corresponding to highest differential probabilities offer the most cognitive value (i.e. to increase sales) to the user (e.g. a car dealer).
- 

Put in other words, we consider all possible SCMs for the choice of factors  $[X, Z, W]$  as input, mediator and confounder. Note, for the standard model considered, only one confounder and one mediator is allowed. For the car evaluation dataset, we consider 4 factors for each SCM. Let us understand the above process for the car evaluation dataset.

Let us understand the usability of the aforementioned algorithm for the case study considered. We are interested in explaining how to improve the sales to the business owner (who is the car dealer in this example). So, the factor  $Y$  corresponds to sales. Suppose,  $X = \text{safety}$  and  $Y = \text{sale}$ . In the model shown in Fig. 1, one possibility could be  $W = \text{number of persons}$  and  $Z$  could be maintenance. This means, safety could affect car sales through the factor number of persons, and the factor maintenance could be a confounder affecting both safety and sales. Another possibility could be that  $W = \text{maintenance}$  and  $Z$  could be number of persons. In this case, the factor number of persons is a confounder and affects both sales and safety, and maintenance is a mediator.

Let us first consider the case wherein  $X$  is safety,  $Z$  is maintenance and let  $W$  be number of persons. Putting this in the standard model of Fig. 1 and using Eq. 1, we can estimate the counterfactual direct effect of safety on sales. The concept of counterfactual direct effect can be best understood through an example. Suppose there is a car with low safety. All other factors unchanged, if the factor safety alone were to be changed to high, then the quantity “counterfactual direct effect” can provide a measure of the improvement in sales for this factor change. Please note, in reality, since all the cars are manufactured, none of the factors can be changed. But, for the time being, assume an imaginary car whose factors can be changed. In that scenario, if the safety of the imaginary car were to be high, then one can ascertain if that change in safety contributes to rise or fall of sales and by how much. Knowing this differential sales probability will help in future design of such cars for the car dealer. Thus, it provides the cognitive value in taking appropriate *action* to the car dealer. We compute counterfactual direct effect for all possible choices of input factors  $X$ . Since the output factor is the sales, we conclude that factors with the highest magnitude of the counterfactual direct effect are the most important ones for the car dealer in improving the sales.

## 5 Dataset Analysis and Results

In this section, we state the results obtained from each of the three methods discussed in Sect. 4. We compare the three methods in terms of their cognitive value to the car dealer.

### 5.1 Results from Random Forests

The algorithm returns the probability value of sales for individual cars. In addition, variable importance scores in terms of mean decreasing impurity is provided that explains the importance of individual factors (i.e. safety, number of persons, etc.) in determining the sale of a car. Table 2 lists the variable importance scores in terms of mean decreasing Gini.

**Table 2.** Results from random forest algorithm.

Input factor	Importance
Buying price	92.15
Price of the maintenance	97.36
Number of doors	27.86
Persons capacity in terms of persons to carry	178.52
Size of luggage boot	51.19
Estimated safety of the car	215.87

It is apparent from the above table that safety and number of persons that can be accommodated are the most important factors in determining the sales of the cars. This information can certainly educate the car dealer about the most important factors that determine the sales.

### 5.2 SBRL

Let us next consider the result from scalable Bayes Rules List. As stated earlier, it is in the form of “if-then” associative rules. The results for the car evaluation dataset is as follows. Please note, the phrase ‘positive probability’ refers to the sale of the car. The rule numbers are generated by the algorithm and simply refer to the condition mentioned beside it in text. For example, rule [105] refers to the condition ‘number of persons = 2’.

If [persons=2] (rule[105]) then positive probability = 0.00173010  
 else if [safety=low] (rule[124]) then positive probability = 0.00259067  
 else if [doors=2,lug-boot=small] (rule[31]) then positive probability = 0.28787879  
 else if [buying=med,safety=high] (rule[22]) then positive probability = 0.98888889  
 else if [buying=low] (rule[16]) then positive probability = 0.94382022

else if [maint=vhigh,lug-boot=small] (rule[94]) then positive probability = 0.03125000  
 else if [buying=med] (rule[25]) then positive probability = 0.84523810  
 else if [lug-boot=small,safety=med] (rule[68]) then positive probability = 0.02631579  
 else if [maint=vhigh] (rule[98]) then positive probability = 0.01515152  
 else if [lug-boot=med,safety=med] (rule[64]) then positive probability = 0.52000000  
 else if [buying=high] (rule[10]) then positive probability = 0.98913043  
 else if [maint=high] (rule[77]) then positive probability = 0.03125000

Thus, SBRLs provide various conditions and state the probability of sales under that condition. Thus, if the number of persons is 2, the probability of sales is 0.1%.

### 5.3 Causal Models

Table 3 provides a summary of the results obtained from the causal model described in Sect. 4.4.

The results summarized in Table 3 can be understood via examples. As an instance, consider the first row corresponding to safety. The result of that row states - “All other factors unchanged, if the safety of the car is changed from low to high, there will be 36.36% improvement in sales.” The next row corresponding to safety reads thus: “All other factors unchanged, if the safety of the car is changed from high to low, there will be 50% drop in sales.”. A positive value of differential probability indicates that there will improvement in sales upon changing the corresponding input factor (e.g. sales) in the stated manner (i.e. from low to high safety). A negative differential probability corresponds to a drop in sales.

**Table 3.** Results from causal explanation.

Input factor	Real car	Imaginary car	Differential probabilities (expressed as %) in selling Real car- selling Imaginary car
Safety	Low	High	+36.36%
Safety	High	Low	-50%
Number of persons	2	4	+32.34%
Number of persons	4	2	-43%
Maintenance	High	Low	+2.5%
Maintenance	Low	High	-10%

Table 3 re-iterates the result of random forest. It can be noted that safety and number of persons are the most important factors in determining sales. Note, Table 3 highlights the most important factors in improving sales and hence some factors (e.g. lug-boot) are omitted from the table.

## 5.4 Discussion

In this section, we discuss the merits and de-merits of all the three methods from the perspective of cognitive value the respective explanations offer to the users

**Random Forest:** The random forest educates the car dealer about the most important factors responsible for car sales in a relative manner. The significance of absolute values of the importance scores is not clear as their range is unknown. Furthermore, knowing the factor importance scores does not help the car dealer in understanding what needs to be done in order to improve the sales. The result may thus only educate the car dealer in knowing the most important factors affecting sales, but it is unclear as to how those factors need to be changed in order to improve sales.

**SBRLs:** There are many if-else statements in the explanation provided by SBRLs. The specific conditions are chosen automatically by the algorithm and can consist of multiple conditions that may be difficult for the user to interpret. Even if one parses for the ones with highest positive probabilities (implying sales of cars), it neither conveys semantically relevant information nor provides actionable insights to the car dealer. For example, the highest probability of 0.989 corresponds to the rule “if buying = high”. Does this mean cars with high buying price sell more? If true, it does not seem practically very true or compelling. Even assuming that it is true, it does not provide actionable insights to the car owner. By how much can the price be increased to achieve a certain sales target? Such kind of information is lacking in this model’s result.

**Causal Models:** Unlike random forest which could not ascertain how those factors are important and in particular how the car dealer should change those to improve sales, the explanation from the causal model provides *actionable* insights to the car dealer in improving sales. Furthermore, the results from the causal model is semantically meaningful and practically relevant.

**Table 4.** Comparison of results: RF denotes random forests, CM denotes causal models, SBRL denotes Scalable Bayesian Rule Lists

Method	Educates the user	Provides actionable insights to the user	Easy to comprehend
RF	Provides relative importance of factors in sales	No	Range of variable importance is not clear
SBRL	Informs about sales under certain conditions	No	Several conditions to parse
CM	Provides relative importance of factors in sales	Explains how the sales can be improved	Yes

Although the explanations based on causal modeling offers cognitive value to the users, it comes at a price. Specifically, one has to try with different structural assumptions. For a non-domain expert, this can really be time consuming. Also, the causal explanation formula works best for binary data. While this is good in providing instance level explanations (local explanations), it may not be easy to derive global explanations.

Table 4 provides a comparison of the three methods in terms of their cognitive value to the car dealer.

## 6 Conclusions

We introduced the concept of “cognitive value” of explanations of AI systems to users. We considered the scenario of a business owner seeking to improve sales of their product and compared explanations provided by some state-of-the-art AI methods in terms of the cognitive value they offer to the business owner. Specifically, we studied random forest, scalable bayesian rule lists and causal explanations towards this end. For the context considered, causal explanations provided the best cognitive value in terms of providing the business owner with actionable insights in improving his/her sales. We hope our work will foster future research in the field of transparent AI to incorporate the cognitive value of explanations in assessing explanations.

## References

1. Gunning, D.: Explainable artificial intelligence (XAI). Defense Advanced Research Projects Agency (DARPA) (2017)
2. Miller, T. : Explanation in artificial intelligence: insights from the social sciences. [arXiv:1706.07269](https://arxiv.org/abs/1706.07269) (2017)
3. Chander, A. et al.: Working with beliefs: AI transparency in the enterprise. In: Explainable Smart Systems Workshop, Intelligent user Interfaces (2018)
4. Ras, G., Gerven, M., Haselager, P.: Explanation methods in deep learning: users. Values, concerns and challenges. [arXiv:1803.07517](https://arxiv.org/abs/1803.07517) (2018)
5. Zhang, J., Bareinboim, E.: Fairness in decision-making - the causal explanation formula. In: AAAI (2018)
6. Zupan, B. et al.: Machine learning by function decomposition. In: ICML (1997)
7. Yang, H., Rudin, C., Seltzer, M.: Scalable Bayesian rule lists. In: ICML (2017)
8. Breiman, L.: Random forests. *Mach. Learn.* **45**(1), 5–32 (2001)
9. Pearl, J.: *Causality: Models, Reasoning, and Inference*. Cambridge University Press, New York (2000)
10. Selvaraju, R. et al.: Grad-CAM: visual explanations from deep networks via gradient-based localization. In: International Conference on Computer Vision (2017)
11. Son, T.: Unsupervised neural-symbolic integration. In: XAI Workshop, IJCAI (2017)
12. Hendricks, L.A., Akata, Z., Rohrbach, M., Donahue, J., Schiele, B., Darrell, T.: Generating visual explanations. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *ECCV 2016*. LNCS, vol. 9908, pp. 3–19. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46493-0\\_1](https://doi.org/10.1007/978-3-319-46493-0_1)


13. Park, D.: Multimodal explanations: justifying decisions and pointing to the evidence. CoRRabs//1802.08129 (2018)
14. Chandrashekar, A., et al.: It takes two to tango: towards theory of AI's mind. [arXiv:1704.00717](#) (2017)
15. Koh, P., Liang, P.: Understanding black-box predictions via influence functions. In: ICML (2017)
16. Melis, D., Jaakkola, T.: A causal framework for explaining the predictions of black-box sequence-to-sequence models. [arXiv:1707.01943](#) 2017
17. Bussone, A., Stumph, S., O'Sullivan, D.: The role of explanations on trust and reliance in clinical decision support systems. In: IEEE Conference on Healthcare Informatics (2015)
18. Herman, B.: The promise and peril of human evaluation for model interpretability. In: NIPS Workshop (2017)
19. Doshi-Veklez, F., Kortz, M.: Accountability of AI under the law: the role of explanation. [arXiv:1711.01134](#) (2017)
20. Narayanan, M., et al.: How do humans understand explanations from machine learning systems? An evaluation of the human-interpretability of explanation. [arXiv:1802.00682](#) (2018)
21. Amershi, S., et al.: Power to the people: the role of humans in interactive machine learning. *AI Mag.* (2017)
22. Lipton, Z.: The mythos of model interpretability. In: International Conference on Machine Learning Workshops (2016)
23. Millers, T., Howe, P., Sonnenberg, L.: Explainable AI: beware of inmates running the asylum. [arXiv:1712.00547](#) (2017)
24. Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning. [arXiv:1702.08608](#) (2017)
25. Lombrozo, T.: The structure and function of explanations. *Trends Cogn. Sci.* **10**(10), 464–470 (2006)
26. Rosemary, R.: What makes an explanation a good explanation? Adult learners' criteria for acceptance of a good explanation, Masters thesis, University of Newfoundland (1999)
27. Molineaux, M., Dannenhauer, D., Aha, D.: Towards explainable NPCs: a relational exploration learning agent. In: Osborn, J.C. (ed.) *Knowledge Extraction from Games: Papers from the AAAI Workshop (Technical report WS-18-10)*. AAAI Press, New Orleans, LA (2018)
28. Ribeiro, M., Singh, S., Guestrin, C.: Why should i trust you? Explaining the predictions of any classifier. In: KDD (2016)
29. Langley, P.: Explainable agency for intelligent autonomous systems. In: AAAI (2017)
30. Sifa, R.: Interpretable matrix factorization with stochasticity constrained nonnegative DEDICOM (2017)
31. Tamagnini, P.: Interpreting black-box classifiers using instance-level visual explanations. In: *Proceedings of the 2nd Workshop on Human-In-the-Loop Data Analytics* (2017)
32. Bibal, A., Freney, B.: Interpretability of machine learning models and representations: an introduction. In: ESANN (2016)
33. Alonso, J.: An exploratory study on the benefits of using natural language for explaining fuzzy rule-based systems. In: IEEE International Conference on Fuzzy Systems (2017)
34. Doran, D., Schulz, S., Besold, T.: What does explainable AI really mean? A new conceptualization of perspectives. [arXiv:1710.00794](#) (2017)

35. Brinkrolf, J., et al.: Interpretable machine learning with reject option. *at-Automatisierungstechnik* **66**(4), 283–290 (2018)
36. Melnikov, A.: Towards dynamic interaction-based model. [arXiv:1801.03904](https://arxiv.org/abs/1801.03904) (2018)
37. Harradon, M., Druce, J., Ruttenberg, B.: Causal learning and explanation of deep neural networks via autoencoded activations. [arXiv:1802.00541](https://arxiv.org/abs/1802.00541) (2018)
38. Holzinger, K.L.: Can we trust machine learning results? Artificial intelligence in safety-critical decision support. *ERCIM News* (2018)
39. Alonso, J.M., Magdalena, L., Gonzalez-Rodriguez, G.: Looking for a fuzzy system interpretability index: an experimental approach. *Int. J. Approx. Reason.* **51**(1), 115–134 (2009)
40. Alonso, J.M., Castiello, C., Mencar, C.: A bibliometric analysis of the explainable artificial intelligence research field. In: Medina, J., et al. (eds.) *IPMU 2018. CCIS*, vol. 853, pp. 3–15. Springer, Cham (2018). [https://doi.org/10.1007/978-3-319-91473-2\\_1](https://doi.org/10.1007/978-3-319-91473-2_1)
41. Gacto, M.J., Alcalá, R., Herrera, F.: Interpretability of fuzzy rule-based systems: an overview of interpretability measures. *Inf. Sci.* **181**(20), 4340–4360 (2011)
42. Dua, D., Efi, K.D.: *UCI Machine Learning Repository*, University of California, Irvine, School of Information and Computer Sciences (2017)
43. Bohanec, M., Rajkovic, V.: Knowledge acquisition and explanation for multi-attribute decision making. In: *International Workshop on Expert Systems and Applications* (1988)





# Measures of Model Interpretability for Model Selection

André Carrington<sup>1</sup> , Paul Fieguth<sup>1</sup>, and Helen Chen<sup>2</sup>

<sup>1</sup> Systems Design Engineering, University of Waterloo, Waterloo, ON, Canada  
{amcarrin,pfieguth}@uwaterloo.ca

<sup>2</sup> School of Public Health and Health Systems, University of Waterloo,  
Waterloo, ON, Canada  
helen.chen@uwaterloo.ca

**Abstract.** The literature lacks definitions for quantitative measures of model interpretability for automatic model selection to achieve high accuracy and interpretability, hence we define inherent model interpretability. We extend the work of Lipton *et al.* and Liu *et al.* from qualitative and subjective concepts of model interpretability to objective criteria and quantitative measures. We also develop another new measure called simplicity of sensitivity and illustrate prior, initial and posterior measurement. Measures are tested and validated with some measures recommended for use. It is demonstrated that high accuracy and high interpretability are jointly achievable with little to no sacrifice in either.

