ICCGI 2022

The Seventeenth International Multi-Conference on Computing in the Global Information Technology

ISBN: 978-1-61208-972-0

May 22nd –26th, 2022

Venice, Italy

ICCGI 2022 Editors

Juho Mäkiö, Hochschule Emden/Leer, Germany

ICCGI 2022

Forward

The Seventeenth International Multi-Conference on Computing in the Global Information Technology (ICCGI 2022) continued a series of events covering a large spectrum of topics related to global knowledge concerning computation, technologies, mechanisms, cognitive patterns, thinking, communications, user-centric approaches, nanotechnologies, and advanced networking and systems. The conference topics focused on challenging aspects in the next generation of information technology and communications related to the computing paradigms (mobile computing, database computing, GRID computing, multi-agent computing, autonomic computing, evolutionary computation) and communication and networking and telecommunications technologies (mobility, networking, bio-technologies, autonomous systems, image processing, Internet and web technologies), towards secure, self-defendable, autonomous, privacy-safe, and context-aware scalable systems.

This conference intended to expose the scientists to the latest developments covering a variety of complementary topics, aiming to enhance one’s understanding of the overall picture of computing in the global information technology.

We take here the opportunity to warmly thank all the members of the ICCGI 2022 technical program committee, as well as all the reviewers. The creation of such a high-quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and effort to contribute to ICCGI 2022. We truly believe that, thanks to all these efforts, the final conference program consisted of top-quality contributions. We also thank the members of the ICCGI 2022 organizing committee for their help in handling the logistics of this event.

ICCGI 2022 Chairs

ICCGI 2022 Steering Committee

Constantin Paleologu, Polytechnic University of Bucharest, Romania
Peter Kieseberg, St. Pölten University of Applied Sciences, Austria
Yasushi Kambayashi, Nippon Institute of Technology, Japan
Ulrich Reimer, University of Applied Sciences St. Gallen, Switzerland
Kathryn E. Stecke, University of Texas at Dallas, USA
Juho Mäkiö, Hochschule Emden / Leer, Germany

ICCGI 2022 Publicity Chairs

Mar Parra, Universitat Politècnica de València (UPV), Spain
Hannah Russell, Universitat Politècnica de València (UPV), Spain

ICCGI 2022
Committee

ICCGI 2022 Steering Committee

Constantin Paleologu, Polytechnic University of Bucharest, Romania
Peter Kieseberg, St. Pölten University of Applied Sciences, Austria
Yasushi Kambayashi, Nippon Institute of Technology, Japan
Ulrich Reimer, University of Applied Sciences St. Gallen, Switzerland
Kathryn E. Stecke, University of Texas at Dallas, USA
Juho Mäkiö, Hochschule Emden / Leer, Germany

ICCGI 2022 Publicity Chairs

Mar Parra, Universitat Politècnica de València (UPV), Spain
Hannah Russell, Universitat Politècnica de València (UPV), Spain

ICCGI 2022 Technical Program Committee

Ahmed M. Abdelmoniem, KAUST, Saudi Arabia
António Abreu, ISEL/IPL, Portugal
Abdelouhab Aitouche, EEA - French School of High Studies in Engineering, France
Mohammad Yahya Al-Shamri, King Khalid University, Saudi Arabia
Alma Y. Alanis. University of Guadalajara, Mexico
Fernando Almeida, INESC TEC & University of Porto, Portugal
José Antonio Apolinário Junior, Military Institute of Engineering (IME), Rio de Janeiro, Brazil
Nancy Arana Daniel, University of Guadalajara, Mexico
Jocelyn Aubert, Luxembourg Institute of Science and Technology (LIST), Luxembourg
Afef Awadid, Technological Research Institute SystemX, Palaiseau, France
Michaela Baumann, NÜRNBERGER Versicherung, Germany
Robert Bestak, Czech Technical University in Prague, Czech Republic
Dorota Bielinska-Waz, Medical University of Gdansk, Poland
Fernando Bobillo, University of Zaragoza, Spain
Zorica M. Bogdanovic, University of Belgrade, Serbia
Eugen Borcoci, University "Politehnica" of Bucharest (UPB), Romania
Jean-Louis Boulanger, CERTIFER, France
Christos Bouras, University of Patras, Greece
Juan Carlos Burguillo-Rial, Universidade de Vigo, Spain
Pricila Castelini, Federal University of Technology, Brazil
DeJiu Chen, KTH Royal Institute of Technology, Sweden
Albert M. K. Cheng, University of Houston, USA
Rebeca Cortazar, University of Deusto, Spain
Pietro Cunha Dolci, Santa Cruz do Sul University, Brazil
Beata Czarnacka-Chrobot, Warsaw School of Economics, Poland
José Carlos da Silva Freitas Junior, UNISINOS (Vale do Rio dos Sinos University), Porto Alegre, Brazil
Maria de los Angeles Cosio León, Universidad Politécnica de Pachuca, Mexico
Laura-Maria Dogariu, University Politehnica of Bucharest, Romania

