Studies in Computational Intelligence 1084

Gintautas Dzemyda
Jolita Bernatavičienė
Janusz Kacprzyk *Editors*

Data Science in Applications

Springer

Studies in Computational Intelligence

Volume 1084

The series "Studies in Computational Intelligence" (SCI) publishes new developments and advances in the various areas of computational intelligence—quickly and with a high quality. The intent is to cover the theory, applications, and design methods of computational intelligence, as embedded in the fields of engineering, computer science, physics and life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in computational intelligence spanning the areas of neural networks, connectionist systems, genetic algorithms, evolutionary computation, artificial intelligence, cellular automata, self-organizing systems, soft computing, fuzzy systems, and hybrid intelligent systems. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution, which enable both wide and rapid dissemination of research output.

Indexed by SCOPUS, DBLP, WTI Frankfurt eG, zbMATH, SCImago.

All books published in the series are submitted for consideration in Web of Science.

Gintautas Dzemyda · Jolita Bernatavičienė ·
Janusz Kacprzyk
Editors

Data Science in Applications

Springer

Editors
Gintautas Dzemyda
Institute of Data Science and Digital Technologies
Vilnius University
Vilnius, Lithuania

Jolita Bernatavičienė
Institute of Data Science and Digital Technologies
Vilnius University
Vilnius, Lithuania

Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
Warsaw, Poland

Faculty of Computer Science, Electronics and Telecommunications
AGH University of Science and Technology
Kraków, Poland

ISSN 1860-949X ISSN 1860-9503 (electronic)
Studies in Computational Intelligence
ISBN 978-3-031-24452-0 ISBN 978-3-031-24453-7 (eBook)
https://doi.org/10.1007/978-3-031-24453-7

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

Preface

One of the main and most challenging characteristic features of virtually all modern technologies is that the volume of data involved is huge, not comparable to what was occurring even some years ago, and—which is often even more challenging—the forms of data, exemplified by images, numbers, data streams, data related to human behavior and physiological parameters, etc. immensely complicate all efforts to handle them in search for a rationale use to analyze and solve problems.

Data science, a rapidly developing discipline of science and technology, has made a remarkable progress to deal in an effective and efficient way with all kinds of those data related challenges and problems. However, there still are many open questions to be solved, both from analytic and logarithmic points of views, and with respect to implementations. This is an important reason for much research efforts in this and related fields that we witness world wide. To just give some examples, these concern the topics like visualizations, statistics, pattern recognition, neurocomputing, image analysis, machine learning, artificial intelligence, databases and data processing, data mining, big data analytics, knowledge discovery in databases, etc. Important relations to new developments like block chaining, cyber-social and cyber-physical systems, Internet of things (IoT), social computing, cognitive computing, high-performance computing, cloud computing, crowdsource analysis, etc. are just some examples of what is a current trend. Of course, much research is also done in more traditional fields exemplified by the use of optimization and metaheuristics, information theoretic analyses, etc. to just mention a few. Potential fields of applications of these modern data science-based approaches, and tools and techniques, are too numerous to be listed, they cover practically all areas of science and technology. This growing demand for data science specialists and data analysts implies a considerable growth in new study programs at virtually all universities, and a growing popularity of research on education processes and their effectiveness and efficiency.

This book contains 11 chapters by well-known researchers working in different fields of the broadly perceived data science, involving both more basic and foundational works and relevant applications.

Anita Juškevičienė, Arnold Pears, Tatjana Jevsikova and Gabrielė Stupurienė ("Computational Thinking Design Application for STEAM Education") is concerned with the integration of the so-called STEAM education and Computational Thinking (CT). That is, first, it concerns the integration of the STEAM education which is basically an approach to learning that uses Science, Technology, Engineering, the Arts and Mathematics as fields that provide tools and techniques for guiding student inquiry, dialog, and critical thinking. Second, it involves computational thinking (CT) which can be described as a skill and ability to use concepts, reasoning, etc. that come from computing and computer science, to solve all kinds of problems. The authors are concerned with and analysis of how STEM and CT can provide a link between research, education, and commercial and industrial partners. As a solution, the use of computational and design thinking is proposed and advocated. The results of this new approach are encouraging.

Audronė Jakaitienė, Rimantas Želvys and Rita Dukynaitė ("Education Data for Science: Case of Lithuania") provide a comprehensive and critical review of various sources of educational data (e.g., international large-scale studies, data registers) and their use for developing policy decisions, and also extend research agenda, in Lithuania. It is shown that a lot of data has already been collected and stored, with ca. 20% for policymaking and even less for research. An important result of analysis and a case study is that national population-based studies and international achievement studies may send different messages and cannot be considered in isolation.

Dalia Breskuvienė and Gintautas Dzemyda ("Imbalanced Data Classification Approach Based on Clustered Training Set") are concerned with an important problem of fraud detection and its possible solutions to prevent criminals from obtaining financial assets. More specifically, the goal of the approach proposed is to increase machine learning prediction quality on fraudulent cases as well as to decrease false positive and false negative cases in prediction results. Since fraudulent data exemplified by credit card transactions are usually imbalanced and this implies problems with the use of the standard machine learning techniques. The authors propose a clustering-based classification method. It is suggested to first find the optimal features and number of clusters to create smaller, more homogeneous training sets to be trained on separate machine learning models, and—second—to find relevant percentages to undersample each cluster to compensate for sharply imbalanced data. The method yields significantly better results.

Giedrė Dzemydaitė, Brigita Šidlauskaitė-Riazanova and Darjuš Bartkevičius ("Baltic States in Global Value Chains: Quantifying International Production Sharing at Bilateral and Sectoral Levels") shows data science -based methods for the analysis of global value chains (GVC) by a decomposition of the gross exports data. The analysis is focused on the participation of the Baltic States in the global value chains via a quantification of the international production shares at bilateral and sectoral levels. An accounting framework is employed that decomposes the country's gross exports into various value-added components which integrate all the previous vertical specialization and value-added trade approaches into a unified framework to assess the countries' participation in the global value chains. The results obtain indicated that the Baltic States' participation in the global value chains is growing during the research period, notably thanks to foreign value-added increases in the countries' exports.

Domnica Dzitac ("The Soft Power of Understanding Social Media Dynamics: A Data-Driven Approach") is concerned with issue related to social media that have become an increasingly popular arena for political debates. Unfortunately, they have often also been misused as a form of soft power to influence voters, spread fear or even destabilize democracies. The author discusses challenges, ethical considerations and moral dilemmas related to these and related issues, notably regarding the new era of a data-driven society. Moreover, a data science approach is proposed to understand the dynamics of controversial political topics on Twitter in the US context. The tweets are analyzed using modern state-of-the-art data science and Natural Language Processing (NLP) tools and techniques. Notably, an extensive analysis on the labeling of emotions of tweets and computing their attention score is provided. The results obtained indicated that anger and fear are the most prominent emotions.

Mirko Armillotta, Konstantinos Fokianos and Ioannis Krikidis ("Bootstrapping Network Autoregressive Models for Testing Linearity") develop a new methodology for spatio-temporal data analyses with special attention to epidemic network structures. The authors provide estimation tools for the linear network autoregressive models for various time series. Non-linear models for inference under the assumption of a known network structure are discussed, and a family of test statistics for testing the linearity of the imposed model is proposed. An empirical comparison of two bootstrap versions of a supremum-type quasi-score test is presented. An application to the analysis of daily COVID-19 cases detected on province-level geographical network in Italy is shown. The results are encouraging.

Mario Manzo, Maurizio Giordano, Lucia Maddalena, Mario Rosario Guarracino and Ilaria Granata ("Novel Data Science Methodologies for Essential Genes Identification Based on Network Analysis") are concerned with the so-called essential genes (EGs) which are fundamental for the growth and survival of a cell or an organism. The essentiality is a context-dependent dynamic attribute of a gene that can vary in different cells, tissues, or pathological conditions, and experimental procedures to identify the essential genes are costly and time consuming. Commonly explored computational approaches are based on the use of machine learning applied to protein-protein interaction and are often not effective. From a biological point of view, the identification of attributes of the node essentiality is challenging, and from a data science perspective the use of suitable graph learning approaches still represents an open problem. The new model proposed is based on both the relationship information and the node attributes. The results are encouraging.

Monika Danilovaitė and Gintautas Tamulevičius ("Acoustic Analysis for Vocal Fold Assessment—Challenges, Trends, and Opportunities") are concerned with a comprehensive and critical review of trends in non-invasive vocal fold assessment to identify the significance of acoustic analysis. A classification scheme is applied to process the selected relevant study set, and a systematic map is used to synthesize data for quantitative analysis. Results show that the non-invasive vocal fold assessment by using machine learning tools and techniques is effective and efficient.

Vytautas Petrauskas, Raimundas Jasinevicius, Egidijus Kazanavičius and Zygimantas Meskauskas ("The Paradigm of an Explainable Artificial Intelligence (XAI) and Data Science (DS)-Based Decision Support System (DSS)") are concerned with some relevant issues related to the decision support systems (DSS) which are gaining popularity as the explainable artificial intelligence (XAI) is considered to be more and more crucial. Unfortunately, most of the DSSs currently employed are mainly meant for some kind of diagnostics and do not provide mechanisms for more sophisticated solutions. The author proposes to use for this purpose the latest XAI techniques based on the use of a new, generalized approach, the newly developed fuzzy SWOT maps (FSM), and elements of the computing with words (CWW) to deal with lists of rules (LoR). Moreover, a new general approach is proposed including elements of various fields of science, notably philosophy and praxeology. Results on the analysis of opportunities and threats faced by Lithuania are shown.

Virgilijus Sakalauskas, Dalia Kriksciuniene and Audrius Imbrazas ("Stock Portfolio Risk-Return Ratio Optimisation Using Grey Wolf Model") propose a risk-return ratio optimization model for stock portfolio that makes it possible to screen the adequate equities for inclusion into the investment portfolio and set its capital allocation ratio. A two-stage model is proposed in which, first, the selection of the initial set of equities is done by using the Self-Organizing Maps (SOMs) to identify a set of the most influential factors to be used as the input variables for the SOM, and second, to find the weight-based ratios for the capital to be distributed among the portfolio equities. The author used the nature-inspired Grey Wolf Optimization (GWO) metaheuristic. Tests are performed on a set from the S&P500 companies. The new model outperforms the traditional approaches.

Li Zhong, Oleksandr Shcherbakov, Dennis Hoppe, Michael Resch and Bastian Koller ("Toward Seamless Execution of Deep Learning Application on Heterogeneous HPC Systems") are concerned with deep learning for extremely large data or very complex neural network architectures. This implies a need for the parallelization of deep learning algorithms and frameworks, and a need for the use of high-performance computing (HPC). The authors demonstrate methodologies for applying deep learning on HPC and present how AI techniques can successfully be integrated with classical simulation codes, as well as they show a comprehensive and critical overview of training neural networks on HPC while successfully leveraging data, model, pipeline, and hybrid types of parallelism. The applications are shown for combining a multi-task neural network with a typical FEM simulation to determine material characteristics, and for the segmentation of high-resolution satellite images. The results obtained are encouraging.

We hope that the coverage of many challenging and interesting problems considered in the volume in the contributions that provide both critical analyses of what has already been done, inspiring analyses and remarks, and new and original solutions will be of much interest and use for a wide research community, as well as practitioners.

We wish to express our deep gratitude to the contributors for their great works. Special thanks are due to anonymous peer referees whose deep and constructive

remarks and suggestions have greatly helped improve the quality and clarity of contributions.

And last but not least, we wish to thank Dr. Tom Ditzinger, Dr. Leontina di Cecco, and Ms. Zainab Liaqat for their dedication and help to implement and finish this important publication project on time, while maintaining the highest publication standards.

Vilnius, Lithuania Gintautas Dzemyda
Vilnius, Lithuania Jolita Bernatavičienė
Warsaw, Poland Janusz Kacprzyk

Contents

Computational Thinking Design Application for STEAM Education

Anita Juškevičienė, Arnold Pears, Tatjana Jevsikova, and Gabrielė Stupurienė

Abstract **Motivation**: Integrating STEAM education and Computational Thinking (CT) provides the skills of analysis, problem-solving and creativity enhancement necessary to twenty-first century citizens. STEAM education can also be seen as a bridge, reinforcing the link between science, schools and industries. Teachers play an important role as mediators and mentors. The difficulties faced by teachers are not only a lack of knowledge of specific disciplinary terms but also the context in which they are applied, such as the computational context. **Problem**: In order to clarify the context for teachers, and extend their competence beyond knowledge of basic concepts and terminology, guidance on CT and STEAM education integration in schools has emerged as a pressing problem. **Solution**: A Design Thinking and CT practices taxonomy interaction framework is proposed, providing scaffolding to teachers as they struggle to understand the context of CT implementation in STEAM education. **Results**: The proposed framework provides concrete guidance to educators in planning class activities, and choosing suitable educational practices in order to engage students. **Implication**: The results support educators looking for guidance in the integration process and those seeking to incorporate integrated aspects of students' STEAM learning into teaching practice.

1 Introduction

Over the last decade, much attention has been paid to the development of Computational Thinking (CT) and its importance has been emphasized. It is important to find out what to integrate in terms of Computational Thinking, as well as what learning content topics and activities to use in the classroom. It is also clear that motivating

A. Juškevičienė (✉) · T. Jevsikova · G. Stupurienė
Institute of Data Science and Digital Technologies, Vilnius University, Vilnius, Lithuania
e-mail: anita.juskeviciene@mif.vu.lt

A. Pears
KTH Royal Institute of Technology, Stockholm, Sweden

G. Dzemyda et al. (eds.), *Data Science in Applications*,
Studies in Computational Intelligence 1084,
https://doi.org/10.1007/978-3-031-24453-7_1

and engaging learning activities providing instant results and feedback are critical to maintaining the interest of the current school generation.

In addition, it is unthinkable that learners in the digital age should not have access to the widest possible set of computing skills in order for them to exercise agency in relation to of twenty-first century skills, such as creativity, critical thinking and problem solving.

The term "Computational Thinking" was introduced by Jeanette Wing when it was published in 2006 [1]. It is argued that CT involves problem solving, systems design, and understanding of human behavior using basic computer science concepts. The theory of Computational Thinking is closely related to problem-based learning and constructionism, drawing heavily on the legacy of Seymour Papert [2].

In addition, recent research shows that students' ability to process data meaningfully increases their ability to think in computational terms. In this way, they consistently visualize, reconstruct, synthesize, and analyze data. Data science, in turn, creates a real-world context that allows learners to explore data handling concepts such as selection bias, causal and correlation problems, responsible use of information, and contextual knowledge of real-world problems [3].

Easy-to-use instructions are needed to help educators integrate computing into the classroom and curriculum. However, researchers have only just begun to explore this area, and advice and study outcomes providing a link between CT theory and school teaching practice are currently scarce. Therefore, a major effort is needed to provide methods which can support the introduction of CT into school education at all levels.

Research shows that it is not enough to define CT and outline its main components [4]. It is also important to explain the concepts that are associated with the application of CT. The World Economic forum [5] in a recent analysis has provided an overview of the main concepts related to computation that arise in education due to technological innovations and how these concepts are interconnected. Digital fluency and STEM skills is one the parts of the education area. CT is also connected to Future computing and a broader computing milieu embracing concepts like virtual and augmented reality, artificial intelligence and biotechnologies. This means that a future citizen, in order to stay competitive in the future job market, should be familiar with the above-mentioned concepts, as adoption of technology increases.

In the efforts being made to address these needs educational institutions have an important role in preparing future entrepreneurs through appropriate curricula and methods that appeal to today's learners. In line with [5], an influential study [6] shows that there is a need for a nuanced understanding of CT, calling for extension and contextualization of CT to include explicitly Machine Learning (ML) and AI (CT 2.0).

Munasinghe et al. [4] have also recently showed that there is a number of concepts related to CT that should be further explained to teachers.

Thus, CT has been actively promoted in schools in an integrative approach, helping to enhance our definition of STEAM (Science, Technology, Engineering, the Arts and Mathematics) education. There is a number of benefits that STEAM education provides, such as enhancing the skills of analysis and problem-solving and creativity enhancement [7, 8].

However, there are still some challenges and issues in STEAM related activities integration to school and within STEAM subjects. Researchers and educators are consequently re-examining the importance of STEAM-related activities and programs, specifically developing CT skills and the integration of maker education where learners imagine, design and create projects that combine learning content with practical hands-on applications. Our approach to STEAM also places emphasis on collaboration and integration. This framework draws on the work of [9], in particular in terms of how CT could be positioned in the curriculum by design and implementation of an integrated STEM and CT lesson. Yang emphasizes the role of a problem-based process for integrating CT problems and solutions into after-school programs using hands-on inquiry activities which were exciting and engaging for students. Moreover, programming and using physical computing objects (robots) enable students to engage in scientific practices and in a such a way learn some engineering aspects (such as, bridge design) as well as gain satisfying experiences. Physical computing in the learning process is achieved through designing and developing the shareable construction (e.g., robot, musical composition, poem) by collaboration. Physical computing in our context involves creative arts and design processes and brings together hardware, such as sensors, LEDs, servos, and software components [10].

The design process takes place through design thinking (DT) that realizes learning through experience and complex problem solving for motivation, openness to new ideas and creative thinking in the learning process [11]. Thus, in the literature, the need to adapt design to learning is often emphasized by scholars and practitioners [12]. Design thinking is a learning design that facilitates a constructive way of learning due to its inherent characteristic of developing certain skills [11]. There are a number of benefits learners receive from integrating STEAM learning with computing by modeling various phenomena, such as Computational Thinking learning [13]. However, researchers and educators still face the challenge of defining Computational Thinking and getting a theoretical grounding for what form it should take in school. One of the possible solutions proposed by [14] is to develop CT taxonomy. In such a way CT can be embedded in the STEAM subjects' context. CT development is then made available through STEAM related activities integration in school especially through hands-on projects [15].

The main question addressed by STEAM researchers and educators is not why to integrate CT, but how. The lack of literature addressing the relationship between CT and DT [16] led to identifying possible implications for how Design Thinking and Computational Thinking are taught within education. Our previous study [17] attempted to combine the above-mentioned theories.

Thus, in order to learn/teach CT we propose the Computational Thinking Design approach by merging DT and CT taxonomy approaches through physical computing. Learners (including teachers) may learn CT by following Design Thinking phases presented by [11] accomplished by CT taxonomy practice implemented through physical computing activities that bring computer concepts from the screen into the real world for learners to interact with [18].

The **aim of this research** is to study DT application for STEAM and present a modified version of Computational Design Thinking (CTD) framework, further recommendations and application examples.

For this purpose, the research questions were posed:

1. What kind of DT and CT frameworks exist in the STEAM context?
2. How CT practices are embedded throughout the DT phases in the STEAM context?

The remainder of the paper is organized as follows. The next section covers the background literature on CT approaches and Design Thinking models. The third section presents CT taxonomy and its association with Design Thinking in Computational Thinking Design framework. The fourth section presents CTD framework application examples. The paper concludes with a discussion of the main findings and outlines our ideas for future work.

2 Background

2.1 Computational Thinking Approaches

Computational Thinking is widely discussed among researchers and practitioners. Many different approaches and definitions are proposed by researchers based on what they are focusing on. Despite the abundance of existing studies, researchers and educators still face the challenge of defining CT and finding the right theoretical underpinning for the form it should take in schools. According to systematic literature review of period 2016–2021 on CT in compulsory education, researchers are dividing CT into different categories and defining it accordingly: generic, operational or model, educational and curricular definitions [19]. Generic definitions relate to computing (including programing) disciplines, however could be independent. In the second category, operational definitions determine the fundamental concepts and practices of CT. Educational category definitions relate to educational frameworks applicable in computer science and computing areas. For example, operational definition is used in the study [20] where the educational framework of CT involves solving problems, designing systems, and understanding human behavior by drawing on the concepts fundamental to CS. It covers a set of broadly applicable problem-solving skills, including abstraction, decomposition, pattern recognition. In contrast, an operational, or model CT definition, presented by [14] classified CT into four major categories: data practices, modeling and simulation practices, computational problem-solving practices, and systems thinking practices.

From the practical point of view, definitions of CT themselves mean nothing to teachers. Zhang and colleagues [21] emphasize that the students' CT learning can be compromised due to teachers' limited knowledge of CT. Therefore, teachers need to build their capacity in relation to fundamental CT concepts and pedagogies.

Results of some recent studies make it easier for teachers to understand CT topic and definition, however there are still some CT terms that teachers are struggling with. Recent study of [4] showed there are still some CT terms difficult to understand for teachers, such as, iteration, control structures, HCI heuristics, and comparative and logical operators. The aim of their empirical study was to understand the nature of teachers' understanding of computational terms related to Computational Thinking concepts.

Teachers need to be supported by professional development and clear step-by-step toolkits that enable a balance between the focus on CT concepts, teaching practices and identifying how CT can be embedded in other subjects [1]. The literature provides several insights on teacher professional development on CT in various settings, although more research is needed on how to support teachers in implementing and designing learning experiences that are explicitly focused on subject- and content-based learning [20].

There have been previous attempts to go beyond the definition and identification of key concepts to look at CT practices [22] proposed a three-dimensional framework of CT: computational concepts, computational practices, and computational perspectives. The dimension of CT concepts refers to the computational concepts that learners develop in programming, such as iteration, parallelism, CT practices—problem-solving practices that students continuously demonstrate in the programming process, such as debugging projects or remixing others' work, and CT perspectives—to self-understanding and relationships with others and the world of technology that they create by expressing, connecting and questioning in programming, such as, designers form about the world around them and about themselves.

As mentioned previously, [14] had a different broader view to Computational Thinking. With the aim to integrate CT into STEM they developed a comprehensive CT taxonomy. Weintrop et al. propose to simplify access to CT conceptual material by breaking it down into four major categories: data practices, modeling and simulation practices, computational problem-solving practices, and systems thinking practices. Each of these categories is composed of a subset of five to seven practices (Table 1).

In such a way CT can be embedded in the science context and in turn promote learning of STEAM content [23]. We argue that this approach creates greater opportunities for CT learning, as STEM subjects are more widely taught than computer science or programming, which are traditionally related to CT education.

2.2 CT in STEAM Context

As we have already noted in our proposal for an integrated approach to STEAM education, Computational Thinking is not exclusively CS focused approaches, but also claims relevance in the teaching of other subject matter (e.g., science or technology) other than CS, and should be integrated into the learning experiences [24].

Table 1 CT taxonomy [14]

CT PRACTICES: data practices	Definition
Collecting data	Systematic data collection through observation and measurement
Creating data	To define computational procedures and run simulations by using computational tools (physical devices or software packages) to generate data in order to investigate phenomena that cannot be easily observed or measured or that are more theoretical in nature
Manipulating data	To manipulate (sorting, filtering, cleaning, normalizing, and joining disparate datasets) data in order to make meaning of them
Analysing data	Using computational tools to analyze data by using different strategies, such as, patterns or anomalies recognition, definition of rules to categorize data, and identify trends and correlations
Visualising data	To use computational tools to produce visualizations (analyzing and sharing data) that convey information gathered during analysis
Modeling & simulation practices	
Using computational models to understand a concept	Models support the inquiry process by recreating phenomena in environments that support systematic investigation and give the user far more control than would be possible in the natural world. Interaction with a model helps to advance understanding of a concept demonstrated by it
Using computational models to find and test solutions	Computational models can also be used to test hypotheses and discover solutions to problems, to find, test, and justify the use of a particular solution, to apply the information gained through using the model when appropriate
Assessing computational models	To articulate the similarities and differences between a computational model and the phenomenon that it is modeling, this includes raising issues of threats to validity as well as identifying assumptions built into the model
Designing computational models	Designing a model involves making technological, methodological, and conceptual decisions by defining the components of the model, describing how they interact, deciding what data will be produced by the model, articulating assumptions being made by the proposed model, and understanding what conclusions can be drawn from the model

(continued)

Table 1 (continued)

CT PRACTICES: data practices	Definition
Constructing computational model	To implement new model behaviors, either through extending an existing model or by creating a new model either within a given modeling framework or from scratch
Systems Thinking practices	
Investigating a complex system as a whole	To pose questions about, design and carry out investigations on, and ultimately interpret and make sense of, the data gathered about a system as a single entity. For example, define and measure inputs and outputs of the system, to black box the details of the underlying systematic interactions by using models and simulations
Understanding a relationship within a system	To identify the constituent elements of a system, articulate their behaviors, and explain how interactions between elements produce the characteristic behaviors of the system
Thinking in levels	To identify different levels (micro and macro) of a given system, articulate the behavior of each level with respect to the system as a whole, and be able to move back and forth between levels, correctly attributing features of the system to the appropriate level
Communicating information about a system	To communicate information learned about a system in a way that makes the information accessible to novice viewers who do not know the exact details of the system from which the information was drawn. It often involves developing effective and accessible visualizations and infographics that highlight the most important aspects of the system in combination with features of a system prioritization, design of intuitive ways to represent it, and identification of what can be left out of the visualization without compromising the information being conveyed
Defining system and managing complexity	To define the boundaries of a system (for example, classroom system, group of galaxies or the human genome) so that the system can later be used as a domain for investigating a specific question as well as to identify ways to simplify an existing system without compromising its ability to be used for a specified purpose

(continued)

Table 1 (continued)

CT PRACTICES: data practices	Definition
Computational Problem-Solving Practices	
Preparing problems for computational solutions	To employ strategies (decomposing problems into sub problems, reframing new problems into known problems for which computational tools already exist, and simplifying complex problems so the mapping of problem features onto computational solutions is more accessible) toward reframing problems into forms that can be solved, or at least progress can be made, through the use of computational tools
Programming	To understand, modify, and create computer programs (encoded instructions for computers to execute) and use these skills to advance scientific and mathematical pursuits
Choosing effective computational tools	To articulate the pros and cons (considering the functionality it provides, its scope and customizability, the type of data the tools expect and can produce) of using various computational tools and be able to make an informed, justifiable decision
Assessing different approaches/solutions to a problem	To assess different approaches/solutions to a problem based on the requirements and constraints of the problem and the available resources and tools. Also, to consider cost, time, durability, extendibility, reusability, and flexibility while getting correct results
Developing modular computational solutions	To develop solutions that consist of modular, reusable components and take advantage of the modularity of the solution in both working on the current problem and reusing pieces of previous solutions when confronting new challenges
Creating computational abstractions	To identify, create, and use computational abstractions (to conceptualize and then represent an idea or a process in more general terms by foregrounding the important aspects of the idea while backgrounding less important features) while working toward scientific and mathematical goals. For example, when writing a program, generating visualizations of data to communicate an idea or finding, defining the scope or scale of a problem, or creating models to further explore or understand a given phenomenon

(continued)

Table 1 (continued)

CT PRACTICES: data practices	Definition
Troubleshooting and Debugging	To identify, isolate, reproduce, and ultimately correct unexpected problems encountered when working on a problem, and do so in a systematic, efficient manner. Troubleshooting—systematically testing the system to isolate the source of the error

CT plays a vital role in the fields of science, technology, engineering, and mathematics (STEAM) providing access points and can be used in the process of guiding student inquiry, dialogue, and critical thinking. The STEAM subjects also provide a natural context for CT learning [25]. The findings of previous studies showed that there is a significant positive correlation between STEM education and CT skills. CT is not just subject-specific skills, but a comprehensive set of thinking skills that can be applied in any STEAM-related field [26]. STEAM is associated with CT as its development is possible through STEAM related activities [15].

In order to find out what other areas can be considered to be related to CT and STEAM association, a systematic literature review was conducted. An initial pool of relevant research was gleaned by searching papers in the *Web of Science* DB using "Computational Thinking in STEM or STEAM" keywords; Publication Years: 2022 or 2021 or 2020; Web of Science Categories: Multidisciplinary Sciences or Education Educational Research. Based on these criteria 79 relevant articles were found published in 2020, 2021 and the first few months of 2022. These papers' titles and abstracts were analyzed with VOSviewer software in order to develop a concept map with the objective of identifying what concepts might have been missed during initial literature review. The resulting clusters show that CT in STEAM context is related to robotics, design, programing, STEM and problem solving in these articles. Figure 1 was developed by using keywords which have been extracted through an analysis of the literature review made.

This result demonstrates that STEAM (in the context of its integration with CT) is interconnected with new emerging areas, such as, deep learning, machine learning, internet of things. As new computational related themes emerge, the set of CT elements expands. For example, ML-enhanced Basic CT (CT 2.0) presented by Tedre and co-authors [6] is the extension of Basic CT, which involves machine learning area concepts and explains intelligent systems that are mostly based on smartly designed technology trained with copious amounts of data.

2.3 Design Thinking

Every day learners face complex real-life problems, analyzing and evaluating them in order to act in a responsible and focused way. Design Thinking puts into practice what

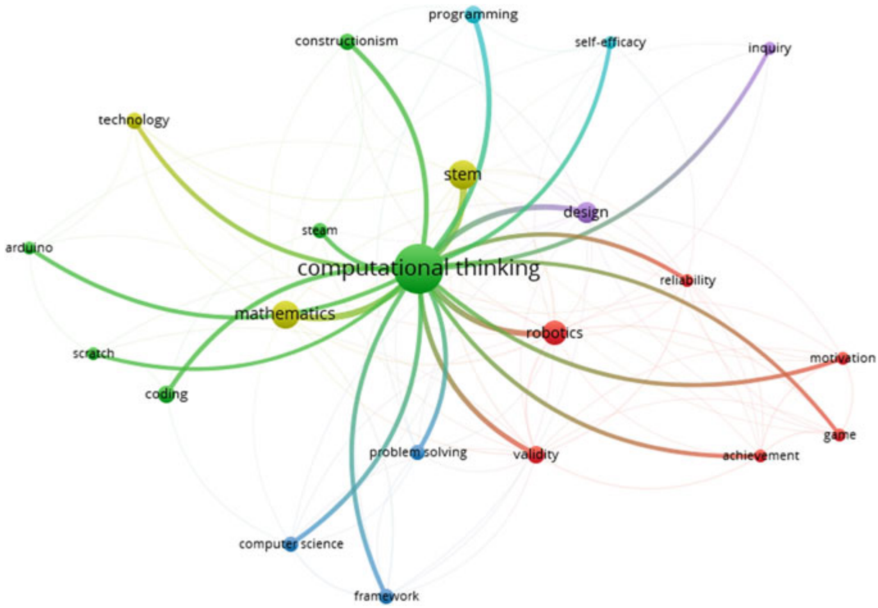


Fig. 1 Map of additional CT related concepts

is recommended in theory. In particular, learning through experience and solving complex problems, which can be applied to all age groups.

The DT term originated from the 'design as profession' area. However, now it is used as a meta-disciplinary methodology. Thinking like a designer involves different kinds of abilities and competencies in different fields of knowledge [27]. It is therefore not at all surprising that DT models are implemented in education, for example, Stanford 'D.school' model [28], other focuses on how it works for STEAM [29, 30].

Design Thinking usually has six phases [11]: (1) Understand and Observe (Expanding), (2) Synthesis (Consolidating), (3) Ideate (Expanding), (4) Prototype (Consolidating), (5) Test (Expanding), (6) Iteration (see Fig. 2).

Depending on the source the first phase Understand sometimes is called Empathize, the second Synthesis—Define [31, 32]. The first phase goal is to find the relation between the problem and its context in order to set in the challenge,

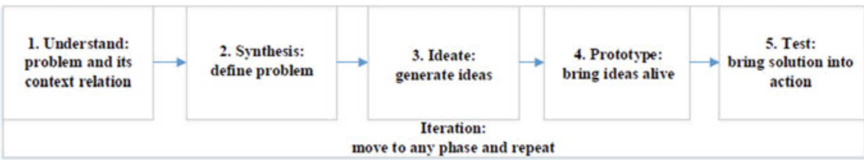


Fig. 2 Design Thinking process based on [11]

be able to immerse oneself in the user's experiences in order to get insights into what people think and feel. This initial attempt to understand the context and nature of a problem is a critical component of the Design Thinking process. The second step defines the problem and its context. Specific problems can be interpreted from different perspectives, so in this phase all the information is synthesized into a clear idea in order to achieve a deep understanding of the users and the design space and, based on this understanding, to generate an actionable problem statement. Ideate phase means the idea generation by (a large quantity of ideas and diversity among those ideas) by Applying existing knowledge, collaborating, and turning the result into a workable problem-solving idea. The fourth stage, Prototype, is the process of creating a tangible object in order to test it and share the abstract idea with others, such low-resolution implementations can be storyboards, role-plays, physical objects, or services. Once the design process is underway, you can move on to the next stage—testing, which includes developing the solution. Testing focuses on the solution, showing how well the problem has been understood and how well it meets the user's needs. Phase iteration refers to the cyclical and iterative nature of the Design Thinking process, where steps are repeated to move from one phase to another as mu.

Design Thinking can also serve as a learning design because of its qualities in developing certain skills that are prerequisites for a constructive way of learning: motivation for exploration, openness to new ideas, creative thinking and other metacognitive competences [11]. DT as a team-based learning process offers support for teachers towards practice-oriented learning in projects. It is also a meta-disciplinary methodology which supports teachers through a formalized process. Design Thinking can give concrete recommendations for distributing a complex phenomenon without abstracting too much, but still being digestible for the student and implementable for the teacher. Design, or Design Thinking, may provide a guiding framework to support an expanded view of STEAM teaching. Design or Design Thinking can help broaden the approach to STEAM teaching. It also provides a framework for teachers to develop more creative and interdisciplinary practice—both as a basis for their thinking and as part of the students' STEAM experience [29].

Given the current need to find innovative ways to figure out how to change teaching practices to meet current and emerging global complexities (i.e., the global pandemic, but also climate change, income inequality, the digital divide), it is essential to have a clear focus on global cooperation. Teachers can benefit from teaching approaches that integrate educational technology within a Design Thinking framework, providing them with engaging, authentic, and meaningful experiential learning opportunities [33].

2.4 Design Thinking in STEAM Context

In recent years, researchers have shown a significant interest in the STEAM concept. Nevertheless, many teachers still find it difficult to integrate STEAM into school subjects. Various solutions have been proposed to address this problem and one of them is Design Thinking (DT) as a way to bring several disciplines together.

Unless existing theoretical and empirical studies on DT in STEAM are tightly interrelated, they address several main directions:

- DT as an approach to holistic learning, STEAM subject integration and development of essential twenty-first century skills, e.g., [29, 32, 34, 35].
- DT as an approach to develop creativity, artistic and "soft" skills within STEAM, e.g., [32, 36, 37].
- Development and an effect of practical DT-based activities in STEAM, e.g., [30, 37, 38].
- Raising STEAM fields' attractiveness for students, especially for female students, e.g., [39].
- DT and STEAM within curricula and teacher training, e.g., [29, 32, 37, 40].
- Learning via reverse engineering through the DT approach, e.g., [41].

DT in STEAM provides a holistic approach which integrates learning about the natural world with constructed world, the social with the cultural, and the economical with the environmental [35]. Henriksen [32] suggest that STEAM intends at framing and solving problems and involves blurring disciplinary boundaries—it involves thinking creatively and working on projects that aim at real-world inquiry, while DT provides a suitable framework to streamline this disciplinary integration.

Based on observations, DT helps to develop insights that can turn into products or services that improve our lives [29, 34]. This closely aligns with the aims of a STEAM education to help students to develop twenty-first century skills, such as innovation and problem-solving [36].

The demonstration on how to use the DT approach to put into practice the STEAM knowledge by students was examined in [30] focusing on the design of microcontroller-based systems, engineering and technology aspects. The results of the study in robotics education, based on pretests and posttests in STEAM subject area (e.g., chemical reactions) as well as on interviews, have shown that the DT framework was found effective for robotics tasks, socioemotional skills and holistic development [42]. Researchers provide educational cases and examples which are positioned as "STEAM by Design" movement, mixing art, design and the environment across traditional K-12 subjects, aligned with existing curricula standards [37].

The idea to use Design Thinking as a guide for teachers in STEAM curriculum design was proposed and tested in [29, 32]. The study analyzes the cases of how DT helps teachers create STEAM-based curricula. Dotson et al. [40] suggest a framework for DT within STEAM curricula integration and provide evidence on effective approaches of young STEAM teachers and active students as teachers' involvement in DT-based STEAM teaching activities. As a student-centered approach, DT offers teachers a way to invoke problem-solving skills and creativity, as related to the current education goals of developing the mindsets of innovators [38].

In STEAM fields like engineering, female students show lower levels of self-efficacy than their male peers (e.g., [43]). The study by [39] provides evidence that after a series of DT-based STEAM workshops, there is a significant increase of interest in engineering, creative confidence, empathy and prosocial perceptions of female students. After 3-day intervention girls' perceptions of STEAM became considerably broader, and there were notable changes in career aspirations related to the STEAM field. Previous studies also provide some evidence of impact of DT-based Science courses on student inclusion: the results have demonstrated the largest benefits for students who were the most socioeconomically disadvantaged students and those who scored lowest on the pretest [38].

While engineering processes include elements of design, the DT framework embraces a component of empathy through which designers consider the needs and values of the users intended to use the designed object [38]. However, unlike most other engineering process approaches, this empathy-related component of DT incorporates artistic elements of personal expression. Therefore, despite the well-known differences in "design" and "art", the researchers emphasize an important role of DT in Arts education within STEAM [32, 36]. So, DT helps to extend the STEM approach into STEAM.

Most of the studies involving DT framework for STEAM focus on DT for forward engineering (i.e., how to "create" a product). However, some studies show the potential of DT in "reverse" tasks as well, where students engage in "taking a final product [...] to understand its functionality, [and obtaining] design and other useful information" [41]. Ladachart et al. [41] in their study demonstrated that reverse engineering activities help students to better perceive their own characteristics of DT, especially the aspect of human-centeredness.

However, Computational Thinking within DT in STEAM is still not a widely researched area. The next section is intended to fill the gap and propose an integrative framework.

3 Computational Thinking Design: Proposed Framework

Our Computational Thinking Design (CTD) framework combines CT practices taxonomy and DT model in order to propose engaging and motivating learning activities, to learn CT and STEAM content in a structured manner. Figure 3 presents the proposed framework visualization that can be used for educators in order to design CT activities for use in class.

In the first DT step, CT's data practices can be adopted. Pupils are provided with instructions dealing with how to collect related data, identify the necessary information from the background information provided by the teacher, and also be encouraged to search for relevant data on the web. Defining the problem is part of the process of forming an attitude towards the problem—our own and others'. The task formulation should therefore be contextualized, relevant and inspire the group, the pupil or the whole class to look for solutions. Thus, the second phase requires the practices for problem preparation, such as, decompose or reframe into a solvable form. The third step, ideation—the process of generating ideas, where the widest

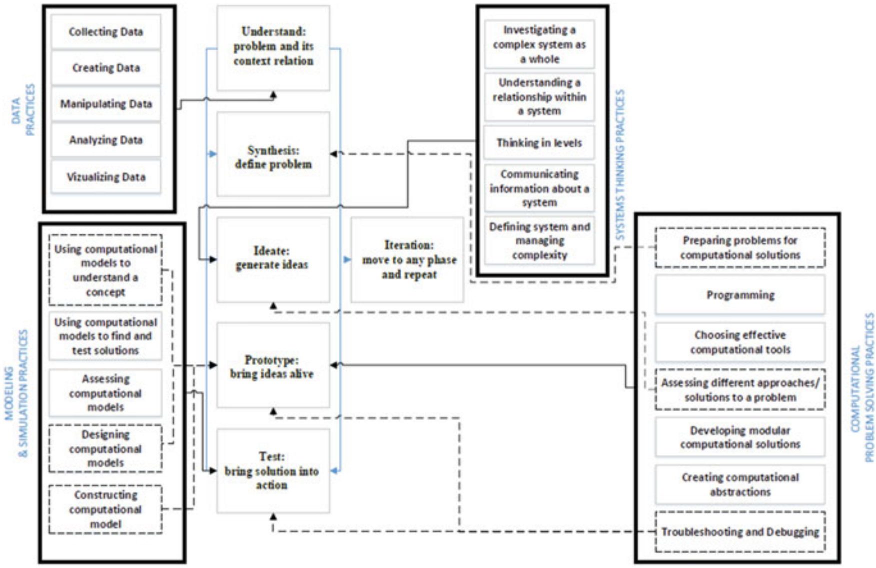


Fig. 3 Computational Thinking Design framework

possible range of possible solutions is created, involves a series of practices like, identify the main elements (characteristics and interaction) of possible solutions as a system, define its boundaries, and visualize the most important features and aspects of the system. Also, it is necessary to assess the different approaches and solutions to the problem in terms of cost, time, durability, extensibility, reusability and flexibility, and the likelihood of achieving a suitable result, at this phase Prototyping—the process of giving proposed solutions a form (often physical) that conveys essential features is the fourth step and requires practices for solutions development, create solutions made up of modular, reusable components and use the solution both to solve the current problem and to reuse parts of previous solutions when faced with new challenges. Such forms may include models that reproduce inquiring phenomena in order to better understand a concept demonstrated by it. Hence, this process also requires model design and construction practices. The last phase, testing—finding out what works and what doesn't, with the aim of improving the proposed solutions includes application of information gained out of using the model, threats validation and assumptions included in the model identification.

3.1 *Applying the Framework*

The context of the problem is related to predatory birds who are destroying the harvest. In grandmother's garden, crows are a constant nuisance: they eat the berries, peck around in the fields, and make a lot of noise. The following tasks are then defined related to each of the 5 DP phases:

1. Try to imagine how Grandma feels, how disappointed she is when she loses her harvest and how much time she spends tidying up the garden. Think about if she is bothered by the noise the crows make. Use the empathy map to do this activity.
2. Define clearly what the problem is. For example, how to scare the crows away? Do this task by using a POV sentence.
3. Using the theory given by the teacher and the information found on the internet, list what crows don't like and how to make the environment uninviting (use the rules of the brainstorm method). A possible solution—build a scarecrow.

   Please note, grandma can't be in the garden all the time to see if the crows have flown in, so the scarecrows need to be automated.
4. Build a prototype of the proposed solution using the guidelines for good solutions in prototyping.
5. Test the prototype as far as possible, evaluate the prototype you have built, and discuss what are the strengths and shortcomings of the proposed solution and what could be improved? Use the product testing guidelines to complete this exercise.

3.2 Implementation Examples

The proposed model was implemented during a GEM ("GEM—Empower Girls to Embrace their Digital and Entrepreneurial Potential[1]") girl's summer school in 2021. The main purpose of the summer school, organized in Lithuania (August 16–19, 2021), was to empower 11–15 years-old girls to embrace their knowledge and skills in various STEM subjects (mathematics, informatics, physics, chemistry, biology, engineering, and technology) as well digital and entrepreneurial potential, pursue related studies and careers and take part actively in Europe's research, innovation and especially digital processes. Figure 4 and Table 2 summarizes the CT practices adopted for these tasks.

The following is a brief description of which CT practices were implemented at each stage of the DT during summer school.

Phase 1. Understand. Girls searched information on the internet and also discussed questions from the Empathy map:1. Which consumer are we trying to empathize with? 2. What decisions do consumers make? 3. What do consumers see? 4. What are consumers talking about? 5. What do consumers do and what is their lifestyle? 6. What do consumers hear most often? 7. What do consumers think or feel?

Girls were discussing these questions in order to understand the problem and propose solution.

Phase 2. Synthesis. POV sentence method was used in order to develop the problem: **Grandmother** needs something that can **scare the crows** as she is spending

[1] https://icse.eu/gem-empower-girls-to-embrace-their-digital-and-entrepreneurial-potential/.

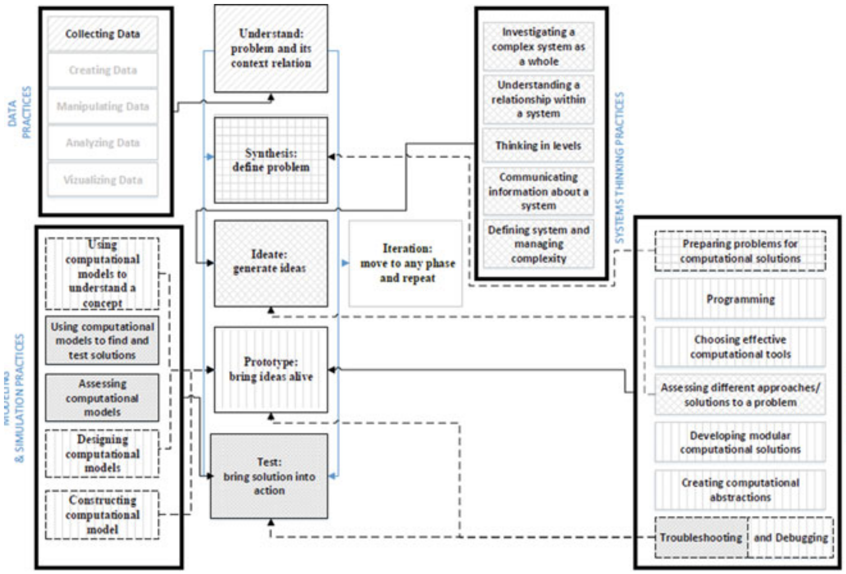


Fig. 4 Computational Thinking Design framework implementation example

a lot of **time** tidying up the garden, also she is bothered by the noise the crows make and **losing** all her harvest.

Phase 3. Ideate. Information search and Brainstorming gives the following recommendations:

1. Close the rubbish and compost bins;
2. Do not have nest boxes and branches suitable for crows' nests;
3. Use models of crows, eagles and snakes (preferably moving);
4. Use a laser or flashing lights to shine when the crow arrives;
5. Hang reflective objects (e.g., old CDs, aluminum foil plates, reflectors);
6. Use loud sound signals.

Phase 4. Prototype. Prototyping was done by drawing a prototype model on the poster (Fig. 5) and implementing it by Arduino microcontroller. The models present the main parts of the system and describe what each part is doing.

During GEM summer school, girls developed scarecrows. In Fig. 6, three different implementations with Arduino microcontrollers are presented.

Pictures in the top row used PIR motion sensors in order to detect and buzzers to set an alarm if the movement was detected. The picture at the bottom shows the scarecrow, which additionally has LEDs that lights as a visual warning signal.

Phase 5. Test. Testing scarecrows shows that light-based scarecrow was not so useful as crows quickly become accustomed to the same stimulus and are no longer afraid of it. It is recommended to try to create a scarecrow with as many lights as possible and with different flashing and color intensities. Sound signal based scarecrows were also not useful. Testing showed that sounds of gunshots, loud music

Table 2 CT practices implementation (description) example

CT PRACTICES: data practices	Implementation	DT phase
Collecting data	The data was collected on the internet and by discussion using Empathy map questions	Understand
Creating data	No computational tools were used for this purpose	n/a
Manipulating data	No computational tools were used for this purpose	n/a
Analyzing data	No computational tools were used for this purpose. Data analysis was done by discussing	n/a
Visualizing data	No computational tools were used for this purpose. Visualization was done on poster	n/a
Modeling and simulation practices		
Using computational models to understand a concept	The desired characteristics of the system (scarecrow) were developed in pairs through Arduino based models and sensors	Prototype
Using computational models to find and test solutions	Developed Arduino based scarecrows were used to imitate the crows getting closer to the tree and to test how it will respond	Test
Assessing computational models	The girls identified the assumptions made for the model and tried to assess similarities and differences between the model and the scarecrow operating principles in order to understand how the build model relates to the process of repelling crows	Test
Designing computational models	Girls defined the components of scarecrow model, described each model element's purpose and interaction, what parts are used for data input and what for particular output	Prototype
Constructing computational model	Before the activities, the girls were introduced to the Arduino kit and the sensors, including the purpose, and working principles of each sensor. The participants then have chosen which sensors to use for which purpose, taking into account the needs of the model they were building or/and the existing solutions found on the internet in order to implement desired behaviors of models	Prototype

(continued)

Table 2 (continued)

CT PRACTICES: data practices	Implementation	DT phase
Systems thinking practices		
Investigating a complex system as a whole	Girls described each scarecrow model element's purpose and interaction, what parts are used for data input and what for particular output in order to understand the system they are building as a whole entity	Ideate
Understanding a relationship within a system	Girls listed the main parts of the Arduino based model, what parts are responsible for, how they interconnected and determines the inherent behavior of the system	Ideate
Thinking in levels	Participants had to identify the parts of proposed solutions, what each part does and also to present as a whole—in model	Ideate
Communicating information about a system	The girls were asked to draw a model of the scarecrow-system they were developing, to discuss it in pairs and then to present it to the whole group, identifying what to include and what not to include in the chosen visualization, in order not to lose the intrinsic information in the process of its communication	Ideate
Defining system and managing complexity	The girls had to clearly define the area to be investigated, what they were investigating, in what context. They tried to design systems that would work in the garden and in the park and elsewhere, but without reducing its ability to be used for the specific purpose of deterring crows and other creatures	Ideate
Computational problem-solving practices		
Preparing problems for computational solutions	In order to simplify a complex problem, the girls used the POV sentence method to clearly define and transform the problem into form to be solved	Define
Programming	Girls slightly modified available codes for the desired performance of the model they developed	Prototype

(continued)

Table 2 (continued)

CT PRACTICES: data practices	Implementation	DT phase
Choosing effective computational tools	The girls reasoned about the choice of the elements from the Arduino kit considering the functionality it provides: the sensors needed and the outputs needed	Prototype
Assessing different approaches/solutions to a problem	The girls tried to assess different ways of solving the problem under the discussion process, whether automated solutions are needed or whether certain parts are needed	Ideate
Developing modular computational solutions	The girls' model is an automated scarecrow. It is based on an Arduino microcontroller and accessories that can always be used in other implementations, such as transfer from the box to the hawk imitation model. But it is possible to reuse pieces, reduce or increase the number of sensors to confront new challenges	Prototype
Creating computational abstractions	To identify, create, and use computational abstractions (to conceptualize and then represent an idea or a process in more general terms by foregrounding the important aspects of the idea while backgrounding less important features) while working toward scientific and mathematical goals. For example, when writing a program, generating visualizations of data to communicate an idea or finding, defining the scope or scale of a problem, or creating models to further explore or understand a given phenomenon. The girls created computer abstractions by drawing a model and expressing the key aspects of the idea, used abstraction to read circuit diagrams, and of course, to construct their models	Prototype

(continued)

Table 2 (continued)

CT PRACTICES: data practices	Implementation	DT phase
Troubleshooting and Debugging	The girls created computational abstractions by drawing a model and expressing the key aspects of the idea, used abstraction to scan circuit diagrams, and of course constructed their models. During code editions in the prototype development phase the debugging was used in order to correct encountered mistakes. During the testing phase some implementation solutions were changed as the system did not work properly or as it was intended. For example, the buzzer was hidden and couldn't be heard, or the light bulbs were badly attached and didn't work, or the motion sensor didn't cover a wide enough scanning area or even found that the model's capabilities should be extended	Prototype, Test

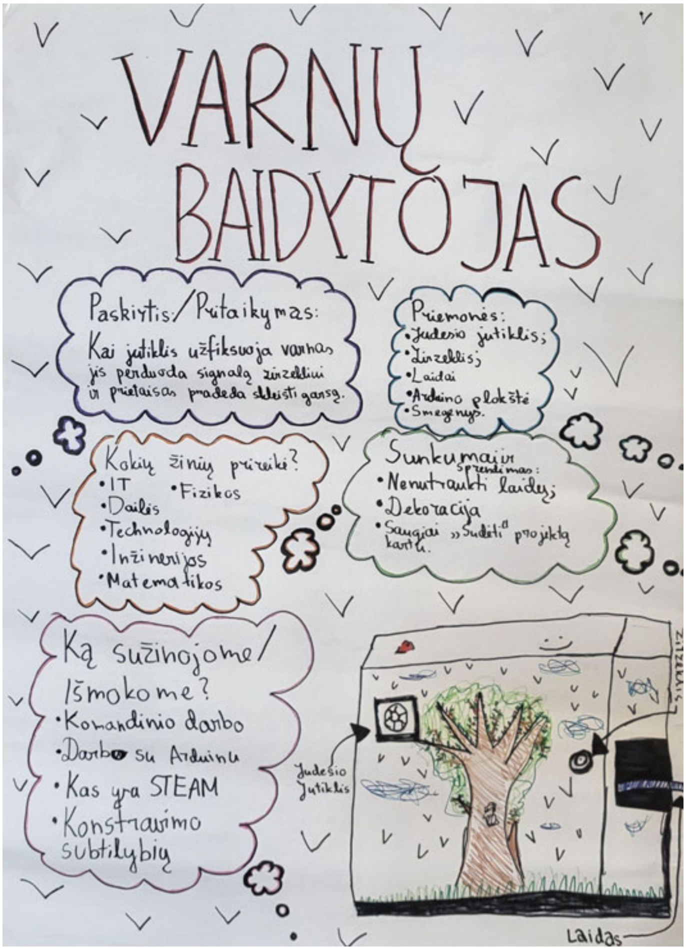


Fig. 5 An example poster

or eagle sounds were more appropriate to scare away crows, however, these also disturb many other stakeholders, including Grandmother. It is recommended to think about what other enemies crows have, what sounds or frequencies they might fear? Another recommendation that came as a result of testing is to combine as much as possible recommendations identified at ideation phase—prepare an unfriendly environment for crows-nests, hide food sources, and combine different discouraging signals in a random sequence. Scarecrow prototype/model implementations with Arduino allowed the girls to try out the proposed solutions. However, to create a real scarecrow, they would need to use light bulbs instead of LEDs and a much more powerful speaker than a buzzer, as well as additional parts providing access to audio tracks from the internet or a memory card.

Fig. 6 Scarecrow projects, student implementations

4 Discussion and Conclusions

During the recent years, there has been a great demand from researchers for STEAM education, as many studies have shown its benefits for students' motivation and effective learning.

Researchers and practitioners recognize an integrative role of the DT approach in STEAM educational context. Existing studies demonstrate successful DT application examples that help to engage students in STEAM, develop STEAM content knowledge and skills, contribute to developing a holistic view of the world, strengthening empathy and human-centeredness while developing a product.

As a consequence, CT and DT are widely observed in educational STEAM context by scholars presenting theoretical interpretations of these areas and emphasizing benefits it may provide for educational aims. However, the challenge to explore the practical integration of activities still requires research efforts.

While learning and teaching of CT has been extensively studied by researchers, the definitions and components expand year by year as new areas (e.g., machine learning) emerge, affecting the computational domain. Examples of the application of CT in education are also being explored, as CT provides students with a range of important skills needed for by the modern citizen. CT can be taught in a fun and engaging way and can be applied to all subjects. But there is a wide variety of tools and ways to teach CT, and a broader view of CT components is needed.

One of the suitable ways to teach Computational Thinking is by adopting the Design Thinking approach.

In our study, we propose an integrative CTD framework by adopting the DT approach in order to integrate CT in STEAM context. We present the results of an experiment based on summer school activities for girls', which demonstrate how the processes of specific integrative STEAM and CT practical activities can be aligned with the proposed CTD framework. The framework presented here also allows us to leverage the potential of Design Thinking in CT education drawing on a systematic review of recent research, and related taxonomic approaches.

Problem-solving methodologies through prototyping in STEAM using similar methods to ours, can be found in the literature. Yang et al. [9] study shows that integrating CT in cross-disciplinary practice (by mapping problem-solving processes with CT components), makes CT integration in K-12 classrooms and STEM curriculum more sustainable by learning.

In line with the observations of Wang [23] our study also shows that despite many theoretical interpretations of CT, the integration of CT into STEAM education remains a challenge. Teachers face many practical issues such as what activities and methods are effective in integrating CT into different STEAM contexts and how CT should be assessed. Our study makes a step by adopting the DT approach and CT practices within the STEAM context. Conducting empirical studies in order to assess the effectiveness of the proposed framework is positioned as a further research direction.

Acknowledgements This project has received funding from European Social Fund (project No 09.3.3-LMT-K-712-21-0068) under grant agreement with the Research Council of Lithuania (LMTLT).

References

1. Paniagua, A., Istance, D.: Teachers as designers of learning environments. Educational research and innovation. Paris: OECD Publishing, France. (2018)
2. Papert, S.: Mindstorms: Computers, Children, and Powerful Ideas. Basic Books, NY (1980)
3. Gilchrist, P.O., Alexander, A.B., Green, A.J., Sanders, F.E., Hooker, A.Q., Reif, D.M.: Development of a pandemic awareness stem outreach curriculum: utilizing a computational thinking taxonomy framework. Educ. Sci. **11**(3), 109 (2021)
4. Munasinghe, B., Bell, T., Robins, A.: Teachers' understanding of technical terms in a computational thinking curriculum. In: Australasian Computing Education Conference, pp. 106–114. (February 2021)
5. WEF2021. https://intelligence.weforum.org/topics/a1Gb0000000LPFfEAO?tab=publications
6. Tedre, M., Denning, P., Toivonen, T.: CT 2.0. In: 21st Koli Calling International Conference on Computing Education Research, pp. 1–8, Nov. 2021
7. Aithal, P.S., Aithal, S.: Innovation in B. Tech. Curriculum as B. Tech. (Hons) by integrating STEAM, ESEP & IPR features. Int. J. Case Stud. Bus. IT Educ. (IJCSBE) **3**(1), 56–71 (2019)
8. Wahyuningsih, S., Nurjanah, N.E., Rasmani, U.E.E., Hafidah, R., Pudyaningtyas, A.R., Syamsuddin, M.M.: STEAM learning in early childhood education: a literature review. Int. J. Pedagog. Teach. Educ. **4**(1), 33–44 (2020)
9. Yang, D., Baek, Y., Ching, Y.-H., Swanson, S., Chittoori, B., Wang, S.: Infusing computational thinking in an integrated stem curriculum: user reactions and lessons learned. Eur. J. STEM Educ. **6**(1), 04 (2021). https://doi.org/10.20897/ejsteme/9560
10. Przybylla, M., Romeike, R.: Physical computing and its scope-towards a constructionist computer science curriculum with physical computing. Inform. Educ. **13**(2), 241–254 (2014)
11. Scheer, A., Noweski, C., Meinel, C.: Transforming constructivist learning into action: design thinking in education. Des. Technol. Educ.: Int. J. **17**(3) (2012)
12. Kirschner, P.A.: Do we need teachers as designers of technology enhanced learning? Instr. sci. **43**(2), 309–322 (2015)
13. Grover, S., Fisler, K., Lee, I., Yadav, A.: Integrating computing and computational thinking into K-12 STEM learning. In: Proceedings of the 51st ACM Technical Symposium on Computer Science Education, pp. 481–482, Feb. 2020
14. Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., Wilensky, U.: Defining computational thinking for mathematics and science classrooms. J. Sci. Educ. Technol. **25**(1), 127–147 (2016)
15. Martín-Ramos, P., Lopes, M.J., da Silva, M.M.L., Gomes, P.E., da Silva, P.S.P., Domingues, J.P., Silva, M.R.: First exposure to Arduino through peer-coaching: Impact on students' attitudes towards programming. Comput. Hum. Behav. **76**, 51–58 (2017)
16. Kelly, N., Gero, J.S.: Design thinking and computational thinking: a dual process model for addressing design problems. Des. Sci. 7 (2021)
17. Juškevičienė, A., Dagienė, V., Dolgopolovas, V.: Integrated activities in STEM environment: methodology and implementation practice. Comput. Appl. Eng. Educ. **29**(1), 209–228 (2021)
18. Rubio, M.A., Hierro, C.M., Pablo, A.P.D.Y.: Using Arduino to enhance computer programming courses in science and engineering. In: Proceedings of EDULEARN13 Conference, pp. 1–3 (2013)
19. Bocconi, S., Chioccariello, A., Kampylis, P., Dagienė, V., Wastiau, P., Engelhardt, K., Earp, J., Horvath, M.A., Jasutė, E., Malagoli, C., Masiulionytė-Dagienė, V., Stupurienė, G.: In: Inamorato dos Santos, A., Cachia, R., Giannoutsou, N., Punie, Y. eds., Reviewing Computational Thinking in Compulsory Education. Publications Office of the European Union, Luxembourg (2022). ISBN 978–92–76–47208–7, https://doi.org/10.2760/126955, JRC128347
20. Jocius, R., Joshi, D., Dong, Y., Robinson, R., Catetè, V., Barnes, T., Lytle, N.: Code, connect, create: The 3c professional development model to support computational thinking infusion. In: Proceedings of the 51st ACM technical symposium on computer science education, pp. 971–977. (February 2020)

21. Zhang, L.C., Nouri, J., Rolandsson, L.: Progression of computational thinking skills in swedish compulsory schools with block-based programming. In: ACE—Proceedings of the Australasian Computing Education Conference, Held Conjunction Australasian Computer Science Week, pp. 66–75 (2020). Scopus. https://doi.org/10.1145/3373165.3373173.
22. Brennan, K., Resnick, M.: New frameworks for studying and assessing the development of computational thinking. In: Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada Vol. 1, pp. 25 (April 2012)
23. Wang, C., Shen, J., Chao, J.: Integrating computational thinking in STEM education: a literature review. Int. J. Sci. Math. Educ. 1–24 (2021)
24. Li, Q.: Computational thinking and teacher education: an expert interview study. Hum. Behav. Emerging Technol. **1–15**,(2020). https://doi.org/10.1002/hbe2.224
25. Grover, S., Pea, R.: Computational thinking: A competency whose time has come. In: Sentance, S., Barendsen, E., Schulte, C. (eds.) Computer Science Education: Perspectives on Teaching and Learning, pp. 19–38. Bloomsbury Academic (2018)
26. Sun, L., Hu, L., Yang, W., Zhou, D., Wang, X.: STEM learning attitude predicts computational thinking skills among primary school students. J. Comput. Assist. Learn. **37**(2), 346–358. Scopus (2021)
27. Buchanan, R.: Design research and the new learning. Des. Issues **17**(4), 3–23 (2001)
28. Pratomo, L.C., Wardani, D.K.: The effectiveness of design thinking in improving student creativity skills and entrepreneurial alertness. Int. J. Inst. **14**(4) (2021)
29. Henriksen, D.: Creating STEAM with design thinking: beyond STEM and arts integration. STEAM J. **3**(1), 11 (2017)
30. Malele, V., Ramaboka, M.E.: The design thinking approach to students STEAM projects. Procedia CIRP **91**, 230–236 (2020)
31. Carroll, M.: Stretch, dream, and do-a 21st century design thinking & STEM journey. J. Res. STEM Educ. **1**(1), 59–70 (2015)
32. Henriksen D., Mehta R., Mehta S.: Design thinking gives STEAM to teaching: a framework that breaks disciplinary boundaries. In: Khine, M., Areepattamannil S. (eds.), STEAM Education. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-04003-1_4
33. Gleason, B., Jaramillo Cherrez, N.: Design thinking approach to global collaboration and empowered learning: virtual exchange as innovation in a teacher education course. TechTrends (2021). https://doi.org/10.1007/s11528-020-00573-6
34. Ambrose, G., Harris, P.: Design Thinking. AVA Publishing SA (2010)
35. Dekay, M.: Integral Sustainable Design. Ashford Color Press, NY, Earthscan (2011)
36. Graham, M.A.: The disciplinary borderlands of education: art and STEAM education (Los límites disciplinares de la educación: arte y educación STEAM). J. Study Educ. Dev. **44**(4), 769–800 (2021). https://doi.org/10.1080/02103702.2021.1926163
37. Keane, L., Keane, M.: STEAM by Design. Des. Technol. Educ.: Int. J. S.l. **21**(1) (2016). https://ojs.lboro.ac.uk/DATE/article/view/2085
38. Cook, K.L., Bush, S.B.: Design thinking in integrated STEAM learning: surveying the landscape and exploring exemplars in elementary grades. Sch. Sci. Math. **118**, 93–103 (2018). https://doi.org/10.1111/ssm.12268
39. Kijima, R., Yang-Yoshihara, M., Maekawa, M.S.: Using design thinking to cultivate the next generation of female STEAM thinkers. Int. J. STEM Educ. **8**, 14 (2021). https://doi.org/10.1186/s40594-021-00271-6
40. Dotson, M.E., Alvarez, V., Tackett, M., Asturias, G., Leon, I., Ramanujam, N.: Design thinking-based STEM learning: preliminary results on achieving scale and sustainability through the IGNITE model. Front. Educ. **5**,(2020). https://doi.org/10.3389/feduc.2020.00014
41. Ladachart, L., Cholsin, J., Kwanpet, S., et al.: Ninth-grade students' perceptions on the design-thinking mindset in the context of reverse engineering. Int. J. Technol. Des. Educ. (2021). https://doi.org/10.1007/s10798-021-09701-6
42. Arevalo, I.J.M., Caliste, R.A.F., Prudente, M.S.: Socioemotional skill domains in robotics performance tasks using design thinking process. In: Proceedings of the 2020 11th International

Conference on E-education, E-business, E-management, and E-learning (IC4E 2020), pp. 81–86. Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3377571.3377586
43. Sobieraj, S., Krämer, N.C.: Similarities and differences between genders in the usage of computer with different levels of technological complexity. Comput. Hum. Behav. **104** (2020)
44. Wing, J.M.: Computational thinking. Commun. ACM **49**(3), 33–35 (2006)

Education Data for Science: Case of Lithuania

Audronė Jakaitienė, Rimantas Želvys, and Rita Dukynaitė

Abstract The article reviews various sources of educational data (e.g., international large-scale studies, data registers) and their use for policy decisions and research in Lithuania. It has been shown that a lot of data has already been collected and that more is being accumulated. Up to 20 percent of the information gathered is used for policy purposes and even less in research. It is noted that most of the data collected is useful for the economic paradigm. The chapter presents case study showing that national population-based studies and international achievement studies may send different messages and cannot be considered in isolation.

Keywords International large-scale studies · Data registers · Education

1 Introduction

The term "data" relates to numerical facts collected for reference or information. However, by using the term we assume that a certain body of knowledge is selected from the overall amount of potentially available sources. In this sense data is a social product: social institutions make decisions about what kind of data is needed, what methods of data collection should be applied and what purposes the obtained data are used for. Finally, decisions are made of different ways of presenting data to selected audiences. Education is one of the social institutions where data are used for a variety of purposes. During the last few decades one can observe a massive increase of data collection and practical application in different domains of education. Williamson

A. Jakaitienė (✉)
Institute of Data Science and Digital Technologies, Faculty of Mathematics and Informatics, Vilnius University, Vilnius, Lithuania
e-mail: audrone.jakaitiene@mf.vu.lt

R. Želvys
Institute of Educational Sciences, Faculty of Philosophy, Vilnius University, Vilnius, Lithuania

R. Dukynaitė
Ministry of Education, Science and Sport, Vilnius, Lithuania

G. Dzemyda et al. (eds.), *Data Science in Applications*,
Studies in Computational Intelligence 1084,
https://doi.org/10.1007/978-3-031-24453-7_2

[29] points out two main trends in contemporary education—“datafication” and “digitization”:

> “Datafication” refers to the transformation of different aspects of education (tests scores, school inspection reports, etc.,) into digital data. Making information about education into digital data allows it to be inserted into databases, where it can be measured, calculations can be performed on it, and through which it can be turned into charts, tables and other forms of graphical presentation. “Digitization” refers to the translation of diverse educational practices into software code: aspects of teaching and learning are digitized as e-learning software products ([29], 5 p.).

Datafication and digitization of education support and complement one another. Results of national examinations and testing, international large-scale student assessment studies (ILSAs), various kinds of educational indicators etc., constitute massive databases. Making sense in these databases can only be accomplished by using software that has been coded to enable particular kinds of analyses and interpretations. Williamson [29] concludes that we are currently witnessing signs of a new way of thinking about education as a datafied and digitized social institution [29].

For practical application of data in education the datafied and digitized information requires analysis and interpretation. In [20] Selwyn argues that education cannot be understood fully without paying proper attention to the accumulation and flow of data. Contrary to the popular understandings of data to be broadly neutral, objective and therefore non-problematic, data are political in nature and loaded with values, interests and assumptions that shape and limit what is done with it and by whom. Selwyn [20] notes that generation, accumulation, processing and analysis of digital data is now being touted as a potential panacea for many current educational challenges and problems [20].

Major international organizations collect and present data which reflect their ideological principles and support their understanding of improving education systems throughout the world. For example, the World Bank has developed many resources for providing information which countries can use in order to assess their educational achievement [4]. The World Bank holds around 2 500 internationally comparable education indicators for access, progression, completion, funding, etc. The data bank covers all cycles from pre-primary to tertiary education. Since 1992 the Organization for Economic Cooperation and Development (OECD) issues annual reviews—“Education at a Glance”—which report the achievement of the established indicators [16]. The reviews present data received from different available sources; however, the OECD‘s own sources constitute the core. OECD experts also publish country and thematic reviews on education which provide numerous statistical data. Country reviews are not limited only on the OECD member states and cover a broader array of countries and education systems. E.g., Lithuania became a full member of the organization in 2018. However, the first OECD review on national policies for education in Lithuania was published in 2002 [13], and the recent one—in 2017 [15], just before the accession. The European Union also follows the developments of its member states in the field of education. As means of monitoring progress and contributing to evidence-informed policy-making through systematic data collection and analysis, the Member States of the European Union agreed to follow the reference levels of European average performance, or EU-level targets [26]. The yearly evaluation of education and training systems across Europe and provision of the latest data is presented in an annual report “Education and Training Monitor” [8]. United Nations Educational, Scientific and Cultural Organization (UNESCO) also helds a database of resources in education. The UNESCO Institute for Statistics, established in 1999, is the depository of cross-nationally comparable statistics on education, science and technology, culture and communication. The organization periodically publishes global education monitoring reports [28]. These are just few examples of a vast amount of available data on education which can be used on a global, regional or national level.

The article reviews various sources of educational data (e.g., international large-scale studies, national data registers) and their use for policy decisions and research in Lithuania. Our goal is to quantify the share of the data used in continuous education monitoring in Lithuania. Furthermore, we argue that education policy decisions cannot be based solely on international surveys regardless of the national information available on the population of country-specific students.

2 International Surveys Versus National Testing?

Data and the ways of its presentation do not necessarily provide an unbiased view of education; it can also be misleading. In [24] Takayama and Lingard note that the influence of datafication on the process of schooling has raised serious concerns among many education policy researchers. They argue that datafication has inserted a new logic in the governance of education: data and its rapid flows act as a powerful tool of social regulation. Sjøberg [21] observes that statistics and indicators do not just describe reality, they construct and shape reality [22]. Therefore, the excessive usage of data raises the need for detailed inquiry and critique. Perhaps the most explicit examples of data application which require a critical approach are the ILSAs, which have become an indirect, but nonetheless influential tool of the new political technology of governing the European educational space by numbers [7]. “Governing by numbers” appear to be one of the key policy instruments of the Global Educational Reform Movement, which has become the leading trend of educational change during the last several decades [19]. There are numerous research publications which analyse cases of governing education through data. Researchers often focus on the pros and cons of the usage of data acquired by initiating large-scale comparative studies. E.g., Addey [1] describes the way the OECD uses the results of ILSAs for policy-making purposes [1]. The author notes that by the use of ILSAs the OECD creates a global system that generates, collects, manages, compares and analyses data and in that way the organization exercises of governance of global education. Ozga [17] observes that constant comparison has become a distinctive mode of operation in education [17]. In this sense comparison in itself can also be considered as a tool of governance:

> Comparison is used to provide evidence that legitimises political actions, through such devices as the “international spectacle” of “success” or “failure” and the “politics of mutual accountability” through league tables of performance. The constant collection of data that apparently estimate or reflect “public opinion” produces a need for further data that justify activity ([17], 158 p.).

Scholars also note that statistical data and league tables of student performance can also be used for justification of educational reforms, initiated by the national governments. Even when countries demonstrate similar ILSAs results, they often implement different policies and take different tracks to try to improve their educational performances [3, 5]. Reaction often depends on the current political situation. Opposition often tends to use poor performance in comparative studies to discredit the ruling political parties. New governments may differently emphasize the purpose of participation in ILSAs to create point of distinction from previous governments [2]. In [7] Grek analysed reactions to PISA (Programme for International Student Assessment) results of governments and media in three European countries: Finland, Germany and United Kingdom. In Finland the results were received with neutrality by the media and by surprise by the government. Surprisingly, the Finnish government decided to proceed with the reforms despite the worldwide acclaim of the existing education system. In Germany negative evaluation of PISA results dominated in the media and German government was urged to undertake urgent educational reforms. As a result, the government introduced national testing of learning outcomes in core subjects. The media in United Kingdom was not very critical about the moderate PISA results. In contrast to the other two countries, the government did not undertake any reform efforts and just noted that PISA is a good marketing instrument for education. Grek [7] summarizes that one can observe at least three different reactions to PISA results: PISA-surprise in Finland, PISA-shock in Germany and PISA-promotion in the United Kingdom [7]. Interestingly, though success of Finland in PISA came as a surprise, Finnish educators themselves are not as excited about PISA results as many foreigners would expect. They are afraid that growing preoccupation with student performance in PISA and governmental reforms will eventually lead to narrowing of school curriculum and creating “PISA classrooms” and “PISA schools” [18]. Jakupec and Meier [10] observe that results of the first PISA study in 2000 in Germany as well as in many other Central European countries caused a wide array of feelings: disbelief, horror, agreement, discontent and rejection [10]. German and French media called the PISA results of their countries catastrophic. Jakupec and Meier [10] think that situation has not improved much since then and Central European countries still experience aftershocks [10]. In [12] Lockheed and Wagemaker reflect on different roles assigned to ILSAs. The authors note that at first the results were mainly used as “thermometers” that measured student achievement at national level; recently they are more often used as “whips” used to motivate countries to take policy actions to improve their education systems. Taken alone, these tools do not provide sufficient information to inform policy. Many poorly performing countries are not happy with their place in the league table which the “thermometer” shows and refuse to publish their scores in international reports or just opt out of future assessments. Lockheed and Wagemaker [12] conclude that in order to make ILSAs useful policy tools, the two major missions need to be aligned and have to be given equal weight [12]. Steiner-Khamsi and Waldow [23], discussing the impact of ILSAs on national education policies, refer to the terms “scandalisation” and “projection” [23]. Scandalisation means highlighting the weaknesses of one’s own educational system as a result of comparison. The result of comparison to a large extent depends on the choice of a country as a reference society. Scandalisation can bear a rather subjective nature, e. g., it can even occur when ILSAs results are very good in case there is a perception that they have been achieved at too high a price. Projection means that observers see what they want to see, though what is observed may not actually exist:

> Projections serve to legitimate or de-legitimate educational policies and agendas in the place from where the projection is made. Conceptions of”good” and”bad” education are projected onto countries or regions like a slide or film is projected onto a projection screen. Reference societies will thus usually be depicted in a very selective way, with certain aspects being emphasised out of proportion and complex or contradictory aspects being presented in a simplified way ([23], 2018, 560 p.)

Results of ILSAs mainly serve for the purposes of making comparisons between countries and composing league tables. Scores of national testing and examination in many countries serve other purposes: they are used for monitoring the performance of national systems of education and student enrolment to higher education institutions. School leaving examinations are treated as high stakes examinations by both students and their parents as their results can determine the acquisition of state grants and/or the possibility to join the desired study program. Besides that, examination results can be used for holding regions, teachers and schools accountable. E.g., in [27] Tyumeneva notes that in Russia schools with poor school leaving examination scores may be subject to closer school inspection. Regional ministries and municipalities use the examination data for school ranking purposes, and some regional ministries even establish funding priorities based on the examination results. Examination results are also used to hold teachers accountable and to distribute bonuses: a national wage system provides teachers with bonuses based on the performance of their students [27]. Tampayeva [25] also admits that in post-socialist countries, in particular, Kazakhstan, national testing has a broader mission than just the assessment of student achievement and enrolment to higher education institutions [25]. Besides educational, it also serves a moral purpose: avoiding cheating among students and preventing corruption among educators.

While results of national testing or school leaving examinations may bear significant consequences for schools, teachers and students, results of ILSAs seem to be more important for politicians and educational decision-makers. However, in countries and territories where strong motivation to demonstrate good performance in international comparative studies prevails (Honkong, Taiwan, Singapore, etc.), parents, students and society at large also tend to take the completion of the tasks seriously [21]. No wonder that these countries and territories are leading in many international student surveys. On the other hand, in a number of Western countries schools, students and parents, unlike educational policy-makers, are less sensitive to the results of ILSAs. In these countries students do not see much reason in making their best and thus leave many test items uncompleted. E.g., in completing PISA 2009 test items, only 91% of Australian students reached the end of the test. In contrast, 98% of Shanghai students reached the end of the test, which means that Australia's average score was negatively affected by the 9% of students who did not complete the test [6]. One can assume that the data obtained reflects not only the factual student achievement, but also the level of their motivation.

Differences of approach towards ILSAs and national testing lead to a more general issue of educational goals. In [10] Jakupec and Meier claim that the OECD, following the Anglo-Saxon tradition, is firmly rooted in promoting *homo economicus*. In contrast there is a Central European dominant cultural content leading to a humanistic development of individuals with a focus on *homo academicus*. In this sense education is understood as a highly individualised act, based on an educational-philosophical ideal and following a certain set of values. Jakupec and Meier [10] wonder whether this humanistic approach is compatible with economization and utilization of education, promoted by the OECD under the banner of economic competitiveness [10]. The increasing importance of quantitative data also implies the shift from *homo academicus* to *homo economicus*. In [22] Sjøberg notes that by using the set of established indicators the OECD seeks to standardize and universalize education systems and tends to ignore the local context and national curricula. In the process of transformation of educational goals into performance indicators the moral and humanistic aspects of education are "lost in translation" and are reduced to several measurable targets [11]. International measurable and comparable targets as a rule reflect the economic dimension of education (key competencies, meeting the needs of the global labour market, etc.). From the point of view of politicians and society at large, performance indicators become more important than educational goals, achievement of which they were supposed to reflect. Eventually policy makers consider international performance indicators as points of reference in planning and implementing educational reforms.