**Keywords:** Model interpretability · Model transparency  
Support vector machines · Kernels

## 1 Introduction

For machine learning (ML) models, data and results, there is a demand for transparency, ease of understanding and explanations [24] to satisfy a citizen’s “right to explanation” in the European Union [20] and to meet health care requirements for justification and explanation [7, 22].

Without quantitative measures of transparency and understandability, doctors (or users) will select models which maximize accuracy but may unnecessarily or unintentionally neglect or sacrifice transparency and understandability, or they will choose models in an ad hoc manner to try and meet all criteria. We refer to the transparency and understandability of models as *inherent model interpretability*—defined further in Sect. 3.

We propose criteria and measures of inherent model interpretability to help a doctor select ML models (Table 1 steps 1 and 2) which are more transparent and understandable, in a quantitative and objective manner. More transparent models can offer additional views of results (Table 1 step 3) for interpretation.

**Table 1.** Measures of inherent model interpretability facilitate model selection (bold text) in steps 1 and 2.

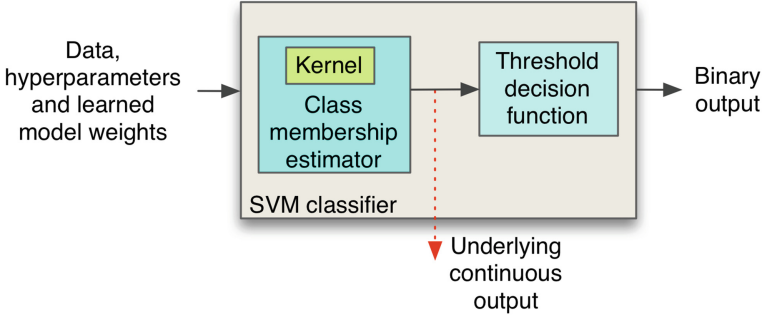
Step	Task	Basis for task
1	The doctor <b>selects candidate models</b> for learning and testing based on...	Data types and distributions, Inherent model interpretability (transparency of model)
2	The machine learns model weights for optimal accuracy with various parameters. The doctor <b>selects the model to use</b> based on...	Accuracy, Inherent model interpretability (transparency of model and understandability of results)
3	The doctor uses the model to classify new data. The doctor understands and interprets the result and model based on...	Theory, Views of results, Additional views of results
4	The doctor explains the result and model to a patient or peer based on...	Selected interpretations, Theory

Our measures facilitate the inclusion of better models as candidates and the selection of better models for use.

Some of our proposed measures are specific to support vector machines (SVM), as one popular ML method. We perform experiments to validate the SVM measures against a set of propositions and evaluate their utility by concordance or matched pair agreement.

Notably, the proposed measures **do not** provide an interpretation or explanation. They also **do not** indicate how useful or meaningful a model is in the context of data. For example, a model that always classifies patient data as belonging to the positive class is very understandable (interpretable). We can easily construct the explanation of the model and result—all patients are classified as positive—but that does not mean that the model is useful, meaningful, appropriate, or unbiased. Accuracy and common sense address the latter issues. The proposed measures only indicate how understandable a model is, i.e., how likely we are **able** to provide an interpretation, as the necessary basis for subsequent explanation.

Making ML more interpretable facilitates its use in health care because there is a perception that ML is a black box [31] lacking interpretability which inhibits its use. Greater use is important because for a good number of health care problems and data, ML methods offer better accuracy in classification [12, 15, 41] than common alternatives among statistical methods, decision trees and rule-based methods and instance-based methods. Interpretable ML also facilitates research on models and model fit.



**Fig. 1.** A model consists of a learning method, SVM in this case, and all of its associated parts as depicted above. Most machine learning and statistical models (or classifiers) have an underlying continuous output that most accurately describes the model’s behaviour.

## 2 Notation

A machine learning task begins with data in a matrix  $X$  consisting of  $N$  instances  $\underline{x}_i$  which are vectors, each containing  $n$  features.

$$X = [\underline{x}_1, \underline{x}_2, \dots, \underline{x}_N]^T \quad \underline{x}_i \in \mathbb{R}^n \tag{1}$$

Entry  $x_{i,j}$  in the matrix is the  $j^{th}$  feature of instance  $\underline{x}_i$ . We assume real-valued features converting any atomic data type to reals as needed (Appendix A).

A supervised learning task also has  $N$  targets (or outcomes) in a vector  $\underline{y}$  which are binary in classification,

$$\underline{y} = [y_1, y_2, \dots, y_N]^T \quad y_i \in \{-1, +1\} \tag{2}$$

or continuous in regression:

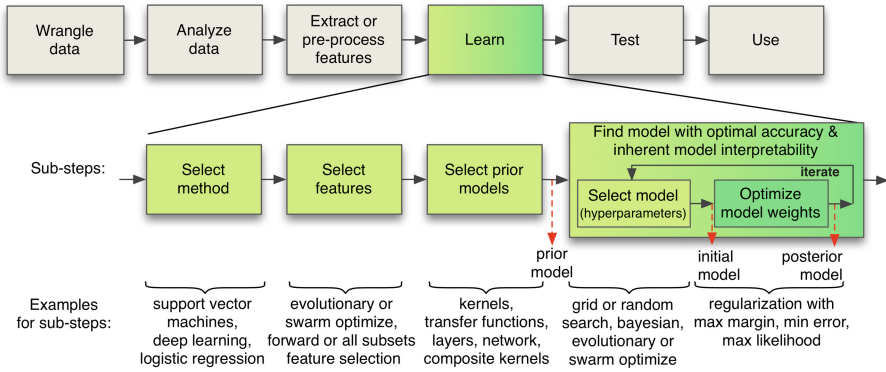
$$\underline{y} = [y_1, y_2, \dots, y_N]^T \quad y_i \in \mathbb{R} \tag{3}$$

In binary classification there are  $N^+$  instances in the positive class and  $N^-$  instances in the negative class.

We refer to a **posterior model** (e.g., Fig. 1), or simply **model**, as a learning method (e.g., SVM, neural networks) with all of its associated learning/estimation functions (e.g., kernels and transfer functions), hyperparameters, structure (e.g., layers, connections, components in a composite kernel), constraints and learned model weights, *in the context of specific data*. A model only learns from, and has meaning in, the context of specific data.

We refer to an **initial model** as a model in the context of specific data with initial model weights prior to learning/iteration.

We refer to a **family of models**, or a **prior model**, as the set of models possible when hyperparameters are variables (not specified)—e.g., SVM with a Gaussian RBF kernel with unspecified box constraint and kernel width.



**Fig. 2.** We measure inherent model interpretability at several points (dashed arrows) in the process of machine learning and/or statistical learning (partially derived from [25]). Note: some steps may not apply to some methods and models.

The prior, initial and posterior models are available at different points in the process of machine learning and/or statistical learning process (Fig. 2). Other notation is introduced in the context of discussion.

### 3 Inherent Model Interpretability Concept and Measures

We propose the concept of inherent model interpretability as distinguished from an individual’s understanding and we propose two measures for any learning method or model with numeric inputs.

Feynman said that if we understand a concept we must be able to describe it at a freshman level, which often requires simplification or reduction, otherwise we don’t really understand it [21]. Badii et al. express that complexity is closely related to understanding and that understanding comes from accurate models which use condensed information or reduction schemes [4]. Miller indicates that selection is a key attribute of explanations [38]. Hence, we posit that the simpler a model is, the easier it is to understand, interpret and describe, with all other aspects of the model being equal. This leads to the following general measure.

#### 3.1 A General Measure of Inherent Model Interpretability

As stated above, the simpler a model is, the more interpretable it is, inherently. Formally, we propose the following definition.

**Definition 1.** *Inherent model interpretability (or understandability)  $U$ , is a measure with range  $[0, 1]$  based on either: a measure of model transparency  $T$  in the same range, the inverse of semi-infinite model complexity  $H_\infty$ , or the inverse of finite model complexity  $H_b$ , respectively as follows:*

$$U = \begin{cases} T & (i) T \in [0, 1] \\ \frac{1}{1+(H_\infty-a)} & (ii) H_\infty \in [a, \infty) \quad a \in \mathbb{R}^+; a < \infty \\ 1 - \left(\frac{H_b-a}{b-a}\right) & (iii) H_b \in [a, b] \quad a, b \in \mathbb{R}^+; a, b < \infty \end{cases} \quad (4)$$

where:

- $H_\infty$  and  $H_b$  are measures of model complexity based on parts [4] in the categories of information, entropy, code length or dimension [33],
- *inherent* indicates that the measure is independent of an individual, e.g., their specific learning and forgetting curves [44], and
- the multiplicative inverse [29] in (4)ii or additive inverse [57] in (4)iii are applied as needed for **absolute** or **relative** measure respectively according to the comparison required. The relative measure is preferred where applicable since it is more intuitive and interpretable (not shown).
  - e.g., to compare a set of models where the range  $[a, b]$  is known to encompass them all, a relative measure (iii) is fine, however, to compare them to any future model where the maximum  $b$  is not known, use an absolute measure (ii), i.e., let  $b = \infty$ .

The separation of model interpretability into at least two parts, one part that is inherent to the model (and data) and another part that depends on the individual, aligns with the functionally-grounded approach [17].

In order to use this general measure, one must further define  $T$ ,  $H_\infty$  or  $H_b$ , as we do in subsequent sections. We note also that measurement may be performed prior to, initially at, or posterior to, optimizing the model weights (Fig. 2).

### 3.2 A New Measure: Simplicity of Output Sensitivity

We consider the continuous underlying output of a classifier (e.g., Fig. 1) to be the most accurate representation of a classifier’s behaviour. It is available most learning classifiers, in machine learning or statistical learning, such as, neural networks, SVM, logistic regression and naive bayes. It is also facilitated by most implementations, e.g., for SVM it is available in Matlab, R, Python, SPSS, Weka, libsvm and Orange, where the output may be the probability of the positive class or a non-probabilistic value, e.g., “classification score”.

Some measure or analyze a classifier’s behaviour based on its binary output instead [46]—this approach lacks fine-grained behavioural information. Others measure classifier behaviour by modeling its responses with a separate explanation model that provides a continuous output [5, 46]—this post hoc approach may not meet untested legal, assurance or business requirements.

We use the underlying continuous output, and the logic similar to the previous measure to posit that:

If a model is **uniformly sensitive** in its output to changing values in input features and instances, then its *sensitivity* is simple to describe, understand and interpret (as one value). Conversely, a model that is **differently sensitive** to each feature and instance is more difficult to describe, understand and interpret, in those terms or from that perspective. Formally, we propose the following definition:

**Definition 2.** *The simplicity of output sensitivity  $U_{H_s}$  is a measure of inherent model interpretability. It describes the simplicity of the sensitivity of the model’s continuous output (e.g., Fig. 1) to changes in input. It is specified as the inverse of Shannon entropy  $H_s$  with a finite range (4)iii, repeated below:*

$$U_{H_s} = 1 - \left( \frac{H_s}{H_{max}} \right) \quad H_s \in [0, H_{max}] \tag{5}$$

$$H_s = - \sum_i f_i(s) \log f_i(s), \quad i = 1 \dots N_s \tag{6}$$

$$H_{max} = - \sum_{i=1}^{|s|} \frac{1}{|s|} \log \frac{1}{|s|} \tag{7}$$

where  $s$  is the set of sensitivities  $S_{j,q}$  of the model’s continuous output  $\hat{y}_c$  (the value which is underlying for a classifier) to small changes  $\varepsilon = (0.1) \cdot 3\sigma$  in each input instance  $j$ , one feature  $q$  at a time,

$$s = \{S_{j,q}\} \tag{8}$$

$$S_{j,q} = \frac{\hat{y}_c(\underline{x}_j + \underline{\varepsilon}_q) - \hat{y}_c(\underline{x}_j - \underline{\varepsilon}_q)}{2\varepsilon} \tag{9}$$

$$\underline{\varepsilon}_q = [\dots 0 \ \varepsilon \ 0 \ \dots]^T \quad \varepsilon \text{ in } q^{th} \text{ cell}$$

and where  $N_s$  is the number of bins according to standard binning methods for histograms [18, 47, 53].

We use entropy to measure the global complexity of sensitivities across the space for input data. In the literature, entropy has been applied quite differently to measure the information loss of perturbed features, to indicate their influence—we use entropy instead to measure the complexity of influence with perturbed features.

Our measure uses a first-order central difference (first derivative approximation) as a standard and easy to understand approach to sensitivity that does not require knowing or differentiating the model’s formulas. We can generalize this idea to second and third-order differences/derivatives, and so on, like the derivatives in deep Taylor decomposition [39]—but the latter requires a model’s formulas and derivatives. Whereas [39] examines the local behaviours of a model, we do that and compute the complexity of the values.

We treat the entries  $S_{j,q}$  as a set or random variable  $s$  (8) because we are measuring model interpretability overall, across features and instances, not within a feature nor within an instance.

We note that instead of Shannon entropy, it may be possible to apply other types of entropy, such as Renyi entropy, Tsallis entropy, effective entropy or total information [19, 45, 56] and/or Kullback-Leibler (K-L) divergence [14], however such a change would require validation. Prior to this study we experimented with discrete Kullback-Leibler (K-L) divergence as implemented by four measures in the ITK toolkit [54, 55], as an alternative to Shannon entropy, however,

our experimental results with K-L divergence did not sufficiently match our expectations, so we focused on Shannon entropy as a more popular and credible measure.

We also implemented differential entropy [14], which is the continuous version of entropy and is defined as the K-L divergence from a uniform probability density function (pdf) to the pdf of interest, but put that aside based on the previously mentioned K-L divergence results and also because it was more compute intensive as it required a kernel density estimate.

Finally we note that the sensitivity portion of our measure (i.e., entropy aspect aside) differs from how other authors compute sensitivity globally across both instances and features [27].

### 4 Criteria for Model Transparency and a Measure for SVM

We identify criteria for model transparency from the literature (Table 2) for any model, and propose new criteria in most cases, which are objective, not subjective, and thus suitable for a (quantitative) measure of model transparency.

We apply the proposed criteria (Table 2) for any model, to create a measure specific to kernel methods or support vector machines (SVM).

We use the seven proposed criteria for inherent prior model interpretability (Sect. 4) to define 6 Dirac (binary) measures for SVM (Table 3) meeting each criterion without overlap, except for criterion d (since all SVM kernels are generalized linear models).

We define an overall measure as follows:

$$\check{U}_\theta = 1/6 (\partial_{\text{essep}} + \partial_{\text{fin}} + \partial_{\text{eM}} + \partial_x + \partial_{\text{uni}} + \partial_{\text{adm}})$$

**Table 2.** We identify criteria for model interpretability in the literature and translate these into proposed criteria which are objective rather than subjective.

Term	Criteria in the literature	ID	Proposed criteria
Interpretable [34] Decomposable [30]	Each calculation has an intuitive explanation [30]	(a)	The feature space is known/explicit
		(b)	The feature space has a finite number of dimensions
	Inputs are interpretable, not anonymous or highly-engineered [30]. Generalized additive models are interpretable [34]	(c)	The model is generalized additive <i>with</i> <sup>a</sup> known/explicit basis/shape functions
	Generalized linear models are interpretable [34]. The contributions of individual features in the model, are understandable [34]	(d)	The model is generalized linear [34]
		(e)	The model is multiplicative, e.g., probabilistic, <i>with</i> known/explicit basis/shape functions
N/A	(f)	Model parts are uniform in function	
Transparent algorithm [30]	The training algorithm converges to a unique solution [30]	(g)	Model weights are learned by convex optimization or direct computation

<sup>a</sup>Note: Unlike functions of a single variable, basis/shape functions are only available if the kernel is separable.

A benefit of this measure is that while independent of the data, it requires little computation and it informs model selection prior to optimization.

**Table 3.** For kernel methods, e.g., SVM, we propose the following Dirac (binary) measures  $\partial$  of model transparency  $T$ . Let  $\mathcal{X}_T$  be the space of transparent features derived from simple transforms of the original features  $\mathcal{X}$  which are not highly engineered: i.e., given data  $\mathcal{X} = \{x\}$ , let  $\mathcal{X}_T = \{x, -x, \frac{1}{x}, \log(x), \tanh(x), \min(c_{\text{top}}, x), \max(c_{\text{bottom}}, x)\}$ .