Alexander Makarenko, Institute of Applied System Analysis at National Technical University of Ukraine (Igor Sikorski Kiev Politechnic Institute), Ukraine
Juho Mäkiö, Hochschule Emden / Leer, Germany
Henrique S. Mamede, INESC-TEC | Universidade Aberta, Lisboa, Portugal
Goreti Marreiros, Institute of Engineering | Polytechnic of Porto, Portugal
Juan Carlos Montes de Oca López, Autonomous University of the State of Mexico, Mexico
Sobhan Moosavi, Ohio State University, USA
Fernando Moreira, Universidade Portucalense Infante D. Henrique, Portugal
Elaine Mosconi, Université de Sherbrooke, Canada
Paulo Moura Oliveira, UTAD University | INESC-TEC, Portugal
Mary Luz Mouronte López, Universidad Francisco de Vitoria, Spain
Marco Mugnaini, University of Siena, Italy
Rafael Nogueras, University of Malaga, Spain
Robert Ohene-Bonsu Simmons, University of Ghana-BS-OMIS / Zenith University College, Ghana
Alexandru Onea, Technical University "Gh. Asachi' of Iasi, Romania
Constantin Paleologu, University Politehnica of Bucharest, Romania
Thanasis G. Papaioannou, Athens University of Economics and Business (AUEB), Greece
Dhaval Patel, IBM TJ Watson Research Center, USA
Bernhard Peischl, AVL List GmbH, Austria
Mansah Preko, Ghana Institute of Management and Public Administration (GIMPA), Ghana
Kornelije Rabuzin, University of Zagreb, Croatia
M. Sohel Rahman, Bangladesh University of Engineering & Technology (BUET), Bangladesh
Ferdinand Regner, University of Vienna, USA
Fernando Reinaldo Ribeiro, Polytechnic Institute of Castelo Branco, Portugal
Éric Renault, ESIEE Paris, France
Ruben Ricart-Sanchez, University of the West of Scotland, UK
Michele Risi, University of Salerno, Italy
Federica Rollo, "Enzo Ferrari" Engineering Department - University of Modena and Reggio Emilia, Italy
Takfarinas Saber, University College Dublin, Ireland
Carlos D. Santos Jr., Universidade de Brasilia, Brazil
Wieland Schwinger, Johannes Kepler University Linz (JKU), Austria
Floriano Scioscia, Polytechnic University of Bari, Italy
Isabel Seruca, Portucalense University, Porto, Portugal
Ashok Sharma, Geethanjali College of Engineering and Technology, India
Shwadhin Sharma, California State University, Monterey Bay, USA
J. P. Shim, Georgia State University, USA
Alessio Signorini, Evidation Health, USA
Luis Silva Rodrigues, Politécnico do Porto / ISCAP, Portugal
Pornpat Sirithumgul, Rajamangala University of Technology Phra Nakhon, Thailand
Pedro Sousa, University of Minho, Portugal
Kathryn E. Stecke, University of Texas at Dallas, USA
Vitalyi-Igorevich Talanin, Khortytsia National Academy, Zaporozhye, Ukraine
Francesco Tedesco, University of Calabria, Italy
Chrisa Tsinaraki, EU JRC, Italy
Ion Tutanescu, University of Pitesti, Romania
Radu Vasiu, Politehnica University of Timisoara, Romania
John Violos, École de Technologie Supérieure, Montreal, Canada
Piotr Waz, Medical University of Gdansk, Poland

Copyright Information

Table of Contents

Cookie Monsters on Media Websites

Dark Patterns in Cookie Consent Notices

Esther van Santen
Department of Business and Management,
Brandenburg University of Applied Sciences
Brandenburg an der Havel, Germany
Email: e.vs@posteo.de

Abstract–The EU's General Data Protection Regulation attempts to improve the protection of user's privacy by demanding that website operators have legitimate reason to process personal data. In the context of the use of cookies, therefore, usually consent is needed. The use of dark patterns stands in the way of valid consent and can be classified as unethical design. This contribution explores the occurrence and frequency of dark patterns in cookie consent notices for media outlet websites in Germany and the United States. Each examined cookie consent notice contained at least one dark pattern and 4.8 dark patterns on average. The dark pattern Privacy Zuckering is present on most researched websites in Germany and the U.S. The dark pattern Preselection is present on a quarter of German and U.S. American websites. The findings indicate that there are dark patterns which could be more prevalent in cookie consent notices. One newly described dark pattern could be specific to the context of consent notices on media websites.

Keywords-cookie consent notice; dark pattern; ethical design; media websites.

I. INTRODUCTION

In an attempt to maintain compliance with the EU's General Data Protection Regulation (GDPR), and at the same time to fulfill their own interests as uncompromisingly as possible, website operators resort to dark patterns when designing cookie consent notices. The concept of dark patterns was introduced into the discourse by Brignull in 2010 and describes design patterns within any kind of user interface that "trick users into things they wouldn't otherwise have done" [1]. In a more recent definitional approach with a legal focus, Martini et al. reject definitions based on users' agency or website operators' intent and speak instead of user interface design that "exploits the design power (...) unilaterally in the interest of website operators" [2]. This exploitation could result in cookie consent forms generating invalid consent under the GDPR, due to the non-voluntary or uninformed nature of the user's decision, as demonstrated in a legal assessment by Kuehling [3].

To improve both compliance and user experience in the context of cookie consent banners, it is important to deepen the public understanding of dark pattern occurrence and mechanisms. Previous research on frequency of dark patterns within cookie consent notices has focused on different aspects: Consent management platforms [4], specific categories of websites [5] or specific countries [6, 7]. This work contributes to dark pattern research insights for cookie consent notices by evaluating a set of 100 media outlet websites in the U.S. and Germany for the occurrence and frequency of different dark patterns. The analysis showed that there is indication that some dark patterns might be more prevalent in the context of cookie consent banners, and that there is one dark pattern that could be specific to consent notices for media websites. In Section 2, previous findings for the context of dark patterns in cookie consent notices are explored. Section 3 describes the methods that were used to examine the cookie consent notices on media outlet websites for dark patterns based on the dark pattern taxonomy suggested by Bösch et al. [2]. In Section 4, a frequency analysis for those dark patterns is performed and three further dark patterns specific for cookie consent notices are described. The conclusion in Section 5 completes this paper.

II. RELATED WORK

In 2016, Bösch et al. described and investigated dark patterns with an explicit tie to privacy concerns occurring on digital user interfaces, to form a framework that facilitates the documentation of dark patterns [8]. Gray et al. developed a taxonomy for Brignull's initial dark patterns in 2018 and enriched this collection with dark patterns that they found while examining their corpus of 118 examples from various websites [9]. Gray's taxonomy is based on five categories that were formed according to the influence that the design patterns included have on a user. Another classification, established in 2019 by Mathur et al., is based on a corpus of approximately 11000 shopping websites, from which they derived seven categories which are also based on how a set of patterns influences users [10].