3 Sources of Education Data in Lithuania

In this section, we will review the data sources Lithuania has and the amounts of data stored there. In total there are 20 information systems (IS)[1] and 9 registers[2] related to education in Lithuania. Of the 20 information systems, nine are national-level. Six IS have information related to libraries, scholarships or other institutions, and we will not consider them in this analysis (see Table 1 for the full title and website link).

KRISIN is IS to start from as it accumulates all legal information about all IS, registries, and classifiers applicable in Lithuania as well as recent changes of them. There one might find links to all other IS and registries. However, all information,

[1] https://www.krisin.smm.lt/aikos2-krisin/public/sistemuSarasas.xhtml.

[2] https://www.krisin.smm.lt/aikos2-krisin/public/registruSarasas.xhtml.

Table 1 List of information systems and registries for education in Lithuania

INFORMATION SYSTEMS
KRISIN. Information System for Accounting of Education and Science Information Systems, Registers and Classifiers. http://www.krisin.smm.lt EMIS. Education Management Information System. http://www.svis.smm.lt NEMIS. Information System for Out-of-School and Non-participating in Education Pupils https://nemis.emokykla.lt/
REGISTRIES
Individual level database Student registry (MR). https://mokiniai.emokykla.lt Teacher registry (PR). https://pedagogai.emokykla.lt/ Student (higher education) registry (SR). https://studentai.emokykla.lt/studreg/ **Institutional database** Registry of educational and scientific institutions (ŠMIR). https://www.smir.smm.lt/ **Programs and Certificates database** Registry of Studies, Training Programs and Qualifications (SMPKR). https://www.smpkr.smm.lt/ Registry of Non-formal Educational Programs (NŠPR). http://www.ktprr.smm.lt/ Registry of Diplomas, Certificates and Qualifications (DAKPR). https://www.dakpr.smm.lt/ Registry of Education Certificates and Forms (IPBR). https://www.ipbr.smm.lt/ Registry of Licenses (LICR). https://www.licr.smm.lt/

except some graphical information from EMIS, is available in Lithuanian. In addition, all IS and registries lacks basic descriptive information, for example, how many variables are stored in IS, how number of records stored is changing over time. Information about IS or registry size can be found in legal documents. From the latter we know that student registry (MR) has around 80 variables about each pupil. The teacher registry collects 40 variables, and the scholar registry (SR) has 95 variables about each student. The largest data base is EMIS which integrates around 300 variables from 26 registries or IS (e.g., Lithuanian Statistics, SR, PR, SR, NEMIS, Centre of Registers, State Tax Inspectorate and other). EMIS is main IS which might be used and analysed for policy making matters as well as research. EMIS collects data on pre-school, primary, basic, secondary education, vocational training, and studies. The data are used to calculate indicators, the monitoring of which allows to assess the state of education in Lithuania. Using the system users might analyse the collected data in various cross-sections. NEMIS accumulates information about out-of-school children and children who do not attend school. All registries and IS mentioned collect data about the entire Lithuanian student population. 54 educational variables from the described sources are available in the Lithuanian Open Data Portal (https://data.gov.lt).

In addition to nationally collected data, Lithuania actively participates in international surveys organized by the International Association for the Evaluation of Educational Achievement (IEA), the Organization for Economic Cooperation and Development (OECD), the Swedish Council for Information on Alcohol and Other Drugs (CAN) and the World Health Organization (WHO). Surveys participated and future survey Lithuania is about to participate are presented in Table 2. We observe that Lithuania actively participates in many international studies starting year 1995. With each survey, we have an additional substantial amount of data. For example, the main PISA 2018 data files will include: the student-questionnaire data file (which also includes estimates of student performance and parent-questionnaire data), the school-questionnaire data file, the teacher-questionnaire data file, and the cognitive item data file. Only for the PISA survey, we count variables in thousands rather than in hundreds. Therefore, we have a large amount of data to monitor and investigate education. However, one should not forget that in all ILSA data are collected for stratified random sample and information is summarized for a population of country. In Table 3, we provide sample size of students that participated in most recent PISA, TIMSS, PIRLS, ICCS studies. For example, population size for 4th and 8th grade is 21–24 thousand students (depending on a year analysed), therefore in TIMSS participate roughly 20–25% of total students.

4 How Much Data Do We Explore?

In the previous section, we summarise all sources of data for education. As presented, there is a substantial amount of data available. Having this data in mind, we should realize that researchers who would like to conduct empirical analysis on national or international data analysis should have applicable statistical and informatics skills to be able to analyse educational data. This leads that sociology studies should reflect this need in school, vocational or university programs. Thus, we have sufficient amount of information to analyse, but how much we use for policy decisions as well as research?

The monitoring of education and science in Lithuania is organized on the basis of the Law on Education of the Republic of Lithuania (Article 53) and the Law on Science and Studies of the Republic of Lithuania. The description of the procedure for monitoring education and science is approved by the order of the Minister of Education, Science, and Sport of the Republic of Lithuania. This procedure provides for the country's strategic, operational, tactical and forward-looking indicators. The list of indicators is reviewed once a year. EMIS regularly collects, calculates, and disseminates strategic and tactical indicators. Strategic indicators are in line with the main educational CIPO model (see Fig. 1): context (C), input (I), process (P), and output/outcome (O).

For monitoring Lithuanian education system, there are in total 42 strategic indicators selected: 6 context, 12 input, 11 process, and 13 output/outcome. In addition, 16 tactical indicators are provided. From Table 4, we read that output/outcome of education is measured using ILSAs results in Lithuania and national examination information is treated as tactical indicators. Strategic indicators used for continuous monitoring of the state of education in Lithuania make up around 14% (42 indicators out of 300) of the information collected by EMIS. Tactical indicators add another 6 percentage points.

Table 2 List of international surveys for Lithuania

Organization	Survey	Past surveys					Future surveys	
		1995–2000	2001–2005	2006–2010	2011–2015	2016–2020	2021–2025	2026–2030
IEA	TIMSS	1995, 1999	2003	2007	2011, 2015	2019	2023	2027
	SITES, ICILS	1996, 1998, 1999, 2000	2001, 2002	2006	2013	–	2023	2028
	PIRLS	–	2001	2006	2011	2016	2021	2026
	ICCS	1999	–	2009	–	2016	2022	2028
OECD	PISA	–	2005	2006, 2009	2012, 2015	2018	2022[1], 2025[2]	2028, 2031[3]
	TALIS	–	–	2008	–	2018	2024	–
	PIAAC	–	–	–	2015[4]	–	2022–2023[5]	–
CAN	ESPAD	1995, 1999	2003	2007	2011, 2015	2019	2023	2027
WHO	HBSC	–	2001, 2005	2009	2013	2017	2021	2025

[1] Planned for 2021. Due to COVID-19 delayed to 2022
[2] Planned for 2024. Due to COVID-19 delayed to 2025
[3] Planned for 2027 and 2030. Due to COVID-19 delayed to some later period
[4] OECD PIAAC survey conducted in 2012, but Lithuania collected data in 2015
[5] Due to COVID-19, PIAAC delayed to 2022–2023
TIMSS—Trends in International Mathematics and Science Study
SITES—Second Information Technology in Education Study
ICILS—International Computer and Information Literacy Study
PIRLS—Progress in International Reading Literacy Study
ICCS—International Civic and Citizenship Education Study
PISA—Programme for International Student Assessment
TALIS—Teaching and Learning International Survey
PIAAC—The Programme for the International Assessment of Adult Competencies
ESPAD—European School Survey Project on Alcohol and Other Drugs
HBSC—Health behaviour in school-aged children

Table 3 Sample size for the latest PISA, TIMSS, PIRLS and ICCS studies for Lithuania

Survey	Sample size in strata			Total sample size
	Lithuanian	Polish	Russian	
PISA 2015	5153	624	748	6525
PISA 2018	5868	475	542	6885
eTIMSS 2019 8th grade	3031	418	377	3826
TIMSS 2019 8th grade	1482	89	116	1687
eTIMSS 2019 4th grade	2877	455	409	3741
TIMSS 2019 4th grade	1367	101	119	1587
PIRLS 2016	2947	564	806	4317
ICCS 2016	2767	356	508	3631

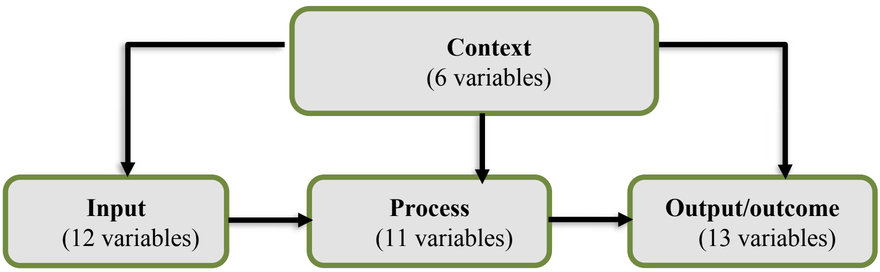


Fig. 1 CIPO model and number of selected variables for monitoring of Lithuanian education system

To find out how many variables were used in research, we explored google scholar with the aim of finding out how often information from the EMIS and NEMIS databases was used in publications. We review articles from 2017. We used following search keywords in English as well as Lithuanian: "Education Management Information and Lithuania", "Education Management Information or Lithuania".

EMIS database was used in 23 research papers of which 5 papers discuss medical issues as the rest is related to education. 1 paper used student registry to analyse links between asthma and pollution in Vilnius (population study). We could not find any paper using NEMIS database. We also found no articles in which the whole set of strategic and tactic indicators would have been analysed for the monitoring of the Lithuanian education system.

Table 4 Total list of variables selected for monitoring of Lithuanian education system in EMIS according CIPO model

Type	Indicators
Context	1. Population change 2. Proportion of people of working age who do not work or study anywhere 3. Proportion of the population living below the poverty line 4. Number of persons suspected of (charged with) criminal offences per 100,000 population 5. Number of suicides per 100,000 population 6. GDP per capita
Input	1. Distribution of students by gender 2. Average age of teaching staff 3. Distribution of teaching staff by gender 4. Share of highly qualified teaching staff 5. Average age of heads of educational institutions 6. Distribution of heads of educational institutions by gender 7. State and municipal budget expenditures on education as % of GDP 8. Average funds per student 9. Share of funds allocated to education by individuals and legal entities 10. Share of educational institutions that do not require major repairs to any part of the building 11. Share of educational institutions adapted for people with disabilities 12. Share of educational institutions with laboratories and / or technical classes
Process	1. Proportion of students who choose to study science or technology 2. Number of citizens of other countries studying in Lithuania 3. Share of students in the age group 4. Share of students with disabilities 5. Share of teachers' contact hours compared to teachers' full working time 6. Proportion of full-time teachers 7. Ratio of teachers and other staff 8. Average number of students in a class set / group 9. Proportion of positively educational institutions implementing pre-school, pre-primary, general education and vocational education programs in which an external evaluation of the quality of activities has been carried out during the last four years 10. Ratio of pupils (students) to teachers (academic staff) 11. Share of teaching staff who have participated in international exchange programs in the last 5 years

(continued)

5 Does National Data is in Line with ILSAs?

As already mentioned, policy makers consider international survey variables as points of reference in monitoring education and Lithuania is not an exception. In a case study we will compare population mathematics results from the 10th grade of the academic year 2014–2015 with mathematics results coming from PISA 2015. Both data sets roughly measure the same cohort, although examinations are different from their nature. The 10th grade examination seeks to measure whether students successfully

Table 4 (continued)

Type	Indicators
Output	1. Share of students who have achieved a reading achievement level of at least 3 in the OECD PISA survey 2. Share of students who have achieved a mathematics achievement level of at least 3 in the OECD PISA survey 3. Share of students who have achieved a science achievement level of at least 3 in the OECD PISA survey 4. Percentage distribution of PIRLS 4th grade results by international levels of reading achievement 5. Percentage distribution of TIMSS 4th grade results by international levels of mathematics and science achievements 6. Percentage distribution of TIMSS 8th grade results by international levels of mathematics and science achievements 7. Percentage distribution of ICCS 8th grade results by level of international civic education and citizenship achievement 8. Individuals who have basic or above basic overall digital skills 9. Share of dropouts 10. Proportion of foreign citizens who have graduated from Lithuanian higher education institutions
Outcome	1. Proportion of educational attainment of the population by age groups 2. Share of graduates registered in the Employment Service one year after graduation by level of education 3. Share of students who graduated and continued their education at next level of education or were employed in the same year
Tactical	1. Share of students with special educational needs 2. Share of students receiving financial and other support 3. Share of integrated students with special educational needs 4. Share of students who went to study under international exchange programs 5. Share of those who come to study under international exchange programs 6. Distribution of grade 2 students according to the results of the National Student Achievement Tests (NAPPs) (achievement groups) 7. Distribution of grade 4 students according to the results of the National Student Achievement Tests (NAPPs) (achievement levels) 8. Distribution of grade 6 students according to the results of the National Student Achievement Tests (NAPPs) (achievement levels) 9. Distribution of grade 8 students according to the results of the National Student Achievement Tests (NAPPs) (achievement levels) 10. Distribution of grade 10 students according to the results of the Basic Education Learning Achievement Tests (achievement levels) 11. Distribution of students according State Matura examinations (achievement levels) 12. Distribution of students according school Matura examinations (achievement levels) 13. Distribution of students by annual assessments (achievement levels) 14. Distribution of students by annual average assessments (achievement levels) 14. Distribution of III gymnasium class students according to the level of study of curriculum subjects 15. Distribution of students by zones of physical capacity

mastered the math learning program. All students solve a single centrally prepared test which is marked by local teachers in 10-point scale (for detailed analysis, see [9]). PISA tests whether 15-year students have enough knowledge to solve real-world problems.

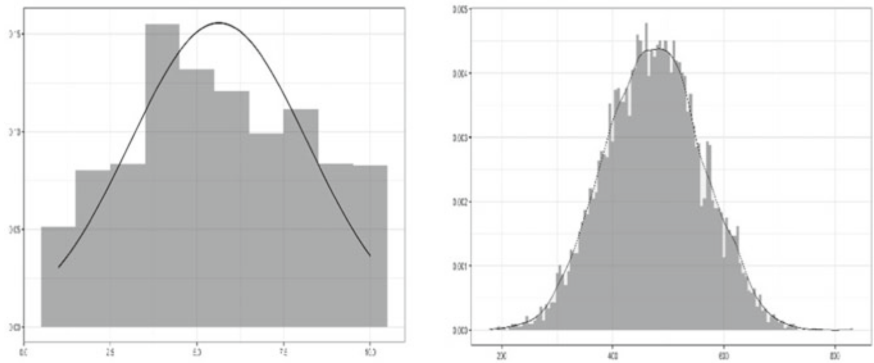
We recall that all ILSA studies are conducted using a common framework for all countries. Each student receives a set of items for the assessment of performance. A generalized partial credit item response theory (IRT) model is used to create achievement scales that are standardized with a mean score of 500 and a standard deviation of 100 among the OECD countries.[3] PISA uses the imputation methodology usually referred to as plausible values [14]. The idea behind plausible values is that the student does not solve all the items of a survey rather a specific set and version of items. Based on the completion of the items and the available contextual information, the plausible values for each set of items (even if not solved by the student) are estimated using IRT theory. The objective of ILSAs is to provide an unbiased assessment of the achievement of a targeted student population rather than an individual student. This means that we do not suppose to monitor the student's performance using ILSA information.

As for now, we have explained some methodological differences between two tests. Let us look at how the resulting distribution is similar or different and whether one can draw common conclusions from both sets. In ILSAs (in line with IRT theory), estimated plausible values are fitted to the normal distribution (see Fig. 2 panel B). Distribution of 10th grade results do not follow normal and is more similar to uniform distribution (see Fig. 2 panel A). Mode of 10th grade achievements is equal to 4, which is lowest positive grade. The average achievement score from PISA is close to 500, meaning that Lithuanian students handle the PISA test close to the OECD average. Consequently, Lithuania's achievements are on average good according to the PISA, and they are on average poor according to national examination. This case study raises many questions about the quality of the tests, the principles of their organization, which we will leave to answer in other studies. However, the answer to the question of whether we can shape policy by analysing only ILSA data is rather no than yes.

6 Conclusions

The article reviews various sources of educational data (e.g., international large-scale student achievement studies, data registers) and their use for policy decisions and research in Lithuania. It has been shown that a lot of data has already been collected and that more is being accumulated. The last section presents case studies showing that national population-based studies and international achievement studies may send different messages and cannot be considered in isolation. The centralised national examination of student achievement has been conducted for two decades in

[3] ILSA-Gateway: PISA 2018 Results | ILSA-Gateway.



A. B.

Fig. 2 Distribution of results from 10th grade academic year 2014–2015 (A.) and PISA 2015 (B.)

Lithuania, which is an invaluable asset of the country that allows the longitudinal education system, schools and individual progress analysis. In our opinion, the use of international surveys can only be indirect or as an expert judgment about the state of education in Lithuania, as information from national examinations is available to the whole population and a representative sample is used in international surveys.

For educational monitoring purposes, we follow up to 20% of the information collected in EMIS. The rest of information is used on some occasions for analysing specific situations. Analysing scientific articles published after 2017, we found that EMIS information is used only in 23 publications. The NEMIS information is not used in scientific articles. Thus, collected information is used to a limited extent in scientific educational research, and it would be interesting to investigate why researchers are not inclined to use the accumulated IS information in their scientific research. We also were not able to find any research paper that would analyse the whole set of strategic and tactic indicators for the monitoring of the Lithuanian education system.

Our research reveals that educational development in Lithuania follows the global tendency of change from socio-cultural to economic paradigm. From the very beginning education in Lithuania was mainly focused on achieving the socio-cultural goals of fostering one's individual skills and talents and developing a culturally and morally advanced society. However, during the last three decades the rise of a new public management and Global Education Reform Movement led to a major change of educational rhetoric. Apparently, the main goal of contemporary education becomes the development of key competences required by the international labour market. The usage of international standardized and unified data contributes to the further shift towards *homo economicus* in education.

References

1. Addey, C.: Golden relics & historical standards: how the OECD is expanding global education governance through PISA for development. Crit. Stud. Educ. (2017). https://doi.org/10.1080/17508487.2017.1352006
2. Addey, C., Sellar, S., Steiner-Khamsi, G., Lingard, B., Verger, A.: The rise of international large-scale assessments and rationales for participation. Comp.: J. Comp. Int. Educ. **47**(3), 434–452 (2017)
3. Baird, J.-A., Johnson, S., Hopfenbeck, T.N., Isaacs, T., Sprague, T., Stobart, G., Yu, G.: On the supranational spell of PISA in policy. Educ. Res. **58**(2), 121–138 (2016)
4. Clarke, M., Luna-Bazaldua, D.: Primer on Large-Scale Assessments of Educational Achievement. International Bank for Reconstruction and Development/The World Bank, Washington, DC (2021)
5. Fischman, G.E., Topper, A.M., Silova, I., Goebel, J., Holloway, J.L.: Examining the influence of international large-scale assessments on national education policies. J. Educ. Policy (2018). https://doi.org/10.1080/02680939.2018.1460493
6. Gorur, R., Wu, M.: Leaning too far? PISA, policy and Australia's 'top five' ambitions. Discourse: Stud. Cult. Polit. Educ. **36**(5), 647–664 (2015)
7. Grek, S.: Governing by numbers: the PISA 'Effect' in Europe. J. Educ. Policy **24**(1), 23–37 (2009)
8. European Commission.: Education and training monitor (2021). Internet access: https://op.europa.eu/webpub/eac/education-and-training-monitor-2021/en/
9. Jakaitienė, A., Želvys, R., Vaitekaitis, J., Raižienė, S., Dukynaitė, R.: Centralised mathematics assessments of Lithuani-an secondary school students: population analysis. Inform. Educ. **20**(3), 439–462 (2021)
10. Jakupec, V., Meier, B.: PISA—Schocks, after shocks and misconceptions. Leibniz Online (17), 1–11 (2015). Internet access: http://www.leibnizsozietaet.de/wp-content/uploads/2015/02/JakupecMeier.pdf
11. King, K.: Lost in translation? The challenge of translating the global education goal and targets into global indicators. Comp.: J. Comp. Int. Educ. **47**(6), 801–817 (2017)
12. Lockheed, M.L., Wagemaker, H.: International large-scale assessments: thermometers, whips or useful polcy tools? Res. Comp. Int. Educ. **8**(3), 296–306 (2013)
13. OECD.: Reviews of National Policies for Education. Lithuania. OECD Publications, Paris (2002)
14. OECD.: Analyses with Plausible Values. In: PISA Data Analysis Manual: SPSS, 2nd edn. OECD Publishing, Paris (2009).https://doi.org/10.1787/9789264056275-9-en
15. OECD.: Reviews of national policies for education. Educ. Lith. (2017). Internet access: https://read.oecd-ilibrary.org/education/education-in-lithuania_9789264281486-en#page1
16. OECD.: Education at a Glance 2021. OECD Indicators (2021). Internet access: https://www.oecd.org/education/education-at-a-glance/
17. Ozga, J.: Governing education through data in England: from regulation to self-evaluation. J. Educ. Policy **24**(2), 149–162 (2009)
18. Sahlber, P.: PISA in Finland: an education miracle or an obstacle to change? CEPS J. **1**(3), 119–140 (2011)
19. Sahlberg, P.: THe global educational reform movement and its impact on schooling. In: Mundy, K., Green, A., Lingard, B., Venger, A. (eds.) The Handbook of Global Education Policy, pp. 128–144. Wiley (2016)
20. Selwyn, N.: Data entry: towards the critical study of digital data and education. Learn. Media Technol. **40**(1), 64–82 (2015)
21. Sjøberg, S.: PISA and "real life challenges": Mission impossible? In: Hopman, S.T., Brinek, G., Retzl, M. (eds.) PISA According to PISA—Does PISA Keep What it Promises?, pp. 203–225. Lit Verlag, Berlin (2007)
22. Sjøberg, S.: The PISA-syndrome—how the OECD has hijacked the way we perceive pupils, schools and education. Confero **7**(1), 12–65 (2019)

23. Steiner-Khamsi, G., Waldow, F.: PISA for scandalisation, PISA for projection: the use of international large-scale assessments in education policy making—an introduction. Glob. Soc. Educ. **16**(5), 557–565 (2018)
24. Takayama, K., Lingard, B.: Datafication of schooling in Japan: an epistemic critique through the 'problem of Japanese education.' J. Educ. Policy **34**(4), 449–469 (2019)
25. Tampayeva, G.Y.: Importing education: Europeanisation and the bologna process in Europe's backyard—the case of Kazakhstan. Eur. Educ. Res. J. **14**(1), 74–85 (2015)
26. The Council of the European Union.: Council resolution on a strategic framework for European cooperation in education and training towards the European Education Area and beyond (2021–2030) (2021). Internet access: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32021G0226(01)&from=EN
27. Tyumeneva, Y.: Disseminating and Using Student Assessment Information in Russia. The International Bank for Reconstruction and Development/The World Bank, Washington (2013)
28. UNESCO: Global education monitoring report, 2021/2: non-state actors in education: who chooses? who loses? UNESCO, Paris (2021)
29. Williamson, B.: Big Data in Education: The Digital Future of Learning, Policy and Practice. Sage, London (2017)

Imbalanced Data Classification Approach Based on Clustered Training Set

Dalia Breskuvienė and Gintautas Dzemyda

Abstract Fraud detection is a system that prevents criminals from obtaining financial assets. The research aims to increase machine learning prediction quality on fraudulent cases as well as decrease false positive and false negative cases in prediction. Fraudulent data like credit card transactions are usually imbalanced data, and standard machine learning techniques cannot achieve the desired quality levels in this scenario. This paper proposes a clustering-based classification method to improve the *recall*. For the experimental evaluation, we use a credit card transaction database. Firstly we suggest finding the optimal features and number of clusters to create smaller, more homogeneous training sets, which we train on separate machine learning models. The second step is to find relevant percentages to undersample each cluster to compensate for sharply imbalanced data. Our baseline *recall* is 0.845. By applying the proposed method, we improved the *recall* to 0.867. Moreover, classification of fraudulent cases that were labeled as regular decreased from 323 to 278, i.e. by 13.9%. The statistical test has shown that decrease is significant.

Keywords Imbalanced data · k-means · Undersampling · *Recall* · Classification · Fraud detection

1 Introduction

Financial fraud is a growing issue for financial institutions and individuals, especially in times of instability, such as the pandemic lockdowns [1]. Fraud can cause a loss of money or do massive harm to the reputation of institutions. Also, fraudster attacks can

D. Breskuvienė (✉) · G. Dzemyda
Institute of Data Science and Digital Technologies, Vilnius University,
Akademijos str. 4, 08412 Vilnius, Lithuania
e-mail: dalia.breskuviene@gmail.com
URL: http://www.mii.lt

G. Dzemyda
e-mail: Gintautas.Dzemyda@mif.vu.lt

G. Dzemyda et al. (eds.), *Data Science in Applications*,
Studies in Computational Intelligence 1084,
https://doi.org/10.1007/978-3-031-24453-7_3

have a wide range of harm to individuals, from small financial losses to significant financial problems, even leading to death. Lawbreakers are searching for and finding different ways to steal credit card information, trick people into transferring large amounts of money, or other frauds.

Financial institutions like banks or insurance companies apply various methods and approaches, including machine learning, on transactional data to fight fraudsters' attacks. In our case, the fraudulent transactions data set is binary as we can split it into two distinct classes—fraudulent versus regular transactions. Fraudulent cases are called Minority class as there are much fewer instances, while Regular transactions are called Majority class. Binary data set is defined as imbalanced when one of two classes is much more prevalent in the data than the other one. As fraudulent transactions are a rare event that leads to a sharply imbalanced data set. Standard machine learning algorithms treat data sets as roughly balanced, which can cause inaccurate results if used with imbalanced data sets. Additionally, researchers are confronted with data set size, label noise, and data distribution problems when working with such data sets. Some scientists state that it is crucial to understand imbalanced data's intrinsic characteristics and their impact on class disbalance [2]. Finally, fraud investigation faces issues with stability as fraudsters change their ways of stealing information and scamming scenarios. Here, the machine learning algorithm needs to be adapted or retrained frequently. The purpose of our research is to increase machine learning predictions quality on fraudulent cases and significantly reduce false positive and false negative cases in prediction.

The following part of this paper includes a literature review on imbalanced data classification problems, their solutions, and fraud detection challenges. Section 3 describes the theoretical approach of training data preprocessing to improve classifier performance. The experimental results can be found in Sect. 4, with an explanation of the data structure. Eventually, the last part of the paper contains the conclusions and aspirations for future work. Our findings can be applied not only to fraudulent transaction data but also to other research areas. Day-to-day life naturally produces imbalanced data, like the healthcare sector, which is an excellent illustration as it provides many imbalanced data examples such as cancer detection [3], Covid-19 detection [4], and similar.

2 Literature Review

In the community of researchers, interest in imbalanced datasets has been growing in the last ten years. Figure 1 shows the query "Imbalanced data" in Clarivate Analytics of Web of Science Core Collection [5] results.

Traditional machine learning algorithms expect to get balanced data set for training. The imbalance data classification problem can be solved on the data level by balancing the training data set or on the algorithm level by adjusting the machine learning algorithm. One of the algorithm-level solutions is modifying the

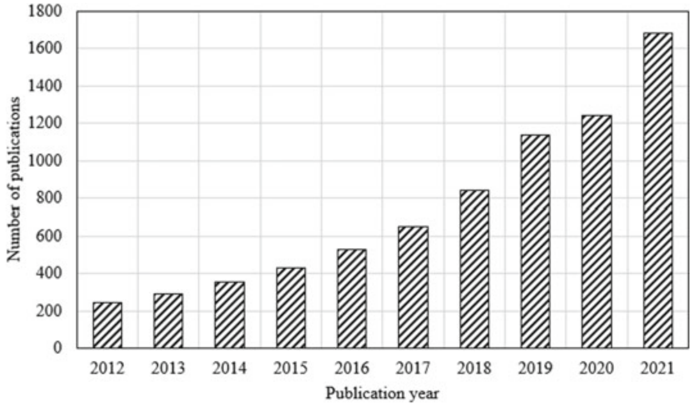


Fig. 1 "Imbalanced data" publications in Clarivate

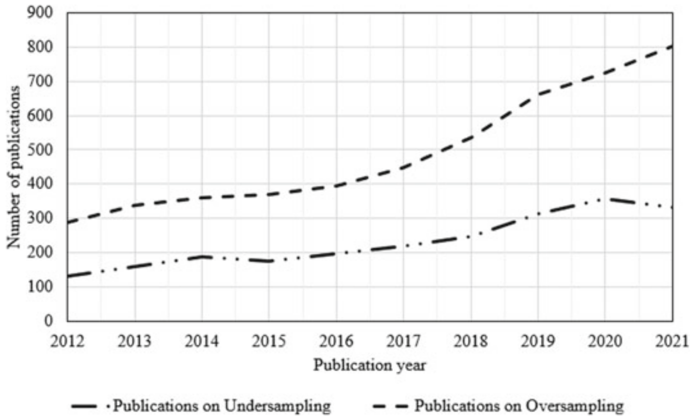


Fig. 2 Number of publications on undersampling and oversampling techniques

classification threshold by the relevant percent to use those algorithms efficiently [6]. Some machine learning algorithms like SVM or XGBoost have parameters to set weights on different classes.

A widespread way to do optimization on the data-level is to resample the training data set. Researchers use various oversampling and/or undersampling techniques for better machine learning performance. Oversampling is currently much more popular among the researcher's community, as shown in Fig. 2. We see these trends when analyzing the Clarivate Analytics of Web of Science Core Collection.

However, both of them carry their advantages and disadvantages. In order to apply resampling methods, the main question that needs to be answered is what share of Minority and Majority classes is the optimal one. The experiment in [7] uses the oversampling method with approximately a 30/70 split on traffic accident data. A popular oversampling technique is SMOTE [8]. It creates synthetic Minority class instances by choosing some of the nearest Minority neighbors, and it generates new samples using the interpolation method between the Minority instances that lie together. Generated data usually do not have accurate probabilistic distribution and are not diverse enough. [9] recommends using "Binary imbalanced data classification based on diversity oversampling using extreme learning machine autoencoder" and "Binary imbalanced data classification based on diversity oversampling by a generative adversarial network." The authors conclude that experimental results show promising performance on imbalanced data classification. However, oversampling techniques require more extensive computational power to generate additional data rows.

When living in the big data world, generating large amounts of additional artificial data does not make sense. In this case, researchers try to find optimal data balance utilizing the undersampling methods. There are many papers published on the topic of undersampling [10–13]. Regardless, a comparison of the undersampling and oversampling techniques showed that the oversampling approach (SMOTE) behaved more robustly than the undersampling (RUS) method in noisy conditions [14]. The experimental results [15] suggest using oversampling rather than undersampling. However, the experimental outcomes were not explicit because undersampling showed better results in several machine learning models in the same experiment. The most significant disadvantage of undersampling is the data loss, which can create a non-representative data set.

Finding a correct metric to measure the model's performance is an additional issue. The classifier outcome can be grouped into four buckets, as shown in Fig. 3.

Traditional classifiers are built to improve accuracy and the percentage of correctly labeled values for the test data, which is unsuitable for an Imbalanced data set. For instance, if the bank has 0.5% of fraudulent transactions, then the model which labels every transaction as non-fraudulent would have an accuracy of 99.5%. One of the measures used in such a case could be the F1 score as suggested in [16]. The F1 score is a harmonic mean of $precision$ and $recall$, and the input of $precision$ and $recall$ have the same preference. $Precision$ is a measure of quality, and $recall$ is a measure of quantity. Higher $precision$ implies that an algorithm produces more relevant outcomes than irrelevant ones. In contrast, high $recall$ indicates that an

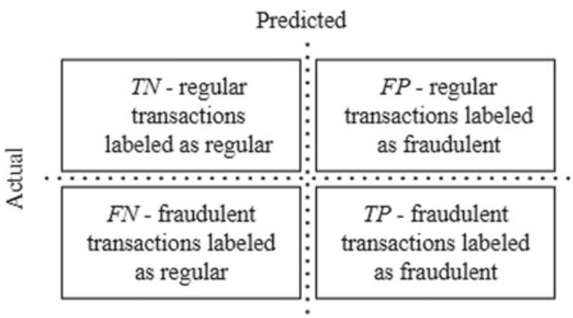


Fig. 3 Confusion matrix

algorithm produces most of the relevant results (nevertheless, irrelevant values are also returned). The ideal value of the F1 score is 1, and the poorest is 0. The formula for the F1 score is:

$$\mathrm{F1} = 2 * \frac{precision * recall}{precision + recall} \tag{1}$$

$$precision = \frac{TP}{TP + FP} \tag{2}$$

$$recall = \frac{TP}{TP + FN}, \tag{3}$$

where TP–a prediction results that correctly indicates the presence of a fraudulent transactions (True Positive). FP–a prediction result which wrongly indicates that a fraudulent transaction is present (False Positive). FN–a prediction result which wrongly indicates that a fraudulent transaction is absent (False Negative)

Additionally, the research's primary goal is to increase the number of correctly labeled fraudulent transactions TP and to reduce the number of fraudulent transactions labeled as regular FN. The secondary goal is to mitigate regular transactions labeled as fraudulent FP. This paper will focus on the primary goal that is aimed by increasing the $recall$.

3 Our Approach

This paper explores ways to improve the model performance for tasks with imbalanced data sets. We apply this approach on transactional data to predict future fraudulent transactions based on past transactions.

Consider the multidimensional data set as an array $X = \{X_i = (x_{i1}, \ldots, x_{in}), i = 1, \ldots, m\}$ of n-dimensional data points $X_i \in \mathbb{R}^n$. In general, data point $X_i = (x_{i1}, \ldots, x_{in})$ is the result of observation of some object or phenomenon dependent on n features $x_1, \ldots, x_n$. In addition, each data point belongs to some class y_i. In our case, features describe particular characteristics of customers financial behaviour, and we have two classes—Regular and Fraudulent transactions, i.e. $y_i \in \{0; 1\}$. The goal is to develop a classification model that assigns the class number to the points with unknown class.

We are creating a classifier that allows us to predict fraudulent transactions. The labeled (regular or fraudulent transactions) data serves for model development. This data consists of training and validation data. Furthermore, the testing data is used to evaluate model performance.

The idea of our approach is to train several classifiers on clustered training data. E.g., k-means may be used for clustering. The optimal number of clusters is chosen. Each machine learning model can be created separately based on the cluster data, i.e., we get some sub-models/sub-classifiers. Which sub-classifier will be activated

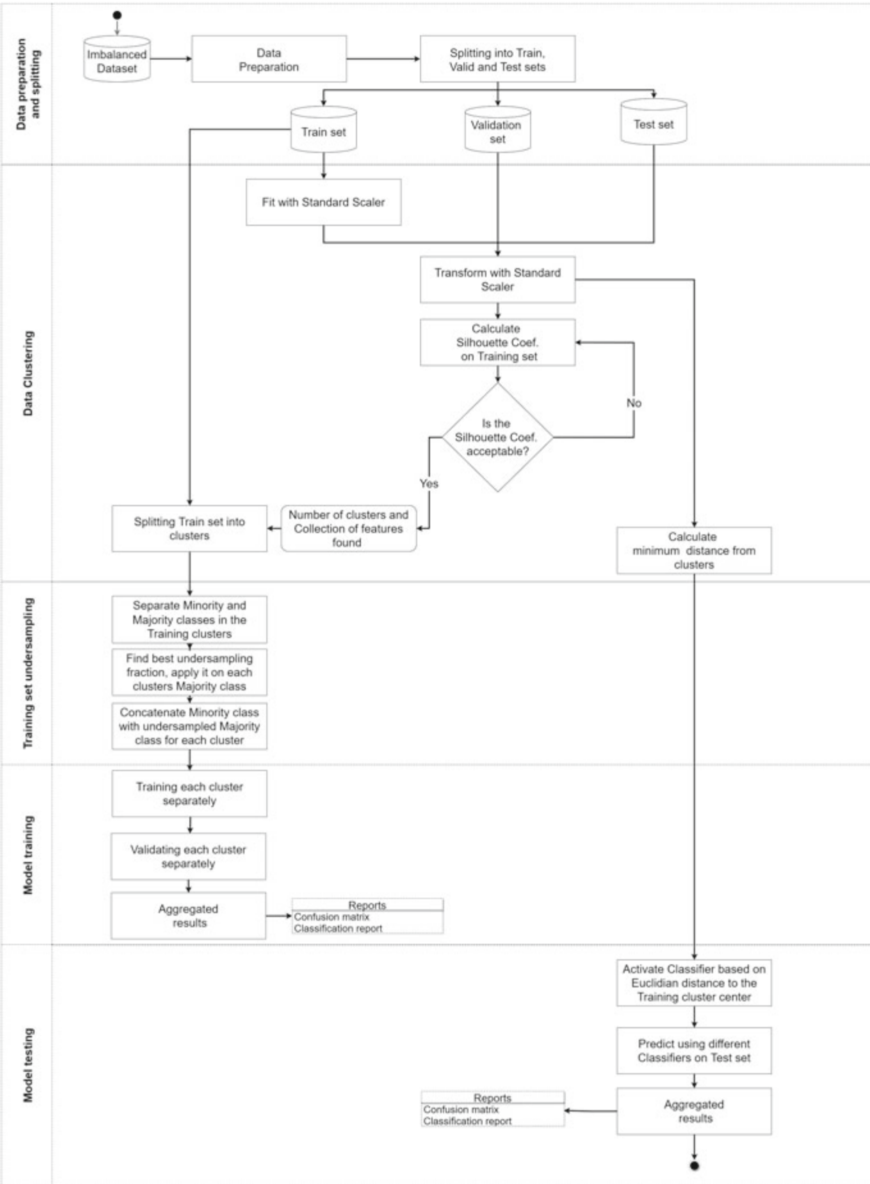


Fig. 4 Algorithm schema

to predict the label depends on the Euclidean distance of the particular test set data point to the training set cluster center. In addition, we use validation data set for optimizing the sub-model performance. The scheme of the decision process can be found in the Fig. 4.

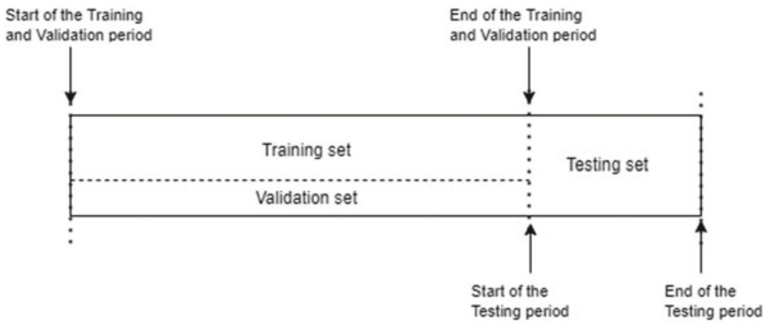


Fig. 5 Data set split

Data set split. We split data into *Train*, *Validation* and *Test* sets, where *Test* set consists of newer data as compared with *Train* and *Validation* sets, so for the training and validation sets, we use the same time period, but for the testing, we have used the period after the training (Fig. 5).

Clustering. Our research focuses on splitting the initial training set into smaller clusters using the k-means clustering algorithm. k-means algorithm clusters data by separating instances into k clusters by reducing within-cluster sum-of-squares:

$$\min \sum_{i=1}^{k} \sum_{X_j \in C_i} \left\| X_j - \mu_i \right\|^2 , \tag{4}$$

where μ_i is the mean of points in C_i.

In order to use the k-means algorithm, it is a must to specify the number of clusters. Even though there is no single way to determine the optimal number of clusters, it can be done visually or using the Silhouette Coefficient.

A well-known way to visually decide on the number of clusters is the *Elbow method*. It helps data scientists to select the optimal number of clusters by drawing the line with the distortion score (sum of square errors) or other relevant scores on the vertical axes and the number of clusters on the horizontal axes. In this case, the "error" is the distance between each data point and the centroid—calculated or actual data point representing the center of the cluster. If the line chart corresponds to an arm, then the "elbow" indicates the point where the model fits the best.

However, sometimes it is hard to identify where an elbow is as a line could be too straight or wavy. In this case, the Silhouette score can be used as a guideline. The highest Silhouette score shows the goodness of the clustering performance. The Silhouette score measures how similar a data point is to its cluster compared to other clusters, and it is calculated for a particular data point X_i as [17]:

$$s(X_i) = \frac{(b(X_i) - a(X_i))}{max\,\{a(X_i), b(X_i)\}} \tag{5}$$

$$a(X_i) = \frac{1}{|C_I| - 1} \sum_{X_j \notin C_I, i \neq j} d(X_i, X_j) \tag{6}$$

$$b(X_i) = \min_{j, i \neq j} \sum_{X_j \in C_J} \frac{d(X_i, X_j)}{|C_J|} \tag{7}$$

where $|C_I|$ is the number of data points belonging to cluster I, $a(X_i)$ is intra-cluster distance, $b(X_i)$ is the Euclidian distance between the data point X_i and the nearest cluster that the point is not a part of. In our case, $d(X_i, X_j)$ is the Euclidean distance between data points X_i and X_j.

The range of the Silhouette Coefficient values is between -1 and 1, where the best score is one and indicates that the instance is far away from the other clusters as compared with its cluster. The negative values imply that samples can be assigned to the wrong clusters [18].

We used Silhouette Coefficient to select relevant features and the number of clusters. Additionally, we evaluated the results by plotting an elbow graph. We suggest empirically checking the feature combinations and the number of clusters until choosing those that satisfy the expectations. The criterion for selecting relevant features and the optimal number of clusters is the highest Silhouette score for various combinations of features.

It is important to mention that the k-means algorithm is sensitive to the amplitude of the feature values, so it is necessary to use the scaling method before the k-means algorithm.

Undersampling. After determining the features used for clustering and the number of clusters k, the initial training set is divided into smaller k training sub-sets, which are used as training sets for individual machine learning models. However, each cluster is still imbalanced. We suggest to utilize the undersampling method to balance data.

We propose to use an individual random undersampling strategy for each training cluster. In our case, the undersampling means leaving all points of the minor class (fraudulent cases) and removing some percentage (call it undersampling percentage) of points from the majority class (regular transactions).

The validation set is utilized to individually determine the best-performing undersampling percentage for each cluster. Let us fix some undersampling percentages for the clusters. When the undersampling of the training set is performed, k sub-classifiers are trained. We go through all validation data set points and apply one of the sub-classifier for decision. The criterion for the selection of a proper classifier is the minimal Euclidean distance between the validation set point and the corresponding cluster center of training data. We check the best performing undersampling percentage for each training cluster by calculating the F1 score on the validation data. While our primary goal is to improve $recall$, we use the F1 score in selecting the best performing resampling percent, because otherwise we could end up having an unacceptable number of regular transactions labeled as fraudulent.

Model performance measure. Model performance is measured by the *recall*. We compare the *recall* calculated for model performance without clustering and undersampling technique on the training set and the *recall* when applying our approach.

4 Experimental Results

4.1 Used Data

Fraudulent transactions are usually sensitive data that is not publically accessible. Moreover, financial institutions' reputational risk and confidence are affected by reporting unusually high numbers of fraud data. Real data of such type are undisclosed by financial institutions. In this case, synthetic data helps to create new algorithms, methods, or strategies for fraud detection. This paper uses Synthetic Credit Card Transaction data created by Erik R. Altman, where patterns and correlations of the actual purchases are recreated. This dataset can be accessed at https://data.world/ealtman/synthetic-credit-card-transactions. It represents the population of the United States with its distribution of age, income, living area, credit score, etc. An article [19] describes deeper insights on the credit card transaction data generation process. The author creates a population of customers who live in different parts of the country with various buying habits depending on their financial situation. In this virtual world, the fraudsters population exists as well. The behavior of the fraudsters is as close to natural as possible. For instance, they buy particular goods on a preferred day and month. They are interested in different deceptions. As shown in Figs. 6 and 7, fraudsters are much more active during the time of 10:00–15:00 and tend to attack older people.

The virtual world has a merchandise population as well. Merchants represent many real-world retailers' behavior, such as McDonald's, WallMart, or luxury goods shops. Retailers' profit is generated depending on their type. So the fraudsters' manners are generated based on the merchant's service.

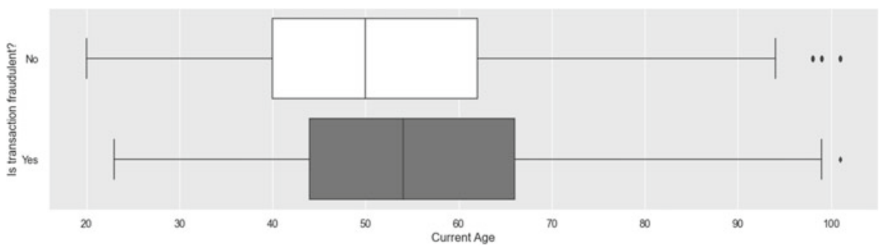


Fig. 6 Fraudsters attacks by age

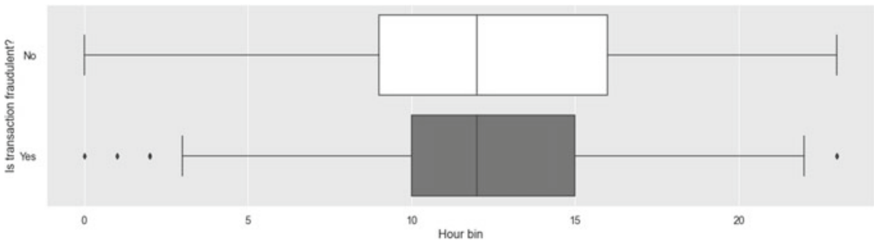


Fig. 7 Fraudsters attacks by hour

4.2 Data Cleaning and Preparation

The published data set is separated into three files. The file with the customer-related information contains 2000 rows, the card-related file contains 6146 rows, and the transaction-related information contains more than 24 million rows. After joining everything into one dataset, it contains more than 24 million rows and 45 features.

Feature encoding and filtering. Columns like Apartment, Merchant State, or Zip were removed as it has many null values that would be complicated to fill in. Furthermore, they do not bring too much value overall as data in other columns gives the equivalent information. Categorical variables with fewer unique values (less than six unique values) were encoded using the OneHotEncodel logic, which creates a binary column for each category. The rest of the categorical features were encoded using a LabelEncoder when random numeric values were assigned to the categorical values. It is essential to state that it is not the best practice to encode categorical values like that because it creates different weights on the feature values.

The initial data set has information from 1991 until 2020, with a growing number of transactions. For the experiment, we took data from 2014. Data from 2014 till 2018 including was used for training and validation, and data starting from 2019 till 2020 was used for testing. The data set for training and validation was split using a 30/70 share. The prepared training set contains 28 features (list of the features can be found in the Appendix) and 5 969 329 rows, of which fraudulent cases are 0.125%. This data set can be called extremely imbalanced. Some simple feature engineering was done on variables like *Expires_Date* or *Acct_Open_Date* to calculate how many days the card is valid.

4.3 Finding the Best Collection of Features and Number of Clusters

After preprocessing, our dataset has different types of features. Some of them, like "Amount" or "Yearly Income—Person" are float; some, like "Current Age" or "Day", are integer, and others like "CardType Debit" or "Gender Male" are binary. The

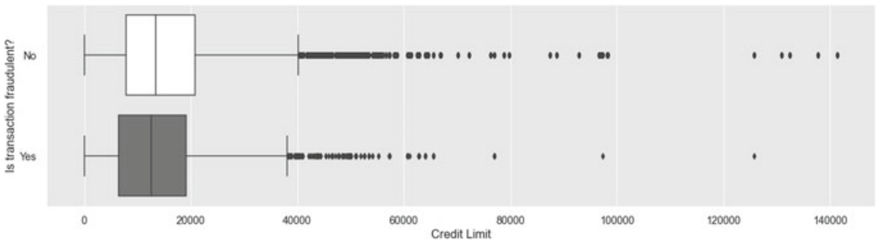


Fig. 8 Fraudsters attacks by credit limit

binary features make issues when using the k-means algorithm [20]. Application of k-means clustering and Euclidean distance for binary data is a controversial topic in the research literature. However, we have chosen this way and the experiments have proved its suitability. We standardized features using the $StandardScale$ package in $Python$, which re-scales each feature separately to have a mean of zero and a standard deviation of one. Such a scaling in binary case of features allows passing the information about the distribution of raw feature values to the standardized features.

There are at least a few approaches to cluster the transactions of the training set into separate sub-sets. One way of clustering could be based on gut feeling and experience. For example, to have clusters of a young male with a higher credit limit, a young female with a higher credit limit, etc. It feels right to think that fraud cases could happen to older people with middle or low-level incomes, as shown in Figs. 6 and 8.

As suggested previously, we are using the Silhouette Score to evaluate the goodness of the clustering. For instance, when splitting clusters as described above (by age, gender, and credit limit), we got a Silhouette score equal to 0.3992, which is not very high and implies that clusters' borders are close to each other. We have tried more than 280 combinations of features and a number of clusters, and the best one with a score of 0.862248 was *[CardType_Debit, HasChip_YES, Use_Chip_Swipe_Transaction]*. An interesting fact is that all three features used for clustering are binary. In our case, features mean:

- *CardType_Debit* feature marks if a transaction was done using a Debit card.
- *HasChip_YES* feature marks if a transaction was done with the card which has a chip. A debit or credit card can have a chip that holds an integrated microchip along with the traditional magnetic stripe. The chip gives customers more security because they are harder to skim.
- *Use_Chip_Swipe_Transaction* feature marks the transactions that are done by swiping the card through the card reader and following its instructions.

The "Elbow" method proves that 4 clusters are the optimal value with the selected features (see Fig. 9).

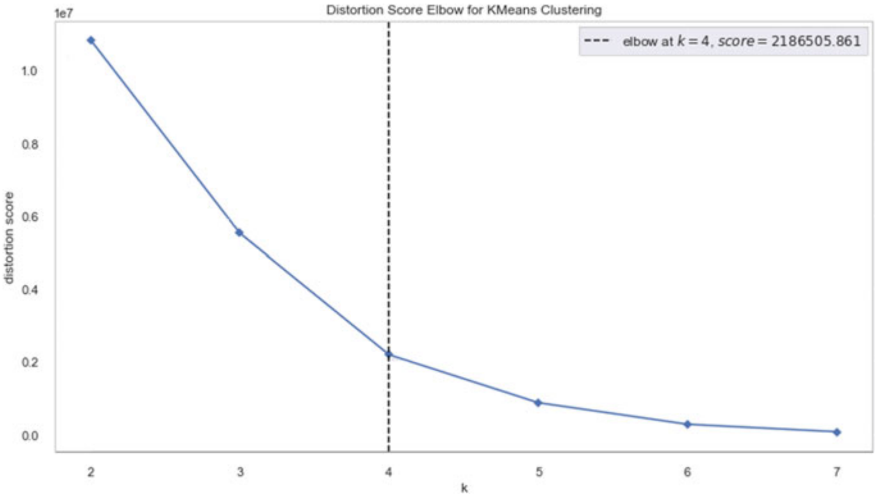


Fig. 9 Elbow for k-means clustering

Table 1 Training set clusters characteristics

Training set clusters characteristics		
Cluster	Number of data points	Share of fraudulent transactions (%)
1	1 522 231	0.19
2	596 897	0.11
3	2 533 953	0.15
4	1 316 248	0.02

In the graph Fig. 9, distortion score—the mean sum of squared distances to centers—is marked on the y-axis while the number of clusters is on the x-axis. The dotted vertical line marks the "elbow" point found using the "knee point detection algorithm" [21].

After the splitting training set into four clusters, we can notice that they are not equal by the size or by the share of the fraudulent cases, as shown in the table below (see Table 1).

Since clustering was done based on the three features, it is possible to plot cluster centers in 3D. We can see from Fig. 10 that one of the clusters is located much further than the others and that this cluster has the lowest number of data points.

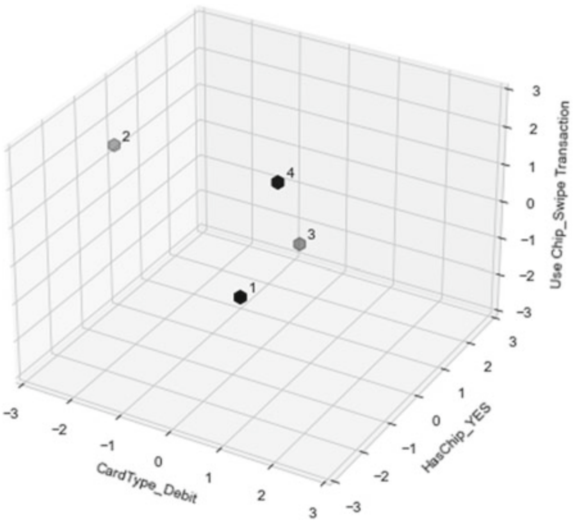


Fig. 10 Centers of the clusters

4.4 Undersampling and Model Fitting

XGBoost classifier was chosen as a machine learning model for each cluster. XGBoost—Extreme gradient boosting—is a widely used machine learning algorithm and usually achieves 'state-of-art' results in competitions like Kaggle. It is built on a gradient-boosting decision tree algorithm. XGBoost is a part of the Ensemble methods of the supervised machine learning algorithms family. However, it is not the only model that could be chosen for this task. Competing candidate is Light GBM—Light Gradient Boosting Machine—for the fraud detection tasks [22].

The validation set is used to individually choose the best-performing undersampling percentage for each cluster. We have chosen undersampling percent, and after resampling, sub-classifiers were trained. We go through all validation data set points and use one of the sub-classifier for the decision. The criterion for selecting an appropriate classifier is the minimal Euclidean distance between the validation set point and the corresponding cluster center of training data. We measure the best performing undersampling percent for each training cluster by computing the F1 score on the validation data. We repeated this procedure 99 times by checking undersampling percentages from 1 to 99.

4.5 Training Results

We see in Table 2 that there is no linear or direct relationship between undersampling percent, the share of fraudulent cases, or the size of the cluster. However, we can see

Table 2 Undersampling performance

Undersampling results							
C	Train set size	Valid. set size	Share of fraud (%)	Undersampling percent	Share in train set after sampling (%)	F1 score of valid. set	*Recall* of valid. set
1	1 522 231	653 420	0.19	87	0.22	0.85	0.75
2	596 897	255 489	0.11	91	0.12	0.77	0.63
3	2 533 953	1 084 892	0.15	49	0.30	0.82	0.72
4	1 316 248	564 483	0.02	7	0.27	0.40	0.31

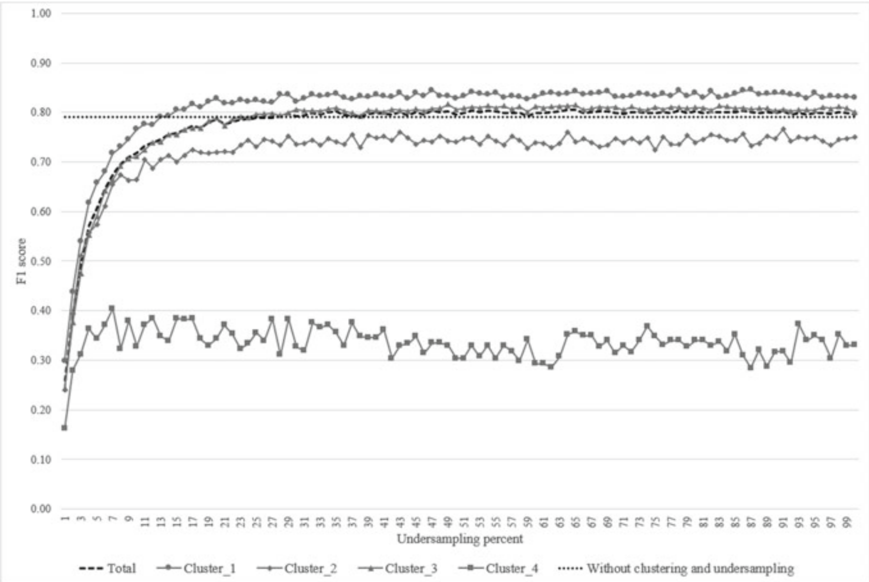


Fig. 11 F1 score with different undersampling %

that the worst-performing cluster no.4 has the lowest share of fraudulent cases, and to achieve better results, it required a low undersampling percent.

Plotted results (see Fig. 11) show that the undersampling percent and F1 score do not have a linear dependency, and the F1 score has fluctuations.

Our baseline is the *recall* equal to 0.69 without the training strategy proposed in Sect. 3. After clustering a training set into four clusters, we ran the training of sub-classifiers ten times, including 99 different values of undersampling percent. The average result of the *recall* was 0.71. With every ten runs, we got improved results compared to the baseline. There was a slight variation between the runs' results, although negligible.

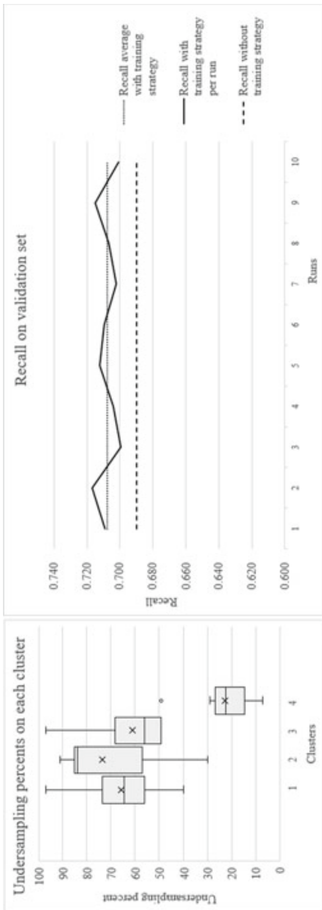


Fig. 12 Undersampling percent and *recall* variations

Figure 12 shows that undersampling percent varies a lot for each cluster with different runs. However, the trend that the cluster with the lowest number of fraudulent cases (in our case, cluster no.4) has the lower undersample percent is obvious.

4.6 Classification Results

The most critical part is to measure performance on the test data set, which are future fraudulent transactions. The procedure with the test data set is similar to the validation data set. The test set data were standardized to specify which classifier will make a prediction decision. The classifier which makes a decision is chosen by the minimal Euclidean distance between the validation set point and the corresponding cluster center of training data. To get reliable results, we ran the prediction ten times as well. Additionally, we calculated the *recall* on the test set with no training strategy to establish a baseline.

Figure 13 shows that experimental results imply that clustering-based classification with optimal undersampling improved the machine learning performance. When predicting fraudulent transactions with the XGBoost classifier with no training strategy, the *recall* is 0.845, and our strategy managed to increase the performance significantly to 0.867.

By comparing the absolute numbers (see Fig. 14), we see that the classification of fraudulent cases that were labeled as regular decreased from 323 to 278, i.e. by 13.9%.

We perform a proportion test using the p-value approach to see if the improvement is significant or in the other words we test if proportions of population is significantly different. First of all, we formulate the hypothesis:

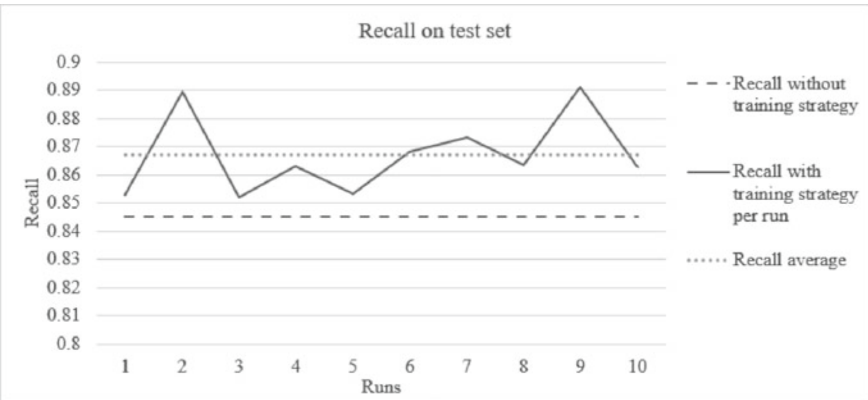


Fig. 13 F1 score on test data set

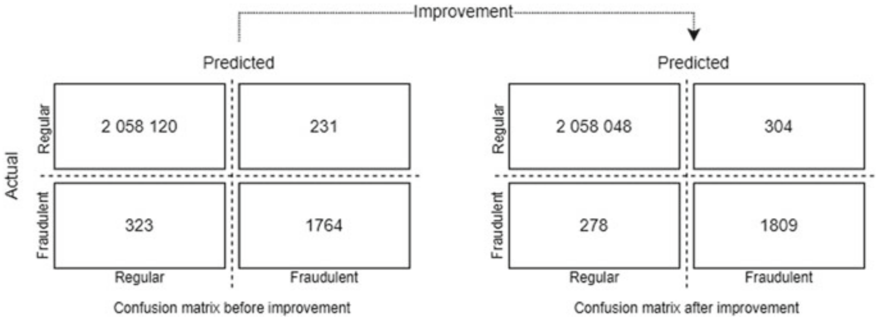


Fig. 14 Confusion matrix before and after applied strategy

$$H_0 : p_0 \geq p$$
$$H_1 : p_0 < p$$

where p and p_0 stand for the conversion rate of the TP after applying proposed strategy and before applying a proposed strategy, respectively. Here we use confidence level of 95%. Test statistic Z is calculated:

$$Z = \frac{\tilde{p} - \tilde{p}_0}{\sqrt{\frac{\tilde{p}_0(1-\tilde{p}_0)}{n} + \frac{\tilde{p}(1-\tilde{p})}{n}}} \tag{8}$$

where $\tilde{p}$ and $\tilde{p}_0$ are estimated proportion before and after applying our approach, respectively. n is a number of total fraudulent cases in the test data set. In our case,

$$\tilde{p}_0 = 1764/2087 = 0.8452$$
$$\tilde{p} = 1809/2087 = 0.8668$$
$$Z = 1.9849$$

Using Z-score table with α = 0.05, we have p value = 0.0256. In this case, we can conclude that the obtained increase in the classifier performance is significant.

5 Conclusions

Fraud detection is an activity to prevent financial assets from being obtained by fraudsters. The goal of the research is to increase machine learning predictions quality on fraudulent cases and to decrease false negative cases in prediction. Fraudulent data like credit card transactions usually are imbalanced data. In this case, standard machine learning algorithms can not reach the expected levels of quality. This paper proposes a clustering-based classification approach to improve the *recall*. The idea lies in undersampling of each cluster and further training the sub-classifiers by the undersampled data. The decision on the dependence of a particular transaction on a regular or fraudulent class is made by one of the sub-classifiers. For the experimental evaluation, we use a credit card transaction database. Our baseline *recall* is 0.845, obtained after the direct training of the XGBoost classifier. By applying the proposed approach, we improved the *recall* to 0.867. Moreover, classification of fraudulent cases that were labeled as regular decreased from 323 to 278, i.e. by 13.9%, that is significant. Moreover, we found that, when the training set is properly split into clusters and balanced separately for each cluster, the prediction score becomes higher.

6 Future Work

We plan to explore the efficiency of the proposed approach to other data sets of fraudulent transactions. Moreover, we plan to explore if the chosen encoding method of categorical features impacts the k-means clustering method. Additionally, we aim to improve sampling methods and use a more advanced approach than random undersampling.

Appendix

List of the final features used for the model training:

- **Current Age**—current age of the card owner.
- **Retirement Age**—retirement age of the card owner.
- **Zipcode**—zipcode of the card owner.
- **Per Capita Income—Zipcode**—per capita income grouped by zipcode of the card owner.
- **Yearly Income—Person**—yearly income of the card owner.
- **Total Debt**—card owner' total debt.
- **FICO Score**—is used by lenders to help make accurate, reliable, and fast credit risk decisions across the customer life cycle. The credit risk score rank-orders consumers by how likely they are to pay their credit obligations as agreed. Even though score intervals vary depending on the credit scoring model, credit scores from 580 to 669 are generally treated as fair; 670 to 739 are treated as a good; 740 to 799 are treated as very good, and 800 and up are treated as an excellent.
- **Num Credit Cards**—number of cards owned by the same person.
- **Credit Limit**—credit limit of the card.
- **Gender_Male**—gender of the card owner.
- **CardBrand_Discover**—binary feature representing if the card is "Discover".
- **CardBrand_Visa**—binary feature representing if the card is "Visa" (Otherwise, card is "MasterCard").
- **CardType_Debit**—binary feature representing if the card is Debit.

- **CardType_Debit (Prepaid)**—binary feature representing if the card is Debit Prepaid (Otherwise, the card is Credit).
- **HasChip_YES** —binary feature representing if the card has a chip. Chips are the small, square computer chips that appear on debit, credit and prepaid cards to help safeguard them against fraud.
- **Month**—the number of the month when transaction was made.
- **Day**—the number of the day when transaction was made.
- **MCC**—id of the merchant. For instance, Apple (MCC = 5045) or McDonalds (MCC = 5814).
- **City_cat**—city of the card owner.
- **Merchant_City_cat**—city of the merchant.
- **State_cat**—State of the card owner.
- **Use Chip_Online Transaction**—binary feature representing if the the transaction was made online.
- **Use Chip_Swipe Transaction**—binary feature representing if the the transaction was made by swiping through the card reader.
- **Valid_in_Days**—number of days until card will be expired.
- **hour_bin**—hour bin, for instance 12:00–13:00, when transaction was made.
- **Amount**—transferred amount.
- **Error_cat1** and **Error_cat2**—error that happen during the transaction.
- **Is_Fraud_Yes**—target feature. It is binary feature representing if the transaction is labeled as fraudulent or regular.

References

1. Kemp, S., Buil-Gil, D., Moneva, A., Miró-Llinares, F., Díaz-Castaño, N.: Empty streets, busy internet: a time-series analysis of cybercrime and fraud trends during COVID-19. J. Contemp. Crim. Justice **37**(4) (2021). https://doi.org/10.1177/10439862211027986
2. Fernández, A., García, S., Galar, M., Prati, R.C., Krawczyk, B., Herrera, F.: Learning from imbalanced data sets (2018). https://doi.org/10.1007/978-3-319-98074-4
3. Assegie, T.A.: An optimized K-nearest neighbor based breast cancer detection. J. Robot. Control (JRC) **2**(3) (2021). https://doi.org/10.18196/jrc.2363
4. Calderon-Ramirez, S., et al.: Correcting data imbalance for semi-supervised COVID-19 detection using X-ray chest images. Appl. Soft Comput. **111** (2021). https://doi.org/10.1016/j.asoc.2021.107692
5. https://www.webofscience.com/wos/woscc/basic-search
6. Provost, F.: Machine learning from imbalanced data sets 101. In: Proceedings of the AAAI'2000 Workshop on ... (2000)
7. Park, S.H., Ha, Y.G.: Large imbalance data classification based on mapreduce for traffic accident prediction (2014). https://doi.org/10.1109/IMIS.2014.6
8. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority oversampling technique. J. Artif. Intell. Res. **16** (2002). https://doi.org/10.1613/jair.953
9. Zhai, J., Qi, J., Shen, C.: Binary imbalanced data classification based on diversity oversampling by generative models. Inf. Sci. **585** (2022). https://doi.org/10.1016/j.ins.2021.11.058

10. Langousis, A., Carsteanu, A.A.: Undersampling in action and at scale: application to the COVID-19 pandemic. Stoch. Environ. Res. Risk Assess. **34**(8) (2020). https://doi.org/10.1007/s00477-020-01821-0
11. Koziarski, M.: Radial-based undersampling for imbalanced data classification. Pattern Recognit. **102** (2020). https://doi.org/10.1016/j.patcog.2020.107262
12. Xie, X., Liu, H., Zeng, S., Lin, L., Li, W.: A novel progressively undersampling method based on the density peaks sequence for imbalanced data. Knowl.-Based Syst. **213** (2021). https://doi.org/10.1016/j.knosys.2020.106689
13. Zuech, R., Hancock, J., Khoshgoftaar, T.M.: Detecting web attacks using random undersampling and ensemble learners. J. Big Data **8**(1) (2021). https://doi.org/10.1186/s40537-021-00460-8
14. Kaur, P., Gosain, A.: Comparing the behavior of oversampling and undersampling approach of class imbalance learning by combining class imbalance problem with noise. In: Advances in Intelligent Systems and Computing, vol. 653 (2018). https://doi.org/10.1007/978-981-10-6602-3_3
15. S.M. V: An emperical study on the effect of resampling techniques in imbalaced datasets for improving consistency of classifiers. Int. J. Appl. Eng. Res. **14**(7) (2019)
16. He, H., Garcia, E.A.: Learning from imbalanced data. IEEE Trans. Knowl. Data Eng. **21**(9) (2009). https://doi.org/10.1109/TKDE.2008.239
17. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. **20**(C) (1987). https://doi.org/10.1016/0377-0427(87)90125-7
18. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12** (2011)
19. Altman, E.R.: Synthesizing credit card transactions (2019)
20. Ordonez, C.: Clustering binary data streams with K-means (2003). https://doi.org/10.1145/882082.882087
21. Satopää, V., Albrecht, J., Irwin, D., Raghavan, B.: Finding a 'kneedle' in a haystack: detecting knee points in system behavior (2011). https://doi.org/10.1109/ICDCSW.2011.20
22. Malik, E.F., Khaw, K.W., Belaton, B., Wong, W.P., Chew, X.: Credit card fraud detection using a new hybrid machine learning architecture. Mathematics **10**(9), 1480 (2022). https://doi.org/10.3390/math10091480

Baltic States in Global Value Chains: Quantifying International Production Sharing at Bilateral and Sectoral Levels

Giedrė Dzemydaitė, Brigita Šidlauskaitė-Riazanova, and Darjuš Bartkevičius

Abstract This monograph chapter presents data science applications in analysing global value chains (GVC) by decomposition of gross exports data. The purpose of this chapter is to evaluate changes in Baltic States' participation in global value chains by quantifying international production sharing at bilateral and sectoral levels. To achieve this purpose, we used an accounting framework that decomposed a country's gross exports into various value-added components, including exports value-added, domestic content in intermediate exports, foreign content, and other double-counted value-added components. Such a framework integrates all the previous vertical specialization and value-added trade approaches into a unified framework. It makes it possible to assess countries' participation in global value chains. We presented the disaggregated decomposition results of Baltic States with their trading partners in 56 sectors from 2000 to 2014 based on the World Input–Output Database. We revealed the patterns of cross-country production sharing. Empirical research results showed that the Baltic States' participation in global value chains was growing during the research period. The biggest driver behind the growth was determined by foreign value-added increases in the countries' exports. Gross exports decomposition into value-added elements revealed that a major part of foreign value-added was impacted by value-added originating from third countries. The growth of double-counted value-added was observed over different economic sectors, which indicated that countries tend to participate in longer global value chains. The paper is organised as follows. First, the input–output model as a basis for quantifying international production sharing is described. It provides the methodology used to break down gross exports into separate value-added components. Then the study's results on the involvement of the Baltic States in GVCs are presented.

G. Dzemydaitė (✉) · B. Šidlauskaitė-Riazanova · D. Bartkevičius
Faculty of Economics and Business Administration, Vilnius University, Saulėtekio av. 9, Vilnius, Lithuania
e-mail: giedre.dzemydaite@evaf.vu.lt

B. Šidlauskaitė-Riazanova
e-mail: brigita.sidlauskaite@evaf.vu.lt

G. Dzemyda et al. (eds.), *Data Science in Applications*,
Studies in Computational Intelligence 1084,
https://doi.org/10.1007/978-3-031-24453-7_4

Keywords Global value chains · International production sharing · Input–output model · Decomposition of exports data · Baltic states

1 Introduction

Global value chains are becoming an essential element in understanding the global economy. The value chain consists of the entire lifecycle of a product or service and the steps from the product's conception to delivery to the final consumer. This phenomenon is unique. The whole process chain is fragmented not only at the level of individual companies but also at the level of individual countries. Companies are gradually moving from a traditional product-oriented approach to a task-oriented approach, which creates the possibility to specialise at a particular stage of the production where the company has qualifications to gain a competitive advantage. Involvement in GVCs varies significantly across sectors, countries, and even regions.

Evaluation techniques of countries' participation in global value chains have evolved during the last decade. Koopman et al. [24] were among the first researchers to distinguish the share of domestic and foreign added value in the gross exports. They suggested a framework for analysing the countries' participation in the GVCs. Daudin et al. [8], Johnson and Noguera [18], and Koopman et al. [22] suggested a methodology for assessing participation in the GVCs, while [40] presented a new system in which gross exports were broken down into very detailed sections. This allowed for a very detailed assessment of the countries' participation in the GVCs.

A methodological approach for evaluating countries' participation in GVCs has been developed further in the most recent research [19]. Antràs and Chor [1] have attempted to describe the various models that have provided valuable quantitative insights into the aggregate consequences of GVCs. Borin and Mancini [3] have extended the set of possible measures to address a wider range of empirical issues. These improvements are likely to become increasingly relevant from a quantitative point of view when the inter-country input–output data will become more and more detailed.

In recent studies, there is an emerging focus on various factors, economic shocks, and events that make an impact on changes in countries' participation in global value chains [13]. For example, Song et al. [37] have found that the impact of the covid-19 pandemic varied remarkably in different industries regarding forward and backward GVC participation and GVC division of labour. Chen et al. [6] provided insights into which regions in the EU and UK would be most affected due to Brexit through the analysis of GVCs. The 4.0 industrial revolution raises questions about how the GVCs will be affected by technological transformation, robotization, and new advanced technologies usage in business [5, 12, 29]. The evolutionary approach to GVCs analysis is still not covered topic [4, 41]. These studies reveal the broad applicability of data science techniques in analyzing various research questions related to GVCs and their relevance for diverse territories and events.

However, most of the studies are focusing on the analysis of major global economies and only a few of them analyse data on small and open economies. Therefore, we selected to analyse the Baltic States which are small and open economies with a path of an economic integration with the EU after the collapse of the Soviet Union. The number of studies on the Baltic States is limited, and in most cases, these countries are only considered in the context of larger regions [7, 20, 21], more focused on certain economic sectors, specialization and influence on economic growth [10, 11, 35] or global inter-industry linkages [9, 36]. Kordalska and Olczyk [25] found a growing involvement in GVCs. Hagemejer [14], having analysed the new EU countries, found that sectors that have imported intermediate goods have experienced higher productivity growth. Moreover, faster productivity growth was found in sectors further away from the final demand and in sectors exporting intermediate goods. They associated the growth of sectoral productivity with the position of a sector or in the GVCs. Hagemejer and Ghodsi [15] found in their study the convergence of the GVC position indicator between the old European Union (EU) and the new EU members. However, the vast majority of the EU's old countries are located at the end of the GVCs chains, i.e., to a large extent specialising in the final stages of the production process. Banh et al. [2] found that the highest participation in the GVCs between the three Baltic States was recorded in Estonia. The majority of Estonia's involvement in the GVCs was driven by a high share of foreign value-added in the country's exports.

It should be noted that the results are insufficient to highlight clear trends due to different investigation periods, the number of countries involved, and different databases and evaluation methods used. For the Baltic States, as small and open economies, participation in GVCs can significantly impact economic development and strengthen international competition. For this reason, it is appropriate to carry out a detailed analysis of the value-added structure of gross exports and exports of individual sectors to assess the level of involvement of countries in the GVCs.

One of the biggest challenges for researchers is the lack of relevant data. Global input–output tables, which require integrating national supply and use tables with official statistics, must be used to explore the country's participation in GVCs fully. This leads to a significant delay in the survey data period from the current year. On the other hand, structural changes in the economy are relatively slow, and the latest methodology in scientific literature offers new opportunities—the possibility of a highly detailed decomposition of gross exports into value-added components with different economic meanings.

The remainder of this paper is organized as follows. The next section goes into detail about the input–output model for quantifying international production sharing. The third section introduces the methodological approach by explaining the breakdown of gross exports into separate value-added components. The fourth section presents the results of the involvement of the Baltic States in GVCs. Finally, the fifth section summarizes and concludes.

2 Input–Output Model for Quantifying International Production Sharing

This research is based on a global input–output model. A modern national input–output model was developed by V. Leontiev in the late 1930s [28]. The most important advantage of the proposed approach is that it makes it possible, without significant information losses, to aggregate a large number of statistics, identify their interdependencies, and describe in mathematical terms the most important macroeconomic indicators and the distribution of production and trade flows by industry.

The national input–output model has been widely used to investigate one country's economy. At a later stage, an adaptation of the gravity model [26] was proposed for the international assessment of cross-border flows between several countries. However, a broader investigation of relations between different countries was introduced relatively recently, with more detailed investigations only taking place between 1970 and 1990 [31–34] and the impact of European economic integration was examined in more detail later.

It is drawn up in the form of a table in which the economic sectors are placed in the same order in rows and columns. The rows additionally distinguish specific value-added elements by income (labour and capital income) and in columns by final consumption elements (household consumption, government expenditure, investment, etc.). The theoretical model for M countries and the N sector is shown in Fig. 1.