Name of measure and criterion met	Symbol for measure	Conditions for measure to be true
Explicit symmetric separable (a)	$\partial_{\text{essep}}$	$k(\underline{x}, \underline{z}) = \phi(\underline{x})\phi(\underline{z})$ , $\phi$ known $x_i, z_i \in \mathcal{X}_0$ , $\mathcal{X}_0 \subseteq \mathcal{X}_T$ , $\phi \in \mathcal{F}$ , $\phi: \mathbb{R}^n \rightarrow \mathbb{R}$
Finite (b)	$\partial_{\text{fin}}$	$\dim(\mathcal{F}) < \infty$
Explicit Mercer (c)	$\partial_{\text{eM}}$	$k(\underline{x}, \underline{z}) = \phi(\underline{x})^T \phi(\underline{z})$ . $= \sum_q \phi_q(x_q)\phi_q(z_q)$ , $\phi_q$ known $x_i, z_i \in \mathcal{X}_0$ , $\mathcal{X}_0 \subseteq \mathcal{X}_T$ , $\phi_q \in \mathcal{F}$ , $\phi_q: \mathbb{R} \rightarrow \mathbb{R}$
Explicit multiplicative (e)	$\partial_{\times}$	$k(\underline{x}, \underline{z}) = \prod_q \phi_q(x_q)\phi_q(z_q)$ , $\phi_q$ known $x_i, z_i \in \mathcal{X}_0$ , $\mathcal{X}_0 \subseteq \mathcal{X}_T$ , $\phi_q \in \mathcal{F}$ , $\phi_q: \mathbb{R} \rightarrow \mathbb{R}$
Uniform (f)	$\partial_{\text{uni}}$	$\phi_q$ known and uniform e.g., (c) or (e) with $\phi_q = \phi \forall q$
Admissible (g)	$\partial_{\text{adm}}$	$k$ is positive definite (p.d.) [37] or $k$ is conditionally p.d. (c.p.d.) [8]

## 5 Creating More Measures Specific to SVM

In this section we propose measures specific to SVM.

**Support Vectors:** In SVM, a subset of the patients in the data set are key to defining the model. They are known as support vectors since they support the definition of the model’s class boundary and decision surface. For example, the decision regarding whether a patient has a disease or not, is determined by a subset of patients, e.g., 5 out of 200 patients, the model learned/picked as positive and negative examples of disease.

The more support vectors there are, the more complex the model is, with all other things being equal:  $H_{sv} = sv$ . SVM models have at least three support vectors in general—at least two to define the line, curve, hyperplane or surface that is the class boundary, and at least one to define the margin, so  $sv \geq 3$ ,  $sv \in \mathbb{N}$ .

To select a model for one data set, or to compare results between two data sets, we know the maximum number of patients  $N$ , so  $sv \leq N$ , and we apply (4)iii to obtain a relative measure,  $U_{sv,r}$ . Or to obtain an absolute measure  $U_{sv,a}$ , to compare against any current or future data set, we assume  $N = \infty$  and apply (4)ii.



**Degrees of Freedom:** Akaike includes all method and kernel hyperparameters and weights as among the degrees of freedom [50]. We calculate the prior complexity measure  $\check{H}_{dof}$  with three terms comprised of: the number of SVM hyperparameters, e.g., 1 for C, the number of kernel hyperparameters, e.g., 1 for the kernel width for a Gaussian RBF kernel, the number of independent inputs, e.g., 1 for a Gaussian RBF kernel or stationary kernel, 2 otherwise. We calculate the posterior complexity measure  $H_{dof}$  with an additional term for the support vectors and apply the general measure for model interpretability.

$$\begin{aligned}\check{H}_{dof} &= \check{dof} = d_{\text{SVM.hyp}} + d_{\text{kernel.hyp}} + d_{\text{input}} \\ H_{dof} &= dof = d_{\text{SVM.hyp}} + d_{\text{kernel.hyp}} + d_{\text{input}} + sv\end{aligned}$$

**Relevant Dimensionality Estimate:** The relevant dimensionality estimate (rde) [9] provides a way to measure the complexity of the SVM feature space induced by a kernel. There are two complexity measures  $H_{rdeT}$  and  $H_{rdeL}$  corresponding to two rde methods: the two-component model and the leave-one-out method, respectively.

## 6 Validation of Measures

We validate our proposed measures with sanity checks on formulas (not shown) and by agreement with propositions that describe our expectations and knowledge about model complexity and interpretability.

We create propositions based on expected relationships between measures, and check/test the propositions with a statement  $\mathbf{P}$  and its inverse  $\mathbf{P}^{-1}$  such as the following,

$$\mathbf{P}: \check{dof}_1 \leq \check{dof}_2 \xrightarrow{\text{usually}} U_{rde1}^* \geq U_{rde2}^* \quad (10)$$

$$\mathbf{P}^{-1}: \check{dof}_1 > \check{dof}_2 \xrightarrow{\text{usually}} U_{rde1}^* < U_{rde2}^* \quad (11)$$

where  $\xrightarrow{\text{usually}}$  is a notation that means “implies the majority of the time”. For brevity  $\mathbf{P}^{-1}$  is implied but not shown in statements that follow. We measure how much our results agree with these propositions using either Kendall’s W coefficient of rank correlation [26] or matched pair agreement [48], where the latter is applied to control for confounding factors.

If a proposition is robust, then the percentage of the concordance coefficient or matched pair agreement indicates how correct and useful the measure is, from that perspective. A measure has some utility, if it is correct the majority of the time, for different models/kernels and data sets, with a confidence interval that does not include 50%.

We validate our propositions using two types of experiments (#1 and #2 as below). We run each experiment five times on each of three data sets from the University of California at Irvine repository: the Statlog Heart, Hepatitis and Bupa Liver data sets. Missing data in the Hepatitis data set are imputed

with Stata, taking one of three multiple imputations with Monte Carlo Markov Chains. Bupa Liver is used with the common target [36] rather than the clinically meaningful target.

- Experiment Type #1: For each of 90 points chosen randomly in the hyperparameter space, we choose a pair of models, matched pairs [48], that differ by one hyperparameter/*dof* that is fixed in one and free in the other, and check propositions as the percentage truth of the propositions. We use 3 pairs of kernels that differ by a single *dof*, e.g., a polynomial kernel of varying degree versus a linear kernel, a Gaussian RBF kernel with/without a fixed kernel width and a Mercer sigmoid kernel [11] with/without a fixed horizontal shift.
- Experiment Type #2: From the experiment type #1 we identify three points in the hyperparameter space which perform well for each kernel. For each of 3 fixed points, we choose 30 values of  $C$  equally spaced (as logarithms) throughout the range from  $10^{-3}$  to  $10^6$  and check propositions as the concordance of the left-hand side with the right-hand side in the propositions, using Kendall's  $W$  coefficient of concordance. If the right-hand side should have opposite rank to the left-hand side then we apply a negative to the measure on the right-hand side for concordance to measure agreement of rank. We use the following kernels: linear, polynomial, Gaussian RBF and Mercer sigmoid kernel [11].

### 6.1 Propositions

**Proposition 1.** *The majority of the time we expect that a model with less degrees of freedom  $\check{dof}_1$ , with all other things being equal when compared to another model with  $\check{dof}_2$ , will be simpler and have a relevant dimensionality estimate (*rde*) [9] that is less than or equal to the other model and therefore be more interpretable/understandable ( $U_{rde}^*$ ):*

$$1a: \quad \check{dof}_1 \leq \check{dof}_2 \quad \xrightarrow{\text{usually}} \quad rde_1 \leq rde_2 \tag{12}$$

$$1b: \quad \check{dof}_1 \leq \check{dof}_2 \quad \xrightarrow{\text{usually}} \quad U_{rde1}^* \geq U_{rde2}^* \tag{13}$$

*This applies to *rde* with the two-component model (*rdeT*) and the leave-one-out method (*rdeL*).*

**Proposition 2.** *In SVM, the hyperparameter  $C$  is called the box constraint or cost of error. Authors have remarked [49, Remark 7.31] that  $C$  is not an intuitive parameter, although it has a lower bound for use  $C \geq \frac{1}{N}$  and its behaviour suggests  $C \doteq \frac{1}{\nu N}$ , where  $\nu$  is a proportion of support vectors. We therefore expect that a model with a higher value  $C_1$  versus a second model with  $C_2$  will have less support vectors (*sv*) and consequently be more interpretable/understandable*

$(U_{H_s})$ :

$$\mathbf{2a} : C_1 \geq C_2 \xrightarrow{\text{usually}} sv_1 \leq sv_2 \tag{14}$$

$$\mathbf{2b} : sv_1 \leq sv_2 \xrightarrow{\text{usually}} U_{H_{s1}} \geq U_{H_{s2}} \tag{15}$$

$$\mathbf{2c} : C_1 > C_2 \xrightarrow{\text{usually}} U_{sv,a1} \geq U_{sv,a2} \tag{16}$$

$$\mathbf{2d} : C_1 > C_2 \xrightarrow{\text{usually}} U_{H_{s1}} \geq U_{H_{s2}} \tag{17}$$

*This applies to simplicity of sensitivity  $U_{H_s}$  with any binning method.*

Our experiment uses three binning methods: Scott  $U_{H_{sc}}$ , Freedman-Diaconis  $U_{H_{fd}}$  and Sturges  $U_{H_{st}}$ .

**Proposition 3.** *The majority of the time we expect that, if a prior measure is useful, then it reflects the same rankings as the posterior measure,*

$$\mathbf{3} : U_{H_{s1}}^* \leq U_{H_{s2}}^* \xrightarrow{\text{usually}} U_{H_{s1}} \leq U_{H_{s2}} \tag{18}$$

**Proposition 4.** *We expect that the linear kernel is the simplest of all kernels with greater transparency than other kernels such as the polynomial, Gaussian RBF kernel, sigmoid and Mercer sigmoid kernels, whereby,*

$$\mathbf{4} : isLinear(k_1) > isLinear(k_2) \rightarrow \check{U}_{\partial 1} > \check{U}_{\partial 2} \tag{19}$$

## 7 Results

We summarize the results of our validation tests (Tables 4 and 5) as follows: we recommend  $\check{U}_{\partial}$  and  $U_{sv}$  as good measures. We find that  $U_{rdeT}^*$ ,  $U_{rdeL}^*$  and  $U_{H_{st}}$  are measures which are of limited use, because they may be wrong one third of the time when providing guidance on decisions.  $U_{H_{sc}}$  and  $U_{H_{fd}}$  are not distinguished from chance by our propositions and are therefore not recommended. If  $U_{H_{st}}$  is validated to a greater degree in the future, then the initial measure  $U_{H_{st}}^*$  has been shown to be a good proxy for it, incurring some loss of information (Table 5).

Our proposed measure of kernel transparency  $\check{U}_{\partial}$ , a prior measure, scored 100% agreement. This is a good measure that may be used a priori, but it is high-level and not specific to the match between a model and data. No surprises or complexities arose regarding the attributes of kernels.

The general measure based on the number of support vectors,  $U_{sv}$ , scored  $81 \pm 2.3\%$  agreement—this is a good measure.

Our proposed simplicity of sensitivity measure with Sturges binning  $U_{H_{st}}$  scored  $64 \pm 3.2\%$  and  $62 \pm 3.5\%$ , which is of limited use—we are interested in agreement that is sufficiently greater than chance (50%), enough to be reliable.

The same measure with Scott binning ( $U_{H_{sc}}$ ), however, is barely distinguishable from chance in one test, and not distinguishable in another, and with Freedman-Diaconis binning ( $U_{H_{fd}}$ ) it is not distinguishable from chance in both

tests. We recommend further validation to examine the role of confounding factors such as kernel width/scale along with  $C$  per [6, 16].

If the simplicity of sensitivity measure  $U_{Hst}$  can be validated to a greater degree in the future, then the initial measure  $U_{Hst}^*$  which scores  $80 \pm 3.2\%$  agreement with it, may be used in its place to avoid optimization, or to gain an initial estimate prior to optimization.

The general measure based on the relevant dimensionality of the feature space,  $U_{rdeT}^*$  and  $U_{rdeL}^*$  scored  $62 \pm 5.0\%$  and  $59 \pm 5.2\%$  agreement, respectively. These are of some use. We did not include Braun’s noise estimate, which in hindsight should improve the measure.

## 8 Application

We apply model interpretability to results in a toy problem. When we select results for maximum accuracy with the Gaussian RBF kernel, we find that the top result in our sorted list of results achieves 100% accuracy (rounded to no decimal places) with 51 support vectors, while the second best result also achieves 100% accuracy with 40 support vectors and the fifth best result according to the list also achieves 100% accuracy with 25 support vectors.

Selecting results for maximum interpretability  $U_{sv,r}$ , we find the top result uses 9 support vectors for 99% accuracy and the fourth best result uses 10 support vectors for the same accuracy.

We plot the results (Fig. 3) of accuracy versus interpretability  $U_{sv,r}$  (above 80% in each) and find that there are many results which are highly accurate and highly interpretable, i.e., above 96% in both. These results indicate that there is not a trade-off between accuracy and model interpretability based on support vectors in this data set.

We also plot the results of accuracy versus interpretability  $U_{sv,r}$  for other data sets (Figs. 4 and 5) and it is clear that there is no trend in all points showing a trade-off between accuracy and model interpretability, although this

**Table 4.** The results from propositions using experiment type #2 validate the support vector measure  $U_{sv}$  and simplicity of sensitivity measure with Sturges binning  $U_{Hst}$ .

Proposition	Measure & Result	Agreement %	Comment
2a	$sv$	$82 \pm 2.3$	$C$ validates $sv$ , supports B3
2b	$U_{Hsc}$	$53 \pm 3.3$	$U_{Hsc}$ not distinguished by $sv$
	$U_{Hfd}$	$48 \pm 3.7$	$U_{Hfd}$ not distinguished by $sv$
	$U_{Hst}$	$62 \pm 3.5$	$sv$ validates $U_{Hst}$
2c	$U_{sv}$	$81 \pm 2.3$	$C$ validates $U_{sv}$
2d	$U_{Hsc}$	$54 \pm 3.3$	$C$ validates $U_{Hsc}$
	$U_{Hfd}$	$49 \pm 3.7$	$U_{Hfd}$ not distinguished by $sv$
	$U_{Hst}$	$64 \pm 3.2$	$C$ validates $U_{Hst}$

Legend: Green = affirmative result. Yellow = inconclusive result. Red = contrary result.

**Table 5.** The results from propositions using experiment #1 validate the relevant dimensionality measures  $rdeT$  and  $rdeL$ , the initial model interpretability measures based on relevant dimensionality  $U_{rdeT}^*$  and  $U_{rdeL}^*$ , the use of prior measures of simplicity of sensitivity as proxies for posterior measures, and the measure of kernel transparency  $\tilde{U}_\partial$ .

Proposition	Measure & Result	Agreement %	Comment
1a	$rdeT$	$63 \pm 5.0$	$\dot{dof}$ validates $rdeT$ , supports A2
	$rdeL$	$59 \pm 5.2$	$\dot{dof}$ validates $rdeL$ , supports A2
1b	$U_{rdeT}^*$	$62 \pm 5.0$	$\dot{dof}$ validates $U_{rdeT}^*$
	$U_{rdeL}^*$	$59 \pm 5.2$	$\dot{dof}$ validates $U_{rdeL}^*$
3	$U_{Hsc}^*$ as a proxy	$72 \pm 3.1$	$U_{Hsc}$ validates $U_{Hsc}^*$ as a proxy
	$U_{Hfd}^*$ as a proxy	$76 \pm 3.5$	$U_{Hfd}$ validates $U_{Hfd}^*$ as a proxy
	$U_{Hst}^*$ as a proxy	$80 \pm 3.2$	$U_{Hst}$ validates $U_{Hst}^*$ as a proxy
4	$\tilde{U}_\partial$	$100 \pm 0$	$k_{Lin}$ vs. others, validates $\tilde{U}_\partial$

Legend: Green = affirmative result. Yellow = inconclusive result. Red = contrary result.