There are also several papers, which focus specifically on dark patterns within cookie consent notices: Nouwens et al. compared consent management platforms for frequency of dark patterns and testing the influence of the most occurring patterns on user's consent decisions in 2020. They found out that the willingness to consent is increased by over 20 % if there is no option to decline on the first view of a consent notice. If there are more options to modify the extent of the consent on the first view, consent is reduced by 8 to 20 %.

ISBN: 978-1-61208-972-0

Through a study with a set of 300 cookie consent notices that were collected from online news websites in 2020, Soe et al. found indications for seven dark patterns specific to cookie consent notices [5]. The research was also based on Gray's taxonomy. Soe et al. give two reasons for focussing on news websites: These websites are intended to appeal to a broad target group and therefore they expect that the consent notices are quite generic in design. In addition, compared to social networks, news websites are not expected to be primarily interested in the advertising suitability of the processed data. In 2021, Kampanos and Shahandashti focused their research on the comparison of cookie consent notices on Greek and British websites in terms of interaction options [6]. Therefore, dark patterns were only addressed implicitly. They found that most of the investigated websites for both countries did not offer direct decline options. UK websites were more likely to violate the GDPR by not including cookie consent notices, even though there was use of third-party cookies. Krisam et al. examined 389 German websites in 2021 for frequency of choice options in consent notices [7]. They provide an analysis on which choice options have to be seen as dark patterns, but refrain from connecting them with existing taxonomy. Out of 389 websites, 69 would not be legally compliant. As Krisam et al. are more focussed on compliance than ethics of design, they do not offer a total for dark patterns within their research set.

III. Methods

The set of 100 media outlet websites (50 German, 50 U.S. American) that this research is based on, was obtained manually. Using U.S. American websites was a deliberate choice to enable a comparison of an EU country with one outside the EU, as the GDPR is still applicable if a non-EU website is accessed by a person within the EU. However, it can be assumed that the GDPR's requirements for cookie consent notices are not as well-known and strictly enforced outside the EU as within the EU. Also, this could lead to different solution approaches for obtaining consent.

In contrast to the consideration made by Soe et al. concerning media websites as subjects, the decision for media outlet websites was based on an attempt to achieve better comparability of the subjects and to possibly find industry-specific dark patterns.

The set of websites was manually obtained and fixed before starting the analysis. The following research hypotheses were formed:

- RQ1: Which already described dark patterns can be found in consent notices on media websites?
- RQ2: Are there dark patterns that have not yet been described and could therefore be specific to cookie consent notices or to media websites?
- RQ3: Are there country-specific differences in which dark patterns occur in cookie consent notices on German and U.S. American websites?

This study is based on the taxonomy by Gray et al. because its context is more generic than the taxonomy by Mathur et al., which is based only on the analysis of shopping websites. Nevertheless, after further reviewing the dark patterns contained in the classification of Gray et al., it became obvious that not all dark patterns made sense in the context of cookie consent notices. Some dark patterns were tied to a specific context, e.g. online shopping. Therefore, each category within the taxonomy was reduced to a set of dark patterns that seemed to be applicable to cookie consent notices. This set of dark patterns formed the basis for the quantitative visual analysis of cookie consent notices within the set of media websites. As Nagging is a category that does not contain any dark patterns, the category itself has to be analysed, counted and measured against other dark patterns rather than other categories.

TABLE I. Categorisation of Dark Patterns

Category	Contained dark patterns
Nagging	(Nagging)
Obstruction	Roach Motel
Sneaking	Bait & Switch, Hidden Costs
Interface Interference	Hidden Information, Preselection, Toying with Emotion, False Hierarchy, Misdirection
Forced Action	Privacy Zuckering

Before starting the analysis, it was necessary to establish a sound research environment. Any network measures, such as IP obfuscation and network-wide tracking blockers were deactivated for the course of the quantitative analysis. Then, the cookie notices were visually evaluated for the occurrence of dark patterns. In case of ambiguities, for example when determining whether a contrast ratio is sufficiently low for the information or button to be considered hidden, the provisions of the WCAG 2.1 (Web Content Accessibility Guidelines) were used to set limit values.

IV. Result

Out of all 100 websites, 67 contained a cookie consent notice. However, the existence of cookie consent notices was not distributed evenly for German and U.S. American websites. While 80 % of German websites contained a consent notice, only 57 % of American websites did.

Out of the 67 websites that did contain a cookie consent notice, all 67 consent notices contained at least one dark pattern each. On average, each cookie consent notice contained 4.8 dark patterns. These are the cases for which the existence of a dark pattern within a consent notice was confirmed:

TABLE II. Occurrence of Dark Patterns

Dark pattern	Properties
Hidden Information	(1) The relevant information is set in a very small font (less than 12 pt). (2) Or the contrast ratio of the text to the background is too low (less than 3.5:1).
Preselection	Checkboxes are already checked or toggles are already set to “confirmed”.

ISBN: 978-1-61208-972-0

Dark pattern	Properties
False Hierarchy	(1) “Accept” and “decline” or “options” are unevenly sized. (2) Or One the options is a button, the other is solely a text link. (3) Or options are unnecessarily stacked on top of each other.
Misdirection	(1) The colour scheme of the option buttons does not match usual colour schemes; usual meaning of colours is reversed (green is used for declining instead of accepting). (2) Or a consent option is offered in legitimate interest, although this reason for permission does not depend on user’s consent.
Hidden Costs	Information about data processing is non-existent or not detailed enough.
Roach Motel	Rejecting cookies cannot be done via the consent notice itself, but only via a more complicated way (for example by email).
Privacy Zuckering	If data is transferred to third parties for ordered data processing or for sale.
Nagging	The website is impossible to navigate without responding to the consent notice.