The red rectangle reflects the matrix of intermediate production, whose dimensions, in this case, are 2464 × 2464. Each matrix field reflects the export value of a given country and sector to the country and sector concerned. For example, the first field represents the production of Sector 1 in Country 1 and the consumption in the same sector in the same country. At that time, the lower value of the first column

			Intermediate consumption by countries and sectors							**Final consumption**			**Total production**
			Country 1				Country M			Country 1	…	Country M	
			Sec-tor1	…	Sec-torN	…	Sec-tor1	…	SectorN				
Supply by countries and sectors	Country1	Sector1											
		…											
		SectorN											
	…	…											
	Country M	Sector1											
		…											
		SectorN											
Value-added													
Total production													

Fig. 1 Global input–output table model with M countries and N sectors

represents the value of Sector N in Country M and exports to Sector 1 in Country 1. The blue rectangle reflects the matrix of final demand in which the first field refers to the value of the final production produced in Sector 1 in Country 1 and consumed in Country 1. At this time, the lower value of the first column indicates the consumption of the final production of Country M by Sector N in Country 1. Finally, the green rectangle reflects the value-added matrix in which each field indicates the amount of value-added in the respective country and sector. All remaining fields of the matrix can be interpreted in the same logic.

In order to carry out an input–output analysis, we need to calculate direct input coefficients. The direct input coefficient a_{ij} is the volume of resource i needed to produce a unit of product j:

$$a_{ij} = \frac{z_{ij}}{X_j} \tag{1}$$

where X_j—j gross output, z_{ij}, —input of i which is necessary for the production of product j. The formula can be transformed into this expression:

$$z_{ij} = a_{ij} X_j \tag{2}$$

We can then write the input–output matrix as follows:

$$\begin{cases} X_1 &= a_{11}X_1 + a_{12}X_2 + \dots a_{1n}X_n + Y_1 \\ X_2 &= a_{21}X_1 + a_{22}X_2 + \dots a_{2n}X_n + Y_2 \\ \vdots &= \\ X_n &= a_{n1}X_1 + a_{n2}X_2 + \dots a_{nn}X_n + Y_n \end{cases} \tag{3}$$

where Y is the final demand for the sector's output. The reconversion of the matrix gives us:

$$\begin{cases} (1 - a_{11})X_1 - a_{12}X_2 - \dots - a_{1n}X_n = Y_1 \\ -a_{21}X_1 + (1 - a_{22})X_2 - \dots - a_{2n}X_n = Y_2 \\ \vdots \\ -a_{n1}X_1 - a_{n2}X_2 - \dots + (1 - a_{nn})X_n = Y_n \end{cases} \tag{4}$$

The input–output model then takes the following form:

$$\begin{bmatrix} 1 - a_{11} & -a_{12} & \dots -a_{1n} \\ -a_{21} & 1 - a_{22} & \dots -a_{2n} \\ \vdots & \vdots & \ddots \vdots \\ -a_{n1} & -a_{n2} & \dots 1 - a_{nn} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} \tag{5}$$

Fig. 2 Global input–output model. *Source* compiled by the authors based on Miller and Blair [28], Linden [39]

$$\begin{array}{ccccc|c}
z^{11}F^{11} & \dots & z^{1s}F^{1s} & \dots & z^{1R}F^{1R} & x^{1}\\
\vdots & \ddots & \vdots & & \vdots & \vdots\\
 & & & & & x^{r}\\
z^{r1}F^{r1} & \dots & z^{rs}F^{rs} & \dots & z^{rR}F^{rR} & \vdots\\
\vdots & & \vdots & \ddots & \vdots & x^{R}\\
z^{R1}F^{R1} & \dots & z^{Rs}F^{Rs} & \dots & z^{RR}F^{RR} & \\
\hline
V^{1}\,V_f^{1} & \cdots & V^{s}\,V_f^{s} & \cdots & V^{R}\,V_f^{R} & \\
\hline
x^{1\prime}\,y^{1\prime} & \cdots & x^{s\prime}\,y^{s\prime} & \cdots & x^{R\prime}\,y^{R\prime} &
\end{array}$$

In a simplified way, we can write this equation:

$$(I - A)X = Y \tag{6}$$

where A is a matrix of direct input coefficients, I—identity matrix, Y is the vector of final demand.

The total output will then be equal to:

$$X = (I - A)^{-1}Y \tag{7}$$

The formula reflects the usual national input–output model. The global input–output model is similar to the national input–output model. The global input–output model includes all the intermediate and final consumption flows between all countries. The complete model is shown in Fig. 2.

Intermediate input and final demand production are denoted respectively by z_{ij}^{rs} and f_{ij}^{rs}, where i and j are the sectors of origin and destination ($i, j = 1, \ldots, $ n), r and s are the countries of origin and destination ($r, s = 1, \ldots$ R), f—categories of final demand ($f = 1, \ldots,$ F). Z^{rs} is the n × n matrix of intermediate production flows from region r to region s, F^{rs} is a n × F matrix of the of final demand flows from region r to region s. X^{r} *is* the vector of the total output per sector in region r. V^{s} is the P*n matrix of value-added created in region s, where P is the number of value-added categories. V_f^{s} is the P*F matrix of primary inputs for final demand in the region s. Finally, y^{s} is a vector of final production demand Y in region s.

The main difference between the national and global input–output model is that the global model reflects indirect effects between the countries involved, whereas the national model only analyses the direct effects of imports. The direct effect is the demand for imports from other countries needed to produce one unit of production in the country j. The indirect effect is the demand needed to produce the output caused by the indirect effect, including the demand of the country itself. Thus, compared to the national input–output model, the global input–output model offers a more detailed opportunity for analysis.

This research uses the World input–output database (WIOD), which, due to detailed data, is used more frequently by other authors [14, 40]. The main reasons for this are: update of data—data for the period 2000–2014, detailed sector breakdown—56 economic sectors; access to data—information easily and publicly accessible to all users. It is noted that the data for the 2014 global input–output tables are sufficiently relevant due to the complexity of their preparation. The global input–output tables in the WIOD database have been compiled by combining national supply-use tables with official bilateral international trade statistics. The resulting data gaps are filled using specific models [38]. The global input–output tables were thus constructed for 15 years, 44 countries (including the rest of the world), and 56 sectors. All tables have a uniform structure and can be easily broken down into several separate intermediate output, final demand, and value-added matrices with dimensions of 2464 × 2464, 2462 × 220, and 1 × 2464, respectively.

Drawing up global input–output tables is a complex process requiring a combination of different countries' statistical trade flows. A detailed description of the preparation of the database is provided by Timmer et al. [38]. The structure of this model is appropriate for the detailed decomposition of the country's gross exports into value-added components and for calculating the indicators of participation in the GVCs.

3 Breakdown of Gross Exports into Separate Value-Added Components

Classical trading models do not always allow a proper assessment of developments in international trade (Ishii and Yi 1997). Vertical specialisation is one of the first international trade indicators still used in scientific literature. This indicator is defined as the imported goods used as inputs to produce a country's export goods (Ishii and Yi 1997; [16, 17]). Although traditional international trade data are sufficient to calculate the VS, this indicator is not sufficient to assess the participation in the GVCs. For example, the share of imported production in exports may be relatively low. However, the country may specialise in the initial stage of intermediate production, the export of which is used for another country's export. This indicator is referred to in scientific literature as VS1 [16, 17]. It should be noted that the calculation of the VS1 is more complex and requires at least bilateral linking of trade flows in input–output tables. This data makes it possible to calculate not only VS1 but also the derived indicator VS1*—the value of a country's exported goods that are used as imported inputs by other counties to produce final goods that are shipped back home [8]. The VS, VS1, and VS1* indicators discussed are focused on both foreign and domestic value-added and have so far been widely used in scientific literature to analyse countries' involvement in the GVCs.

More recent literature points out that the indicators described by Hummels et al. [16] had several key weaknesses. For example, it does not consider the fact that countries may import intermediate products, process them, and further export them to other countries for final production. It also does not consider the fact that the countries may import intermediate products that already contain the country's own value-added [23]. In other words, vertical specialisation only reflects foreign value-added in domestic exports.

Koopman et al. [24] were among the first authors to distinguish two important indicators: domestic value-added and foreign value-added in exports (DVA and FVA, respectively). The latter corresponds to the VS indicator already presented. Johnson and Noguera [18] presented a new indicator—the ratio of gross value-added generated in the sector and country concerned and consumed in the country of destination to the value of gross exports (VAX). Koopman et al. [23, 22] carried out the first detailed decomposition of the value-added in international trade and presented new indicators that have not been analysed before. He also systematised many of the indicators used up to then, i.e., VS, VS1, VS1*, and VAX. He divided the value of gross exports into nine different components. This fragmentation has created new opportunities for the analysis of value-added in international trade. It has also become a significant source for further research and the search for new methodologies.

Wang et al. [40] expanded the gross export breakdown into nine components with eight new components. The detailed breakdown of gross exports into 16 components makes it possible not only to assess detailed export indicators but also to use the results to calculate participation in GVCs. Wang et al. [40] present the final equation for the gross export decomposition and describe its components.

The global input–output table serves as a basic data source for further calculations. The equation for the gross breakdown of exports:

$$
\begin{aligned}
E^{sr} = {} & \underbrace{\left(V^s B^{ss}\right)^T \# Y^{sr}}_{(1)-DVA_{FIN}} + \underbrace{\left(V^s L^{ss}\right)^T \# \left(A^{sr} B^{rr} Y^{rr}\right)}_{(2)-DVA_{INT}} \\
& + \underbrace{\left(V^s L^{ss}\right)^T \# \left[A^{sr} \sum\nolimits_{t \neq s,r}^{G} B^{rt} Y^{tt} + A^{sr} B^{rr} \sum\nolimits_{t \neq s,r}^{G} Y^{rt} + A^{sr} \sum\nolimits_{t \neq s,r}^{G} B^{rt} \sum\nolimits_{u \neq s,t}^{G} Y^{tu}\right]}_{(3)-DVA\_INTrex} \\
& + \underbrace{\left(V^s L^{ss}\right)^T \# \left[A^{sr} B^{rr} Y^{rs} + A^{sr} \sum\nolimits_{t \neq s,r}^{G} B^{rt} Y^{ts} + A^{sr} B^{rs} Y^{ss}\right]}_{(4)-RDV\_G} \\
& + \underbrace{\left[\left(V^s L^{ss}\right)^T \# \left(A^{sr} B^{rs} \sum\nolimits_{t \neq s}^{G} Y^{st}\right) + \left(V^s L^{ss} \sum\nolimits_{t \neq s}^{G} A^{st} B^{ts}\right)^T \# \left(A^{sr} X^r\right)\right]}_{(5)-DDC} \\
& + \underbrace{\left[\left(V^r B^{rs}\right)^T \# Y^{sr} + \left(\sum\nolimits_{t \neq s,r}^{G} V^t B^{ts}\right)^T \# Y^{sr}\right]}_{(6)-FVA\_FIN} \\
& + \underbrace{\left[\left(V^r B^{rs}\right)^T \# \left(A^{sr} L^{rr} Y^{rr}\right) + \left(\sum\nolimits_{t \neq s,r}^{G} V^t B^{ts}\right)^T \# \left(A^{sr} L^{rr} Y^{rr}\right)\right]}_{(7)-FVA\_INT} \\
& + \underbrace{\left[\left(V^r B^{rs}\right)^T \# \left(A^{sr} L^{rr} E^{r*}\right) + \left(\sum\nolimits_{t \neq s,r}^{G} V^t B^{ts}\right)^T \# \left(A^{sr} L^{rr} E^{r*}\right)\right]}_{(8)-FDC}
\end{aligned}
\tag{8}
$$

where E is the country's gross exports, A is a matrix of direct input coefficients, L—Leontief inverse matrix, Y is the vector of final demand, V is the vector of value-added coefficients, B is the global Leontief inverse matrix, "#" is an element-wise matrix multiplication operation. The detailed proof of the equation can be found in the Wang et al. [40] annex.

The gross export breakdown equation consists of 16 components, divided into 8 categories, which can be aggregated into larger groups of value-added components. Detailed decomposition of all components used into appropriate groups is presented in Table 1.

First, the decomposition presented makes it possible to distinguish the shares of domestic value-added and foreign value-added in gross exports. It should be noted that domestic value-added in gross exports is not equivalent to the VAX indicator presented by Johnson and Noguera [18], as it defines only the country of origin and not the place of value-added consumption [27]. Thus, the share of domestic value-added in exports is grouped into sub-sets. First, it is the VAX element that has already been discussed, which is presented in Wang et al. [40] as VAX_G. VAX_G indicator is further subdivided into five components: the share of domestic value-added in the exports of final and intermediate production, the indicators DVA_FIN and DVA_INT, respectively, and the share of domestic value-added in intermediate exports used by the direct importer to produce its final domestic goods and consumed there, to produce final goods exports to third countries and to produce intermediate exports to third countries, the indicators DVA_INTrex1, DVA_INTrex2 and DVA_INTrex3 respectively [30].

Another set of indicators reflects the domestic value-added first exported and then returned home (RDV_G), which can be broken down into three sub-components: returned domestic value-added in final goods imports from the direct importer (RDV_FIN1), returned domestic value-added in final goods imports via third countries (RDV_FIN2) and returned domestic value-added in intermediate imports used to produce final goods consumed at home (RDV_INT) [14].

The third set of indicators reflects the double-counted value-added presented by Koopman et al. [22] as PDC. This value-added category is divided into four sub-components: double-counted domestic value-added used to produce final goods exports (DDC_FIN), double-counted domestic value-added used to produce intermediate exports (DDC_INT), direct importer's value-added double-counted in home country's exports production (MDC) and third countries' value-added double-counted in home country's exports production (ODC) [40].

Finally, the decomposition of gross export is completed by the FVA, more commonly named vertical specialisation (VS) in literature, to indicate the use of intermediate products imported from foreign countries for the production and export of goods [16]. In the detailed breakdown of gross exports this indicator is subdivided into sub-components: MVA_FIN and MVA_INT refer respectively to the direct importer's value-added in exporting country's final goods and intermediate goods exports, OVA_FIN and OVA_INT reflect the value-added in exporting country's final goods and intermediate goods exports.

Table 1 Decomposition of gross exports into value-added elements

E-gross exports	VAX_G—domestic value-added absorbed abroad	• DVA_FIN—domestic value-added in final goods exports • DVA _INT—domestic value-added in intermediate exports to the direct importer and is absorbed there • DVA_INTrex1—domestic value-added in intermediate exports used by the direct importer to produce intermediate exports for production of domestic used final goods in third countries • DVA_INTrex2—domestic value-added in intermediate exports used by the direct importer to produce final goods exports to third countries • DVA_INTrex3—domestic value-added in intermediate exports used by the direct importer to produce intermediate exports to third countries
	RDV_G—domestic value-added first exported then returned home	• RDV_FIN1—returned domestic value-added in final goods imports from the direct importer • RDV_FIN2—returned domestic value-added in final goods imports –via third countries • RDV_INT—returned domestic value-added in intermediate imports used produce final goods consumed at home
	PDC—pure double-counted terms	• DDC_FIN—Double-counted Domestic Value-added used to produce final goods exports • DDC_INT—Double-counted Domestic Value-added used to produce intermediate exports • MDC—direct importer's value-added double-counted in home country's exports production • ODC—third countries' value-added double-counted in home country's exports production

(continued)

Table 1 (continued)

	FVA—foreign value-added	• MVA_FIN—direct importer's value-added in exporting country's final goods exports • OVA_FIN—third countries' value-added in exporting country's final goods exports • MVA_INT—direct importer's value-added in exporting country's intermediate goods exports • OVA_INT—third countries' value-added in exporting country's intermediate goods exports

Source compiled by authors based on [40]

The decomposition of gross exports presented by Wang et al. [40] provides an opportunity to analyse in detail the value-added components of gross exports and assess the participation in GVC, taking into account the value-added generated domestically and abroad. Participation and position indicators are calculated using the relevant components of the gross export decomposition Koopman, etc. [23]. The equations of participation and position that are used to assess the participation of the Baltic States in the GVCs are given below.

The equation of the participation in GVC:

$$GVC_{participation} = \left(\frac{DVA_{INTrex}}{E} + \frac{FVA}{E}\right) * 100 \tag{9}$$

where E is the country's gross exports, DVA_INTrex is the domestic value-added in intermediate exports used by the direct importer to produce further exports, FVA is the foreign value-added used for domestically produced intermediate and final production exports.

The first element in formula nine is interpreted as forward participation (i.e. producing and shipping inputs that are further re-exported), and the second element reflects backward participation (i.e. using imported inputs to produce goods that are shipped abroad).

The equation of the position in GVC:

$$GVC_{position} = ln\left(1 + \frac{DVA\_INTrex}{E}\right) - \ln\left(1 + \frac{FVA}{E}\right) \tag{10}$$

The $GVC_{position}$ indicator defines the overall position of a country on an aggregate level in GVCs. This can be expressed as the log ratio of a country-sector's supply of intermediate products used in other countries' exports to the use of imported intermediate products in its production.

This can be expressed as the log ratio of a country's DVX supply to its FVA use. An indicator may have a positive value or a negative value. A positive value indicates that a country contributes more value-added to other countries' exports than other countries contribute to its exports, so is at the beginning of the GVC (upstream position). A negative value indicates that a country sources more foreign value-added inputs for its exports than it sells domestic inputs to other countries' exports, so is at the end of the GVC (downstream position).

4 Results of the Study on the Involvement of the Baltic States in GVCs

To assess the evolution of countries' participation in the GVC, a decomposition of gross exports into value-added components was carried out, which were subsequently used to calculate the participation rate in GVC. For this purpose, the distinction is made between forward participation and backward participation for all years of the reference period. The participation in GVC indicator is complemented by a position in GVC indicator in order not only to see how participation changes but also to assess whether the country takes an upstream or downstream position in GVCs. The figures are presented in Table 2.

Increased participation in GVCs can facilitate economic development, including productive employment opportunities, increasing labor productivity, and gaining a larger share of global exports. The survey shows that Estonia is a leader in the GVCs participation over the whole reference period compared to the other Baltic States. Participation in the GVCs increased in all Baltic States between 2000 and 2014. In Estonia, the rate increased from 43.8 to 46.6%, in Lithuania from 34.5 to 40.0% and in Latvia from 37.9 to 40.7%. Although Lithuania has recorded the fastest increase in participation in the GVCs, the country remains the least participating in the GVC state in the Baltic region. It should be noted that during the investigation period, the indicator reached its peak in 2011–2012 and has decreased slightly since then.

An important trend is observed when analysing the evolution of the position in the GVC indicator. First, only Latvia was considered as having a positive indicator value, i.e., Latvia was at the beginning of the GVCs and was more specialised in the initial stages of production. However, during the investigation period, there is a movement towards the production stages closer to the final consumer. Nevertheless, Lithuania's and Estonia's average position in GVCs shows that these countries were much closer to the final stages of production.

The decomposition of the participation in the GVC indicator into components makes it possible to determine which part of the value-added (foreign or domestic) has led to a change in the overall indicator.

First of all, the participation of all the Baltic States in the GVCs is largely based on backward participation. In other words, the share of foreign value-added in domestic exports (FVA) is greater than the domestic value-added in intermediate exports that the direct importer uses for further exports, also referred to as indirect exports

Table 2 Indicators of the participation and position in GVC in the Baltic States

	Forward participation in GVC, %	Backward participation in GVC, %	Participation in GVC, %	Position in GVC	Forward participation in GVC, %	Backward participation in GVC, %	Participation in GVC, %	Position in GVC	Forward participation in GVC, %	Backward participation in GVC, %	Participation in GVC, %	Position in GVC
	Estonia				Lithuania				Latvia			
2000	18,1	25,7	43,8	−0,06	16,7	17,9	34,5	−0,01	20,5	17,4	37,9	0,03
2001	17,6	25,7	43,3	−0,07	15,3	19,7	35,0	−0,04	20,8	17,7	38,5	0,03
2002	17,2	25,9	43,1	−0,07	16,1	17,9	34,0	−0,02	21,3	16,7	38,0	0,04
2003	17,9	25,0	42,9	−0,06	16,8	18,4	35,2	−0,01	20,5	17,5	38,0	0,03
2004	17,4	26,0	43,4	−0,07	17,0	20,7	37,7	−0,03	19,8	18,7	38,5	0,01
2005	16,7	27,3	43,9	−0,09	15,4	23,5	38,9	−0,07	18,9	19,1	38,0	0,00
2006	16,8	27,7	44,5	−0,09	14,8	24,2	39,0	−0,08	18,6	20,7	39,3	−0,02
2007	17,5	26,6	44,1	−0,07	17,5	20,7	38,3	−0,03	19,4	19,8	39,3	0,00
2008	17,3	27,4	44,6	−0,08	16,0	24,9	40,8	−0,07	19,9	19,1	39,0	0,01
2009	17,7	24,9	42,6	−0,06	15,8	21,4	37,1	−0,05	19,4	17,7	37,2	0,01
2010	17,2	28,5	45,7	−0,09	16,3	23,6	39,9	−0,06	19,6	20,2	39,7	−0,01
2011	15,9	31,9	47,9	−0,13	16,2	25,3	41,5	−0,08	19,6	21,5	41,1	−0,02
2012	16,0	32,0	48,0	−0,13	16,3	24,9	41,2	−0,07	18,8	22,9	41,6	−0,03
2013	16,0	31,3	47,3	−0,12	15,4	25,4	40,8	−0,08	19,1	22,1	41,1	−0,03
2014	15,8	30,8	46,6	−0,12	15,1	24,9	40,0	−0,08	18,9	21,9	40,7	−0,02

(DVA_INTrex). The exception was observed in Latvia at the beginning of the investigation period as well as in 2008 and 2009 when the higher domestic value-added indicator resulted in a positive position in the GVCs, i.e., Latvia was at production stages more distant from the final consumer.

The most considerable difference between forward and backward participation in the GVC indicators was in Estonia over 2011–2014, with a difference of 15–16 pp. Secondly, during the investigation period, there was a slight, albeit marginal, decline in the forward participation in the GVC in all the Baltic countries, with the most significant negative change in 2014 compared to 2000, again recorded in Estonia (2.3 pp).

Interestingly, the EU average forward participation in the GVCs, unlike the Baltic States, rose by 1.1 pp during the reference period (data not shown). The increase in the participation in GVC rate in all the Baltic countries resulted from an increase in backward participation. The largest changes were recorded in Lithuania, where the indicator rose from 17.9 to 24.9%, then Estonia from 25.7 to 30.8%, and Latvia from 17.4 to 21.9%. At that time, the EU average rose by 4.4 pp. to 26.2%.

At the time, the decline in participation in GVCs in the Baltic States discussed earlier in 2011–2014, was due to different factors. In Estonia, domestic value-added changed marginally, but foreign value-added in gross exports fell by 1.2 pp. The opposite was recorded in Lithuania, where forward participation declined by 1.1 pp, backward participation—by 0.4 pp. The backward participation even rose in Latvia by 4.5 pp, and forward participation declined by 0.7 pp at that time. In summary, the analysis showed the growing participation of the Baltic countries in the GVCs in the production stages closer to the final consumer.

The methodology used in the research allows a country's gross exports to be broken down into sixteen more detailed components. Some aggregated components have already been used to analyse participation and position in the GVC. However, the decomposition into more detailed components of value-added gives a much deeper view, particularly by detailing the structure of both foreign and domestic value-added and by distinguishing the remaining part of the value-added. To this end, this study provides a detailed breakdown of the Baltic countries' gross exports and provides an analysis of the results obtained. Table 3 shows the breakdown of the Baltic countries' gross exports into the main components for the years 2000, 2009, 2011, and 2014.

The first column reflects the country's total exports, which is the sum of the other elements in the table. First, the results show consistent growth in total exports in all Baltic countries. The average annual growth rate was 19% in Estonia, almost 20% in Lithuania, and over 16% in Latvia. It should be noted that the average growth for the last three years declined sharply to 4%, 6%, and 4%, respectively. This undoubtedly had a negative impact on countries' participation in the GVC rates. Exports from all Baltic countries are dominated by domestic value-added. However, the VAX_G indicator was the lowest in Estonia in 2014 at 56.5%, followed by a decrease of 8.0 pp. since the beginning of the investigation period, an even more significant decrease in Lithuania of 12.7 pp. to 64.2%, and in Latvia's by 7.2 pp. to 68.8%.

Looking at the structure of VAX_G in more detail, there are a number of similarities between the Baltic States. In Estonia and Latvia, the largest share of domestic

Table 3 Breakdown of Baltic countries' gross exports into value-added components

	EXP	VAX_G	DVA_FIN	DVA_INT	DVA_INTrex	RDV_G	FVA	MVA	OVA	PDC
	(1)	(2)	(2a)	(2b)	(2c)	(3)	(4)	(4a)	(4b)	(5)
Estonia										
2000	2,0	1,3	0,4	0,5	0,4	0,0	0,5	0,1	0,5	0,2
%	*100*	*64,4*	*20,3*	*26,0*	*18,1*	*0,1*	*25,7*	*3,0*	*22,7*	*9,8*
2009	9,5	6,2	1,8	2,7	1,7	0,0	2,4	0,2	2,2	0,9
%	*100*	*65,1*	*18,7*	*28,7*	*17,7*	*0,1*	*24,9*	*1,9*	*23,0*	*9,9*
2011	16,3	9,1	2,6	3,9	2,6	0,0	5,2	0,4	4,8	2,0
%	*100*	*56,0*	*16,2*	*23,8*	*15,9*	*0,1*	*31,9*	*2,6*	*29,3*	*12,0*
2014	18,3	10,3	3,0	4,4	2,9	0,0	5,6	0,4	5,2	2,3
%	*100*	*56,5*	*16,4*	*24,2*	*15,8*	*0,1*	*30,8*	*2,2*	*28,6*	*12,7*
Lithuania										
2000	3,2	2,4	1,0	0,9	0,5	0,0	0,6	0,0	0,5	0,2
%	*100*	*76,9*	*32,3*	*27,9*	*16,7*	*0,0*	*17,9*	*1,4*	*16,5*	*5,2*
2009	16,6	11,7	4,4	4,7	2,6	0,0	3,6	0,2	3,3	1,3
%	*100*	*70,5*	*26,3*	*28,4*	*15,8*	*0,1*	*21,4*	*1,4*	*20,0*	*8,0*
2011	27,6	17,4	6,2	6,7	4,5	0,0	7,0	0,4	6,6	3,2
%	*100*	*62,8*	*22,5*	*24,2*	*16,2*	*0,2*	*25,3*	*1,6*	*23,7*	*11,7*
2014	32,7	21,0	8,2	7,9	4,9	0,0	8,1	0,9	7,3	3,5
%	*100*	*64,2*	*25,0*	*24,1*	*15,1*	*0,1*	*24,9*	*2,6*	*22,3*	*10,8*
Latvia										
2000	2,0	1,6	0,4	0,7	0,4	0,0	0,4	0,0	0,3	0,1

(continued)

Table 3 (continued)

	EXP	VAX_G	DVA_FIN	DVA_INT	DVA_INTrex	RDV_G	FVA	MVA	OVA	PDC
%	*100*	*76,0*	*21,0*	*34,5*	*20,5*	*0,1*	*17,4*	*2,0*	*15,4*	*6,5*
2009	9,0	6,8	2,0	3,1	1,8	0,0	1,6	0,2	1,4	0,6
%	*100*	*75,4*	*22,1*	*33,8*	*19,4*	*0,2*	*17,7*	*1,7*	*16,0*	*6,7*
2011	12,9	8,9	2,7	3,7	2,5	0,0	2,8	0,2	2,5	1,2
%	*100*	*68,9*	*20,8*	*28,4*	*19,6*	*0,2*	*21,5*	*1,9*	*19,6*	*9,4*
2014	14,7	10,1	3,2	4,2	2,8	0,0	3,2	0,3	2,9	1,3
%	*100*	*68,8*	*21,6*	*28,3*	*18,9*	*0,2*	*21,9*	*1,9*	*19,9*	*9,1*

Note EXP is export value bln. USD. VAX_G (2) = (2a) + (2b) + (2c). DVA_INTrex (2c) is the sum of DVA_INTrex1, DVA_INTrex2 and DVA_INTrex3. RDV_G is the sum of RDV_FIN1, RDV_FIN2 and RDV_INT. FVA (4) = (4a) + (4b). MVA is the sum of MVA_FIN and MVA_INT. OVA is the sum of OVA_FIN and OVA_INT. PDC is the sum of DDC_FIN, DDC_INT, MDC and ODC. EXP (1) = (2) + (3) + (4) + (5)

value-added consumed abroad was the value-added related to intermediate output (DVA_INT). At the same time, Lithuania dominated the domestic value-added absorbed in the production of the final output (DVA_FIN) and amounted to about a quarter of the country's total exports. At the beginning of the period, this indicator accounted for almost one-third of the country's exports and decreased significantly.

The domestic value-added first exported, then returned home and consumed in the country of origin (RDV_G) represented a negligible share in all the Baltic States. For example, in Latvia in 2014, RDV_G accounted for only 0.2% of the country's total exports. In Estonia and Latvia, the value of the indicator was even lower. A low value indicates that domestic production after processing almost does not return to the country of origin for final consumption. As discussed, the foreign value-added in the country's exports (FVA) increased during the investigation period and was an essential component of the growth of the GVCs. It should be noted that the bulk of the FVA was third countries' value-added (OVA) rather than the direct importer's value-added (MVA) in the exports of the Baltic States.

One of the most interesting indicators obtained during the gross export decomposition is the double-counted value-added (PDC). One essential condition for this phenomenon is that production must cross national borders several times by moving out and in. As this indicator is not considered in classical trade statistics, export volumes are overestimated. The high PDC values in the Baltic States indicate the increasing cross-border production. In Estonia and Latvia, similar growth in PDC was recorded, respectively, by 2.9 and 2.6 pp, while the largest component growth was recorded in Lithuania—5.6 pp. In 2014, the PDC indicator was 12.7% in Estonia, 10.8% in Lithuania, and 9.1% in Latvia. The results indicate the increasing cross-border production, especially in Lithuania.

There were several similarities in the export structure of all Baltic States during the investigation period. The most significant differences were observed in the gross exports' ratios in domestic and foreign value-added. To obtain a more detailed picture of the similarities and differences between countries, the following part of the study is devoted to assessing individual economic sectors. The five most exported economic sectors were selected for this purpose based on 2014 data:

- C16—manufacture of wood and products of wood and cork, except furniture; manufacture of articles of straw and plaiting materials. It is the largest sector in Latvia with USD 1.6 billion and the second largest sector in Estonia with USD 1.7 billion;
- C19—manufacture of coke and refined petroleum products. It is the largest sector in Lithuania with USD 5.9 billion;
- C26—manufacture of computer, electronic and optical products. It is the largest sector in Estonia with USD 2.4 billion;
- G46—wholesale trade, except for motor vehicles and motorcycles. It is the second-largest sector in Lithuania with USD 3.6 billion;
- H49—land transport and pipeline transport. It is the second-largest sector in Latvia, with USD 1.6 billion.

In sector C16, there is a clear dominance of intermediate production in the gross exports. In all countries, this rate exceeded 90%. Exports from this sector are dominated by domestic value-added. The average DVA_G indicator in the Baltic countries was 66.2% in 2014. It should be noted that this indicator in sector C16 was still decreasing, especially in Lithuania and Latvia, where a decrease of around 10 pp. was recorded during the investigation period. It should be noted that most of the domestic value-added was observed in intermediate exports. Indirect exports fluctuated around 30% throughout the period considered, resulting in high forward participation in the GVC. Baltic States specialise in sector C16 at an early stage of the production process. One-third of the vertical specialisation is also explained by the PDC indicator, which refers to GVC and frequent cross-border movements of goods.

In the C19 sector, more than two-thirds of the gross exports in all countries were made up of intermediate products. At the time, the value-added components differed significantly among the Baltic States. In Lithuania, the vertical specialisation rate (FVA) in 2014 was 75.2%, twice as high as in the other countries surveyed, and increased significantly over the investigation period. In Estonia, this period is marked by a significant decline in vertical specialisation, expressed mainly as a decrease in the share of foreign value-added in intermediate production exports. At the same time, a very high level of participation in GVC was also observed in C19 in Lithuania, reaching 58.4% in 2014 (46.6 and 46.0% in Estonia and Latvia), and position in GVC was closer to the final consumer. The detailed decomposition of the DVA_G revealed that most of the domestic value-added was observed in intermediate exports. Sector C19 in Lithuania can be described as an economic sector with a high degree of participation and specialisation in the final production stages of GVC.

The C26 sector was characterised by similar export volumes of intermediate and final production in Estonia. However, in Lithuania and Latvia, the share of final production in exports during the investigation period increased around twice, reaching 71.0% and 86.4%, respectively. Apparent differences between countries' export structures are also observed in the origin of the export value-added. Foreign value-added in Estonia's gross exports was 56.2% in 2014, and only 21.4% was domestic. When analysing the structure of domestic value-added, it was observed that all its components decreased during the investigation period and such decrease was offset by growing foreign value-added.

Interestingly, the PDC indicator has doubled since 2000, indicating increasing GVCs. The overall conclusion is that all countries are close to the end of the GVCs although there are differences between countries. The detailed indicators also suggest that Lithuania and Latvia specialise only in certain small production stages that require particular domestic value-added when Estonia is more likely deeply involved in the entire production process.

In the G46 sector, there are apparent differences between the value-added components in the gross export between Lithuania and the other Baltic countries. In Lithuania, 43.1% of exports were final production, whereas the value of intermediary services was dominated in Latvia and Estonia. In the same way, differences are also visible in domestic and foreign value-added structures. DVA_G was as high as

92.3% in Lithuania and was largely due to domestic value-added in the final output, while other countries dominated domestic value-added in intermediate production. In Lithuania, the lowest participation rate in GVCs was observed in this sector. The results may indicate that Lithuania is involved in shorter GVCs and is more focused on providing the final services.

The H49 sector has the most similar structure among all the Baltic countries. The value of exports of intermediary services fluctuates between 66 and 67% in all three countries. However, differences can be observed in domestic and foreign value-added shares in the gross exports. In the case of Lithuania, 83.2% of total sector exports are explained by domestic value-added, while in Estonia and Latvia these components have a much smaller weight, 66.3%, and 69.4%, respectively. Both DVA_G and FVA indicators are very similar, but participation and position in GVCS differ. In Lithuania, the sector's participation in GVC is the lowest due to the lowest share of foreign value-added in the country's exports, which at the same time indicates a higher position at the beginning of the GVCs, while Estonia and Latvia have a higher share of foreign value-added closer to the final stages of the GVCs. Accordingly, those countries show a higher degree of involvement, particularly in exporting foreign value-added.

5 Conclusions

With the expansion of global trade databases, a methodology has been developed in scientific literature to break down countries' total exports into detailed value-added components with different economic meanings and calculate indicators of participation and position in GVCs, both at the country and sector levels. Nevertheless, the number of studies on the participation of the Baltic States in the GVC is limited.

The overall export structure of the Baltic States has many similarities in terms of value-added components. Domestic value-added is dominated by all countries, although in Estonia, at the end of the investigation period, it represented a much smaller share compared to the other Baltic countries. The domestic value-added structure is also similar. Its lowest share consists of a gross export component, defined as domestic value-added, used by the direct importer to produce exports to third countries. This indicator is directly used in calculating participation in the GVC. Production produced in the Baltic countries is very rarely returned to the country of origin. This is evidenced by the negligible share of the value-added first exported and then returned home for final consumption, which was less than half a percent in the Baltic States during the period under investigation.

As expected, the increase in foreign value-added in gross exports of the Baltic countries was observed during the period analyzed. In part, the latter was composed not of the importer's but of third countries' value-added in exporting country's exports. At that time, the double-counted value-added indicator revealed the increasing cross-border production share and lengthening GVC in the Baltic States. While the share of foreign value-added in final production exports decreased by more than a quarter, it increased in intermediate production. Double-counted value-added has also increased. This indicator refers to growing GVCs and increasing cross-border production processes in the Baltic States. Although the results show a more significant shift of Lithuania to higher value-added activities, the structure of vertical specialisation in all the Baltic States became very similar in 2014.

Finally, the analysis of the value-added components of the gross exports revealed significant differences between the economic sectors of the Baltic States. The wood and products of wood and cork manufacturing sector is dominated by exports of intermediate production in all Baltic States. The countries specialise in the initial production stage in this sector. A high proportion of double-counted value-added indicates long GVC and frequent movements of goods between countries. The computer, electronic and optical products manufacturing sector is Estonia's largest export sector, based mainly on foreign value-added, indicating the lower of the country's role in the production process. The land transport and pipeline transport sector are largely structurally similar in all countries. However, in Lithuania, exports of this sector are dominated by domestic value-added, leading to a lower level of participation in the GVC and the position at the beginning of the GVC compared to neighbouring countries.

Appendix: Decomposition of Gross Exports of Baltic States into Value-Added Components in 2000 and 2014

	EST				LTU				LVA			
	2000	%	2014	%	2000	%	2014	%	2000	%	2014	%
EXP	2,0	*100,0*	18,3	*100,0*	3,2	*100,0*	32,7	*100,0*	2,0	*100,0*	14,7	*100,0*
EXP_INT	1,4	*67,3*	12,6	*68,8*	1,8	*58,3*	21,1	*64,4*	1,5	*71,5*	10,1	*68,7*
EXP_FIN	0,7	*32,7*	5,7	*31,2*	1,3	*41,7*	11,6	*35,6*	0,6	*28,5*	4,6	*31,3*
DVA_FIN	0,4	*20,3*	3,0	*16,4*	1,0	*32,3*	8,2	*25,0*	0,4	*21,0*	3,2	*21,6*
DVA_INT	0,5	*26,0*	4,4	*24,2*	0,9	*27,9*	7,9	*24,1*	0,7	*34,5*	4,2	*28,3*
DVA_INTrex	0,4	*18,1*	2,9	*15,8*	0,5	*16,7*	4,9	*15,1*	0,4	*20,5*	2,8	*18,9*
DVA_INTrex1	0,2	*9,1*	1,5	*8,4*	0,2	*7,7*	2,6	*7,9*	0,2	*10,5*	1,5 ara>	*10,1*
DVA_INTrex2	0,1	*6,9*	1,0	*5,2*	0,2	*7,0*	1,7	*5,2*	0,2	*7,5*	0,9	*6,3*
DVA_INTrex3	0,0	*2,1*	0,4	*2,1*	0,1	*1,9*	0,7	*2,1*	0,1	*2,5*	0,4	*2,5*
RDV_INT (RDV_G)	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*	0,0	*0,1*
RDV_FIN	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*	0,0	*0,1*	0,0	*0,0*	0,0	*0,1*
RDV_FIN2	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*
PDC	0,2	*9,8*	2,3	*12,7*	0,2	*5,2*	3,5	*10,8*	0,1	*6,5*	1,3	*9,1*
DDC	0,0	*0,0*	0,0	*0,1*	0,0	*0,0*	0,0	*0,1*	0,0	*0,0*	0,0	*0,1*

(continued)

(continued)

	EST				LTU				LVA			
DDC_FIN	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*
DDC_INT	0,0	*0,0*	0,0	*0,1*	0,0	*0,0*	0,0	*0,0*	0,0	*0,0*	0,0	*0,1*
FDC	0,2	*9,8*	2,3	*12,6*	0,2	*5,2*	3,5	*10,8*	0,1	*6,5*	1,3	*9,0*
ODC	0,2	*8,7*	2,1	*11,7*	0,2	*4,8*	3,3	*10,0*	0,1	*5,9*	1,2	*8,4*
MDC	0,0	*1,1*	0,2	*0,9*	0,0	*0,3*	0,3	*0,8*	0,0	*0,6*	0,1	*0,6*
FVA	0,5	*25,7*	5,6	*30,8*	0,6	*17,9*	8,1	*24,9*	0,4	*17,4*	3,2	*21,9*
FVA_FIN	0,3	*12,4*	2,7	*14,8*	0,3	*9,4*	3,5	*10,6*	0,2	*7,4*	1,4	*9,7*
MVA_FIN	0,0	*1,5*	0,2	*1,0*	0,0	*0,8*	0,4	*1,1*	0,0	*0,9*	0,1	*0,8*
OVA_FIN	0,2	*10,9*	2,5	*13,7*	0,3	*8,6*	3,1	*9,5*	0,1	*6,5*	1,3	*8,8*
FVA_INT	0,3	*13,3*	2,9	*16,0*	0,3	*8,4*	4,7	*14,3*	0,2	*9,9*	1,8	*12,2*
OVA_INT	0,2	*11,8*	2,7	*14,9*	0,2	*7,9*	4,2	*12,8*	0,2	*8,9*	1,6	*11,1*
MVA_INT	0,0	*1,5*	0,2	*1,1*	0,0	*0,6*	0,5	*1,5*	0,0	*1,1*	0,2	*1,1*
OVA	0,5	*22,7*	5,2	*28,6*	0,5	*16,5*	7,3	*22,3*	0,3	*15,4*	2,9	*19,9*
MVA	0,1	*3,0*	0,4	*2,2*	0,0	*1,4*	0,9	*2,6*	0,0	*2,0*	0,3	*1,9*
DVA_G	1,3	*64,5*	10,3	*56,5*	2,4	*76,9*	21,0	*64,3*	1,6	*76,1*	10,2	*69,0*
VAX_G	1,3	*64,4*	10,3	*56,5*	2,4	*76,9*	21,0	*64,2*	1,6	*76,0*	10,1	*68,8*

References

1. Antràs, P., Chor, D.: Global Value Chains. Working Paper 28549 (2021). http://www.nber.org/papers/w28549
2. Banh, H., Wingender, P., Gueye, C.: Global Value Chains and Productivity: Micro Evidence from Estonia. IMF Working Papers **20**(117) (2020). https://doi.org/10.5089/9781513542300.001
3. Borin, A., Mancini, M.: Measuring what matters in global value chains and value-added trade. Policy Research Working Paper 8804 (2019). https://doi.org/10.1596/1813-9450-8804
4. Boschma, R.: Global value chains from an evolutionary economic geography perspective: a research agenda. Area Dev. Policy **1–24**,(2022). https://doi.org/10.1080/23792949.2022.2040371
5. Castelo-Branco, I., Oliveira, T., Simões-Coelho, P., Portugal, J., Filipe, I.: Measuring the fourth industrial revolution through the Industry 4.0 lens: the relevance of resources, capabilities and the value chain. Comput. Ind. **138**, 103639 (2022). https://doi.org/10.1016/j.compind.2022.103639
6. Chen, W., Los, B., McCann, P., Ortega-Argilés, R., Thissen, M., van Oort, F.: The continental divide? economic exposure to Brexit in regions and countries on both sides of the channel. Pap. Reg. Sci. **97**(1), 25–54 (2018). https://doi.org/10.1111/pirs.12334
7. Cieslik, E., Bieganska, J., Sroda-Murawska, S.: The intensification of foreign trade in post-socialist countries and their role in global value chains. Acta Oecon. **66**, 465–487 (2016). https://doi.org/10.1556/032.2016.66.3.5
8. Daudin, G., Rifflart, C., Schweisguth, D.: Who produces for whom in the world economy? Can. J. Econ. **44**(4), 1403–1437 (2011). https://doi.org/10.1111/j.1540-5982.2011.01679.x
9. Dzemydaite, G. (2017). External influences for balance of trade in small and open economies. *Journal of Applied Economic Sciences (JAES), 12*(48), 402–406.

10. Dzemydaitė, G.: The impact of economic specialization on regional economic development in the European union: insights for formation of smart specialization strategy. Economies **9**(2), 76 (2021). https://doi.org/10.3390/economies9020076
11. Dzemydaitė, G.: Agriculture's impact for the economy: inter-industry linkages and multiplier effects. In: International Scientific Conference RURAL DEVELOPMENT 2017, pp. 1004–1009 (2017). https://doi.org/10.15544/RD.2017.057
12. Dzemydienė, D., Dzemydaitė, G., Gopisetti, D.: Application of multicriteria decision aid for evaluation of ICT usage in business. Cent. Eur. J. Oper. Res. **30**, 323–343 (2022). https://doi.org/10.1007/s10100-020-00691-9
13. Fernandes, A.M., Kee, H.L., Winkler, D.: Determinants of global value chain participation: cross-country evidence. World Bank Econ. Rev. **36**(2), 329–360 (2022). https://doi.org/10.1093/WBER/LHAB017
14. Hagemejer, J.: Trade and growth in the new member states: the role of global value chains. Emerg. Mark. Financ. Trade **54**(35) (2016). https://doi.org/10.1080/1540496X.2017.1369878
15. Hagemejer, J., Ghodsi, M.: Up or down the value chain? a comparative analysis of the GVC position of the economies of the new EU member states. Cent. Eur. Econ. J. **1**(48) (2017). https://doi.org/10.1515/ceej-2017-0003
16. Hummels, D., Ishii, J., Yi, K.M.: The nature growth of vertical specialization in world trade. J. Int. Econ. **54**(1), 75–96 (2001). https://doi.org/10.1016/S0022-1996(00)00093-3
17. Hummels, D., Rapoport, D., Yi, K.M.: Vertical specialization and the changing nature of world trade. Econ. Policy Rev. **4**(2) (1998). https://core.ac.uk/download/pdf/6792678.pdf
18. Johnson, R.C., Noguera, G.: Accounting for intermediates: production sharing and trade in value added. J. Int. Econ. **86**(2), 224–236 (2012). https://doi.org/10.1016/j.jinteco.2011.10.003
19. Jones, L., Demirkaya, M., Bethmann, E.: Global value chain analysis: concepts and approaches. J. Int. Commer. Econ. (2019). https://www.usitc.gov/publications/332/journals/concepts_approaches_in_gvc_research_final_april_18.pdf
20. Kersan-Škabic, I.: Assessment of EU member states' positions in global value chains. East. J. Eur. Stud. **8**(2) (2017a). https://ejes.uaic.ro/articles/EJES2017_0802_KER.pdf
21. Kersan-Škabic, I.: Trade in Value Added (TiVA) in EU new member states (EU NMS). Croat. Econ. Surv. **19**(2) (2017b). https://doi.org/10.15179/ces.19.2.4
22. Koopman, R., Wang, Z., Wei, S.: Tracing value-added and double counting in gross exports. Am. Econ. Rev. **104**(2), 459–494 (2014). https://doi.org/10.1257/aer.104.2.459
23. Koopman, R., Powers, W., Wang, Z., Wei, S.: Give credit where credit is due: tracing value-added in global production chains. NBER Working Paper 16426 (2010). https://doi.org/10.3386/w16426
24. Koopman, R., Wang, Z., Wei, S.: How much of chinese exports is really made in China? assessing domestic value-added when processing trade is pervasive. NBER Working Paper 14109 (2008). https://doi.org/10.3386/w14109
25. Kordalska, A., Olczyk, M.: Global competitiveness and economic growth: a one-way or two-way relationship? Equilib. Q. J. Econ. Econ. Policy **11**(1), 121–142 (2016). https://doi.org/10.12775/EQUIL.2016.006
26. Leontief, W., Strout, A.: Multiregional input-output analysis. In: Miller, R.E., Blair, P.D. (eds.) (2009). Input-Output Analysis: Foundations and Extensions. Cambridge University Press (1963)
27. Matto, A., Wang, Z., Wei, A.J.: Measuring trade in value-added when production is fragmented across countries: an overview. In: Trade in Value Added: Developing New Measures of Cross-Border Trade. Centre for Economic Policy Research and the World Bank, London (2013). https://openknowledge.worldbank.org/handle/10986/15809
28. Miller, R.E., Blair, P.D.: Input-Output Analysis: Foundations and Extensions. Cambridge University Press (2009)
29. Núñez-Merino, M., Maqueira-Marín, J.M., Moyano-Fuentes, J., Castaño-Moraga, C.A.: Industry 4.0 and supply chain: a systematic science mapping analysis. Technol. Forecast. Soc. Chang. **181**, 121788 (2022). https://doi.org/10.1016/j.techfore.2022.121788

30. Olczyk, M., Kordalska, A.: Gross exports versus value-added exports: determinants and policy implications for manufacturing sectors in selected cee countries. Working Paper Series A. 10. 1–27 (2016). https://www.researchgate.net/publication/312056440_gross_exports_versus_value_added_exports_determinants_and_policy_implications_for_manufacturing_sectors_in_selected_cee_countries
31. Oosterhaven, J.: Interregional Input-Output Analysis and Dutch Regional Policy Problems. Aldershot, Gower (1981)
32. Polenske, K.R.: An empirical test of interregional input–output models: estimate of 1963 Japanese production. Am. Econ. Rev. **60**, 76–82 (1970)
33. Polenske, K.R.: The US Multiregional Input-Output Accounts and Model. Lexington Books, Lexington, MA (1980)
34. Polenske, K.R.: Leontief's spatial economic analyses. Struct. Chang. Econ. Dyn. **6**, 309–318 (1995)
35. Šidlauskaitė, B., Miškinis, A.: The development of economic structure and interindustry linkages in the Baltic countries. Ekonomika **92**(2), 32–48 (2013). https://doi.org/10.15388/Ekon.2013.0.1416
36. Šidlauskaitė-Riazanova, B. ir Miškinis, A. (2019). Aspects of the Development of Lithuanian Economic Specialisation in the Context of Globalization. *Socialiniai tyrimai*, 42(2), 59–73. https://doi.org/10.21277/st.v42i2.273
37. Song, Y., Hao, X., Hu, Y., Lu, Z.: The Impact of the COVID-19 pandemic on china's manufacturing sector: a global value chain perspective. Front. Public Health **9**, 509 (2021). https://doi.org/10.3389/fpubh.2021.683821
38. Timmer, M.P., Dietzenbacher, E., Los, B., Stehrer, R., Vries, G.J.: An illustrated user guide to the world input-output database: the case of global automotive production. Rev. Int. Econ. **23**(3), 575–605 (2015). https://doi.org/10.1111/roie.12178
39. Van der Linden, J.A.: Interdependence and specialisation in the European Union: intercountry input-output analysis and economic integration. Theses on Systems, organizations and management. University of Groningen (2000).
40. Wang, Z., Wei, S.J., Zhu, K.: Quantifying international production sharing at the bilateral and sector levels. Rev. Int. Econ. **23**(3), 575–605 (2013). https://doi.org/10.1111/roie.12178
41. Zhu, S., He, C.: What can evolutionary economic geography learn from global value chain and global production network research on developing and emerging economies? Area Dev. Policy **7**(2), 162–176 (2022). https://doi.org/10.1080/23792949.2022.2061542

The Soft Power of Understanding Social Media Dynamics: A Data-Driven Approach

Domnica Dzitac

Abstract Social media has become an increasingly used arena for political debates. Many political leaders around the world have misused this tool as a form of soft power to influence voters, spread fear or even destabilize democracies. In this paper, I discuss challenges, ethical considerations and moral dilemmas regarding the new era of a data driven society. Furthermore, I take a data science approach to understanding the dynamics of controversial political topics on Twitter, a widely used social media network, in the US context. To support my work, I collect an extensive dataset of 899,651 tweets that I analyze using state of the art Data Science and Natural Language Processing (NLP) techniques. I conduct an extensive analysis including labeling emotions of tweets and computing their attention score. I investigate which emotion gains the most engagement and spreads the most. The results suggest that anger and fear are the most prominent emotions. Moreover, fear has a significantly higher average attention score than other emotions.

Keywords Social media · Democracy · Soft power · Data science · BERT · NLP

1 Introduction

One of the biggest challenges of our age is navigating the social media dilemma. While it brought many benefits to the world, such as global interconnectedness, social media networks have caused a lot of societal damage in recent years [1]. Numerous social media networks have been associated with serious incidents, from negative impact on one's mental health to complicity in genocide incitation [1, 2]. Moreover, it is highly suggested that social media has shaped the course of action of multiple events in our recent history. For example, the Facebook-Cambridge Analytica data scandal, where non-consensually collected data of millions of Facebook users was

D. Dzitac (✉)
New York University Abu Dhabi, Abu Dhabi, UAE
e-mail: domnica.dzitac@nyu.edu

G. Dzemyda et al. (eds.), *Data Science in Applications*,
Studies in Computational Intelligence 1084,
https://doi.org/10.1007/978-3-031-24453-7_5

used in political campaigns, is a well-known case of altering political elections using social media platforms [3].

In 1990, Joseph Nye coined the term “soft power” to describe the ability of using attraction and persuasion in order to influence human behaviour [4]. As opposed to “hard power” (e.g., military force, economic sanctions, etc.), “soft power” is subtle and very often hard to recognize [4]. In this paper, I claim that social media is a form of soft power. Social media has become an increasingly used arena for political debates between people that very often have never met one another [5]. These online debates are very often less civil than face-to-face political conversations. It is thought that social media is one of the biggest contributing factors to political polarization as it exponentially spreads misinformation and boosts the growth of extreme ideologies [5]. Some researchers also argue that social media is an extreme threat to democracy [6]. We have seen cases in which many political leaders around the world have misused this tool as a form of soft power to influence voters, spread fear or even destabilize democracies [7]. For example, Facebook is considered at fault by legal authorities in the US and UK for facilitating the genocide of Rohingya Muslims in Myanmar after its algorithms boosted hate speech and neglected the removal of geocidal incitation content [6].

Considering all of the negative impact social media causes, as well as its benefits and indispensable nature in today’s society, many of us wonder: “What should we do about the social media issue?”. While we do not have an exact answer regarding this highly complex dilemma, it is clear that we can no longer accept the negative influence of social media on our lives and work tirelessly to use this phenomenal technology to flourish our well-being, support democracy and benefit from interconnectivity, without having to trade-off our privacy and safety. In this paper, I discuss the risks of social media to democracy and the need for a data science approach in informing the policy making process. While technology is developing exponentially, our regulations are years behind and our regulators are often overwhelmed by all the advancement. I argue that data-driven policies are crucial to minimizing the risks and maximizing the benefits of social media. As an example, I take a data science approach to understanding the dynamics of controversial political topics on Twitter, a widely used social media network, in the US context. The significance of understanding social media dynamics in political debates is crucial to making humane technology policies. Furthermore, I shed light on state of the art data science and Natural Language Processing (NLP) technologies that are extremely useful. This paper is an application of data science to understanding political dynamics on Twitter.

2 Motivation—Why Should We Care?

From an individuals’ perspective, the addiction to social media has a negative impact on the users’ mental and physical health [2]. From a global perspective, social media channels shape human interactions in a way that can amplify extremism, hate speech and political polarization [2, 6]. Recently, many entities, from government entities to NGOs and scholars, have shown concern regarding the negative impact of social media on our society. One of the pioneers in starting this discussion is the Center for Humane Technology (CHT), an NGO that supports the development of solutions that shift the addictive and toxic social media technologies to humane ones that benefit humanity, rather than generating a negative impact. Concerned entities, like CHT, fear the end of democracy if the current social media networks' approach to profit maximization at any cost is not regulated.

A popular paper from 2018 by Yascha Mounk and Roberto Foa titled The End of the Democratic Century addresses the issue of the decline of democracies [8]. The authors' claim is that a political system's stability and its expansion power are primarily dependent on the state's economic ascension under that form of governance. On this note, by gaining more economic power, autocratic states would have the tools to dominate, influence other states and thus end the democratic century. Even if I agree with many points from this article, I argue that social media is a huge defining factor in this discussion which is very often dismissed by scholars.

Mounk and Foa [8] suggest that the United States and its political system, namely liberal democracy, dominated the 20th century. This was mainly due to its economic power. In their article, the authors use economic measures such as GDP per capita to compare autocratic versus democratic states over time. They use this as evidence to support their claim. These results signal a threat for the western-coalition which lost its economic status worldwide and is in a continuous degradation. The authors admit that the economic factor is not the only one contributing to the decline of democracies and the rise in authoritarian soft power. However, they do not discuss these factors thoroughly, which in my opinion is crucial.

Even above the rise in economic power of autocratic states, social media in the hands of authoritarians is an assault on democracy. Countries running democratic elections are highly targeted by authoritarians whose main goal is to destabilize certain states [7]. Left in the hands of authoritarians or other stakeholders, social media platforms are relatively cheap methods of manipulating and polarizing an entire state. We have seen this happen in the Philippines, Myanmar or even in the United States [6].

Predicting the fall of a democratic century by comparing economic measures is indeed very intuitive, however it misses out on the social media component. Even though I agree with the authors on the fact that wealthier countries have more financial resources to influence other states, Mounk and Foa [8] do not mention the effect extreme polarization and fake news have on destabilizing world's democracies. With problematic financial focused business models and supercomputers pointed at users' brains, social media companies offered a new medium for authoritarians to manipulate others and even change voting behavior by simply paying for polarizing ads. It is also highly important to note that world's biggest authoritarian states such as China and Russia have banned most forms of western social media platforms, potentially because they acknowledged the threat these present to their political power. This is a sort of authoritarian soft power with an extreme capability of controlling masses without triggering their awareness.

Hence, the importance of regulating social media in a data-driven way is crucial in keeping democracy secure. It is important to showcase the relevance of understanding and observing how people behave in this online environment and regulate it wisely, as well as inform more humane technology methods. In this paper, I will walk the reader through a political dynamic analysis on Twitter in the US context. The relevance of this work is increasing as it can suggest evidence-based information that can shape policies.

3 Data

In order to better understand the political arena on Twitter, I collected tweets related to three of the most disputed topics in the US, namely vaccination, abortion and same-sex marriage. While there are many polarizing issues, I choose three issues that always come up in every political debate and are highly disputed between the left and the right wing followers. This allows me to narrow my analysis and offer a better understanding on each issue, rather than have a shallow overview. The dataset contains 899,651 tweets approximately equally balanced across the three topics (Table 1). The tweets were collected based on significant keywords. For example, for vaccination keywords such as “vaccine”, “vaccinate”, “anti-vax”, etc. were used to scrape relevant data. For abortion, keywords such as “pro life”, “pro choice”, “abortion”, etc. were used. For same-sex marriage, some examples are “gay marriage”, “same-sex marriage”, “love wins”, etc. The data was collected over a year’s period, from 20th of March 2021 to 20th of March 2022. The dataset includes the tweets’ text, user ID, the date, the number of likes, comments and retweets of each tweet, as well as other information that is not particularly used in this paper.

The data was collected using snscrape, a state of the art scraper for social media networks. This scraper is free of charge and it allows the user to scrape millions of tweets with no limitations in terms of quantity or time period. This tool completely changed the accessibility researchers have to getting social media data, specifically Twitter. Previous scrapers were limited in the amount of tweets one could scrape daily and often would limit historical data to a few weeks in the past. The snscrape tool changed the way researchers and other interested parties can gather data, significantly shortening the collection time.

Table 1 Number of tweets in each topic

Topic	Vaccination	Abortion	Same-sex marriage
Number of tweets	299,872	299,888	299,891

4 Methodology

Analyzing the sentiment or emotion of tweets is a very popular approach to understanding the public opinion. Previous related work analyzed emotions on social media in various ways. For example, Brady et al. [9] looked at emotion in the spread of moralized content on Twitter using principles of moral psychology. Xue et al. [10] classified tweets related to the COVID-19 pandemic using various emotion labels. In this paper, I look at which emotions gain the most attention and spread the fastest on Twitter, in the context of three of the most debated topics in the US, namely, vaccination, abortion and same-sex marriage.

Following the data collection, the tweets have been cleaned and pre-processed. This process involves replacing ULRs, user names and emojis from a tweet to placeholders such as [URL], [USER] and [EMOJI] (followed by the name of the emoji, e.g., happy face). I classified the tweets into one of the following six emotions: sadness, joy, love, anger, fear, and surprise. I decided to use a wider range of emotions rather than just positive, negative and neural labels which are too simplistic for the purpose of this work. For labeling the tweets, I used a T5-base fine-tuned model for emotion classification. T5 is based on transfer learning, where a model is first pre-trained with millions of data entries before being fine-tuned on a task that you actually want to solve [11]. Systems like BERT [12] or T5 [11] are state of the art technologies in NLP and they have changed the way researchers in the field operate. Instead of spending valuable time and resources training models with large datasets, one can now use these pre-trained systems to downstream to their desired task. Hugging Face' T5-base fine-tuned for Emotion Recognition performs very well with the mean f-score metric being approximately 0.9 with a standard deviation of approximately 0.06 across emotions. In Fig. 1 you can see examples of labeled tweets. In this work, I make use of these state of the art technologies to do emotion analysis, save valuable resources, time and generate reliable classifications. Moreover, it is important to mention that to handle such a large dataset, I have used data parallelism techniques to shorten the preprocessing and classification process. This means that the dataset was split into 4 equal batches and each batch was operating on a different core concurrently.

$$AttentionScore = No.Likes + No.Retweets + No.Comments \tag{1}$$

Besides classifying Tweets based on emotions, I have computed an attention score which is the combined number of likes, retweets and comments that each tweet received. I do that simply by summing up the mentioned metrics (1). I use this attention score to observe which type of emotions get most attention on Twitter regarding three of the most disputed topics in the US, namely vaccination, abortion and same-sex marriage. Moreover, I look at the combined dataset, including all three topics, and analyze the same metrics as mentioned before.

Emotion	Topic		
	Vaccination	Abortion	Same-Sex Marriage
Fear	"I reviewed Pfizer vaccine data again this morning; I am totally nauseated. To me, unless I am totally wrong, it is a scandal not being covered by the news. I can't believe the CDC, FDA, most doctors, Fauci (of course) have gone missing discussing these rather haunting data."	"Every Senate Democrat but one voted to legalize abortion through all nine months of pregnancy for any reason. Not sure how a political party advocating for the legalized slaughter of babies—including a 40 week, 7 lb squirming infant—could ever get the vote of a Christian."	"Wait I'm confused I thought the bill oposes gay marriage not pedophilia. Why are the Democrats claiming the bill is homophobic not pedophobic?"
Anger	"How dumb have we become? There really are people who think mask and vaccine mandates enacted for the purpose of saving citizens' lives are the equivalent of an unprovoked military invasion of a sovereign nation that will kill untold thousands and upset the post-WWII order?"	"NEW A bill in Missouri makes illegal to get an abortion if the patient has an ectopic pregnancy. Facts about ectopic pregnancies: - They're are not viable. Full stop. - They're the 1 cause of death for 1st trimester patients."	"Any answer to that question could result in the teacher being sued by a homophobic parent who doesn't like that someone mentioned same sex marriage in class, it just seems rife for abuse by people who don't like LGBTQ teachers. It'd be different if it wasn't targeted at a group."
Joy	"Especially when the COVID-19 vaccine shows rapid decline in efficacy among children, and many already have natural immunity after two years. Full release: https://t.co/ylktqwpnN"	"Wow it's very fun to have this connection with everyone on Earth! Dear everyone: be kind to each other, catch up on your health screening tests, take care of trans kids, and support your local abortion providers. xo"	"A children's book that reflects the reality that families are different, that's what we're talking about. The idea that representation is important; children notice when things aren't mentioned or forbidden. Next you'll tell me that "traditional marriage" isn't homophobic jargon"
Sadness	"I'm officially coming out. I'm vaccine injured. I'm sad, frustrated, and I'm angry. Nobody deserves this. I'm not anti vax, and I'm not a sheep. I'm Michelle."	"I feel like people who can't physically have an abortion, shouldn't be making laws about abortions."	"A gay man can live as his life repressed, living a unhappy marriage in a homophobic society"
Surprise	"BREAKING: According to documents received by a FOIA request to the Dept. of HHS, hundreds of news organizations were paid by the federal government to advertise vaccines as part of a "comprehensive media campaign." This is shocking."	"There's a new billboard in Murfreesboro that says "abortion is a right." I'm here for it and haven't gone to check out the website they linked, but I am surprised no one has set it on fire yet."	"i'm a lesbian and i'm shocked that some lgbtq+ people think if you're straight you're homophobic, so cringe"
Love	"We need more of this! Thank you for supporting vaccine choice."	"Abortion clinics are sacred, complicated places, and providers show up every day and hold it all. Today is Abortion Provider Appreciation Day, in honor of Dr. David Gunn, who was murdered in 1993. I #CelebrateAbortionProviders bc loving them out loud is part of keeping them safe."	"Love wins. Make same-sex marriage legal."

Fig. 1 Examples of tweets labeled with each emotion

5 Results

The results suggest that anger is the emotion that is the most predominant across topics, with the highest volume of posts (see Fig. 2). Looking at the combined dataset, including all three topics, posts labeled with the emotion "anger" make up 63.51% of the entire dataset. Tweets labeled as "anger" and "fear" combined make up 75.7% of the tweets posted across all three topics. The remaining are "joy" with 17.28%, "sadness" with 5.47%, and "love" and "surprise" with under 1% of the posts each. The amount of posts representing positive feelings, such as "love" and "joy", are insignificant compared to their negative counterparts. The overall result is consistent across individual topics (see Fig. 3). Posts related to same-sex marriage have the highest volume of angry posts, making up 85.14% of the whole topic's dataset.

The overall attention score was computed for each tweet. The combined average attention score for each emotion was calculated (Fig. 4) and a t-test for unequal variances with alpha = 0.05 was run. The t-test was used to determine statistical significant differences between groups of emotions, for example I looked to see if the average attention score statistically significant difference between the group of tweets labeled with joy, as opposed to fear. The results over the entire combined dataset show that the average attention score for tweets labeled with fear is higher

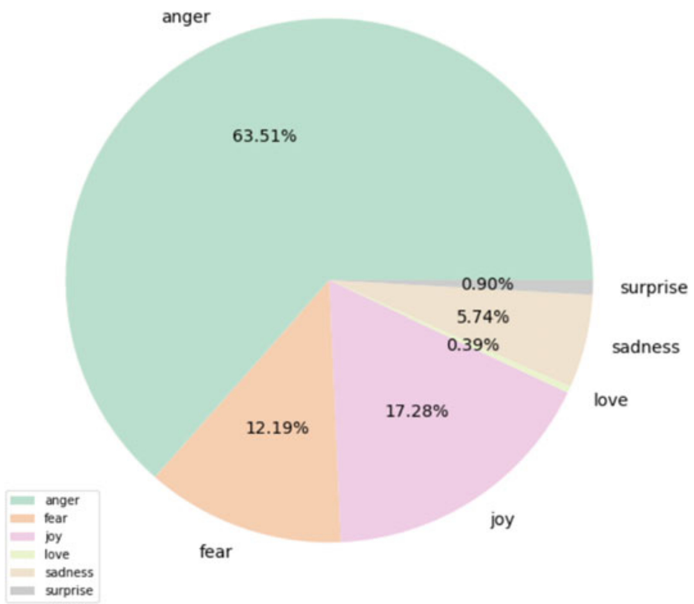


Fig. 2 Percentage of tweets labeled with each emotion in the whole combined dataset

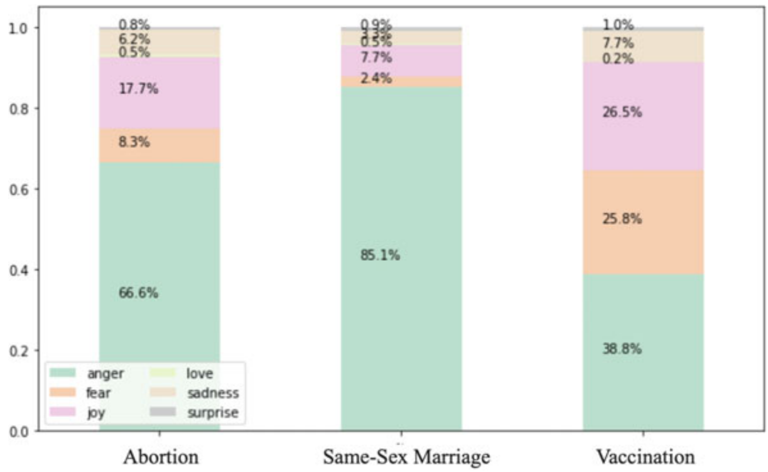


Fig. 3 Percentage of tweets labeled with each emotion for all three topics

than those labeled with anger, joy or sadness. Figure. 5 for individual p-values. No other statistically significant differences were identified in the combined dataset. However, these results vary across independent topics. Moreover, it is crucial to mention that the highest ranked tweets by attention score are fear and anger (see Fig. 6).

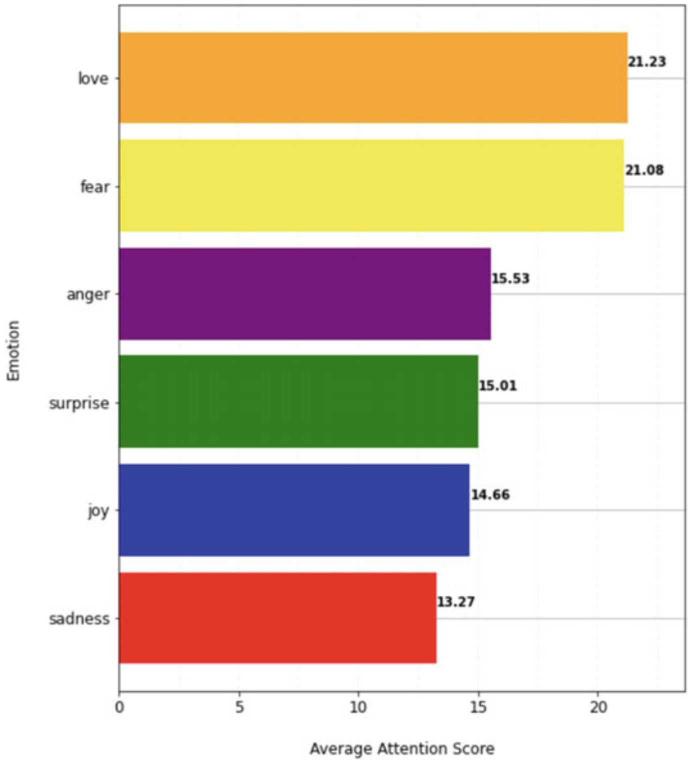


Fig. 4 Average attention score for each emotion in the whole combined dataset

p-values - Attention Score across topics						
	Fear	Anger	Joy	Sadness	Surprise	Love
Fear	1	**0.012***	**0.003***	**0.002***	0.216	0.924
Anger	**0.012***	1	0.220	0.127	0.900	0.333
Joy	**0.003***	0.220	1	0.350	0.937	0.265
Sadness	**0.002***	0.127	0.350	1	0.710	0.193
Surprise	0.216	0.900	0.937	0.710	1	0.384
Love	0.924	0.333	0.265	0.193	0.384	1

Fig. 5 P-values for attention scores in the whole combined dataset

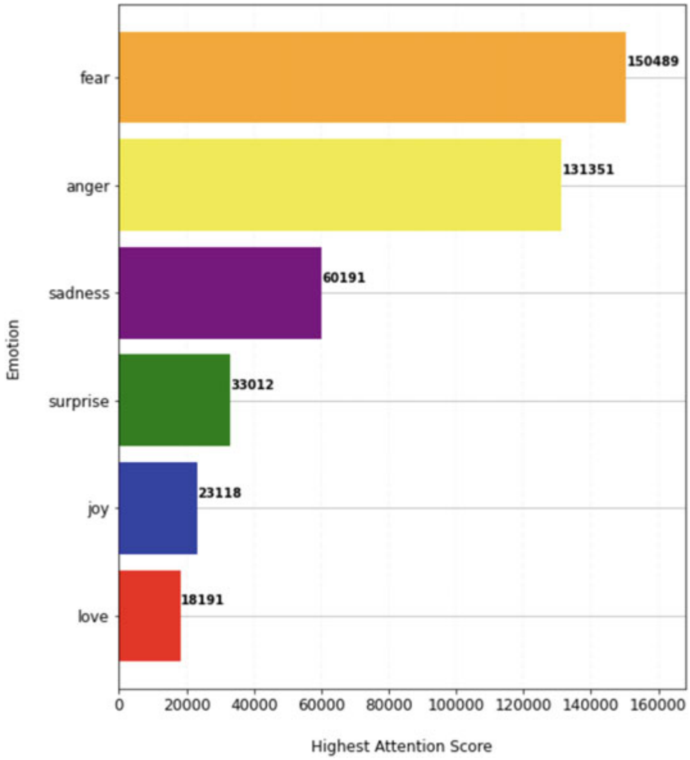


Fig. 6 Highest attention score for each emotion in the whole combined dataset

6 Discussion and Conclusion

Using techniques of computational intelligence for data analytics have proved extremely valuable. The obtained results offer an understanding of the dynamics of some controversial political topics on Twitter in the US context. A concerning high volume of tweets labeled with emotions such as anger and fear are posted in regards to three of the most disputed topics in the US, namely vaccination, abortion and same-sex marriage. Out of a combined dataset of 899,651 collected tweets, those labeled with emotions such as "anger" and "fear" combined make up 75.7% of the dataset. The widest spread emotion across topics is anger, with the highest tweet counts across all individual topics. These results help us understand how people express emotions regarding these controversial topics. It is suggested from this analysis that most people feel anger and fear, rather than joy and love. From the results, we can also see on average a significantly higher attention score for fear as opposed to other emotions. These results are consistent with related work [13].

The implications of these results are of extreme importance. My findings suggest that Twitter users are prone to emotions of anger and fear while discussing polarizing topics. Consequently, social media in the wrong hands can become a devastating threat to democracies by amplifying polarization and hate speech. This is a type of soft power with the capability of controlling masses without triggering their awareness. With very little resources, anyone can use social media as a tool to polarize and benefit from a space that is already charged with negative emotions. The world's biggest authoritarian states such as China and Russia have banned most forms of western social media platforms, potentially because they acknowledged the threat these present to their political power. While banning social media is not a viable solution for democratic states, these findings need to be acknowledged. Hence, the importance of regulating social media in a data-driven way is crucial in keeping democracy secure. It is important to showcase the relevance of understanding and observing how people behave in this online environment and regulate it wisely, as well as inform more humane technology methods.

All in all, these results suggest that Twitter shapes human interactions in a way that can amplify anger, hate speech and political polarization. From a policy-making perspective, it is important to understand these dynamics on Twitter and other social media channels, in order to regulate these networks effectively. The significance of understanding social media dynamics in political debates is crucial to making humane technology policies and protecting democracy. Moreover, on a personal level, it is crucial to understand that what we consume everyday on social media is affecting our mental health [2].

7 Further Developments

This paper includes an example of a data science application to understanding the political dynamics on Twitter. While it is informative, it can be improved. One of the limitations is that this paper focuses just on the US context, excluding other parts of the world. Moreover, the tweets were scraped based on only three of the most debatable topics, but there are many ways the collection could have occurred. A similar methodological approach can be applied to various topics. While Twitter is widely used for online political debates in the US, other social media networks, such as Facebook and Instagram are more known for causing harm. It would be interesting to analyze and explore other social media platforms. Perhaps, future work can compare various social media networks. Moreover, the paper could have been taken a step further by labeling also the comments given to each post, to see how people respond to different emotions.

Declarations

- Funding: No funding was needed for this project.
- Conflict of interest/Competing interests (check journal-specific guidelines for which heading to use): No conflict of interest.
- Ethics approval: Not applicable
- Consent to participate: Not applicable
- Consent for publication: Not applicable
- Availability of data and materials: Data is available upon request.
- Code availability: Code is available upon request.
- Authors' contributions: Not applicable

Appendix A: Supplementary information

Here are some additional plots. The dataset and code are available upon request (Figs. 5 and 6).

References

1. Siddiqui, S., Singh, T.: Social media its impact with positive and negative aspects. Int. J. Comput. Appl. Technol. Res. **5**(2), 71–75 (2016)
2. Amedie, J.: The impact of social media on society. Pop Cult. Intersect. (2) (2015)
3. Isaak, J., Hanna, M.J.: User data privacy: facebook, cambridge analytica, and privacy protection. Computer **51**(8), 56–59 (2018)
4. Nye, J.S.: Bound to Lead : The Changing Nature of American Power. Basic Book, New York (1990)
5. Bennett, W.L.: The personalization of politics: political identity, social media, and changing patterns of participation. Ann. Amer. Acad. Polit. Soc. Sci. **644**(1) (2012)
6. Whitten-Woodring, J., et al.: Poison if you don't know how to use it: facebook, democracy, and human rights in myanmar. Int. J. Press/Polit. **25**(3) (2020)
7. Aral, S., Eckles, D.: Protecting elections from social media manipulation. Science **365**, 6456 (2019)
8. Mounk, Y., Foa, R.S.: The end of the democratic century. Foreign Affairs (May/June 29-36) (2018)
9. Brady, W., Wills,J., Jost, J., Tucker, J., Van Bavel, J.: Emotion shapes the diffusion of moralized content in social networks. Proc. Natl. Acad. Sci. **114**, 7313–7318 (2017). https://doi.org/10.1073/pnas.1618923114
10. Xue, J., et al.: Twitter discussions and emotions about the covid-19 pandemic: machine learning approach. J. Med. Internet Res. **22**(11) (2020)
11. Raffel, C., et al.: Exploring the limits of transfer learning with a unified text-to-text transformer (2019). arXiv:1910.10683
12. Devlin, J., et al.: Pre-training of deep bidirectional transformers for language understanding (2018). arXiv:1810.04805
13. Wollebæk, D., et al.: Anger, fear, and echo chambers: the emotional basis for online behavior. Soc. Media+ Soc. **5**(2) (2019)

Bootstrapping Network Autoregressive Models for Testing Linearity

Mirko Armillotta, Konstantinos Fokianos, and Ioannis Krikidis

Abstract We develop methodology for network data with special attention to epidemic network spatio-temporal structures. We provide estimation methodology for linear network autoregressive models for both continuous and count multivariate time series. A study of non-linear models for inference under the assumption of known network structure is provided. We propose a family of test statistics for testing linearity of the imposed model. In particular, we compare empirically two bootstrap versions of a supremum-type quasi-score test. Synthetic data are employed to demonstrate the validity of the methodological results. Finally, an epidemic application of the proposed methodology to daily COVID-19 cases detected on province-level geographical network in Italy complements the work.