**Table 6.** Result for  $\tilde{U}_\partial$  confirm that the linear kernel is more transparent than other kernels.

Dirac measure	Linear	Polynomial	Gaussian RBF	Sigmoid	Mercer Sigmoid
$\partial_{essep}$	✓	×	×	×	×
$\partial_{fin}$	✓	✓	×	×	✓
$\partial_{eM}$	✓	×	✓ [13]	×	✓
$\partial_\times$	×	×	×	×	×
$\partial_{uni}$	✓	×	×	×	✓
$\partial_{adm}$	✓	✓	✓	×	✓
$\tilde{U}_\partial$ (%)	83	33	33	0	67

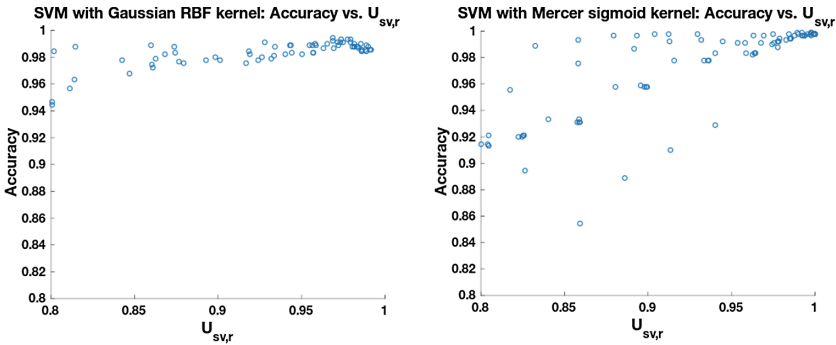
Legend: Green = top result. Light green = second best result.

trend may be present at the pareto front. A trade-off trend would show as an inverse correlation, a trend line running from the top left to the bottom right—instead, high interpretability is consistently achievable with high accuracy, i.e., there are points toward the top right of a bounding box for all points.

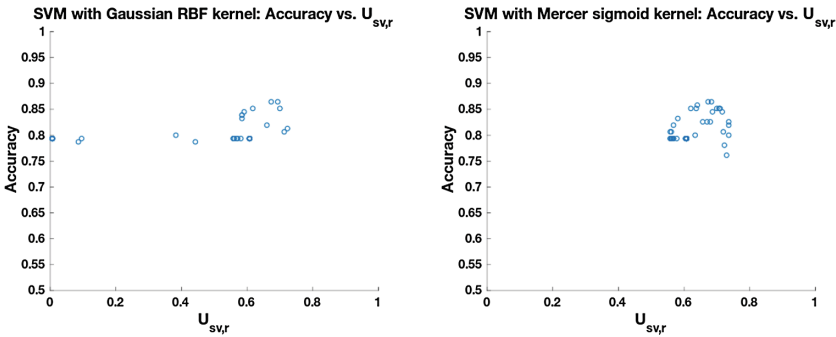
## 9 Related Work

Lipton [30] provides a good taxonomy for **model interpretability** with concepts falling into two broad categories: transparency (the opposite of a black box) and post-hoc interpretability.

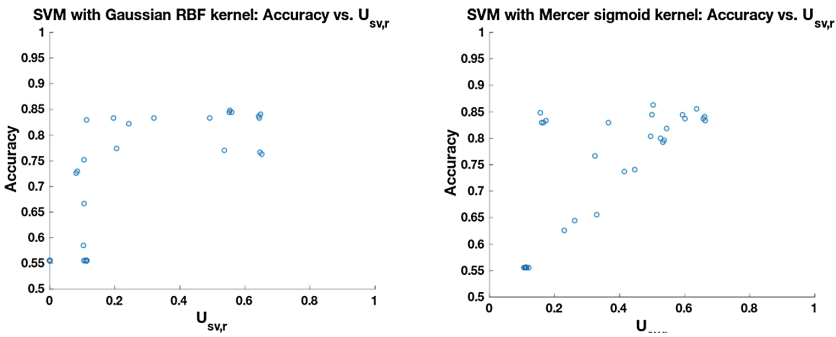
**Post-hoc interpretability** involves an explanatory model separate from the predictive model, or visuals that transform data where the transformation is also a separate explanatory model. Liang [28] cautions against explaining a black box predictive model with another black box explanatory model.



**Fig. 3.** In classification for the toy problem, there are many results with high accuracy and high model interpretability, with almost no sacrifice in the latter for maximum accuracy.



**Fig. 4.** In classification with the Hepatitis data set there is a less than 5% sacrifice in interpretability for the highest accuracy.



**Fig. 5.** In classification with Statlog Heart data there are points with high accuracy and interpretability, with minimal sacrifice, 1% and 2%, respectively.

Riberio et al. [46] create an external **local linear model** to approximate the prediction model in a post-hoc approach called LIME. They jointly optimize accuracy and model complexity but they do not elucidate much about model complexity as in our work. LIME perturbs features in a separate binary representation of features, which sometimes map to **non-local** features in the original space of data. In their examples they use the binary model output, only referring in passing to the possibility of using a continuous output for classifiers, as we do.

**Transparency**, on the other hand, focuses on the predictive model itself, and has three aspects: decomposability, simulatability and algorithmic transparency [30].

**Decomposability** refers to being able to see and understand the parts of the model of the model, e.g., kernels and parameters and the parts of the data, i.e., features and instances—and how they contribute to a result from the predictive model. Some authors refer to the output from decomposition as an **interpretation**, e.g., initial understanding, separate from an **explanation** [24, 39] that may require **analysis**, **selection** or perhaps **synthesis**. Miller adds that explanations are **selected** and **social** [38].

Since the social and synthesis tasks are more suitable to a person than a computer—it is reasonable for our work to focus on inherent measures of interpretability, rather than explanations.

[34] express that some types of models are more **intelligible** (i.e., decomposable) than others. We include categories for generalized linear and generalized additive models in our measures as a result of their work.

**Simulatability**, as another aspect of transparency, refers to a model that a person can mentally simulate or manually compute in reasonable time [30] and is correlated, for example, with the number of features in a linear model, or the depth of the tree in a decision tree. **Model complexity** is implied Lipton’s examples but the term is not invoked although other authors refer to it [10, 35, 42].

**Ockham’s razor**, also called the principle of **parsimony** [50], is a well known principle related to model complexity. Regarding models, it says that among sufficient explanations (e.g., equally accurate<sup>1</sup> models), the simplest<sup>2</sup> should be preferred. A quick note on sufficiency: for multiple equally accurate models, none are necessary, because any one of them is sufficient. Model accuracy is sought first, then simplicity. Using our proposed measure one can search for the model with highest interpretability among equally accurate models.

Backhaus et al. propose a quantitative measure of model interpretability [3]—but that is for a different meaning or definition—the ability for a model to interpret data, with relevance in relevance vector machines as the context.

Related to our work, **sensitivity analysis of model outputs** (SAMO) [2, 23] describe how sensitive a model output is to a change in feature values, one at a time—which is the approach of our proposed general measure.

<sup>1</sup> Where accuracy cannot be distinguished with statistical significance.

<sup>2</sup> [Sober] refers to [Akaike]’s definition of the simplest model as the model with the least degrees of freedom, i.e., least number of (independent) coefficients.

In variance-based sensitivity analysis, Sobol [51] finds the variance in the output explained by an input feature. Liu et al. [32] performs entropy-based sensitivity analysis, called global response probabilistic sensitivity analysis (GRPSA), to find the influence of input features—where entropy is used to compute the effect as information loss. Lemaire *et al.* [27] apply sensitivity analysis but their perturbations are non-local and could easily create points outside of any known clusters of instances and true states of nature. Poulin *et al.* [43] provides effective visualization and analysis tools but for SVM they only apply their method to linear SVM and its binary output.

**Automatic model selection** methods have been proposed for accuracy [1, 40]—these are based on rules computed from many data sets. The rule-based approach is brittle in comparison to our measures, since it only works with a fixed set of candidate kernels.

## 10 Conclusions

We developed and validated measures for inherent model interpretability to enable automatic model selection and ongoing research. Two measures are recommended: our proposed kernel transparency measure  $\tilde{U}_\partial$  which is an inexpensive prior measure, and a posterior measure based on support vectors  $U_{sv}$ . Three other measures,  $U_{rdeT}^*$ ,  $U_{rdeL}^*$  and  $U_{Hst}$  were found to be of limited use but may be further validated by future work.

We also contributed ideas as a foundation for these measures: the concept of inherent model interpretability, a general measure, a simplicity of sensitivity measure, and measurement of interpretability at different points in the learning process, i.e., via prior, initial and posterior models.

We applied our measure to model selection and demonstrated that choosing a model based on a sorted list of accuracy alone can result in models with substantively less inherent model interpretability despite the consistent availability of models with high accuracy and high interpretability in multiple data sets. The notion of a trade-off between accuracy and interpretability does not hold for these data sets.

## A Appendix: Treating Features of Any Atomic Data Type as Continuous

Assuming that we are not given a fixed pre-trained model, but can instead the machine learning method and model, we can select one that handles continuous values, and we can treat features of any atomic data type (defined below) as continuous. This treatment requires three steps—and most of the content in these steps are standard practice, with a few exceptions denoted by an asterix\*.



**Table 7.** Atomic data types are based on Steven’s scales of measurement

Atomic data type	Steven’s scale	Summary of key attributes			
		Continuous	Discrete	Ordered	Fixed zero
Real	Ratio	✓		✓	✓
Integer	Ratio		✓	✓	✓
Datetime	Interval	✓		✓	
Date	Interval		✓	✓	
Ordinal	Ordinal		✓	✓	
Binary	Nominal		✓		
Nominal	Nominal		✓		

We define **atomic data types** (Table 7) as the following set of data types which are fundamental building blocks for all electronic data<sup>3</sup>: reals, integers, datetimes, dates, ordinals, binary and nominals. These atomic data types are based on Steven’s scales of measurement [52], but are specified at a level that is more interpretable and useful.

Although binary values may also be considered nominals, we identify them separately because there are methods in the literature specific to binary data (e.g., for imputation and similarity measurement) and the data type is specifically defined in programming languages, machine learning platforms, database schema and data extraction tools.

1. Treat missing data. Assuming data are missing completely at random (MCAR) do the following, otherwise refer to [58].
  - (a) Impute missing data for reals, integers, datetimes, dates and ordinals, using whichever method meets requirements—e.g., multiple imputation with Monte Carlo Markov chain, expectation maximization, hot-deck imputation or mean imputation.
  - (b) Impute missing data for nominals using the mode, i.e., the most frequent level.
  - (c) Impute missing binary data with a method that will produce continuous values and which is appropriate for binary distributions—e.g., multiple imputation or expectation maximization. We refer to the output as continuously-imputed binary data.
2. Convert nominals to binary indicators, one for each level.
3. Center and normalize data
  - (a) For continuously-imputed binary data, bottom-code and top-code the data to the limits, then min-max normalize the data to the range  $[-1, +1]$  for SVM or  $[0, 1]$  for neural networks and logistic regression.

<sup>3</sup> E.g., a combination of atomic data types can make up a complex data type—e.g., a combination of letters or symbols (nominals) make up a string as a complex data type.

- (b) For binary data, min-max normalize the data to the set  $\{-1, +1\}$  for SVM or  $\{0, 1\}$  for neural networks and logistic regression. This data will be treated as reals by the methods/models, but  $\{-1, +1\}$  makes more sensible use of the symmetric kernel geometry in SVM than  $\{0, 1\}$ .
- (c) For all other data types, center and normalize each feature using z-score normalization (or scalar variations based on 2 or 3 sigma instead of 1 sigma).

Now all of the data are ready to be treated as reals by the methods/models.

## References

1. Ali, S., Smith, K.A.: On learning algorithm selection for classification. *Appl. Soft Comput.* **6**(2), 119–138 (2006)
2. Auder, B., Iooss, B.: Global sensitivity analysis based on entropy. In: Proceedings of the ESREL 2008 Safety, reliability and risk analysis Conference, pp. 2107–2115 (2008)
3. Backhaus, A., Seiffert, U.: Quantitative measurements of model interpretability for the analysis of spectral data. In: IEEE Symposium on Computational Intelligence and Data Mining (CIDM), pp. 18–25. IEEE (2013)
4. Badii, R., Politi, A.: Complexity: Hierarchical Structures and Scaling in Physics, vol. 6. Cambridge University Press, Cambridge (1999)
5. Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., MÄzler, K.-R.: How to explain individual classification decisions. *J. Mach. Learn. Res.* **11**, 1803–1831 (2010)
6. Ben-Hur, A., Weston, J.: A user’s guide to support vector machines. In: *Data Mining Techniques for the Life Sciences*, pp. 223–239. Springer (2010)
7. Berner, E.S.: *Clinical Decision Support Systems*. Springer, New York (2007). <https://doi.org/10.1007/978-0-387-38319-4>
8. Boughorbel, S., Tarel, J.-P., Boujemaa, N.: Conditionally positive definite kernels for SVM based image recognition. In: IEEE International Conference on Multimedia and Expo, ICME 2005, pp. 113–116. IEEE (2005)
9. Braun, M.L., Buhmann, J.M., MÄzler, K.-R.: On relevant dimensions in kernel feature spaces. *J. Mach. Learn. Res.* **9**, 1875–1908 (2008)
10. Breiman, L.: Statistical modeling: the two cultures (with comments and a rejoinder by the author). *Stat. Sci.* **16**(3), 199–231 (2001)
11. Carrington, A.M., Fieguth, P.W., Chen, H.H.: A new mercer sigmoid kernel for clinical data classification. In: 36th Annual International Conference on Engineering in Medicine and Biology Society (EMBC), pp. 6397–6401. IEEE (2014)
12. Caruana, R., Niculescu-Mizil, A.: An empirical comparison of supervised learning algorithms. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 161–168. ACM (2006)
13. Cotter, A., Keshet, J., Srebro, N.: Explicit approximations of the Gaussian kernel. arXiv preprint [arXiv:1109.4603](https://arxiv.org/abs/1109.4603) (2011)
14. Cover, T.M., Thomas, J.A.: *Elements of Information Theory*. Wiley, Hoboken (2012)

15. Cruz, J.A., Wishart, D.S.: Applications of machine learning in cancer prediction and prognosis. *Cancer Inform.* **2**, 59–78 (2006)
16. Devos, O., Ruckebusch, C., Durand, A., Duponchel, L., Huvenne, J.-P.: Support vector machines (SVM) in near infrared (NIR) spectroscopy: focus on parameters optimization and model interpretation. *Chemom. Intell. Lab. Syst.* **96**(1), 27–33 (2009)
17. Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning (2017)
18. Freedman, D., Diaconis, P.: On the histogram as a density estimator:  $L_2$  theory. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete* **57**(4), 453–476 (1981)
19. Gell-Mann, M., Lloyd, S.: Information measures, effective complexity, and total information. *Complexity* **2**(1), 44–52 (1996)
20. Goodman, B., Flaxman, S.: European union regulations on algorithmic decision-making and a “right to explanation”. In: 1st Workshop on Human Interpretability in Machine Learning, International Conference of Machine Learning (2016)
21. Goodstein, D.L., Goodstein, J.R.: Feynman’s Lost Lecture: The Motion of Planets Around the Sun, vol. 1. W. W. Norton & Company, New York (1996)
22. Greenes, R.A.: *Clinical Decision Support: The Road Ahead*. Academic Press, SanDiego (2011)
23. Hanson, K.M., Hemez, F.M.: Sensitivity analysis of model output. In: Proceedings of the 4th International Conference on Sensitivity Analysis of Model Output (SAMO 2004), Santa Fe, 8–11 March 2004. Los Alamos National Laboratory (2005)
24. Holzinger, A., Biemann, C., Pattichis, C.S., Kell, D.B.: What do we need to build explainable AI systems for the medical domain? arXiv preprint [arXiv:1712.09923](https://arxiv.org/abs/1712.09923) (2017)
25. Jernigan, M.E., Fieguth, P.: *Introduction to Pattern Recognition*. University of Waterloo (2004)
26. Kendall, M.G.: The treatment of ties in ranking problems. *Biometrika* **33**(3), 239–251 (1945)
27. Lemaire, V., Féraud, R., Voisine, N.: Contact personalization using a score understanding method. In: IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), IJCNN 2008, pp. 649–654. IEEE (2008)
28. Liang, P.: Provenance and contracts in machine learning. In: Proceedings of the 2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016) (2016)
29. Lin, D.: An information-theoretic definition of similarity. *ICML* **98**, 296–304 (1998)
30. Lipton, Z.C., et al.: The mythos of model interpretability. In: *IEEE Spectrum* (2016)
31. Lisboa, P.J.G.: Interpretability in machine learning – principles and practice. In: Masulli, F., Pasi, G., Yager, R. (eds.) *WILF 2013. LNCS (LNAI)*, vol. 8256, pp. 15–21. Springer, Cham (2013). [https://doi.org/10.1007/978-3-319-03200-9\\_2](https://doi.org/10.1007/978-3-319-03200-9_2)
32. Liu, H., Chen, W., Sudjianto, A.: Relative entropy based method for probabilistic sensitivity analysis in engineering design. *J. Mech. Des.* **128**(2), 326–336 (2006)
33. Lloyd, S.: Measures of complexity: a nonexhaustive list. *IEEE Control Syst. Mag.* **21**(4), 7–8 (2001)