These already described dark patterns were present in the research set: Privacy Zuckering (59 instances), Nagging (42), False Hierarchy (40), Hidden Costs (37), Hidden Information (18), Preselection (16), Misdirection (14) and Roach Motel (11). There were no instances of either Toying with Emotion or Bait & Switch.

The occurrence of a category was affirmed for a website as soon as a dark pattern of this category was found. Forced Action occurred for almost every website (60 instances) and the remaining categories were found on more than half of the websites: Interface Interference (55 instances), Obstruction (47) and Sneaking (35). It is noticeable that the four most frequently found dark patterns belong to different categories. Since the categories in Grey et al.’s taxonomy are based on the effect that design choices have on the way that they influence a user, this is not surprising. After all, if dark patterns are supposed to persuade users to consent, why not work multiple angles? The results of this study are merely a small contribution to answering the first research question, as it was too broad to begin with. However, this indication could be a starting point for comparing these results with a larger set of media websites.

In addition, three dark patterns were found and described, for which no existing dark pattern definitions could be found in the reviewed papers. These dark patterns are:

Unclear Directions, which is a form of Sneaking and shall be defined as the concealment of a path which is required by the user. This is the case when the cookie management button is not recognisable because it is misleadingly named, e.g., “show purposes”, or if the link for more specific information on data processing is not sufficiently labelled, e.g., with “here” or “more”.

Denial Maze, which is a form of Obstruction and which shall be defined as an interface design where it is more difficult to express disagreement than agreement. For this study, this was the case when more clicks were needed to reject than to accept the use of cookies.

Conditional Access, which is a form of Forced Action and which shall be defined as a situation in which access can only take place if one accepts undesirable consequences. This means that users were only informed about the use of cookies without the possibility of interaction and, above all, the possibility of refusing the use of cookies or if the use of cookies is part of the business model and therefore only possible to refuse by paying for it. This business model approach could only be found on German websites. These results can only be seen as indications. In order to actually arrive at an answer to the second research question, it would be necessary to compare all three dark patterns with user interfaces other than consent notices in order to find out whether this mechanism is specific to consent notices. Furthermore, for the dark pattern Conditional Access, it would have to be examined whether this mechanism also applies to non-media websites.

For this research set, the dark pattern that occurred most for both countries was Privacy Zuckering, which appeared on 85 % of German websites and on 92.6 % of U.S. American websites. The dark pattern Preselection occurred on almost the same percentage of websites for both countries: It could be found on 22.5 % of German websites and on 25 % of U.S. American websites.

There were notable differences for the occurrence of two dark patterns concerning the German and the U.S. American websites: The dark pattern False Hierarchy occurred on 70 % of German websites and on 44 % of U.S. American websites. The dark pattern Roach Motel occurred on 10 % of German websites but on 25 % of U.S. American websites. For all remaining dark patterns, the difference in occurrence was lower than 15 %.

V. CONCLUSION

Starting with an already small sample size and then further limiting it, minimizes the significance of the results. For a further attempt, it would be useful to either choose a bigger set of websites in the beginning or to filter the set, so that every website contains a cookie consent notice. Research questions should be more specific to generate valid results. With regard to the fact that only 67 percent of the selected research objects contained a cookie consent notice, the following should be noted: It can be legal and in compliance with the GDPR to operate a website without a consent notice. However, the low percentage of U.S. American websites that contained a consent notice and the presence of dark patterns within every consent notice that was examined, indicates that the research set at least partially contained websites on which the absence of a cookie consent notice might not be lawful. Whether this is actually the case, however, would require a deeper technical and legal analysis.

REFERENCES

[1] H. Brignull, “Dark Patterns: Dirty Tricks Designers use to make People do Stuff,” Available from: https://90percentofeverything.com/2010/07/08/dark-patterns-dirty-tricks-designers-use-to-make-people-do-stuff [retrieved: May, 2022].

[2] M. Martini, C. Drews, P. Seelinger, and Q. Weinzierl, “Dark Patterns: Phänomenologie und Antworten der Rechtsordnung [Dark Pattern: Phenomenology and Responses of the Legal

System],” Zeitschrift für Digitalisierung und Recht [Journal for Digitization and Law], vol. 2021, no. 1, pp. 47-74.

[3] J. Kuehling, ”Rechtliche Rahmenbedingungen sogenannter ‘Dark Patterns‘ [Legal Determinants of so-called ‘Dark Patterns‘],” Available from: https://www.bevh.org/fileadmin/content/04_politik/Europa/Kuehling-Gutachten-BEVH-Dark-pattern-22-02-16-final.pdf [retrieved: May, 2022].

[4] M. Nouwens, I. Liccardi, M. Veale, D. Karger, and L. Kagal, „Dark Patterns after the GDPR: Scraping Consent Pop-ups and Demonstrating their Influence,” Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI ’20), April 2020, paper no. 194, pp. 1-13, doi: 10.1145/3313831.3376321.

[5] T. H. Soe, O. E. Nordberg, F. Guribye, and M. Slavkovik, “Circumvention by Design – Dark Patterns in Cookie Consents for Online News Outlets,” Proceedings of the 11th Nordic Conference on Human-Computer Interaction: Shaping Experiences, Shaping Society (NordiCHI '20), October 2020, article no. 19, pp. 1-12, doi:10.1145/3419249.3420132.

[6] G. Kampanos and S. F. Shahandashti, “Accept All: The Landscape of Cookie Banners in Greece and the UK,” ICT Systems Security and Privacy Protection, June 2021, pp. 213-227, doi: 10.1007/978-3-030-78120-0_14.

[7] C. Krisam, H. Dietmann, M. Volkamer, and O. Kulyk, “Dark Patterns in the Wild: Review of Cookie Disclaimer Designs on Top 500 German Websites,” European Symposium on Usable Security 2021 (EuroUSEC '21), October 2021, pp. 1–8, doi.org/10.1145/3481357.3481516.