1 Modelling Network Time Series

New sources of data like social networks, GPS data, or epidemic counting processes, usually recorded over a timespan and a specific geographical area, has motivated a lot of interest in network data modelling. In particular, understanding the effect of a network to a multivariate time series is of essential importance for many applications and has attracted considerable recent attention. The methodology outlined in this work has potential application in several network science related fields.

Knight et al. [30] defined such multivariate streaming data as network time series and proposed a methodology for modelling them. This approach has been originally

M. Armillotta (✉) · K. Fokianos
Department of Mathematics and Statistics, University of Cyprus, P.O. Box 20537, 1678 Nicosia, Cyprus
e-mail: armillotta.mirko@ucy.ac.cy

K. Fokianos
e-mail: fokianos@ucy.ac.cy

I. Krikidis
Department of Electrical and Computer Engineering, University of Cyprus,
P.O. Box 20537, 1678 Nicosia, Cyprus

G. Dzemyda et al. (eds.), *Data Science in Applications*,
Studies in Computational Intelligence 1084,
https://doi.org/10.1007/978-3-031-24453-7_6

proposed in the context of spatio-temporal data analysis and is referred to Space-Time Autoregressive Moving Average (STARMA) models. See Cliff and Ord [9] and Martin and Oeppen [36], among many others. Indeed, a wide variety of available spatial streaming data related to physical phenomena fits this framework. In general, any stream of data for a sample of units whose relations can be modelled through an adjacency matrix (neighborhood structure), adhere to statistical techniques reviewed in this work.

We review some recent literature for network time series. Zhu et al. [54] developed inferential theory for Network Autoregressive models (NAR) when the network dimension N is increasing ($N \to \infty$), under the Independent Identic Distributed (IID) assumption on the innovation error sequence, where a continuous response random variable is observed for each node of a network. Technically speaking, in this approach the observed variable Y, for the node i at time t, is denoted by $Y_{i,t}$. To understand its behavior, as it evolves in time, it is assumed to depend on the past value of the variable for the node itself, say $Y_{i,t-1}$, and of the past values of the average variable between its neighbors, i.e. the mean of the variable Y, at time $t-1$ observed among the nodes connected to the node i. These authors develop Ordinary Least Squares (OLS) inference and study the asymptotic behaviour of the related estimator. Further extensions of network autoregressive models consider quantile autoregression [55], grouped least squares estimation [53], as well as a network extension for GARCH models [52]. The latter has been considered only for the case of fixed network dimension. Finally, Knight et al. [31] studied the more elaborate neighbourhood structures of STARMA models in the context of network analysis, named as Generalized NAR (GNAR), which considers the effect of several layers of connections between the nodes of the network and provide R software for fitting such models, for continuous variables only.

1.1 The Case of Discrete Responses

Interesting datasets collected from social network analysis have integer-valued nature, e.g. number of characters contained in users posts, number of likes, etc. However, the literature on models for multivariate count time series is sparse; see Fokianos [17] for a recent review. To fill this gap, Armillotta and Fokianos [3] proposed a linear and a log-linear Poisson network autoregression model (PNAR) for Poisson distributed data, under the assumption of α-mixing innovations. For details about and weak dependence related literature see Rosenblatt [43] and Doukhan [15]. This model generalizes the linear NAR model, by linking it with the context of Generalized Linear Models [37], since the observations are marginally Poisson distributed, conditionally to their past history. The joint dependence among different variables is specified by a copula construction, see Fokianos et al. [21, Sect. 2]. Armillotta and Fokianos [3] have further established parametric estimation under the framework of quasi maximum likelihood inference [26, 50] and associated asymptotic theory when the network dimension increases. Bracher and Held [6] study the related problem from a Bayesian point of view.

1.2 Nonlinear Models

All previous contributions assume linearity of the model, which is restrictive assumption in practice. Literature for univariate nonlinear time series models is well established; this is especially true for continuous-valued variables. The interested reader can see Tong [46], Fan and Yao [16], Gao [23] and Teräsvirta et al. [45], among many others, for more details. For integer-valued data there exists a more recent stream of works, although still under development. Suitable smoothing conditions for inference on nonlinear models are provided by Fokianos et al. [20], Neumann [38] with Poisson data, Christou and Fokianos [7] for the Negative Binomial case, and Gorgi [25] for the Beta Negative Binomial distribution. See also Wang et al. [47] for a threshold autoregressive model with Poisson distribution. In a more general framework, related works are by Ahmad and Francq [1], Davis and Liu [12] and Douc et al. [14], among others. For a recent review see Davis et al. [13]. Despite this flourishing literature related to nonlinear models, the previous works are not directly applicable to network autoregressive models, because of their multivariate structure. Multivariate models for discrete observations include the work by Pedeli and Karlis [40–42] and Fokianos et al. [21], among others, who consider linear models. Armillotta and Fokianos [4] specified a general nonlinear network autoregressive model for both continuous and discrete-valued processes, establishing also the related stationarity results and asymptotic theory of suitable quasi maximum likelihood estimators.

1.3 Testing for Linearity

Testing the linearity of a given model is a classical subject of study in time series analysis and econometrics. For continuous-valued random variables, general results have been reported when the parameters are identifiable or non-identifiable under the null hypothesis; see Boos [5] for the former and Francq et al. [22] for the latter case. Other linearity tests for specific nonlinear models and with non identifiable parameters, have been specified in Luukkonen et al. [35], for the Smooth Transition Autoregression (STAR) case, Li and Li [33], for the Threshold Autoregression (TAR) model, among others. For discrete-valued time series, Christou and Fokianos [8] suggest a score type test for univariate (mixed) Poisson random variables, in the case of correctly identifiable parameters. Finally, Andrews and Ploberger [2] and Hansen [28] proposed general methods for testing linearity under non-identifiability for univariate models. Non parametric tests have been also proposed; see, for example, Gao et al. [24] and Fokianos and Neumann [18], for continuous and count data, respectively. However, these latter test become computationally intensive when considering multivariate time series models. Armillotta and Fokianos [4] proposed testing procedures for examining linearity (or nonlinearity) of NAR models, for both continuous and count data, with and without the presence of non identifiable parameters under the null hypothesis.

1.4 Outline

The main aim of the work is to compare different bootstrap methods for testing linearity of NAR models. Such comparison will be conducted with the use of simulated synthetic data as well as by an application to real world data.

The paper is organized as follows: Sect. 2 introduces general nonlinear frameworks for network time series autoregressive models, for continuous and count processes and also discusses specific models of interest. Details about the inference to unknown parameters of the model are also provided. Then, in Sect. 3, results concerning the quasi-score test for testing linearity in network autoregressive models are discussed. The testing methodology is analyzed with and without non identifiable parameters under the null assumption. Practical computational aspects are taken into account, by describing different ways to compute the p-values of the proposed test statistics, by feasible bounds and bootstrap methodologies. Section 4 presents the results obtained on simulated data regarding the comparison between different computations of the linearity test. Finally, the proposed methodology is also applied to a real data analysis on epidemic networks to daily new COVID-19 cases observed on province-level geographical network in Italy.

Notation

For a $q \times p$ matrix $M = (m_{ij})$, $i = 1, \ldots, q$, $j = 1, \ldots, p$, denotes the generalized matrix norm $|||M|||_r = \max_{|x|_r=1} |Mx|_r$. If $r=1$, then $|||M|||_1 = \max_{1\le j\le p} \sum_{i=1}^{q} |m_{ij}|$. If $r = 2$, $|||M|||_2 = \rho^{1/2}(M'M)$, where $\rho(\cdot)$ is the spectral radius. If $r = \infty$, $|||M|||_\infty = \max_{1\le i\le q} \sum_{j=1}^{p} |m_{ij}|$. If $q = p$, these norms are matrix norms. The symbol I denotes an identity matrix, 1 a vector of ones, 0 a vector of zeros, whose dimensions depend on the context in which they are applied.

2 Network Autoregressive Models

When a network with N nodes, indexed by $i = 1, \ldots N$ is a priori known to the researcher, the neighbourhood structure of such a network is completely described by using its adjacency matrix $A = (a_{ij}) \in \mathbb{R}^{N\times N}$. The single element of such matrix would be $a_{ij} = 1$, if there is a directed edge from i to j (e.g. user i follows j on Twitter, a flight take off from airport i landing to airport j), and $a_{ij} = 0$ otherwise. Undirected graphs are allowed ($A = A'$), which means that the edge between two nodes, i and j, has no specific direction. Typically, self-relationships are excluded i.e. $a_{ii} = 0$ for any $i = 1, \ldots, N$. This is a restriction for many applications, such as social networks; see Wasserman et al. [49] and Kolaczyk and Csárdi [32], for more details on network definitions. Since the information on the network is assumed to be known in advance, the network structure is treated as a known component of the analysis. The row-normalised adjacency matrix is defined by $W = \text{diag}\{n_1, \ldots, n_N\}^{-1} A$ where $n_i = \sum_{j=1}^{N} a_{ij}$ is the total number of connections starting from the node i, such that $i \to j$; it is called out-degree. Then, W is constructed with the property $|||W|||_\infty = 1$. Moreover, define e_i the N-dimensional unit vector with 1 in the ith position and 0 everywhere else, such that $w_i = a_{ij}/n_i = (e_i' W)' = (w_{i1} \ldots, w_{iN})'$ is the vector containing the ith row of W.

Define a N-dimensional vector of time series $\{Y_t,\ t = 1, 2 \ldots, T\}$, where $Y_t = (Y_{1,t}, \ldots, Y_{i,t}, \ldots Y_{N,t})'$, which is observed on the given network; in this way, a univariate time series is detected for each node, say $Y_{i,t}$, with corresponding conditional expectation $\lambda_{i,t}$, denoted by $\{\lambda_t \equiv \text{E}(Y_t|\mathcal{F}_{t-1}),\ t = 1, 2 \ldots, T\}$, with $\lambda_t = (\lambda_{1,t}, \ldots, \lambda_{i,t}, \ldots, \lambda_{N,t})'$ being the conditional expectation vector, and denote the history of the process by $\mathcal{F}_t = \sigma(Y_s : s \le t)$. When the stochastic process $\{Y_t : t \in \mathbb{Z}\}$ is integer-valued, the first lag order nonlinear Poisson Network Autoregression (PNAR) is generally specified as follow [4]

$$Y_t = N_t(\lambda_t), \qquad \lambda_t = f(Y_{t-1}, W, \theta) \tag{1}$$

where $f(\cdot)$ is a function depending on the past lags of the count random vector, the known network structure W, and an m-dimensional parameter vector θ. The process $\{N_t\}$ is a sequence of N-variate copula-Poisson processes describing the joint dependence structure of the time series vector Y_t, where the marginal probability distribution of the count variables is $Y_{i,t}|\mathcal{F}_{t-1} \sim Poisson(\lambda_{i,t})$, for $i = 1, \ldots, N$. The joint distribution between univariate variables is generated by a copula structure, say $C(\cdot, \rho)$, on waiting times of a Poisson process, defined by Armillotta and Fokianos [3, Sect. 2.1]. An extension of (1) for a general lag order $p > 1$ models is given by, see Armillotta and Fokianos [4]

$$\lambda_t = f(Y_{t-1}, \ldots, Y_{t-p}, W, \theta)\,.$$

When the time series are continuous-valued, the nonlinear Network Autoregression (NAR) is defined by Armillotta and Fokianos [4] such that

$$Y_t = \lambda_t + \xi_t, \qquad \lambda_t = f(Y_{t-1}, W, \theta) \tag{2}$$

where $\xi_{i,t} \sim IID(0, \sigma^2)$, for $1 \le i \le N$ and $1 \le t \le T$. Obviously, we can extend (2) by incorporating a larger number of lags.

Models (1)–(2) have been proved to be stationary under suitable smoothness conditions on the function $f(\cdot)$ which are easily verifiable. See Armillotta and Fokianos [4, Sects. 2.2–2.3] for details about stability conditions.

Denote by $X_{i,t} = n_i^{-1} \sum_{j=1}^{N} a_{ij} Y_{j,t}$ the so called network effect; it represents the average impact of node i's connections. Recall models (1)–(2). The parameter vector can be split in two parts $\theta = (\theta^{(1)\prime}, \theta^{(2)\prime})'$, where the vectors $\theta^{(1)}$ and $\theta^{(2)}$ are of dimension m_1 and m_2, respectively, such that $m_1 + m_2 = m$. In general, $\theta^{(1)}$ denotes parameters associated with the linear part the model, whereas $\theta^{(2)}$ denotes the vector of nonlinear parameters. For $t = 1 \ldots, T$, both (1)–(2) have element-wise components

$$\lambda_{i,t} = f_i(X_{i,t-1}, Y_{i,t-1}; \theta^{(1)}, \theta^{(2)}), \quad i = 1, \ldots, N, \tag{3}$$

where $f_i(\cdot)$ is defined as the ith component of the function $f(\cdot)$, and it ultimately depends on the specific nonlinear model of interest which is taken into account.

2.1 Examples of Specific Models of Interest

We give some illustrative examples of specific nonlinear models of (3). We first introduce the linear model as a special case.

Linear Model

Recall that $X_{i,t} = n_i^{-1} \sum_{j=1}^{N} a_{ij} Y_{j,t}$ is the neighbourhood mean. The first order linear NAR(1) model,

$$\lambda_{i,t} = \beta_0 + \beta_1 X_{i,t-1} + \beta_2 Y_{i,t-1}, \tag{4}$$

is a special case of (3), with $\theta^{(1)} = (\beta_0, \beta_1, \beta_2)'$, but without nonlinear parameters $\theta^{(2)}$. For each single node i, model (4) allows the conditional mean of the process to depend on the past of the variable itself, for the same node i, and the average of the other nodes $j \neq i$ by which the focal node i is connected. Implicitly, only the nodes connected with the node i can affect its conditional mean $\lambda_{i,t}$. The parameter β_1 measures the impact of the network effect $X_{i,t-1}$. The coefficient β_2 determines the impact of the lagged variable $Y_{i,t-1}$. Model (4) was originally introduced by Knight et al. [30] and Zhu et al. [54] for the case of continuous random variables Y_t, with $Y_{i,t} = \lambda_{i,t} + \xi_{i,t}$. Armillotta and Fokianos [3] extended (4) to count random variables. In this case, (4) is the linear PNAR(1) model with $Y_{i,t} | \mathcal{F}_{t-1} \sim Poisson(\lambda_{i,t})$ for $i = 1, \ldots, N$ and a copula structure for joint distribution.

Intercept Drift (ID)

When Y_t is integer-valued, a drift in the intercept term of (4) introduces the nonlinear model

$$\lambda_{i,t} = \frac{\beta_0}{(1 + X_{i,t-1})^{\gamma}} + \beta_1 X_{i,t-1} + \beta_2 Y_{i,t-1}\,, \tag{5}$$

where $\gamma \geq 0$. Model (5) behaves like a linear model for small values of γ, and $\gamma = 0$ reduces (5) to (4) exactly. Instead, when γ takes values far from zero, model (5) introduce a perturbation, deviating from the linear model (4). Hence, (5) is a special case of (3), with $\theta^{(1)} = (\beta_0, \beta_1, \beta_2)'$ and $\theta_0^{(2)} = \gamma$. A slightly modified version of (5) allows to treat the case where $Y_t \in \mathbb{R}^N$, by taking the absolute value of $X_{i,t-1}$ defined at the denominator of the intercept term.

Smooth Transition (STNAR)

A Smooth Transition version of the NAR model, say STNAR(1), is specified as

$$\lambda_{i,t} = \beta_0 + (\beta_1 + \alpha \exp(-\gamma X_{i,t-1}^2))X_{i,t-1} + \beta_2 Y_{i,t-1}\,, \tag{6}$$

where $\gamma \geq 0$. This models introduces a smooth regime switching behaviour on the network effect, by mimicking the smooth transition time series models suggested by Haggan and Ozaki [27], Teräsvirta [44] and Fokianos and Tjøstheim [19]. When $\alpha = 0$ in (6), the linear NAR model (4) is recovered. Moreover, (6) is a special case of (3), with $\theta^{(1)} = (\beta_0, \beta_1, \beta_2)'$ and $\theta_0^{(2)} = (\alpha, \gamma)'$.

Threshold Effect (TNAR)

Another regime switching nonlinear time series model of particular interest is Threshold NAR model, TNAR(1), defined by

$$\lambda_{i,t} = \beta_0 + \beta_1 X_{i,t-1} + \beta_2 Y_{i,t-1} + (\alpha_0 + \alpha_1 X_{i,t-1} + \alpha_2 Y_{i,t-1}) I(X_{i,t-1} \leq \gamma)\,, \tag{7}$$

where $I(\cdot)$ is the indicator function and γ is the threshold parameter. Unlike the STNAR model, (7) induces an abrupt shift in the parameters of the models. For details about threshold-type models, the reader is referred to Lim and Tong [34], Wang et al. [47] and Christou and Fokianos [8], among others. When $\alpha_0 = \alpha_1 = \alpha_2 = 0$, model (7) reduces to the linear counterpart (4). Clearly, $\theta^{(1)} = (\beta_0, \beta_1, \beta_2)'$ and $\theta^{(2)} = (\alpha_0, \alpha_1, \alpha_2, \gamma)'$ show that (7) is a special case of (3).

2.2 Inference

Estimation for the true unknown parameter vector θ_0 in models (1) is developed by means of quasi-maximum likelihood methodology, see Wedderburn [50], Gourieroux et al. [26] and Heyde [29], for example. The Quasi Maximum Likelihood Estimator (QMLE) is the vector of parameters $\hat{\theta}$ maximizing the function

$$l_T(\theta) = \sum_{t=1}^{T} \sum_{i=1}^{N} \Big(Y_{i,t} \log \lambda_{i,t}(\theta) - \lambda_{i,t}(\theta) \Big) , \tag{8}$$

which is not necessarily the *true* log-likelihood of the process but it serves as an approximation. In particular, following Armillotta and Fokianos [4], (8) is the log-likelihood that it would have been obtained if all time series were contemporaneously independent. Note that although the joint copula structure $C(\dots, \rho)$ and the corresponding set of parameters ρ are not included in the maximization of (8), the QMLE is still computed under the assumption of dependence as it is implicitly taken into account in the past values of multivariate counts Y_t. Maximizing (8) simplifies computations of the estimation and guarantees consistency and asymptotic normality of the resulting estimator. The derivative of (8) yields the score function

$$S_T(\theta) = \sum_{t=1}^{T} \sum_{i=1}^{N} \left(\frac{Y_{i,t}}{\lambda_{i,t}(\theta)} - 1 \right) \frac{\partial \lambda_{i,t}(\theta)}{\partial \theta} \equiv \sum_{t=1}^{T} s_t(\theta) . \tag{9}$$

Define $\partial \lambda_t(\theta)/\partial \theta'$ the $N \times m$ matrix of derivatives, $D_t(\theta)$ the $N \times N$ diagonal matrix with elements equal to $\lambda_{i,t}(\theta)$, for $i = 1, \dots, N$ and $\xi_t(\theta) = Y_t - \lambda_t(\theta)$ is a Martingale Difference Sequence (MDS). Then, the empirical Hessian and conditional information matrices are given, respectively, by

$$H_T(\theta) = \sum_{t=1}^{T} \sum_{i=1}^{N} \frac{Y_{i,t}}{\lambda^2_{i,t}(\theta)} \frac{\partial \lambda_{i,t}(\theta)}{\partial \theta} \frac{\partial \lambda_{i,t}(\theta)}{\partial \theta'} - \sum_{t=1}^{T} \sum_{i=1}^{N} \left(\frac{Y_{i,t}}{\lambda_{i,t}(\theta)} - 1 \right) \frac{\partial^2 \lambda_{i,t}(\theta)}{\partial \theta \partial \theta'} ,$$

$$B_T(\theta) = \sum_{t=1}^{T} \frac{\partial \lambda'_t(\theta)}{\partial \theta} D_t^{-1}(\theta) \Sigma_t(\theta) D_t^{-1}(\theta) \frac{\partial \lambda_t(\theta)}{\partial \theta'} ,$$

where $\Sigma_t(\theta) = \mathrm{E}\left(\xi_t(\theta)\xi'_t(\theta) \mid \mathcal{F}_{t-1}\right)$ is the conditional covariance matrix evaluated at θ. Under suitable network assumptions and smoothness conditions on the nonlinear function $f(\cdot)$, Armillotta and Fokianos [4] proved the consistency and asymptotic normality of the estimator, that is $\sqrt{NT}(\hat{\theta} - \theta_0) \xrightarrow{d} N(0, H^{-1}BH^{-1})$, when $N \to \infty$ and $T_N \to \infty$, where H and B are the theoretical Hessian and information matrices, respectively, evaluated at the true value of the parameters $\theta = \theta_0$.

Analogous inferential result are obtained for model (2), by maximizing the quasi log-likelihood $l_{NT}(\theta) = -\sum_{t=1}^{T}(Y_t - \lambda_t(\theta))'(Y_t - \lambda_t(\theta))$, being equivalent to perform a nonlinear Least Squares (LS) estimation of the unknown parameters.

3 Linearity Test

In this section we introduce the linearity test for nonlinear networks autoregressive models (1)–(2), discussed in Sect. 2. Recall model (3) and consider the following hypothesis testing problems

$$H_0 : \theta^{(2)} = \theta_0^{(2)} \quad \text{versus} \quad H_1 : \theta^{(2)} \neq \theta_0^{(2)}, \quad \text{componentwise}\,, \tag{10}$$

where, under the null hypothesis H_0, the nonlinear parameters take a value $\theta_0^{(2)}$, which yields the linear model (4). For example, when Y_t is integer-valued following (1) and the mean process λ_t is defined as in (5), then $\theta^{(2)} = \gamma$ and $\theta_0^{(2)} = 0$. Indeed the problem $H_0 : \gamma = 0$ versus $H_1 : \gamma > 0$ becomes an hypothesis test between a linear null assumption versus ID alternative model.

To develop a test statistic for (10), we employ a quasi-score test based on the quasi-log-likelihood (8). This is a convenient choice, since such type of test requires only the estimation of model under the null hypothesis, which will be the linear model (4), say $\tilde{\theta} = (\tilde{\beta}_0, \tilde{\beta}_1, \tilde{\beta}_2)'$; this is usually a simpler task compared to the estimation of the nonlinear alternative model. Recall the partition of the parameters θ in (3), then $S_T(\theta) = (S_T^{(1)\prime}(\theta), S_T^{(2)\prime}(\theta))'$ denotes the corresponding partition of the quasi-score function (9). The quasi-score test statistic is given by

$$LM_T = S_T^{(2)\prime}(\tilde{\theta})\Sigma_T^{-1}(\tilde{\theta})S_T^{(2)}(\tilde{\theta})\,, \tag{11}$$

with $\Sigma_T(\tilde{\theta}) = JH_T^{-1}(\tilde{\theta})J'\left(JH_T^{-1}(\tilde{\theta})B_T(\tilde{\theta})H_T^{-1}(\tilde{\theta})J'\right)^{-1}JH_T^{-1}(\tilde{\theta})J'$, where $J = (O_{m_2 \times m_1}, I_{m_2})$, I_s is a $s \times s$ identity matrix and $O_{a \times b}$ is a $a \times b$ matrix of zeros. $\Sigma_T(\tilde{\theta})$ is a the estimator for the unknown covariance matrix $\Sigma = \mathrm{Var}[S_T^{(2)}(\tilde{\theta})]$. It can be proved that the quasi-score test (11) converges, asymptotically, to a $\chi^2_{m_2}$ distribution [4, Theorem 7]. Then, we reject H_0, if the value of LM_T computed in the available sample is greater than the critical values of the $\chi^2_{m_2}$ distribution, computed at ordinary significance levels. Analogous results hold for the continuous-valued model (2).

3.1 The Case of Non Identifiable Parameters

For model (3), consider the case where $f_i(\cdot)$ is defined as

$$\lambda_{i,t} = \beta_0 + \beta_1 X_{i,t-1} + \beta_2 Y_{i,t-1} + h_i(Y_{t-1}, \gamma)\alpha \,, \tag{12}$$

where $h_i(Y_{t-1}, \gamma)$ is a B-dimensional vector of nonlinear functions, say $h_i^b(Y_{t-1}, \gamma)$, with $b = 1, \ldots, B$, and α is the associated B-dimensional vector of nonlinear parameters. In practice, model (12) assumes that the nonlinear part of the network autoregressive models is of the form of an additive component. Note that the function $h_i(\cdot)$ depends on the lags of the variable and on k-dimensional vector of parameters γ. Several nonlinear models are included in (12). For example, the STNAR model (6), where $B = 1$ and $h_i(Y_{t-1}, \gamma) = \exp(-\gamma X_{i,t-1}^2)X_{i,t-1}$, for $i = 1, \ldots, N$, and the TNAR model (7), where $B = 3$ and $h_i^1(Y_{t-1}, \gamma) = I(X_{i,t-1} \leq \gamma)$, $h_i^2(Y_{t-1}, \gamma)) = X_{i,t-1} I(X_{i,t-1} \leq \gamma)$ and $h_i^3(Y_{t-1}, \gamma) = Y_{i,t-1} I(X_{i,t-1} \leq \gamma)$. Testing linearity on model (12) is equivalent to testing

$$H_0 : \alpha = 0 \,, \quad \text{versus} \quad H_1 : \alpha \neq 0 \,, \quad \text{elementwise}, \tag{13}$$

However, in this particular case, it is not possible to estimate the value of the parameter γ, because it is not identifiable under the null hypothesis H_0. Note that the parameter γ exists in the score partition function (9) because it is related to the nonlinear parameter $\theta^{(2)} = \alpha$. We conclude that the relevant quantities for inference and testing-see (11)—depend on γ, that is $S_T^{(2)}(\tilde{\theta}, \gamma)$, $\Sigma_T(\tilde{\theta}, \gamma)$ and $LM_T(\gamma)$. The model is then subject to non identifiable parameters γ under the null assumption. When this problem appears the standard theory does not apply and a chi-square type test is not suitable any more; see Davies [11] and Hansen [28], among several other references. Clearly, the value of the test changes over different values of $\gamma \in \Gamma$, where Γ is the domain of γ. A summary function of the test computed under different values of γ is then required; a typical choice is $g_T = \sup_{\gamma \in \Gamma} LM_T(\gamma)$. In practice, the space Γ is replaced by $\Gamma_F = (\gamma_L, \gamma_1, \ldots, \gamma_l, \gamma_U)$, a grid of values for the non identifiable parameters γ, and the maximum of the tests computed over such grid would be the test statistics employed for the evaluation of the test (13). Armillotta and Fokianos [4] established the convergence of the test g_T to g, when $T \to \infty$, being a function of a chi-square process, $LM(\gamma)$, in symbol $g = \sup_{\gamma \in \Gamma} LM(\gamma)$. The values of the latter asymptotic distribution cannot be tabulated, as this depends on unknown values of γ. We describe next methodology for computing the p-values of the sup-type test statistic.

3.2 Bootstrapping Test Statistics

Based on the previous arguments, we suggest to approximate the p-values of the test statistic by employing the following bootstrap algorithm

Algorithm 1 Score bootstrap

1: **for** $j = 1, \ldots, J$ **do**
2: Generate $\left\{\nu_{t,j} : t = 1, \ldots, T\right\} \sim IIDN(0, 1)$.
3: Compute $S_T^{\nu_j}(\tilde{\theta}, \gamma) = \sum_{t=1}^{T} s_t(\tilde{\theta}, \gamma)\nu_{t,j}$.
4: Compute the test $LM_T^{\nu_j}(\gamma)$, for $\gamma \in \Gamma_F$, and $g_T^j = \sup_{\gamma \in \Gamma_F} LM_T^{\nu_j}(\gamma)$.
5: **end for**
6: Compute $p_T^J = J^{-1} \sum_{j=1}^{J} I(g_T^j \geq g_T)$.

An approximation of the p-values is obtained from step 6 of Algorithm 1, where g_T is the value of the test statistic computed on the available sample. When the bootstrap replication J is big enough p_T^J is a good approximation of the unknown p-values of the test. Then, the null hypothesis H_0 is rejected if p_T^J is smaller than a given significance level. In order to test the robustness and performances of Algorithm 1, we propose here a comparison with an alternative parametric bootstrap procedure.

Algorithm 2 Parametric bootstrap

Estimate parameters of the linear model (4), $\tilde{\theta}$.
for $j = 1, \ldots, J$ **do**
By using $\tilde{\theta}$ at step 1, generate from (1), with $f(\cdot)$ defined as in (4), a bootstrap sample $\bar{Y}_t^j$, with $t = 1, \ldots, T$.
Compute $\bar{S}_T^j(\tilde{\theta}, \gamma)$ from (9), by using the observations generated at step 3.
Compute the test $\overline{LM}_T^j(\gamma)$, for $\gamma \in \Gamma_F$, and $\bar{g}_T^j = \sup_{\gamma \in \Gamma_F} \overline{LM}_T^j(\gamma)$.
end for
Compute $\bar{p}_T^J = J^{-1} \sum_{j=1}^{J} I(\bar{g}_T^j \geq g_T)$.

The bootstrap p-values are obtained from step 7 of Algorithm 2. The parametric bootstrap method differs from the former approach because the source of randomness in the bootstrap iterations is not a multiplicative Gaussian noise $\nu_{t,j}$ but a resampling process which generates new pseudo-observations from the estimated model. The same methods apply unaffected to the continuous-valued model (2). We omit the details. In the following section we compare the performances of the testing methods proposed so far.

4 Applications

In this part of the chapter we illustrate the described methodologies for testing linearity for network autoregressive models on a set of synthetic and real data.

4.1 Simulation Results

Synthetic data obtained by Monte Carlo simulation are considered in this section. A network structure is required in the application of NAR models. Moreover, recall that the structure of the network is completely described by its adjacency matrix $A = (a_{ij}) \in \mathbb{R}^{N \times N}$ with a_{ij} such that $a_{ij} = 1$, if there is a directed edge from i to j and 0 otherwise. In this simulation study such network is generated following one of the most popular network structure models, the Stochastic Block Model (SBM), see Nowicki and Snijders [39], Wang and Wong [48] and Zhao et al. [51]. A block label $(l = 1, \ldots, K)$ is assigned for each node with equal probability and K is the total number of blocks. Then, set $\mathrm{P}(a_{ij} = 1) = N^{-0.3}$ if i and j belong to the same block, and $\mathrm{P}(a_{ij} = 1) = N^{-1}$ otherwise. Practically, the model assumes that nodes within the same block are more likely to be connected with respect to nodes from different blocks. Throughout we assume the existence of two blocks ($K = 2$) and $N = 8$. The network is practically generated by using the `igraph` package of `R` software [10].

The observed count time series $\{Y_t : t = 1, \ldots, T = 1000\}$ is generated recursively as in (1), with λ_t coming from the linear model (4), using the copula-based data generating process of Armillotta and Fokianos [3, Sect. 2.1]. A choice of the copula function $C(\cdot)$ and the starting N-dimensional vector of the process λ_0 are required. The selected copula structure is Gaussian, $C_R^{Ga}(\ldots)$, with correlation matrix $R = \rho \bar{I}$, where $\bar{I}$ is a $N \times N$ matrix of ones; $\rho = 0.5$ is the copula parameter. Then $C_R^{Ga}(\ldots) = C^{Ga}(\ldots, \rho)$. We set $\lambda_0 = 1$ and use a burnout sample, by discarding the 300 first temporal observations to reduce the impact of the starting value of the process. The time series observations are obtained by setting the value of the linear parameters equal to $\theta^{(1)} = (\beta_0, \beta_1, \beta_2)' = (0.5, 0.2, 0.1)'$. This procedure is replicated $S = 200$ times. Then, the linear QMLE estimation $\tilde{\theta}$ optimising (8) is computed for each replication.

To generate the process Y_t in the continuous-valued case, the random errors $\xi_{i,t}$ are simulated from standard normal distribution $N(0, 1)$. For the data generating process of the vector Y_t, the initial value Y_0 is randomly simulated according to its stationary distribution [54, Proposition 1]. This is Gaussian with mean $\mu = \beta_0(1 - \beta_1 - \beta_2)^{-1}1$ and covariance matrix $\mathrm{vec}[\mathrm{Var}(Y_t)] = (I_{N^2} - G \otimes G)^{-1}\mathrm{vec}(I)$, where $1 = (1 \ldots 1)' \in \mathbb{R}^N$, I is the $N \times N$ identity matrix, $G = \beta_1 W + \beta_2 I$,$\otimes$ denotes the Kronecker product and $\mathrm{vec}(\cdot)$ the vec operator. Once the starting value Y_0 is given, the process $\{Y_t : t = 1, \ldots, T\}$ is generated recursively according to (4) and $Y_t = \lambda_t + \xi_t$, coming from (2). Then, the LS estimation of the linear parameters is computed for each replication. In this case, the resulting estimator is the ordinary least squares, which has closed form solution [54, Eq. 2.9].

We give here an example of a non standard case, by testing the linearity of model (4) versus the STNAR model; this is done by setting the hypothesis test $H_0 : \alpha = 0$ versus $H_1 : \alpha > 0$ in (6), inducing lack of identifiability on the parameter γ. According to Sect. 3.1, for each of the S replications, we can approximate the p-values of the sup-type test, $\sup_{\gamma \in \Gamma_F} LM_T(\gamma)$, where Γ_F is a grid of 10 equidistant values picked on $[0.01, 3]$, by the two bootstrap approximation procedures described in Sect. 3.2,

Table 1 Empirical size at nominal significance levels $\alpha_{H_0} = \{0.1, 0.05, 0.01\}$ of the test statistics (11) for testing $H_0 : \alpha = 0$ in $S = 200$ simulations of model (6), for $N = 8$, $T = 1000$. Data are integer-valued and generated from (1), with the linear model (4). The empirical power is also reported for data generated from model (6) with $\alpha = \{0.3, 0.4\}$ and $\gamma = \{0.1, 0.2\}$. The network is derived from the SBM. The approximated p-values are computed by score bootstrap (p_T^J), in the first row, and parametric bootstrap ($\bar{p}_T^J$), second row

Method	Size			Power								
				$\gamma = 0.2, \alpha = 0.3$			$\gamma = 0.1, \alpha = 0.4$			$\gamma = 0.2, \alpha = 0.4$		
	10%	5%	1%	10%	5%	1%	10%	5%	1%	10%	5%	1%
p_T^J	0.020	0.015	0.000	0.260	0.155	0.075	0.445	0.300	0.075	0.590	0.475	0.270
$\bar{p}_T^J$	0.020	0.010	0.000	0.265	0.180	0.060	0.510	0.300	0.085	0.590	0.510	0.275

Table 2 Empirical size at nominal significance levels $\alpha_{H_0} = \{0.1, 0.05, 0.01\}$ of the test statistics (11) for testing $H_0 : \alpha = 0$ in $S = 200$ simulations of model (6), for $N = 8$, $T = 1000$. Data are continuous-valued and generated from (2), with the linear model (4). The empirical power is also reported for data generated from model (6) with $\alpha = \{0.3, 0.4\}$ and $\gamma = \{0.1, 0.2\}$. The network is derived from the SBM. The approximated p-values are computed by score bootstrap (p_T^J), in the first row, and parametric bootstrap ($\bar{p}_T^J$), second row

Method	Size			Power								
				$\gamma = 0.2, \alpha = 0.3$			$\gamma = 0.1, \alpha = 0.4$			$\gamma = 0.2, \alpha = 0.4$		
	10%	5%	1%	10%	5%	1%	10%	5%	1%	10%	5%	1%
p_T^J	0.070	0.025	0.000	0.970	0.940	0.755	0.370	0.140	0.020	0.995	0.990	0.950
$\bar{p}_T^J$	0.070	0.020	0.000	0.970	0.905	0.720	0.275	0.105	0.005	0.990	0.985	0.915

with $J = 299$ bootstrap replications. The fraction of cases over S simulations in which the p-value approximations is smaller than the usual significance levels 0.1, 0.05 and 0.01 is the frequency of cases where H_0 is rejected and constitutes the empirical size of the test. The empirical power of the test is again the frequency of cases where H_0 is rejected but obtained when data were generated by the model (6) instead. This is accomplished by using the same generating mechanism described for the linear model, by setting various combinations of values of nonlinear parameters $\alpha = \{0.3, 0.4\}$ and $\gamma = \{0.1, 0.2\}$.

The results of the simulation study for the count data case are reported in Table 1. We note that the empirical size is smaller than or close to the expected nominal levels; the empirical power is low when α is small and tends to grow for larger values of α far from the value of the null assumption. The two bootstrap methods show similar behavior, but the parametric bootstrap yields slightly better when compared to the score based bootstrap. Such results show that both tests works satisfactorily with a slight preference given to the parametric bootstrap methodology.

Table 2 considers results regarding the continuous case. Firstly, we see an overall improvement of the performances compared with the integer-valued case. This is expected since here the errors ξ_t are generated from Normal random variables and also the stationary distribution of the process Y_t is Gaussian. Hence, the χ^2 (process) distribution of the test is approached more quickly. Instead in the integer-valued case such distribution will be reached only asymptotically, with $N \to \infty$, $T_N \to \infty$. The results of the two bootstrap procedures are again similar, but we note that the score bootstrap slightly outperforms the parametric one.

4.2 New COVID-19 Cases on Italian Provinces

We study a dataset which consists of daily new cases of COVID-19 virus detected for each province of Italy, according to the Nomenclature of Territorial Units for Statistics, Level 3 (NUTS-3) classification, as established on Regulation (EC) No 1059/2003 of the European Parliament and of the Council. Data is provided by the Presidenza del Consiglio dei Ministri—Dipartimento della Protezione Civile.[1] The total number of provinces is $N = 107$. The time series starts at 25/02/2020 and is updated daily until 07/02/2022 ($T = 714$). For the considered regions and time window, we observed two instances of negative numbers of new cases. These values are replaced by zero counts.

An undirected network structure can be derived by exploiting available data on geographical coordinates. The geodesic distance between the centroids of pairs of provinces $\{i, j\}$ are computed, say d_{ij}. Then, two provinces $\{i, j\}$ are connected with an undirected edge if $d_{ij} \leq 200$ km. We consider such cut-off reasonable by considering that a smaller distance would results in few connections for most remote regions, like the islands, whereas a bigger distance will result in a fully connected network, i.e. a network which connects each node to all the others, which is not of interest in the current analysis. The density of the network is 21.58%. The histogram of the number of connections is shown in Fig. 1. The maximum number of connections is 45. The median number of connections is 22.

We see from Fig. 2 a typical time series for each province. The data show that it is possible to detect at least two regimes of variation; one during pandemic seasonal waves, with high numbers of daily new cases and one where the virus cases are relatively stable for several months. We address the question that a linear model is suitable for fitting such data. The partial autocorrelation function (PACF) of the time series indicates a significant effect of the past counts so an autoregressive model may be adequate to model the dataset. The median number of daily new cases is 27.

Estimation of the linear PNAR model (4) is performed by QMLE. For testing linearity, the quasi-score linearity test is computed according to (11). For the identifiable case, the asymptotic chi-square test is employed, for the nonlinear model (5), testing $H_0 : \gamma = 0$ versus $H_1 : \gamma > 0$. For non identifiable case, we test linearity against the presence of smooth transition effects, as in (6), with $H_0 : \alpha = 0$ versus $H_1 : \alpha > 0$. A grid of 10 equidistant values in the interval $\Gamma_F \equiv [0.001, 3]$ is chosen for values of the nuisance parameter γ. The p-values are computed for the test

[1] Dataset available at https://github.com/pcm-dpc/COVID-19/blob/master/dati-province/dpc-covid19-ita-province.csv.

Fig. 1 Histograms of number of connections (degrees) between provinces of Italy

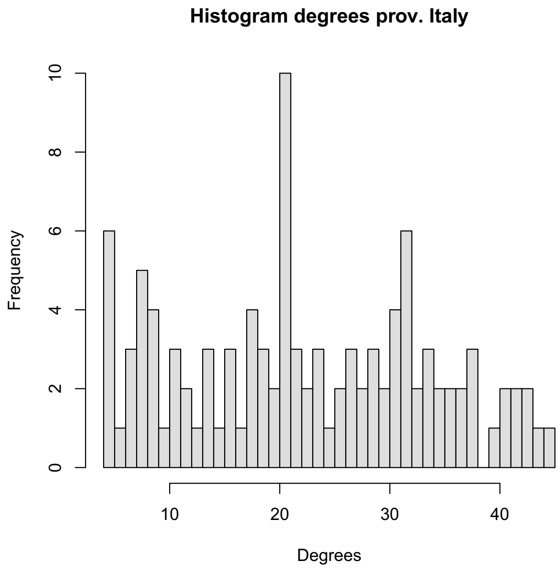


Fig. 2 Time series of counts and partial autocorrelation function for the number of daily new COVID-19 cases in Benevento province, Italy. Dashed blue line: 5% confidence bands

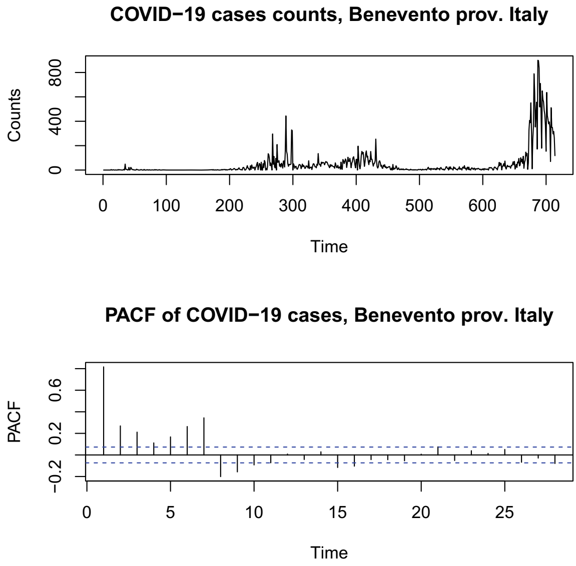


Table 3 QMLE estimates of the linear model (4) for daily COVID-19 new cases in Italy. Standard errors in brackets. Linearity is tested against the ID nonlinear model (5), with χ^2_1 asymptotic test (11); against the STNAR model (6), with approximated p-values computed by score bootstrap (p^J_T), parametric bootstrap ($\bar{p}^J_T$); and versus TNAR model (7)

Models	$\tilde{\beta}_0$	$\tilde{\beta}_1$	$\tilde{\beta}_2$
Linear	1.665 (0.462)	0.149 (0.016)	0.842 (0.025)
Models	χ^2_1	p^J_T	$\bar{p}^J_T$
ID	3.585	–	–
STNAR	–	<0.001	<0.001
TNAR	–	0.600	0.800

sup$LM_T = \sup_{\gamma\in\Gamma_F} LM_T(\gamma)$ through the two bootstrap approximation procedures described in this work. The number of bootstrap replication is set to $J = 299$. For the parametric bootstrap, the generation of pseudo-observations requires the choice of a copula and related parameter. We chose the Gaussian copula with correlation matrix $R = \rho\bar{I}$, and $\rho = 0.5$. Finally, a linearity test against threshold effects, as in (7), is also performed, which leads to the test $H_0 : \alpha_0 = \alpha_1 = \alpha_2 = 0$ versus $H_1 : \alpha_l > 0$, for some $l = 0, 1, 2$. In order to determine a feasible range of values for the non identifiable threshold parameter, we compute the quantiles at 10% and 90% of the empirical distribution for the process $\{X_{i,t} : t = 1, \dots, T\}$, at each $i = 1, \dots, N$. Then, we take the minimum of 10% quantiles and the maximum of 90% quantiles as the extremes of Γ_F, from which a grid of 10 equidistant values is picked.

The results are summarized in Table 3. The estimated parameters for the linear model (4) are highly significant. The magnitude of the network effect β_1 appears to agree with intuition, as an increasing number of cases in a province can lead to a growth in cases found in a close geographic area. The effect of the lagged variable has a upwards impact on the number of cases, as expected by the observed temporal dependence. The linearity test against the nonlinear model (5) is rejected at 0.1 significance level, since the value of the test statistics is greater than the critical values of the χ^2_1 distribution, but not at 0.05 and 0.01 levels. This gives a mild evidence for possible nonlinear drifts in the intercept. The linearity is strongly rejected when tested against the STNAR model, by both bootstrap tests at all levels 0.1, 0.05 and 0.01. Nevertheless, bootstrap sup-type tests do not show evidence of threshold effects in the model. Then, we conclude that there is a clear evidence in accordance to regime switching effects with smooth switching rather than abrupt shifts. These findings are in line with the values of the time series, as shown in Fig. 2.

Acknowledgements This work has been co-financed by the European Regional Development Fund and the Republic of Cyprus through the Research and Innovation Foundation, under the project INFRASTRUCTURES/1216/0017 (IRIDA).

References

1. Ahmad, A., Francq, C.: Poisson QMLE of count time series models. J. Time Ser. Anal. **37**, 291–314 (2016)
2. Andrews, D.W.K., Ploberger, W.: Optimal tests when a nuisance parameter is present only under the alternative. Econometrica **62**, 1383–1414 (1994)
3. Armillotta, M., Fokianos, K.: Poisson network autoregression (2021). arXiv:2104.06296
4. Armillotta, M., Fokianos, K.: Testing linearity for network autoregressive models (2022). arXiv:2202.03852
5. Boos, D.D.: On generalized score tests. Amer. Stat. **46**, 327–333 (1992)
6. Bracher, J., Held, L.: Endemic-epidemic models with discrete-time serial interval distributions for infectious disease prediction. Inter. J. Forecast. in press (2020)
7. Christou, V., Fokianos, K.: Quasi-likelihood inference for negative binomial time series models. J. Time Ser. Anal. **35**, 55–78 (2014)
8. Christou, V., Fokianos, K.: Estimation and testing linearity for non-linear mixed Poisson autoregressions. Electron. J. Stat. **9**, 1357–1377 (2015)
9. Cliff, A., Ord, J.K.: Space-time modelling with an application to regional forecasting. Trans. Inst. Brit. Geograph. 119–128 (1975)
10. Csardi, G., Nepusz, T.: The igraph software package for complex network research. Inter. J. Complex Syst. 1695 (2006). https://igraph.org
11. Davies, R.B.: Hypothesis testing when a nuisance parameter is present only under the alternative. Biometrika **74**, 33–43 (1987)
12. Davis, R.A., Liu, H.: Theory and inference for a class of nonlinear models with application to time series of counts. Stat. Sinica **26**, 1673–1707 (2016)
13. Davis, R.A., Fokianos, K., Holan, S.H., Joe, H., Livsey, J., Lund, R., Pipiras, V., Ravishanker, N.: Count time series: a methodological review. J. Amer. Stat. Assoc. **116**, 1533–1547 (2021)
14. Douc, R., Fokianos, K., Moulines, E.: Asymptotic properties of quasi-maximum likelihood estimators in observation-driven time series models. Electr. J. Stat. **11**, 2707–2740 (2017)
15. Doukhan, P.: Mixing. Lecture Notes in Statistics, vol. 85. Springer, New York (1994)
16. Fan, J., Yao, Q.: Nonlinear Time Series: Nonparametric and Parametric Methods. Springer, New York (2003)
17. Fokianos, K.: Multivariate count time series modelling. To appear in Econometrics and Statistics (2022)
18. Fokianos, K., Neumann, M.H.: A goodness-of-fit test for Poisson count processes. Electr. J. Stat. **7**, 793–819 (2013)
19. Fokianos, K., Tjøstheim, D.: Nonlinear Poisson autoregression. Ann. Inst. Stat. Math. **64**, 1205–1225 (2012)
20. Fokianos, K., Rahbek, A., Tjøstheim, D.: Poisson autoregression. J. Amer. Stat. Assoc. **104**, 1430–1439 (2009)
21. Fokianos, K., Støve, B., Tjøstheim, D., Doukhan, P.: Multivariate count autoregression. Bernoulli **26**, 471–499 (2020)
22. Francq, C., Horvath, L., Zakoïan, J.M.: Sup-tests for linearity in a general nonlinear AR(1) model. Econom. Theory **26**, 965–993 (2010)
23. Gao, J.: Nonlinear Time Series: Semiparametric and Nonparametric Methods. CRC Press, Boca Raton (2007)
24. Gao, J., King, M., Lu, Z., Tjøstheim, D.: Specification testing in nonlinear and nonstationary time series autoregression. Ann. Stat. **37**, 3893–3928 (2009)
25. Gorgi, P.: Beta-negative binomial auto-regressions for modelling integer-valued time series with extreme observations. J. R. Stat. Soc.: Ser. B **82**, 1325–1347 (2020)
26. Gourieroux, C., Monfort, A., Trognon, A.: Pseudo maximum likelihood methods: theory. Econometrica 681–700 (1984)
27. Haggan, V., Ozaki, T.: Modelling nonlinear random vibrations using an amplitude-dependent autoregressive time series model. Biometrika **68**(1), 189–196 (1981)

28. Hansen, B.E.: Inference when a nuisance parameter is not identified under the null hypothesis. Econometrica **64**, 413–430 (1996)
29. Heyde, C.C.: Quasi-likelihood and its Application. A General Approach to Optimal Parameter Estimation. Springer Series in Statistics. Springer, New York (1997)
30. Knight, M., Nunes, M., Nason, G.: Modelling, detrending and decorrelation of network time series (2016). arXiv:1603.03221
31. Knight, M., Leeming, K., Nason, G., Nunes, M.: Generalized network autoregressive processes and the GNAR package. J. Stat. Softw. **96**, 1–36 (2020). https://www.jstatsoft.org/v096/i05
32. Kolaczyk, E.D., Csárdi, G.: Statistical Analysis of Network Data with R, vol. 65. Springer, Berlin (2014)
33. Li, G., Li, W.K.: Testing a linear time series model against its threshold extension. Biometrika **98**, 243–250 (2011)
34. Lim, K., Tong, H.: Threshold autoregressions, limit cycles, and data. J. R. Stat. Soc. Ser B **42**, 245–92 (1980)
35. Luukkonen, R., Saikkonen, P., Teräsvirta, T.: Testing linearity against smooth transition autoregressive models. Biometrika **75**, 491–499 (1988)
36. Martin, R.L., Oeppen, J.: The identification of regional forecasting models using space: time correlation functions. Trans. Inst. Brit. Geograph. 95–118 (1975)
37. McCullagh, P., Nelder, J.A.: Generalized Linear Models, 2nd edn. Chapman & Hall, London (1989)
38. Neumann, M.: Absolute regularity and ergodicity of Poisson count processes. Bernoulli **17**, 1268–1284 (2011)
39. Nowicki, K., Snijders, T.A.B.: Estimation and prediction for stochastic blockstructures. J. Amer. Stat. Assoc. **96**, 1077–1087 (2001)
40. Pedeli, X., Karlis, D.: A bivariate INAR (1) process with application. Stat. Modell. **11**, 325–349 (2011)
41. Pedeli, X., Karlis, D.: On composite likelihood estimation of a multivariate INAR (1) model. J. Time Ser. Anal. **34**, 206–220 (2013)
42. Pedeli, X., Karlis, D.: Some properties of multivariate INAR (1) processes. Comput. Stat. & Data Anal. **67**, 213–225 (2013)
43. Rosenblatt, M.: A central limit theorem and a strong mixing condition. Proc. Natl. Acad. Sci. U. S. A. **42**, 43–47 (1956)
44. Teräsvirta, T.: Specification, estimation, and evaluation of smooth transition autoregressive models. J. Amer. Stat. Assoc. **89**, 208–218 (1994)
45. Teräsvirta, T., Tjøstheim, D., Granger, C.W.J.: Modelling Nonlinear Economic Time Series. Oxford University Press, Oxford (2010)
46. Tong, H.: Non-linear Time Series: A Dynamical System Approach. Oxford University Press, Oxford (1990)
47. Wang, C., Liu, H., Yao, J.F., Davis, R.A., Li, W.K.: Self-excited threshold Poisson autoregression. J. Amer. Stat. Assoc. **109**, 777–787 (2014)
48. Wang, Y.J., Wong, G.Y.: Stochastic blockmodels for directed graphs. J. Amer. Stat. Assoc. **82**, 8–19 (1987)
49. Wasserman, S., Faust, K., et al.: Social Network Analysis: Methods and Applications, vol. 8. Cambridge University Press, Cambridge (1994)
50. Wedderburn, R.W.: Quasi-likelihood functions, generalized linear models, and the Gauss-Newton method. Biometrika **61**(3), 439–447 (1974)
51. Zhao, Y., Levina, E., Zhu, J., et al.: Consistency of community detection in networks under degree-corrected stochastic block models. Ann. Stat. **40**(4), 2266–2292 (2012)
52. Zhou, J., Li, D., Pan, R., Wang, H.: Network GARCH model. Stat. Sin. **30**, 1–18 (2020)
53. Zhu, X., Pan, R.: Grouped network vector autoregression. Stat. Sin. **30**, 1437–1462 (2020)
54. Zhu, X., Pan, R., Li, G., Liu, Y., Wang, H.: Network vector autoregression. Ann. Stat. **45**, 1096–1123 (2017)
55. Zhu, X., Wang, W., Wang, H., Härdle, W.K.: Network quantile autoregression. J. Econometr. **212**, 345–358 (2019)

Novel Data Science Methodologies for Essential Genes Identification Based on Network Analysis

Mario Manzo, Maurizio Giordano, Lucia Maddalena, Mario Rosario Guarracino, and Ilaria Granata

Abstract Essential genes (EGs) are fundamental for the growth and survival of a cell or an organism. Identifying EGs is an important issue in many areas of biomedical research, such as synthetic and system biology, drug development, mechanistic and therapeutic investigations. The essentiality is a context-dependent dynamic attribute of a gene that can vary in different cells, tissues, or pathological conditions, and wet-lab experimental procedures to identify EGs are costly and time-consuming. Commonly explored computational approaches are based on machine learning techniques applied to protein-protein interaction networks, but they are often unsuccessful, especially in the case of human genes. From a biological point of view, the identification of the node essentiality attributes is a challenging task. Nevertheless, from a data science perspective, suitable graph learning approaches still represent an open problem. Node classification in graph modeling/analysis is a machine learning task to predict an unknown node property based on defined node attributes. The model is trained based on both the relationship information and the node attributes. Here, we propose the use of a context-specific integrated network enriched with biological and topological attributes. To tackle the node classification task we exploit different machine and deep learning models. An extensive experimental phase demonstrates

M. Manzo
University of Naples L'Orientale, Via Marina 59, 80131 Naples, Italy
e-mail: mmanzo@unior.it

M. Giordano · L. Maddalena · I. Granata
National Research Council (CNR), Via Pietro Castellino 111, 80131 Naples, Italy
e-mail: maurizio.giordano@cnr.it

L. Maddalena
e-mail: lucia.maddalena@cnr.it

I. Granata
e-mail: ilaria.granata@cnr.it

M. R. Guarracino
National Research University Higher School of Economics, 136 Radionova Ulitsa, Nizhny Novgorod, Russia

University of Cassino and Southern Lazio, Campus Folcara, 03043 Cassino, Italy
e-mail: mario.guarracino@unicas.it

G. Dzemyda et al. (eds.), *Data Science in Applications*,
Studies in Computational Intelligence 1084,
https://doi.org/10.1007/978-3-031-24453-7_7

the effectiveness of both network structure and attributes associated with the nodes for EGs identification.

Keywords Data science · Node classification · Essential genes identification · Integrated network

1 Introduction

Deciphering the set of genes which are essential for guaranteeing survival and reproduction of cells has an enormous interest in biological and health sciences. Indeed, the identification of the so-called "essential genes" (EGs) in different organisms has a great impact in several branches of research, as EGs can contribute to investigate the molecular mechanisms underlying the biological processes [1], the origin and evolution of organisms [2], the minimum cellular demands in the context of synthetic biology [3, 4], and the pathological basis of disease [5, 6]. Furthermore, being involved in cellular basic functions, EGs represent candidate druggable targets for antimicrobial [7] or antitumoral therapies [8, 9]. It has been estimated that EGs represent around the 10% of the genome in humans [1, 10].

The experimental procedures to analyze the essentiality of genes rely on gene deletion/knock-out approaches. It is quite obvious that working on model organisms, such as *Drosophila melanogaster* and microorganisms (e.g., *Escherichia coli*, *Saccharomyces cerevisiae*), or on human cells makes a great difference. In the first case, *in vivo* experiments and direct evaluation of organism viability are allowed. Instead, in the case of humans, the cell-based *in vitro* experiments determine a high heterogeneity of response due to cell type and experimental conditions. A kind of human gene essentiality *in vivo* may be assessed through population genome sequencing data, considering essential those genes rarely or never disrupted or truncated in the general population. Various scores are proposed as essential metrics from human genetic variation data, such as haploinsufficiency probability, loss-of-function intolerance probability, missense Z-score, and others [11]. However, the conversion of such scores, obtained by either *in vitro* or *in vivo* approaches, to dichotomic labels of Essential (E)/Not Essential (NE) genes is not that obvious, being influenced by the setting of threshold values.

Although these evaluations provide interesting insights, they represent one aspect of the essentiality. Indeed, *in vitro* cell-based assays provide a different set of EGrom the one obtained through *in vivo* human population studies [12], likely due to the different contexts: tumour cells viability versus organism fitness. Gene essentiality is not a static property, but a changeable characteristic contextualized by environmental, genetic, and evolutionary factors [13]. Therefore, a bias can be introduced when analyzing the gene essentiality through an organism- rather than a context-based approach. Achilles Cell Line Gene Essentiality Profiles [14] project made an attempt of genome-wide experiments, creating a catalog of EGs by exploiting CRISPR (Clustered Regularly Interspaced Short Palindromic Repeats)-Cas9 and RNA interference

(RNAi)-based screens across hundreds of cancer cell lines to silence or knockout individual genes and identify those genes that affect cell survival. However, these screening methods are complex, costly, labor-, and time-intensive [15].

Consequently, data science approaches were deployed to complement the experimental techniques and, thus, to minimize the resources required for essentiality assays [15].

Over the years, several analytics methods have been devised to predict EGs. Most of the works have been done on model organisms or microorganisms for synthetic biology purposes. Compared to humans, these models present a lower heterogeneity and an easier definition of essentiality labels for the learning task. Although the results are not generalizable to humans, they still represent an important source of approaches developed to classify EGs. One of the main tasks to address is undoubtedly the identification of the attributes that define the EGs. An attempt to collect biological and genetic characteristics related to the essentiality has been made in [1], where the authors provided a comprehensive study of human EGs, including their genomic, epigenetic, proteomic, evolutionary, and embryonic patterning characteristics. According to the rule of centrality-lethality [16, 17], most of the EGs-related characteristics come from protein-protein interaction (PPI) networks [18], used to derive centrality metrics as attributes for essentiality prediction models. From PPI to disease networks, from healthcare systems to scientific knowledge, biomedical networks are general descriptions of systems of interacting entities. In the last decade, we have seen a quick expansion of representation learning approaches for modeling, analyzing, and learning such networks, thanks to the extraordinary effectiveness in giving significant predictions and insights.

Given the wide variety of knowledge that can be extracted from network-based representations and the high complexity of the essentiality concept, it seems simplistic to rely only on physical interactions. According to the above considerations, here, we present a work on the classification of EGs through a tissue-specific approach, applied to kidney, which, to the best of our knowledge, is the first one in this context. Furthermore, to associate network attributes to the genes we use an integrated network made of physical and metabolic interactions. In particular, the metabolic machinery and the underlying connections have a crucial role in cellular functionalities and response to stimuli, so much that the metabolic networks are widely exploited for precision medicine purposes [19].

Finally, to address the problem of defining E/NE genes, we introduce a novel methodology which provides tissue-specific gene labels for kidney.

In order to validate the proposed method, we refer to supervised learning approaches from the computational point of view. In this context, the identification of EGs concerns a binary classification task and the algorithms learn a prediction model based on features related to gene essentiality from both biological and network contexts. Two types of approaches were adopted. The first by associating to the nodes the description of their biological and network attributes. In this case, Machine Learning (ML) algorithms were applied. The second, otherwise, by adding information through an integrated network. Structural information was adopted to apply geometric Deep Learning (DL) algorithms. Here we show the workflow and performance of both sets of methods, exploring their results.

The paper is structured as follows. Section 2 includes an overview of the state-of-art on node classification applied to the EGs identification task. Section 3 provides details on how the graph-based data was built and on the selected attributes. Section 4 describes the various machine and deep learning techniques adopted to tackle the node classification task related to EGs identification. Section 5 provides a comprehensive experimental phase, while Sect. 6 concludes the paper.

2 State-of-the-Art

Over the years, various attempts have been made to propose workflows and provide guides to tackle the task of EGs classification. A wide literature exists and we only report some of the most recent reviews and results.

Dong et al. [20] report several studies in which different computational methods and biological features have been exploited to identify EGs both in prokaryotes and eukaryotes, and pointed out the crucial role of the features selection and combination to improve the performance. To this extent, they implement and test five features they consider representative (i.e., evolutionary conservation, domain information, network topology, sequence component, and expression level) on data of *Escherichia coli* MG1655, *Bacillus subtilis* 168, and human. They discuss modeling approaches based on ML algorithms, useful to deal with large and complex data sets, and empiric formulas of specific features, and mention popular online services to predict EGs.

Li et al. [3] present a survey on network-based methods for predicting EGs or proteins. Topology-based methods are grouped according to their exploitation of neighborhood, path, or eigenvector information, or their combination. Methods integrating PPI networks with biological information, exploiting dynamic networks, and based on ML are also discussed. Further useful information is provided on available databases and tools.

The very recent review by Aromolaran et al. [15] focuses on gene essentiality prediction by the ML approach. Among its challenges, they identify the incomplete and prone to errors information from model organisms that affects the classifiers, favoring those studies that assemble class labels from multiple sources, rather than a single one. A comparative analysis is performed on essentiality prediction for *Caenorhabditis elegans* using different features. These include intrinsic and extrinsic features (i.e., those that can be directly derived from gene and protein sequences or those that can be computed only from the sequence's interaction with another sequence or its environment), the latter type also including topology-based features.

Aromolaran et al. [21] present an ML approach to EG prediction based on the combination of a wide set of intrinsic and extrinsic features applied to *Drosophila melanogaster*, but also extended to human data. Performance comparisons are made against the methods proposed by Campos et al. [22], who shared the source code[1] for their systematic analysis of EG prediction within and among species. The authors demonstrated that ML models (Generalised Linear Model, Artificial Neural Network, Gradient Boosting, Support Vector Machine (SVM) [23], and Random Forest (RF) [24]) trained with subsets of essentiality-related data performed better than random guessing of gene essentiality for a particular species.

Hasan and Lonardi [2] propose a MultiLayer Perceptron (MLP) network for predicting EGs based only on intrinsic features (only gene primary sequence and corresponding protein sequence). Interestingly, they balance the data by down-sampling the class of non-EGs. Moreover, they identify a possible data leak in case two homologous genes are used one for training the model and the other for testing it. To avoid this bias, they cluster (via OrthoMCL) the set of all genes into orthologous, homologous, and paralog, and make sure that no gene from a single cluster is assigned to both training and test set.

Zeng et al. [25] propose a DL framework to automatically learn biological features without prior knowledge. Specifically, they adopt topological features extracted by PPI networks via node2vec [26], gene expression features extracted via bidirectional long short-term memory cells, and subcellular localization information exploited through an indicator vector. The concatenation of these features is fed to a fully connected layer with a sigmoid activation function to perform classification. Extensive comparisons are provided against some topology-based and machine learning-based methods. Moreover, an ablation study is carried out to investigate the role of each of the three biological information, revealing that the PPI embedding is the most important component, but still, the other two sources help in enhancing performance.

Dai et al. [27] propose a network embedding approach to human EGs identification. The node embeddings of a human PPI network are first computed, based on random walks and word2vec, and then classified using state-of-the-art classifiers. Extensive experiments are provided on two different human PPI networks (from the Reactome [28] and the InBio Map [29] databases, respectively), varying the classifier (SVM, Deep Neural Network (DNN), Decision Trees, Naïve Bayes, k-Nearest Neighbor, Logistic Regression, RF, and Extra Tree) and comparing the results to those achieved by other methods (Z-curve, centrality-based, DeepWalk, and LINE). Interestingly, to handle the imbalance between EGs and non-EGs, in cross-validation, they consider both stratified partitions (i.e., partitions having class distributions similar to the whole data) and partitions where the ratio of the two classes is 1:1, in the second case achieving better performance results.

Rezaeia et al. [30], adopt a critical node detection problem (CNDP) coupled with a genetic algorithm to define the correlation between essential proteins and their centrality properties in a PPI network of two species, *Escherichia coli* and *Saccharomyces cerevisiae*. The CDNP is a well-known graph theory method that allows detecting the nodes crucial to the stability of the network. The authors demonstrate that EGs are significantly present in the set of the identified critical nodes and have

[1] https://bitbucket.org/tuliocampos/essential.

different topological properties compared to the EGs found by ranking methods using the standard centrality measures.

Schapke et al. [31] introduce EPGAT, a novel method for Essentiality Prediction with Graph Attention Networks (GATs). Based on Graph Neural Networks (GNNs), it learns directly and automatically the nodes' relations from the PPI network. Furthermore, the authors incorporate in the model learning process 4 biological features one by one (i.e., gene expression profiles, orthology information, and subcellular localization), and evaluate the performance of *S. cerevisiae*, *E. coli*, *D. melanogaster* and *H. sapiens*. They compare the AUC (Area Under The Curve) ROC (Receiver Operating Characteristics) curve values of the proposed method with those obtained with other network-based and ML techniques commonly exploited. EPGAT shows performance comparable to node2vec embedding, with a shorter training time.

Zhang et al. [32] describe a DL-based framework, called DeepHE, which integrates sequence features, both at DNA and protein level, and network features automatically extracted from the PPI network by using node2vec. In order to address the imbalanced learning problem, a cost-sensitive technique is adopted during the training of the architecture. The embedding features learned by node2vec improved the results on human datasets and make DeepHE outperform compared to ML methods and node centrality measures. The authors have demonstrated that human EGs can be accurately detected by designing an effective artificial intelligence framework and integrating representative features captured from available biological data. They also underline the benefits that may be obtained by applying DL methods to automatically derive biological features.

Kuang et al. developed XGEP (eXpression-based Gene Essentiality Prediction), a ML approach to predict the essentiality of both protein-coding genes and long non-coding RNAs (lncRNAs) in cancer cells through a collaborative embedding applied to TCGA transcriptomic profiles [33]. The biological information relevant for the learning task is extracted through three different approaches: collaborative filtering, which gives the best performance, Gene2Vec, and autoencoder. Finally, the feature vectors derived from collaborative embedding are adopted to build gradient boosted tree, SVM, and DNN models for human EGs prediction.

3 Materials

3.1 Gene Integrated Network: PPI+MET

The kidney-specific network used in this study has been generated by integrating the kidney metabolic (MET) [34, 35] and kidney PPI networks. The metabolic network has been obtained by extracting enzymes relationships from the kidney-specific genome-scale metabolic model, downloaded from the Human Metabolic Atlas repository [36]. Two enzymes are connected if they catalyze reactions producing and consuming a given metabolite, respectively. The metabolic network consists of 2945 nodes and 663397 edges. The kidney PPI network has been downloaded from the Integrated Interactions Database, one of the most comprehensive sets of context-specific human PPI networks [37]. It is based on physical connections between proteins and consists of 11741 nodes and 569585 edges. The integration of the two networks has been performed at the gene-level, converting proteins and enzymes to the corresponding gene symbol, thus having genes as nodes connected according to metabolic and/or physical interactions. The edges have been weighted by summing up the HPA (Human Protein Atlas) [38] expression values in the kidney of the two enzymes involved. In order to simplify the network, although the metabolic network is naturally directed, we considered the integrated network undirected, as the PPI network. Self-loops and duplicated edges have been removed. The largest connected component has been considered for the analysis, resulting in a network with 12538 nodes and 1066252 edges.

3.2 Biological Attributes

Many studies have been conducted to determine the genetic and functional features that characterize EGs, ranging from genome-based to protein-based and mechanistic-based features [1, 39]. In the current study, given the focus on specific tissue, we collected both tissue-specific and generic biological attributes, summarized in Table 1.

The tissue-specific attributes consist of gene expression levels in kidney normal and tumor tissue. EGs are indeed highly expressed as fundamental for main cellular functionalities. Furthermore, the expression levels seem to be altered in tumor tissues compared to normal ones, as they are sensitive to tumorigenesis [39]. The tissue-specific attributes used in this study are: “GTEx_kidney”, the gene median transcripts per million (tpm) counts in kidney cortex and kidney medulla, downloaded from GTEx portal [40]; “OncoDB_DEG”, differentially expressed genes (DEG) from OncoDB [41] in renal tumors (Kidney Renal Clear Cell Carcinoma KIRC, Kidney Renal Papillary Carcinoma KIRP, Kidney Chromophobe KICH) selected by FDR adjusted p-value $\leq$ 0.05; “HPA_kidney”, normalized transcript expression levels summarized per gene in kidney tissue based on RNA-seq from HPA [38]; “GTEX-*”, gene tpm expression of 89 kidney cortex and medulla samples, from the GTEx portal.

The generic biological attributes regard both genetic and functional characteristics. “Gene_length”, “GC_content” and “Transcript_count” have been included as associated with DNA stability and obtained from the Ensembl database through biomaRt R package v 2.50.3 [42]. EGs are in their definition crucial to fundamental biological processes. As they have been found in many model organisms as hub nodes of PPI networks showing high centrality scores [27], we thought that this aspect can be translated into biological centrality through the involvement in many processes, functions, and pathways, as well as the expression in multiple cellular compartments and tissues. Based on this idea, we added several attributes coming from the gene enrichment analysis. In particular, we used the DAVID bioinformatics database [43, 44] to annotate our nodes/genes list for several functional annotations from Gene Ontology (GO) and others. The counts of involved biological functions from GO-Molecular Functions ("MF") and GO-Biological Processes ("BP"), pathways from "Biogrid", "KEGG", "Reactome", expression from GO-Cellular Component ("CC"), and "UP_tissue", have been added as attributes for each gene. Moreover, as EGs likely interact with many transcription factors and have conserved motifs, we also obtained from DAVID the count of predicted TFs (Transcription Factors Binding Sites) from "Ucsc_tfbs". The UCSC TFBS conserved track identifies motifs that are conserved across humans, mice, and rats and scores these sites based on the motif match. Several findings have pointed out that EGs are highly conserved, even if the essentiality seems to be a species-specific prerogative [1, 45]. According to this, we included the orthologs count "Orth_count" for each gene, an attribute obtained from NCBI Gene database [46]. Finally, the attribute "Gene_Disease_ass_count" was included as an indicator of gene association to human diseases, as it seems that disease-associated genes are intermediates between highly essential and non-essential genes [47]. The gene-disease associations have been downloaded from DisGeNET [48], a dedicated database, and the number of associations has been calculated for each gene.

Table 1 Biological attributes

Name	Description	Data source
GTEx_kidney	Gene median tpm in kidney	GTEX
OncoDB_DEG	Significant DEG in renal tumors	OncoDB
HPA_kidney	Median gene expression in kidney	HPA
GTEX-*	Samples gene tpm	GTEX
Gene_Disease_ass_count	Gene-disease association count	DISGENET
Gene_length	Length from gene start to end	Ensembl BioMart
GC_content	Guanine-Cytosine % content	Ensembl BioMart
Transcript_count	Number of transcripts	Ensembl BioMart
MF	Count of GO-MF terms	DAVID
BP	Count of GO-BP terms	DAVID
CC	Count of GO-CC terms	DAVID
Biogrid	Count of Biogrid pathways	DAVID
KEGG	Count of KEGG pathways	DAVID
Reactome	Count of Reactome pathways	DAVID
Ucsc_tfbs	Count of predicted TFs	DAVID
UP_tissue	Count of expression in tissues	DAVID
Orth_count	Number of orthologs	NCBI

Table 2 Network structural attributes

Name	Description
Degree	Number of adjacent edges
Strength	Sum of the weights of adjacent edges
Eccentricity	Centrality based on the max. shortest path distance to any other node
Closeness	Centrality based on the no. of steps to reach any other node
Betweenness	Centrality based on the no. of shortest paths passing through the node
Eigen-centrality	Centrality based on the importance of its adjacent nodes [51]
Hub score	Centrality based on the connections to important nodes [57]
Page Rank	Scores obtained by the Google PageRank algorithm [58]
Transitivity	Clustering coefficient [53], measures local cohesiveness based on node triplets
Triangles	Number of node triangles the node is part of
Motif#	Number of motifs [56] of order 3, type # (#=1, 2, 3, 5)

3.3 Network Attributes

In many model organisms and microorganisms, EGs have been demonstrated to have a high degree of connectivity in PPI networks. Network topology attributes are then investigated to analyze EGs in their neighborhood context. In our opinion, considering only the physical interactions between proteins as a symptom of essentiality is reductive, as the functional connections in the context of signaling and metabolic pathways are not taken into account using only interaction information. According to this, we evaluated in the PPI+MET network the network attributes briefly described in Table 2. They represent typical topological information extracted from the network [49], frequently adopted in the literature due to their strong connections with the role of EGs [3, 15, 20, 50]. Indeed, “degree”/“strength” centrality has a strict correlation with essentiality, as a gene with a high number of incident edges is more likely to be essential [15], and this is even more evident when those incident edges have a high weight. In a PPI network, a low “eccentricity” of a protein can be interpreted as its easiness to be functionally reached by all the other proteins in the network. “Closeness” describes how fast a node can communicate with the other nodes of a network, while “betweenness” centrality quantifies the ability of a node to monitor the communication between other nodes of the network [15]. The “eigen-centrality” of a node [51, 52] describes its importance in a graph, based on that of its adjacent nodes. The “hub score” (or Kleinberg’s centrality) takes into account the observation that a node can be important not only if it contains valuable content and hence receives many links from other important sources (authority centrality), but also because it links to other important nodes (hub centrality). The “Page Rank” score, typically used in social network modeling, provides a measure of the importance of a website page, obtained by counting the number and quality of links to it. In our opinion, it clearly applies well also in the case of gene networks. The “transitivity” (or weighted clustering coefficient [53]) is a measure of the local cohesiveness in a graph that takes into account the importance of the clustered structure based on the amount of interaction intensity actually found on local node triplets. It counts for each triplet formed in the neighborhood of a node ("triangle") the weight of the two participating edges of the node. All these attributes have been computed using the igraph R package v. 1.2.11 [54].

Our network attributes also include "motifs", which are recurrent and statistically significant subgraphs or patterns of a network [55]. Sporns and Kötter defined functional motifs as combinations of nodes and connections that could occur within the constraints of a given structural motif and analyzed their frequency in brain networks [56]. We exploited their definition to obtain the frequency of motifs of order $M = 3$, where M is the number of nodes involved in the specific motif, for each node of our integrated network. Using the Matlab code[2] provided by the authors, we obtained four of the thirteen possible node configurations for motifs (referred to as Motif1, Motif2, Motif3, and Motif5).

3.4 Data Pre-processing

Data pre-processing dealt with missing information and data normalization. Missing values of node attributes were fixed by replacing them with the mean of the values for that attribute throughout the node samples. Node attribute values were normalized by means of the z-score normalization, in such a way the new ranges of attribute values have zero mean and unit standard deviation. On the other side, values of the single edge attribute, i.e., the edge weight, were replaced by the min-max normalized value, such that the new weights are scaled down to the range [0, 1], where 0 and 1 correspond to the lowest and the highest edge weight, respectively.

3.5 Gene Essentiality Labeling

One of the main challenges in classifying human EGs is the assignment of the E/NE labels. As discussed in Sect. 1, unlike microorganisms and model organisms, where the experimental procedures consist of verifying the lethality of the gene knock-out in the whole organism *in vivo*, in the case of humans the experiments are performed on cell cultures *in vitro*, determining a high heterogeneity of response, depending on tissue/cell-specific factors, and providing scores difficult to convert in boolean logic.

One of the main sources for getting a list of EGs is the Database of Essential Genes, which contains EGs for 48 kinds of bacteria, 26 eukaryotes, and one of archaea. Since here we propose to overcome the heterogeneity of the tissue through a specific-tissue approach, we could not exploit the human list. One of the main essentiality screening methods is the CRISPR/Cas9-based test. CRISPR/Cas9 is a gene-editing/deleting

[2] https://sites.google.com/site/bctnet/.

technology that has facilitated the investigation of cellular responses to stimuli at a genome-wide scale. Gene Effect scores of 39 kidney cell lines derived from CRISPR knockout screens published by Broad's Achilles and Sanger's SCORE projects were downloaded from the DepMap portal.[3] Negative scores imply cell growth inhibition and/or death following gene knockout. Scores are normalized such that non-EGs have a median score of 0 and independently identified common essentials have a median score of −1.

Taking inspiration from the approach presented in [1], we divided the scores into 11 CRISPR score groups, from CS0 to CS10. The labels vector for the classification task was obtained by assigning the label of the gene to the most frequent score group among the 39 cell lines. Nodes belonging to CS0 group were labeled E. Some of the genes of the network were not present in the experimental data and an "ND" label was assigned. The threshold values of the 11 score groups are shown in Table 3.