34. Lou, Y., Caruana, R., Gehrke, J.: Intelligible models for classification and regression. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 150–158. ACM (2012)
35. Martens, D., Baesens, B.: Building acceptable classification models. In: Stahlbock, R., Crone, S., Lessmann, S. (eds.) *Data Mining. Annals of Information Systems*, pp. 53–74. Springer, Boston (2010). [https://doi.org/10.1007/978-1-4419-1280-0\\_3](https://doi.org/10.1007/978-1-4419-1280-0_3)
36. McDermott, J., Forsyth, R.S.: Diagnosing a disorder in a classification benchmark. *Pattern Recognit. Lett.* **73**, 41–43 (2016)
37. Mercer, J.: Functions of positive and negative type, and their connection with the theory of integral equations. *Philos. Trans. R. Soc. Lond. Ser. A* **209**, 415–446 (1909). Containing papers of a mathematical or physical character
38. Miller, T., Howe, P., Sonenberg, L.: Explainable AI: beware of inmates running the asylum. In: *IJCAI-17 Workshop on Explainable AI (XAI)*, p. 36 (2017)
39. Montavon, G., Lapuschkin, S., Binder, A., Samek, W., Müller, K.-R.: Explaining nonlinear classification decisions with deep Taylor decomposition. *Pattern Recognit.* **65**, 211–222 (2017)
40. Nahar, J., Ali, S., Chen, Y.-P.P.: Microarray data classification using automatic SVM kernel selection. *DNA Cell Biol.* **26**(10), 707–712 (2007)
41. Olson, R.S., La Cava, W., Orzechowski, P., Urbanowicz, R.J., Moore, J.H.: PMLB: a large benchmark suite for machine learning evaluation and comparison. *BioData Min.* **10**(1), 36 (2017)
42. Perez, P.S., Nozawa, S.R., Macedo, A.A., Baranauskas, J.A.: Windowing improvements towards more comprehensible models. *Knowl. Based Syst.* **92**, 9–22 (2016)
43. Poulin, B., et al.: Visual explanation of evidence with additive classifiers. In: *Proceedings of the National Conference On Artificial Intelligence*, vol. 21, p. 1822. AAAI Press, Menlo Park (1999). MIT Press, Cambridge (2006)
44. Pusic, M.V., Boutis, K., Hatala, R., Cook, D.A.: Learning curves in health professions education. *Acad. Med.* **90**(8), 1034–1042 (2015)
45. Rényi, A., et al.: On measures of entropy and information. In: *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*. The Regents of the University of California (1961)
46. Ribeiro, M.T., Singh, S., Guestrin, C.: Why should i trust you? Explaining the predictions of any classifier. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144. ACM (2016)
47. Scott, D.W.: On optimal and data-based histograms. *Biometrika* **66**(3), 605–610 (1979)
48. Selvin, S.: *Statistical Analysis of Epidemiologic Data*. Oxford University Press, New York (2004)
49. Shawe-Taylor, J., Cristianini, N.: *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York (2004)
50. Sober, E.: Parsimony and predictive equivalence. *Erkenntnis* **44**(2), 167–197 (1996)
51. Sobol, I.M.: Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Math. Comput. Simul.* **55**(1), 271–280 (2001)
52. Stevens, S.S.: *On the theory of scales of measurement* (1946)
53. Sturges, H.A.: The choice of a class interval. *J. Am. Stat. Assoc.* **21**(153), 65–66 (1926)
54. Szabó, Z., Póczos, B., Lőrincz, A.: Undercomplete blind subspace deconvolution. *J. Mach. Learn. Res.* **8**, 1063–1095 (2007)

55. Szabó, Z., Póczos, B., Lőrincz, A.: Separation theorem for independent subspace analysis and its consequences. *Pattern Recognit.* **45**, 1782–1791 (2012)
56. Tsallis, C.: Possible generalization of Boltzmann-Gibbs statistics. *J. Stat. Phys.* **52**(1), 479–487 (1988)
57. Tussy, A., Gustafson, R.: *Elementary Algebra*. Nelson Education (2012)
58. Donders, A.R.T., Van Der Heijden, G.J.M.G., Stijnen, T., Moons, K.G.M.: A gentle introduction to imputation of missing values. *J. clin. epidemiol.* **59**(10), 1087–1091 (2006). Elsevier



# Regular Inference on Artificial Neural Networks

Franz Mayr<sup>(✉)</sup> and Sergio Yovine<sup>(✉)</sup>

Universidad ORT Uruguay, Montevideo, Uruguay  
{mayr,yovine}@ort.edu.uy

**Abstract.** This paper explores the general problem of explaining the behavior of artificial neural networks (ANN). The goal is to construct a representation which enhances human understanding of an ANN as a sequence classifier, with the purpose of providing insight on the rationale behind the classification of a sequence as positive or negative, but also to enable performing further analyses, such as automata-theoretic formal verification. In particular, a probabilistic algorithm for constructing a deterministic finite automaton which is approximately correct with respect to an artificial neural network is proposed.

**Keywords:** Artificial neural networks · Sequence classification  
Deterministic finite automata  
Probably Approximately Correct learning

## 1 Introduction

The purpose of explainable artificial intelligence is to come up with artifacts capable of producing intelligent outcomes together with appropriate rationalizations of them. It means that besides delivering best possible model performance metrics (e.g., accuracy) and computational performance metrics (e.g., algorithmic complexity), they must provide adequate and convincing reasons for effectively justifying the judgment in a human-understandable way.

Artificial neural networks (ANN) are the state-of-the-art method for many fields in the area of artificial intelligence [20]. However, ANN are considered to be a rather obscure model [21], meaning that understanding the specifics that were taken into consideration by the model to make a decision is not a trivial task. Human understanding of the model is crucial in fields such as medicine [18], risk assessment [4], or intrusion detection [33]. From the point of view of explaining the rationale of an outcome, an important issue is that ANN lack an explicit and constructive characterization of their embedded decision-making strategy.

This limitation of ANN explanatory capabilities motivated a large amount of research work aiming at improving ANN explainability. Several approaches to tackle this issue have been identified [10, 14]. In particular, [14] characterizes the *black-box model* explanation problem. It consists in providing a human-understandable model which is able to mimic the behavior of the ANN. In [10],

the problem of *processing* explanation is defined. This approach seeks answering *why* a given input leads the ANN to produce a particular outcome. Another approach consists in allowing a human actor to interact with the learning process. This human-in-the-loop method of addressing explainability, called *glass box* interactive machine-learning approach, is presented in [19].

In this paper we follow a black-box model and processing explanation approach for ANN. We are interested in studying explainability in the context of ANN trained to solve *sequence* classification problems [32, 34], which appear in many application domains. In the past few years several classes of ANN, such as Recurrent Neural Networks (RNN), e.g., Long-Short Term Memory (LSTM) [17], have been successfully applied for such matter [6, 8, 22, 27, 28, 30, 31, 35].

We restrict the study to binary classification. This problem is a case of *language membership*, where the language of the ANN is the set of sequences classified as positive by the network. The trained ANN hides a model of such sequences which it uses to predict whether a given input sequence belongs to the language. If the language from which the training samples have been drawn is known, the question is how well the ANN learned it [8, 9, 26]. Another, may be more realistic situation occurs when the target language is unknown. In this case, the question to answer becomes what is the language learned by the ANN, or more precisely, whether it could be characterized operationally instead of denotationally.

Typically, these questions are addressed by looking at the accuracy of the network on a given test set. However, it has been observed that networks trained with millions of samples which exhibit 100% accuracy on very large development test sets could still incorrectly classify random sequences [12, 31]. Thus, exact convergence on a set of sequences, whatever its size, does not ensure the language of the network is the same as the target language. Hence, in both cases, the question remains whether the language recognized by the network could be explained, even approximately, in some other, comprehensible, way.

The goal of this paper is to provide means for extracting a constructive representation of the model hidden inside the ANN. To the best of our knowledge, all previous works devoted to model and processing explanation for ANN are either white-box, that is, they look into and/or make assumptions about the network's structure and state, or they are focused on extracting decision trees or rules. The reader is referred to [3, 10, 14] for recent surveys on this topic.

Now, when it comes to operationally explain the dynamical system that produces sequences of events, rules and trees are not expressive enough. In this case, a widely used formalism are automata [13]. This model provides a language-independent mathematical support for studying dynamical systems whose behavior could be understood as sequences corresponding to words of a regular language. In the automata-theoretic approach, when the system under analysis is a black box, that is, its internal structure (composed of states and transitions) is unknown, the general problem of constructing an automaton that behaves as the black box is called *identification* or *regular inference* [16].

Many times it is not theoretically or practically feasible to solve this problem precisely, in which case, it needs to be solved approximately. That is, rather than exactly identifying the automaton inside the black box, we attempt to find an automaton which is a reasonable approximation with some confidence.

The Probably Approximately Correct (PAC) framework [29] is a general approach to solve problems like the one we are considering here. A *learner*, which attempts to identify the hidden machine inside the black box, can interact with a *teacher*, which has the ability to answer queries about the unknown machine to be learned. For this, the teacher uses an oracle which draws positive and negative samples with some probability distribution. There are several specific problem instances and algorithms to solve them, depending on the assumptions that are made regarding how the behavior of the black box is observed, what questions could be asked, how the answers to these questions could be used to build an automaton, etc. The reader is referred to [2, 16] for a thorough review.

In this context, two general settings can be distinguished, namely *passive* or *active* learning. The former consists in learning a language from a set of given (chosen by the teacher) positive and/or negative examples [25]. It has been shown in [11] that this problem is *NP-complete*. In the latter, the learner is given the ability to draw examples and to ask membership queries to the teacher. A well known algorithm in this category is Angluin's  $L^*$  [1].  $L^*$  is *polynomial* on the number of states of the minimal deterministic finite automaton (DFA) and the maximum length of any sequence exhibited by the teacher.

The relationship between automata and ANN has been thoroughly studied: [26] presents mechanisms for programming an ANN that can correctly classify strings of arbitrary length belonging to a given regular language; [9] discuss an algorithm for extracting the finite state automaton of second-order recurrent neural networks; [31] look at this problem in a white-box setting. Therefore, according to [14], a black-box model explanation approach, that is, using regular inference algorithms that do not rely on the ANN structure and weights is a problem that has not been addressed so far.

Of course, one may argue that an automaton could be directly learned from the dataset used to train the network. However, this approach has several drawbacks. First, passive learning is NP-complete [11]. Second, it has been shown that ANN such as LSTM, are much better learners, as they learn faster and generalize better. Besides, they are able to learn languages beyond regular ones. Third, the training dataset may not be available, in which case the only way to construct an explanation is to query the ANN.

The contribution of this work is an adaptation of Angluin's  $L^*$  algorithm that outputs a DFA which approximately behaves like an input ANN whose actual structure is completely unknown. This means that whenever a sequence is recognized by the DFA constructed by the algorithm, it will most likely be classified as positive by the ANN, and vice versa. We stress the fact that our algorithm is completely agnostic of the structure of the ANN.

*Outline.* In Sect. 2 we precisely present the problem we are going to address and the method used for solving it. In Sect. 3 we discuss the proposed algorithm and a variation of the general PAC framework to analyze its behavior. In Sect. 4 we present the experimental results carried out on several examples which validate the theoretical analyses. In Sect. 5 we compare and contrast our approach with related works. Finally we present the conclusions and future work.



## 2 Preliminaries

### 2.1 Problem Statement

Let of  $\mathcal{U} \subseteq \Sigma^*$  be some *unknown* language over an alphabet  $\Sigma$  of symbols, and  $\mathcal{S} \subseteq \Sigma^*$  be a sample set such that it contains positive and negative sequences of  $\mathcal{U}$ . That is, there are sequences in  $\mathcal{S}$  which belong to  $\mathcal{U}$  and others that do not.

The *language* of an ANN  $\mathcal{N}$ , denoted  $\mathcal{L}(\mathcal{N})$ , is the set of sequences classified as positive by  $\mathcal{N}$ . Suppose  $\mathcal{N}$  is obtained by training it with a sample set  $\mathcal{S}$ , and then used to predict whether a sequence  $u \in \Sigma^*$  does belong to  $\mathcal{U}$ . In other words, the unknown language  $\mathcal{U}$  is considered to be somehow approximated by  $\mathcal{L}(\mathcal{N})$ , that is, with high probability  $x \in \mathcal{L}(\mathcal{N}) \iff x \in \mathcal{U}$ .

But, what is the actual language  $\mathcal{L}(\mathcal{N})$  learned by  $\mathcal{N}$ ? Is it a regular language? That is, could it be expressed by a deterministic finite automaton? Is it possible to approximate it somehow with a regular language? The interest of having an automaton-based, either precise or approximated, characterization of  $\mathcal{L}(\mathcal{N})$ , allows to explain the answers of  $\mathcal{N}$ , while providing insight on the unknown language  $\mathcal{U}$ . This approach enhances human understanding because of the visual representation but also because it enables performing further analyses, such as automata-theoretic formal verification [5].

### 2.2 Probably Approximately Correct (PAC) Learning

In order to study the questions above, we resort to Valiant’s PAC-learning framework [2,29]. Since we are interested in learning languages, we restrict ourselves to briefly describing the PAC-learning setting for languages.

Let  $\mathcal{D}$  be an unknown distribution over  $\Sigma^*$  and  $\mathcal{L}_1, \mathcal{L}_2 \subseteq \Sigma^*$ . The *symmetric difference* between  $\mathcal{L}_1$  and  $\mathcal{L}_2$ , denoted  $\mathcal{L}_1 \oplus \mathcal{L}_2$ , is the set of sequences that belong to only one of the languages, that is,  $\mathcal{L}_1 \oplus \mathcal{L}_2 = \mathcal{L}_1 \setminus \mathcal{L}_2 \cup \mathcal{L}_2 \setminus \mathcal{L}_1$ .

The *prediction error* of  $\mathcal{L}_1$  with respect to  $\mathcal{L}_2$  is the probability of a sequence to belong to their symmetric difference, denoted  $\mathbf{P}_{\mathcal{D}}(\mathcal{L}_1 \oplus \mathcal{L}_2)$ . Given  $\epsilon \in (0, 1)$ , we say that  $\mathcal{L}_1$  is  $\epsilon$ -*approximately correct* with respect to  $\mathcal{L}_2$  if  $\mathbf{P}_{\mathcal{D}}(\mathcal{L}_1 \oplus \mathcal{L}_2) < \epsilon$ .

The *oracle*  $\mathbf{EX}_{\mathcal{D}}(\mathcal{L}_1)$  draws an *example* sequence  $x \in \Sigma^*$  following distribution  $\mathcal{D}$ , and tags it as *positive* or *negative* according to whether it belongs to  $\mathcal{L}_1$  or not. Calls to  $\mathbf{EX}$  are independent of each other.

A *PAC-learning algorithm* takes as input an *approximation* parameter  $\epsilon \in (0, 1)$ , a *confidence* parameter  $\delta \in (0, 1)$ , *target* language  $\mathcal{L}_t$  and oracle  $\mathbf{EX}_{\mathcal{D}}(\mathcal{L}_t)$ , and if it terminates, it outputs a language  $\mathcal{L}_o$  such that  $\mathcal{L}_o$  is  $\epsilon$ -approximately correct with respect to  $\mathcal{L}_t$  with probability at least  $1 - \delta$ .

A PAC-learning algorithm can also be equipped with an *approximate equivalence* test  $\mathbf{EQ}$  which checks a candidate output  $\mathcal{L}_o$  against the target language  $\mathcal{L}_t$  using a *sufficiently large* sample of tagged sequences  $S$  generated by  $\mathbf{EX}$ . If the sample is such that for every  $x \in S$ ,  $x \in \mathcal{L}_t \iff x \in \mathcal{L}_o$ , the algorithm successfully stops and outputs  $\mathcal{L}_o$ . Otherwise, it picks any sequence in  $S \cap (\mathcal{L}_o \oplus \mathcal{L}_t)$  as *counterexample* and continues.