[8] C. Bösch, B. Erb, Benjamin, F. Kargl, H. Kopp, and S. Pfattheicher, “Tales from the Dark Side: Privacy Dark Strategies and Privacy Dark Patterns,” Proceedings on Privacy Enhancing Technologies, vol. 2016, no. 4, pp. 237-254, 2016, doi:10.1515/popets-2016-0038.

[9] C. M. Gray, Y. Kou, B. Battles, J. Hoggatt, and A. L. Toombs, “The Dark (Patterns) Side of UX Design,” Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18), April 2018, paper no. 534, pp. 1-14, doi:10.1145/3173574.3174108.

[10] A. Mathur et al., “Dark Patterns at Scale: Findings from a Crawl of 11K Shopping Websites,” Proceedings of the ACM on Human-Computer Interaction, Volume 3, Issue CSCW, November 2019, pp 1-32, doi:10.1145/3359183.

Semantic Approaches for Cognitive Data Processing

Lidia Ogiela
AGH University of Science and Technology
30 Mickiewicza Ave, 30-059 Kraków, Poland
e-mail: logiela@agh.edu.pl

Urszula Ogiela
AGH University of Science and Technology
30 Mickiewicza Ave, 30-059 Kraków, Poland
e-mail: ogiela@agh.edu.pl

***Abstract*— This paper describes a semantic-based technique for intelligent data processing and cognitive analysis. The described methods are used to create knowledge extraction procedures, which apply the semantic content and meaning for data handling. Such methods are designed for efficient data management and protection in cloud environment.**

Keywords-Cognitive reasoning; data processing and management; semantic description.

I. INTRODUCTION

Modern security protocols very often use semantic content description of secured data and involve it in creation of security algorithms. Such methods were proposed in the new area of cognitive cryptography in [1][2]. In such methods the semantic content should be evaluated and applied in the security protocol, which finally results in the encrypted data being dependent on its semantic meaning. Such protocols define an important extension of traditional security procedures, which usually do not make any connection between semantic content and final encryption results.

Similar connections to the ones mentioned in the previous paragraph between semantic content and protocols can be implemented in data management techniques as well. Such techniques will be described in following sections, in which we will define semantic-based secure data management approaches. The main idea of such procedures is to create a new class of strategic data management procedures oriented for information splitting and distribution in complex, hierarchical management structures. Information splitting and distribution will be strongly dependent on the content of shared data [3]-[5]. The main action of such techniques will be connected with a semantic content evaluation, which will provide the data feature vector. Feature parameters from this vector will be used in the information division task.

The rest of the paper is structured as follows. In Section II, we present the concept of data management using semantic information. In Section III, we mention some applications of management protocols. We conclude the paper in Section IV.

II. DATA MANAGEMENT USING SEMANTIC INFORMATION

In order to define semantic-based management algorithms, it is necessary to introduce two different types of protocols. The first type of protocols includes techniques which allow to evaluate the semantic content of encrypted information. The second type of protocols contains efficient data division protocols which allow to share secret data into a particular number of parts, which can then be distributed among users in management procedures. In such techniques, the distribution of secret parts should be connected with the content and implemented with the application of semantic parameters extracted at the beginning of the procedures.

For extraction of semantic description, we can use the cognitive information systems defined in [1][6]. In the past, several different classes of cognitive procedures were defined, which focused on the evaluation of different types of data, from visual patterns, to economical or secret data.

Cognitive systems are aimed at extracting the semantic content from analyzed data and evaluating some important knowledge which is present in the data. Very often, this requires extensive analysis, including the application of advanced Artificial Intelligence (AI), or cognitive resonance procedures. As a result of cognitive analysis, it is possible to build a data record which contains the semantic description of the analyzed information. Such semantic record can contain a large number of parameters describing different global or local features. Depending on the goal of the information splitting in management procedures, it is possible to select the most important parameters from this information, which can then be applied to perform the data splitting and distribution tasks in an efficient and secure manner.

When we select several semantic features, we can implement them using a division and management protocol. To perform such task, first, it is necessary to select a data sharing technique [3][5] and apply it for complex hierarchical management structures. To do this, it is necessary to determine the number of levels and layers in the hierarchical structure, as well as the number of participants at each level. Having selected the parameters and having evaluated the semantic features of the divided information, we can start the division procedures with the following input parameters:

- semantic factors,
- defined numbers of layers and participants,
- secret information that needs to be splitted,
- starting parameters for sharing procedures.

After finishing secret data division sequences, we obtain a particular number of secret parts, which can then be distributed to each level in the hierarchical structure. Distribution can be done in different ways depending on the number of persons and the access privileges. We can consider a specially defined distribution topology for the obtained secret parts, which can be placed in an irregular manner over different levels in the hierarchical structure.

III. Application of Management Protocols

The defined semantic-based sharing and management procedures have several possible areas of application. Such techniques extend classical management procedures towards including semantic content. Such techniques are dependent on features, and the information can be splitted and distributed in different ways. The possibilities of selection of semantic parameters introduce an additional security level because the whole protocol allows to reconstruct the original data only in the situation when the input parameters are known. The knowledge about the procedure will not be enough to perform unauthorized data reconstruction from the generated parts.

The security feature allows to apply these types of protocols in different management or security areas. In particular, they may be applied in secure and trusted data management in distributed systems, like cloud structures [7][8]. It can be also implemented in distant services management, as well as secure data storage and distribution. Performing analytics tasks with the application of semantic feature on the analyzed data makes such protocols also applicable in predictive analysis towards prognosis of user trends or behaviors [9][10].

IV. Conclusions

In this paper, we described a new idea of creation and application of semantic-based protocols in security areas. Such methods can be used in a broad range of management activities, especially connected with secret data division in complex and distributed structures. The main idea of such protocols lays in the extraction of the semantic meaning of encrypted data and the application of such information in security protocols. The extraction of semantic meaning can be done with the application of cognitive information systems, and the extracted features can decide about the way of information encoding and distribution. Such techniques can be widely applied in cloud computing and distributed services management, as well as secure data distribution in complex structures. Such methods enrich traditional management approaches and have influence in the creation of new security protocols in cognitive cryptography [1][11].