The distribution of genes in CS groups for some biological and network attributes are shown in Fig. 1 (bottom part of each panel). For each attribute, the pairwise Wilcoxon Rank-Sum test has been performed to get the significant differences between each couple of CS groups (Fig. 1 top of each panel). Among the biological attributes, the most significant and evident differences are given by the gene expression-related attributes, where in particular "HPA_kidney" shows a descendent trend going from CS0 to CS9 (Fig. 1a, h, i), the involvement in pathways (Fig. 1g), and the number of orthologs (Fig. 1f). Concerning the network attributes, instead, the centrality attributes, such as Hub score, Degree, Eigen-centrality and Page Rank, show an interesting descending order going till CS7 group, but that then change for CS8 and CS9 genes (Fig. 1m, j, n, k), which for sure requires further investigations.

The gene essentiality labeling proposed in this work was driven by experiments, where we performed seven study trials. In each trial, we assumed as E the genes belonging to the CS0 score group containing the most negative values, while as NE the genes from the union of groups CSX-CS9, with X varying from 1 to 7. The PPI+MET network and CS nodes are represented in Fig. 2 CS0 nodes are mostly located in the core of the network (Fig. 2a), are highly connected (Fig. 2b shows the direct neighborhood of CS0 nodes), and interconnected (Fig. 2c shows only the CS0 nodes and the edges connecting them). We did not consider the score group CS10, since it never appears as the most frequent, due to the very few samples falling into this group. We removed from the network data all genes not included in the chosen groups CS0 and CSX-CS9. In each trial, we carried out a 5-fold stratified cross-validation measuring Accuracy (total and per class) and MCC metrics

Table 3 Breakpoints used to assign nodes to CS groups

Group	CS0	CS1	CS2	CS3	CS4	CS5	CS6	CS7	CS8	CS9	CS10
Low th	−3.480	−1.056	−0.508	−0.293	−0.167	−0.070	0.010	0.087	0.172	0.287	1.295
High th	−1.056	−0.508	−0.293	−0.167	−0.070	0.010	0.087	0.172	0.287	1.295	2.530

[3] https://depmap.org/portal/.

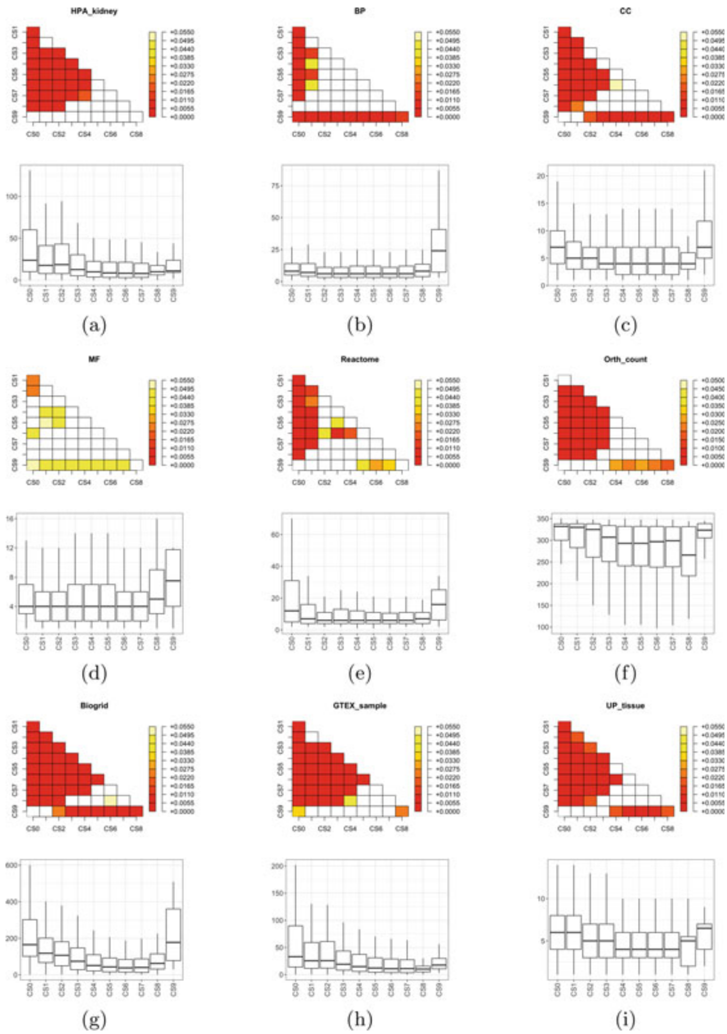


Fig. 1 Statistical significance of pairwise differences and distribution of some Biological and Network attributes according to CS grouping. For the specific attribute indicated in the main title of each panel, the top plot reports the color matrix of the Benjamini-Hochberg adjusted pvalues resulting from pairwise Wilcoxon Rank-Sum tests performed for each couple of CS groups. The pvalue color key is shown on the right. The white cells indicate a pvalue ≥ 0.055. On the bottom part of each panel, the boxplot shows the distribution of the specific attribute throughout the CS groups. Within each box, horizontal lines indicate median values; boxes extend from the 25th to the 75th percentile of each group's distribution of values; vertical extending lines denote adjacent values (i.e., the most extreme values within 1.5 interquartile range of the 25th and 75th percentile of each group); observations outside the range of adjacent values have been removed in favour of the visualization. GTEX_sample is one of the GTEX_* samples gene tpm. Attributes for which few or no significant differences have been obtained are not shown

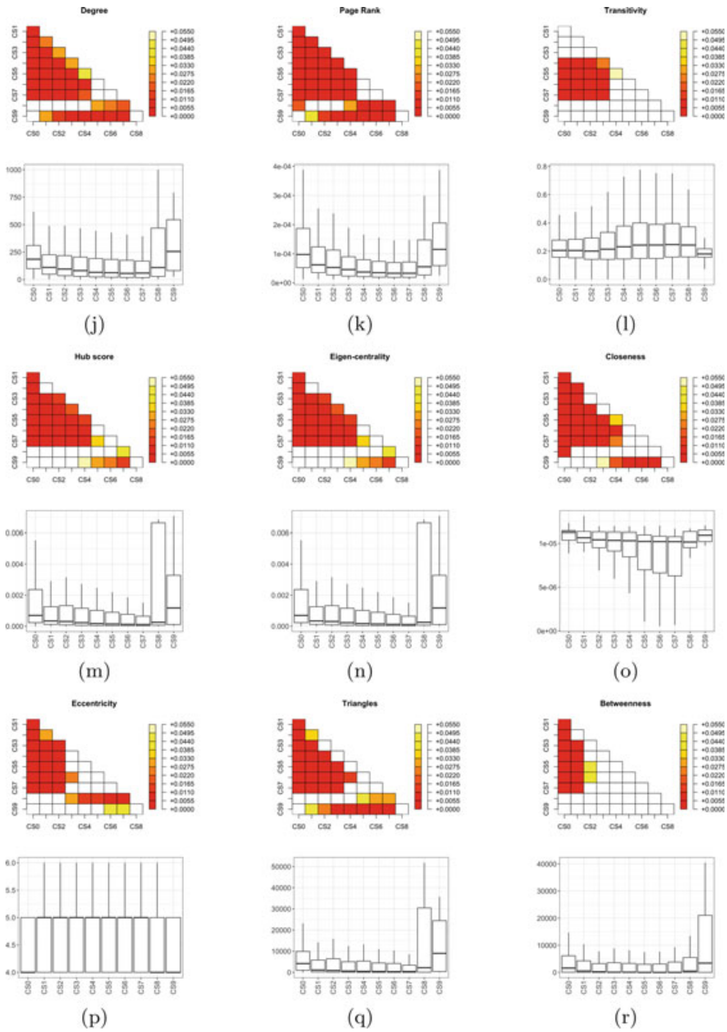


Fig. 1 (continued)

using the Random Undersampling Boosting (RUS) classifier [60]. The results are reported in Table 4 (see Sect. 5 for the definition of the adopted metrics). Since in the experiments the best overall accuracy and MCC values were obtained in the CS0 v. CS6-CS9 classification problem, we adopted this choice as a reference for the gene essentiality labeling criterion. As a consequence, for genes having scores in intermediate intervals, we assumed there is ambiguity and we cannot state gene essentiality within an acceptable tolerance. All these genes were removed from the network data, resulting in a reduction of the original gene dataset to 3814 genes (745 E and 3069 NE).

Table 4 Performance results of the RUS classifier on the Kidney MET-PPI network for different binary problems (Bio attributes, class = NE = 1, E = 0)

Problem	Acc	Sens	Spec	MCC	Confusion matrix
CS0 versus CS7-9	75.51±3.69	80.26±6.64	69.87±7.08	0.51±0.07	[[598, 147], [189, 438]]
CS0 versus CS6-9	87.44±1.92	72.20±4.32	91.14±2.04	0.62±0.05	[[538, 207], [272, 2797]]
CS0 versus CS5-9	86.62±1.30	56.35±7.68	90.17±1.32	0.40±0.06	[[420, 325], [623, 5716]]
CS0 versus CS4-9	87.13±0.92	55.97±6.30	89.83±0.77	0.36±0.05	[[417, 328], [873, 7711]]
CS0 versus CS3-9	86.68±1.01	55.72±4.61	89.10±0.96	0.33±0.04	[[415, 330], [1038, 8489]]
CS0 versus CS2-9	85.81±1.14	54.91±6.57	88.07±1.13	0.30±0.05	[[409, 336], [1213, 8956]]
CS0 versus CS1-9	83.90±0.83	56.13±8.01	85.79±0.92	0.27±0.05	[[418, 327], [1556, 9395]]

Table 5 Classifiers and their parameter settings

Acronym	Method	Params	Library
SVM	Support Vector Machine	n. kernel = rbf, degree = 3, tol = 10^{-1}, C = 1.0	sklearn
RF	Random Forest	n. estimators = 100, criterion = gini, min samples split = 2, min samples leaf = 1, max depth = None	sklearn
XGB	eXtreme Gradient Boosting	n. estimators = 100, learning rate = 0.3, gamma = 0, max depth = 6, booster = gbtree	sklearn
MLP	MultiLayer Perceptron	hidden layer size: 32, dropout: 0.2, epochs = 1000	sklearn
RUS	Random Undersampling Boosting	n. base estimator = Decision Tree, max depth = 1, n. estimators = 50, learning rate = 1.0. algorithm = SAMME.R	imblearn
N2V+MLP	node2vec embedder + MLP classifier	N2V (embedding dim: 128, walk length: 64, context size: 64, walks per node: 64, epochs: 50) MLP (1layer = 32+droput(0.2)+2layer = 32, epochs: 1000)	pytorchGeo
1-GCN	OneLayer Graph Convolutional Neural Network	hidden layer size = 16, droput = 0, learning rate = 0.01, weight decay = 5 10^{-4}, epochs = 1000	pytorchGeo
ChebGCN	Chebyshev Spectral Graph Convolutional Neural Network	hidden layer size = 16, dropout = 0, learning rate = 0.01, weight decay = 5 10^{-4}, epochs = 200	pytorchGeo

4 Methods

Biological networks are data objects with structure and topological properties in a non-Euclidean data space. GNNs [61] are DL models designed to apply directly to non-Euclidean data in the form of graph nodes and their connections (edges). They extract and learn from networked data by simulating how this information is propagated via neighborhood by following network connections. In this context, learning on networks is usually referred to as *graph representation learning*. On the other hand, ML methods apply to structured data objects in the form of vectors, or their multidimensional generalization, the tensors in the Euclidean k-dimensional real space. As a consequence, they can be applied to networks only when node/edge properties (*features*) are extracted before the learning task and put in the form of k-sized real tensors. Here, we refer to network learning as *feature-based representation learning*. The main difference between the above two approaches is that while GNNs automatically learn features from input networks, in the case of ML methods, the data scientist decides a priori which network features to extract and use for the learning process. A hybrid approach is represented by *graph embedding* methods [62–65]: by processing properties of nodes, edges, and node neighbourhoods, these methods perform a transformation of graph nodes into a d-dimensional vectors in a new real space (called *latent space*). This *embedded* node representation can be more manageable or straightforward to process by ML, and, very often, it results in better

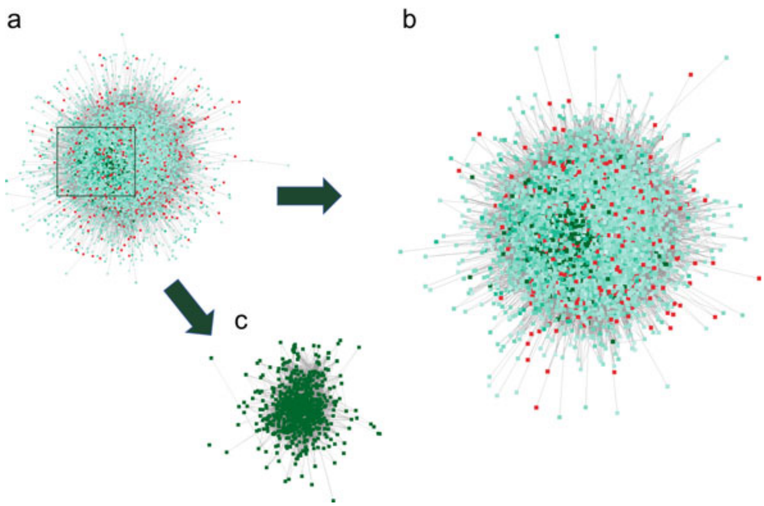


Fig. 2 Representation of the network and CS nodes. **a** PPI+MET network. Nodes are colored with scaling shades of green according to CS group. CS0 nodes are colored by dark green and the box indicates their main localization. Red nodes are those belonging to the "ND" group, for which the label is missing; **b** CS0 nodes and direct neighbors. The extracted subnetwork is densely connected, highlighting the high centrality degree of the CS0 nodes; **c** Zoom on CS0 nodes. The figure has been realized through Cytoscape 3.9.1 [59]

performances. In this case, network learning can be referred to as *embedding-based learning*.

Table 5 lists the classification methods used in the experimental study together with parameters settings and implementation libraries.[4] The methods can be associated to the aforementioned network learning categories as follows

- *Feature-based representation learning*: SVM, RF, XGB [66], MLP, and RUS treat nodes as independent entities: they do not leverage (implicitly) the network structure. Since network-related information of nodes (e.g., centrality, neighborhood, and density) is manually extracted before classification and associated as attributes to nodes, we can say that ML methods use the network information chosen by the user as the only input to the learning process.
- *Graph representation learning*: 1-GCN [67] and ChebGCN [68] implicitly use the input network topology as a path for information flow during the learning process. GNNs also exploit user-specified node/edge information for graph learning. Our study also describes nodes using their role and connectivity inside the network, making the GNNs exploit this information twice.
- *Embedding-based methods*: N2V+MLP is a pipeline consisting of an embedding stage performed by a node2vec model [26] followed by a classification stage implemented by an MLP. Node2vec uses the input network topology as a path for learning a mapping (*embedding*) of nodes in a low-dimensional features space that maximizes the likelihood of preserving network neighborhoods of nodes. In our study, node embeddings are added to the user-defined node properties (Bio, GTEX, and net attributes). The resulting matrix of node attributes is the input of an MLP classifier that performs node class learning and prediction.

5 Results

We conducted a stratified 5-fold cross-validation for gene essentiality classification in the network described in Sect. 3. At each validation step, 80% of nodes were randomly selected to form the dataset for the classifier training, while the remaining 20% of nodes were input to the built models for predictions.

We remind that edges data and attributes are only processed by GNN models and node2vec. As already mentioned, we adopted min-max normalization of edge weights since the z-score normalization would have produced negative weights with anomalous effects on the GNNs data propagation rule. For all the other methods, only the node attribute normalization may affect the performance.

Cross-validation has been performed using each of the classifiers listed in Table 5. We repeated the experiments for different selections of the node attributes, using only one of the node attributes subsets: generic biological (Bio), tissue-specific (GTEX),

[4] Scikit-Learn: https://scikit-learn.org/stable/, Pytorch Geometric: https://pytorch-geometric.readthedocs.io/en/latest/, Imbalanced-learn: https://imbalanced-learn.org/stable/.

and network (net) attributes. In addition, we also run experiments for three combinations of these subsets: all attributes (Bio+GTEX+net), Bio+GTEX, and Bio+net.

The experimental results[5] are reported in Tables 6 and 7. Here, performance is measured by using the following metrics: Accuracy (Acc), Sensitivity (Sens), Specificity (Spec), Balanced Accuracy (BA), and Matthews Correlation Coefficient (MCC), defined as

$$\text{Acc} = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

$$\text{Sens} = \frac{TP}{TP + FN} \tag{2}$$

$$\text{Spec} = \frac{TN}{TN + FP} \tag{3}$$

$$\text{BA} = \frac{\text{Sensitivity} + \text{Specificity}}{2} \tag{4}$$

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{5}$$

where True Positive (TP) is the number of positive class (E) samples the model predicted correctly; True Negative (TN) is the number of negative class samples (NE) the model predicted correctly; False Positive (FP) is the number of negative class samples (NE) the model predicted incorrectly, and False Negative (FN) is the number of positive class samples (E) the model predicted incorrectly. For completeness, in Tables 6 and 7 we also report the Confusion Matrix ([[TP,FN], [FP,TN]]).

To better evaluate the performance measures reported in Tables 6 and 7, we would like to highlight which are the values of these metrics in the two *null-classification* cases. The first null-classification occurs when the method predicts all genes as "essential": since the total number of genes is 3814 and the amount of genes labeled as E is 745, we have: TP $= 745$, TN $= 0$, FP $= 3069$, FN $= 0$, Acc ≈ 0.195, Sens $= 1$, Spec $= 0$, BA $= 0.5$, MCC $= 0$. The second null-classification occurs when the method predicts all genes as "not-essential": since the amount of genes labeled as NE is 3069, we have: TP $= 0$, TN $= 3069$, FP $= 0$, FN $= 745$, Acc ≈ 0.805, Sens $= 0$, Spec $= 1$, BA $= 0.5$, MCC $= 0$. From these numbers, and due to the extreme imbalance of class samples, it is clear how the accuracy, the sensitivity, and the specificity alone do not allow us to completely evaluate the gene prediction performance of a method. Indeed, it is quite easy to get a high accuracy given by a good classification performance on the much more abundant class of NE genes. On the other hand, the higher the balanced accuracy and the MCC are, the better the method capability to predict gene essentiality is.

[5] Google Colab notebook for result reproducibility are available at: https://github.com/giordamaug/EG-identification---Data-Science-in-App-Springer/tree/main/notebook.

Table 6 CS0 versus CS6-CS9 problem: performance of 5-fold stratified cross-validation under different node attribute selections (Part 1). The highest BA and MCC values for each set of attributes are in boldface

Method	Acc	BA	Sens	Spec	MCC	Confusion Matrix
Bio+GTEX+net attributes (119)						
SVM	0.865	0.698	0.425	0.970	0.510	[[320, 425], [90, 2979]]
RF	0.882	0.740	0.507	0.972	0.585	[[380, 365], [84, 2985]]
XGB	0.891	0.786	0.616	0.956	0.630	[[463, 282], [134, 2935]]
MLP	0.871	0.773	0.615	0.931	0.574	[[463, 282], [209, 2860]]
RUS	0.825	0.800	0.762	0.839	0.536	[[571, 174], [492, 2577]]
N2V+MLP	0.858	**0.838**	0.806	0.870	0.610	[[601, 144], [397, 2672]]
1-GCN	0.842	0.769	0.651	0.887	0.517	[[489, 256], [347, 2722]]
ChebGCN	0.892	0.789	0.622	0.957	**0.633**	[[466, 279], [132, 2937]]
Bio+GTEX attributes (105)						
SVM	0.861	0.712	0.469	0.955	0.506	[[351, 394], [137, 2932]]
RF	0.889	0.756	0.537	0.975	0.615	[[400, 345], [77, 2992]]
XGB	0.889	0.785	0.616	0.955	0.623	[[459, 286], [139, 2930]]
MLP	0.879	0.773	0.600	0.946	0.594	[[447, 298], [163, 2906]]
RUS	0.818	0.804	0.782	0.825	0.532	[[585, 160], [534, 2535]]
N2V+MLP	0.862	**0.846**	0.821	0.871	0.627	[[613, 132], [393, 2676]]
1-GCN	0.843	0.763	0.634	0.892	0.511	[[477, 268], [332, 2737]]
ChebGCN	0.893	0.793	0.630	0.955	**0.639**	[[473, 272], [136, 2933]]

(continued)

Table 6 (continued)

Method	Acc	BA	Sens	Spec	MCC	Confusion Matrix
Bio+net attributes (30)						
SVM	0.865	0.699	0.428	0.970	0.513	[[322, 423], [90, 2979]]
RF	0.883	0.746	0.525	0.967	0.586	[[396, 349], [99, 2970]]
XGB	0.878	0.762	0.574	0.950	0.584	[[433, 312], [152, 2917]]
MLP	0.869	0.760	0.583	0.937	0.560	[[438, 307], [191, 2878]]
RUS	0.812	0.791	0.760	0.822	0.513	[[572, 173], [544, 2525]]
N2V+MLP	0.868	**0.845**	0.809	0.881	**0.631**	[[604, 141], [363, 2706]]
1-GCN	0.847	0.774	0.658	0.891	0.529	[[494, 251], [334, 2735]]
ChebGCN	0.886	0.771	0.582	0.959	0.609	[[436, 309], [125, 2944]]
Bio attributes (16)						
SVM	0.858	0.704	0.452	0.956	0.492	[[339, 406], [135, 2934]]
RF	0.873	0.738	0.519	0.957	0.554	[[389, 356], [130, 2939]]
XGB	0.869	0.741	0.532	0.950	0.548	[[399, 346], [152, 2917]]
MLP	0.866	0.752	0.566	0.939	0.547	[[422, 323], [188, 2881]]
RUS	0.802	0.789	0.771	0.807	0.500	[[576, 169], [588, 2481]]
N2V+MLP	0.866	**0.847**	0.818	0.877	**0.632**	[[610, 135], [376, 2693]]
1-GCN	0.846	0.761	0.626	0.897	0.514	[[471, 274], [315, 2754]]
ChebGCN	0.881	0.763	0.572	0.954	0.587	[[430, 315], [140, 2929]]

Table 7 CS0 versus CS6-CS9 problem: performance of 5-fold stratified cross-validation under different node attribute selections (Part 2). The highest BA and MCC values for each set of attributes are in boldface

Method	Acc	BA	Sens	Spec	MCC	Confusion Matrix
GTEX attributes (89)						
SVM	0.820	0.561	0.135	0.987	0.257	[[98, 647], [39, 3030]]
RF	0.869	0.707	0.440	0.973	0.530	[[327, 418], [81, 2988]]
XGB	0.863	0.703	0.440	0.966	0.509	[[326, 419], [104, 2965]]
MLP	0.860	0.684	0.395	0.973	0.489	[[294, 451], [82, 2987]]
RUS	0.750	0.653	0.488	0.818	0.271	[[351, 394], [559, 2510]]
N2V+MLP	0.807	**0.798**	0.784	0.813	**0.513**	[[584, 161], [575, 2494]]
1-GCN	0.822	0.705	0.513	0.896	0.418	[[384, 361], [318, 2751]]
ChebGCN	0.847	0.652	0.333	0.971	0.424	[[251, 494], [90, 2979]]
net attributes (14)						
SVM	0.834	0.608	0.238	0.979	0.355	[[177, 568], [65, 3004]]
RF	0.862	0.709	0.460	0.959	0.506	[[346, 399], [126, 2943]]
XGB	0.857	0.697	0.437	0.957	0.485	[[330, 415], [130, 2939]]
MLP	0.852	0.695	0.441	0.949	0.472	[[334, 411], [154, 2915]]
RUS	0.748	0.720	0.680	0.760	0.378	[[515, 230], [730, 2339]]
N2V+MLP	0.846	**0.815**	0.767	0.864	**0.571**	[[573, 172], [417, 2652]]
1-GCN	0.832	0.751	0.621	0.882	0.487	[[465, 280], [360, 2709]]
ChebGCN	0.850	0.675	0.388	0.962	0.450	[[289, 456], [115, 2954]]
no attributes (0)						
N2V+MLP	0.812	0.797	0.510	0.937	0.516	[[577, 549], [168, 2520]]
1-GCN	0.671	0.500	0.200	0.800	0.000	[[126, 619], [636, 2433]]
ChebGCN	0.805	0.500	0.000	1.000	0.000	[[0, 745], [0, 3069]]

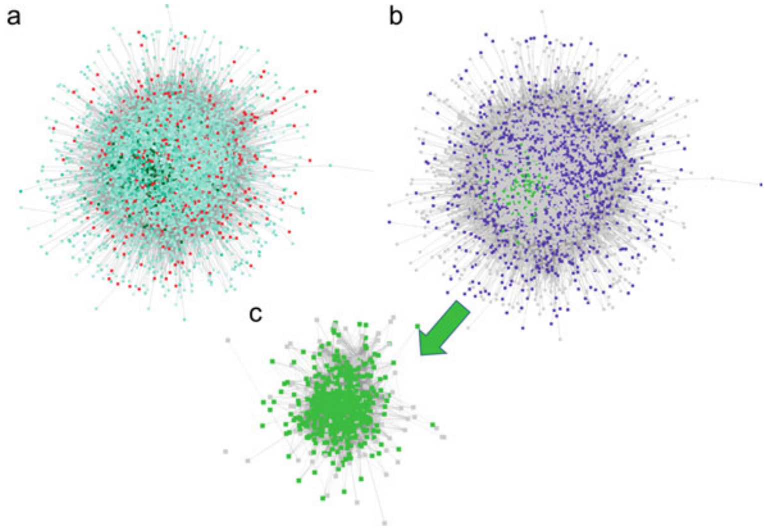


Fig. 3 Representation of the network and the correctly classified nodes. **a** PPI+MET network. Nodes are colored with scaling shades of green according to CS group. CS0 nodes are colored dark green and the box indicates their main localization. Red nodes are those belonging to the "ND" group, for which the label is missing; **b** PPI+MET network with colors indicating the correctly classified nodes according to the N2V+MLP method using Bio+GTEX attributes. TP CS0/E genes, corresponding to the positive class (E) nodes the model predicted correctly, are colored in light green, while TN CS6-9/NE genes, i.e., negative class nodes (NE) the model predicted correctly, are colored in violet. All the others are in grey; **c** Zoom on CS0 nodes with TPs colored in light green. The wrongly classified genes are in grey. The figure has been realized through Cytoscape 3.9.1 [59]

6 Discussion

In Tables 6 and 7, we separated the considered methods in two groups. ML methods of the top group act only on the node attribute matrix and, thus, they do not perform learning on the network data structure. The bottom group of methods includes two GNNs and the N2V+MLP pipeline: while GNNs naturally exploit graph representation learning techniques, the latter method first performs learning of node embeddings based on the network structure and then uses these embeddings jointly with the user-defined node attribute matrix as a new input for an MLP classifier. The highest balanced accuracy and MCC values for each set of attributes are in boldface.

The overall highest MCC and balanced accuracy are obtained by ChebGCN and N2V+MLP when using only biological attributes (Bio+GTEX), respectively. A representation of the nodes classified through N2V+MLP is provided by Fig. 3. By looking at the case when using all attributes (Bio+GTEX+net), we record results very close to those obtained when network attributes are not included. This outcome seems to prove that the user-defined network attributes do not contribute significantly to the discriminating capability of the trained model.

For all the other combinations of node attribute sets, the N2V+MLP pipeline outperforms the other methods in both balanced accuracy and MCC. This is even more evident when we consider the case in which no node attributes are used at all (see bottom of Table 7). In this case, only the N2V+MLP can be applied, since node2vec produces node embeddings by means of a training process on the PPI+MET network. Thus, even if the user does not provide node attributes, the node embeddings can be used as input to the MLP classifier with predictions that are acceptable in terms of balanced accuracy and MCC performance. On the other hand, when applying GNNs on a null node feature matrix, the ChebGCN performs a perfect null-classification, while the 1-GCN shows behavior close to null-classification.

It is worth noting how also the ChebGCN method, which learns on both node attributes and PPI+MET network interconnections, outperforms in most of the cases all the other ML methods acting only on the user-defined node descriptions.

In our interpretation of the experimental study, the PPI-MET network contains a significant amount of information which was not extracted by the user (i.e., the network attributes as computed in Sect. 3.3) and that can be learned by an embedding technique like node2vec or by the GNN learning scheme based on inter-node message passing. This additional information, implicitly encoded in the network, can be profitably exploited to better accomplish the task of gene essentiality detection.

It is difficult to compare our results with related works in literature. The first reason is that only a few works consider gene essentiality identification in the human organism. The second is that there is no uniformity in the metrics adopted to evaluate the results. We found only two works in the literature [27, 31] with which we can compare our best results (i.e., those obtained by N2V+MLP on Bio+GTEX attributes, giving Acc = 0.862, BA = 0.844, MCC = 0.627). On a problem with an imbalance ratio of the essentiality classes similar to our work (1:4), Dai et al. [27] report a similar MCC but balanced accuracy lower than our (BA = 0.783, MCC = 0.641). Nonetheless, the comparison cannot be discussed further since our approach to gene essentiality detection is human tissue-specific and it is based on metabolic+PPI network processing, while Dai et al. tackle the same task in the more general domain of human organisms and rely only on a PPI network. Another related work is EPGAT [31], although also in this case the domain is the human organism in general and only PPIs are considered in the processing. The problem with comparing our results with this work is that the performance is reported only in terms of a metric (AUC ROC which does not give enough details about the actual performance of the method. Indeed, as we have already discussed, the inherent imbalance of class samples in our opinion pushes the use of different metrics, like the MCC and the balanced accuracy, which were not used by those authors.

7 Conclusions

The definition of gene essentiality is a complex and challenging task, both for wet and *in silico* research. The evaluation of the computational approaches for identifying EGs must consider different aspects of the entire workflow, which go from the choice of the starting data to the assignment of labels, the attributes selection, and, finally, the learning methods. In this paper, we approached the problem by using a tissue-specific integrated network containing metabolic and physical interactions.

From the above-discussed results obtained by using embedding methods, we can state that the network topology, and thus how it is built, gives an important contribution with attributes automatically extracted by the embedding itself. They seem to be more discriminating than those usually considered a priori, mainly related to the centrality of the nodes. However, from the network representation (Fig. 2), it is clear that the EGs are located in a specific central area of the network and are highly connected both with the other genes and between each other. On the other hand, since the essentiality is a complex characteristic, we included a wide variety of biological attributes, both tissue-specific and generic, and, in particular, the expression profiles in the specific tissue and the related number of transcripts, which seem to be highly associated with the essentiality.

Regarding the labeling of the nodes, we did not exploit a pre-packaged list of EGs, but derived the labels from experimental scores obtained on kidney cell lines. The process was driven by assuming as "essential" the genes belonging to the CS0 score group containing the most negative values, while as "not-essential" the genes from the union of groups CSX-CS9, with X varying from 1 to 7. It seems evident from our results with the several CS groupings (Table 4) that genes can be partially identified as EGs/Non-EGs. Some are clearly distinguishable, but there are some genes in between with intermediate behavior. Although we tried to overcome the tissue heterogeneity issue, there is still a grade of heterogeneity, probably due to the cell lines and the experimental conditions, which lead to difficult labeling and classification of the intermediate groups. This particular aspect needs to be more deeply investigated through different approaches that are able to capture a trend more than a binary behaviour.

After this first investigation work, we intend to extend the kidney tissue-specific approach to other tissues to get the crucial differences and similarities among them. Other approaches will be further explored to handle the intermediate CS groups and their mixed behaviors. An additional, not secondary, goal concerns the performance improvement of the EGs identification by enriching the features adopted for nodes description via the exploration of compliant network embedding methods and enhancing robustness via adversarial learning techniques [69]. Finally, we intend to adopt alternative machine and deep learning models to enhance the validation phase.

Acknowledgements This work has been partially funded by the BiBiNet project (H35F21000430002) within POR-Lazio FESR 2014-2020. It was carried out also within the activities of the authors as members of the ICAR-CNR INdAM Research Unit and partially supported by the INdAM research project "Computational Intelligence methods for Digital Health". The work of Mario R. Guarracino was conducted within the framework of the Basic Research Program at the National Research University Higher School of Economics (HSE). Mario Manzo thanks Prof. Alfredo Petrosino for the guidance and supervision during the years of working together.

References

1. Chen, H., Zhang, Z., Jiang, S., Li, R., Li, W., Zhao, C., Hong, H., Huang, X., Li, H., Bo, X.: New insights on human essential genes based on integrated analysis and the construction of the hegiap web-based platform. Brief. Bioinform. **21**(4), 1397–1410 (2020)
2. Hasan, M.A., Lonardi, S.: DeeplyEssential: a deep neural network for predicting essential genes in microbes. BMC Bioinform. **21**(367) (2020). https://doi.org/10.1186/s12859-020-03688-y
3. Li, X., Li, W., Zeng, M., Zheng, R., Li, M.: Network-based methods for predicting essential genes or proteins: a survey. Brief. Bioinform. **21**(2), 566–583 (2019). https://doi.org/10.1093/bib/bbz017
4. Hutchison III, C.A., Chuang, R.-Y., Noskov, V.N., Assad-Garcia, N., Deerinck, T.J., Ellisman, M.H., Gill, J., Kannan, K., Karas, B.J., Ma, L., et al.: Design and synthesis of a minimal bacterial genome. Science 351(6280), 6253 (2016)
5. Dickerson, J.E., Zhu, A., Robertson, D.L., Hentges, K.E.: Defining the role of essential genes in human disease. PLoS ONE **6**(11), 27368 (2011)
6. Park, D., Park, J., Park, S.G., Park, T., Choi, S.S.: Analysis of human disease genes in the context of gene essentiality. Genomics **92**(6), 414–418 (2008)
7. Juhas, M., Eberl, L., Church, G.M.: Essential genes as antimicrobial targets and cornerstones of synthetic biology. Trends Biotechnol. **30**(11), 601–607 (2012)
8. Luo, L., Zheng, W., Chen, C., Sun, S.: Searching for essential genes and drug discovery in breast cancer and periodontitis via text mining and bioinformatics analysis. Anticancer Drugs **32**(10), 1038 (2021)
9. Chang, L., Ruiz, P., Ito, T., Sellers, W.R.: Targeting pan-essential genes in cancer: challenges and opportunities. Cancer Cell **39**(4), 466–479 (2021)
10. Wang, T., Birsoy, K., Hughes, N.W., Krupczak, K.M., Post, Y., Wei, J.J., Lander, E.S., Sabatini, D.M.: Identification and characterization of essential genes in the human genome. Science **350**(6264), 1096–1101 (2015)
11. Bartha, I., di Iulio, J., Venter, J.C., Telenti, A.: Human gene essentiality. Nat. Rev. Genet. **19**(1), 51–62 (2018). https://doi.org/10.1038/nrg.2017.75
12. Bartha, I., di Iulio, J., Venter, J.C., Telenti, A.: Human gene essentiality. Nat. Rev. Genet. **19**(1), 51–62 (2018)
13. Gurumayum, S., Jiang, P., Hao, X., Campos, T.L., Young, N.D., Korhonen, P.K., Gasser, R.B., Bork, P., Zhao, X.-M., He, L.-J., et al.: Ogee v3: Online gene essentiality database with increased coverage of organisms and human cell lines. Nucleic Acids Res. 49(D1), 998–1003 (2021)
14. Cowley, G.S., Weir, B.A., Vazquez, F., Tamayo, P., Scott, J.A., Rusin, S., East-Seletsky, A., Ali, L.D., Gerath, W.F., Pantel, S.E., et al.: Parallel genome-scale loss of function screens in 216 cancer cell lines for the identification of context-specific genetic dependencies. Sci. Data **1**(1), 1–12 (2014)

15. Aromolaran, O., Aromolaran, D., Isewon, I., Oyelade, J.: Machine learning approach to gene essentiality prediction: a review. Brief. Bioinform. **22**(5) (2021). https://doi.org/10.1093/bib/bbab128
16. Jeong, H., Mason, S.P., Barabási, A.-L., Oltvai, Z.N.: Lethality and centrality in protein networks. Nature **411**(6833), 41–42 (2001)
17. Liu, X., Hong, Z., Liu, J., Lin, Y., Rodríguez-Patón, A., Zou, Q., Zeng, X.: Computational methods for identifying the critical nodes in biological networks. Brief. Bioinform. **21**(2), 486–497 (2020)
18. Manipur, I., Giordano, M., Piccirillo, M., Parashuraman, S., Maddalena, L.: Community detection in protein-protein interaction networks and applications. IEEE/ACM Trans. Comput. Biol. Bioinform. 1 (2021). https://doi.org/10.1109/TCBB.2021.3138142
19. Granata, I., Manzo, M., Kusumastuti, A., Guarracino, M.R.: Learning from metabolic networks: current trends and future directions for precision medicine. Curr. Med. Chem. **28**(32), 6619–6653 (2021)
20. Dong, C., Jin, Y.-T., Hua, H.-L., Wen, Q.-F., Luo, S., Zheng, W.-X., Guo, F.-B.: Comprehensive review of the identification of essential genes using computational methods: focusing on feature implementation and assessment. Brief. Bioinform. **21**(1), 171–181 (2018). https://doi.org/10.1093/bib/bby116
21. Aromolaran, O., Beder, T., Oswald, M., Oyelade, J., Adebiyi, E., Koenig, R.: Essential gene prediction in drosophila melanogaster using machine learning approaches based on sequence and functional features. Comput. Struct. Biotechnol. J. **18**, 612–621 (2020). https://doi.org/10.1016/j.csbj.2020.02.022
22. Campos, T.L., Korhonen, P.K., Gasser, R.B., Young, N.D.: An evaluation of machine learning approaches for the prediction of essential genes in eukaryotes using protein sequence-derived features. Comput. Struct. Biotechnol. J. **17**, 785–796 (2019). https://doi.org/10.1016/j.csbj.2019.05.008
23. Cortes, C., Vapnik, V.: Support-vector networks. Mach. Learn. **20**(3), 273–297 (1995)
24. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001). https://doi.org/10.1023/A:1010933404324
25. Zeng, M., Li, M., Fei, Z., Wu, F.-X., Li, Y., Pan, Y., Wang, J.: A deep learning framework for identifying essential proteins by integrating multiple types of biological information. IEEE/ACM Trans. Comput. Biol. Bioinf. **18**(1), 296–305 (2021). https://doi.org/10.1109/TCBB.2019.2897679
26. Grover, A., Leskovec, J.: node2vec: scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '16, pp. 855–864. Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2939672.2939754
27. Dai, W., Chang, Q., Peng, W., Zhong, J., Li, Y.: Network embedding the protein-protein interaction network for human essential genes identification. Genes **11**(2), 153 (2020)
28. Wu, G., Feng, X., Stein, L.: A human functional protein interaction network and its application to cancer data analysis. Genome Biol. **11**(R53) (2010). https://doi.org/10.1186/gb-2010-11-5-r53
29. Li, T., Wernersson, R., Hansen, R., et al.: A scored human protein-protein interaction network to catalyze genomic interpretation. Nat. Methods **14**, 61–64 (2017). https://doi.org/10.1038/nmeth.4083
30. Rezaei, J., Zare Mirakabad, F., Marashi, S.-A., MirHassani, S.A.: The assessment of essential genes in the stability of PPI networks using critical node detection problem. AUT J. Math. Comput. **3**(1), 59–76 (2022)

31. Schapke, J., Tavares, A., Recamonde-Mendoza, M.: EPGAT: gene essentiality prediction with graph attention networks. IEEE/ACM Trans. Comput. Biol. Bioinf. **19**(3), 1615–1626 (2022). https://doi.org/10.1109/TCBB.2021.3054738
32. Zhang, X., Xiao, W., Xiao, W.: Deephe: accurately predicting human essential genes based on deep learning. PLoS Comput. Biol. **16**(9), 1008229 (2020)
33. Kuang, S., Wei, Y., Wang, L.: Expression-based prediction of human essential genes and candidate lncrnas in cancer cells. Bioinformatics **37**(3), 396–403 (2021)
34. Granata, I., Guarracino, M.R., Kalyagin, V.A., Maddalena, L., Manipur, I., Pardalos, P.M.: Supervised classification of metabolic networks. In: 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 2688–2693 (2018). https://doi.org/10.1109/BIBM.2018.8621500
35. Manipur, I., Granata, I., Maddalena, L., Guarracino, M.R.: Clustering analysis of tumor metabolic networks. BMC Bioinform. (2020). https://doi.org/10.1186/s12859-020-03564-9
36. Wang, H., Robinson, J.L., Kocabas, P., Gustafsson, J., Anton, M., Cholley, P.-E., Huang, S., Gobom, J., Svensson, T., Uhlen, M., et al.: Genome-scale metabolic network reconstruction of model animals as a platform for translational research. Proceed. Natil. Acad. Sci. **118**(30) (2021)
37. Kotlyar, M., Pastrello, C., Malik, Z., Jurisica, I.: Iid 2018 update: context-specific physical protein-protein interactions in human, model organisms and domesticated species. Nucleic Acids Res. **47**(D1), 581–589 (2019)
38. Uhlén, M., Fagerberg, L., Hallström, B.M., Lindskog, C., Oksvold, P., Mardinoglu, A., Sivertsson, Å., Kampf, C., Sjöstedt, E., Asplund, A., et al.: Tissue-based map of the human proteome. Science **347**(6220), 1260419 (2015)
39. Nandi, S., Subramanian, A., Sarkar, R.R.: An integrative machine learning strategy for improved prediction of essential genes in escherichia coli metabolism using flux-coupled features. Mol. BioSyst. **13**(8), 1584–1596 (2017)
40. Carithers, L.J., Ardlie, K., Barcus, M., Branton, P.A., Britton, A., Buia, S.A., Compton, C.C., DeLuca, D.S., Peter-Demchok, J., Gelfand, E.T., et al.: A novel approach to high-quality postmortem tissue procurement: the gtex project. Biopreservation Biobanking **13**(5), 311–319 (2015)
41. Tang, G., Cho, M., Wang, X.: Oncodb: an interactive online database for analysis of gene expression and viral infection in cancer. Nucleic Acids Res. **50**(D1), 1334–1339 (2022)
42. Durinck, S., Spellman, P.T., Birney, E., Huber, W.: Mapping identifiers for the integration of genomic datasets with the r/bioconductor package biomart. Nat. Protoc. **4**, 1184–1191 (2009)
43. Huang, D.W., Sherman, B.T., Lempicki, R.A.: Systematic and integrative analysis of large gene lists using david bioinformatics resources. Nat. Protoc. **4**(1), 44–57 (2009)
44. Huang, D.W., Sherman, B.T., Lempicki, R.A.: Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. Nucleic Acids Res. **37**(1), 1–13 (2009)
45. Hart, T., Chandrashekhar, M., Aregger, M., Steinhart, Z., Brown, K.R., MacLeod, G., Mis, M., Zimmermann, M., Fradet-Turcotte, A., Sun, S., et al.: High-resolution crispr screens reveal fitness genes and genotype-specific cancer liabilities. Cell **163**(6), 1515–1526 (2015)
46. Wheeler, D.L., Barrett, T., Benson, D.A., Bryant, S.H., Canese, K., Chetvernin, V., Church, D.M., DiCuccio, M., Edgar, R., Federhen, S., et al.: Database resources of the national center for biotechnology information. Nucleic Acids Res. **36**(suppl_1), 13–21 (2007)
47. Cacheiro, P., Muñoz-Fuentes, V., Murray, S.A., Dickinson, M.E., Bucan, M., Nutter, L.M., Peterson, K.A., Haselimashhadi, H., Flenniken, A.M., Morgan, H., et al.: Human and mouse essentiality screens as a resource for disease gene discovery. Nature Commun. **11**(1), 1–16 (2020)
pagebreak
48. Piñero, J., Ramírez-Anguita, J.M., Saüch-Pitarch, J., Ronzano, F., Centeno, E., Sanz, F., Furlong, L.I.: The disgenet knowledge platform for disease genomics: 2019 update. Nucleic Acids Res. **48**(D1), 845–855 (2020)

49. Granata, I., Guarracino, M.R., Maddalena, L., Manipur, I.: Network distances for weighted digraphs. In: Kochetov, Y., Bykadorov, I., Gruzdeva, T. (eds.) Mathematical Optimization Theory and Operations Research. CCIS, vol. 1275, pp. 389–408. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58657-7_31
50. Rasti, S., Vogiatzis, C.: A survey of computational methods in protein-protein interaction networks. Ann. Oper. Res. **276**(1), 35–87 (2019). https://doi.org/10.1007/s10479-018-2956-2
51. Bonacich, P.: Factoring and weighting approaches to status scores and clique identification. The Journal of Mathematical Sociology **2**(1), 113–120 (1972). https://doi.org/10.1080/0022250X.1972.9989806
52. Granata, I., Guarracino, M.R., Kalyagin, V.A., Maddalena, L., Manipur, I., Pardalos, P.M.: Model simplification for supervised classification of metabolic networks. Ann. Math. Artif. Intell. **88**, 91–104 (2020). https://doi.org/10.1007/s10472-019-09640-y
53. Barrat, A., Barthélemy, M., Pastor-Satorras, R., Vespignani, A.: The architecture of complex weighted networks. Proc. Natl. Acad. Sci. **101**(11), 3747–3752 (2004). https://doi.org/10.1073/pnas.0400087101
54. Csardi, G., Nepusz, T.: The igraph software package for complex network research. Inter. J. Complex Syst. 1695 (2006)
55. Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., Alon, U.: Network motifs: simple building blocks of complex networks. Science **298**(5594), 824–827 (2002)
56. Sporns, O., Kötter, R., Friston, K.J.: Motifs in brain networks. PLoS Biol. **2**(11), 369 (2004)
57. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. J. ACM **46**(5), 604–632 (1999). https://doi.org/10.1145/324133.324140
58. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. Comput. Netw. ISDN Syst. **30**(1), 107–117 (1998). https://doi.org/10.1016/S0169-7552(98)00110-X. Proceedings of the Seventh International World Wide Web Conference
59. Shannon, P., Markiel, A., Ozier, O., Baliga, N.S., Wang, J.T., Ramage, D., Amin, N., Schwikowski, B., Ideker, T.: Cytoscape: a software environment for integrated models of biomolecular interaction networks. Genome Res. **13**(11), 2498–2504 (2003)
60. Seiffert, C., Khoshgoftaar, T.M., Van Hulse, J., Napolitano, A.: Rusboost: Improving classification performance when training data is skewed. In: 2008 19th International Conference on Pattern Recognition, pp. 1–4 (2008). IEEE
61. Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Philip, S.Y.: A comprehensive survey on graph neural networks. IEEE Trans. Neural Netw. Learn. Syst. **32**(1), 4–24 (2020)
62. Yue, X., Wang, Z., Huang, J., Parthasarathy, S., Moosavinasab, S., Huang, Y., Lin, S.M., Zhang, W., Zhang, P., Sun, H.: Graph embedding on biomedical networks: methods, applications and evaluations. Bioinformatics **36**(4), 1241–1251 (2020)
63. Nelson, W., Zitnik, M., Wang, B., Leskovec, J., Goldenberg, A., Sharan, R.: To embed or not: network embedding as a paradigm in computational biology. Front. Genet. **10**, 381 (2019)
64. Manipur, I., Manzo, M., Granata, I., Giordano, M., Maddalena, L., Guarracino, M.R.: Netpro2vec: a graph embedding framework for biomedical applications. IEEE/ACM Trans. Comput. Biol. Bioinf. **19**(2), 729–740 (2022). https://doi.org/10.1109/TCBB.2021.3078089
65. Maddalena, L., Manipur, I., Manzo, M., Guarracino, M.R.: In: Mondaini, R.P. (ed.) On Whole-Graph Embedding Techniques, pp. 115–131. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-73241-7_8
66. Chen, T., Guestrin, C.: Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '16, pp. 785–794. Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2939672.2939785
67. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations (ICLR) (2017)

68. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In: Proceedings of the 30th International Conference on Neural Information Processing Systems. NIPS'16, pp. 3844–3852. Curran Associates Inc., Red Hook, NY, USA (2016)
69. Manzo, M., Giordano, M., Maddalena, L., Guarracino, M.R.: Performance evaluation of adversarial attacks on whole-graph embedding models. In: Simos, D.E., Pardalos, P.M., Kotsireas, I.S. (eds.) Learning and Intelligent Optimization. Lecture Notes in Computer Science, vol. 12931, pp. 219–236. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-92121-7_19

Acoustic Analysis for Vocal Fold Assessment—Challenges, Trends, and Opportunities

Monika Danilovaitė and Gintautas Tamulevičius

Abstract The goal of this study was a review of trends in non-invasive vocal fold assessment to identify the significance of acoustic analysis within the scope of proposed methods. A review protocol for selected relevant studies was developed using systematic review guidelines. A classification scheme was applied to process the selected relevant study set, data were extracted and mapped in a systematic map. A systematic map was used to synthesize data for a quantitative summary of the main research question. A tabulated summary was created to summarize supporting topics. Results show that non-invasive vocal fold assessment is influenced by general computer science trends. Machine learning techniques dominate studies and publications, i.e., 51% of the set used at least one method to detect and classify vocal fold pathologies.

1 Introduction

The complex mechanism of voice production has evolved in modern humans, and its complexity makes human communication a unique phenomenon [1]. For this reason, voice research has become a complex and multifaceted domain. Voice training, clinical voice assessment, and various applications of speech technology are examples of voice research topics.

Voice production is controlled by laryngeal and respiratory muscles, but vibrational vocal fold patterns are what characterizes the voice [2]. Therefore, the voice research domain is highly focused on vocal fold analysis. A significant part of the population is affected by voice pathologies—between 3 and 9% of the USA population is affected by vocal fold pathologies [3, 4]. Thus, there exists a clinical demand

M. Danilovaitė (✉) · G. Tamulevičius
Institute of Data Science and Digital Technologies, Vilnius University, Vilnius, Lithuania
e-mail: monika.danilovaite@mif.stud.vu.lt

G. Tamulevičius
e-mail: gintautas.tamulevicius@mif.vu.lt

G. Dzemyda et al. (eds.), *Data Science in Applications*,
Studies in Computational Intelligence 1084,
https://doi.org/10.1007/978-3-031-24453-7_8

for fast, reliable non-invasive vocal fold assessment. A promising basis for such analysis methods is non-invasive acoustic analysis.

Between 1999 and 2022 a total of 30283 studies on voice pathology were published (number found on ScienceDirect). With such volumes of information, a comprehensive review is complicated and time-consuming. It is increasingly difficult to ensure the comprehensiveness of systematic reviews and narrow research scope may lead to missed challenges, trends, and opportunities in the focus domain. This is especially notable in voice research, as this domain is multifaceted and largely multidisciplinary (e.g., voice production involves physiology, vocal fold vibration, and acoustics as a whole) [5].

The goal of this study is to identify challenges, trends, and opportunities in non-invasive vocal fold assessment via acoustic analysis. To realize this goal, a Systematic Mapping Study (SMS) was conducted to quantitively assess techniques, research objects, and tasks (see Sect. 4 for review protocol). The analyzed set was acquired from the Web of Science reference database. Set includes topics such as image processing, classification via machine learning algorithms, and model optimization. This allows to identify a broader scope of vocal fold assessment and place acoustic analysis within this scope.

This study has the following structure: Sect. 2 presents an analysis of related studies. Section 3 presents the dimensions of conducted research and justification. Section 4 describes applied SMS. Section 5 shows the results of conducted SMS. Section 6 presents a discussion on the reliability, generalizability, validity, and bias of conducted research. Finally, Sect. 7 concludes conducted study.

2 Related Studies

The acoustic analysis-based pathological voice analysis concept was formed in the 1970s [6, 7]. The first steps in voice research were taken by individual effort—exploratory research of voice, pathological voice qualities, and signal modeling. However, in the 1990s, the focus was shifted to acoustic analysis based on automatic voice assessment. Increasingly complex solutions, based on complex acoustic feature sets, statistics, information science methods, and artificial neural networks were proposed [8]. In the 21st century Voice research became multidisciplinary science—signal and image processing methods are combined with big data and existing research on voice pathology, and vocal fold physiology [9].

This section presents related Systematic Literature Reviews (SLR) and SMS involving voice, voice pathologies, and computer-assisted analysis tools and methods. As noted, speech signal features are influenced by physiology, vocal fold vibration, vocal tract configuration, etc. Thus, selected reviews include topics like computer-assisted speech therapy systems and computer-assisted voice condition analysis. This range of topics was chosen to represent novelties in non-invasive vocal fold assessment.

Computer-based systems are a prominent topic in speech therapy (e.g. systems for personalized therapy, systems for disordered speech enhancement, pathology detection, etc.). SMS with different proposals for computer-based speech therapy was selected [10]. Authors provide quantitative evaluation of Situational Awareness criteria realization and methods for situational assessment (depending on the system, situational assessment can mean identification of disordered speech, identification of pathology, etc.). Study shows that Hidden Markov Models (HMMs) are used in tasks, such as speech temporal modeling and decoding. Authors note that HMM technique requires model fitting for application, thus systems' adaptability could be compromised. Mel-Frequency Cepstral Coefficients (MFCCs) for speech processing are widely used in pathology classification tasks. MFCC analysis provides robustness against signal noise and allows frequency-based analysis. Other speech analysis techniques, such as Linear Predictive Coding (LPC) or Autoregressive (AR) modeling are less popular as they are susceptible to signal noise. However, studies show that LPC allows for a more accurate estimation of individual vocal properties, which is important in voice quality assessment [11]. Authors note that sophisticated machine learning models (e.g., Gaussian Mixture Model (GMM)) are costly when memory and computational requirements are considered. Thus, classification models such as Support Vector Machine (SVM) are more popular, as their implementation is not resource costly, shows high classification accuracy, and is robust to missing speech segments [12].

A review of Automatic Voice Condition Analysis (AVCA) systems was presented in [13]. Authors emphasize that speech signal provides a simple and inexpensive way to perform a non-invasive diagnosis procedure. Authors found that AVCA systems are applicable in voice pathology detection (e.g. nodules, polyps, or dysphonia), but there is an interest in other pathologies that impact speech signal indirectly (e.g., Alzheimer's disease). Overview provides introductory concepts, such as physiological voice pathology phenomena relationship with perceptual features of voice, techniques, and methods for automatic voice condition analysis and system performance variability factors. Authors note the following trends regarding AVCA systems:

- Limited speech corpus used in research.
- Extralinguistic (e.g. sex, age) and paralinguistic (e.g., accent) factors are not evaluated in AVCA systems.
- Analysis of pathology's mechanics and their connection to vocal quality descriptors (e.g. better understanding of Parkinson's and Alzheimer's disorders creates a need for novel features to identify dysphonic conditions).
- Popularity of machine learning classifiers, such as SVM, GMM. However, it is noted that these techniques are sensitive to data and its properties.
- Insufficient accuracy validation both in an experimental and clinical setting.

Table 1 Research sub-questions and motivation

Research sub-questions	Motivation
RQ1: What tasks are performed for vocal fold assessment?	To examine how research objects are analyzed
RQ2: What subjects are explored in vocal fold assessment?	To determine volume of what is being researched in non-invasive vocal folds assessment
RQ3: What research methods are used in vocal fold assessment?	To determine how non-invasive vocal fold assessment is being researched

3 Dimensions of Research

Vocal fold assessment is a multifaceted problem (Sect. 2). Systematic Literature Reviews (SLR) and Systematic Mapping Studies (SMS) analyze specific aspects of voice research (e.g. what challenges are faced in AVCA systems). However, because of narrow research focus it is difficult to distinguish trends within the research of non-invasive vocal fold assessment. Thus, based on the identified problem, the main research question (**RQ**) is ”What are the trends in non-invasive vocal fold assessment?”.

Main **RQ** was split into sub-questions (see Table 1) to assess dimensions of research, such as applied techniques, used methods, and finally, the nature of data used. A wide variety of methods and techniques are proposed for non-invasive vocal fold assessment. Thus, sub-question **RQ1** was defined to compare techniques by volume in the analyzed set (e.g., how often machine-learning classification algorithms are used, the segmentation process is applied, etc.). Data for non-invasive vocal fold assessment is extracted from various sources (e.g., electroglottograph (EGG), using voice signal recording, and others). To evaluate the range and spread of the acoustic data-based techniques in the set, a sub-question **RQ2** was formed. Finally, to quantify the volume of used methods in non-invasive vocal fold assessment, sub-question **RQ3** was defined. Sub-question will help to identify and highlight trends in research design and approach.

4 Research Method

Considering the broad scope of main **RQ**, the SMS method was selected and applied. SMS is a literature analysis method suitable for questions that are more general in nature [17]. The method allows for managing biases because it requires application of a rigorous and clearly defined review protocol [14, 15]. This method is based on document search and review protocol techniques and is used to analyze the main research question area, topics with sufficient studies, and topics where more primary studies are needed [16]. Thus, to summarize current trends in vocal folds assessment

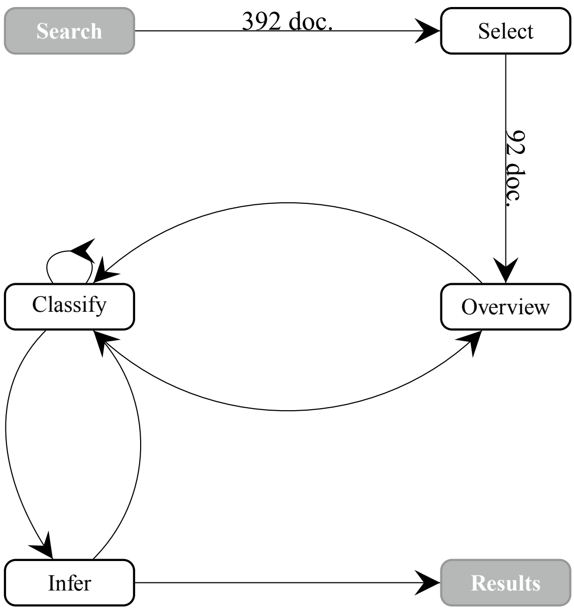


Fig. 1 Method flowchart

(**RQ**) and defined research sub-questions (**RQ1**, **RQ2**, **RQ3**), protocol was prepared and implemented [17, 18]. See Fig. 1 for the method flowchart.

Search

According to recommendations, SMS should be conducted using multiple reference databases [19]. This approach allows the covering of unique citations in specific reference databases. In addition, reference databases contain records of "grey literature"—e.g., theses, books, and reports [20]. These documents require in-depth analysis and are hard to process with protocol, thus valuable insights might be lost. Sources such as unpublished studies, sites, and research websites were excluded from the scope of this study. As a result, a restricted approach was used [21]. Based on this approach, reference databases that include human-curated collection were selected and analyzed to choose the most suitable. Web of Science reference database was selected [22]. Table 2 provides selection criteria for the initial selected relevant study set. Criteria was adapted according to distinct document types. To represent novelties in research, proceedings papers were selected from a 5-year period (2016–2020). Long-term trends were identified via articles and reviews; thus, these document types were selected from a 20-year period (2000–2020).

Records from the Web of Science reference database were downloaded in July of 2021. The downloaded set contained records of 392 documents. Search by keywords causes low sensitivity scores (e.g. Web of Science reference database allowed for 14% sensitivity within their experimental setting) [23]. For this reason, studies that are not within the scope of computer science were identified and excluded from the selected relevant study set.

Table 2 Set selection criteria

Criteria	Value
Reference database	Web of Science
Search keyword	Vocal folds state assessment OR vocal folds state
Searched in	Topic
Document type	Article, review, proceedings paper
Category	Computer science, medical informatics, multidisplinary sciences
Citation index	SCI-EXPANDED, CPCI-S
Publication year	2000–2020 (if article or review), 2016–2020 (if proceedings paper)

Table 3 Inclusion and exclusion criteria

Id	Criteria
IC1	Study was cited at least 1 time per year OR published on 2021 (the year of this study)
EC1	Study was published before 2010
EC2	Study is duplicate
EC3	Study is not in English
EC4	Study is not within scope of computer science
EC5	Study focuses on human laryngeal apparatus

Table 4 Quality criteria

Id	Criteria	Motivation
Q1	Structure	Study has clear structure—defined hypothesis, research design description, results (quantified or summarized) and discussion
Q2	Validity	Selected research design in suitable for defined hypothesis and consistent with related studies
Q3	Reliability	Study provides information about used result measurement and evaluation method and results are consistent with related studies

Select

To evaluate the set, selection criteria were defined. Defined selection criteria are inclusion and exclusion, and quality criteria. Inclusion and exclusion criteria were defined to extract studies relevant to non-invasive vocal fold assessment which is within the scope of the computer science domain. As a result, these criteria are very general and typical in SMS [24]. Inclusion and exclusion criteria are defined in Table 3. Quality criteria were defined and applied according to [25, 26]. Criteria definition and motivation are provided in Table 4.

Some authors recommend assessing defined criteria with scoring (e.g. evaluation of positive engagement and motivation) [27]. However, the goal of this study is to identify the significance of acoustic analysis in non-invasive vocal fold analysis. For this reason, scoring techniques, which would be applicable in the quality assessment were not used—the focus was placed on taxonomy analysis.

Selection was performed by First Author. Defined criteria (Tables 3 and 4) were applied in a two-step procedure, and 92 studies were extracted:

- **1st step**: Criteria from Table 3 were applied—a study was reviewed by reading title, keywords, an abstract, and introductory section of a study (partial reading).
- **2nd step**: Criteria from Table 4 were applied—the entire study was analyzed. Study sections defining scientific basis, research design, and validation techniques were identified and analyzed.

Table 5 Taxonomy classes

RQ	Category	Supplementary information
RQ1—Tasks	Coding	Applied instruments, if applicable—population
	Classification	Segmentation (e.g. healthy vs. patients, healthy vs. multiple pathologies)
	Detection	
	Enhancement	
	Evaluation	
	Optimization	
	Overview	
	Prediction	
	Segmentation	
	Simulation	
	Visualization	
RQ2—Subjects	Features	Data, datasets, nature of data (e.g. recordings of sustained vowels, images, videos), data segments (e.g. gender, age)
	Glottal area	
	Glottal dynamics	
	Methodology	
RQ3—Method	Experiment	Applied evaluation methods (e.g. ROC, F-score)
	Modeling	
	Systematic review	

Overview and Classification

Data were extracted in the 3rd step. Data relevant to **RQ1**, **RQ2**, **RQ3** (supplementary data and initial class assignment) was documented. An initial taxonomy scheme was created when analyzing extracted data. The taxonomy scheme was reviewed and updated to represent classes that possess content relation between members. As an example, a study on airflow modeling was grouped with a study on glottal closure—both studies involve the concept of *Glottal dynamics* as both are focused on vocal fold movement (in one case airflow journey through vocal folds, other—the gap between vocal folds) [28, 29]. If ambiguities were detected supplementary information was used. If matches were found and a category was already assigned, this category was assigned to the study in question. If no matches were found, a new category was created. The final taxonomy scheme and supplementary information summary are provided in Table 5.

5 Inference and Results of the SMS

To evaluate non-invasive vocal fold assessment via acoustic analysis, SMS was conducted. The analysis results are presented in bubble plots (Figs. 2 and 3) representing intersections between **RQ1**, **RQ2** and **RQ3**. In addition, a bubble plot representing the most popular topics in the set was presented (Fig. 3). Results have been analyzed and summarized from three perspectives: trends, challenges, and opportunities.

Trends

Figure 2 shows relationship between **RQ3** and **RQ2** (research method and research subject) and relationship between **RQ2** and **RQ1** (research subject and research task). Relationship between **RQ1** and **RQ3** are presented in Fig. 3.

39% of selected relevant studies in the set analyzed feature-based classification task. This was the most common research topic in vocal fold assessment. Another common topic was glottal dynamics simulation (18% of the set). The most popular research technique was the Experiment (82% of the set). The set included Systematic Review Studies (SLRs) (4% of the set), covering a wide variety of topics (techniques on simulation, classification, visualization, and prediction—see Fig. 3). Studies provide an overview of laryngeal pathology analysis with imaging techniques, evaluation of available silent speech interfaces, tomography techniques and application in the analysis of various tissues, and, finally an overview of vocal fold anatomy, physiology, voiced speech modeling, and model application [30–33]. Figure 3 shows that the Experimental method was applied in all taxonomy classes of **RQ1**—Tasks. Modeling taxonomy class intersects with Simulation, Visualization, and Optimization classes. This shows that vocal fold modeling is focused on the accurate representation of the vocal fold vibrational process [34, 35].

55% of the analyzed studies were feature-based. All analyzed feature systems can be grouped to:

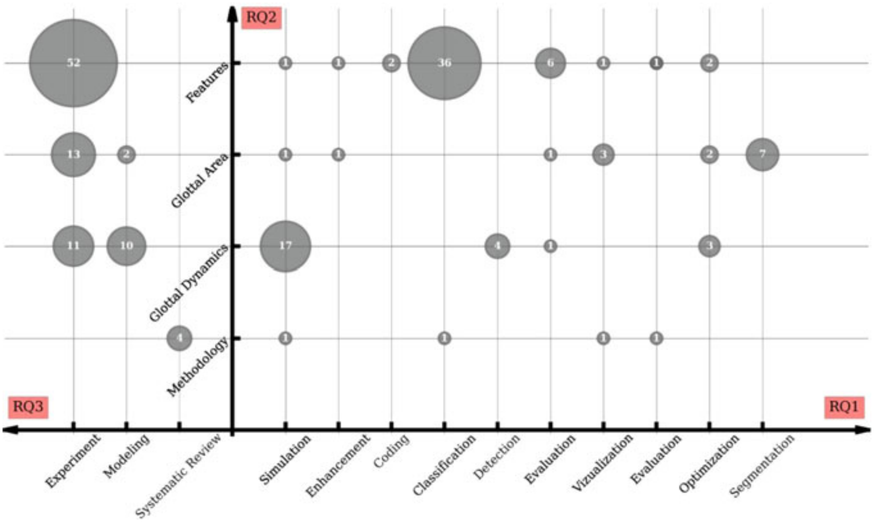


Fig. 2 Mapping of **RQ1**, **RQ2** and **RQ3** with regard to research object

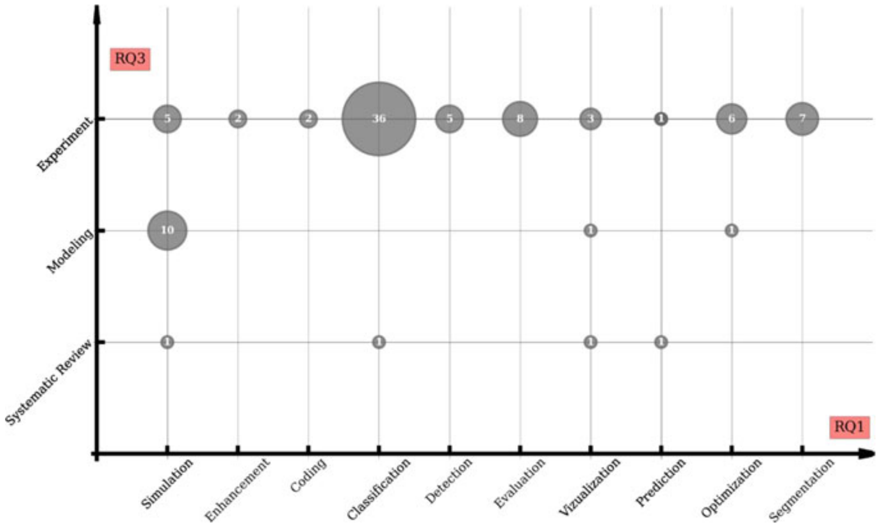


Fig. 3 Mapping of RQ1, RQ2 and RQ3 with regard to research object

1. Noise features (e.g. Harmonic to Noise Ratio (HNR), Cepstrum based Harmonics to Noise ratio (CHNR), Voice Turbulence Index (VTI), Soft Phonation Index (SPI)).
2. Stability and periodicity (e.g., Jitter, Shimmer, Relative Average Perturbation (RAP))
3. Spectral and Cepstral features (e.g., MFCC, LPC, formants, Fundamental frequency)
4. Nonlinear features (e.g., Locally-Linear Embedding (LLE), Lem-Ziv complexity (LZC)).

The list shows that a wide variety of features are used in vocal fold assessment.

To assess trend changes, a plot with **RQ1** and **RQ2** pairs aligned in time (2010–2021) were given in Fig. 4. The most popular topic in non-invasive vocal fold assessment is feature-based classification (most popular topic in 2010, 2014–2019, and 2021). To assess general trends within this topic, the set was filtered to extract studies published in 2010, 2014–2019, and 2021. The obtained set was filtered again to extract taxonomy classes *Features* and *Classification*. The extracted results were compared with the supplementary information (see Table 5). Extracted trends within this topic are provided in Table 6.

It was found that Massachusetts Eye and Ear Infirmary Voice Disorder Database (MEEI) is the most popular corpus used to assess vocal folds (29% of studies within filtered set used this database) [36].

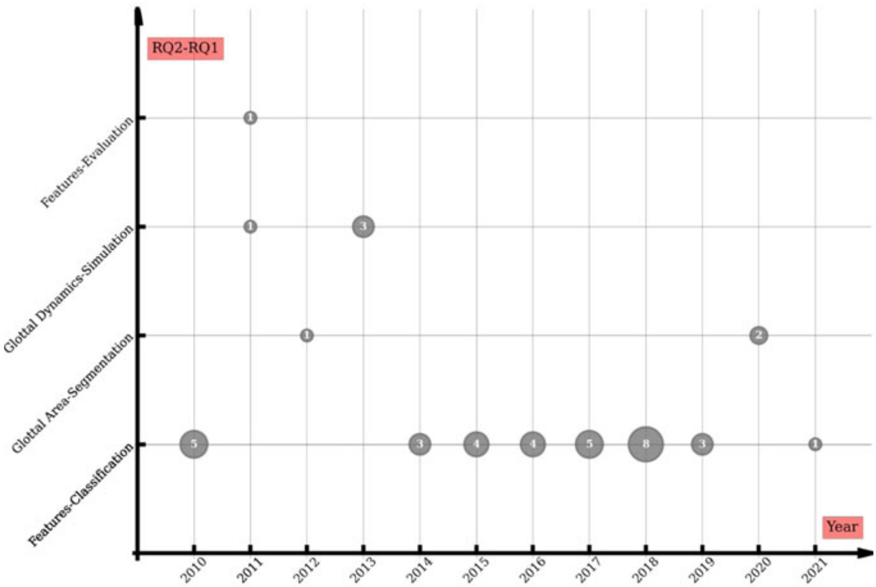


Fig. 4 Mapping of RQ1, RQ2 and RQ3 with regard to research object

Table 6 Supplementary information summary for most popular topic

RQ1 and RQ2	Insight	Percentage of sample (%)
Features—Classification	Task type: binary classification	71
	Task instrument: support vector machine	74
	Task evaluation: classification error measurements	97
	Subject: acoustic features	81
	Data: MEEI database	29

The database contains a wide variety of pathological speech samples and sustained vowel utterances (53 normal voices and 657 pathological) segmented by age and sex. Samples of pathological records include diagnoses such as psychogenic pathologies (e.g. depression), organic (e.g., physiological pathologies such as vocal nodules—these influence vocal folds the most—and functional pathologies) [37]. However, recordings were captured in different conditions (e.g. different sampling rates) and classes are unbalanced. Some studies suggest that these irregularities may impact classification performance [38, 39]. Researchers note that the MEEI database is not suitable for the normal-versus-dysphonic classification task, as classes are perfectly separable—models' generalizability is impacted [38, 39]. The authors suggest using combinations of different voice databases for model training and validation—e.g., Saarbrucken Voice Database (SVD) [40]. SVD database contains 869 healthy and 1356 pathologic voice records and allows segmentation by age and sex, as well as specific pathology [40]. Considering the size of SVD, it could help to extend the experimental verification of proposed models and obtain more robust and reliable results.

Usage of a dataset like MEEI (unbalanced, well-separated normal and pathological voices) in non-invasive vocal fold assessment places a limitation on research possibility. 71% of filtered set preferred binary classification paradigm. In this case, proposed techniques are focused on healthy-vs-dysphonic classification and rarely focus on specific pathology identification or severity assessment.

Alternative paradigms exist. In [41] classification into nodular, diffuse, and healthy was conducted. Authors propose to use combined data: vocal fold image features (e.g., geometrical), acoustic voice signal features (e.g., Fundamental frequency, MFCCs), and extralinguistic information (e.g., age, smoking) with a questionnaire to assess Voice Handicap Index (VHI) [41, 42]. This improves robustness and effectiveness in solving multi-class classification task.

Analysis of supplementary information has shown that 19% of studies in filtered set use features extracted from vocal fold images or videos. 81% of studies within the filtered set use various acoustic features (e.g., HNR, jitter, shimmer, MFCC) and nonlinearity measurements (e.g., LLE). Some trends were observed within this group. MFCC features were used in 19% of the filtered set. However, these features are combined with periodicity and stability features as well as noise measures. The general approach in these studies is the creation of large feature sets (e.g., 6506 features per utterance and PCA technique for dimensionality reduction) [43].

The most popular classification algorithm was found to be the Support Vector Machine (SVM) (71% of the filtered set). The reason could be relatively low complexity and high classification efficiency (considering accuracy, and training). However, there also exist studies using neural networks (e.g., Probabilistic Neural Network (PNN), General Regression Neural Network (GRNN), Convolutional Neural Network (CNN))—19% of the filtered set applied these methods. It was found that more sophisticated classification algorithms are applied to image features.

Finally, the most popular validation method was classification error measurement (97% of the filtered set). Classification Accuracy, Sensitivity, and Specificity are prevalent. Cross-validation is used to resample and evaluate machine learning models. However, some studies compare results with expert diagnoses [44]. As the rule comparison is correlation-based, however, there is a lack of studies where proposed methods would be evaluated with data from other datasets.

Challenges

By analyzing obtained results, challenges in vocal fold assessment were identified. The first identified challenge is the lack of labeled data. Most topical research studies (feature-based classification) are run using the same MEEI database (29% of the filtered set). Because of limited labeled data, proposed methods cannot be evaluated quantitatively to check for stability and reproducibility.

The lack of labeled data leads to another challenge: the limited ability to assess individual voice features. Vocal folds are shown to change with age and voices are sexually dimorphic [45, 46]. Besides, vocal folds are impacted by factors such as climate as well as many others (e.g., climate, smoking) [41, 47]. For this reason, the classification of healthy-vs-pathologic voice becomes complex, as these populations may overlap (because of limited datasets there is a lack of research on acceptable ranges for healthy and pathologic voice acoustic features).

The final identified challenge is the relationship between subjective and objective assessment of the vocal fold state. Most popular voice quality assessment methods are based on subjective perceptual evaluation [48]. It is unclear whether the subjective assessment can be converted to objective (computer-based non-invasive) and vice versa. However, research results show that there exists a relation between physiological features of voice and voice type (e.g., thick vocal folds are linked with vocal fry) as well as between acoustic features and voice type (e.g., turbulent noise is linked with breathy voices) [49]. This relation should be analyzed as it allows insight to voice pathology physiological mechanism—this information can be used to improve both mathematical models and parametric models.

Opportunities

Identified trends and challenges were analyzed and opportunities were identified. The first opportunity is in a multidisciplinary approach to vocal fold assessment. The speech signal is a complex physiological process, influenced by various factors. Thus, a knowledge combination of signal production, acoustic analysis, and signal modeling techniques could enhance vocal fold assessment techniques (e.g., experts could provide insight into the physiology of specific pathologies and voice researchers could model these cases).

Another opportunity was observed in parametric model usage. Currently, the most common topic is feature-based pathology classification, but no clear trends in parametric modeling were observed. If relation and causality between physiological process and perceptual features could be established, the objective features could be used to model specific pathologies (e.g., patients with Parkinson's disease were evaluated using perturbation measures, energy content, nonlinear dynamics) [50–52].

The final opportunity would be the detailed vocal fold assessment and pathology identification. Current research is focused on vocal fold state assessment—models are used to identify healthy and pathologic voices. However, research on pathology identification, estimation of disorder dynamics, and assessment of voice quality and its dynamics will be of great interest and a highly relevant research topic.

6 Discussion

This section presents a discussion on the reliability, generalizability, validity, and bias of conducted study and obtained results.

Descriptive Validity

Descriptive validity is defined as "threats to the ability to capture and accurately represent the observations made" [53]. SMS is mostly used to analyze software engineering topics. There exist SMS and SLR protocols tailored specifically to software engineering topic analysis [17–19]. In addition, examples of selection criteria, taxonomies, and their application in studies done by other authors were found [24, 54–56]. For this reason, actions to ensure Descriptive validity were taken. For our study, the protocol was developed based on example studies and recommendations [17, 18, 24]. However, bias (citation, selection, observer) is possible: there are no studies where SMS would be used to quantify and classify methods, subjects, and tasks of a broad subject (e.g. non-invasive vocal fold assessment). Various strategies were used to minimize biases. The reasoning for these strategies, as well as bias description, are analyzed further in this section.

Bias

During Research method definition phase (Sect. 3) following biases were identifies:

1. Citation bias—tendency to use studies published by known sources [57]
2. Selection bias—error in choosing participants (in this case, appropriate studies to analyze) [58]
3. Observer bias—error in observing or recording information [59].

Citation bias was introduced in Sect. 4 (paragraph Search). For this study, only one reference database was selected (Web of Science). Different reference databases overlap but possess some differences (e.g., 14.1% of Engineering and Computer science citations are unique to Scopus, while 3.1% are unique to Web of Science) [22]. There exist studies on the relationship between selected reference databases and comprehensiveness of SLR (e.g., the effect of specific reference databases on systematic review results in research on maternal morbidity and mortality) [60]. However, studies investigating this relation in the computer science domain were not found. The observation was made that SMS involving software, software design, and approaches may start off with a large set collected from various reference databases. After the selection process, only a small part of the initial set is retained for detail analysis (e.g. in [24] 236 studies were selected out of the initial 3409, in [56] only 44 studies out of 1206). It can be inferred that studies unique to reference databases were not a significant part of the final set.