The algorithm may also be allowed to call directly a *membership* oracle **MQ**, such that **MQ**( $x, \mathcal{L}_t$ ) is true if and only if  $x \in \mathcal{L}_t$ . Notice that the **EX** oracle may also call **MQ** to tag sequences.

A *distribution-free* algorithm is one that works for every  $\mathcal{D}$ . Hereinafter, we will focus on distribution-free algorithms, so we will omit  $\mathcal{D}$ .

### 2.3 $L^*$

$L^*$  [1] learns *regular languages*, or equivalently, *deterministic finite automata* (DFA). Given a DFA  $\mathcal{A}$ , we use  $\mathcal{L}(\mathcal{A})$  to denote its language, that is, the set of sequences accepted by  $\mathcal{A}$ . We denote  $\mathcal{A}_t$  and  $\mathcal{A}_o$  the target and output automata, respectively. The symmetric difference between  $\mathcal{A}_o$  and  $\mathcal{A}_t$ , denoted  $\mathcal{A}_o \oplus \mathcal{A}_t$ , is defined as  $\mathcal{L}_o \oplus \mathcal{L}_t$ . We say that  $\mathcal{A}_o$   $\epsilon$ -approximates  $\mathcal{A}_t$  if  $\mathcal{L}_o$   $\epsilon$ -approximates  $\mathcal{L}_t$ .

$L^*$  uses **EQ** and **MQ**. Each time **EQ** is called, it must draw a sample of a size large enough to ensure a *total* confidence of the algorithm of at least  $1 - \delta$ . That is, whenever the statistical test is passed, it is possible to conclude that the candidate output is  $\epsilon$ -approximately correct with confidence at least  $1 - \delta$ .

Say **EQ** is called at iteration  $i$ . In order to guarantee the aforementioned property, a sample  $S_i$  of size  $r_i$  is drawn, where:

$$r_i = \left\lceil \frac{1}{\epsilon} (i \ln 2 - \ln \delta) \right\rceil \tag{1}$$

This ensures that the probability of the output automaton  $\mathcal{A}_o$  *not* being  $\epsilon$ -approximately correct with respect to  $\mathcal{A}_t$  when *all* sequences in a sample pass the **EQ** test, i.e.,  $S_i \cap (\mathcal{A}_o \oplus \mathcal{A}_t) = \emptyset$ , is *at most*  $\delta$ , that is:

$$\sum_{i>0} \mathbf{P}(S_i \cap (\mathcal{A}_o \oplus \mathcal{A}_t) = \emptyset \mid \mathbf{P}(\mathcal{A}_o \oplus \mathcal{A}_t) > \epsilon) < \sum_{i>0} (1 - \epsilon)^{r_i} < \sum_{i>0} 2^{-i} \delta < \delta$$

*Remark.* It is worth noticing that from the point of view of statistical hypothesis testing, a sample  $S_i$  that passes the test, gives a confidence of at least  $1 - 2^{-i} \delta$ .

## 3 PAC-Learning for ANN

We address the following problem: given a ANN  $\mathcal{N}$ , is it possible to build a DFA  $\mathcal{A}$ , such that  $\mathcal{L}(\mathcal{A})$  is  $\epsilon$ -approximately correct with respect to  $\mathcal{L}(\mathcal{N})$ ?

### 3.1 Basic Idea

The basic idea to solve this problem is to use  $L^*$  as follows. The **MQ** oracle consists in querying  $\mathcal{N}$  itself. The **EQ** oracle consists in drawing a sample set  $S_i$  with size  $r_i$  as defined in Eq. (1) and checking whether  $\mathcal{N}$  and the candidate automaton  $\mathcal{A}_i$  completely agree in  $S_i$ , that is,  $S_i \cap (\mathcal{A}_i \oplus \mathcal{N})$  is empty.

The results reviewed in the previous section entail that if  $L^*$  terminates, it will output a DFA  $\mathcal{A}$  which is an  $\epsilon$ -approximation of  $\mathcal{N}$  with probability at least

$1 - \delta$ . Moreover,  $L^*$  is proven to terminate *provided*  $\mathcal{L}(\mathcal{N})$  is a regular language. However, since ANN are strictly more expressive than DFA [23], there is no guarantee that  $L^*$  will eventually terminate; it may not exist a DFA  $\mathcal{A}$  with the same language as  $\mathcal{N}$ . In other words, there is no upper bound  $n_{\mathcal{N}}$  such that  $L^*$  will terminate in at most  $n_{\mathcal{N}}$  iterations for every target ANN  $\mathcal{N}$ . Therefore, it may happen that for every  $i$  the call to **EQ** fails, that is,  $S_i \cap (\mathcal{A}_i \oplus \mathcal{N}) \neq \emptyset$ .

### 3.2 Bounded- $L^*$

To cope with this situation, we resort to imposing a bound to the number of iterations of  $L^*$ . Obviously, a direct way of doing it would be to just fix an arbitrary upper bound to the number of iterations. Instead, we propose to constrain the maximum number of states of the automaton to be learned and to restrict the length of the sequences used to call **MQ**. The latter is usually called the *query length*. Typically, these two measures are used to determine the complexity of a PAC-learning algorithm [15].

#### 3.2.1 Algorithm

Similarly to the description presented in [16] the algorithm Bounded- $L^*$  (Algorithm 1) can be described as follows:

---

#### Algorithm 1. Bounded- $L^*$

---

**Input** : MaxQueryLength, MaxStates,  $\epsilon$ ,  $\delta$

**Output**: DFA  $\mathcal{A}$

```

1 Lstar-Initialise;
2 repeat
3   while OT is not closed or not consistent do
4     if OT is not closed then
5       | OT, QueryLengthExceeded  $\leftarrow$  Lstar-Close(OT);
6     end
7     if OT is not consistent then
8       | OT, QueryLengthExceeded  $\leftarrow$  Lstar-Consistent(OT);
9     end
10  end
11  if not QueryLengthExceeded then
12    | LastProposedAutomaton  $\leftarrow$  Lstar-BuildAutomaton(OT);
13    | Answer  $\leftarrow$  EQ(LastProposedAutomaton);
14    | MaxStatesExceeded  $\leftarrow$ 
15      | STATES(LastProposedAutomaton) > MaxStates;
16    | if Answer  $\neq$  Yes and not MaxStatesExceeded then
17      | | OT  $\leftarrow$  Lstar-UseEQ(OT, Answer);
18    | end
19  end
20  BoundReached  $\leftarrow$  QueryLengthExceeded or MaxStatesExceeded;
21 until Answer = Yes or BoundReached;
22 return LastProposedAutomaton;
```

---

The observation table  $OT$  is initialised by Lstar-Initialise in the same manner that it is for  $L^*$ . This step consists in building the structure of the observation table  $OT$  as proposed by Angluin. Then the construction of hypotheses begins.

If  $OT$  is not *closed* an extra row is added by the Lstar-Close procedure. If  $OT$  is *inconsistent*, an extra column is added by the Lstar-Consistent procedure. Both procedures call **MQ** to fill the holes in the observation table. The length of these queries may exceed the maximum query length, in which case the QueryLengthExceeded flag is set to true.

When the table is closed and consistent, and in the case that the query length was not exceeded, an equivalence query **EQ** is made and the automaton number of states is compared to the maximum number of states bound. If **EQ** is unsuccessful and the maximum number of states was not reached, new rows are added by Lstar-UseEQ, using the counterexample contained in *Answer*.

Finally, if the hypothesis passes the test or one of the bounds was reached, the algorithm stops and returns the last proposed automaton.

### 3.2.2 Analysis

Bounded- $L^*$  will either terminate with a successful **EQ** or when a bound (either the maximum number of states or query length) is exceeded. In the former case, the output automaton  $\mathcal{A}$  is proven to be an  $\epsilon$ -approximation of  $\mathcal{N}$  with probability at least  $1 - \delta$  by Angluin’s results. In the latter case, the output  $\mathcal{A}$  of the algorithm will be the *last automaton proposed by the learner*.  $\mathcal{A}$  may not be an  $\epsilon$ -approximation with confidence at least  $1 - \delta$ , because  $\mathcal{A}$  failed to pass the last statistical equivalence test **EQ**. However, the result of such test carries statistical value about the relationship between  $\mathcal{A}$  and  $\mathcal{N}$ . The question is, what could indeed be said about this automaton?

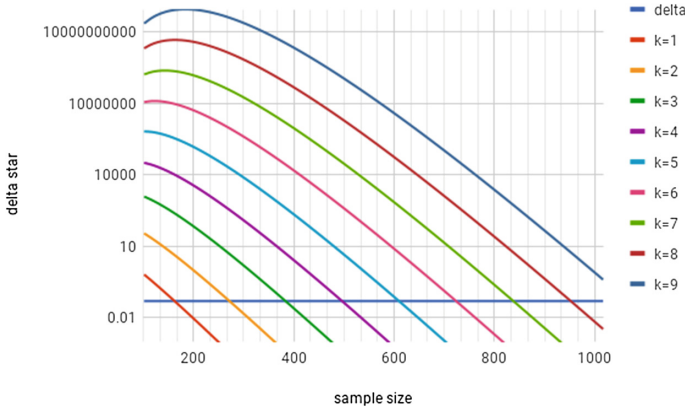
Assume at iteration  $i$  **EQ** fails and the number of states of  $\mathcal{A}_i$  is greater than or equal to the maximum number of states, or at the next iteration  $i + 1$ , **MQ** fails because the maximum query length is exceeded. This means that  $\mathcal{A}_i$  and  $\mathcal{N}$  disagree in, say,  $k > 0$ , of the  $r_i$  sequences of  $S_i$ . In other words, there are  $k$  sequences in  $S_i$  which indeed belong to the symmetric difference  $\mathcal{A}_i \oplus \mathcal{N}$ .

*Confidence Parameter.* Let  $p \in (0, 1)$  be the actual probability of a sequence to be in  $\mathcal{A}_i \oplus \mathcal{N}$  and  $K_{r_i}$  the random variable defined as the number of sequences in  $\mathcal{A}_i \oplus \mathcal{N}$  in a sample of size  $r_i$ . Then, the probability of  $K_{r_i} = k$  is:

$$\mathbf{P}(K_{r_i} = k) = \binom{r_i}{k} (1 - p)^{r_i - k} p^k$$

Let us first set as our hypothesis that  $\mathcal{A}_i$  is an  $\epsilon$ -approximation of  $\mathcal{N}$ . Can we accept this hypothesis when  $K_{r_i} = k$  with confidence at least  $1 - \delta'$ , for some  $\delta' \in (0, 1)$ ? In other words, is there a  $\delta'$  such that the probability of  $K_{r_i} = k$  is smaller than  $\delta'$  when  $p > \epsilon$ ? Suppose  $p > \epsilon$ . Then, it follows that:

$$\mathbf{P}(K_{r_i} = k \mid p > \epsilon) = \binom{r_i}{k} (1 - p)^{r_i - k} p^k < \binom{r_i}{k} (1 - \epsilon)^{r_i - k} < \binom{r_i}{k} e^{-\epsilon(r_i - k)}$$



**Fig. 1.** Values of  $\delta_i^*$  in log scale as function of  $r_i$ ,  $k \in [1, 9]$ ,  $i \in [3, 70)$

Therefore, if the following condition holds

$$\binom{r_i}{k} e^{-\epsilon(r_i-k)} < \delta' \tag{2}$$

we have that  $\mathbf{P}(K_{r_i} = k, p > \epsilon) < \delta'$ . That is, the probability of incorrectly accepting the hypothesis with  $k$  discrepancies in sample  $S_i$  of size  $r_i$  is smaller than  $\delta'$ . Then, we could accept the hypothesis with a confidence of at least  $1 - \delta'$ .

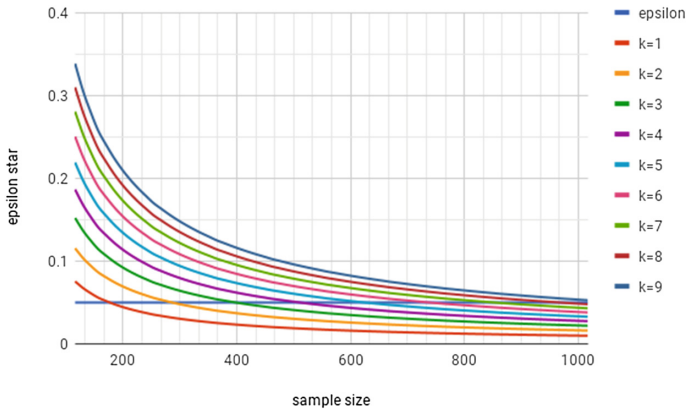
The left-hand-side term in condition (2) gives us a lower bound  $\delta_i^*$  for the confidence parameter such that we could accept the hypothesis with probability at least  $1 - \delta'$ , for every  $\delta' > \delta_i^*$ :

$$\delta_i^* = \binom{r_i}{k} e^{-\epsilon(r_i-k)} \tag{3}$$

A major problem, however, is that  $\delta_i^*$  may be greater than 1, and so no  $\delta' \in (0, 1)$  exists, or it may be just too large, compared to the desired  $\delta$ , to provide an acceptable level of confidence for the test.

Figure 1 shows  $\delta_i^*$ , in log scale, for  $\epsilon = 0.05$ ,  $k \in [1, 9]$  and  $r_i$  computed using Eq. (1), and compares it with a desired  $\delta = 0.05$ . We see that as  $k$  increases, larger values of  $r_i$ , or equivalently, more iterations, are needed to get a value of  $\delta_i^*$  smaller than  $\delta$  (horizontal line). Actually, for fixed  $k$  and  $\epsilon \in (0, 1)$ ,  $\delta_i^*$  tends to 0 as  $r_i$  tends to  $\infty$ . In other words, there is a large enough sample size for which it is possible to make  $\delta_i^*$  smaller than any desired confidence parameter  $\delta$ .

*Approximation Parameter.* An alternative would be to look at the approximation parameter  $\epsilon$ , rather than the confidence parameter  $\delta$ . In this case, we set as our hypothesis that  $\mathcal{A}_i$  is an  $\epsilon'$ -approximation of  $\mathcal{N}$ , for some  $\epsilon' \in (0, 1)$ . Is the probability of accepting this hypothesis with the test tolerating  $k$  discrepancies, when the hypothesis is actually false, smaller than  $\delta$ ? That is, we are asking whether the following condition holds for  $\epsilon'$ :



**Fig. 2.** Values of  $\epsilon_i^*$  as function of  $r_i$ ,  $k \in [1, 9]$ ,  $i \in [3, 70]$

$$\mathbf{P}(K_{r_i} = k \mid p > \epsilon') < \binom{r_i}{k} e^{-\epsilon'(r_i-k)} < \delta$$

Now, we could determine a lower bound  $\epsilon_i^*$ , such that this condition holds for every  $\epsilon' > \epsilon_i^*$ , in which case we could conclude that  $\mathcal{A}_i$  is an  $\epsilon'$ -approximation of  $\mathcal{N}$ , with confidence at least  $1 - \delta$ . Provided  $r_i - k \neq 0$ , we have:

$$\epsilon_i^* = \frac{1}{r_i - k} \left( \ln \binom{r_i}{k} - \ln \delta \right)$$

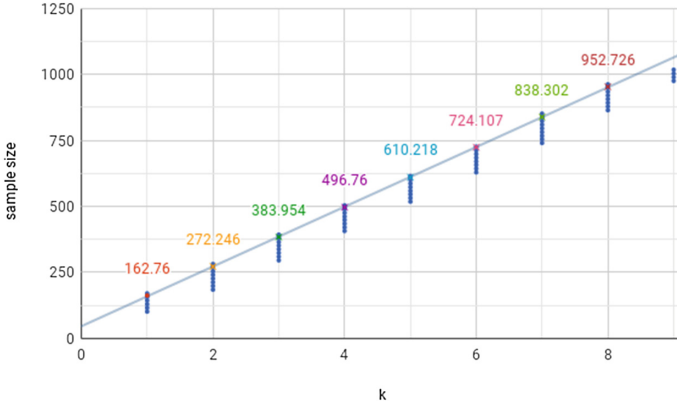
Figure 2 shows  $\epsilon_i^*$  for  $\delta = 0.05$ ,  $k \in [1, 9]$  and  $r_i$  computed using Eq. (1), and compares it with a desired  $\epsilon = 0.05$ . We see that as  $k$  increases, larger samples, i.e., more iterations, are needed to get a value smaller than  $\epsilon$  (horizontal line). Nevertheless, for fixed  $k$  and  $\delta \in (0, 1)$ ,  $\epsilon_i^*$  tends to 0 as  $r_i$  tends to  $\infty$ . In other words, there is a large enough sample size for which it is possible to make  $\epsilon^*$  smaller than any desired approximation parameter  $\epsilon$ .