Acknowledgment

This work has been supported by the National Science Centre, Poland, under project number DEC-2016/23/B/HS4/00616.

References

[1] L. Ogiela and M. R. Ogiela, “Cognitive security paradigm for cloud computing applications,” Concurr. Comput.: Pract. Exp. 32(8), e5316, 2020, doi: 10.1002/cpe.5316.

[2] M. R. Ogiela, L. Ogiela, and U. Ogiela, “Biometric methods for advanced strategic data sharing protocols,”. In: 2015 9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing IMIS 2015, pp. 179–183, 2015, doi: 10.1109/IMIS.2015.29.

[3] M. R. Ogiela and U. Ogiela, “Secure information splitting using grammar schemes,” New Challenges in Computational Collective Intelligence. Studies in Computational Intelligence, vol. 244, pp. 327–336. Springer, Heidelberg, 2009, doi: 10.1007/978-3-642-03958-4_28.

[4] S. Nakamura, L. Ogiela, T. Enokido, and M. Takizawa, “Flexible Synchronization Protocol to Prevent Illegal Information Flow in Peer-to-Peer Publish/Subscribe Systems,” in: Barolli, L., Terzo, O. (Eds.), Complex, Intelligent, and Software Intensive Systems, Advances in Intelligent Systems and Computing. 611, pp. 82-93, 2018.

[5] N. Ferguson and B. Schneier, “Practical Cryptography,” Wiley, 2003.

[6] L. Ogiela, “Transformative computing in advanced data analysis processes in the cloud,” Inf. Process. Manage. 57(5), 102260, 2020.

[7] R. A. Ancheta, F. C. Reyes, J. A. Caliwag, and R. E. Castillo, “FEDSecurity: implementation of computer vision thru face and eye detection,” Int. J. Mach. Learn. Comput., 8, pp. 619–624, 2018.

[8] S. Gil et al., “Transformative effects of IoT, Blockchain and Artificial Intelligence on cloud computing: Evolution, vision, trends and open challenges,” Internet of Things, 8, 100118, 2019.

[9] C. Guan, J. Mou, and Z. Jiang, “Artificial intelligence innovation in education: a twenty-year data-driven historical analysis,” Int. J. Innov. Stud. 4(4), 134–147, 2020.

[10] S. J. H. Yang, H. Ogata, T. Matsui, and N.-S. Chen, “Human-centered artificial intelligence in education: seeing the invisible through the visible,” Comput. Educ.: Artif. Intell. 2, 100008, 2021.

[11] M. Del Giudice, V. Scuotto, B. Orlando, and M. Mustilli, “Toward the human – Centered approach. A revised model of individual acceptance of AI,” Human Resource Management Review, 2021, 100856, doi: 10.1016/j.hrmr.2021.100856.

Autoencoder vs. Regression Neural Networks for Detecting Manipulated Wine Ratings*

Michaela Baumann
Business Intelligence / Analytics Competence Center
NÜRNBERGER Versicherung
Nürnberg, Germany
email: michaela.baumann@nuernberger.de
ORCID: 0000-0001-5066-9624

Michael Heinrich Baumann
Department of Mathematics
University of Bayreuth
Bayreuth, Germany
email: michael.baumann@uni-bayreuth.de
ORCID: 0000-0003-2840-7286

***Abstract*—In this study, we analyze the ability of different (neural network based) detection methods to identify manipulated wine ratings for two "vinho verde" datasets. We find that autoencoders perform best on unmanipulated test data. However, regressions outperform autoencoders in terms of true/false positive rates on the manipulated test data in median. This is interesting, since autoencoders are generally used for outlier detection. Furthermore, hyperparameter tuning via sequential accumulative selection is established.**

***Keywords*—anomaly detection; manipulation identification; wine preferences; artificial neural networks; autoencoder.**

I. Introduction

In a world of increasingly differentiated products and customers who frequently change their buying behavior, it is difficult to assess whether the price-performance ratio is appropriate before making a purchase. An important and much-used assistance in such buying decisions are ratings. In this study, we are going to approach the question of whether and how manipulated ratings can be detected using wine quality ratings as an example. When ratings come from an official or non-official authority (such as Gambero Rosso's *Vini d'Italia* [1], Robert Parker's *The Wine Advocate* [2], *Gault&Millau* [3], or *Guide Michelin* [4], when dealing with wines, hotels, restaurants, or related topics), it is possible to verify with little effort whether ratings given by a merchant or producer are genuine by simply looking up the relevant work. However, since by far not all wines are represented and rated in one of the works published by an authority, there are countless other ratings. These other ratings, which are not given by an authority, are difficult to verify for authenticity, and it might even be possible that they are not objective, but rather paid for by someone. In the following, we are going to show possibilities for detecting such manipulated or faked ratings.

A very basic idea for how to identify manipulated ratings would be via (linear) regressions. That means, when we have other, exactly measurable features, such as alcohol content, pH value, or density, we can learn how to predict the rating using these independent variables on correctly rated data objects. Ratings that differ (strongly) from the predicted ones on unseen data might be suspicious. The described methodology is commonly used in many contexts, such as in economics and finance, and often leads to useful results (cf. [5]). However, a linear regression does not lead to good results in our case, i.e., when trying to detect manipulated wine ratings. Thus, the research question is how manipulated wine ratings may be detected in a better way. Since artificial neural networks are currently en vogue, one can of course use a regression by means of a neural network (cf. [6]–[9]). Note that a linear regression is the same as an exactly trained, fully connected neural network without any hidden layer with linear activation functions (i.e., id resp. pass-through), when adding a dummy column (filled with 1s) in the data for the intercept and using Mean Squared Error (MSE) as loss. Regressions based on shallow or deep neural networks are likely to outperform a linear regression.