Selection bias was introduced in Sect. 4 (paragraph Select). Because selection was done by the first author, incorrect criteria interpretation could lead to unreliable results. To minimize this risk, defined criteria (Tables 3 and 4) were discussed, analyzed, and approved by all study authors. In addition, criteria were applied in two steps. Firstly, Inclusion and Exclusion criteria were applied, and after Quality criteria were applied. If it was unclear if the study should be included in the selected study set, a decision was made using the context of previously reviewed studies. Observer bias (introduced in Sect. 4, paragraphs Overview, Classify) is caused by primary author bias when classifying concepts to produce taxonomy. To mitigate this risk, defined **RQ**s were conceptualized to be strictly categorical questions (what? how? why? are vocal folds assessed). This guaranteed that data relevant to **RQ**s was extracted (e.g., used validation tools, used instruments) and no subjective evaluation regarding quality was done. If consensus regarding applicable taxonomy class was found, a study was compared to other studies which analyze similar subjects, tasks, or methods, thus ensuring descriptive validity.

Reliability of Results

The reliability of the study was ensured by multiple evaluations during the data inference and classification step (see Fig. 1). Results and conclusions were verified against a related study. Taxonomy scheme creation was an iterative process, which continued through the entire study. Two cases of related studies were selected and summarized in Table 7. These studies analyze acoustic features and classification, these topics were identified as the most popular (see Fig. 4).

As seen in Table 7, all study findings were confirmed with exception of "Markov models" and "paralinguistic and extralinguistic factors". Identification of vocal fold pathologies factor in some extralinguistic features (e.g., sex) and paralinguistic features (a prominent part of research uses sustained vowels instead of continuous speech, thus minimizing the impact of accent and language specifics). However, factors such as age are largely ignored, unless the study focuses on a specific population (e.g., children's voices). HMMs were not used in any studies included in the analyzed study set. However, 2 studies of HMMs usage were found in excluded studies [61, 62]. Studies were excluded for not meeting the inclusion criteria IC1 (see Table 3). This means that the application of HMMs in vocal fold assessment was not very

Table 7 Summary of related study

Study	Finding	Findings of this study	Confirmation
[11]	Markov Models	No studies found	No
	MFCC features	7 studies used these features	Yes
	SVM classifiers	26 studies used this method	Yes
[13]	Machine learning	47 studies applied at least one machine learning method	Yes
	Limited speech corpus	15 use MEEI, 4 use Saarbrucken	Yes
	Paralinguistic and extralinguistic factors	21 studies segmented according to sex, in acoustic feature-based analysis there were no studies where population sample was segmented according to age	Partial
	Insufficient accuracy validation	5 studies compared experimental results with other studies directly	Yes

popular but can be applied to decode speech (application in non-invasive speech therapy).

Generalizability

Generalizability is defined as two-fold [63, 64]:

- Internal: Generalizing within the community, group, or institution studied to persons, events, and settings that were not directly observed or interviewed.
- External: Refers to the extent to which one can extend the account of a particular situation or population to other communities, groups, or institutions.

The internal generalizability of this study was impacted by citation bias. Recommendations for SMS in software engineering note that multiple reference databases should be used to maximize internal generalizability [65]. Ideally, "grey literature" should be included [22]. However, the goal of this study was to research challenges, trends, and opportunities in non-invasive vocal fold assessment via acoustic analysis. As such, the study focuses on journal articles, conference proceedings, and reviews and is largely of interest to the scientific community.

General trends observed in this study could be extended to the computer science domain (external generalizability). Artificial intelligence-related research grew by 150%, while the overall body of indexed publications grew by 50% [66]. As artificial intelligence is a part of machine learning, it can be inferred that the popularity of various machine learning methods in vocal fold assessment is directly influenced by general trends in the computer science domain. Thus, challenges identified in this study could be detected in the computer science research domain.

7 Conclusions

The goal of this study was to identify challenges, trends, and opportunities in non-invasive vocal fold assessment via acoustic analysis. To achieve this goal, SMS was conducted. Web of Science reference database was used to extract documents with the topic “vocal folds state assessment” OR “vocal folds state”. Of 392 found documents, 92 were selected and classified according to created taxonomy scheme.

Acoustic feature-based analysis was identified as the main trend in the non-invasive vocal fold assessment area (**RQ**). Future analyses of this topic should continue this trend. Nevertheless, a lack of causality analysis was observed and will continue to impact future research. *Comprehensive* indicators would improve assessment of vocal folds state, enable estimation of state dynamics, and indicators’ relation with specific pathologies.

RQ1: Significant part of studies solve *Classification* task (39% of the set). Machine learning methods are most common (51% of the set used at least one method, e.g., *SVM*). The binary classification was found to be a common classification task type; however, this approach is influenced by existing reference databases (e.g. recordings of healthy and pathological voices are perfectly separable—this allows to achieve good results). There is a lack of labeled data to test proposed techniques and methods for validity and robustness thoroughly. For future work, we intend to experiment with existing reference databases to identify gaps in labeled data and it’s impact on *Classification*.

RQ2: Feature-based analysis was found to be most prevalent in vocal fold assessment (55% of the set). Acoustic features, such as noise content (e.g., HNR), stability and periodicity (e.g., RAP), and spectral-cepstral feature modeling (e.g., MFCC) are common. Medical professionals use several subjective evaluation methods (e.g., GRBAS, VHI) to identify specific pathologies. Therefore, one of the challenges is not only a universal set of features to diagnose voice pathologies, but also to differentiate disorders and pathologies. The vast majority of studies have focused on the classification of healthy-vs-pathologic, which allows to achieve high accuracy. However, differentiation of pathologies based on the acoustic features will become increasingly important, as an accurate initial diagnosis can be of great help in early diagnosis and treatment.

RQ3: Found trends indicate that vocal fold assessment research is exploratory and experimental (82% of the set used the Experimental research technique). Non-parametric models based on acoustic features are proposed, and experimental studies are carried out to demonstrate the superiority of the selected features. This research technique makes vocal fold assessment a highly productive research domain. However, the domain was found to be widely dispersed due to a variety of approaches. Soon we can expect the rise of deep learning methods application for voice pathology assessment, as this is a trend across the entire computer science.

The identified trends, challenges, and opportunities indicate the need for a multidisciplinary approach to vocal fold and voice pathology assessment if we seek to develop objective, non-invasive assessment techniques that are acceptable and understandable to the clinician community.

References

1. Henshilwood, C., d'Errico, F., Yates, R., et al.: Emergence of modern human behavior: middle stone age engravings from South Africa. Science **295**, 1278–1280 (2002). https://doi.org/10.1126/science.1067575
2. Hirano, M.: Morphological structure of the vocal cord as a vibrator and its variations. Folia Phoniatrica et Logopaedica **26**, 89–94 (1974). https://doi.org/10.1159/000263771
3. Ramig, L., Verdolini, K.: Treatment efficacy. J. Speech Lang. Hear. Res. (1998). https://doi.org/10.1044/jslhr.4101.s101
4. Roy, N., Merrill, R., Thibeault, S., et al.: Prevalence of voice disorders in teachers and the general population. J. Speech Lang Hear. Res. **47**, 281–293 (2004). https://doi.org/10.1044/1092-4388(2004/023)
5. Zhang, Z.: Mechanics of human voice production and control. J. Acoust. Soc. Am. **140**, 2614–2635 (2016). https://doi.org/10.1121/1.4964509
6. Lieberman, P.: Some acoustic measures of the fundamental periodicity of normal and pathologic larynges. J. Acoust. Soc. Am. **35**, 344–353 (1963). https://doi.org/10.1121/1.1918465
7. Koike, Y.: Vowel amplitude modulations in patients with laryngeal diseases. J. Acoust. Soc. Am. **45**, 839–844 (1969). https://doi.org/10.1121/1.1911554
8. Cairns, D., Hansen, J., Riski, J.: Detection of hypernasal speech using a nonlinear operator. In: Proceedings of 16th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. https://doi.org/10.1109/iembs.1994.412058
9. Moro-Velazquez, L., Gomez-Garcia, J., Godino-Llorente, J., et al.: Phonetic relevance and phonemic grouping of speech in the automatic detection of Parkinson's Disease. Sci. Rep. (2019). https://doi.org/10.1038/s41598-019-55271-y
10. Franciscatto, M., Augustin, I., Lima, J., Maran, V.: Situation awareness in the speech therapy domain: a systematic mapping study. Comput. Speech Lang. **53**, 92–120 (2019). https://doi.org/10.1016/j.csl.2018.08.002
11. Rybakovas, A., Beiša, V., Strupas, K., et al.: Inverse filtering of speech signal for detection of vocal fold paralysis after thyroidectomy. Informatica **29**, 91–105 (2018). https://doi.org/10.15388/informatica.2018.159
12. Kim, M., Kim, Y., Yoo, J., et al.: Regularized speaker adaptation of KL-HMM for dysarthric speech recognition. IEEE Trans. Neural Syst. Rehabil. Eng. **25**, 1581–1591 (2017). https://doi.org/10.1109/tnsre.2017.2681691
13. Gómez-García, J., Moro-Velázquez, L., Godino-Llorente, J.: On the design of automatic voice condition analysis systems. Part I: review of concepts and an insight to the state of the art. Biomed. Signal Process. Control **51**, 181–199 (2019). https://doi.org/10.1016/j.bspc.2018.12.024
14. Kasurinen, J., Knutas, A.: Publication trends in gamification: a systematic mapping study. Comput. Sci. Rev. **27**, 33–44 (2018). https://doi.org/10.1016/j.cosrev.2017.10.003

15. Kitchenham, B., Charters, S.: Guidelines for Performing Systematic Literature Reviews in Software Engineering, Technical Report EBSE 2007-001. Keele University and Durham University Joint Report (2007)
16. Nakamura, W.T., Oliveira, E.H., Conte, T.: Usability and User Experience Evaluation of Learning Management Systems—A Systematic Mapping Study. ICEIS (2017)
17. Kitchenham, B., Budgen, D., Pearl Brereton, O.: Using mapping studies as the basis for further research—a participant-observer case study. Inf. Softw. Technol. **53**, 638–651 (2011). https://doi.org/10.1016/j.infsof.2010.12.011
18. Kitchenham, B.A.: Procedures for Performing Systematic Reviews (2004)
19. Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M.: Systematic mapping studies in software engineering. In: Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering (EASE'08), pp. 68–77. BCS Learning & Development Ltd., Swindon, GBR (2008)
20. Kuhrmann, M., Fernández, D., Daneva, M.: On the pragmatic design of literature studies in software engineering: an experience-based guideline. Empir. Softw. Eng. **22**, 2852–2891 (2017). https://doi.org/10.1007/s10664-016-9492-y
21. Kitchenham, B., Budgen, D., Brereton, P.: Evid.-Based Softw. Eng. Syst. Revi. (2015). https://doi.org/10.1201/b19467
22. Martín-Martín, A., Orduna-Malea, E., Thelwall, M., Delgado Lózar E.: Google Scholar, web of science, and scopus: a systematic comparison of citations in 252 subject categories. J. Informetr. **12**, 1160–1177 (2018). https://doi.org/10.1016/j.joi.2018.09.002
23. Linder, S., Kamath, G., Pratt, G., et al.: Citation searches are more sensitive than keyword searches to identify studies using specific measurement instruments. J. Clin. Epidemiol. **68**, 412–417 (2015). https://doi.org/10.1016/j.jclinepi.2014.10.008
24. Sayago-Heredia, J., Pérez-Castillo, R, Piattini, M.: A systematic mapping study on analysis of code repositories. Informatica 619–660
25. Petersen, K., Vakkalanka, S., Kuzniarz, L.: Guidelines for conducting systematic mapping studies in software engineering: an update. Inf. Softw. Technol. **64**, 1–18 (2015). https://doi.org/10.1016/j.infsof.2015.03.007
26. Kitchenham, B., Brereton, P.: A systematic review of systematic review process research in software engineering. Inf. Softw. Technol. **55**, 2049–2075 (2013). https://doi.org/10.1016/j.infsof.2013.07.010
27. Pedreira, O., García, F., Brisaboa, N., Piattini, M.: Gamification in software engineering—a systematic mapping. Inf. Softw. Technol. **57**, 157–168 (2015). https://doi.org/10.1016/j.infsof.2014.08.007
28. Frank-Ito, D., Schulz, K., Vess, G., Witsell, D.: Changes in aerodynamics during vocal cord dysfunction. Comput. Biol. Med. **57**, 116–122 (2015). https://doi.org/10.1016/j.compbiomed.2014.12.004
29. Aneeja, G., Kadiri, S., Yegnanarayana, B.: Detection of glottal closure instants in degraded speech using single frequency filtering analysis. Interspeech **2018** (2018). https://doi.org/10.21437/interspeech.2018-1018
30. Turkmen, H., Karsligil, M.: Advanced computing solutions for analysis of laryngeal disorders. Med. Biol. Eng. Comput. **57**, 2535–2552 (2019). https://doi.org/10.1007/s11517-019-02031-9
31. Gonzalez-Lopez, J., Gomez-Alanis, A., Martin Donas, J., et al.: Silent speech interfaces for speech restoration: a review. IEEE Access **8**, 177995–178021 (2020). https://doi.org/10.1109/access.2020.3026579
32. Baumann, B.: Polarization sensitive optical coherence tomography: a review of technology and applications. Appl. Sci. **7**, 474 (2017). https://doi.org/10.3390/app7050474
33. Erath, B., Zañartu, M., Stewart, K., et al.: A review of lumped-element models of voiced speech. Speech Commun. **55**, 667–690 (2013). https://doi.org/10.1016/j.specom.2013.02.002
34. Cveticanin, L.: Review on mathematical and mechanical models of the vocal cord. J. Appl. Math. **2012**, 1–18 (2012). https://doi.org/10.1155/2012/928591

35. Jiang, W., Zheng, X., Xue, Q.: Computational modeling of fluid-structure-acoustics interaction during voice production. Front. Bioeng. Biotechnol. (2017). https://doi.org/10.3389/fbioe.2017.00007
36. Massachusetts Eye and Ear Infirmary, Voice disorders database, version.1.03, Lincoln Park, 625 NJ: Kay Elemetrics Corp (1994)
37. Voice Disorders. In: Asha.org (2022). https://www.asha.org/practice-portal/clinical-topics/voice-disorders. Accessed 03 Mar. 2022
38. Al-nasheri, A., Muhammad, G., Alsulaiman, M., et al.: An investigation of multidimensional voice program parameters in three different databases for voice pathology detection and classification. J. Voice **31**, 113.e9-113.e18 (2017). https://doi.org/10.1016/j.jvoice.2016.03.019
39. Daoudi, K., Bertrac, B.: On classification between normal and pathological voices using the MEEI-KayPENTAX database: issues and consequences. In: Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH (2014)
40. Woldert-Jokisz, B.: Saarbruecken Voice Database (2007)
41. Verikas, A., Gelzinis, A., Bacauskiene, M., et al.: Combining image, voice, and the patient's questionnaire data to categorize laryngeal disorders. Artif. Intell. Med. **49**, 43–50 (2010). https://doi.org/10.1016/j.artmed.2010.02.002
42. Jacobson, B., Johnson, A., Grywalski, C., et al.: The voice handicap index (VHI). Am. J. Speech-Lang. Pathol. **6**, 66–70 (1997). https://doi.org/10.1044/1058-0360.0603.66
43. Wu, Y., Chen, H., Liao, Y. et al.: Modeling perceivers neural-responses using lobe-dependent convolutional neural network to improve speech emotion recognition. Interspeech **2017** (2017). https://doi.org/10.21437/interspeech.2017-562
44. Voigt, D., Döllinger, M., Yang, A., et al.: Automatic diagnosis of vocal fold paresis by employing phonovibrogram features and machine learning methods. Comput. Methods Prog. Biomed. **99**, 275–288 (2010). https://doi.org/10.1016/j.cmpb.2010.01.004
45. Rogers, D., Setlur, J., Raol, N., et al.: Evaluation of true vocal fold growth as a function of age. Otolaryngol.-Head Neck Surg. **151**, 681–686 (2014). https://doi.org/10.1177/0194599814547489
46. Lenell, C., Sandage, M., Johnson, A.: A tutorial of the effects of sex hormones on laryngeal senescence and neuromuscular response to exercise. J. Speech Lang. Hear. Res. **62**, 602–610 (2019). https://doi.org/10.1044/2018_jslhr-s-18-0179
47. Everett, C., Blasi, D., Roberts, S.: Climate, vocal folds, and tonal languages: connecting the physiological and geographic dots. Proc. Nat. Acad. Sci. **112**, 1322–1327 (2015). https://doi.org/10.1073/pnas.1417413112
48. Bhuta, T., Patrick, L., Garnett, J.: Perceptual evaluation of voice quality and its correlation with acoustic measurements. J. Voice **18**, 299–304 (2004). https://doi.org/10.1016/j.jvoice.2003.12.004
49. Childers, D., Lee, C.: Vocal quality factors: analysis, synthesis, and perception. J. Acoust. Soc. Am. **90**, 2394–2410 (1991). https://doi.org/10.1121/1.402044
50. Little, M., McSharry, P., Hunter, E., et al.: Suitability of dysphonia measurements for telemonitoring of Parkinson's Disease. IEEE Trans. Biomed. Eng. **56**, 1015–1022 (2009). https://doi.org/10.1109/tbme.2008.2005954
51. Orozco-Arroyave, J., Hönig, F., Arias-Londoño, J. et al.: Spectral and cepstral analyses for Parkinson's disease detection in Spanish vowels and words. Expert Syst. **32**, 688–697 (2015). https://doi.org/10.1111/exsy.12106
52. Zhang, Y., Jiang, J., Rahn, D.: Studying vocal fold vibrations in Parkinson's disease with a nonlinear model. Chaos: Interdiscip. J. Nonlinear Sci. **15**, 033903 (2005). https://doi.org/10.1063/1.1916186
53. Genero, M., Fernández-Saez, A., Nelson, H., et al.: Research review. J. Database Manag. **22**, 46–70 (2011). https://doi.org/10.4018/jdm.2011070103
54. Dit, B., Revelle, M., Gethers, M., Poshyvanyk, D.: Feature location in source code: a taxonomy and survey. J. Softw.: Evol. Process **25**, 53–95 (2011). https://doi.org/10.1002/smr.567
55. Kagdi, H., Collard, M., Maletic, J.: A survey and taxonomy of approaches for mining software repositories in the context of software evolution. J. Softw. Maint. Evol.: Res. Pract. **19**, 77–131 (2007). https://doi.org/10.1002/smr.344

56. Cavalcanti, Y., da Mota Silveira Neto, P., Machado, I. et al.: Challenges and opportunities for software change request repositories: a systematic mapping study. J. Softw.: Evol. Process **26**, 620–653 (2013). https://doi.org/10.1002/smr.1639
57. Citation bias (2022). In: TheFreeDictionary.com. https://medical-dictionary.thefreedictionary.com/citation+bias. Accessed 1 Apr. 2022
58. NCI Dictionary of Cancer Terms. In: National Cancer Institute (2022). https://www.cancer.gov/publications/dictionaries/cancer-terms/def/selection-bias?redirect=true. Accessed 1 Apr. 2022
59. Mahtani, K., Spencer, E., Brassey, J., Heneghan, C.: Catalogue of bias: observer bias. BMJ Evid.-Based Med. **23**, 23–24 (2018). https://doi.org/10.1136/ebmed-2017-110884
60. Betrán, A., Say, L., Gülmezoglu, A., et al.: Effectiveness of different databases in identifying studies for systematic reviews: experience from the WHO systematic review of maternal morbidity and mortality. BMC Med. Res. Methodol. (2005). https://doi.org/10.1186/1471-2288-5-6
61. Hojo, N., Ohsugi, Y., Ijima, Y., Kameoka, H.: DNN-SPACE: DNN-HMM-Based generative model of voice F0 contours for statistical phrase/accent command estimation. INTERSPEECH (2017)
62. Chavan, R.S., Ganesh, D., Sablé, S.: An Overview of Speech Recognition Using HMM (2013)
63. Badampudi, D., Wohlin, C., Petersen, K.: Software component decision-making: In-house, OSS, COTS or outsourcing—a systematic literature review. J. Syst. Softw. **121**, 105–124 (2016). https://doi.org/10.1016/j.jss.2016.07.027
64. Barbosa, O., Alves, C.: A systematic mapping study on software ecosystems. In: Proceedings of the International Workshop on Software Ecosystems (2011)
65. Petersen, K., Gencel, C.: Worldviews, Research methods, and their relationship to validity in empirical software engineering research. In: 2013 Joint Conference of the 23rd International Workshop on Software Measurement and the 8th International Conference on Software Process and Product Measurement (2013). https://doi.org/10.1109/iwsm-mensura.2013.22
66. OECD: The Digitalisation of Science. Technology and Innovation: Key Developments and Policies, OECD Publishing, Paris (2020). https://doi.org/10.1787/b9e4a2c0-en

The Paradigm of an Explainable Artificial Intelligence (XAI) and Data Science (DS)-Based Decision Support System (DSS)

Vytautas Petrauskas, Raimundas Jasinevicius, Egidijus Kazanavicius, and Zygimantas Meskauskas

Abstract Decision support systems (DSS) are becoming a very important and widespread element of different fields of contemporary life in the age of explainable artificial intelligence (XAI). All of them somehow elaborate on the well-known procedures of data science transforming the data and/or signals into information, knowledge, and wisdom at last. However, most of the current DSS are limited to a mere finding of the situation, i.e. a kind of diagnostics, and do not have a unified integrated mechanism to offer adequate solutions. The main goal of this work and its novelty as well is to combine system analysis with the proposal of solutions using the latest XAI techniques based on the usage of the generalized approach and the newly developed fuzzy SWOT maps (FSM) method and on the elements of computing with words (CWW) according to the certain vocabulary and the lists of rules (LoR). They must be constructed on the available information base and are not the object of research in this article. The last goal of the article is to offer an approach that allows every phenomenon or system studied, following the philosophy of Hegel's triads, to detect its systematic, methodological and praxiological component, i.e. to form the SMP approach and use it in practice. An example of the case analyzed in the context of the proposed paradigm here was presented the assessment of opportunities and threats of such an entity as a state of Lithuania, to determine the state's risks and to generate optimal recommendations, actions and leverages for state's control. This work in general for the first time has demonstrated the vitality and possible efficiency of the paradigm.

Keywords Systemology · Methodology · Praxeology · Explainable artificial intelligence (XAI) · Computing with words (CWW) · Decision support systems (DSS) · Fuzzy expert maps (FEM) · Fuzzy SWOT maps (FSM) · Risk · Leverages · Fuzzy logic based reasoning · Verbalization · Degree of certainty · Membership function · Fuzzy logic terms

V. Petrauskas (✉) · R. Jasinevicius · E. Kazanavicius · Z. Meskauskas
Center of Real Time Computer Systems of Kaunas University of Technology, Kaunas, Lithuania
e-mail: vytautas.petrauskas@ktu.lt

G. Dzemyda et al. (eds.), *Data Science in Applications*,
Studies in Computational Intelligence 1084,
https://doi.org/10.1007/978-3-031-24453-7_9

1 Introduction

In today's age of a wide variety of computer systems and artificial intelligence (AI) technologies, no one doubts the importance, capabilities, and significance of decision support systems (DSS). Just ask in the GOOGLE system, for example, "decision support systems today" and the system provides data on more than 50 million links in 0.89 s.

In this abundance of information, attempts were made even in the 1980s to make some generalizations, provide a classification of such systems, review decision support systems and fields of their applications, define the concepts serving as a background for their structures, and show how it relates to other areas [1].

In fact, over the last 40–50 years, the range of human needs and capabilities of up to-days computer has expanded dramatically. Even the internet of things (T), services (S), actions (A), or behaviors and phenomena (P) has just emerged as a digitalized space (IoTSAP). And DSS has in one form or another become an integral part of the practice in many areas of life: industry, production, planning, logistics, business, medicine, banking, environment, engineering, and so on and so for. This can even be judged from several reviews of significant new literature sources such as [2–13].

The analysis performed on the basis of the sources mentioned above and in particular on the insights of [2, 9], can provide a picture reflecting the weight and role of DSS research in applications in various aspects of life. A general relative picture of this issue is given in Fig. 1.

This analysis already provides the following first conclusions: (1) a relatively large amount of research is deservedly devoted to the computer field and medicine; (2) some important areas are not yet covered by an adequate amount of research (for example, law, media evaluation, environmental side, university or company and so on); (3) unjustifiably low focus on DSS applications focuses on human safety and the promotion of DSS opportunities themselves.

The second part of the conclusions of this analysis focuses on the fragmentation of research diversity: in almost every field of DSS application, application specialists together with information technology specialists develop DSS structures and methodologies specific and dedicated to that particular field, i.e. there is a lack of commonalities in all DSS development and application processes. However, most of the DSS that exist today are limited to a mere finding of the situation, i.e. a kind of diagnostics, and do not have a unified integrated mechanism to offer adequate solutions. So, we feel the need for an efficient and unified universalized DSS paradigm covering a development methodology that is less scope-dependent.

The third part of the conclusions extracted from our analysis covers our attempt to evaluate the most popular and effective methods and software as well as simulation tools suitable to assist the proposition of the generalized approach to the creation of the functional organization of a DSS. As a matter of fact, practically all fields of activities presented in Fig. 1 and even banking, military operations planning, and/or private life are forced to cope with some external issues and signify some positive or negative consequences or results achieved under the presence or influence of some internal

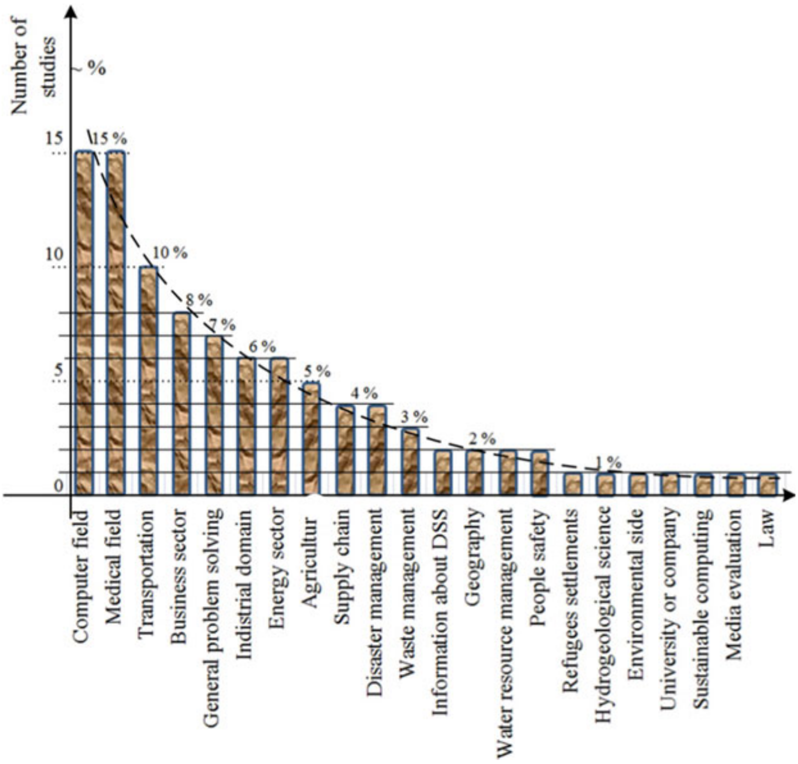


Fig. 1 Relative number of research studies in different fields of DSS applications [2]

stimulating or impeding factors or forces. Such scenarios correspond perfectly to the worldwide spread situation analysis and decision-making methodology called SWOT where S stands for Strength (ST), W—for Weakness (WK), O—for Opportunities (OP), and T for Threats (TH). It would be worth emphasizing that a few years ago the Center of Real Time Computer Systems (CRTCS) of the Department of Informatics at Kaunas University of Technology (KTU) started research in a very sensitive area of application of decision support systems including diagnostics, prediction of development and recommendations for further systems behavior under consideration. The joint team of researchers of the CRTCS encouraged by [14] has started the new approach to the diagnostics DSS based on analysis of strengths, weaknesses, opportunities, and threats of the system or problem under investigation (SWOT analysis) combined with elements of explainable artificial intelligence (XAI), computing with words (CWW), Data Science (DS) and general complex systems theory. This approach step-by-step was published in [15–17] and its vitality, at last, was confirmed and presented in [18], and the new universalized concept of a system using a dynamic SWOT analysis network called fuzzy SWOT map (FSM) for fuzzy control of risk in complex situations and environments was proposed.

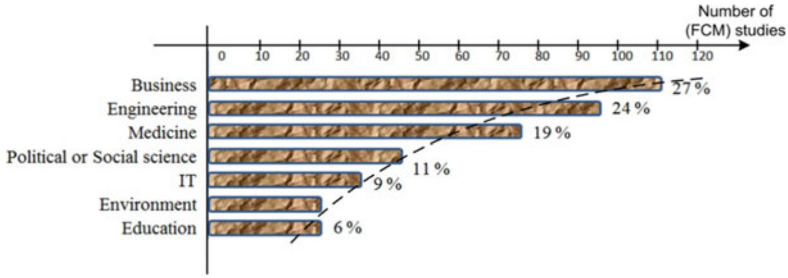


Fig. 2 Relative number of FCM trials and applications during the 2003–2013 in various fields of activities [2]

To the same tier of conclusions belong attempts to select methods and tools suitable to analyze, describe and simulate the phenomenon of mutual linkages and influences of elements connected as a system under consideration. As it is evident from the analysis of papers dedicated to DSS the fuzzy cognitive maps (FCM) approach is the most popular and promising in the historical discourse.

The cognitive maps approach to decision processes' analysis was started by R. Axelrod at Princeton University [19]. But it is widely reorganized that only after famous L.A. Zadeh's papers [20–22] did the contemporary avalanche of fuzzy sets applications burst. Fuzzy control systems (FCS) in industrial applications and fuzzy cognitive maps (FCM) for decision-making are the best confirmation of this tendency. By the way, the background for fuzzy thinking and for fuzzy cognitive maps' applications to decision-making processes was preliminary and mainly developed at the USC (University of Southern California) by Kosko [23–25] and later by Carvalho et al. [21–31] and many, many others.

At the moment popularity of the FCM-based DSS approach is demonstrated in Fig. 2.

Following the worldwide experience in soft computing (for example, [32–34]) as well as requirements, emphasized by different decision-makers, looking for efficient computerized advise in various cases of very sophisticated and sensitive situations like financial risk management, medical diagnostics, politics and international relations, environmental protection, terrorism, and security and so on [35–44], we have summarized the main theoretical features, capabilities, and limitations implemented and/or specific to FCM-based decision-making tools such as DSS [45–52].

In short, the three main conclusions drawn from the analysis of the literature and the summarization of our own scientific experience are as follows: (1) we noticed a lack of systematization in DSS paradigm development, (2) analysis and evaluation of most situations are based on SWOT analysis, and (3) most successful advises are based on FCM methodology. So, the main goal of this work and its novelty as well is to combine system analysis with the proposal of solutions using the latest XAI techniques based on the usage of the generalized approach and the newly developed fuzzy SWOT maps (FSM) method and on the elements of computing with words (CWW) according to the certain vocabulary and the lists of rules (LoR). That is the reason why Sect. 2 is devoted to the systemic approach to the DSS in general, Sect. 3 contains the description of experimental studies of a case and the demonstration of the vitality of the proposed approach. Section 4 concludes this chapter.

2 Systemic Approach to the Functional Organization of the DSS Paradigm

A systemic approach leading to acceptable results produced by DSS suggests the necessity to involve some philosophical background in the process of construction of the functional organization of the DSS inherent in the theory of complex systems.

It is obvious that each goal can be successfully achieved, or the efficient solution to each problem can be proposed if and only if the so-called three-dimensional (3D) approach is applied [53]. The 3D dimensions mean systemology (S), methodology (M), and praxeology (P) because for each case we need good theory, idea and/or spirit (S), efficient methods, tools, and/or modes of action (M) if we wish to receive practical advice and/or acceptable results (P). Therefore, it is worth renaming the 3D approach or at least putting it in a more adequate meaning and naming it as SMP-approach. Figure 3 corresponds to such a philosophy of possible efficiency of the SMP-approach concept.

Let us assume that elements s_i that belongs to the set $S(s_i \in S)$ and represents different theories are presented as circles in a certain abstract co-ordinate plain. In a similar way, all possible methods $m_j \in M$ that belong to the set M are represented by rectangles in the same plain system of coordinates expressing a certain abstract degree of proximity. Moreover, the practical results $p_k \in P$ as the elements of the set P are presented as crosses. It is easy to notice that well-founded results can be achieved when applied methods are based on proper theories. Consequently, the scientifically promising area corresponds to the *AND* relationship $S\&M\&P$. Such an approach implies a trend to ensure balance between a theory or idea, methods or policies to be used, and desirable results in the form of DSS suggestions in each field of activity, when analyzing and implementing any global system long-term existence or solving different worldwide problems mentioned in the first section of this chapter.

In any case, the situation described as belonging to the S&M&P must be somehow characterized, evaluated, and presented to the DSS. As is emphasized in the Sect. 1,

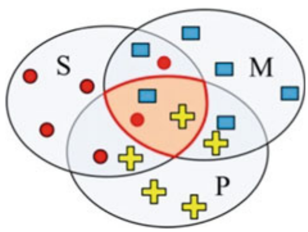


Fig. 3 Overlapping of the SMP entities

the enhanced fuzzy SWOT analysis serves as the most adequate and promising tool for such an evaluation.

2.1 SWOT Analysis

The most commonly and widely used means of evaluating ideas, plans, and activities are strengths, weaknesses, opportunities, and threats (SWOT) analysis. There is a large amount of literature devoted to SWOT analysis, even from its beginning somewhere at Harvard or Stanford schools times in the 1960s [54]. It is clear that this methodology played an important role in a variety of fields, including politics, military, economics, industry, health services, demographics, technology, and government.

First of all, it is considered that a given situation originates in some element e_e of the abstract environment (as indicated in Fig. 4), characterized by the following vectors, indicating strengths, weaknesses, opportunities, and threats, respectively:

$$\left.\begin{array}{l}
\overrightarrow{ST_e} = (ST_{e1},\ \ldots,\ ST_{es},\ \ldots,\ ST_{eS}),\ \text{es} = (1, \ldots, \text{es},\ \ldots,\ \text{eS}) \\
\overrightarrow{WK_e} = (WK_{e1},\ \ldots,\ WK_{ew},\ \ldots,\ WK_{eW}),\ \text{ew} = (1, \ldots, \text{ew},\ \ldots,\ \text{eW}) \\
\overrightarrow{OP_e} = (OP_{e1},\ \ldots,\ OP_{eo},\ \ldots,\ OP_{eO}),\ \text{eo} = (1,\ \ldots, \text{eo},\ \ldots,\ \text{eO}) \\
\overrightarrow{TH_e} = (TH_{e1},\ \ldots,\ TH_{et},\ \ldots,\ TH_{eT}),\ \text{et} = (1,\ \ldots, \text{et},\ \ldots,\ \text{eT})
\end{array}\right\} \quad (1)$$

All elements of vectors presented are given in the numerical form.

The evaluation of positive and negative features of the situation is conducted using the well-known procedure of SWOT analysis [17], incorporating a SWOT engine (Fig. 5).

In the expressions (2)

$$\left.\begin{array}{l}
OP_{\sum e} = \sum_{eo=1}^{eO}\left\{c_{eo}\left(\rho_{eo} + \sum_{es=1}^{eS} ST_{eos} + \sum_{ew=1}^{eW} WK_{oew}\right)\right\} \\
TH_{\sum e} = \sum_{et=1}^{eT}\left\{c_{et}\left(\rho_{et} + \sum_{es=1}^{eS} ST_{ets} + \sum_{ew=1}^{eW} WK_{etw}\right)\right\}
\end{array}\right\} \quad (2)$$

Fig. 4 Element e_e of the environment

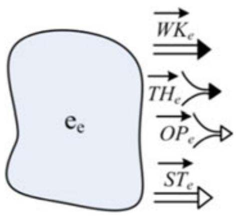
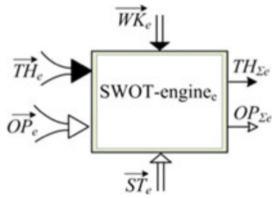


Fig. 5 SWOT engine



c_{eo} ($o = 1,\ldots,O$) indicates the degree of importance for each possible or predicted opportunity, and c_{et} ($t = 1,\ldots,T$) is the importance degree of each possible or predicted threat in the interval [0–1] for this particular situation or project. The initial values of truth ρ_{eo}, ρ_{et} for all $o = 1,\ldots,O$ and $t = 1,\ldots,T$ in the same interval [0–1] are shown as well. In general, $e = 1,\ldots,e, \ldots, E$ is a number of element under consideration.

2.2 *Elements of CWW*

The novelty of our DSS paradigm is the proposal for experts to use words from the selected vocabulary for the verbal evaluation of all possible entities during SWOT analysis [17]. As far as we know, no other study has included the possibility of a CWW paradigm for SWOT analysis.

An important feature of the approach is that estimates of the parameters to be processed can be either numerical or verbal. For example, the overall estimate x_s for any parameter s can be given in numeric form as $x_s = [x_s]$ or in verbal form as $x_s = \{x_s\}$. These notations [*] or {*} are used when it is necessary to emphasize the type of parameter estimate under processing; that is, in the absence of such a necessity, simply a generalized estimate notation x_s is used.

Since the SWOT engine e has to operate/process both numerical and verbal estimates of parameters and answers to questions or other evaluations, it is necessary to have some vocabulary of correspondences between digital and verbal estimates and fuzzy logic-based terms allowing the level of certainty μ (x) of such compliance to be assessed [17]. An example of such vocabulary and fuzzy logic terms used in practice is given in Fig. 6.

$$\begin{cases} \{\mathrm{N}\} - \mathrm{None} \\ \{\mathrm{S}\} - \mathrm{Small} \\ \{\mathrm{M}\} - \mathrm{Medium} \\ \{\mathrm{L}\} - \mathrm{Large} \end{cases}$$

The verbalization (fuzzification) of the digital estimate can be conveniently explained by the example in Fig. 7. Here a digital INPUT estimate $[x_1]$ is transformed into OUTPUT consisting of two words: the word $\{M\}$ (Medium) with the certainty μ (M) = 0.7 and the word $\{S\}$ (Small) with the certainty μ (S) = 0.3.

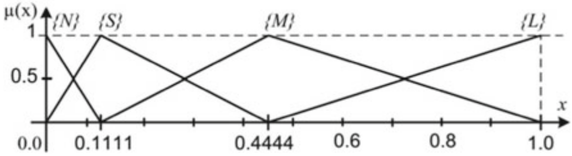


Fig. 6 An example of vocabulary and fuzzy logic terms

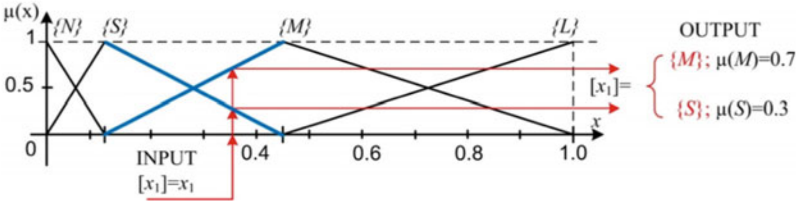


Fig. 7 An example of verbalization

The process of digitalization (defuzzification) of any verbal estimate is performed in an analog way and is demonstrated in Fig. 8. In this case a verbal estimate $\{x_1\} = \{S\}$ (Small) is presented at the INPUT together with the value of certainty of this statement let as say $\mu\ (S) = 0.75$.

The statement $\{S\}$ with the proclaimed degree of certainty generates in the OUTPUT two possible digital estimates: $[x_1] = x_{1L}$ and $[x_1] = x_{1R}$ which denote the left and right points of the number interval. And the higher certainty of the verbal estimate corresponds to the narrower range of the output estimate.

As for the SWOT engine it is important to note on the one hand that it is covered by elements using the conventional SWOT methodology [14] and on the other—that those elements are enriched with computing with words (CWW) capabilities and can process both normal digital information as well as verbal information, i.e., words representing one or another linguistic estimate of a parameter or indicator. Symbolically, the structure of such a SWOT engine which uses a CWW enhancement

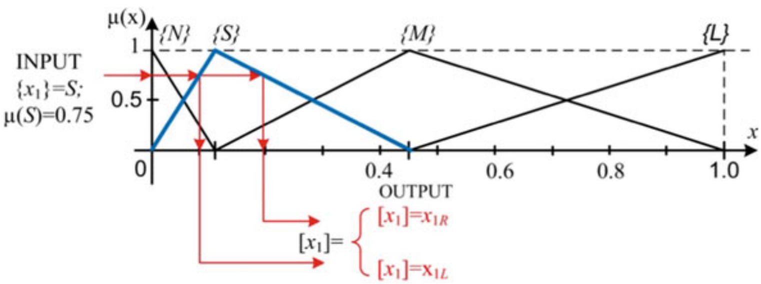


Fig. 8 An example of digitalization

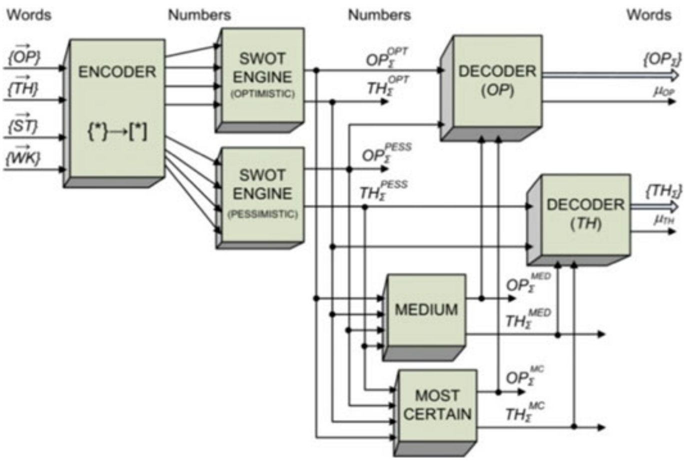


Fig. 9 SWOT engine enriched with computing with words (CWW) capabilities

and is based on the fuzzy logic and fuzzy reasoning mathematical apparatus is shown in Fig. 9 [17].

It is understood that the processing of vague verbal reasoning is related both to the vocabulary used and to the degree of certainty of each estimate. These characteristics and parameters will be discussed in the article as we move on to addressing specific problems of cultural, political, and economic interactions and relations.

2.3 SWOT and FCM Combination

The next novelty of our case is that here the normalized fuzzy cognitive maps (FCM) are used instead of SWOT tables. This is proposed in [46, 48, 63] for a combination of SWOT analysis and FCM in the attempt to better monitor the dynamic of SWOT analysis. Such a novelty perfectly corresponds to the FCM use tendency emphasized in the introduction and illustrated there by Fig. 2.

Regarding the combination and alignment of SWOT and FCM capabilities, as well as their possible substitutions, it should be noted that: a) the functional characteristics of the FCM nodes involved in the SWOT analysis are simply linear: y (x) = x and limited to the range [0–1]; and b) the output values of the FCM output nodes, like the SWOT tables, are normalized, i.e. the result obtained is divided by the maximum possible output value or result that can be obtained in that situation [15, 16, 18, 46, 48, 60, 63].

The interactions between SWOT engines on the SWOT analysis network-level usually reflect the real interactions of phenomena in the complex environment *E* under study. The model of these interactions at the SWOT engines level is implemented using the newly developed network of fuzzy SWOT maps (FSM) proposed at the

CRTCS center; the efficiency of those FSM has been tested and confirmed in other projects [16]. The main operation describing the influences and interactions between SWOT engines of any FSM is the matrix W_{MIL} representing influential linkages (MIL) on the FSM under consideration.

This matrix W_{MIL} performs the following operation on the INPUT vector $\overrightarrow{X}$ in order to receive the OUTPUT vector $\overrightarrow{Y}$:

$$\vec{Y} = W_{\text{MIL}}\,\vec{X} \tag{3}$$

$$\text{Here } \vec{X} = (\text{OP}_{\Sigma 1},\ \text{TH}_{\Sigma 1}, \ldots,\ \text{OP}_{\Sigma \text{e}},\ \text{TH}_{\Sigma \text{e}}, \ldots,\ \text{OP}_{\Sigma \text{E}},\ \text{TH}_{\Sigma \text{E}}), \tag{4}$$

$$\text{and } \vec{Y} = (\text{OP}_{\text{W}\Sigma 1},\ \text{TH}_{\text{W}\Sigma 1}, \ldots,\ \text{OP}_{\text{W}\Sigma \text{e}},\ \text{TH}_{\text{W}\Sigma \text{e}}, \ldots,\ \text{OP}_{\text{W}\Sigma \text{E}},\ \text{TH}_{\text{W}\Sigma \text{E}}). \tag{5}$$

The coefficients $W_{xi \to yj}$ of the matrix W_{MIL} are correspondingly such that $i = 1, \ldots, 2E$ and $j = 1, \ldots, 2E_{,:}$ and mean positive or negative influence of the element x_{i} to the element y_i. It should be noted that during the calculations, the influence coefficients are normalized so that each component of the $\overrightarrow{Y}$ vector fits in the range [0–1].

2.4 Risk Evaluation and Actions

The determination of the level of fuzzy risk concealed in a situation or project under consideration is based on our paradigmatic definition of risk presented in [55], and it somewhat constructively contradicts the opinion expressed in [56]. In this paper, risk is considered to be a normalized subjective level of the uncertainty of the consequences of activity and/or the state of the system of entities in complex environments. The results of SWOT analysis support the understanding or evaluation (numerical or verbal) of possible negative results ($TH_{\sum e}$), such as losses, threats, or disappointments, or possible positive results ($OP_{\sum e}$), such as achievements, or profits, opportunity, or joy. If activities (*EFF*) such as effort or investment that are part of this are included in the consideration, and the dimension of uncertainty (*HES or PROB*), whether hesitancy, randomness, possibility, or probability of certain events, are taken into account [57, 58, 61, 62], a measurable level of risk *R* can be calculated as a value for a certain function R, depending on *EFF*, *OP*, *TH*, and *HES* in an intuitive manner, as shown in Eq. (6):

$$R = \text{R}\,(EFF \uparrow; OP \downarrow; TH \uparrow; HES \uparrow /PROB \downarrow). \tag{6}$$

The arrows ↑ and ↓ mean increase and decrease in R, respectively.

A generalized risk evaluation engine is schematically presented in Fig. 10. The informal reasoning presented in Eq. (6) is constructed using fuzzy evaluation of

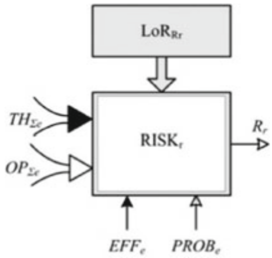


Fig. 10 Generalized risk-evaluation engine

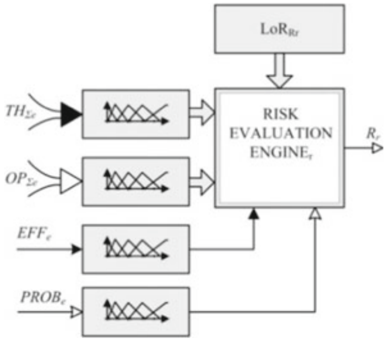


Fig. 11 Detailed elaboration of the risk-evaluation engine

$TH_{\sum e}$, $OP_{\sum e}$, $EFF_{e,}$ and $PROB_e$ prescribed by an *IF …THEN* type list or fuzzy rules (LoR_{Rr}) drafted by experts. In general, all lists of rules (LoR must be constructed on the available information base and they are not the objects of research in this article. The final output R_r can be aggregated using different strategies, as delivered, and thoroughly discussed in [25, 32], and elsewhere. The center of gravity (CoG) method ([59]) is used throughout this paper for its simplicity and efficiency.

A symbolic elaboration of the same risk-evaluation engine is given in Fig. 11.

The structure of this risk evaluation engine is based on the enriched fuzzy cognitive map (FCM) node concept, firstly proposed in [47] as fuzzy expert maps (FEM) and later developed and extensively studied in [49–52].

The risk assessment phase usually creates a natural need to answer the question: what next? Actions, recommendations, and leverages must be offered for the effective operation of each DSS. For generalized simplicity reasons, the part of DSS responsible for the actions here is called LEVERAGE, and its aggregation also is performed in the form of FEM [47] as it is presented in Fig. 12. This generalized structure of the AGGREG-engine permits to determine of at least one item L_l for leverage.

In this case, the combinations of the elements of the vector of different risks $\overrightarrow{R} = (R_1, \ldots, R_e, \ldots, R_E)$ obtained in the early stages of the risk evaluation are expertly evaluated verbally, using a list of the *IF …THEN* type fuzzy rules (LoR_{AI}), and then the reasoning is summarized, also using the different strategies [23, 32]; for this, the center of gravity (CoG) methodology is generally preferred in our applications.

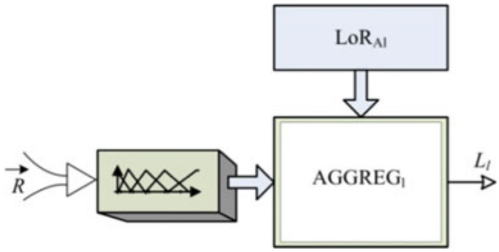


Fig. 12 Generalized structure of the engine for producing of leverage

Usually leverage item L_l is used as a support of decision-making to close the loop of the feedback system and enable the improvement of performance in the situation or project under investigation.

2.5 Systemic Structure of DSS

The systemic structure of generalized DSS is based on the description of the functional organization of its main parts delivered in 2.1–2.5 subsections of this chapter and is presented in Fig. 13.

The primary system of coordinates is borrowed from a universal philosophical approach to human life and activity, which includes S—systemology (the science of systems, theories, visions), M—methodology (the science of modes of operation, methods, tools), and P—praxeology (the science of results and their use).

Here the SWOT analysis block (*SWOT ANALYSIS*) creates corresponding fuzzy SWOT maps (FSM) discussed in Sect. 2.1 of this chapter and evaluates opportunities and threats (OP and TH) emerging in all overlapping S, M, and P environments. The

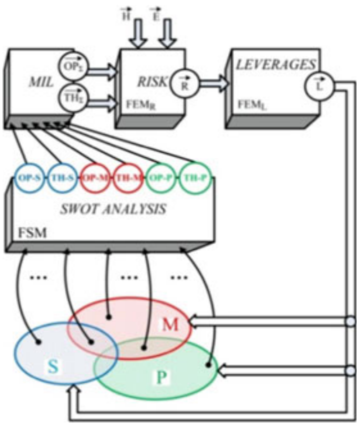


Fig. 13 The systemic structure of a generalized DSS

MIL block using the corresponding matrix W_{MIL}, representing influential mutual linkages of all possible OPs and THs, presents all summarized vectors of opportunities $\overrightarrow{OP}_{\Sigma}$ as well as threats $\overrightarrow{TH}_{\Sigma}$.

The *RISK* block uses fuzzy expert maps (FEM_R) mechanism and evaluates all possible risks $\overrightarrow{R}$ necessary to be able to suggest the usage of certain leverages $\overrightarrow{L}$ as a result of DSS reasoning in the block *LEVERAGES* using the corresponding fuzzy expert maps FEM_L mechanism similar to the FEM_R.

Such a systemic structure of generalized DSS is built and its activity is demonstrated in the following Sect. 3 for a simplified real case.

3 Experimental Studies, Validation and Applications

This section is devoted to presenting a simplified example of a real case under investigation and to demonstrating the efficient activities of the corresponding DSS by simulating and validating its different applications.

3.1 Description of a Real Case Under Investigation

It was decided to build the DSS according to its paradigm as a tool enabling to receive results of analysis and to get certain recommendations, advices, and suggestions of corresponding actions for people working in the field of governmental foreign relations.

In such a case it is natural to assign some specific and essential entities to the environment of the situations, presented in Fig. 13 as S, M, and P, respectively,—as C—culture (cultural, spiritual, educational, etc. background), P—policy (political, methodological, instrumental possibilities, sanctions and means and etc. background), and E-economy (economic, industrial background, the situation on the worldwide market, standard of living, and etc.). Symbolically, Fig. 13 becomes a new one corresponding to the simplified case under investigation and is shown in Fig. 14.

In assessing the state of society and the state, the first coordinate would be CULTURE (in a broad sense) and reflect the spiritual, cultural, and educational aspects of society; the second—POLICY (methodology of operation) would assess both: human and individual status and external/inter-institutional methodologies of group activities, and cross-border and/or international relations, etc.; the third—ECONOMY (material aspects) would cover gross domestic product, economy, production, the standard of living, etc. Understandably, these three aspects interact, are intertwined, and only a peculiar balance between them roughly determines the value state that satisfies us and is attainable. After all, it is impossible to create a powerful economy (E) if there is a low level of culture (C), education, and spiritual

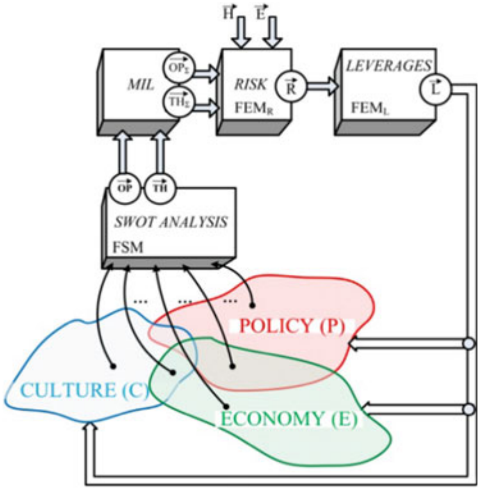


Fig. 14 The structure of a DSS tool for the case under investigation

life; as well as the unimaginable opportunity to create an advanced political system (P) in a weak economy. It is difficult to imagine the necessary support for cultural development in an environment of an unfavorable political system. It is also difficult to expect positive changes in the spiritual life without an efficient economy and smart policies and so on.

As the literature shows, the coordinates mentioned here, such as C, P, E, are difficult to estimate not only qualitatively but also quantitatively. However, their individual components (such as GDP per capita, or mortality, or exports, etc.) in a given case can be measured and compared quantitatively. The change in the estimate of the state of society (and of the individual) is fully examined and described according to the E coordinate. Apparently, the individual and individual groups in human society are most sensitive to practical changes in various aspects of the economy. As a result, most statistical information and all its changes and their interpretations can be found in the literature. There are significantly fewer generalized statistical (quantitative) surveys related to, for example, trends in cultural change. The policy area also receives few quantitative assessments. The latter areas are studied more by historians, culturologists, and political scientists than by mathematicians and/or statisticians; and qualitative estimates are more appropriate here.

In order to give more reality to the illustrative analysis of entities C, M, E, a specific state of the European Union was chosen – the Republic of Lithuania. The specific data required for the DSS tool of the model case in question were collected from various sources prepared in the offices of the EU and Lithuanian governmental institutions. The main ones are: [63–74, 76]. After the analysis, performed by experts of the Center of Real Time Computer Systems (CRTCS) of the Faculty of Informatics in Kaunas University of Technology (KTU), and supported by consultations of specialists from the Ministry of Foreign Affairs of Lithuania, the nomenclature of Fuzzy SWOT Maps (FSM) was produced and delivered in Table 1.

Table 1 Nomenclature of FSM for analysis of entities C, P and E in a case of Lithuania

	Oportunities			Threats			Strenghts			Weaknesses		
Culture_C	1	National identity	NID	1	Loss of national values	VAL	1	Historical experience	HEX	1	Strong trend of liberalism	LIB
	2	Intelligence and creativity	INT	2	Emigration	EMG	2	Western European Christian tradition of culture	WEU	2	Illeagal imigration	IIM
	3	Honesty	HON	3	Low level of education	LLE						
	4	Tolerance	TOL	4	Cosmopolitism	COS						
Policy_P	1	State sovereignity	STS	1	Hybrid war	HYW	1	Nato membership	ANT	1	Misinformation	MIN
	2	Demokracy	DEM	2	Military actions	MIL	2	Patriotism of cityzens	PTR	2	Insufficient defence funding	FUN
	3	Safety and security	S&S	3	Diplomatic pressure	DIP				3	Unfavorable geographical location	GEO
				4	Artificial conflicts of political parties	CPP						
Economy_E	1	High GDP	GDP	1	Artificial crisis	CRI	1	Eu membership	EUM	1	Privatization of strategic objects	PRV
	2	Optimal market regulation	REG	2	Sabotage ofunfriendly countries and companies	SAB	2	High level of university and professional education	PTR	2	Lack of state commercial banks	LCB
	3	Acceptable dignity of life for all cityzens	DIG									
	4	Participation in the world economy	WRD									

3.2 FCM for Analysis of Culture, Politics and Economy

According to the nomenclature of FSM presented in Table 1, the same expert groups have presented their SWOT-type influences of corresponding weaknesses and threats to opportunities and threats for each entity: culture—C, policy—P, and economy—E. All influences are shown in a convenient format of fuzzy cognitive maps (FCM) and are correspondingly delivered in Figs. 15, 16 and 17.

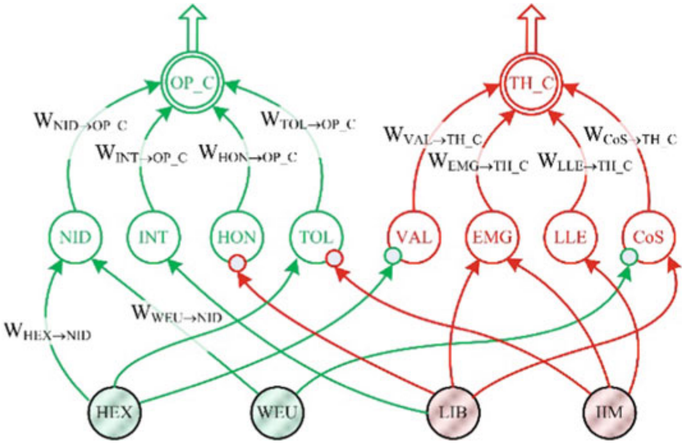


Fig. 15 The FCM_C: SWOT FCM for the entity of culture

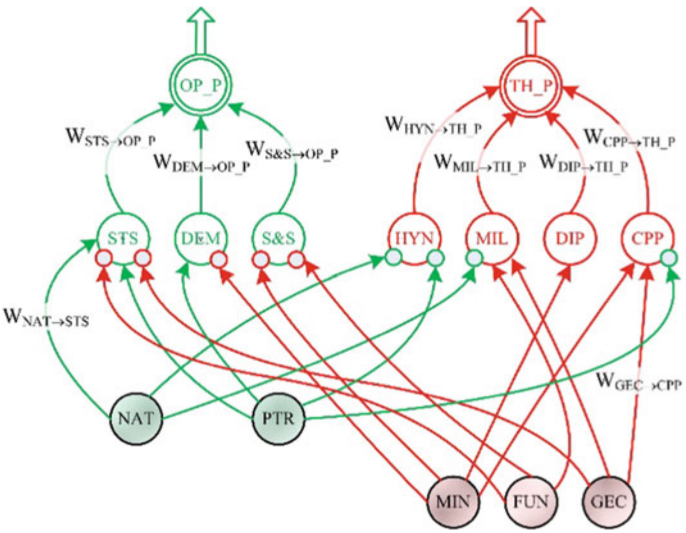


Fig. 16 The FCM_P: SWOT FCM for the entity of policy

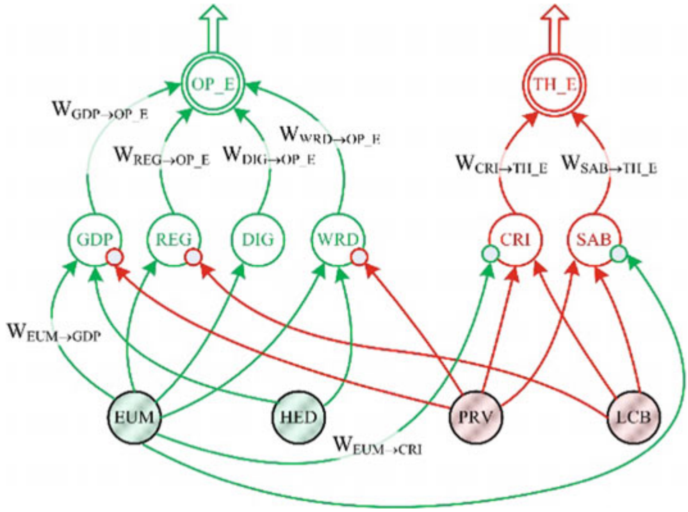


Fig. 17 The FCM_E: SWOT FCM for the entity of economy

Coefficients of influences corresponding to the situations described as fuzzy cognitive maps (FCM) in Figs. 15, 16, and 17 are presented in a convenient format of tables (Table 2 for the FCM_C, Table 3 for the FCM_P, and Table 4 for the FCM_E) for the following quantitative evaluations. Must be emphasized that initial values for the entities involved also are presented in these tables in the format NAME(0) (for example, NID(0) = 0.90, INT(0) = 0.80, and so on).

According to the generalized structure of the DSS, presented in Fig. 14, the influential mutual linkages of all possible OPs and THs must be considered. This is done in the *MIL* block using the methodology, expressed in (3)–(5), where the matrix W_{MIL} is produced by the experts' team and is presented in Table 5.

The corresponding fuzzy SWOT map (FSM) for this case is delivered in Fig. 18.

3.3 Notes on the Scaling of FSM Variables

All calculations related to the analysis of opportunities and threats in all reported cases of FCM and FSM must be performed in accordance with the principles of fuzzy cognitive maps and computing with words paradigms, which are widely described in the literature and highly targeted in [17, 25, 31, 32, 47, 61], and elsewhere. This means that the numeric variables fit in the range [0–1] and the variables match the vocabulary of the verbal estimates chosen by the specialists, experts, and/or users as it was described in Sect. 2.2.

Table 2 Quantitative values of influences for the FCM_C (experts' decision)

FCM_C					**OP_C**	**TH_C**
$W_{HEX \to NID}$ 0.90	$W_{WEU \to NID}$ 0.70			**NID(0)** **0.90**	$W_{NID \to OP\_C}$ 0.70	
		$W_{LIB \to INT}$ 0.60		**INT(0)** **0.80**	$W_{INT \to OP\_C}$ 0.80	
		$W_{LIB \to HON}$ -0.70		**HON(0)** **0.60**	$W_{HON \to OP\_C}$ 0.70	
$W_{HEX \to TOL}$ 0.80			$W_{IIM \to TOL}$ -0.30	**TOL(0)** **0.20**	$W_{TOL \to OP\_C}$ 0.30	
$W_{HEX \to VAL}$ -0.70				**VAL(0)** **0.40**		$W_{VAL \to TH\_C}$ 0.70
		$W_{LIB \to EMG}$ 0.80	$W_{IIM \to EMG}$ 0.60	**EMG(0)** **0.50**		$W_{EMG \to TH\_C}$ 0.60
			$W_{IIM \to LLE}$ 0.70	**LLE(0)** **0.30**		$W_{LLE \to TH\_C}$ 0.50
	$W_{WEU \to CoS}$ -0.80	$W_{LIB \to CoS}$ 0.90		**CoS(0)** **0.70**		$W_{CoS \to TH\_C}$ 0.80
HEX(0) **0.90**	**WEU(0)** **0.80**	**LIB(0)** **0.80**	**IIM(0)** **0.50**			

Table 3 Quantitative values of influences for the FCM_P (experts' decision)

FCM_P						**OP_P**	**TH_P**
$W_{NAT \to STS}$ 0.90	$W_{PTR \to STS}$ 0.80		$W_{FUN \to STS}$ -0.40	$W_{GEO \to STS}$ -0.60	**STS(0)** **0.90**	$W_{STS \to OP\_P}$ 0,90	
	$W_{PTR \to DEM}$ 0.80	$W_{MIN \to DEM}$ -0.20			**DEM(0)** **0.70**	$W_{DEM \to OP\_P}$ 0,80	
		$W_{MIN \to S\&S}$ -0.30	$W_{FUN \to S\&S}$ -0.60		**S&S (0)** **0.80**	$W_{S\&S \to OP\_P}$ 0,80	
$W_{NAT \to HYN}$ -0.80	$W_{PTR \to HYN}$ -0.70				**HYW(0)** **0.40**		$W_{HYN \to TH\_P}$ 0.70
$W_{NAT \to MIL}$ -0.60			$W_{FUN \to MIL}$ 0.60	$W_{GEO \to MIL}$ 0.80	**MIL(0)** **0.30**		$W_{MIL \to TH\_P}$ 0.40
		$W_{MIN \to DIP}$ 0.50			**DIP(0)** **0.50**		$W_{DIP \to TH\_P}$ 0.60
	$W_{PTR \to CPP}$ -0.60	$W_{MIN \to CPP}$ 0.70		$W_{GEO \to CPP}$ 0.70	**CPP(0)** **0.60**		$W_{CPP \to TH\_P}$ 0.50
NAT(0) **0.90**	**PTR(0)** **0.80**	**MIN(0)** **0.50**	**FUN(0)** **0.40**	**GEO(0)** **0.80**			

Table 4 Quantitative values of influences for the FCM_E (experts' decision)

FCM_E					OP_E	TH_E
$W_{EUM \to GDP}$ 0.80	$W_{HED \to GDP}$ 0,70	$W_{PRV \to GDP}$ -0,50		**GDP(0) 0.40**	$W_{GDB \to OP\_E}$ 0.90	
$W_{EUM \to REG}$ 0.60			$W_{LCB \to REG}$ -0.70	**REG(0) 0.70**	$W_{REG \to OP\_E}$ 0.30	
$W_{EUM \to DIG}$ 0.70				**DIG(0) 0.50**	$W_{DIG \to OP\_CE}$ 0.40	
$W_{EUM \to WRD}$ 0.80	$W_{HED \to WRD}$ 0.70	$W_{PRV \to WRD}$ -0.40		**WRD(0) 0.10**	$W_{WRD \to OP\_E}$ 0.70	
$W_{EUM \to CRI}$ -0.10		$W_{PRV \to CRI}$ 0.50	$W_{LCB \to CRI}$ 0.60	**CRI(0) 0.10**		$W_{CRI \to TH\_E}$ 0.50
$W_{EUM \to SAB}$ -0.70		$W_{PRV \to SAB}$ 0.7	$W_{LCB \to SAB}$ 0.6	**SAB(0) 0.40**		$W_{SAB \to TH\_E}$ 0,70
EUM(0) 0.80	**HED(0) 0.70**	**PRV(0) 0.80**	**LCB(0) 1.0**			

Table 5 The mutual linkages W_{MIL} representing influences of Opportunities $\overrightarrow{OP}$ and Threats $\overrightarrow{TH}$ to summarized $\overrightarrow{OP}\Sigma$ and $\overrightarrow{TH}\Sigma$

	$OP_{\Sigma C}$	$TH_{\Sigma C}$	$OP_{\Sigma P}$	$TH_{\Sigma P}$	$OP_{\Sigma E}$	$TH_{\Sigma E}$
OP_C	**1**		**+0.5**	**-0.3**	**+0.3**	**-0.4**
TH_C	**-0.7**	**1**	**-0.4**		**-0.2**	**+0.4**
OP_P	**+0.1**		**1**		**+0.5**	**-0.2**
TH_P		**0.4**		**1**	**-0.1**	**0.3**
OP_E	**+0.1**		**+0.3**		**1**	
TH_E	**-0.3**	**0.1**	**-0.1**	**0.3**		**1**

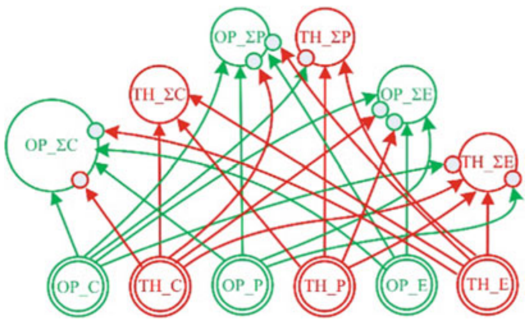


Fig. 18 FSM corresponding to the matrix W_{MIL} (Table 5)

Table 6 Scales for FCM

MAX					
OP_C	TH_C	OP_P	TH_P	OP_E	TH_E
4.34	4.51	4.67	3.76	5.16	2.66
MAX					
$OP_{\Sigma C}$	$TH_{\Sigma C}$	$OP_{\Sigma P}$	$TH_{\Sigma P}$	$OP_{\Sigma E}$	$TH_{\Sigma E}$
5.32	6.28	8.39	4.56	8.78	5.59
Scales					
$OP_{\Sigma C}$	$TH_{\Sigma C}$	$OP_{\Sigma P}$	$TH_{\Sigma P}$	$OP_{\Sigma E}$	$TH_{\Sigma E}$
0.1878	0.1592	0.1192	0.02193	0.01136	0.1788

In this case, the data presented in Tables 2, 3, 4 and 5 and fuzzy cognitive maps in Figs. 15, 16, 17 and 18 can be represented as matrices with rows numbered j = 1, 2, … and columns i = 1, 2, …, respectively.

Since each opportunity or threat under consideration has and can be characterized at its own hierarchical level, unit, or highest level of verbal estimation, it is appropriate to set those maximum estimates at biased input signal estimates.

In order to obtain the maximum OP, TH estimates and to calculate the normalization scales from them, the procedure is as follows:

1. Maximum estimates of each OP and TH are calculated by taking NAME(0) value as 1 and adding up all the strengths and weaknesses influences W that increase the sum for that OP or TH (skipping negative influences), then the resulting sum is multiplied by $W_{NAME \to OP\_X}$ or $W_{NAME \to TH\_X}$ value.
2. For each project, the obtained separate OP and TH maximum estimates are aggregated, and the total OPX and THX maximum of the project X are calculated as $MAX(OP_X)$ and $MAX(TH_X)$ (the resulting maximum OP and TH estimates for the projects are listed at the top part of Table 6).
3. For the calculation of the maximum estimates of $OP_{\Sigma X}$ and $TH_{\Sigma X}$ for each project X, influences between the projects are considered as listed in Table 5. Maximum estimates of $OP_{\Sigma X}$ and $TH_{\Sigma X}$ calculation steps:

a. Each column of Table 5 reflects the project's $OP_{\Sigma X}$ and $TH_{\Sigma X}$.
b. The rows contain the influence values of the other projects.
c. Only rows with positive influence are considering.
d. To the calculated project X OP_X or TH_X maximum estimate, the other project Y OP_Y or TH_Y maximum estimate is added multiplied by the value of the projects influence and this is repeated for all positive influences. The result is $OP_{\Sigma X}$ and $TH_{\Sigma X}$ maximum estimates for project X after external influences of the other projects evaluation. (the resulting maximum $OP_{\Sigma X}$ and $TH_{\Sigma X}$ estimates for the projects are listed in the middle of Table 6).

4. The scaling of the project OP is calculated as $1/OP_{\Sigma X}$ and the scaling of the project TH is calculated as $1/TH_{\Sigma X}$.

Multiplying the project results by the resulting scales ensures that the situation under consideration is always assessed in the [0–1] ranges. Obtained scales are delivered in Table 6.

These scales are used in the following subsection for the simulation of the described mutual cultural, political, and economic interactions in the real life of the state under investigation.

3.4 *Results of Simulation of the Cultural, Political and Economic Interactions*

The screenshot of the DSS tool ([75]) for the case of this given state's analysis of its cultural opportunities and threats, as well as political and economic ones, are presented in Fig. 19. There all parts of fuzzy SWOT maps (FSM), explained in Figs. 15, 16, 17 and 18, are collected and connected together for convenient simulation and analysis FCM (Fig. 20).

Results of simulation from this DSS tool are ready for: (1) an analysis of possible opportunities and threats in all three entities under investigation; (2) investigation of possible dynamics of results with changes of input values, and (3) for a risk evaluation in each field of state's activity (C, P, and E).

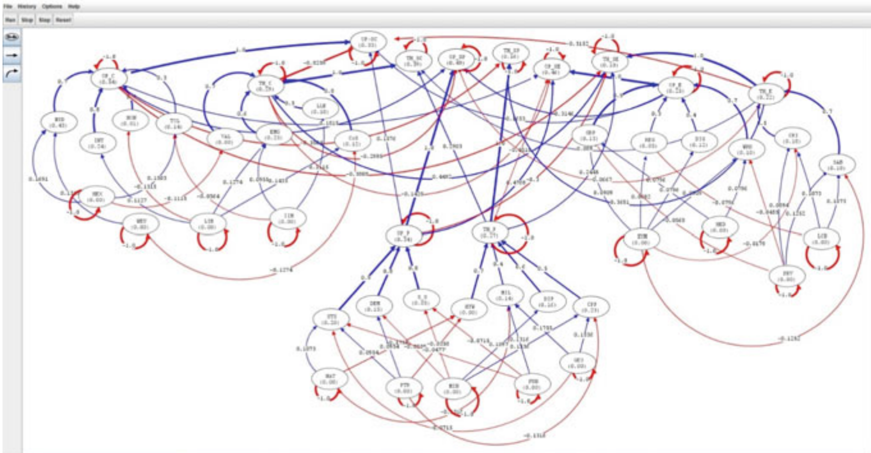


Fig. 19 The fuzzy SWOT map for the system under simulation (the case of Republic of Lithuania)

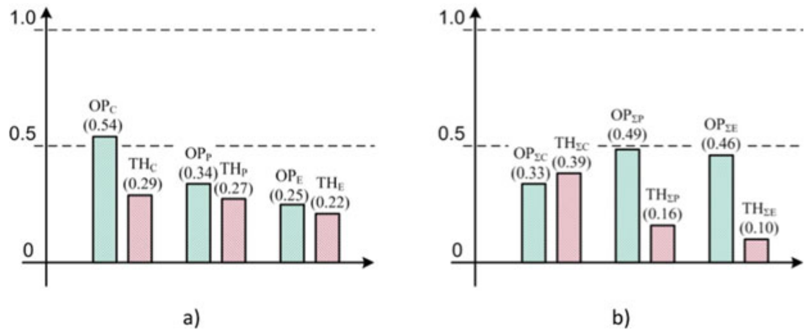


Fig. 20 Results of simulation of the FSM in case of Lithuania: **a**—opportunities and threats in C, P and E entities, **b**—outputs of block *MIL* assessing the effect of interactions of the C, P and E

3.5 Risk Evaluation

The evaluation of the state's activity risk in all three fields under consideration in this DSS version is performed according to the methodology delivered in Sect. 2.4 and based on formula (6) keeping in mind the experts' verbal conviction that efforts demonstrated in the field of culture (C) are SMALL, efforts in the field of policy (P) are LARGE, and efforts in the field of the economy (E) are MEDIUM. Such an opinion will be reflected in the three different lists of fuzzy rules adequate to the field to be described. Experts, in this case, do not have any doubts and/or hesitations, and they speak about probabilities of all events as equal to 1. This is the reason why risk formula (6) is simplified and looks as

$$R = \mathrm{R}\,(OP\downarrow; TH\uparrow). \tag{7}$$

Figure 10 must be simplified and used in the form as shown in the following illustrations of the fuzzy Tech software [77] presented later.

Must be emphasized that for fuzzy evaluations verbal vocabulary was used:

$$\begin{cases} \{\mathrm{N}\} - \text{None} \\ \{\mathrm{S}\} - \text{Small} \\ \{\mathrm{M}\} - \text{Medium} \\ \{\mathrm{L}\} - \text{Large} \\ \{\mathrm{VL}\} - \text{Very large;} \end{cases}$$

And the fuzzy logic terms as they are proposing in [17], and also seen in the following illustrations.

So, for risk evaluation in each field of state's activity here we present the structure of the risk-evaluation engine, two sets of input terms, one set of output terms, list of corresponding rules (LoR), and the three-dimensional diagram of results. All methodologies for the risk concept and evaluation in the fields of country's culture (C), policy (P), and economy (E) are based on the paradigm (6) and systemic structures, presented in Figs. 10 and 11. The main aspects of really working models are elaborated and delivered in the following Sects. 3.5.1–3.5.3.

3.5.1 The Field of Culture (C)

According to the methodology presented in Sect. 2.4 the structure for RISK_C evaluation using the fuzzyTech software [77] is presented in Figs. 21 and 22.

Using the list of corresponding experts' rules (LoR) for the RISK_C (Fig. 23) the results of the risk evaluation are presented in Figs. 24 and 25.

The cultural risk for Lithuania is quite large {L} with the certainty $\mu = 0.87$. And the three-dimensional results are presented in Fig. 25 predicting upcoming situations.