*Number of Discrepancies and Sample Size.* Actually, we could also search for the largest number  $k^*$  of discrepancies which the **EQ** test could cope with for given  $\epsilon$ ,  $\delta$  and whichever sample size  $r$ , or the smallest sample size  $r^*$  for fixed  $\epsilon$ ,  $\delta$  and number of discrepancies  $k$ , independently of the number  $i$  of iterations.

The values  $k^*$  and  $r^*$  could be obtained by solving the following equation for  $k$  and  $r$ , respectively:

$$\ln \binom{r}{k} - \epsilon \ln(r - k) - \ln \delta = 0 \tag{4}$$

Figure 3 plots the relationship between  $k \in [1, 9]$ ,  $r_i$  computed using Eq. (1), and  $r^*(k)$ , where  $r^*(k)$  denotes the value of  $r^*$  for the given  $k$ . Each point in the vertical dotted segments corresponds to a value of sample size  $r_i$ . For a given



**Fig. 3.** Comparison between  $r_i$ ,  $r^*$  and  $k$

value of  $k$ , we plot all values of  $r_i$  up to the first one which becomes greater than  $r^*(k)$ . For each value of  $k$ , the value of  $r^*(k)$  is shown in numbers.

Whenever  $r_i$  is greater than  $r^*(k)$ , if **EQ** yields  $k$  discrepancies at iteration  $i$  then  $\mathcal{A}_i$  could be accepted as being  $\epsilon$ -approximately correct with respect to  $\mathcal{N}$  with confidence at least  $1 - \delta$ . For values of  $r_i$  smaller than  $r^*(k)$ , **EQ** must yield a number of discrepancies smaller than  $k$  for the automaton to be accepted as  $\epsilon$ -approximately correct.

The diagonal line in Fig. 3 is a linear regression that shows the evolution of  $r^*$  as a function of  $k$ . Notice that the value of  $r^*$  seems to increase linearly with  $k$ . However, if we look at the increments, we observe that this is not the case. As Fig. 4 shows, they most likely exhibit a log-like growth.

*Sample Size Revisited.* The previous observations suggest that it could be possible to cope with an a-priori given number  $k$  of acceptable discrepancies in the **EQ** test by taking larger samples. This could be done by revisiting the formula (1) to compute sample sizes by introducing  $k$  as parameter of the algorithm.

Following Angluin’s approach, let us take  $\delta_i$  to be  $2^{-i}\delta$ . We want to ensure:

$$\mathbf{P}(K_{r_i} = k \mid \mathbf{P}(\mathcal{A}_o \oplus \mathcal{A}_t) > \epsilon) < \binom{r_i}{k} e^{-\epsilon(r_i - k)} < 2^{-i}\delta$$

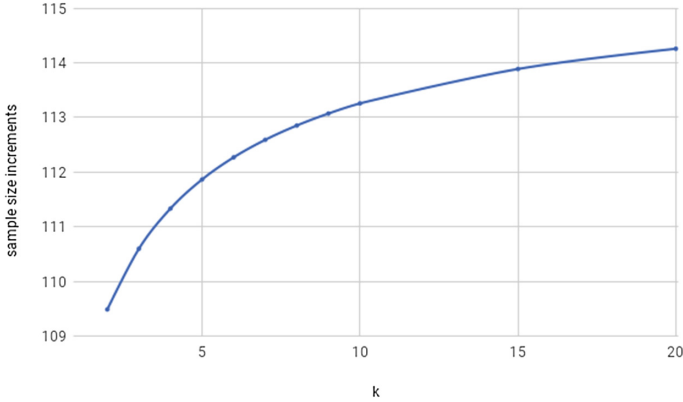
Hence, the smallest such  $r_i$  is:

$$r_i = \arg \min_{r \in \mathbb{N}} \left\{ \ln \binom{r}{k} - \epsilon(r - k) + i \ln 2 - \ln \delta < 0 \right\} \tag{5}$$

Notice that for  $k = 0$ , this gives the same sample size as in Eq. (1).

Now, with sample size  $r_i$ , we can ensure a total confidence of  $1 - \delta$ :

$$\sum_{i>0} \mathbf{P}(K_{r_i} = k \mid \mathbf{P}(\mathcal{A}_o \oplus \mathcal{A}_t) > \epsilon) < \sum_{i>0} 2^{-i}\delta < \delta$$



**Fig. 4.** Increments of  $r^*$  as function of  $k$

Although computing the sample size at each iteration using Eq. (5) does allow to cope with up to a given number of discrepancies  $k$  in the **EQ** test, it does not ensure termination. Moreover, solving Eq. (5) is computationally expensive, and changing the **EQ** test so as to accepting at most  $k$  divergences in a set of  $r_i$  samples guarantees the same confidence and approximation as passing the standard zero-divergence test proposed in PAC. Hence, in this work we compute  $r_i$  as in Eq. (1). Then, we analyze the values of  $\epsilon_i^*$  that result *after* stopping Bounded- $L^*$  when a complexity constraint has been reached, and compare them with  $\epsilon$  and measured errors on a test set.

*Remark.* One could argue that this analysis may be replaced by running the algorithm with less restrictive bounds. The main issue here is that the target concept (the language of the ANN) may not be in the hypothesis space (regular languages). Thus, there is no guarantee a hypothesis exists for which the PAC condition holds for whichever  $\epsilon$  and  $\delta$ . So, even if the user fixed looser parameters, there is no guarantee the algorithm ends up producing a PAC-conforming DFA.

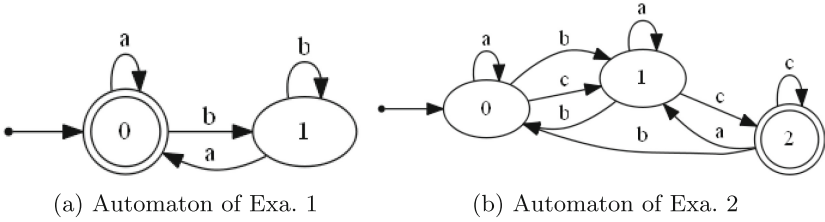
## 4 Experimental Results

We implemented Bounded- $L^*$  and applied it to several examples. In the experiments we used LSTM networks. Two-phase early stopping was used to train the LSTM, with an 80–20% random split for train-test of a randomly generated dataset. For evaluating the percentage of sample sequences in the symmetric difference we used 20 randomly generated datasets. The LSTM were trained with sample datasets from known automata as a way of validating the approach. However it is important to remark that in real application scenarios such automata are unknown, or the dataset may not come from a regular language.

*Example 1.* Let us consider the language  $(a + b)^*a + \lambda$ , with  $\Sigma = \{a, b\}$  and  $\lambda$  being the empty sequence (Fig. 5a). We performed 100 runs of the algorithm with



different values for the  $\epsilon$  and  $\delta$  parameters. For each run Bounded- $L^*$  terminated normally and obtained a DFA that was an  $\epsilon$ -approximation of the neural network with confidence at least  $1 - \delta$ . Actually, every learned automata was equivalent to the original automaton and had no differences in the evaluation of the test datasets with regards to the neural network.  $\square$



**Fig. 5.** Automata examples

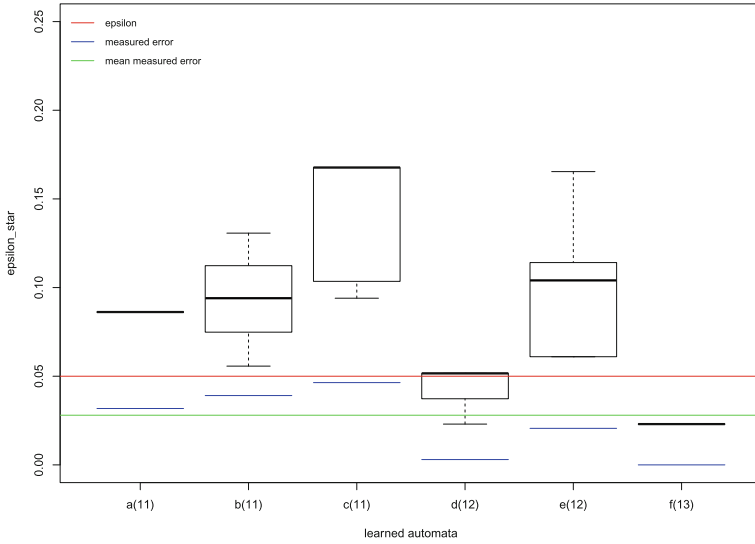
The previous experiment illustrates that when the neural network is well trained, meaning that it exhibits zero error with respect to all test datasets, the learner will, with high probability, learn the original automaton, which is unknown to both teacher and learner. This case would be the equivalent to using the automaton as the **MQ** oracle, falling in the setting of PAC based  $L^*$ . This situation rarely happens in reality, as neural networks, or any model, are never trained to perfectly fit the data. Therefore we are interested in testing the approach with neural networks that do not perfectly characterize the data.

*Example 2.* Consider the DFA shown in Fig. 5b, borrowed from [26]. The training set contained 160K sequences of variable length up to a maximum length of 10. The error measured on another randomly generated sample test set of 16K sequences between the trained LSTM and the DFA was 0.4296. That is, the LSTM does not perform very well: what language did it actually learn?

Bounded- $L^*$  was executed 20 times with  $\epsilon = \delta = 0.05$  and a bound of 10 on the number of states. All runs reached the bound and the automaton of the last iteration was the one with smallest error of all iterations.

Figure 6 summarizes the results obtained. It can be observed that for each run of the experiment, not always the same automaton is reached due to the random nature of the **EQ** oracle sample picking. In the 20 runs, 6 different DFA were produced, identified with letters  $a$  to  $f$ , with a number of states between 11 and 13 (indicated in parenthesis).

For the each automaton, the measured error on the test set is obviously the same. However,  $\epsilon_i^*$  and  $\delta_i^*$  may differ, because they depend on the number of iterations and on the number of discrepancies on the randomly generated sample sets used by oracle **EQ** in each case. This is depicted in the figure with a boxplot of the  $\epsilon_i^*$  values for each automaton produced. It is important to notice that the percentage of sequences in the symmetric difference (measured error) between each learned automaton and the neural network, measured on a set of randomly

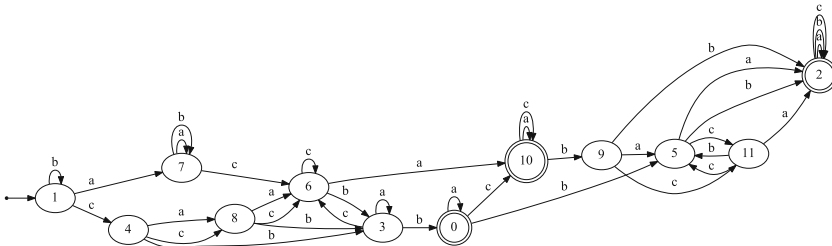


**Fig. 6.** Measured error,  $\epsilon$ , and  $\epsilon_i^*$  for Example 2

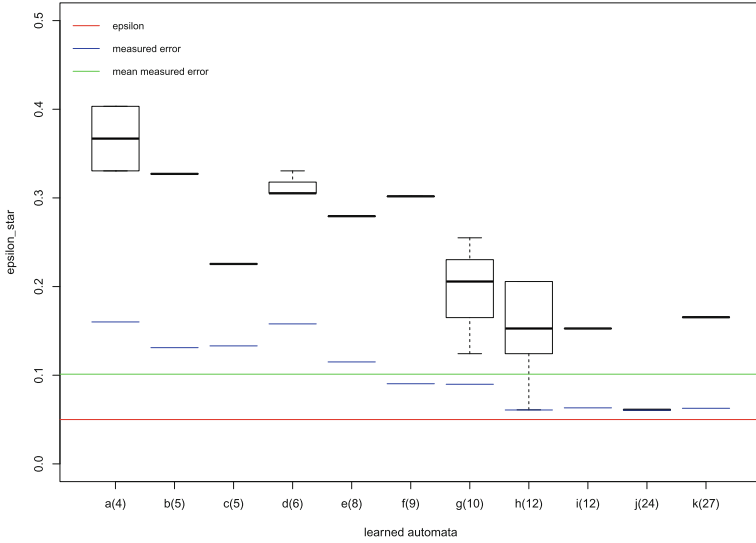
generated sample test sets, is always below the calculated  $\epsilon_i^*$  value. It means that the measured empirical errors are consistent with the theoretical values. It is interesting to observe that the measured error was smaller than  $\epsilon$ .  $\square$

*Example 3.* This example presents the results obtained with an LSTM trained with a different dataset with positive and negative sequences of the same automaton as the previous example. In this case, the measured error was 0.3346.

We ran Bounded- $L^*$  20 times with  $\epsilon = \delta = 0.05$ , and a maximum query length of 12. All runs reached the bound. Figure 8 summarizes the results obtained. The experiment produced 11 different DFA, named with letters from  $a$  to  $k$ , with sizes between 4 and 27 number of states (indicated in parenthesis). Figure 7 shows the 12-state automaton with smallest error. All automata exhibited measured errors smaller or equal than  $\epsilon_i^*$ . In contrast, they were all above the proposed  $\epsilon$ . For



**Fig. 7.** 12-state automaton with smallest error

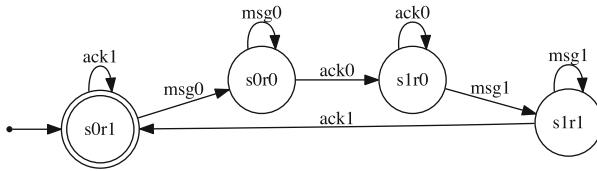


**Fig. 8.** Measured error,  $\epsilon$ , and  $\epsilon_i^*$  of Example 3

automata  $h$  (12 states) and  $j$  (24 states), measured errors are slightly smaller than the respective minimum  $\epsilon_i^*$  values.

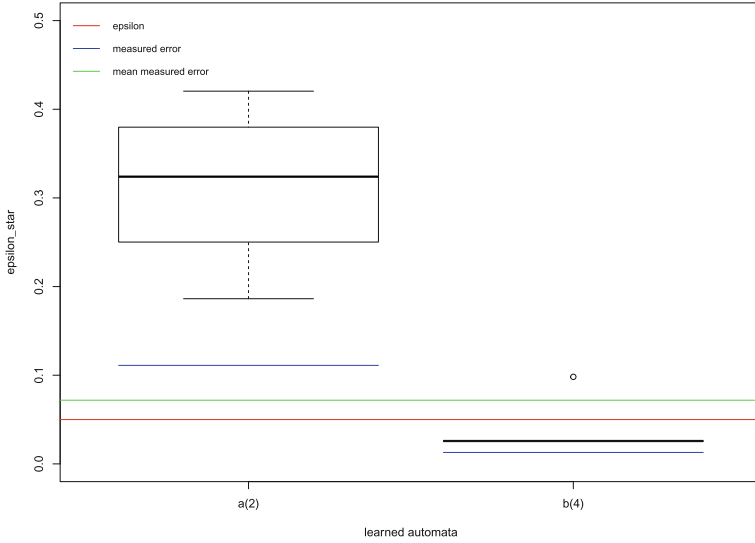
Notice that this experiment shows higher variance in the number of automata, number of automaton states, and  $\epsilon^*$  values than the previous one.  $\square$

*Example 4.* We study here the Alternating Bit Protocol (Fig. 9). The LSTM measured error was 0.2473. We ran Bounded- $L^*$  with  $\epsilon = \delta = 0.05$ , and a maximum query length of 5.