Especially when dealing with outlier detection, so-called autoencoders (resp. reconstruction networks or auto-associative neural networks) are a common means [10]–[13]. Autoencoders consist of two parts (i.e., two regressions), an encoder and a decoder. The encoder compresses the input data to a lower dimensional representation usually referred to as the code; the decoder takes the code as input and aims to reconstruct the original input.

Given a well trained autoencoder, when the input and the output differ (strongly), the data might be manipulated (or in other contexts: an outlier, an anomaly, fraudulent, or suspicious). Note that there are much more application areas of autoencoders, such as dimensionality reduction, data compression, or denoising. Although it is in principal assumed that the quality depends on the other features, the autoencoder does not use this information, that is, the quality and all other features are considered as coequal input (and output) variables. Since the autoencoder does not use all the information that is actually available, it would be very interesting if it nevertheless achieved better results.

In the work at hand, we investigate how *Regression Neural Networks* (RNN) and *Neural Network based Autoencoders* (NNA) can be used to identify manipulated data. Additionally, as benchmark models we use a *linear regression* (Linear Model; LM; see [5]) and an autoencoder that implements two linear regressions (*Benchmark Autoencoder;* BA; see Section IV-D). Clearly, there are several other data analytics methods that might be applied, e.g., support vector machines

*Corrected version, October 2022

ISBN: 978-1-61208-972-0

[14], however, an investigation is postponed to future work, as it would go well beyond the scope of this paper, especially since the number of those techniques keeps growing (see [15][16]).

We find that neural network based autoencoders perform best on unmanipulated test data. However, they are not that useful for detecting manipulated wine ratings since for this task, the regression neural networks show a better detection performance in median. This may be unexpected since one of the main application areas of autoencoders is anomaly detection. However, the regression neural network shows a great variability on the unmanipulated test data, i.e., its behavior is not that stable especially compared to a benchmark linear regression model, whose computation time is considerably lower.

The remainder of this paper is organized as follows: Section II reviews both the literature on wine data analysis and those on anomaly detection in general while Section III specifies the data we are using. Sections IV and V describe the method we use and Section VI presents the results. Finally, Sections VII and VIII conclude and describe possibilities for ongoing work.

II. Literature Review

The closely related literature roughly splits into two groups, namely data analytics of wine quality and general outlier/anomaly detection resp. fraud identification. The analytics of wine quality mostly covers the prediction of wine ratings based on measurable features. Cortez et al. [17][18] compare several data mining regression methods for predicting wine preferences based on easily available data during the certification of wines. In this context, they originally published the two datasets that are also used in the work at hand. They use the vinho verde white wine dataset [18] and both the vinho verde red wine and white wine datasets [17]. Besides these papers, also the importance of the selection of the most relevant features before predicting the wine quality with machine learning regression methods is investigated for the vinho verde datasets [19]. The vinho verde white wine dataset is used for classifying wine preferences via fuzzy inductive reasoning [20]. Deep neural networks are applied for classifying wine ratings (in detail: multiple classification) on the vinho verde datasets [21].

Several regression models for assessing wine quality are developed with data from southern France consisting of altogether 137 variables (vineyard variables and enological variables) to assist the winemakers in their business [22]. Also, the effect of weather and climate changes as well as the effect of expert ratings on the prices of Bordeaux wines are analyzed [23][24]. With this, the efficiency of the Bordeaux wine market is assessed. Tree models are used for predicting the relative quality of German Rhinegau Riesling considering terrain characteristics obtained through cartographic studies [25]. A framework is developed that automatically finds an appropriate set of classifiers and hyperparameters via evolutionary optimization for predicting wine quality for arbitrary wine datasets [26].

Having in mind the literature reviewed above, which predicts wine ratings or conducts data analyses of wine quality, the work at hand contributes by connecting wine rating predictions and anomaly detection. The topic of outlier/anomaly detection and fraud identification is addressed in a lot of related work in various contexts (see, for example, the surveys and summaries [27]–[31]) and we can only touch on this broad topic here. Generally, according to Chandola et al., "*Anomaly detection* refers to the problem of finding patterns in data that do not conform to expected behavior" [27]. Usually, anomalies have to be identified throughout the analysis of data so that they can be treated separately and do not distort the results of the analysis of "normal" data. However, in the case of fraud and also in our case of manipulation detection they are of special interest. Fraudulent and manipulated data objects inhibit abnormal patterns but they try to appear as normal. The detection of anomalies, especially of intentional, malicious anomalies, such as fraud or manipulation, is very challenging and there are many approaches that try to accomplish this task. The approaches basically fall in one of the following three categories [28]:

- Unsupervised methods (e.g., clustering); labels are not needed here and new patterns (normal ones and outliers) may be processed correctly.
- Supervised methods (e.g., classification); this needs prelabeled data, however, anomalies are usually very rare and the labeled datasets are, thus, highly unbalanced; new patterns are unlikely to be processed correctly.
- Semi-supervised methods (e.g., autoencoders); normal behavior is known, i.e., (a part of) the training data is labeled as normal, and new, unlabeled data objects are compared to the normal case.

In addition to methods that require tabular data (a priori tabular data, but also image, audio, or video data transferred to tabular data) there are methods that operate on graph based data [32], which are especially useful when identifying anomalies in highly connected data. The approach of the work at hand falls into the third category, i.e., semi-supervised methods, and works on tabular data.

III. Data

The approach described in this work is applicable to various working areas (see Section VIII). We demonstrate it using wine data as an example because of the following reasons.

A rather simple advantage is the good data availability and (if no wine names or winemaker names are used) the innocuousness of the data. Further, the explaining variables (except for wine or winemaker names) are metric, clearly defined, and exactly measurable (e.g., alcohol content, acid, pH value, red/white). The used datasets further have a unique target feature and not a list of ratings (see also Section VIII).