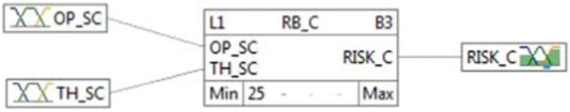


Fig. 21 The risk-evaluation engine for RISK_C

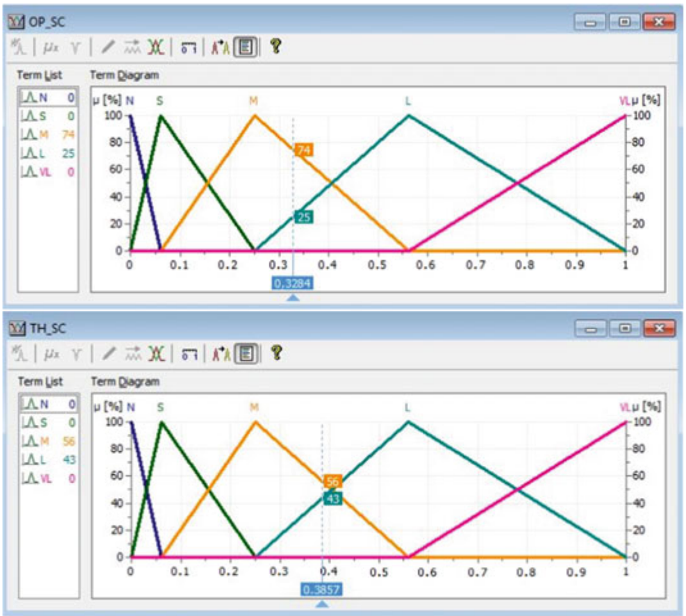
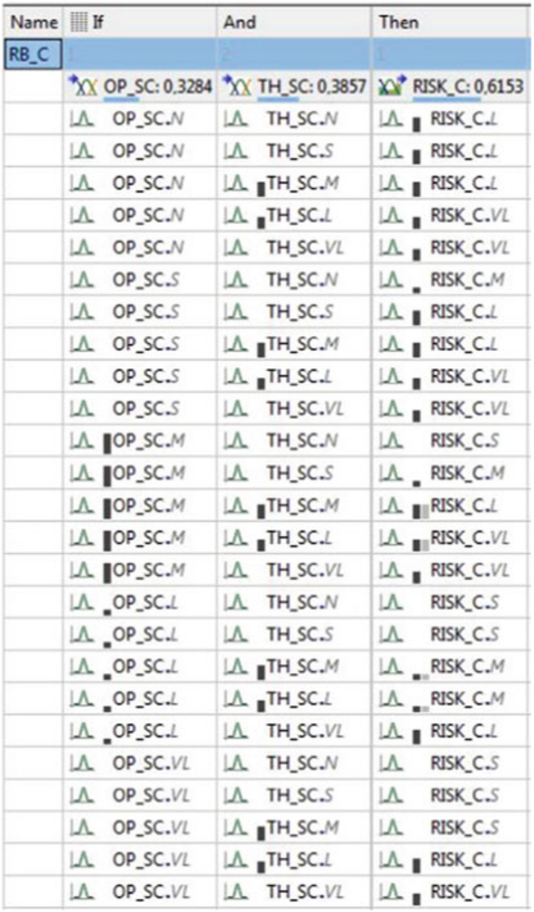


Fig. 22 The two sets of input terms OP_SC and TH_SC



Name	If	And	Then
RB_C	1	2	1
	OP_SC: 0,3284	TH_SC: 0,3857	RISK_C: 0,6153
	OP_SC.N	TH_SC.N	RISK_C.L
	OP_SC.N	TH_SC.S	RISK_C.L
	OP_SC.N	TH_SC.M	RISK_C.L
	OP_SC.N	TH_SC.L	RISK_C.VL
	OP_SC.N	TH_SC.VL	RISK_C.VL
	OP_SC.S	TH_SC.N	RISK_C.M
	OP_SC.S	TH_SC.S	RISK_C.L
	OP_SC.S	TH_SC.M	RISK_C.L
	OP_SC.S	TH_SC.L	RISK_C.VL
	OP_SC.S	TH_SC.VL	RISK_C.VL
	OP_SC.M	TH_SC.N	RISK_C.S
	OP_SC.M	TH_SC.S	RISK_C.M
	OP_SC.M	TH_SC.M	RISK_C.L
	OP_SC.M	TH_SC.L	RISK_C.VL
	OP_SC.M	TH_SC.VL	RISK_C.VL
	OP_SC.L	TH_SC.N	RISK_C.S
	OP_SC.L	TH_SC.S	RISK_C.S
	OP_SC.L	TH_SC.M	RISK_C.M
	OP_SC.L	TH_SC.L	RISK_C.M
	OP_SC.L	TH_SC.VL	RISK_C.L
	OP_SC.VL	TH_SC.N	RISK_C.S
	OP_SC.VL	TH_SC.S	RISK_C.S
	OP_SC.VL	TH_SC.M	RISK_C.S
	OP_SC.VL	TH_SC.L	RISK_C.L
	OP_SC.VL	TH_SC.VL	RISK_C.VL

Fig. 23 The list of corresponding rules (LoR) for the RISK_C

		Numerical	Verbal	Certainty μ
INPUTS	OP_ΣC	0.3284	L	0.25
			M	0.74
	TH_ΣC	0.3857	L	0.43
			M	0.56
OUTPUT	RISK-C	0.6153	VL	0.13
			L	0.87

Fig. 24 The set of output terms RISK_C: VL – 0.13, L – 0.87

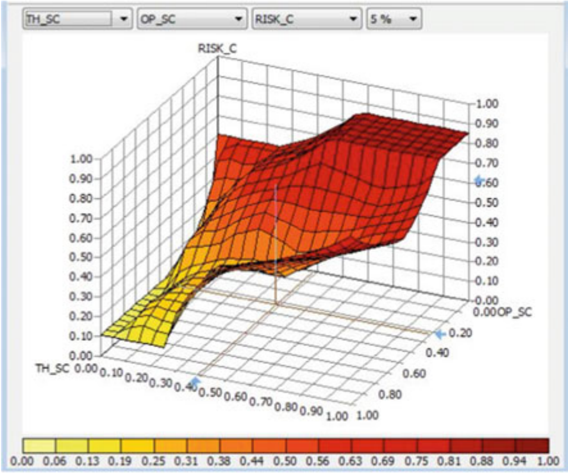


Fig. 25 The RISK_C dependence on possible OP_C and TH_C fluctuations

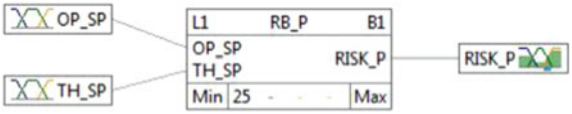


Fig. 26 The risk-evaluation engine for RISK_P

3.5.2 The Field of Policy (P)

According to the methodology presented in Sect. 2.4 the structure for RISK_P evaluation using the fuzzyTech software [77] is presented in Figs. 26 and 27.

Using the list of corresponding experts' rules (LoR) for the RISK_P (Fig. 28) the results of risk evaluation are presented in Figs. 29 and 30.

The political risk for Lithuania is quite large {L} as well with the certainty $\mu = 0.92$. And the three-dimensional results are presented in Fig. 30 predicting upcoming possible fluctuations in political situations.

3.5.3 The Field of Economy (E)

Analogue's methodology was used structure for RISK_E evaluation using the same fuzzyTech software [77], and the RISK_E evaluation structure is presented in Figs. 31 and 32.

Using the list of corresponding experts' rules (LoR) for the RISK_E (Fig. 33) the results of the risk evaluation are presented in Figs. 34 and 35.

The risk in the field of the economy for Lithuania is medium {M} with the certainty $\mu = 0.75$. And the three-dimensional results are presented in Fig. 35 predicting upcoming fluctuations in the situations.

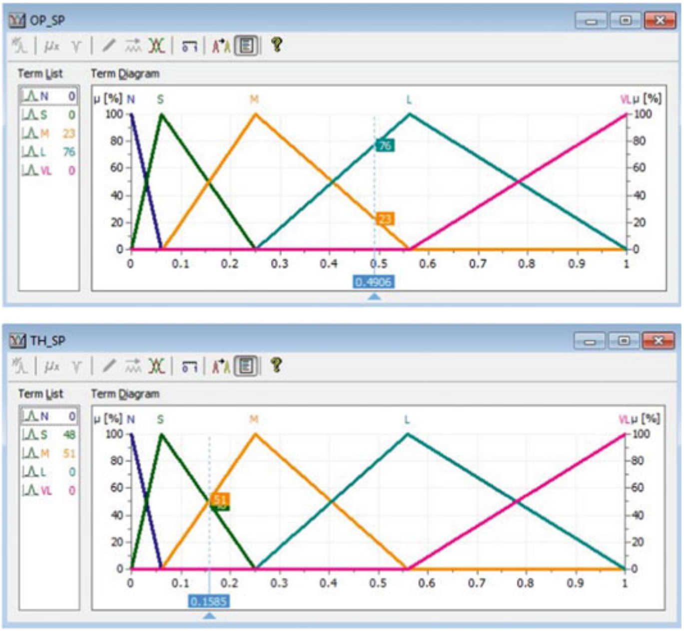


Fig. 27 The two sets of input terms OP_SP and TH_SP

The simulation results described, elaborated, and delivered in Sects. 3.5.1–3.5.3 mainly serve two purposes: (1) they are ready to be presented to the experts and specialists for evaluation and exploration of real risks, and (2) they are ready to be used in the next stage according to the Fig. 14 for the selection and proposition adequate recommendations, creation of corresponding signals and/or activation suitable leverages listed in the list of acceptable measures.

3.6 Recommendations, Leverages and Actions

In fact, only those decision support systems (DSS) are fully operational that provide not only assessments of opportunities (OP) and threats (TH), not only risk (RISK) assessments but also suggestions on what action needs to be taken, what measures to use to make their advices as effective as possible and realizable. For the simplicity of explanations all recommendations, actions, signals, and other proposed influences here are called LEVERAGES, which are under our control. The methodology of determinations of those LEVERAGES is delivered in Sects. 2.4 and 2.5. They are based on Figs. 12 and 13.

Name	If	And	Then
RB_P	1	2	1
	OP_SP: 0,4906	TH_SP: 0,1585	RISK_P: 0,53472
	OP_SP.N	TH_SP.N	RISK_P.VL
	OP_SP.N	TH_SP.S	RISK_P.VL
	OP_SP.N	TH_SP.M	RISK_P.VL
	OP_SP.N	TH_SP.L	RISK_P.VL
	OP_SP.N	TH_SP.VL	RISK_P.VL
	OP_SP.S	TH_SP.N	RISK_P.L
	OP_SP.S	TH_SP.S	RISK_P.L
	OP_SP.S	TH_SP.M	RISK_P.VL
	OP_SP.S	TH_SP.L	RISK_P.VL
	OP_SP.S	TH_SP.VL	RISK_P.VL
	OP_SP.M	TH_SP.N	RISK_P.S
	OP_SP.M	TH_SP.S	RISK_P.L
	OP_SP.M	TH_SP.M	RISK_P.VL
	OP_SP.M	TH_SP.L	RISK_P.VL
	OP_SP.M	TH_SP.VL	RISK_P.VL
	OP_SP.L	TH_SP.N	RISK_P.S
	OP_SP.L	TH_SP.S	RISK_P.M
	OP_SP.L	TH_SP.M	RISK_P.L
	OP_SP.L	TH_SP.L	RISK_P.L
	OP_SP.L	TH_SP.VL	RISK_P.VL
	OP_SP.VL	TH_SP.N	RISK_P.S
	OP_SP.VL	TH_SP.S	RISK_P.S
	OP_SP.VL	TH_SP.M	RISK_P.M
	OP_SP.VL	TH_SP.L	RISK_P.L
	OP_SP.VL	TH_SP.VL	RISK_P.VL

Fig. 28 The list of corresponding rules (LoR) for the RISK_P

		Numerical	Verbal	Certainty µ
INPUTS	OP_ΣP	0.4906	L	0.76
			M	0.23
	TH_ΣP	0.1585	M	0.51
			S	0.49
OUTPUT	RISK-P	0.5347	L	0.92
			M	0.08

Fig. 29 The set of output terms RISK_P: L – 0.92, M – 0.08

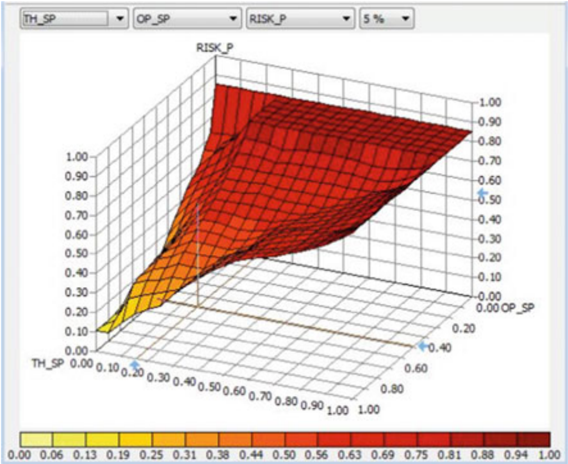


Fig. 30 The RISK_P dependence on TH_SP and OP_SP

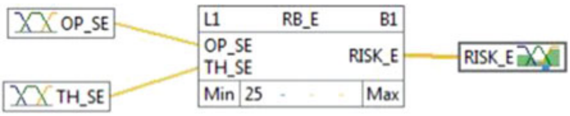


Fig. 31 The risk-evaluation engine for RISK_E

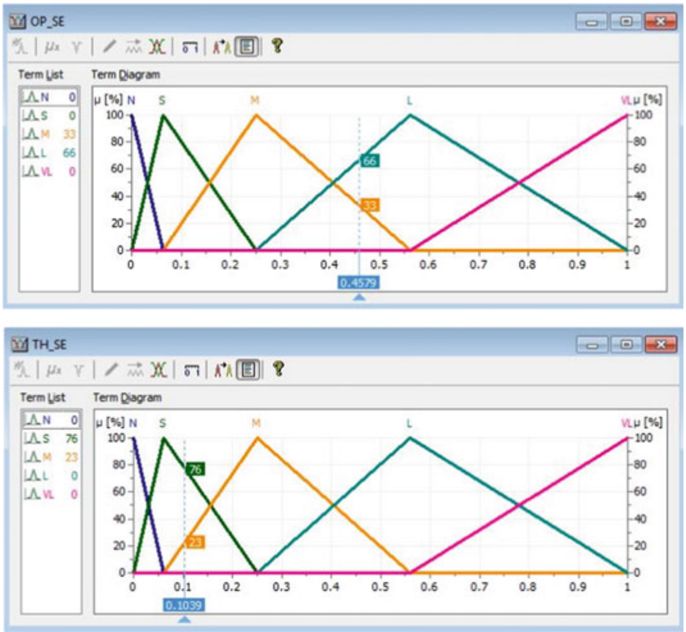


Fig. 32 The two sets of input terms OP_SE and TH_SE

Name	If	And	Then
RB_E	1	2	1
	OP_SE: 0,4579	TH_SE: 0,1039	RISK_E: 0,33588
	OP_SE.N	TH_SE.N	RISK_E.L
	OP_SE.N	TH_SE.S	RISK_E.L
	OP_SE.N	TH_SE.M	RISK_E.L
	OP_SE.N	TH_SE.L	RISK_E.VL
	OP_SE.N	TH_SE.VL	RISK_E.VL
	OP_SE.S	TH_SE.N	RISK_E.M
	OP_SE.S	TH_SE.S	RISK_E.L
	OP_SE.S	TH_SE.M	RISK_E.L
	OP_SE.S	TH_SE.L	RISK_E.VL
	OP_SE.S	TH_SE.VL	RISK_E.VL
	OP_SE.M	TH_SE.N	RISK_E.S
	OP_SE.M	TH_SE.S	RISK_E.M
	OP_SE.M	TH_SE.M	RISK_E.L
	OP_SE.M	TH_SE.L	RISK_E.VL
	OP_SE.M	TH_SE.VL	RISK_E.VL
	OP_SE.L	TH_SE.N	RISK_E.S
	OP_SE.L	TH_SE.S	RISK_E.S
	OP_SE.L	TH_SE.M	RISK_E.M
	OP_SE.L	TH_SE.L	RISK_E.M
	OP_SE.L	TH_SE.VL	RISK_E.L
	OP_SE.VL	TH_SE.N	RISK_E.S
	OP_SE.VL	TH_SE.S	RISK_E.S
	OP_SE.VL	TH_SE.M	RISK_E.M
	OP_SE.VL	TH_SE.L	RISK_E.L
	OP_SE.VL	TH_SE.VL	RISK_E.VL

Fig. 33 The list of corresponding rules (LoR) for the RISK_E

		Numerical	Verbal	Certainty μ
INPUTS	OP_ΣE	0.4579	L	0.66
			M	0.33
	TH_ΣE	0.1039	M	0.23
			S	0.76
OUTPUT	RISK-E	0.3359	L	0.25
			M	0.75

Fig. 34 The set of output terms RISK_E: M – 0.75, L – 0.25

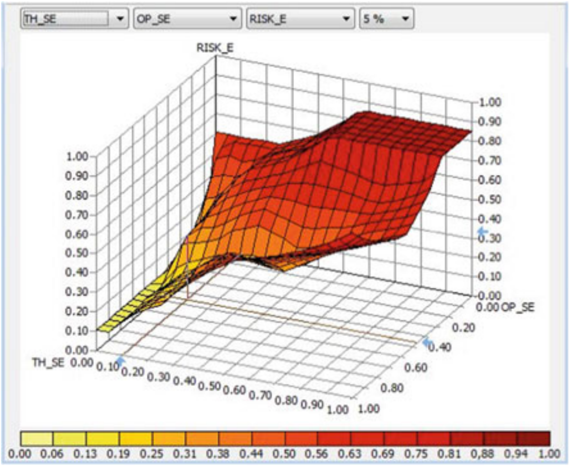


Fig. 35 The RISK_E dependence on OP_ΣE and TH_ΣE

The process starts from the analysis of the lists of all available LEVERAGES. The nomenclature of leverages, actions, recommendations, and so on as well as their acronyms and lists for their verbal evaluation of the case under consideration are presented in Table 7.

This table serves as a source for the LEVERAGE determination block structure shown in Fig. 36. The expert team's opinion, based on their knowledge and experience, produced the list of decision rules given in Table 8.

Table 7 Nomenclature of leverages available in a case of Lithuania under consideration

	Leverage, action and/or recommendation		Terms for verbal evaluation	Acronym
Culture	Strenghtening of lithuanian national identity	C1	N, S, M, L, VL	NI
	National agreement on education	C2	YES, N	ED
	Involvement of society into decision making processes	C3	N, S, M, L, VL	IS
Policy	Create eu border control forces	P1	N, S, M, L, VL	BC
	Minimize beurocratic administration	P2	N, S, M, L, VL	MB
	Restore real representative demokracy	P3	N, S, M, L, VL	RD
	Strengthen support for military aviation	P4	N, S, M, L, VL	MA
Economy	Reform of taxes	E1	N, S, M, L, VL	RT
	Establishing of national commercial bank	E2	YES, N	CB
	Achieve the same level of work organization in state and private sector of economy	E3	N, S, M, L, VL	LW

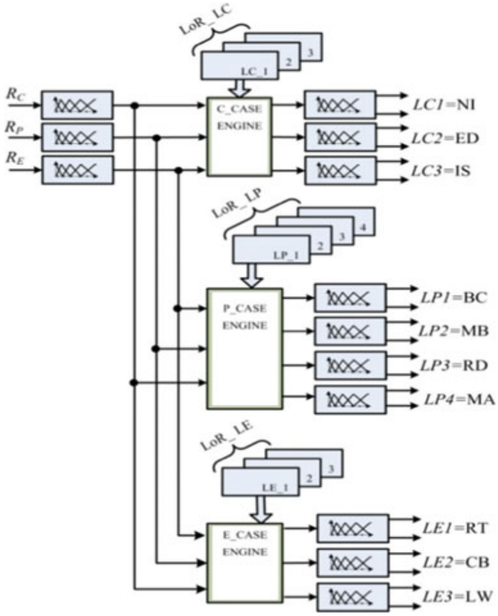


Fig. 36 The LEVERAGE determination block structure

The whole simulation of the activity of the LEVERAGE determination block structure according to the rules presented in Table 8 was simulated on the fuzzyTech 6.82f software package [77]. The model of the main structure is shown in Fig. 37. It must be emphasized that the numerical inputs (RISK_C, RISK_P, and RISK_E) were verbalized using three terms: SMALL, MEDIUM, and LARGE in spite of the fact that all qualitative outputs (LC1, LC3, LP1–LP4, LE1, LE3) were verbalized using five ones: N, S, M, L, and VL; and two quantitative outputs—LC2 and LE2 were verbalized using only two terms: YES and N which better reflects the realities under consideration.

The examples of real RISK input terms for the case are shown in Fig. 38.

The simulation was performed according to the full LoR presented in Fig. 39 using different input numerical and verbal values. Here the illustrative results are presented to show the force and vitality of the proposed generalized approach to the description and evaluation of a real case. We have selected only results connected to the following leverages: LC1 = NI, LP1 = BC, and LE2 = CB. For each leverage, we demonstrate the final suggestion to be used and delivered in verbal as well as in numerical form and three diagrams permitting us to evaluate leverage's reactions to changes possibly caused by fluctuations in combinations of RISK_C, RISK_P, and RISK_E.

Such results for the LC1 = NI are presented in Fig. 40, and it is seen, that the recommendation is 0.55 from the interval 0–1 and it means that the efforts to strengthen Lithuanian national identity (NI) must be LARGE with the certainty 0.95.

Table 8 The list of rules (LoR) for suggestions and recommendations to be made in the LEVERAGE determination block

IF INPUT R =	*AND* R_C	*AND* R_P	*AND* R_E	*THEN* LC1	*THEN* LC2	*THEN* LC3	*THEN* LP1	*THEN* LP2	*THEN* LP3	*THEN* LP4	*THEN* LE1	*THEN* LE2	*THEN* LE3
Number of rule													
1	S	S	S	N	NO	N	N	N	N	N	N	NO	N
2	S	S	M	S	NO	N	N	N	N	S	S	YES	N
3	S	S	L	S	YES	S	S	N	S	S	M	YES	S
4	S	M	S	S	NO	S	S	S	S	N	M	NO	S
5	S	M	M	M	NO	S	S	M	S	M	L	YES	M
6	S	M	L	M	YES	M	L	M	M	L	M	YES	M
7	S	L	S	S	NO	M	L	S	M	M	M	NO	S
8	S	L	M	M	NO	L	M	M	M	L	L	YES	M
9	S	L	L	M	YES	S	VL	L	M	VL	VL	YES	L
10	M	S	S	S	NO	N	S	S	S	N	S	NO	S
11	M	S	M	S	NO	S	S	S	M	M	S	YES	M
12	M	S	L	M	YES	M	L	M	M	L	L	YES	M
13	M	M	S	M	NO	M	M	S	L	M	M	NO	M
14	M	M	M	M	NO	L	M	M	L	L	L	YES	M
15	M	M	L	M	YES	L	L	M	L	L	L	YES	M
16	M	L	S	VL	YES	M	L	M	M	M	S	NO	M
17	M	L	M	VL	YES	M	L	L	L	L	M	YES	M
18	M	L	L	VL	YES	VL	VL	VL	L	VL	VL	YES	L
19	L	S	S	M	YES	S	M	M	L	S	S	NO	S
20	L	S	M	M	YES	S	L	M	M	M	L	YES	M

(continued)

Table 8 (continued)

IF INPUT R =	*AND* R_C	*AND* R_P	*AND* R_E	*THEN* LC1	*THEN* LC2	*THEN* LC3	*THEN* LP1	*THEN* LP2	*THEN* LP3	*THEN* LP4	*THEN* LE1	*THEN* LE2	*THEN* LE3
21	L	S	L	M	YES	M	M	L	L	L	M	YES	L
22	L	M	S	L	YES	M	M	M	M	M	M	NO	L
23	L	M	M	L	YES	M	L	L	M	L	M	YES	L
24	L	M	L	L	YES	L	L	L	L	VL	L	YES	L
25	L	L	S	VL	YES	VL	L	L	VL	M	M	NO	L
26	L	L	M	VL	YES	VL	VL	VL	VL	L	L	YES	VL
27	L	L	L	VL	YES	VL	VL	VL	VL	VL	VL	YES	VL

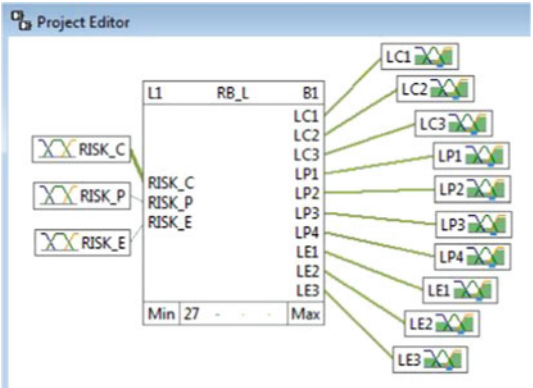


Fig. 37 A model of the main structure of the LEVERAGE determination block

The results concerning the leverage LP1 = BC to create an efficient EU border control (BC) are presented in Fig. 41, and it is seen, that the certainty of this recommendation is 0.55 from the interval 0–1. It means that the efforts to establish a strong EU border control system must be LARGE with a high certainty 0.95. Such a result perfectly corresponds to the opinion of French president F. Mitterrand.

The results presented in Fig. 42 convincingly demonstrate that the opening of the national Lithuanian commercial bank is vitally recommended for the successful future of the Lithuanian state.

The answer YES is delivered with the certainty 0.85 and there are no high opposing signals when the combinations of RISK_C, or RISK_P, or RISK_E suffer from some fluctuations in situations under consideration.

Similar results, concerning all leverages of the case under consideration are collected, summarized, and presented in the Table 9.

They can be received and must be used as a feedback influence as is advocated in Fig. 13 or less abstractive, as it is described in Sect. 3.1 and shown in Fig. 14. Another possibility of usage of information existing in the LEVERAGES is based on the ability to simulate the environment hidden in Fig. 14. Such a simulation strengthens the guarantee of the success of DSS recommendations because the simulation results are received before the practical its use.

4 Conclusions

The article presents a generalized approach to the development of decision support systems (DSS), and provides a methodology for analyzing the opportunities and threats of the situation on the basis of fuzzy SWOT maps (FSWOTM) networks, risk assessment and recommendations, measures, and ways to propose mitigation or

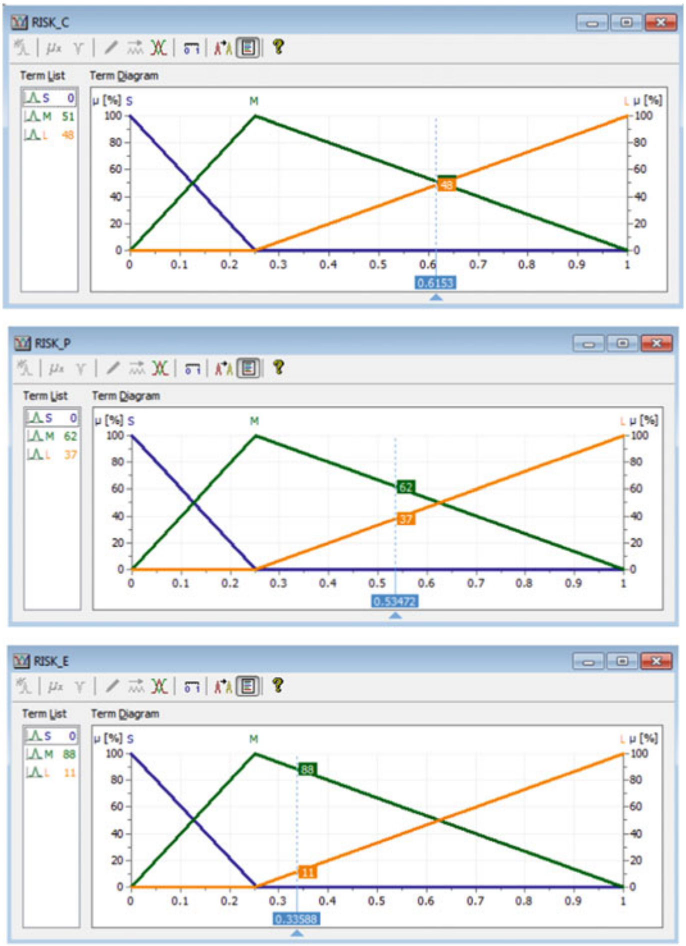


Fig. 38 Real RISK input terms for RISK_C, RISK_P and RISK_E

elimination of those threats. The methodology uses elements of explainable artificial intelligence (XAI) and verbal information processing or computing with word (CWW) as well as the verbal assessment of the situations under consideration.

The so-called three-dimensional (3D) approach is applied and the 3D dimensions mean systemology (S), methodology (M), and praxeology (P). For each particular domain, S, M, and P represent the situation and/or entity-specific to that domain. In this section, a generalized model of the state with its cultural (C), political (P), and economic (E) problems is selected and examined in a specific area. The article presents three levels of non-compliance: the level of opportunities and threats, the level of risk assessment, and the level of recommendations, advices, signals, and other measures for environmental impacts. The modeling of the problems of the

Name	If	And	And	Then	With	And
RB_L						
	RISK_C: 0,6153	RISK_P: 0,53472	RISK_E: 0,33588	LC1: 0,55516	DoS [%]	LP1: 0,55516
	RISK_C.S	RISK_P.S	RISK_E.S	LC1.N	100	LP1.N
	RISK_C.S	RISK_P.S	RISK_E.M	LC1.S	100	LP1.N
	RISK_C.S	RISK_P.S	RISK_E.L	LC1.S	100	LP1.S
	RISK_C.S	RISK_P.M	RISK_E.S	LC1.S	100	LP1.S
	RISK_C.S	RISK_P.M	RISK_E.M	LC1.M	100	LP1.S
	RISK_C.S	RISK_P.M	RISK_E.L	LC1.M	100	LP1.L
	RISK_C.S	RISK_P.L	RISK_E.S	LC1.S	100	LP1.L
	RISK_C.S	RISK_P.L	RISK_E.M	LC1.M	100	LP1.M
	RISK_C.S	RISK_P.L	RISK_E.L	LC1.M	100	LP1.VL
	RISK_C.M	RISK_P.S	RISK_E.S	LC1.S	100	LP1.S
	RISK_C.M	RISK_P.S	RISK_E.M	LC1.S	100	LP1.S
	RISK_C.M	RISK_P.S	RISK_E.L	LC1.M	100	LP1.L
	RISK_C.M	RISK_P.M	RISK_E.S	LC1.M	100	LP1.M
	RISK_C.M	RISK_P.M	RISK_E.M	LC1.M	100	LP1.M
	RISK_C.M	RISK_P.M	RISK_E.L	LC1.M	100	LP1.L
	RISK_C.M	RISK_P.L	RISK_E.S	LC1.VL	100	LP1.L
	RISK_C.M	RISK_P.L	RISK_E.M	LC1.VL	100	LP1.L
	RISK_C.M	RISK_P.L	RISK_E.L	LC1.VL	100	LP1.VL
	RISK_C.L	RISK_P.S	RISK_E.S	LC1.M	100	LP1.M
	RISK_C.L	RISK_P.S	RISK_E.M	LC1.M	100	LP1.L
	RISK_C.L	RISK_P.S	RISK_E.L	LC1.M	100	LP1.M
	RISK_C.L	RISK_P.M	RISK_E.S	LC1.L	100	LP1.M
	RISK_C.L	RISK_P.M	RISK_E.M	LC1.L	100	LP1.L
	RISK_C.L	RISK_P.M	RISK_E.L	LC1.L	100	LP1.L
	RISK_C.L	RISK_P.L	RISK_E.S	LC1.VL	100	LP1.L
	RISK_C.L	RISK_P.L	RISK_E.M	LC1.VL	100	LP1.VL
	RISK_C.L	RISK_P.L	RISK_E.L	LC1.VL	100	LP1.VL

Fig. 39 The fragment of a simulation list of rules (LoR) for suggestions and recommendations to be made in the LEVERAGE determination block

selected real case (state) demonstrated not only the novelty of the methodology but also its effective vitality.

The work with the models of the examined levels showed the compatibility and simplicity of the selected modeling tools. At the same time, it has formulated topics for the further expansion and development of this approach. Therefore, the continuation of works envisages: (1) development of a unified general-purpose software tool that can play the role of a decision support system (DSS-tool), (2) development of methodology and tools for DSS application in various IoTSAP areas (Internet of Things, Services, Actions, and Phenomena), (3) proposing a concept for creating an adequate to the reality a hierarchical DSS network, (4) developing theoretical and practical foundations to analyze the dynamics of DSS networks, and (5) to propose and test a software environment for modeling the reality of SMP.

LC1: L = 0.95,
M = 0.05

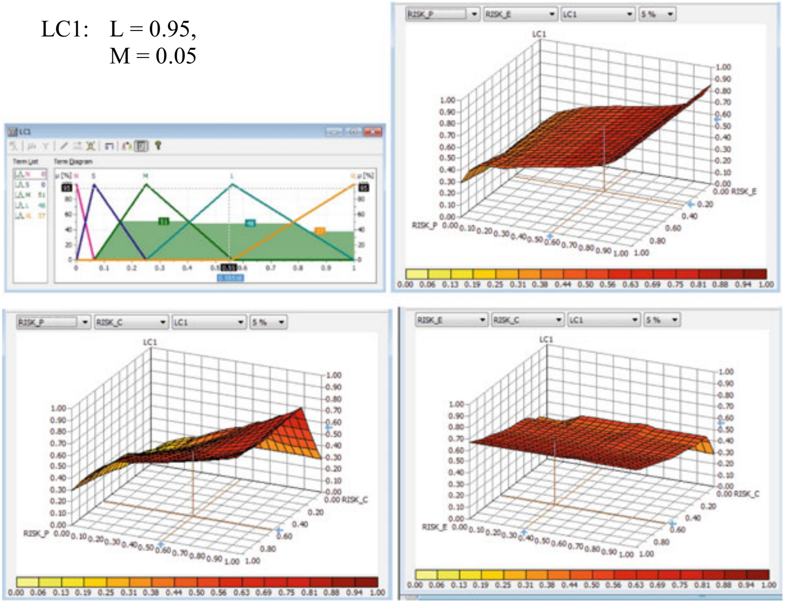


Fig. 40 Recommended efforts to strengthen Lithuanian national identity (NI)

LP1: L = 0.95,
M = 0.05

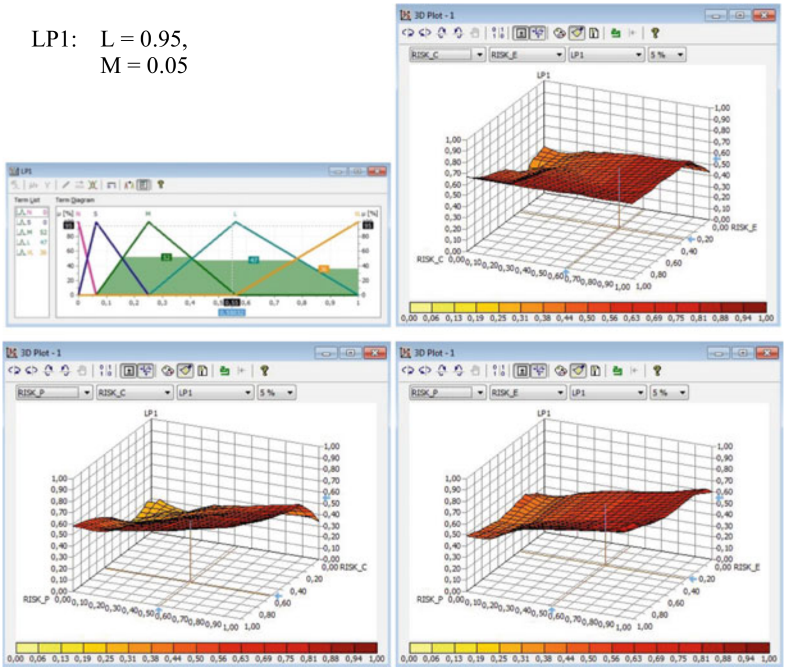


Fig. 41 Recommended efforts to create efficient EU border control (BC)

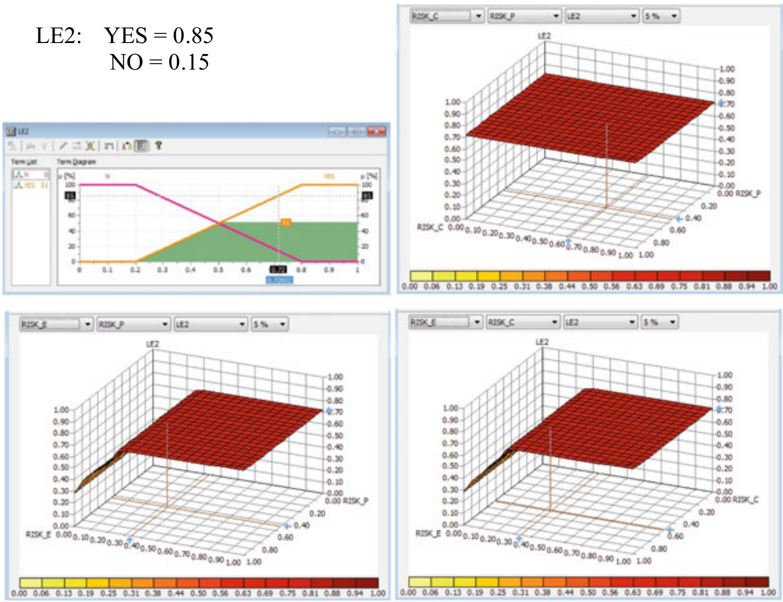


Fig. 42 Recommendation to open national commercial bank (CB)

An example of the case analyzed in the context of the proposed paradigm here was presented the assessment of opportunities and threats of such an entity as a state of Lithuania, to determine the state's risks and to generate optimal recommendations, actions and leverages for state's control. The viability of this paradigm and the successful demonstration of a solution to a complex situation (examination of the state's global management) allow us to say that it is on the basis of this paradigm that software tools for the management of specific situations can be created.

Table 9 The summarized results of LEVERAGES proposed using this DSS paradigm for the case of Lithuania

Leverages		Numerical	Verbal	Certainty μ
Inputs	RISK-C	0.6153	L	0.48
			M	0.51
	RISK-P	0.5347	L	0.37
			M	0.62
	RISK-E	0.3359	L	0.11
			M	0.88
Outputs—Leverages	LC1	0.5552	L	0.95
			M	0.05
	LC2	0.4943	YES	0.49
			NO	0.5
	LC3	0.5638	VL	0
			L	1
	LP1	0.5552	L	0.95
			M	0.05
	LP2	0.5552	L	1
			M	0
	LP3	0.5606	VL	0
			L	1
	LP4	0.633	VL	0.18
			L	0.82
	LE1	0.5105	L	0.82
			M	0.18
	LE2	0.7200	YES	0.85
			NO	0.15
	LE3	0.5552	L	1
			VL	0

Acknowledgements The work belongs to the series of promising research works of the Center of Real Time Computer Systems (CRTCS) of the Kaunas University of Technology and for this reason, the authors thank the working group of this Center and its management. In addition, the authors express their deep gratitude for the consultations with the adviser to the President of Lithuania ambassador A. Skaisgiryte, and to the members of the round table discussions of the Lithuanian Ambassadors' Club, led by ambassador A. Rimkūnas.

References

1. Ittmann, H.: Decision support systems (DSS): a survey. S.-Afr. Tydskr. Bedryfsl. **1**S (4), 189–196 (1984). http://pen.ius.edu.ba
2. Qinxia, H., Shah, N., Xiaoxu, G., Li, M., Muhammad, I.: A Review on Multicriteria Decision Support System and Industrial Internet of Things for Source Code Transformation, vol. 2021 | Article 2021, ID 6661272 | https://doi.org/10.1155/2021/6661272
3. Fentahun, M.K., Glen, B., Walker, A.: Decision support systems in manufacturing: a survey and future trends. J. Model. Manag. **12**(3), 432–454 (2017). www.emeraldinsight.com/1746-5664.htm; Emerald Publishing Limited, 1746–5664. https://doi.org/10.1108/JM2-02-2016-0015
4. Shahmoradi, L., Asieh, S., Niloofar, M., Marsa, G.: Designing and evaluating a decision support system on Childhood Leukemia to improve medication management. Appl. Health Inf. Technol. **1**(1), 1–10 (2020)
5. Kamran, F., Bisma, S.K., Muaz, A.N., Stephen, J.L.: Clinical decision support systems: a visual survey. Informatica **42**, 485–505 (2018). https://www.researchgate.net/publication/332179909
6. Musbah, J.A., Omar, A.N., Ayodeji, A.: Decision Support Systems Classification in Industry, Periodicals of Engineering and Natural Sciences, vol. 7, No. 2, pp. 774–785, Aug. 2019. ISSN 2303–4521. http://pen.ius.edu.ba
7. Macher, C., Steins, A.N., Ballesteros, M., Kraan, M., Frangoudes, K., Bailly, D., Bertignac, M., Colloca, F., Fitzpatrick, M., Garcia, D., Little, R., Mardle, S., Murillas, A., Pawlowski, L., Philippe, M.,Prellezo, R., Sabatella, E., Ulrich, O. T.: Towards transdisciplinary decision-support processes in fisheries: experiences and recommendations from a multidisciplinary collective of researchers. Aquat. Living Resour. **34**, 13 EDP Sciences 2021 (2021). https://doi.org/10.1051/alr/2021010
8. Ojha, V., Abraham, A., Snasel, V.: Heuristic Design of Fuzzy Inference Systems: A Review of Three Decades of Research, Engineering Applications of Artificial Intelligence (85), pp. 845–864. doi.org/https://doi.org/10.1016/j.engappai.2019.08.010
9. Billis, A.S., Papageorgiou, E.I., Frantzidis, C.A., Marianna, S., Tsatali, M.S., Tsolaki, A.C., Bamidis, P.D.: A decision-support framework for promoting independent living and ageing well. IEEE J. Biomed. Health Inform. **19**(1), 199–209 (2015). https://doi.org/10.1109/JBHI.2014.2336757
10. Lesauskaite, V., Damuleviciene, G., Knasiene, J.; Kazanavicius, E., Liutkevicius, A., Janaviciute, A.: Older adults–potential users of technologies // Medicina. Basel: MDPI AG, vol. 55, no. 6, art. no. 253, p. 1–9 (2019). ISSN 1010–660X. eISSN 1648–9144, https://doi.org/10.3390/medicina550602
11. Chrysostomos, D.S., Voula, C.G.: Medical Decision Support Systems based on Soft Computing techniques, Preprints of the 18th IFAC World Congress Milano (Italy), 6pp., Aug. 28–Sept. 2 2011
12. Mannina, G., Taise, R., Alida, C., Karina, G.: Decision support systems (DSS) for wastewater treatment plants—a review of the state of the art. Biores. Technol. **290**, 121814 (2019). https://doi.org/10.1016/j.biortech.2019.121814
13. Aqel, M., Nakshabandi, O.: Decision Support Systems Classification in Industry, Periodicals of Engineering and Natural Sciences (PEN), Aug. 2019. https://doi.org/10.21533/pen.v7i2.550, https://www.researchgate.net/publication/342788248
14. Hillson, D.: Effective Opportunity Management for Projects: Exploiting Positive Risk, p. 316. Marcel Dekker Inc., New York (2004)
15. Petrauskas, V., Jasinevicius, R., Kazanavicius, E, Meskauskas, Z.: Concept of a system using a dynamic SWOT analysis network for fuzzy control of risk in complex environments, mathematics and computer science (MCS). Math. Comput. Sci. **5**(2), 42–55 (2020). https://doi.org/10.11648/j.mcs.20200502.11 (ISSN Print: 2575-6036; ISSN Online: 2575-6028)
16. Meskauskas, Z., Jasinevicius, R., Kazanavicius, E., Petrauskas, V.: XAI-Based fFuzzy SWOT maps for analysis of complex systems. In: 2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE): Proceedings of 2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE) IEEE Catalog Number: CFP20FUZ-ART, 8pp. ISBN: 978–1–7281–6932–3

17. Petrauskas, V., Jasinevičius, R., Kazanavicius, E., Meskauskas, Z.: CWW elements to enrich SWOT analysis. J. Intell. Fuzzy Syst. **34**(1), 307–320 (2018)
18. Petrauskas, V., Damuleviciene, G., Dobrovolskis, A., Dovydaitis, J., Janaviciute, A., Jasinevicius, R., Kazanavicius, E., Knasiene, J., Lesauskaite, V., Liutkevicius, A., Meskauskas, Z.: XAI-based medical decision support system model // Int. J. Sci. Res. Publ. New Delhi: IJSRP Inc. **10**, no. 12, 598–607, p10869 (2020). ISSN 2250–3153. https://doi.org/10.29322/IJSRP.10.12.2020
19. Axelrod, R.: Structure of Decision: the Cognitive Maps of Political Elites. Princeton University Press, Princeton, NJ (1976)
20. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
21. Zadeh, L.A.: Fuzzy algorithms. Inf. Control **12**, 94–102 (1968)
22. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning. Inf. Sci. **8**, 43–80 (1975)
23. Kosko, B.: Fuzzy cognitive maps. Int. J. Man Mach. Stud. **24**, 65–75 (1986)
24. Kosko, B.: Fuzzy Thinking: the New Science of Fuzzy Logic. Flamingo, London (1994)
25. Kosko, B.: Fuzzy Engineering. Prentice-Hall, N.J. (1997)
26. Carvalho, J.P., Tome, J.A.: Fuzzy mechanisms for causal reasoning. In: Proceedings of the Eighth International Fuzzy Systems Association World Congress, IFSA'99 Taiwan, pp. 1009–1013 (1999)
27. Carvalho, J.P., Tome, J.A.: Interpolated linguistic terms. In: IEEE Annual Meeting of the Fuzzy Information. Processing NAFIPS'04 vol. 1, pp. 151–156. IEEE (2004)
28. Kahn, M.S., Quaddus, M.: Group decision support using fuzzy cognitive maps for causal reasoning. Group Decis. Negot. J. **13**(5), 463–480 (2004)
29. Xirogiannis, G., Stefanou, J., Glykas, M.: A fuzzy cognitive map approach to support urban design. J. Expert Syst. Appl. **26**(2), 257–268 (2004)
30. Xirogiannis, G., Glykas, M., Staikouras, C.: Fuzzy cognitive maps as a back end to knowledge-based systems in geographically dispersed financial organizations. Knowl. Process Manag. **11**(2), 137–154 (2004)
31. Papageorgiou, E.I.: Review Study on Fuzzy Cognitive Maps and Their Applications during the Last Decade, Jan. 2013. In book: Business Process Management, pp. 281–298 (2013). https://doi.org/10.1007/978-3-642-28409-0_11
32. Konar, A.: Computational Intelligence: Principles, Techniques and Applications. Springer (2005)
33. Lin, C.-T., Lee S. G.: Neural Fuzzy Systems. Prentice Hall (1996)
34. Passino, P.M, Jurkovich, S.: Fuzzy Control. Addison-Wesley (1998)
35. Maringer, D.: Heuristic optimization for portfolio management. IEEE Comput. Intell. **3**(4), 31–34 (2008)
36. Brabazon, A., O'Neil.: Biologically Inspired Algorithms for Financial Modelling. Springer (2005)
37. Berner, E.S. (ed.): Clinical Decision Support Systems: Theory and Practice. Springer, New York (1999)
38. Schrodt, P.: Patterns, Rules and Learning: Computational Models of International Behaviour, Vinlard, Kansas, USA (2004)
39. Aguilar, J.: A survey about fuzzy cognitive maps papers (Invited Paper). Int. J. Comput. Cogn. **3**(2), 27–33 (June2005)
40. Goward, D.A.: Maritime Domain Awareness the Key to Maritime Security. IAC Luncheon, US Coast Guard Maritime Domain Awareness, 23 May 2006. http://www.actgov.org/actiac/documents/pptfiles/060523DanaGoward.ppt
41. Beaton, S.: Maritime Security & Maritime Domain Awareness. Infra Gard 2005 National Conference, Hosted by the InfraGard National Members Alliance and the FBI, 9 Aug. 2005. http://www.infragard.net/library/congress_05/maritime_port/port_sec.ppt 37, C
42. Li, H., Chen, P., Huang, H-P.: Fuzzy Neural Intelligent Systems: Mathematical Foundations and the Applications in Engineering. RCA Press LLC (2001)
43. Lin, C.-T., Lee, C.S.G.: Neural Fuzzy Systems. Prentice Hall (1996)

44. Mohr, T.S.: Software Design for a Fuzzy Cognitive Map Modelling Tool, Master's Project 66.698 Rensselaer Polytechnic Institute, 19p. (1997)
45. Jasinevicius, R., Petrauskas, V.: The new tools for systems analysis // Informacinės Technologijos ir valdymas = Information Technology and Control, nr. 2(27). p. 51–57/Kauno Technologijos Universitetas (2003). ISSN 1392–124X
46. Jasinevicius, R., Petrauskas, V.: Dynamic SWOT Analysis as a Tool for System Experts. Engineering Economics, No. 5(50), pp. 33–35/Kaunas university of technology. Technologija, Kaunas (2006). ISSN 1392–2785
47. Jasinevicius, R., Petrauskas, V.: Fuzzy expert maps: the new approach// WCCI 2008 Proceedings: 2008 IEEE World Congress on Computational Intelligence, 1–6 June 2008, Hong Kong: 2008 IEEE International Conference on Fuzzy Systems. 2008 IEEE International Joint Conference on Neural Networks.2008 IEEE Congress on Evolutionary Computation. Piscataway: IEEE, pp. 1511–1517 (2008). ISBN 978 – 1 - 4244–1819–0
48. Jasinevičius R., Petrauskas V.: Dynamic SWOT analysis as a tool for environmentalists // Environmental Research, Engineering and Management, No. 1(43), pp. 14–20. Technologija, Kaunas (2008)
49. Jasinevicius, R., Petrauskas, V.: Fuzzy expert maps for risk management systems // US/EU-Baltic 2008 International Symposium: Ocean Observations, Ecosystem-based Management & Forecasting, 27–29 May 2008, Tallin, Estonia. Piscataway: IEEE (2008). ISBN 978-1-4244-2268-5
50. Jasinevicius R.: Fuzzy inference tools for decision makers // ISAGA 2008: the 39th Conference International Simulation and Gaming Association: Games: Virtual Worlds and Reality: 7–11 July 2008, Kaunas, Lithuania: Conference book. Kaunas: Technologija, p. 28 (2008). ISBN 978 - 9955–25–528–4
51. Jasinevicius, R., Petrauskas, V.: Rule-based extensions of fuzzy cognitive maps for decision support systems // Information Technologies' 2008: Proceedings of the 14th International Conference on Information and Software Technologies, IT 2008, Kaunas, Lithuania, 24–25 Apr. 2008/Kaunas University of Technology, pp. 72–77 (2008). ISSN 2029–0020
52. Jasinevicius, R., Krusinskiene, R., Petrauskas, V., Tkaciov, A.: Dynamic fuzzy expert maps: idea and implementation. In: Information Technologies' 2011: Proceedings of the 17th International Conference on Information and Software Technologies, IT 2011, Kaunas, Lithuania, 27–29 Apr. 2011, pp. 17–22 (2011)
53. Jasinevicius, R., Petrauskas, V.: On fundamentals of global systems control science (GSCS). In: Sanayei, A., Zelinka, I., Rössler O. (eds.), ISCS 2013: Interdisciplinary Symposium on Complex Systems. Emergence, Complexity and Computation, vol. 8, 77–86pp. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-45438-7_8
54. Gurel, E., Tat, M.: SWOT analysis: a theoretical review. J. Int. Soc. Res. **10**, 994–1006 (2017)
55. Balzekiene, A., Gaule, E., Jasinevicius, R., Kazanavicius, E., Petrauskas, V.: Risk evaluation: the paradigm and tools. In: Dregvaite, G., Damasevicius, R. (eds.), Information and Software Technologies. ICIST 2015.Communications in Computer and Information Science, vol. 538, Springer, Cham, pp. 330–342 (2015)
56. Šotic, A., Rajic, R.: The review of the definition of risk. Online J. Appl. Knowl. Manag. **3**(3), 17–26 (2015)
57. Atanassov, K.T.: On Intuitionistic Fuzzy Sets Theory. Springer, New York, NY (2012)
58. Chen, L.-H., Tu, C.-C.: Dual bipolar measures of Atanassov's intuitionistic fuzzy sets. IEEE Trans. Fuzzy Syst. **22**(4), 966–982 (2014)
59. Jasinevičius R., Petrauskas V.:Sprendimų pagrindimo kompiuterizavimas (Computerization of decision making), Kaunas, Lithuania: Technologija, p. 156 (2011)
60. Liao, H., Mi, X., Xu, Z., Xu, J., Herrera, F.: Intuitionistic fuzzy analytic network process. IEEE Trans. Fuzzy Syst. **26**(5), 2578–2590 (2018)
61. Herrera-Viedma, E., Cabrerizo, F.J., Kacprzyk, J., Pedrycz, W.: A review of soft consensus models in a fuzzy environment. Inf. Fusion **17**, 4–13 (2014)
62. Xu, Z.: Hesitant fuzzy sets theory. Studies in Fuzziness and Soft Computing. Springer (2014)
63. https://europeanvaluesstudy.eu/

64. https://ec.europa.eu/eurostat/portal/page/portal/culture/documents/AVERAGE_ANNUAL_CULTURAL_EXPENDITURE_PER_HOUSEHOLD.pdfEurostat, Cultural statistics, Average annual cultural expenditure per household
65. Dreher A.: KOF Index of Globalization, Zurich (2010). http://globalization.kof.ethz.ch
66. International Energy Agency Website: www.iea.org. Lithuania 2021 Energy Policy Review 172 pp.
67. https://www.lrp.lt/en/news/the-foreign-policy-coordination-council-discussed-lithuanias-key-objectives-in-foreign-policy-in-2021/35343. The Foreign Policy Coordination Council discussed Lithuania's key objectives in foreign policy in 2021
68. Integrated Country Strategy Lithuania May 8, 2020, 17 p.p. https://www.state.gov/wp-content/uploads/2020/08/ICS_EUR_Lithuania_Public-Release.pdf
69. All the Strategy related information is available at: www.Lietuva2030.lt and social network Facebook (www.facebook.com/Lietuva2030)
70. http://info.worldbank.org/governance/wgi/index.asp
71. Lithuania's Progress Strategy "Lithuania 2030". https://lrv.lt/uploads/main/documents/files/EN_version/Useful_information/lithuania2030.pdf
72. APPROVED by Resolution No 1281 of the Government of the Republic of Lithuania of 18 December 2013 "The Lithuanian Innovation Development Programme 2014–2020", Ministry of the Economy and Innovation of the Republic of Lithuania, 27 pp.
73. Ministry of National Defense of the Republic of Lithuania "Lithuanian Defence System: facts and figures 2020" 12pp. (2020)
74. Whitepaper, Lithuanian Defence Policy, Ministry of National Defence of the Republic of Lithuania, Vilnius 64 pp. (2017)
75. Jasinevicius, R., Petrauskas, V.: The new tools for systems analysis. II Inf. Technol. Control **2**(27), 51–57 (2003)
76. Ministry of Education, Science and Sport, "Goals and objectives of the Ministry of Education, Science and Sport", "Agreement on National Education Policy" (2021–2030)
77. https://www.fuzzytech.com; fuzzyTECH 8.62f; 2019.09.03

Stock Portfolio Risk-Return Ratio Optimisation Using Grey Wolf Model

Virgilijus Sakalauskas, Dalia Kriksciuniene, and Audrius Imbrazas

Abstract In today's high inflation environment, it is very important to protect our capital from depreciation. One way to preserve your capital is to try to invest it in stocks anticipating for big return. But sometimes the expectations may fail due to underestimating the risk level of stock portfolio investments. The objective of this work is to develop a risk-return ratio optimization model for stock portfolio enabling to screen the adequate equities for inclusion to investment portfolio and set its capital allocation ratio. We propose a two-stage model, firstly enabling to select the initial set of equities by applying Self-Organizing Maps (SOM) by identifying a set of most influential factors to use as input variables for SOM. The method for the second stage of research is proposed for deciding the weight-based ratios for capital to be distributed among the portfolio equities. The nature-inspired Grey Wolf Optimization (GWO) algorithm is applied for finding the optimal weights allocation among the portfolio shares based on Mean−Variance portfolio minimization condition, which correspondingly define the Risk and Return rating of portfolio. The sensitivity of the GWO algorithm to the number of iterations, wolf herd size and stocks weight limits was investigated for defining optimal values of these parameters for the best portfolio diversification. The experimental verification of the model was performed on stock set from S&P500 companies. The proposed model based on SOM selected and GWO balanced portfolio outperformed the direct investment to S&P500 index by 3.52% higher profitability.

Keywords Grey Wolf Optimization (GWO) algorithm · Self-organizing maps (SOM) · Portfolio diversification · Mean/Variance stock portfolio selection

V. Sakalauskas (✉) · D. Kriksciuniene · A. Imbrazas
Institute of Social Science and Applied Informatics, Vilnius University, Universiteto Str.3, Vilnius, Lithuania
e-mail: virgilijus.sakalauskas@knf.vu.lt

D. Kriksciuniene
e-mail: dalia.kriksciuniene@knf.vu.lt

G. Dzemyda et al. (eds.), *Data Science in Applications*,
Studies in Computational Intelligence 1084,
https://doi.org/10.1007/978-3-031-24453-7_10

1 Introduction

According to the Cambridge Dictionary, an investment is the employment of money for the profit. In other words, an investment is the use of available money or other resources for future benefits. In today's world, there are many different investment opportunities.

Active employment of money takes place when a business is being created. Passive money can be employed with the help of a bank, when buying long-term government bonds or company shares, transfer own capital to investment funds or acquire financial derivatives. However, by choosing one asset class or only single security, the investor is exposed to high risk and his success depends entirely on the success of the chosen security. When making financial decisions, investors tend to maximize returns and control risk. For higher profits, the level of risk is always higher. An investment portfolio is required to ensure balance among risk and return. According to the business glossary, an investment portfolio is a set of different investment assets that an investor expects to earn a profit from and seeks to preserve the capital invested.

The portfolios can be rated from low risk—low return to high risk—high return. An individual portfolio may be designed by random selection decisions or it may be the result of careful, responsible planning. Thus, one of the solutions for reducing the risk that any individual may face is to diversify money, thus optimizing the investment portfolio.

To properly manage your assets, we need to understand portfolio management processes. Investment management is a complex activity that can be defined in eight steps [3]:

1. ***Specification of investment objectives and restrictions***. Typical goals pursued by investors are current income, capital gains, and security of the principal amount invested. The investor should rank these goals in order of importance. In addition, the investor must evaluate possible profit constraints due to liquidity, period, taxes and other specific circumstances.

2. ***Quantify capital market expectations***. In order to distribute the available capital fairly, it is necessary to compare the return and risk ratios of different asset classes. When allocating capital, market expectations should be quantified.

3. ***Decide which asset classes will be included in the portfolio***. The most important decision in portfolio management is to decide which asset classes to invest in. This even includes deciding what proportions to invest, such as 70% to shares and 30% to bonds. It depends on the investor's personal risk tolerance.

4. ***Portfolio strategy formation***. Once it has been decided in which asset classes to invest, then the right portfolio management strategy must be chosen. There are two different main portfolio management strategies—active and passive. An active portfolio strategy seeks to achieve higher returns by taking into account changes in the asset class sector and by continuously adjusting the portfolio itself. While the passive, meanwhile, offers a good distribution of the portfolio, minimizing risk as often without changing the composition of the portfolio.

5. ***Selection of securities***. The investor should choose the securities by applying wisdom and knowledge-based criteria. Funding is usually based on fundamental or technical analysis.

6. ***Portfolio implementation***. In this step of portfolio management, the investor, having performed previous actions and analyses, must implement this by acquiring selected securities and other financial instruments.

7. ***Portfolio revision***. The value of a portfolio depends on its components, which may fluctuate due to price movements in financial instruments. As preliminary analyses may not work, it is necessary to review and rebalance the portfolio at certain intervals.

8. ***Portfolio valuation***. The activities performed with the portfolio and its changing content must be evaluated periodically. The key aspects of assessing the performance of a portfolio are risk and return, and the main assessment criterion is whether the return on the portfolio is proportionate to its risk.

It is useful to define the position of each investment instrument according to its risk and possible reward (see Fig. 1).

The types of portfolio are characterized as conservative, moderate and aggressive (Fig. 1). Due to the large number and diverse range of suggested investment strategies, many researchers chose the approach to evaluate and categorize different financial instruments according to the ratio of risk to return on investment.

There is no such thing as a risk-free investment. Risks affect both people and businesses. The portfolio diversification effect is designed to manage the expected risk. Individual risk reduction occurs by combining several assets and forming a

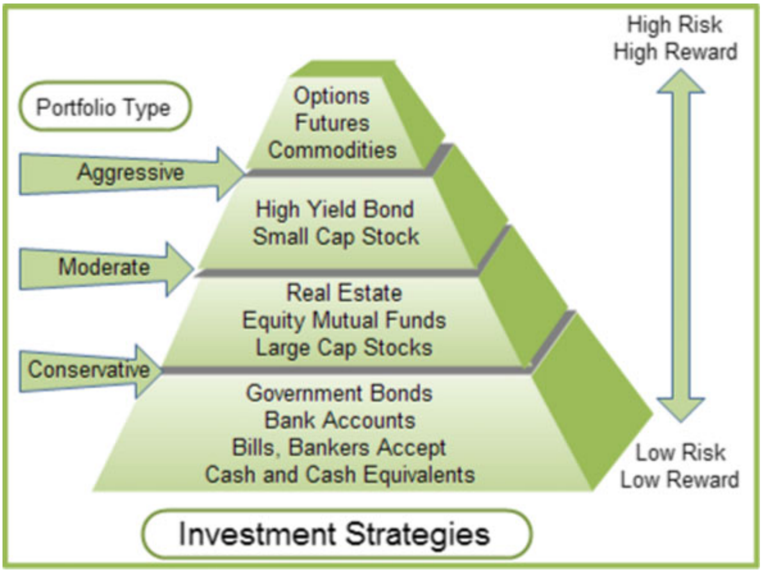


Fig. 1 Portfolio types by investment instruments

portfolio. An investor becomes exposed to general asset risk when his portfolio consists of a single asset. It is important to understand the individual risk of the asset in order to understand how the risk is distributed in the portfolio. According to [24], diversifying a naive portfolio without using any mathematical optimization model and buying stocks in equal parts, is a simple and powerful way to effectively reduce portfolio risk without losing the expected rate of return. The results of the study showed that for an infinite set of assets, a portfolio of a maximum of 20 assets would be sufficient to eliminate 95% of the assets non-systemic risks.

When the list of stocks at a stock exchange is large, the investors face problems of selecting the profitable stocks for the portfolio. However, when stock exchanges are small, the portfolio may consist of all listed shares. Either way, there can be many different portfolio combinations. Therefore, the use of active selection methodology is mandatory [10, 25].

Investing in a portfolio rather than a separate asset is gaining people's attention more and more because of its ability to reduce risk and optimize reward. The decisions which the investors make for designing an exceptionally good portfolio may be based on various methods and models and significantly affect the performance of the portfolio.

In the Sect. 2, we will introduce the Mean–Variance (M-V) based portfolio selection problematics.

2 Mean-Variance (M-V) Based Portfolio Selection

When making financial decisions, investors tend to maximize returns and minimize the risk. The risk level is always higher for higher expected profits. A good investment portfolio is required to have a balance between risk and return. Usually portfolio selection is based on different Risk-Return ratios.

Diversification of the investment portfolio was described by H. M. Markowitz [15]. The usual diversification strategy is based on including the securities from different sectors, companies or countries [4]. H. Markowitz combined probability and optimization theories to model the investors' behaviour. He stated that the return on investment should depend on the investor's expected earnings, taking into account possible price volatility. The Mean–Variance (M-V) based portfolio selection means the sensible balance between portfolio risk and return. The M-V model solves the problem of portfolio selection in order to find the best securities suitable for inclusion in the portfolio. By selecting the weights for portfolio assets it's possible to form the portfolio with the lowest risk and maximum return values [14].

The Mean-Variance (M-V) portfolio optimization theory of H. Markowitz [15] helps to set a portfolio weights that provide the optimal tradeoff between the mean (as a measure of profit) and the variance (as a measure of risk). The standard M-V optimization problem can be expressed as an optimization model where the solution timely maximizes expected return and minimizes portfolio variance.

Suppose an investment portfolio consists of n assets with expected return $R = \{r_i\}_{(i=1)}^n$, and covariance matrix $K = \{\sigma_{ij}\}_{(i,j=1)}^n$ Let $X = \{x_i\}_{i=1}^n$ stands for initial investment proportion (weights) of corresponding asset, such as $\sum_{i=1}^n x_i = 1$. The portfolio return is equal $R(x) = X^T \cdot R$ and variance $V(x) = X^T \cdot K \cdot X$.

Given a fixed target value of portfolio return $\hat{R}$, Markowitz characterizes an efficient portfolio by the weights vector $\hat{X}$, that minimizes the risk $V(x)$ subject to return $R(x) = \hat{R}$.

When looking for the optimal portfolio under the M-V method, the average cost of each asset and the covariance between each pair of assets are included in the calculations. These calculations are always based on historical data. However, the M-V method can lead to an inaccurate forecast because of:

1. ***Multi-period investment***—the M-V method can show an inaccurate forecast based on a long-term, multi-cycle investment where premiums are periodic, e.g. pension accumulation, long-term investment funds [17].

2. ***Small data sample—***using too small sample of historical data may disregard the economic cycle and lead to statistically significant deviations in the mean and covariance calculations, which may result in only a few assets whose volatility was reflected in the portfolio [21].

3. ***Extremes***—assets with very large deviations from the average in the M-V model are automatically discarded and not given any weight in the portfolio. However, such premature rejection is applicable not only to highly unprofitable but also highly profitable assets, which causes investors to lose their ability to earn maximum returns [21].

For these reasons, the M-V model was upgraded to S-V model (Semi-Variance). This new model is more focused on stock returns, which may fall short of the normal distribution by including a skewness ratio [4].

Other researchers suggested to measure a risk not by covariance matrix, but to use the Variance at Risk (VaR) or Conditional Value at Risk (CVaR) [13]. These improvements of the model allow investors to create more reliable stock portfolio.

An alternative to the Mean−Variance (M-V) model is the Mean-Absolute Deviation (MAD) model, proposed by [9]. The M-V model assumes normality of stock returns, which is not always the case. The MAD model does not make this assumption. The MAD model minimizes a measure of risk—mean absolute deviation. MAD is easier to compute than Markowitz because it eliminates the need for calculating a covariance matrix. MAD portfolios typically have fewer shares—this reduces the transaction costs of changing the portfolio.

Markowitz and MAD methods are often criticized for equally treating the positive and negative deviations of mean, while investors desire for large positive deviations, but not negative.

These disadvantages do not have the MiniMax model [26]. Portfolio selection of MiniMax model is done by minimizing the maximal loss of historical observations. MiniMax model is appropriate for investors that seek to evade downside-risk. The author identifies Minimax model as not appropriate if the investors lack for historical return data.

Table 1 Suggested steps of portfolio screening

No	Steps	Description
1	The initial selection of equities from stock exchanges	Keeping in mind investors portfolio return preferences we use some restrictions on fundamental economic, financial or technical indicators letting us to get a set of not more than 100 shares
2	The final stock portfolio equities screening	This step can be made from selected equities using Kohonen's Self-Organizing Map (SOM) algorithm. It allows all equities distribute along the clusters depending on the chosen price and trade-related data factors. To form an investment portfolio, we suggest to select the shares from most adequate cluster we got using SOM. Usually not more than 20 shares
3	Allocation of investment capital to portfolio shares	For this task we suggest to employ the Grey Wolf Optimization algorithm, which let us find the optimal weights assignment along the portfolio shares based on Variance/Mean portfolio minimization condition

An analysis of the most popular traditional portfolio selection models shows that they give investors a theoretical probable result, but they do not always work well in practice. Because of the shortages in the models, scientists and investors continue to try to improve traditional models, or look for alternatives. The way we propose, is to combine them with increasingly popular method of screening the equities by the Self-Organizing Maps (SOM) and genetic optimization algorithms.

In this research we rely on screening the shares in the investment portfolio by the Self-Organizing Maps (SOM) and design the genetic optimization algorithm for determining optimal weights of the equities, allowing to get the highest return with minimal risk. The framework of this proposal can by split into 3 steps (Table 1).

The next section will discuss the selection of shares using well known Kohonen's Self-Organizing Map. The SOM algorithm [8] is a machine-learning approach that is generally used to classify the data according to the similarity between the data. In Sect. 4 we describe the Grey Wolf Optimization (GWO) algorithm and its application for finding the optimal weights along the portfolio shares based on Mean−Variance portfolio optimization condition. The verification of proposed method with real investment portfolio selection is outlined in Sect. 5. The paper finishes with Conclusions and Main Results section.

3 Self-Organizing Map

SOM denotes model of self-organizing maps that belongs to a common class of neural networks. They are used for organizing the data, revealing patterns or structures of data that are initially unknown.

SOM's seeks the representation of data, which is easy for human's analysis. Usually this representation is 2dimensional, can be plotted, and suitable for analysis. SOM has a very wide application area: image analysis [12], financial investments [11], speech recognition [2] etcetera. The results of research [7, 23] show that SOM can become a tool for classifying large amounts of stock exchange trading data.

According to the SOM algorithm the high dimensional input vectors X maps to bi-dimensional neuron space (map) depending on their characteristic features. It helps to understand high-dimensional data by grouping similar data together. A simple SOM consists of two layers—input and output space. A representation of SOM with output nodes in a two-dimensional grid view is provided in Fig. 2.

Each neuron *I(X)* has a prototype vector $W_{I(X)}$, which corresponds to a point in the input space. An input vector X will select the neuron with closest $W_{I(X)}$ to it. Adjustment of the weight vector for the winning output neuron and its neighbours is done through selecting the quantity of $\Delta W_{I(X)}$ [22].

This method is widely applied, many software tools have been developed for its implementation. Some of the most popular products that can perform self-organizing map (SOM) analysis are as follows:

- SAS neural network application;
- NeuralWorks Professional II + developed by NeuralWare;

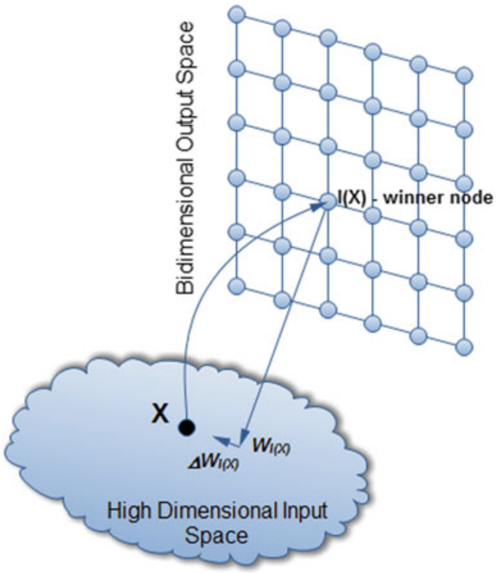


Fig. 2 SOM training algorithm

- MATLAB neural network tool;
- NeuroLab, which is adapted for Python programming language;
- havFmNet + + , which is adapted to JAVA programming language;
- Neural Connection;
- Trajan 2.1 Neural Network Simulator;
- Viscovery SOMine.

In our research we will take advantage of Viscovery SOMine (www.viscovery.net/somine/) software.

Silva and Marques [23] have also used SOM to construct an investment portfolio. For their analysis, they selected 1998 to 2009 historical price data for forty-nine stocks and calculated correlations between them and gold. SOM analysis was performed with netSOM software, normalized data was used and ten clusters were determined. Their analysis showed that self-organizing maps categorize securities according to their historical similarities. For example, all insurance companies fall into one cluster and financial companies fall into two clusters. In this case, SOM put together very similar and correlating companies in one cluster, so the authors decided to take one best share from each cluster when making their investment portfolio.

Khan et al. [7] used technical analysis indicators and self-organizing maps to identify profitable shares by putting them in one of the best clusters. Data was taken from the National Stock Exchange (NSE) over a two-month period using technical indicators such as MACD, Williams% R, RSI and others. The results were compared with the price of the NSE index over the same period. During this test, it was found that the shares of the best cluster selected gave 9.53%. higher yields compared to the NSE index.

In this research SOM algorithm was applied for stock portfolio equities screening across the clusters depending on the financial stock trading data. The investment portfolio is formed from the shares within most adequate cluster identified by SOM algorithm. The best practice-based advice may suggest to select no more than 20 shares.

4 Grey Wolf Optimization Algorithm

The GWO algorithm let us solve the optimisation problems by simulating the leadership hierarchy and hunting mechanism of grey wolves in nature. In 2014 developed GWO algorithm is one of the most popular and promising optimisation methods [1, 10, 18, 20, 27].

Grey wolves tend to live in groups. The average group size is from five to twelve wolves. The leadership hierarchy are implemented by four types of grey wolves: alpha (α), beta (β), delta (δ) and omega (ω).

The wolves herd leaders are male and female, who are called alpha. Alphas are primarily responsible for making decisions related to hunting, sleeping location, lifting time, and so on. Alpha solutions are mandatory for herd. Alpha is also called the dominant wolf because the herd has to listen to his / her instructions.

The second level in the grey wolf hierarchy is beta. These wolves are subordinate alpha and help him make decisions or perform other activities. Beta can be male or female, which are the best candidates for alpha ranks. The beta wolf has to respect alpha, but he also leads the lower level wolves. The grey beta wolf is an advisor to the alphas and a disciplinarian to the lower levels.

The third level of grey wolves is the delta. They obey alpha and betas, but lead to the lowest level—omega. This category includes wolves such as scouts, guards, hunters. Scouts are responsible for overseeing the area and alerting the herd to danger. The guards protect and guarantee the safety of the hard. Hunters help alpha and betas hunt prey and are responsible for feeding the herd [18].

The fourth level of grey wolves is omega. This is the lowest level of the herd of wolves. Omega must always conform all wolves at higher levels and they are the last to get a chance to eat. It may appear that omega wolves are not very important to the whole group, but the whole group suffering after losing them, as omega support is very important for all leading wolves. This helps satisfy the entire herd and maintain a dominant structure of alpha, beta and delta wolves [5].

Grey wolves are always in herd, as well as in hunting. However, for the hunt to run smoothly, this phenomenon has its own phases. It is shown that two simple rules controlling the movement of each wolf are enough to reproduce the main features of the wolf-herd hunting behaviour: tracking the prey, carrying out the pursuit, and encircling the prey until it stops moving. The rules are [19]:

1. move towards the prey until a minimum safe distance to the prey is reached
2. when close enough to the prey, move away from the other wolves that are close to the safe distance to the prey.

Some scientists say grey wolves demonstrate the ability to be in ambush and predict future events. They also understand complex relationships and, using this ability of their own, are able to plan and consciously and consistently follow it to achieve a goal. In hunting sessions, such mental processes are manifested in the ability to supposedly pass information to another wolf squatting in ambush, and to realize while waiting in ambush that this improves his chances of approaching the prey [16].

The mathematical model of hunting strategy also turns on the optimisation steps—searching the prey, encircling prey, attacking prey.

In order to use GWO and perform optimisation the hunting strategy and social hierarchy of grey wolves are mathematically modelled.

Social hierarchy of wolves mathematically indicate the most fitted optimisation solutions. The best solution is α, second and third suitable solutions are marked β and δ respectively, the rest solutions are denoted by ω. So, the GWO algorithm (optimisation) is under control by alpha, beta and delta solutions and the omega solution changes according the most optimal solutions α, β and δ [6].

During the hunting grey wolves try to encircle the prey. Let's denote the grey wolf position vector in time t as $\overrightarrow{X(t)}$ and let $\overrightarrow{X_p(t)}$ stands for a pray position. The grey wolves position changes on time $t + 1$ can be calculated using the equation:

$$\overrightarrow{X(t+1)} = \overrightarrow{X_p(t)} - \vec{A} \cdot \vec{D} \quad (1)$$

where $\vec{D} = \left|\vec{C} \cdot \overrightarrow{X_p(t)} - \overrightarrow{X(t)}\right|$-distance to prey, and $\vec{A} = 2a \cdot \overrightarrow{r_1} - a$ – is a coefficient vector. Here $\vec{C} = 2\overrightarrow{r_2}$ and $\overrightarrow{r_1}$, $\overrightarrow{r_2}$ denotes random vectors in [0, 1]. The a coefficient is linearly decreasing from 2 to 0 at each iteration step and can be calculated by the formula $a = 2 - t\left(\frac{2}{T}\right)$, where T stands for expected maximal iterations number.

Initially the wolves randomly arrange themselves around the pray, as the exact position of the prey is unknown. According to the GWO algorithm, initial solutions are chosen randomly. Then, we fix the 3 most accurate solutions α, β and δ and recalculate the other solutions according to the values of the leaders:

$$\overrightarrow{X(t+1)} = \frac{\overrightarrow{X_1(t)} + \overrightarrow{X_2(t)} + \overrightarrow{X_3(t)}}{3} \quad (2)$$

where $\overrightarrow{X_1(t)} = \overrightarrow{X_\alpha(t)} - \overrightarrow{A_1} \cdot \overrightarrow{D_\alpha}$; $\overrightarrow{X_2(t)} = \overrightarrow{X_\beta(t)} - \overrightarrow{A_2} \cdot \overrightarrow{D_\beta}$;$\overrightarrow{X_3(t)} = \overrightarrow{X_\delta(t)} - \overrightarrow{A_3} \cdot \overrightarrow{D_\delta}$, are calculated depending on leaders $\overrightarrow{X_\alpha(t)}$, $\overrightarrow{X_\beta(t)}$, $\overrightarrow{X_\delta(t)}$ positions, $\vec{A}$ and $\vec{C}$ as in (1),
$\overrightarrow{D_\alpha} = \left|\overrightarrow{C_1} \cdot \overrightarrow{X_\alpha(t)} - \overrightarrow{X(t)}\right|$; $\overrightarrow{D_\beta} = \left|\overrightarrow{C_2} \cdot \overrightarrow{X_\beta(t)} - \overrightarrow{X(t)}\right|$; $\overrightarrow{D_\delta} = \left|\overrightarrow{C_3} \cdot \overrightarrow{X_\delta(t)} - \overrightarrow{X(t)}\right|$ So, only the α, β and δ wolves-solutions estimates the optimal pray-solution.