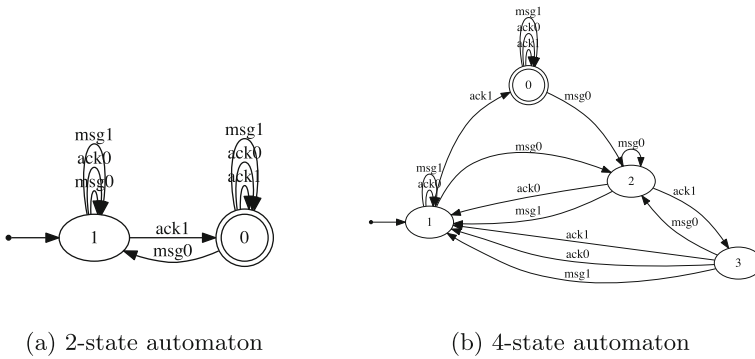


**Fig. 9.** Alternating bit protocol automaton

Two different automata were obtained, named  $a$  and  $b$ , with 2 and 4 states respectively (Fig. 11). All runs that produced  $a$  reached the bound. This is explained in Fig. 10 where the boxplot for  $a$  is above  $\epsilon$ . On the other hand, almost all runs that produced  $b$  completed without reaching the bound. Nevertheless, in all cases where the bound was reached, the sample size was big enough to guarantee  $\epsilon^*$  to be smaller than  $\epsilon$ . Overall, the empirical error measures were consistent with the theoretical values.  $\square$



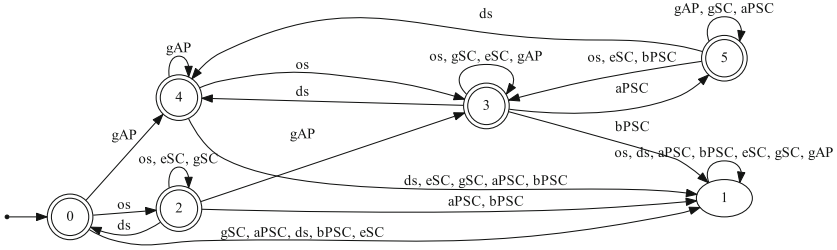
**Fig. 10.** Measured error,  $\epsilon$ , and  $\epsilon_i^*$  of Example 4



**Fig. 11.** Generated automata for ABP

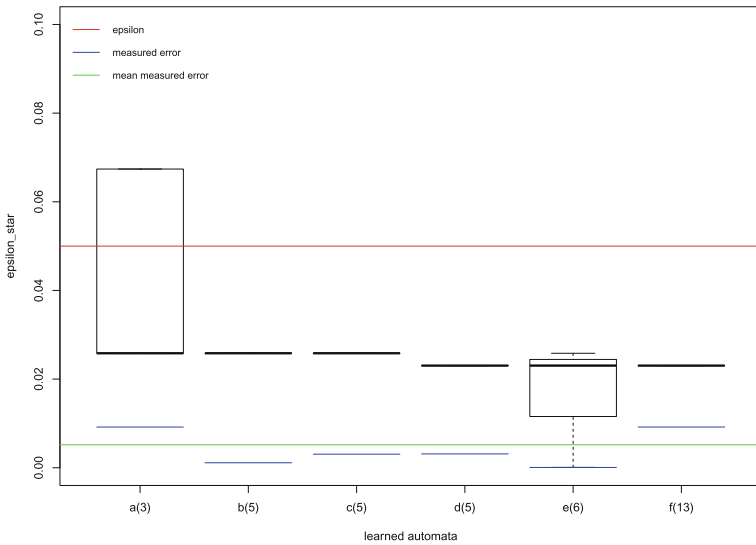
*Example 5.* We consider here an adaptation of the e-commerce website presented in [24] (Fig. 12). The labels are explained in the following table:

os: open session	ds: destroy session
gAP: get available product	eSC: empty shopping cart
gSC: get shopping cart	aPSC: add product to shopping cart
bPSC: buy products in shopping cart	



**Fig. 12.** Model of the e-commerce example adaptation

The training set contained 44K sequences up to a maximum length of 16. The test set contained 16K sequences. The measured error of the ANN on the test set was 0.0000625. We ran Bounded- $L^*$  with  $\epsilon = \delta = 0.05$ , a maximum query length of 16 and a bound of 10 on the number of states. Figure 13 shows the experimental results.



**Fig. 13.** Measured error,  $\epsilon$ , and  $\epsilon_i^*$  of Example 5

Figure 14 shows one of the learned automata. It helps interpreting the behavior of the network. For example, given  $oS, gAP, aPSC, aPSC, bPSC$ , the output of the network is 1, meaning that the sequence is a valid sequence given the concept the network was trained to learn. Besides, this sequence also accepted by the automaton, yielding a traceable way of interpreting the result of the network. Moreover, for the input sequence  $oS, gAP, aPSC, gSC, eSC, gAP, gAP, bPSC$ , we find that the network outputs 1. However, this sequence is not accepted by

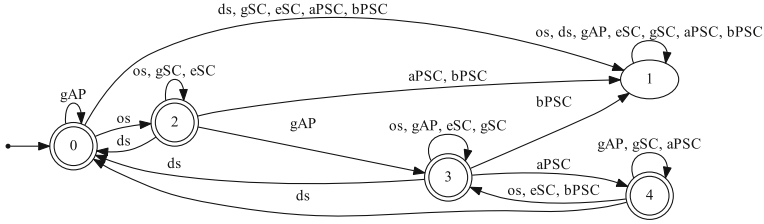


Fig. 14. One of the learned models of the e-commerce example adaptation

the automaton (like the majority of the learned models). This highlights that the network misses an important property of the e-commerce site workflow: it is not possible to buy products when the shopping cart is empty. □

*Remark.* The variance on the outcomes produced by Bounded- $L^*$  for the same input ANN could be explained by representational, statistical and computational issues [7]. The first occurs because the language of the network may not be in the hypothesis space, due to the fact that ANN are strictly more expressive than DFA. The second and third are consequences of the sampling performed by EQ and the policy used to choose the counter-example.

## 5 Related Work

A thorough review of the state of the art in explainable AI is presented in [10, 14].

To the best of our knowledge, the closest related works to ours are the following. The approaches discussed in [3] are devoted to extracting decision trees and rules for specific classes of multi-layer feed-forward ANN. Besides, such models are less expressive than DFA. Approaches that aim at extracting DFA are white box, that is, they rely on knowing the internal structure of the ANN. For instance, [9] deals with second-order RNN. The algorithm developed in [31] proposes an equivalence query based on the comparison of the proposed hypotheses with an abstract representation of the RNN that is obtained through an exploration of its internal state.

Work on regular inference [15] focused on studying the learnability of different classes of automata but none was applied to extracting them from ANN.

None of these works provide means for black-box model explanation in the context of ANN. Moreover, our approach using PAC regular inference is completely agnostic of the model.

## 6 Conclusions

We presented an active PAC-learning algorithm for learning automata that are approximately correct with respect to neural networks. Our algorithm is a variant

of Angluin's  $L^*$  where a bound on the number of states or the query length is set to guarantee termination in application domains where the language to be learned may not be a regular one. We also studied the error and confidence of the hypotheses obtained when the algorithm stops by reaching a complexity bound.

The experimental evaluation of our implementation showed that the approach is able to infer automata that are reasonable approximations of the target models with high confidence, even if the output model does not pass the usual 0-divergence **EQ** statistical test of the PAC framework. These evaluations also provided empirical evidence that the method exhibits high variability in the proposed output models. This is a key concern to be addressed in future work.

**Acknowledgments.** This work has been partially funded by an ICT4V - Information and Communication Technologies for Verticals master thesis grant under the code POS\_ICT4V\_2016\_1.06.

## References

1. Angluin, D.: Learning regular sets from queries and counterexamples. *Inf. Comput.* **75**(2), 87–106 (1987)
2. Angluin, D.: Computational learning theory: survey and selected bibliography. In: *Proceedings of the Twenty-Fourth Annual ACM Symposium on Theory of Computing, STOC 1992*, pp. 351–369. ACM, New York (1992)
3. Bologna, G., Hayashi, Y.: Characterization of symbolic rules embedded in deep DIMLP networks: a challenge to transparency of deep learning. *J. Artif. Intell. Soft Comput. Res.* **7**(4), 265–286 (2017)
4. Calderon, T.G., Cheh, J.J.: A roadmap for future neural networks research in auditing and risk assessment. *Int. J. Account. Inf. Syst.* **3**(4), 203–236 (2002). Second International Research Symposium on Accounting Information Systems
5. Clarke Jr., E.M., Grumberg, O., Peled, D.A.: *Model Checking*. MIT Press, Cambridge (1999)
6. Deng, L., Chen, J.: Sequence classification using the high-level features extracted from deep neural networks. In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6844–6848, May 2014
7. Dietterich, T.G.: Ensemble methods in machine learning. In: Kittler, J., Roli, F. (eds.) *MCS 2000. LNCS*, vol. 1857, pp. 1–15. Springer, Heidelberg (2000). [https://doi.org/10.1007/3-540-45014-9\\_1](https://doi.org/10.1007/3-540-45014-9_1)
8. Gers, F.A., Schmidhuber, E.: LSTM recurrent networks learn simple context-free and context-sensitive languages. *IEEE Trans. Neural Netw.* **12**(6), 1333–1340 (2001)
9. Giles, C.L., Miller, C.B., Chen, D., Chen, H.H., Sun, G.Z., Lee, Y.C.: Learning and extracting finite state automata with second-order recurrent neural networks. *Neural Comput.* **4**(3), 393–405 (1992)
10. Gilpin, L.H., Bau, D., Yuan, B.Z., Bajwa, A., Specter, M., Kagal, L.: Explaining explanations: an approach to evaluating interpretability of machine learning. *CoRR abs/1806.00069* (2018)
11. Gold, E.M.: Complexity of automaton identification from given data. *Inf. Control* **37**(3), 302–320 (1978)
12. Gorman, K., Sproat, R.: Minimally supervised number normalization. *TACL* **4**, 507–519 (2016)

13. Grzes, M., Taylor, M. (eds.): Proceedings of the Adaptive and Learning Agents Workshop (2010)
14. Guidotti, R., Monreale, A., Turini, F., Pedreschi, D., Giannotti, F.: A survey of methods for explaining black box models. CoRR abs/1802.01933 (2018)
15. Heinz, J., de la Higuera, C., van Zaanen, M.: Formal and empirical grammatical inference. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts of ACL 2011, HLT 2011, pp. 2:1–2:83. Association for Computational Linguistics, Stroudsburg (2011)
16. de la Higuera, C.: Grammatical Inference: Learning Automata and Grammars. Cambridge University Press, New York (2010)
17. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
18. Holzinger, A., Biemann, C., Pattichis, C.S., Kell, D.B.: What do we need to build explainable AI systems for the medical domain? [arXiv:1712.09923](https://arxiv.org/abs/1712.09923) (2017)
19. Holzinger, A., Plass, M., Holzinger, K., Crisan, G.C., Pintea, C.M., Palade, V.: A glass-box interactive machine learning approach for solving NP-hard problems with the human-in-the-loop. [arXiv:1708.01104](https://arxiv.org/abs/1708.01104) (2017)
20. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* **521**, 436–444 (2015)
21. Lei, T., Barzilay, R., Jaakkola, T.: Rationalizing neural predictions. In: Empirical Methods in Natural Language Processing (EMNLP) (2016)
22. Malhotra, P., Vig, L., Shroff, G., Agarwal, P.: Long short term memory networks for anomaly detection in time series. In: Proceedings of 23rd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (2015)
23. McCulloch, W.S., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.* **5**(4), 115–133 (1943)
24. Merten, M.: Active automata learning for real life applications (2013)
25. Murphy, K.: Passively learning finite automata. Technical report, 96-04-017, Santa Fe Institute (1996)
26. Omlin, C.W., Giles, C.L.: Constructing deterministic finite-state automata in recurrent neural networks. *J. ACM* **43**(6), 937–972 (1996)
27. Pascanu, R., Stokes, J.W., Sanossian, H., Marinescu, M., Thomas, A.: Malware classification with recurrent networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2015, South Brisbane, 19–24 April 2015, pp. 1916–1920 (2015)
28. Sarikaya, R., Hinton, G.E., Deoras, A.: Application of deep belief networks for natural language understanding. *IEEE/ACM Trans. Audio Speech Lang. Process.* **22**(4), 778–784 (2014)
29. Valiant, L.G.: A theory of the learnable. *Commun. ACM* **27**(11), 1134–1142 (1984)
30. Wang, Y., Tian, F.: Recurrent residual learning for sequence classification. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, 1–4 November 2016, pp. 938–943 (2016)
31. Weiss, G., Goldberg, Y., Yahav, E.: Extracting automata from recurrent neural networks using queries and counterexamples. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning, PMLR, vol. 80. Stockholmsmässan, Stockholm, 10–15 July 2018
32. Xing, Z., Pei, J., Keogh, E.: A brief survey on sequence classification. *SIGKDD Explor. Newsl.* **12**(1), 40–48 (2010)
33. Zhang, C., Jiang, J., Kamel, M.: Intrusion detection using hierarchical neural networks. *Pattern Recognit. Lett.* **26**(6), 779–791 (2005)



34. Zhou, C., Cule, B., Goethals, B.: Pattern based sequence classification. *IEEE Trans. Knowl. Data Eng.* **28**(5), 1285–1298 (2016)
35. Zhou, C., Sun, C., Liu, Z., Lau, F.C.M.: A C-LSTM neural network for text classification. *CoRR* abs/1511.08630 (2015)

## Author Index

- Abdollahpouri, Alireza 11  
Akata, Zeynep 295  
Alayba, Abdulaziz M. 179  
André, Totohasina 79  
Angeli, Alessia 282
- Bharadwaj, Varun 160  
Biemann, Chris 192, 212  
Bologna, Guido 304  
Boycheva, Svetla 223  
Bugeja, Mark 65  
Buitrago, Katherine Espíndola 134
- Camporesi, Francesco 265  
Carrington, André 329  
Cepeda, Catia 28  
Chander, Ajay 295, 314  
Cheetham, Marcus 28  
Chen, Helen 329  
Choi, Ben 118
- Dhiman, Hitesh 239  
Dias, Maria Camila 28  
Dingli, Alexiei 65
- England, Matthew 179
- Felsberger, Lukas 98  
Ferri, Massimo 282  
Fieguth, Paul 329  
Frosini, Patrizio 265
- Gamboa, Hugo 28  
Garcia, Juan Carlos 134  
Germanakos, Panagiotis 147  
Gilmanov, Rustem R. 273  
Goebel, Randy 295  
Gschwander, Tjorben 212
- Harrimann, Ramanantsoa 79  
Hartung, Dirk 212  
Heinz, Mario 239, 248
- Holzinger, Andreas 1, 147, 295  
Holzinger, Katharina 295
- Iqbal, Rahat 179
- Johannßen, Dirk 192
- Kalyuzhnyuk, Alexander V. 273  
Kieseberg, Peter 1, 295  
Killing, Tobias 212  
Kranzlmüller, Dieter 98
- Lamperti, Gianfranco 43  
Lecue, Freddy 295
- Majd, Shahnaz Mohammadi 11  
Manjunath, Varun 160  
Mayr, Franz 350  
Mercieca, Simon 65  
Monti, Eleonora 282  
Moreno-Sandoval, Luis G. 134  
Mourlas, Constantinos 147
- Oliveira, Diogo 28
- Palade, Vasile 179  
Parfait, Bemarisika 79  
Pomares-Quimbaya, Alexandra 134
- Quercioli, Nicola 265
- Radhakrishnan, Sreedhar 160  
Rahimi, Shadi 11  
Rindlisbacher, Dina 28  
Röcker, Carsten 239, 248  
Rodrigues, Joao 28  
Rönneburg, Lennart 212  
Ruppert, Eugen 212
- Salavati, Chiman 11  
Sánchez-Barriga, Carolina 134  
Seychell, Dylan 65

Sittig, Phillip 212  
Sotirakou, Catherine 147  
Srinath, Ramamoorthy 160  
Srinivasan, Ramya 314  
Stumpf, Simone 295

Taimanov, Iskander A. 273  
Timofeyev, Andrey 118  
Tjoa, A Min 1

Todd, Benjamin 98  
Tomba, Ivan 282

Weippl, Edgar 1

Yakovlev, Andrey A. 273  
Yovine, Sergio 350

Zanella, Marina 43  
Zhao, Xiangfu 43