For our model, we use the risk and return ratio of assets as a fitness function for the grey wolf algorithm and constrain the initialized wolf pack vector with number of dimensions equal to number of assets in portfolio.

Our approach to use the GWO for setting the proportion of capital to assets of chosen portfolio can be highlighted by pseudo code in Fig. 3.

To finalize the GWO method description for evaluation of assets weights, we need to define the Mean−Variance fitness function. This can be done according the Mean−Variance (M-V) portfolio optimization theory of Harry Markowitz explained in Sect. 2. It is worth to recall that the portfolio return mean we defined $R(x) = X^T \cdot R$ and variance as $V(x) = X^T \cdot K \cdot X$, where X stands for initial investment proportion (weights) to corresponding asset, such as $\sum_{i=1}^{n} x_i = 1$.

The proposed fitness function for GWO algorithm can be expressed as follows:

$$\begin{cases} Find\, the\, weights \{x_i\}_{i=1}^{n} which\, minimizes\; F(x) = \frac{V(x)}{R(x)} \\ subjectto \sum\limits_{i=1}^{n} x_i = 1, 0 \le x_i \le 1 \end{cases} \quad (3)$$

It is worth noting that the efficiency of the GWO algorithm highly depends on the parameters of the method: the number of shares selected, the number of iterations and the capital limit per share. The limitation of the maximum capital allocated to one

Initialise the grey wolf population $\overrightarrow{X_i}$ *vectors (dimension is equal to number of assets in portfolio)*
Initialise the a, A, and C parameters
Calculate the Mean-Variance fitness function for each search agent (wolf)
Assign three best search agent values to $\overrightarrow{X_\alpha(0)}, \overrightarrow{X_\beta(0)}, \overrightarrow{X_\delta(0)}$ *respectively*
***While** (t < T - Max number of iterations)*
    ***For** each search agent*
        Update the position of the current search agent by formula (2)
    End for
    Update a, A, and C
    Calculate the Mean-Variance fitness function for all search agents
    Update the vectors $\overrightarrow{X_\alpha(t)}, \overrightarrow{X_\beta(t)}, \overrightarrow{X_\delta(t)}$ *to the best search agent values*
    t=t+1
end while
return $\overrightarrow{X_\alpha(T)}$

Fig. 3 Pseudo code of GWO algorithm application

share is necessary as otherwise the investment will be distributed to a small number of the most profitable shares with high risk level. The numeric example presented in the next section investigate the influence of these parameters on optimal weights selection.

We have tested GWO performance for 3, 6, 12, 30, 50, 100 wolves herd, by using the 10, 20, 30, 50, 100 iterations and verify the equities weight limits (capital allocation) to 0.2, 0.3, 0.4, 0.5.

Our investigation highlighted the optimal values of the parameters allowing to create a profitable portfolio in the presence of adequate risk level.

5 Simulation (Numerical) Experiment

This section will conduct an experimental study for selecting stocks from S&P500 index using the SOMine software package and introduce a weight-optimized investment portfolio by finding the proportion of capital allocation to shares.

The selection of candidates to our research portfolio we will perform in two steps. Firstly, we will filter out a small number of stocks from the S&P500 index taking into account the initial public offering (IPO) date (no earlier than 5 years ago); P/E (not higher than 20) and BETA value (between 1 and 2). Surely, in this step, also other restrictions can be applied to filter the stocks.

After application this procedure, we have selected 54 shares. Because so many shares are still too much for the portfolio, we suggest to apply Viscovery SOMine (www.viscovery.net/somine/) software to cluster the shares screened in step 1. This step allows not only to reduce the number of stocks in the portfolio, but also ensures the similarity of the selected stocks according to certain criteria as we select stocks from the same cluster. We suggest clustering shares by the similarity of factors: P/E,

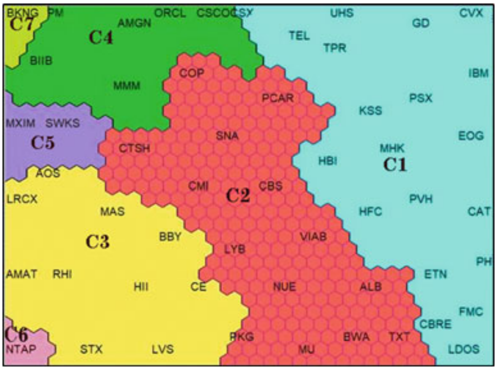


Fig. 4 SOM clusters

ROA, P/B, BETA, market capitalization, Earnings Per Share (EPS), ROI, liquidity ratio and profit margin.

By applying Viscovery SOMine we got seven clusters (Fig. 4). The selected S&P500 companies stock symbols are seen directly on the figure (see https://en.wikipedia.org/wiki/List_of_S%26P_500_companies).

As we can see, the clusters differ not only in size but also in the factors values of the shares represented. Our goal is to select such a cluster in which the number of stocks is sufficient for the portfolio and the values of the clustering factors are the best.

It is known that the lower P/E ratio indicate more profitable shares. Also other factors like ROI, liquidity ratio, EPS, profit margin and especially beta value determines cluster selection priorities.

In order to simplify cluster selection, we have calculated the cluster averages of all used factors (Table 2).

As we can see from Table 2, most promising clusters are C1 and C2. They have the lowest P/E values, one of the highest beta, and sufficiently good other indicators. As C2 cluster has adequate shares number of 14 and the lowest P/E value, we have selected it for further study.

When we have selected the stocks for the portfolio, we need to find the optimal investment capital distribution along the portfolio shares based on the Mean−Variance portfolio minimization conditions described in Sect. 2.

To apply the GWO algorithm for setting a portfolio share weights, we need a return data of C2 cluster shares. The fourteen quarterly return data are extracted from Yahoo Finance (https://finance.yahoo.com/) website to calculate returns and risk (see Table 3).

We will write quarterly calculated expected return as 14 dimensional vector:

$R = (3.1, 3.5, 0.5, 2.0, 3.1, -2.7, -1.9, 1.9, 0.2, -0.6, 3.1,2.\ 5, 0.3, 10.1)$.

By using MS Excel we have calculated the covariance matrix of stocks returns (Table 4).

As we have a return and covariance date, we can apply GWO algorithm using formulas (1) and (2), estimate the portfolio shares weights and calculate the fitness

Table 2 Mean values of factors used in clusterisation

Cluster	P/E	Beta	Market Capitalization Billion USD	Annual Return, in %	ROA in %	ROI in %	P/B	Liquidity ratio	EPS
C1	14.58	1.302	40.4	9.56	7.04	12.09	3.06	1.455	6.68
C2	10.62	1.451	23.1	13.60	10.29	16.38	2.43	2.307	7.08
C3	16.60	1.449	22.6	12.96	13.79	28.74	6.11	1.880	6.24
C4	16.60	1.108	130.8	27.00	14.33	24.90	7.03	2.017	10.03
C5	19.53	1.220	16.0	30.10	19.10	25.20	6.23	6.550	3.89
C6	14.53	1.570	13.4	17.20	12.20	38.90	25.69	1.300	4.03
C7	19.27	1.090	78.8	29.00	19.70	25.50	13.39	1.800	97.56

Table 3 Quarterly Return data of C2 cluster shares

Date	COP	PCAR	SNA	CTSH	CMI	CBS	VIAB	LYB	NUE	ALB	PKG	BWA	TxT	MU
Feb-17	−0.1	10.1	2.3	10.8	7.0	13.7	16.5	3.5	0.0	23.9	8.4	19.6	1.2	36.1
May-17	−6.9	−7.6	−4.1	12.9	4.8	−9.9	−16.3	−12.4	−5.7	9.2	11.9	3.4	−0.6	22.1
Aug-17	0.6	4.7	−9.9	5.2	1.5	5.3	−18.6	14.0	−4.8	4.3	9.5	5.8	3.1	4.5
Nov-17	17.6	6.1	15.8	0.9	4.2	−11.0	0.3	16.5	3.9	11.7	4.0	18.3	11.5	28.8
Feb-18	5.0	−1.5	−8.7	14.2	−3.0	−6.3	20.9	3.9	19.2	−27.9	0.5	−12.5	5.1	17.0
May-18	26.8	−5.9	−0.8	−4.9	−9.0	−5.9	−21.2	6.5	−2.4	−0.5	2.9	4.4	16.6	19.6
Aug-18	8.0	9.1	17.8	2.3	−0.5	6.5	10.4	0.5	−4.4	1.7	−7.8	−12.1	2.8	−10.6
Nov-18	−9.5	−8.6	−5.4	−8.9	7.4	2.5	6.1	−16.5	-2.8	1.2	−10.4	-9.2	−18.6	−26.6
Feb-19	4.6	13.3	−3.8	1.5	3.0	−5.9	−3.9	−7.8	0.7	-5.4	−0.8	3.8	−3.0	7.8
May-19	−14.0	−2.4	−1.3	−13.9	−1.7	−4.6	−0.7	−11.7	−19.9	−30.0	−6.7	−12.9	−16.7	−21.6
Aug-19	−11.0	0.1	-4.0	−0.5	−0.1	−12.6	−13.4	5.6	2.8	−2.0	13.9	−7.6	−0.6	38.8
Nov-19	15.7	24.6	8.6	4.8	23.4	−3.6	−2.9	21.2	16.0	6.5	12.1	29.5	2.8	4.9
Average	3.1	3.5	0.5	2.0	3.1	−2.7	−1.9	1.9	0.2	−0.6	3.1	2.5	0.3	10.1

Table 4 Covariance matrix K (each value is multiplied by 100)

Clumn1	COP	PCAR	SNA	CTSH	CMI	CBS	VIAB	LYB	NUE	ALB	PKG	BWA	TxT	MU
COP	1.44	0.42	0.49	0.15	0.00	−0.06	−0.14	0.89	0.52	0.44	0.10	0.83	0.95	0.62
PCAR	0.42	0.88	0.39	0.19	0.46	0.17	0.25	0.64	0.38	0.42	0.20	0.81	0.20	0.20
SNA	0.49	0.39	0.75	−0.04	0.17	0.06	0.28	0.36	0.04	0.46	−0.09	0.39	0.26	0.01
CTSH	0.15	0.19	−0.04	0.66	0.12	0.07	0.24	0.28	0.46	0.39	0.36	0.35	0.36	0.96
CMI	0.00	0.46	0.17	0.12	0.57	0.09	0.14	0.24	0.27	0.48	0.18	0.67	−0.14	−0.11
CBS	−0.06	0.17	0.06	0.07	0.09	0.58	0.42	−0.02	−0.13	0.40	−0.15	0.10	−0.12	−0.42
VIAB	−0.14	0.25	0.28	0.24	0.14	0.42	1.73	−0.17	0.42	−0.29	−0.47	−0.25	−0.27	−0.41
LYB	0.89	0.64	0.36	0.28	0.24	−0.02	−0.17	1.33	0.94	0.10	0.27	0.46	0.42	0.80
NUE	0.52	0.38	0.04	0.46	0.27	−0.13	0.42	0.62	0.94	2.14	0.53	1.37	0.45	1.27
ALB	0.44	0.42	0.46	0.39	0.48	0.40	-0.29	0.52	0.10	2.14	0.64	0.64	0.36	1.26
PKG	0.10	0.20	−0.09	0.36	0.18	−0.15	−0.47	0.51	0.27	0.53	0.64	1.84	0.57	1.19
BWA	0.83	0.81	0.39	0.35	0.67	0.10	−0.25	0.98	0.46	1.37	0.64	1.84	0.91	1.29
TxT	0.95	0.20	0.26	0.36	−0.14	−0.12	−0.27	0.79	0.42	0.45	0.36	0.57	0.91	1.29
MU	0.62	0.20	0.01	0.96	−0.11	−0.42	−0.41	1.08	0.80	1.27	1.26	1.19	1.29	4.15

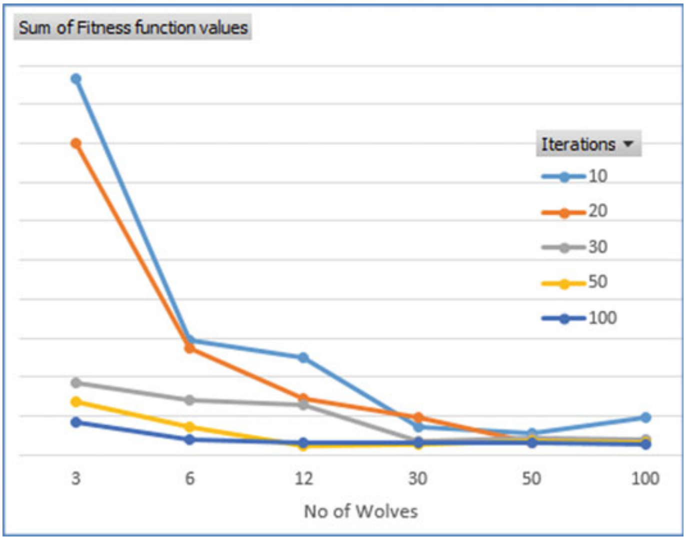


Fig. 5 Fitness function dependence from number of wolves and iterations

function according the formula (3). For this case we write a program code in MATLAB (see Annex 1). Using this program, we can change various program parameters and determine when the fitness function takes a best value.

We have performed the 45 experiments with different number of iterations, wolf herd size and max capital allocation for share.

Firstly, we investigate the sensitivity of fitness function to changes of wolves heard size and iterations. For this case we fixed the max amount of capital per share less than 30%. The results of research were presented in Fig. 5 (the lowest fitness value means better result).

The figure let us see the lowest fitness value after 50 iterations with 12 wolves herd. The wolves herd of 3 or 6 wolves is not enough quickly find the optimal value. Only after about 100 trials 6 wolves can achieve enough low fitness value.

The next question that arises is whether the weight limit for share of 30% is optimal. A different capital allocation percentages and its influence on fitness is presented in Fig. 6. For this case we fix 12 wolves herd.

The research results disclose that for 100 iterations max stock weight has no affect-the fitness function value is stable, although the optimal size of the fitness function is reached at max 40% capital allocation for shares and 50 iterations.

So, finally, we found the best values for our parameters: 12 wolves, 50 iterations and max 40% capital allocation for shares. By using these parameters with GWO algorithm by utilising Quarterly Return (years 2017–2019) data of portfolio shares we estimated the optimal capital allocation percentage for every share in portfolio (Table 5).

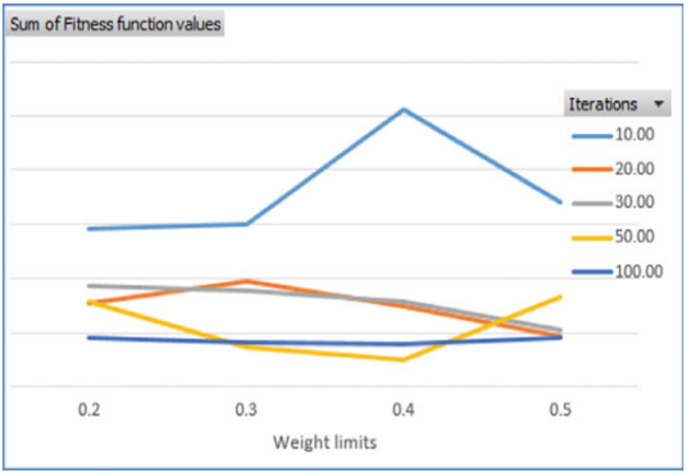


Fig. 6 Fitness function dependence from share weight limits and number of iterations

Table 5 Proportion of capital distributed across the portfolio shares

COP	PCAR	SNA	CTSH	CMI	CBS	VIAB	LYB	NUE	ALB	PKG	BWA	TxT	MU
18%	18%	3%	2%	18%	2%	1%	0%	0%	0%	18%	0%	0%	18%

As we can see, we have selected to invest in only 9 of 14 stocks, excluding LYB, NUE, ALB, BWA and TXT stocks. Five shares (COP, PCAR, CMI, PKG, MU) got the 18% of capital each, and a very small percentage of capital is offered to invest in SNA, CSTH, CBS, VIAB shares.

With this capital allocation, the expected return and variance are 4.103 and 0.533%, respectively, and the value of the fitness function is 0.1299. Table 6 presents the highest Return, lowest Risk and Fitness portfolios comparison to the characteristics of the naive portfolio (when capital is distributed equally to all equities).

So the investors can choose the portfolio according their preferences—to hope for max return, experience min risk or got the optimal balance for Risk-Return Ratio. Our study confirmed the assumption that it is inefficient to consider a naive

Table 6 Comparison of different portfolios

	Iterations	Wolves	Max weight	Return (%)	Variance (%)	Fitness
Best fitness	50	12	40%	4.103	0.533	0.1299
Max return	100	3	30%	4.620	0.640	0.1385
Min variance	10	12	30%	2.746	0.438	0.1595
Naïve portfolio	with equal capital allocation for shares			1.803	0.444	0.2463

investment portfolio. The application of the GWO algorithm let us increase the return and minimize the investment risk.

Another way to check the usefulness of proposed portfolio management method is to check whether the weights of capital from Table 5, let as hope for adequate profit in the case of a real investment.

Our research was done on S&P500 index companies based on historical quarterly data from February, 2017 to November, 2019. We have collected the adequate stocks and set the optimal weight for portfolio shares. Let's try to calculate our portfolio return for other time window-from September 22, 2019 till December 20th, 2019.

Using the simple calculations, we got that our portfolio achieves a return of 10.67%, while the direct investing in the S&P500 index over the same period would limited a 7.15% return. This demonstrates the advantages of our proposed portfolio selection and capital allocation method.

It is understandable that the stock portfolio we have formed can be safely maintained for some time. However, when changing the investment time window, we should double-check the allocation of invested capital to equities using the GWO algorithm. Timely updating of historical data is of great importance in achieving the optimal Risk-Return ratio of our investments. How often this should be done is the task of the next our study.

6 Main Results and Conclusion

When forming an investment portfolio, it is very important to evaluate its optimality using the Mean−Variance method. This allows us to achieve an optimal balance between the expected profit and the level of possible risk.

In this article, we present an innovative method for selecting stocks for a portfolio and optimizing it according to the Mean−Variance principle using the nature-inspired Gray Wolf Optimization (GWO) algorithm.

The stocks screening from S&P500 index companies are done by help of Self-Organizing Maps (SOM). This method allows us to construct share classes based on their similarity in relation to some economic and financial factors. For this case we advise to cluster shares by similarity of factors P/E, ROA, P/B, BETA, market capitalization, Earnings Per Share (EPS), ROI, liquidity ratio and profit margin. The cluster with the best P/E and Beta estimates is selected.

Once the stocks for the portfolio are identified, it remains to determine the optimal percentage of capital allocation to them. For this task, we need the GWO algorithm, which allows us to identify the weights of the shares, at which the optimal value of the fitting function is obtained. The fitting function is defined as the ratio of risk and return (see formula 3).

The performance of GWO algorithm depends on number of iterations, wolf-pack size and weight limits of stocks. The relevant estimation of these parameters have a high influence on investment outcomes.

By using a numeric example, we found the best values of parameters, calculated the stock weight distributions to obtain max return, effective risk level, and optimal risk-return portfolio. It was shown that a naive portfolio, when capital is divided equally among shares, does not meet the criteria of a profitable investment strategy.

The advantages of proposed method for portfolio formation was checked on real data (S&P500, September 22, 2019 to December 20, 2019). We have noticed a 3.52% higher profitability than in case of direct investment to the S&P500 index.

Appendix: GWO Algorithm in Matlab for the Set of Selected Equities

```
function o = F1(x)
% covariation function of 14 securities, just as example
B = [1.439 0.418 0.491 0.149 0.003–0.057–0.142 0.888 0.519 0.445 0.102 0.831 0.948 0.625;…
0.418 0.878 0.393 0.193 0.455 0.173 0.248 0.636 0.380 0.415 0.198 0.810 0.198 0.202;…
0.491 0.393 0.746 −0.045 0.171 0.061 0.281 0.360 0.044 0.462 −0.091 0.394 0.255 0.012;…
0.149 0.193 −0.045 0.658 0.124 0.066 0.237 0.283 0.460 0.388 0.360 0.355 0.357 0.960;…
0.003 0.455 0.171 0.124 0.569 0.085 0.143 0.236 0.274 0.477 0.185 0.671 −0.143 −0.112;…
−0.057 0.173 0.061 0.066 0.085 0.584 0.422 −0.019 −0.127 0.396 −0.148 0.102 −0.120–0.424;…
−0.142 0.248 0.281 0.237 0.143 0.422 1.730 −0.167 0.421 −0.286 −0.474–0.252–0.270–0.410;…
0.888 0.636 0.360 0.283 0.236 −0.019 −0.167 1.331 0.943 0.100 0.271 0.463 0.420 0.801;…
0.519 0.380 0.044 0.460 0.274 −0.127 0.421 0.624 0.943 2.138 0.530 1.371 0.451 1.268;…
0.445 0.415 0.462 0.388 0.477 0.396 −0.286 0.521 0.100 2.138 0.637 0.641 0.358 1.257;…
0.102 0.198 -0.091 0.360 0.185 −0.148 −0.474 0.513 0.271 0.530 0.637 1.835 0.565 1.195;…
0.831 0.810 0.394 0.355 0.671 0.102 −0.252 0.983 0.463 1.371 0.641 1.835 0.912 1.287;…
0.948 0.198 0.255 0.357 −0.143 −0.120 −0.270 0.791 0.420 0.451 0.358 0.565 0.912 1.287;…
0.625 0.202 0.012 0.960 −0.112 −0.424 −0.410 1.080 0.801 1.268 1.257 1.195 1.287 4.146];
% Mean
```

```
r = [3.1;3 .5; 0.5; 2.0; 3.1; −2.7; −1.9; 1.9; 0.2; −0.6; 3.1; 2.5; 0.3; 10.1];
% Loss function
o = (x*B*x')/(x*r);
end
%Main program
clc
clear
% -*- coding: utf-8 -*-
%"""
%Created on Sun March 15 12:47:20 2020
.
%@author: Virgilijus Sakalauskas
%"""
T = 10; %Number of Rounds
W = 12; %Number of wolfs
for i = 1:W %12 wolfs
R = unifrnd(0, 1, 1, 14); %initial random values
A(i, 1:14) = R./sum(R); % standardized vector of random values
A(i, 15) = F1(A(i, 1:14)); % calculated loss function values added to 15 column
end
.
A = sortrows(A, 15); % matrix sor4ted according the column 15
disp(' Weights for all 14 securities and Loss function values ');
A = sortrows(A, 3);
for v = 1:T
aa = (2–2*v/T);
for i = 1:W % for Da, Db and Dg calculation
r1 = unifrnd(0, 1, 1, 14);
r2 = unifrnd(0, 1, 1, 14);
Da(i, 1:14) = abs(2*r1.*A(1, 1:14)-A(i, 1:14));
X1(i, 1:14) = (A(1, 1:14)-aa.*(2.*r2-1).*Da(i, 1:14));
r1 = unifrnd(0, 1, 1, 14);
r2 = unifrnd(0, 1, 1, 14);
Db(i, 1:14) = abs(2*r1.*A(2, 1:14)-A(i, 1:14));
X2(i,1:14) = (A(2,1:14)-aa.*(2.*r2-1).*Db(i,1:14));
r1 = unifrnd(0, 1, 1, 14);
r2 = unifrnd(0, 1, 1, 14);
Dg(i, 1:14) = abs(2*r1.*A(3, 1:14)-A(i, 1:14));
X3(i, 1:14) = (A(3, 1:14)-aa.*(2.*r2-1).*Dg(i, 1:14));
Vid(i, 1:14) = abs((X1(i, 1:14) + X2(i, 1:14) + X3(I, 1:14))/3);
Vid(i, 1:14) = (Vid(i, 1:14)./sum(Vid(i, 1:14)));
Vid(i, 15) = F1(Vid(i, 1:14)); % calculating new value of loss function
end
B = sortrows(Vid,15); %sorting in ascending order of loss function
for k = 1:3
```

```
if A(k, 15) < B(k, 15)
B(k, 1:15) = A(k, 1:15);
end
end
A = B; fprintf('Round = '); disp(v);
disp([' Weight1 Weight2 Weight3 Weight4 Weight5 Weight6 Weight7' …
' Weight8 Weight9 Weight10 Weight11 Weight12 Weight13 Weight14 Loss func']);
disp(A);
end
```

References

1. Abraham, A., Elhariri, E., El-Bendary, N., Hassanien, A.E.: Grey wolf optimization for one-against-one multi-class support vector machines. In: 7th International Conference of Soft Computing and Pattern Recognition (SoCPaR), Fukuoka, Japan, pp. 7–12 (2015). https://doi.org/10.1109/SOCPAR.2015.7492781
3. Chandra, P.: Investment Analysis and Portfolio Management. McGraw-Hill Education (2017). ISBN 978 – 93 – 85965 – 57 – 9
4. Ertenlice, O., Ir Kalayci, C.B.: A survey of swarm intel-ligence for portfolio optimization: algorithms and ap-plications. Swarm Evol. Comput. **39**, 36–52 (2018)
5. Kamboj, V.K., Bath, S.K., Ir Dhillon, J.S.: Solution of non-convex economic load dispatch problem using Grey Wolf Optimizer. Neural Comput. Appl. **27**(5), 1301–1316 (2016)
7. Khan, A.U., Gour, B., Khan, A.U.: Portfolio formation and its performance calculation with the stocks selected by SOM using technical indicators. Int. J. Adv. Eng. Ir Technol. **8** (1) (2016)
8. Kohonen, T.: The self-organizing map. Proc. IEEE **78**(9), 1464–1480 (1990)
9. Konno, H., Yamazaki, H.: Mean-absolute deviation portfolio optimization model and its applications to Tokyo stock market. Manage. Sci. **37**(5), 519–531 (1991)
10. Kriksciuniene, D., Sakalauskas, V., Imbrazas, A.: Grey wolf optimization model for the best mean-variance based stock portfolio selection. In: Abraham, A., Sasaki, H., Rios, R., Gandhi, N., Singh, U., Ma, K. (eds.), Innovations in Bio-Inspired Computing and Applications. IBICA 2020. Advances in Intelligent Systems and Computing, vol. 1372. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-73603-3_11
11. Kriksciuniene, D., Pitner, T., Sakalauskas, V.: Tracking customer portrait by unsupervised classification techniques. In: Transformations in Business and Economics, vol. 11, no. 3, pp. 167–189. Vilniaus universiteto leidykla, Vilnius (2012). ISSN 1648–4460
12. Lacerda, E.B., Mello, C.A.B.: Segmentation of touching handwritten digits using self-organizing maps. In: 2011 IEEE 23rd International Conference on Tools with Artificial Intelligence, Boca Raton, FL, pp. 134–137 (2011)
13. Lim, A.E., Shanthikumar, J.G., Ir Vahn, G.Y.: Conditional value-at-risk in portfolio optimization: Coherent but fragile. Oper. Res. Lett. **39**(3), 163–171 (2011)
14. Mamanis, G.: Portfolio optimization with metaheuristics. Financ. Mark. **2**(2) (2017)
15. Markowitz, H.: Portfolio selection. J. Financ. **7**, 77–91 (1952)
16. Mech, L.D.: Possible use of foresight, understanding, and planning by wolves hunting muskoxen. Arctic 145–149 (2007)
17. Michaud, R.O., Michaud, R.O.: Efficient Asset Management: A Practical Guide to Stock Portfolio Optimization and Asset Allocation. Number 9780195331912 in OUP Catalogue. Oxford University Press (2008)

18. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey wolf optimizer. Adv. Eng. Softw. **69**, 46–61 (2014)
19. Muro, C., Escobedo, R., Spector, L., Coppinger, R.P.: Wolf-pack (Canis lupus) hunting strategies emerge from simple rules in computational simulations. Behav. Proc. **88**(3), 192–197 (2011)
20. Mustaffa, Z., Sulaiman, M.H.: Price predictive analysis mechanism utilizing grey wolf optimizer-Least Squares Support Vector Machines. J. Eng. Appl. Sci. **10**, 17486–17491 (2015)
21. Rather, A.M., Sastry, V.N., Agarwal, A.: Stock market prediction and Portfolio selection models: a survey. OPSEARCH**54**, 558–579 (2017). 10. 1007/s12597-016-0289-y
22. Ritter, H., Martinetz, T., Schulten, K.: Neural computation and self-organizing maps. An Introduction, Addison-Wesley, New York (1992)
6. Salehi, K.: An application of Grey Wolf Optimization algorithm for fuzzy portfolio selection problem. In: Proceedings of 12th International Conference on Industrial Engineering, 2016/1 (2016)
23. Silva, B., Ir Marques, N.C.: Feature clustering with self-organizing maps and an application to financial time-series for portfolio selection. In: IJCCI (ICFC-ICNC), pp. 301–309 (2010)
2. Souza Júnior, A.H., Barreto, G.A., Varela, A.T.: A speech recognition system for embedded applications using the SOM and TS-SOM networks. Self Organizing Maps—Applications and Novel Algorithm Design, Josphat Igadwa Mwasiagi, IntechOpen (2011). https://doi.org/10.5772/14401
24. Tang, G.Y.: How efficient is naive portfolio diversification? Educ. Note. Omega **32**(2), 155–160 (2004)
25. Van Nieuwerburgh, S., Veldkamp, L.: Information acquisition and under-diversification. Rev. Econ. Stud. **77**(2), 779–805 (2010)
26. Young, M.R.: A minimax portfolio selection rule with linear programming solution. Manage. Sci. **44**, 673–683 (1998)
27. Zainal, N.A., Ir Mustaffa, Z.: Developing a gold price predictive analysis using Grey Wolf Optimizer. In: 2016 IEEE Student Conference on Research and Development (SCOReD), pp. 1–6. IEEE (2016)

Towards Seamless Execution of Deep Learning Application on Heterogeneous HPC Systems

Li Zhong, Oleksandr Shcherbakov, Dennis Hoppe, Michael Resch, and Bastian Koller

Abstract Deep learning has been already successfully applied in many areas of science and industry. Since we are dealing often with extremely large data or very complex neural network architectures, parallelization of deep learning algorithms and frameworks is becoming more and more important. These solutions can no longer be processed on commodity hardware with the high requirement of data security; this is where HPC comes in. When going from classical artificial intelligence (AI) to high-performance AI, we need to ensure that HPC is ready for this endeavour. Thus, today's HPC centers need to provide seamless workflows to enable analytics and deep learning solutions, so that data scientists can fully exploit the performance of HPC systems. In this paper, we demonstrate methodologies for applying deep learning on HPC, and how AI techniques can successfully be integrated with classical simulation codes (e.g. to achieve better accuracy). Furthermore, we present an overview about training neural networks on HPC while successfully leveraging data, model, pipeline and hybrid parallelism. Finally, we adopt these techniques for two use cases: (i) novel hybrid workflow to combine a multi-task neural network with a typical FEM simulation to determine material characteristics, and (ii) segmentation of high-resolution satellite images to identify rice paddies without manual labelling.

Keywords Deep learning · HPC · Hybrid workflow · Material characteristics · Image segmentation

---

L. Zhong (✉) · O. Shcherbakov · D. Hoppe · M. Resch · B. Koller
HLRS, University of Stuttgart, Stuttgart, Germany
e-mail: li.zhong@hlrs.de

O. Shcherbakov
e-mail: oleksandr.shcherbakov@hlrs.de

D. Hoppe
e-mail: dennis.hoppe@hlrs.de

M. Resch
e-mail: michael.resch@hlrs.de

B. Koller
e-mail: bastian.koller@hlrs.de

G. Dzemyda et al. (eds.), *Data Science in Applications*,
Studies in Computational Intelligence 1084,
https://doi.org/10.1007/978-3-031-24453-7_11

1 Introduction

High-performance computing (HPC) has long been crucial to running the large-scale simulation and analytic workloads that foster scientific progress, product innovation, and companies' competitiveness. With the growing numbers of artificial intelligence (AI), high-performance data analytics (HPDA), and modelling/simulation workflows, there is a need to leverage high-performance computing infrastructure to address the increasing need for computing power of these workflows. For example, the required computing power for the largest AI training runs has increased exponentially, with a doubling time of around 3.5-months, as reported by OpenAI [8]. These emerging needs are expanding the scope of HPC and making HPC infrastructures more necessary than ever to tackle the eruption of Artificial Intelligence.

Two main drivers are responsible for expanding the reach of HPC for AI. On the one hand, the amount of digital data output that is expected to exceed 163 zettabytes by 2025 [9] and the need to analyze them for meaningful information will increase the need for high-performance infrastructure. On the other hand, the other important factor is the increase in computing power at affordable costs for HPC centers. HPC resources power the integration of AI and HPDA combined with simulation to deliver the computational performance and throughput to scale resource-intensive AI workloads, process real-time data streams, and train complex deep learning models. Nonetheless, it requires an additional research and practical experience to exploit its maximum potential through the convergence of AI with HPC.

1.1 Related Work

A lot of research has been carried out to drive the convergence of HPC and Deep learning. Mozaffari et al. [36] developed an HPC-oriented canonical workflow for climate and weather prediction using machine learning. Lee et al. [37] examined how coupling DL approaches with MD simulations can lead to effective approaches to fold small proteins on supercomputers, it is demonstrated that demonstrate that the DL-coupled MD workflow on HPC is able to effectively learn latent representations and drive adaptive simulations. Archibald et al. [38] present some of the current challenges in designing deep learning and integrating it with traditional high-performance computing (HPC) simulations.

In this work, we advance the convergence of HPC and deep learning towards effective parallel execution of deep learning codes on massively parallel machines having hundreds of graphical processing units (GPUs) and thousands of CPUs. We study the execution of our parallel deep learning code on the our HPC system "Hawk" and on a Cray CS-Storm, which are both highly-qualified to run AI workloads while utilizing accelerators. In order to achieve the best performance on HPC systems, we introduce parallelization methodologies that can be applied to deep neural networks (DNN) training. This includes data, model and pipeline parallelism. Furthermore, we will discuss communication techniques that are used to optimize the performance on Infiniband. In addition, because I/O is often the performance bottleneck with large training datasets, we will describe how an efficient input pipeline can be designed. The performance of our systems is demonstrated by conducting two use case studies, which are performed by designing and training DNN through TensorFlow in a distributed manner.

1.2 Scope

This paper is organized as follows. Section 2 provides the required background information to better understand the requirements of deep learning workloads on HPC architectures, and the current technology gap that needs to be tackled in the next few years. Section 3 then introduces best practices of deep learning to leverage the performance of HPC systems including an overview of suitable frameworks for parallelization of AI workloads, and an introduction to hybrid HPC/AI workflows. Section 4 then continues to apply the introduced methodologies and best practices to two case studies from engineering and food safety. Finally, Sect. 5 present the conclusion with a brief outlook towards further merging AI and HPC workloads in the near future.

2 Background

In this section, we present the background of Deep Learning, HPC and the gaps between them, which are the key concepts to understand the remainder of this work.

2.1 Deep Learning

In the past years, deep learning (DL) has shown its success in various fields, such as natural language processing (NLP) [2], computer vision (CV) [3], robotics, etc. DL, now often regarded as a new research area, is a subset of machine learning (ML) algorithms that uses neural networks with multiple layers, namely DNNs, to achieve the goal of ML. A DNN is composed of neurons and links among neurons, where each neuron in the network learns a simple function and the overall function is created by combining the all these simple functions. Therefore, DL is particularly suited for contexts where the correlations among features are complex and where there are large dataset available. The same as machine learning, deep learning can also be divided into several major categories, i.e., supervised learning, unsupervised learning, reinforcement learning, et cetera.

2.2 HPC Architectures

HPC is a well-established domain, in which everything from hardware to software is optimized for performance. The hardware is often composed of server-based components and interconnects that provide high-throughput and low-latency. For example, the Hawk system at HLRS [21], which is composed of 5,632 CPU nodes and 24 GPU nodes (Apollo 6500 nodes with 8 NVIDIA A100 GPUs per node), deploys an Infiniband HDR based interconnect with a 9-dimensional enhanced hypercube topology. Infiniband HDR has a bandwidth of 200 Gbit/s and a MPI latency of 1.3us per link. Hawk has a peak performance of 26 Petaflops and its GPU accelerator extension has a peak performance of 120 Petaflops for DL training. In addition, Cray CS-Storm [22] (part of Vulcan cluster) is also provided to users. It is composed of 8 NVIDIA Tesla V100 GPU nodes for DL workloads and 8 Intel Xeon Gold 6230 CPU nodes (CS-500) for big data workloads. To address the demand for processing-intensive applications in the realms of ML and DL, the Vulcan and Hawk-AI partitions support a wide variety of well-known and established AI frameworks and tools, such as Apache Spark, Python-based data science libraries like scitkit-learn, and frameworks steered toward DL like TensorFlow and PyTorch. The detailed configuration of Hawk and Vulcan are listed in Table 1.

The storage of HPC systems is available globally through distributed parallel file systems like Lustre or Network File System (NFS). The Lustre at HLRS, which provides about 25 PB of storage to its users, is accelerated with DDN IME to achieve highest performance especially when dealing with large amount of small files. And the ability of providing high performance solution for small files is utmost important for DL applications.

The detailed environment setup and framework deployment for DL on Hawk and Vulcan can be found in Sect. 3.

Table 1 Technical specification of Hawk and Vulcan

System	HPE Apollo (Hawk)	Apollo 6500	Cray CS-Storm
Number of compute nodes	5632	24	8
Peak performance	26 Pflops	120 Pflops (AI)	8 Pflops (AI)
CPU type	AMD EPYC 7742	AMD EPYC 7702	Intel CLX 6240
GPU type	–	Nvidia A100	Nvidia Tesla V100
Number of cores	720,896 (CPU)	1,327,104 (CUDA)	327,680 (CUDA)
CPU frequency	2.25 GHz	2.0 GHz	2.6 GHz
Interconnect	InfiniBand HDR200	InfiniBand HDR200	InfiniBand HDR100

2.3 Gaps Between HPC and Deep Learning

Developing for HPC requires a deep understanding and good knowledge about the underlying architectures, network topologies, programming environments and libraries [4]. For instance, MPI-based applications require knowledge in communication concepts (e.g. point-to-point, one sided, collective, blocking vs non-blocking, etc). declarative concepts (groups and topologies), and process management. Such optimization and parallelization is often a big challenge to DL experts, who are often not HPC experts. Therefore, to scale DL model training on HPC is often a non-trivial task, due to the fact that there exist several gaps between the two workflows. For example, DL frameworks are often updated at a much faster pace than libraries deployed cluster-wide on HPC platforms. Furthermore, producing and training DL models usually requires a unique and dynamic set of package dependencies. Therefore, the traditional use of modules on HPC has limited effectiveness since software dependencies change between projects and sometimes evolve even during a single project [15]. A detailed description of the difference between HPC and DL is shown in Table 2.

Table 2 Comparison of HPC and DL stack

Type	HPC	Deep learning
Frameworks	OpenMP, MPI, OpenFoam, ..	Tensorflow, Pytorch, ...
Programming language	C/C++, Fortran	Python, C++, Java
Network	Infiniband	Ethernet
Storage	Storage & I/O nodes, NAS	Local storage, NAS/SAN
Processors	CPUs, GPUs, FPGAs	CPUs, GPUs, TPUs, FPGAs

2.4 Frameworks

The software stack in HPC is in general limited, but very optimized to perform embarrassingly-parallel tasks. The operating environment is often based on customized Linux derivatives (e.g. CentOS), and for development, languages such as FORTRAN, C, and C++ are supported. In order to allow for parallel programming, modules such as MPI and SHMEM are available. Accessing the system is done solely via the command line through remote access.

However, most DL frameworks are not built with HPC in mind, which are usually not able to leverage the full power of HPC. Over the years, numerous DL frameworks have been developed to support training and deployment of DL applications. Almost all major DL frameworks provide some support for distributed training of DNNs. Among them, Tensorflow and Pytorch are the two most popular frameworks.

TensorFlow [16], developed by Google and community contributors, is one of the most widely used ML/DL frameworks. It natively supports distributed and parallel training through both model parallelism and data parallelism. And it shows high scalability of computation across machines on huge data sets through its highly optimized data pipeline. PyTorch[17], developed by Facebook and community contributors, is another popular DL framework. In PyTorch, distributed, data-parallel training as well as model-parallel training are supported out-of-the-box.

Although the frameworks mentioned above offer native support for distributed computation, most of them are not suited for HPC architectures. For example, Mathuriya et al. [11] proved that native support of distributed training from TensorFlow was accompanied by a decrease in the efficiency on the worker when it is scaled up to 128 nodes. To overcome this obstacle and achieve better performance, several frameworks that can be regarded as distributed training middleware sitting between the DL framework and the communication runtime have been developed. Horovod [10], developed at Uber, is seamlessly integrated into TensorFlow and PyTorch programming. It uses MPI as a mechanism for communication to allow multi-node training, enabling benefits from optimizations made in the underlying MPI library, such as *allgather* and *allreduce* during handling of cross-replicas communication and weight updates. This is different from the TensorFlow hierarchical architecture in which a centralized parameter server is utilized to pass parameter updates.

2.5 Distributed Training

The success of DL models replies on creating highly complex interrelationships between the raw data input and the output data, which often requires several million or even billion parameters that have to be dynamically changed during the training process. Because of the large number of parameters in DL models, training a single network could take several days or even months on single GPU or single machine. For example, the GPT3 [1] model which is used for NLP has 175 billion weight parameters, and requires roughly 350 GB memory and 355 years on single Nvidia V100 GPU and 90 years on single AMD A100 GPU for training. Therefore, making these algorithms highly scalable by leveraging the high computational power and massive parallelism offered by HPC systems is highly desirable.

To efficiently scale DL training on distributed devices, three predominant parallelization methods are developed, namely data parallelism [33], model parallelism [32] and pipeline parallelism [34].

- Data parallelism: When the training dataset is too large to fit into the memory and storage of single device or machine, the whole dataset is split into non-overlapping chunks. Meanwhile, an identical copy of the DL model is created and assigned to each device in the cluster and each model replica is trained on the data chunks assigned to this device. The model parameters from different devices are synchronized after each step.

- Model parallelism: When the DL model is very huge and has too many parameters to fit into device memory, the whole model is partitioned and each device in the cluster only loads a part of the model. Each subset of parameters on each worker is updated in parallel, and the updates from each subset are highly compatible and dependent in order to ensure correctness
- Pipeline parallelism: Combines both data parallelism and model parallelism, where not only the model is split into different devices, but also the dataset is fractured into chunks. In particular, when training of DL model, data are processed through the network in parallel (data parallelism) and the length of the pipeline is determined by the DNN structure (model parallelism) [31].

3 Adopting Deep Learning for HPC

In this section, we introduce best practices of deep learning to leverage the performance of HPC systems including an overview about suitable frameworks for parallelization of AI workloads and the environment setups. We also describe two main usage scenarios of DL on HPC, i.e., scaling DL applications on HPC and hybrid HPC/DL workflows.

3.1 Environment Setup

A variety of frameworks, libraries and their versions make the setup and maintenance of software environments in multi-tenant environments–like HPC Systems–quite challenging. There are different possible approaches to address these issues. HPC Systems and HPC simulations are very performance oriented and hardware optimized, thus software and libraries must be compiled for the architecture of the system. On the other hand side most of AI computations are done on GPUs and using builds for a generic CPU can also be reasonable.

Cray Urika-CS software illustrates a mixture of both approaches: software is shipped in a Singularity container, some of the libraries are platform-optimized. Other containerized solutions or the ‘bring-your-own-container’ approach will also in most cases consist of a mixture of platform-dependent and generic libraries. The libraries using GPUs or other specific hardware are platform- or hardware-dependent, as they must be built for corresponding version of the drivers.

Another implementations of environments for AI workflows are pip and Anaconda environments, the latter of which is used for most simulation runs. Anaconda packages are compiled for a generic CPU, but the GPU versions of the libraries have builds for multiple CUDA version (for NVIDIA GPUs) and missing packages can also be installed from the pip repositories. As pip packages can be built on the system during their installation, they are even more platform optimized.

3.2 Hybrid Workflow

While DL has demonstrated strong abilities at extracting high-level representations of complex processes, the lack of sufficient ground truth data is often a critical issue faced in various areas. In fact, it is almost impossible to generate enough data in real life for supervised learning in many real-world problems, which are limited by scientific instrument, physical phenomenon itself, or the complexity of modeling. Nowadays, different methods have been developed to solve this problem, e.g. transfer learning, data augmentation, usage of synthetic data, generation of new data through Generative Neural Networks (GANs) [35], etc. Recently, scientists and engineers have begun experimenting with a relatively new approach to understand complex systems using Deep Neural Networks (DNNs), trained by the virtually unlimited data sets produced by simulations [5]. Studies have proven that these "synthesis models," combining DL and traditional simulation, can improve accuracy, accelerate time to solution and significantly reduce costs [6].

Apart from the benefits that DL applications can gain from the simulations, simulations as typical HPC tasks can also benefit from the DL processes. For example, to perform simulations, input parameters have to be determined and validated by a large number of tests to produce accurate simulation results [7]. Furthermore, the evaluation and validation of such input parameters for the simulation often requires a deep understanding of domain specific knowledge, software and programming skills. Thus, how to efficiently define and validate the input parameters for the simulation becomes the key factor in the development and design of numerical models. While

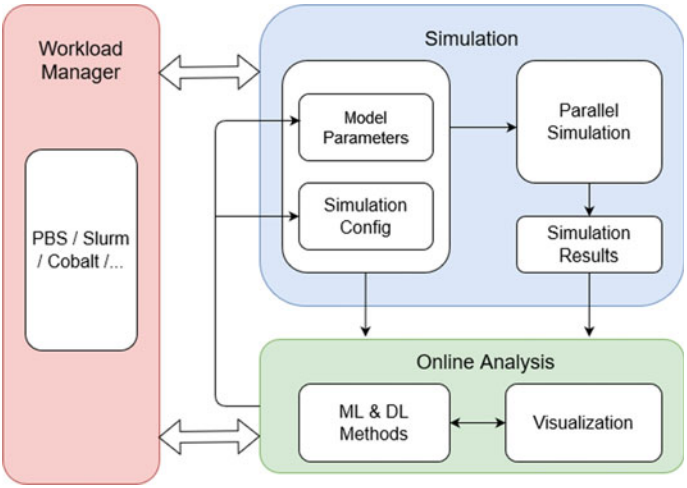


Fig. 1 General workflow which integrates workload management, simulation, and ML/DL analysis

simulation can solve the data sparsity problem for DNN models, DL based methods can in turn solve the difficulty in determining and validating the input parameters for simulation by training DNN models. Both training of DNN model and the running of simulations are compute-intensive tasks, in which the supercomputers can manifest their computation efficiency.

Therefore, hybrid workflows combing simulation and DL applications/training are becoming more and more attractive in both research and industry. Fig. 1 demonstrates a typical workflow on HPC, whose target is to provide a flexible, easy to use method for interacting at runtime between the DL and simulation. So that the online DL analysis is utilized to improve the result of simulation while simulation output is continuously fed to the DL model as training dataset and the progression of this integration can be visualized.

4 Case Study

In this section, we will demonstrate the integration of DL applications on HPC with two distinctive use cases: material characteristic identification and unsupervised image segmentation for remote sensing images. The use case 1 detailed in Sect. 4.1 exhibits a hybrid workflow of simulation and DL training (implemented with Tensorflow) on HPC while the second use case described in Sect. 4.2 showed how a compute intensive DL training (implemented with Pytorch) is scaled on HPC, so that the two use scenarios in Sect. 3 are covered.

4.1 Case Study I: Material Characteristic Identification

In industry, it is significant that the production of material (e.g. sheet metal components) must be done in relatively high quantities and via cost-efficient process development, due to the low margins and high equipment costs. Thus, manufacturing companies are required to do a comprehensive process design and determine the most accurate correct input parameters. Therefore, they are always under constant time, cost and quality pressure. . Additionally, particular attention must be paid to avoiding surface defects during the sheet metal forming process. For this reason, current research activities focus on predicting such surface defects as precisely as possible in the early development stages of sheet metal components by using FEM simulation.

Therefore, defining and validating the material parameters for the selected FE model becomes the key factor in development and design of robust forming processes. However, the material parameters have to be determined and validated by a large

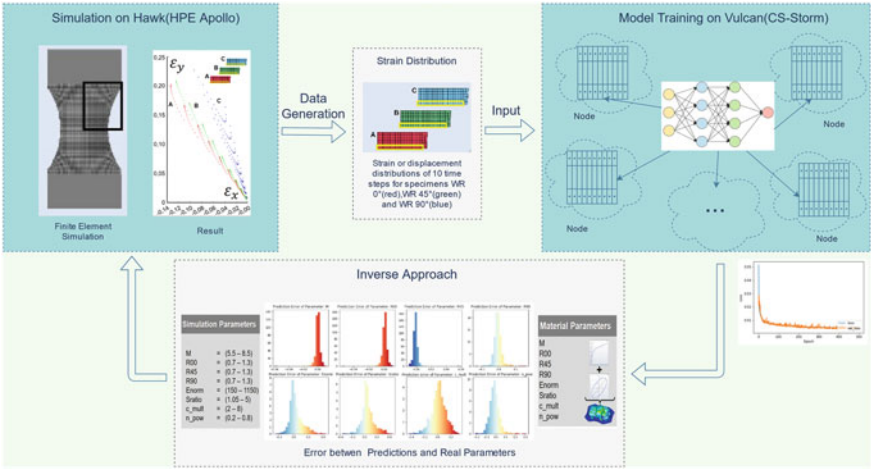


Fig. 2 Hybrid workflow of simulation and distributed deep learning for material characteristic identification on HPC

number of tests to expect accurate simulation results [19], which not only requires deep knowledge and expertise of the theory of plasticity, material characterization, cost-intensive simulation software and programming skills, but also huge time and financial cost.

Due to the fast development and success of ML especially DL in various areas, DL aided modeling is becoming more and more popular in identification of material's characteristics. However, the lack of sufficient ground truth data is often a critical issue, due to the fact that identification of material characteristics requires huge cost of time and manual power getting observational data from real life.

Therefore, a method which combines the DL methods and the FE simulation was proposed. As is depicted in Fig. 2, the proposed workflow consists of two phases: a data generation phase and a training phase. In the data generation phase, a set of material characterization tests were simulated using a variety of material parameters and the deformation field of the samples was recorded. Material parameters are used as simulation inputs to calculate the strain distributions, which are outputs of the simulation. In the second phase, a DNN model is trained with the strain distributions as inputs and the material parameters as outputs. Therefore, the FEM simulation can address the problem of data sparsity and the introduction of ML methods can reduce the high demand of expertise. The FEM simulation was carried out on the HPC system Hawk (HPE Apollo), and the DNN model was trained in a distributed manner on the GPU HPC system Vulcan (CS-Storm), which is one of HLRS' machines to accelerate artificial intelligence (AI) workloads. The two HPC systems use a common file system via a workspace mechanism [20], so that data can be freely exchanged between the two phases of the workflow.

4.1.1 Dataset and Pre-processing

In the data generation phase, the FEM simulation was carried based on the Barlat-3 parameter model [18], where the input parameters are optimized to make sure that generated dataset is sufficient while retaining a manageable computation effort. The whole dataset is roughly 3TB, composed of 4,941,258 records, where each record denotes a vector of 1080 strain values obtained per FEM simulation. Each vector can be divided into two halves: the first half represents the values of x-strains and the second half represents the values of y-strains. And each half can be divided into three parts, where each part denotes a specimen which is composed of 18 elements. Since for each finite element simulation, the longitudinal and transverse strains were exported for 10-time steps, 10 consecutive strain values of each element are recorded at 10-time steps, where the elements at the edges were excluded. The train and test dataset are split from with a ratio (0.9, 0.1).

Since 3TB is too big to fit into the memory of single GPU or node, a data parallelism strategy is adopted, where the whole dataset is split into different batches and assigned to different GPUs. In addition, an efficient data pipeline is implemented by overlapping the input pipeline with the computation pipeline to achieve the best performance on HPC systems. Input files are read and pre-processed individually with individual outputs as an embarrassingly parallel process. In addition, data are 'prefetched' to ensure that there is always a specified number of batches ready for the consumption. Moreover, we cached the data in memory to save some of the operations like file opening and reading data from being executed during training. By applying the cache method, the transformations before the cached one are executed only during the first epoch, the following epochs will reuse the cached data.

4.1.2 Methodology

In the training phase, a multi-task neural network is implemented. The overall structure of the model is depicted in Fig. 3. The whole network is composed of two main parts: the shared network and individual network for each parameter output. In the shared network, 1D CNN layer and max-pooling layers are used to extract the global features. When designing the parts of the network for each output, we found that for the outputs (MP1, MP2 and MP4), one CNN layer and one max-pooling layer followed by two fully connected layers would produce the best prediction performance, while for individual network of the outputs (MP3, MP5, MP6, MP7 and MP8), a more complicated network which has more CNN and max-pooling layers should be designed to better learn the features. In addition, Batch Normalization is employed here to stabilize the learning process and Dropout layers are used to avoid overfitting.

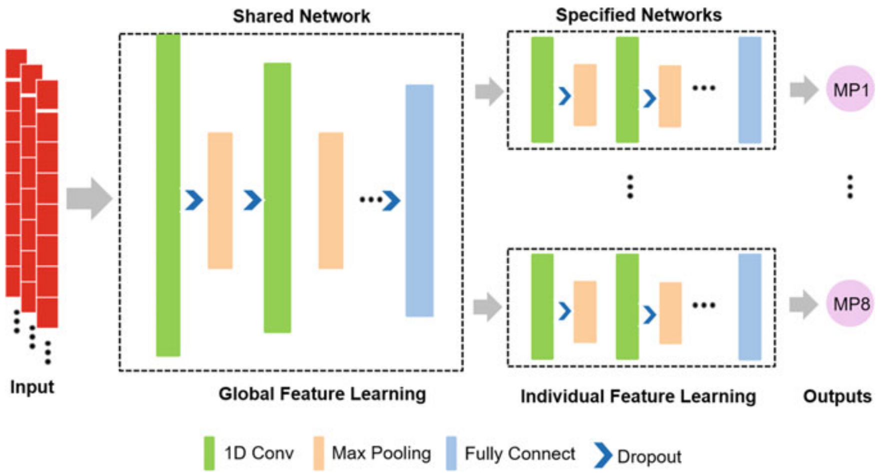


Fig. 3 Architecture of the multi-task learning model

Meanwhile, as stated above, the data parallelism strategy is adopted to accelerate the training. The whole dataset is split into multiple batches and assigned to different GPUs. Furthermore, the model is replicated and allocated to each GPU and each variable in the model is mirrored across all the replicas. All variables of the trained model are synchronized through identical updates. Furthermore, the efficient all-reduce algorithms implemented in NVIDIA Collective Communication Library (NCCL) are used to do the communication across all GPUs which can reduce the overhead of the synchronization significantly.

4.1.3 Performance Optimization

In order to make full use of the great computation power provided by supercomputers, different optimization methods are adopted to improve the training performance, e.g. learning rate schedule, data I/O optimization, etc.

The comparison of execution time between the training before optimization and after optimization is shown in Fig. 4.

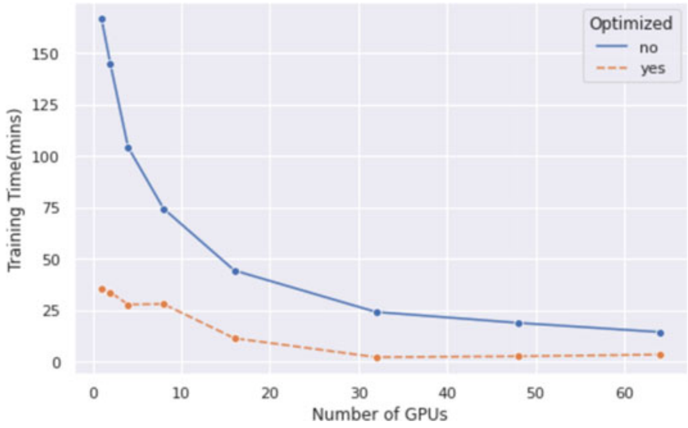


Fig. 4 Scaling DNN on multiple nodes, Training time versus number of GPUs

4.1.4 Result and Analysis

The performance of the proposed model is evaluated by inspection of the changes during the training and testing processes. The loss function is the weighted average of errors of all the outputs, which can be described as:

$$L = \sum_{1}^{N} \sum_{i=1}^{I} \frac{1}{I} (y_i - \hat{y}_i)^2 \tag{1}$$

where N denotes the number of outputs to be predicted. At the end of the training of 500 epochs, the the total training loss is around 0.0343 and the total validation loss is around 0.0386.

Finally, we measured the model performance by comparing the differences between the real parameters and the predicted parameters. The error histogram that depicts the normalized maximum error of the values is shown in Fig. 5, where 90% of the errors are below 0.1 and the normalized maximum error of the values (R45, Enorm, Sratio, c_mult, n_pow) are about 0.2. For parameters (M, R00, R90), it is noted that the maximum error is only 0.03 and 90% of the errors are below 0.02. Therefore, we can conclude that the experiments conducted successfully prove that parameters which reflect the material characteristics can be predicted by the proposed DNN model with a very high accuracy.

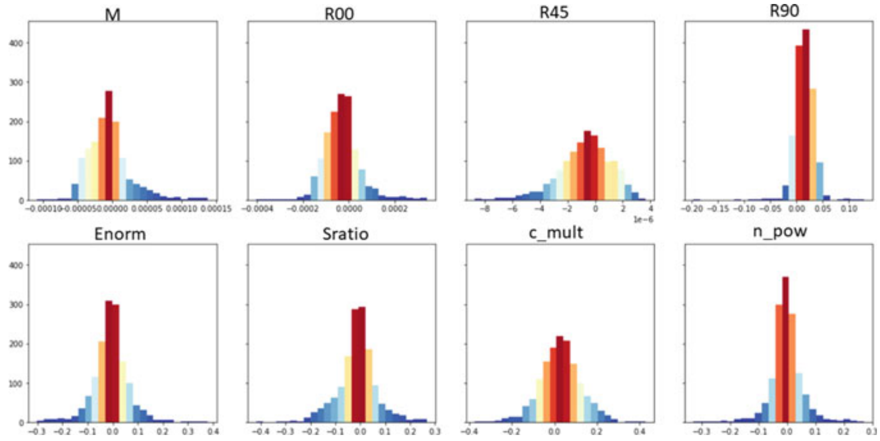


Fig. 5 Error histogram of the result on the test dataset

4.2 *Case Study II: Satellite Image Segmentation*

Over the next few decades, the continuous increase in global population and need for nutrition, in combination with the impact of climate change on food production, is expected to affect the food sector significantly [28]. Therefore, the agriculture productivity is required to be timely and accurately monitored at a large scale in order to provide the necessary knowledge for evidence-based decision making on food security management at a national and continental level [29]. In South Korea, rice is widely planted and the most important food staple, and there is pressing need for timely and accurate knowledge of rice's spatial distribution and its expected yield at national scales. This, in turn, requires continuous area monitoring and large-scale mapping, at the parcel level, through the processing of big satellite data of high spatial resolution.

However, the current monitoring and estimation strongly rely on costly and time-consuming traditional methods, i.e., field visits and the collection of field data at sample points, which is then spatially interpolated through statistical techniques in order to extract the required nationwide rice production assessments. Due to the great success of DL in the past years, more and more research has been performed on applying DL models to automatic or semi-automatic processing of high-resolution remote sensing images. Among these solutions, DL based image segmentation technology has been widely employed to extract the important part of an image and to identify the rice paddies. However, such DL based methods are supervised classifiers, which require a significant amount of ground truth data, especially labeled data, to train the prediction models. The labelled data are mostly manually collected and annotated, which in most cases is scarce and of poor quality. The issue of getting access to reliable ground truth data becomes even more challenging when dealing with large scale applications that cover vast areas at a national or continental level [30]. To overcome such challenges, unsupervised image segmentation methods in combination with data augmentation techniques are attracting more and more attention. In this regard, a large scale fully unsupervised image segmentation pipeline on HPC for high spatial resolution rice paddy classification that is independent of the hard-to-attain ground truth information is implemented.

4.2.1 Dataset and Pre-processing

Sentinel-2 [23], which freely and systematically supplies images of high spatial and temporal resolution at a global scale, is ideal for the monitoring of agriculture. The coverage of large areas, the short revisit times and the high spatial resolution of SAR and optical imagery has made Sentinel-1 and Sentinel-2 missions the main sources of EO data for numerous studies that address the monitoring of food security and the control of agricultural policies [24]. In this study, a time-series of Sentinel-2 MSI scenes were acquired through the Umbrella Sentinel Access Point for the period June to October 2018. The images obtained from Sentinel-2 consist of 13 spectral bands at 10, 20 and 60 m of spatial resolution. The study area mainly focuses on the region around the cities of Seosan and Dangjin that are located at the northwestern end of the South Chungcheong province in South Korea, which are located in separate climatic and agro-climatic zones and are described as having diverse rice paddy cultivation characteristics.

However, the remote sensing images provided by Sentinel-2 often suffer the noise introduced by different weather conditions, especially clouds. Therefore, actions have been taken to remove the noise to recover the cloudless feature information from satellite images contaminated by clouds. In addition, atmospheric correction processing was performed to eliminate the errors caused by atmospheric scattering, absorption, and reflection. Furthermore, data augmentation methods, e.g. random image rotation, image mirroring, image blurring, adding noise, etc. are utilized to increase the number of available training samples.

Although this study focuses on unsupervised image segmentation and rice paddy identification, a small amount of labeled dataset has been both acquired and generated to serve as the ground truth data, which can be used to do comparative study of the correctness of the designed unsupervised method.

4.2.2 Methodology

W-net proposed by Xia et al. [25] is implemented to perform unsupervised segmentation of the remote sensing image. The architecture of W-net is illustrated in Fig. 6, which adopts the encoder-decoder structure. The encoder part maps the input to a compact feature representation, and then the decoder part reproduces the input from its lower-dimensional representation. Both the encoder and the decoder part are composed of typical U-shaped architecture of a U-Net [26] network, and further form

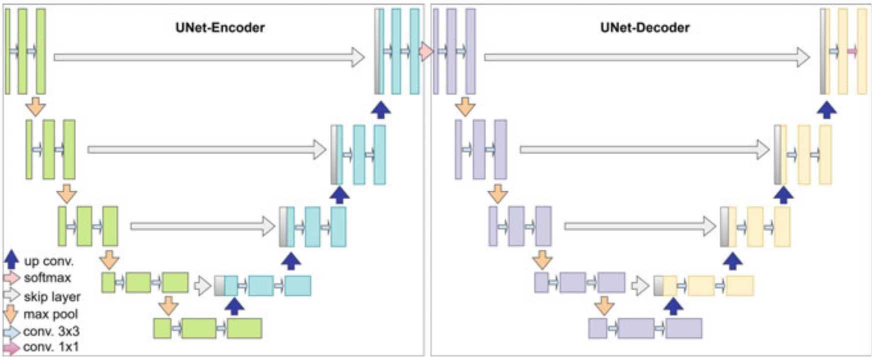


Fig. 6 W-net network structure

a W-shaped architecture such that it reconstructs original input images as well as predicts the segmentation maps without any labeling information.

In the encoder part of the network, at first a contracting path is built to capture context and a corresponding expansive path is built to enable precise localization. The contracting path starts with an initial module which performs convolution on input images and consists of several repeating network modules. Each repeating network module has two convolution layers followed by a rectified linear unit and max pooling operation for downsampling. The number of feature channels are doubled in each downsampling operation. In the expansive path, the opposite of contractive path, each repeating module uses a deconvolution that halves the number of feature channels and then stitches the result of the deconvolution with the corresponding feature map in the contracting path. The final convolutional layer of the encoder is a 1×1 convolution followed by a softmax layer, so that the feature vector can be mapped to the desired number of classes and the softmax layer rescales them to make sure the elements of outputs can be summed to 1. The architecture of the decoder part is similar to the encoder part.

In summary, each image fed into W-net is first downsampled by convolution and pooling, then upsampled by deconvolution, and then downsampled by convolution and pooling, and finally upsampled by deconvolution. The segmentation method based on W-net extracts the feature map obtained by convolution and the feature map obtained by deconvolution when the feature map is processed and connected by jump, so that the ability to capture image edge information can be improved [27].

4.2.3 Results and Analysis

The result of segmentation for high-resolution remote sensing images obtained from Sentinel-2 based on W-net is visualized in Fig. 7, where different segments (which represent different landscape types) are annotated with different colors. Clouds which are not completely removed in the data pre-processing which can be seen in region B

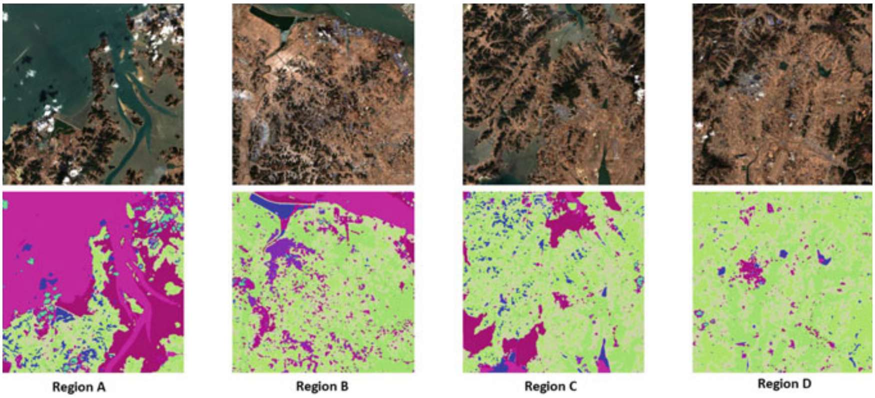


Fig. 7 Results of the W-net method tested on Seosan/Dangjin area, where different segments are shown in different colors

Table 3 Matrics of W-net's performance on the remote sensing images of four sub-regions in Seosan/Dangjin area

	Region A	Region B	Region C	Region D
Precision	93.02%	86.76%	90.95%	90.77%
Recall	89.36%	83.01%	87.51%	86.19%

are mis-segmented and thus downgrade the performance, which is also reflected in Table 3. It can also be shown that, for areas which has contra-distinctive landscapes, like region A has sea, snow, mountain, et cetera, W-net demonstrates the best performance, while for areas that are landscapes which are not in sharp contrast like region D, the result is not so promising. Further, due to the fact that a rice paddy is a distinctive crop type, mainly due to its inundation period, it constitutes a particularly easy target to discriminate. Therefore, W-net method did an excellent job identifying the rice paddies.

It is worthwhile to note that such fully unsupervised segmentation and identification of elements of high-resolution remote sensing images has great advantages in very large areas, compared to the traditional method which generates ground truth data from field visits, which are limited in number, fragmented in space and thus potentially less representative. Therefore, the method can be extended from a single city region to the whole country, where HPC can play an even more vital role.

5 Conclusion

Today's HPC centers are in need to lower the hurdle for users to fully leverage the potential of AI on HPC resources by providing methods and tools to seamlessly execute DL workloads. The demand, especially from the AI domain, is growing steadily due to the ongoing increase in training data and architecture complexity, so that HPC is the obvious choice when results are expected in a timely manner. In this report, we therefore have presented methodologies and best practices to execute machine learning and deep learning workloads on HPC (cf. Sect. 3). This requires expert knowledge about both HPC and AI architectures, since there is today still a technological gap between HPC and deep learning (cf. Sect. 2).

Of most interest for HPC users are so-called hybrid HPC/AI workflows. These workflows combine classical simulations with AI, e.g. simulations that can be exploited to create vast amounts of synthetic data that then can be used to train DL models. In this context, we presented our first case study on the prediction of material characteristics (cf. Sect. 4.1). Experiments showed that such a hybrid workflow successfully improves typical simulation-only workflows; reducing the overall runtimes while still achieving a high accuracy of parameter predictions; 90% of errors are, across all experiments, significantly below 0.1.

Furthermore, the second case study on satellite image segmentation successfully demonstrated leveraging HPC for typical AI workloads, such as unsupervised image segmentation, where large regions need to be analyzed (cf. Sect. 4.2). The experiment showed that the implemented W-net method is fully dynamic and independent of labeled data while retaining a precision accuracy of around 90%. Such fully unsupervised segmentation and identification of elements in high-resolution remote sensing images on HPC has great advantages in very large areas since it can eliminate most of the cost and time introduced by manual work.

Looking into the future, we see several bottlenecks that currently restrain the adoption of AI on HPC, and thus impede the breakthrough of AI worldwide. Although some DL frameworks, e.g. Horovod have enabled scaling of AI applications on HPC clusters, more essential works still need to be performed in order to improve their feasibility, portability and compatibility.In addtion, techniques applied in scaling distributed DL on HPC are converging to the point where a standard programming interface (or framework) can be designed, which can make the definition of a training scheme easier and hide most of the low-level layers omnipresent in HPC. Ideally, automated AI services must be offered to all stakeholders ranging from AI experts over SMEs to non-experts to lower the hurdle of using HPC to solve AI challenges.

Acknowledgements This work has been supported by the project CATALYST, which is funded by the Ministry of Science, Research and Arts (MWK), Baden-Württemberg, Germany. The authors would like to acknowledge the Institute for Metal Forming Technology (IFU) of the University of Stuttgart and National Observatory of Athens (NOA) for providing the dataset of the two use cases.

References

1. Dale, R.: GPT-3: What's it good for?. Nat. Lang. Eng. **27**(1), 113–118 (2021)
2. Manning, C. D., Manning, C. D., & Schütze, H.: Foundations of statistical natural language processing. MIT press (1999)

3. Forsyth, D. A., Ponce, J.: Computer Vision: a Modern Approach. Prentice Hall Professional Technical Reference, (2002)
4. Hoppe, D., Gienger, M., Bönisch, T., Shcherbakov, O., Moise, D.: Towards seamless integration of data analytics into existing HPC infrastructures. In: Proceedings of the Cray User Group (CUG), Redmond, WA, USA. HPE Apollo (Hawk), https://www.hlrs.de/systems/hpe-apollo-hawk/(2017). accessed
5. Kadupitige, K.: Intersection of hpc and Machine Learning, Digital Science Center, (2017)
6. Kerestély, Á. (2020). HIGH PERFORMANCE COMPUTING FOR MACHINE LEARNING. Bulletin of the Transilvania University of Brasov. Mathematics, Informatics, Physics. Series III, 13(2), 705-714
7. Abspoel, M., Scholting, M.E., Lansbergen, M., An, Y., Vegter, H.: A new method for predicting advanced yield criteria input parameters from mechanical properties. J. Mater. Process. Technol. **248**, 161–177 (2017)
8. Amodei, D., Hernandez, D.: AI and Compute. https://openai.com/blog/ai-and-compute/(2019). accessed on 29 Apr 2022
9. Reinsel, D., Gantz, J., Rydning, J.: Data Age 2025: The Evolution of Data to Life-Critical. https://www.import.io/wp-content/uploads/2017/04/Seagate-WP-DataAge2025-March-2017.pdf (2017). accessed on 29 Apr 2022
10. Sergeev, A., Del Balso, M.:"Horovod: fast and easy distributed deep learning in TensorFlow "(2018). *arXiv preprint* arXiv:1802.05799
11. Mathuriya, A., Kurth, T., Rane, V., Mustafa, M., Shao, L., Bard, D., Lee, V. W.: Scaling grpc tensorflow on 512 nodes of cori supercomputer (2017). arXiv preprint arXiv:1712.09388
12. Bathe, K.-J.: Finite Element Method, Wiley Online Library, (2008)
13. Lorente, D., Martínez-Martínez, F., Rupérez, M.J., Lago, M.A., Martínez-Sober, M., Escandell-Montero, P., Martín-Guerrero, J.D.: A framework for modelling the biomechanical behaviour of the human liver during breathing in real time using machine learning. Expert. Syst. Appl. **71**, 342–357 (2017)
14. Luo, R., Shao, T., Wang, H., Xu, W., Zhou, K., Yang, Y.: Deepwarp: Dnn-based nonlinear deformation (2018). arXiv preprint arXiv:1803.09109
15. Huerta, E.A., Khan, A., Davis, E. et al.: Convergence of artificial intelligence and high performance computing on NSF-supported cyberinfrastructure. J. Big Data. **7**, 88 (2020). https://doi.org/10.1186/s40537-020-00361-2
16. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Zheng, X.: TensorFlow: Large-scale machine learning on heterogeneous systems, Software available from tensorflow.org (2015)
17. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Chintala, S.: PyTorch: An Imperative Style, High-Performance Deep Learning Library, In: Advances in Neural Information Processing Systems 32, pp. 8024-8035. Curran Associates, Inc (2019)
18. Barlat, F., Aretz, H., Yoon, J.W., Karabin, M.E., Brem, J.C., Dick, R.E.: Linear transfomation-based anisotropic yield functions. Int. J. Plast. **21**(5), 1009–1039 (2005)
19. Abspoel, M., Scholting, M.E., Lansbergen, M., An, Y., Vegter, H.: A new method for predicting advanced yield criteria input parameters from mechanical properties. J. Mater. Process. Technol. **248**, 161–177 (2017)
20. hpc-workspace. https://github.com/holgerBerger/hpc-workspace. accessed 2 May 2022
21. HPW APOLLO (HAWK). https://www.hlrs.de/systems/hpe-apollo-hawk/. accessed 2 May 2022
22. CRAY CS-STORM. https://www.hlrs.de/systems/cray-cs-storm/. accessed 2 May 2022
23. Drusch, M., Del Bello, U., Carlier, S., Colin, O., Fernandez, V., Gascon, F., Hoersch, B., Isola, C., Laberinti, P., Martimort, P., Meygret, A., Spoto, F., Sy, O., Marchese, F., Bargellini, P.: SENTINEL-2: ESA's optical high-resolution mission for GMES operational services. Remote. Sens. Environ. **120**, 25–36 (2012)
24. Inglada, J., Vincent, A., Arias, M., Marais-Sicre, C.: Improved early crop type identification by joint use of high temporal resolution SAR and optical image time series. Remote. Sens. **8**(5), 362 (2016)

25. Xia, X., Kulis, B.: W-net: A deep model for fully unsupervised image segmentation (2017). arXiv preprint arXiv:1711.08506
26. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional Networks for Biomedical Image Segmentation. In: International Conference on Medical image computing and computer-assisted intervention, pp. 234-241. Springer, Cham (2015)
27. Shi, L., Huang, H., Shi, Y., Hu, Y.: W-net: The convolutional network for multi-temporal high-resolution remote sensing image arable land semantic segmentation. In: Journal of Physics: Conference Series, Vol. 1237, No. 3, p. 032067. IOP Publishing (2019)
28. Fritz, S., See, L., You, L., Justice, C., Becker-Reshef, I., Bydekerke, L., Woodcock, C.: The need for improved maps of global cropland. vol. 94 (3), pp. 31–32. Eos, Transactions American Geophysical Union (2013)
29. Yifang, B., Gong, P., Gini, C.: Global land cover mapping using Earth observation satellite data: Recent progresses and challenges. ISPRS J. Photogramm. Remote. Sens. (Print) **103**(1), 1–6 (2015)
30. Sitokonstantinou, V., Koukos, A., Drivas, T., Kontoes, C., Papoutsis, I., Karathanassi, V.: A scalable machine learning pipeline for paddy rice classification using multi-temporal sentinel data. Remote. Sens. **13**(9), 1769 (2021)
31. Ben-Nun, T., Hoefler, T.: Demystifying parallel and distributed deep learning: An in-depth concurrency analysis. ACM Comput. Surv. (CSUR) **52**(4), 1–43 (2019)
32. Li, M., Andersen, D. G., Park, J. W., Smola, A. J., Ahmed, A., Josifovski, V., Su, B. Y.: Scaling distributed machine learning with the parameter server. In: 11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14). pp. 583-598 (2014)
33. Krizhevsky, A.: One weird trick for parallelizing convolutional neural networks (2014). arXiv preprint arXiv:1404.5997
34. Huang, Y., Cheng, Y., Bapna, A., Firat, O., Chen, D., Chen, M., Wu, Y.: Gpipe: Efficient training of giant neural networks using pipeline parallelism. In: Advances in Neural Information Processing Systems, pp. 103-112 (2019)
35. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems, vol. 27, (2014)
36. Mozaffari, A., Langguth, M., Gong, B., Ahring, J., Campos, A.R., Nieters, P., Schultz, M.G.: HPC-Oriented canonical workflows for machine learning applications in climate and weather prediction. Data Intell. **4**(2), 271–285 (2022)
37. Lee, H., Turilli, M., Jha, S., Bhowmik, D., Ma, H., Ramanathan, A.: Deepdrivemd: Deep-learning driven adaptive molecular simulations for protein folding. In 2019 IEEE/ACM Third Workshop on Deep Learning on Supercomputers (DLS), IEEE, 12-19 (2019)
38. Archibald, R., Chow, E., D'Azevedo, E., Dongarra, J., Eisenbach, M., Febbo, R., Yin, J.: Integrating deep learning in domain sciences at exascale. In: Smoky Mountains Computational Sciences and Engineering Conference, pp. 35-50 . Springer, Cham (2020)