We use the "Wine Quality Datasets" [17] from the Universidade do Minho [33], more specifically the datasets "White

Wine Quality—Simple and clean practice dataset for regression or classification modelling" [34] and "Red Wine Quality—Simple and clean practice dataset for regression or classification modelling" [35] downloaded from *kaggle,* which are licensed under "Database Contents License (DbCL) v1.0," *Database: Open Database, Contents: Database Contents* [36]. Both datasets contain anonymized *vinho verde* wines and have the same twelve columns, i.e., features, namely: "fixed acidity," "volatile acidity," "citric acid," "residual sugar," "chlorides," "free sulfur dioxide," "total sulfur dioxide," "density," "pH," "sulfates," "alcohol," and "quality." There, *quality* is the wine rating, which is supposed to depend on the other, explaining features, called independent. All values except for the ratings are in some meaningful physical unit, while the ratings range from 0 (very bad) to 10 (excellent) in integer steps [17]. The red wine dataset consists of 1,599 entries while the white wine data has 4,898 rows, leading to a combined data set with 6,497 rows and 13 columns. For the distinction of red and white wines we added a binary encoded categorical column. Please note that we do include neither descriptive statistics like plots or correlations, nor explorative analyses, such as clusterings, nor distribution estimations etc. in this work. There is already a lot of such work done for the vinho verde datasets. Such statistics and many more analyses can be found in the work of Cortez et al. [17][18], in other papers [19]–[21], and further tutorials or notebooks [37]–[41].

IV. Methodology

As outlined in Section I, the aim of this work is to identify manipulated ratings. For this, we train several network based models on the provided, correct data. We then make predictions on unseen data objects where we manipulate a certain part of these objects. As manipulation, we increase the original rating of very low rated wines as this seems to be a "reasonable" manipulation in the context of wine ratings (when someone wants to increase sales numbers). By comparing the provided, potentially manipulated data and the predicted data we aim at identifying the manipulated data objects. Objects for which the predicted values strongly differ from the provided data are more likely to be manipulated. We assess the models' detection performance, i.e., their ability to identify manipulations through calculating the true and false positive rates when marking the most deviating data objects as suspicious. To prevent overfitting and account for other random effects we apply bootstrapping. That is, we repeat the process of randomly splitting the data and training the models. Finally, we take among others the median over the particular results.

In the following, we explain our methodology in detail. The implementation is done in `R` using the `Keras` library, which is an API to `TensorFlow`, for the neural networks.

A. Bootstrapping and Data Splitting

The bootstrapping is in our case a Monte-Carlo-like approach of repeatedly and independently splitting the complete dataset $\underline{all}$ (6,497 rows, 13 columns) with a ratio of 70:30 into training data (4,547 rows, 13 columns) and test data (1950 rows, 13 columns) 100 times: $\underline{all} = \underline{train} \,\dot{\cup}\, \underline{test}$. To make this process reproducible, we set an initial seed and randomly draw 100 seeds $(seed_1, \dots, seed_{100})$. Before every splitting we explicitly set the seed to the respective run's seed. The training data is further split with a ratio of 70:30 into development data (3,182 rows, 13 columns) and validation data (1,365 rows, 13 columns): $\underline{train} = \underline{dev} \,\dot{\cup}\, \underline{val}$.

B. Data Manipulation

As an example, we manipulate the 5% worst ranked test data by averaging the original rating and the highest possible rating (10) and rounding up. That is, we split the test data $\underline{test} = \underline{low} \,\dot{\cup}\, \underline{high}$ with a ratio of 5:95 (with a random tie breaking), manipulate $\underline{low} \mapsto \underline{low_{manip}}$ and get the manipulated test data $\underline{manip} := \underline{low_{manip}} \,\dot{\cup}\, \underline{high}$. We also add a flag column to the manipulated test data for marking the manipulated entries for evaluation purposes.

C. Data Normalization

The independent features are all normalized by min-max-scaling where $\underline{train}$ serves as reference. That means, also the test datasets are normalized with the minimum and maximum values of $\underline{train}$. For LM and RNN, the target variable "quality" is not normalized. For BA and NNA, "quality" is an input variable like the others and, hence, normalized. To obtain comparable results, the performance of the regression models is normalized afterwards (using $\underline{train}$).

D. Models

We consider four different kinds of models: LM, RNN, BA, and NNA. The two simple models LM and BA are solely for benchmarking the general performance of the two corresponding (deep) neural network models on the unmanipulated test data $\underline{test}$. We measure the manipulation detection performance for the two (deep) neural network models only. LM uses `R`'s `lm` function. BA is a fully connected, three layer network with input layer (size 14), code layer (size 4), and output layer (size 14). The input is the 13 dimensional data plus a constant column of 1s (intercept) in order to mimic two nested linear regressions. Thus, linear activation functions and MSE are used.

E. Hyperparameter Tuning

In every step during the bootstrapping, the deep models are trained with hyperparameter optimization over a grid. How these grids are obtained is outlined in Section V. The hyperparameter grid for RNN is:

- Activation function (hidden layers): `linear`, `softplus`, `ReLU`
- Activation function (output layer): `linear`
- Number of hidden layers: 1, 3, 5, 7
- Dropout rate: 0%, 5%, 10%
- Number of neurons in each hidden layer: 32, 64, 128
- Number of neurons the input layer: 12
- Number of neurons the output layer: 1

- Batch size: 32, 64
- Learning rate: 5%, 10%
- Patience for early stopping: 15
- Patience for learning rate reduction: 7
- Loss function: MSE
- Evaluation measure: Mean Absolute Error (MAE)
- Optimizer: `Adam`
- Number of epochs: 75
- Batch normalization: between every layer

The grid for NNA is defined as follows:

- Activation function (hidden layers, except the code): `softplus`, `ReLU`
- Activation function (code layer and output layer): `linear`
- Number of hidden layers (except code layer): 4, 6
- Dropout rate: 0%
- Number of neurons in each hidden layer (except the code): 64, 128
- Number of neurons the input layer as well as in the output layer: 13
- Number of neurons the code layer: 4
- Batch size: 32, 64
- Learning rate: 5%, 10%
- Patience for early stopping: 15
- Patience for learning rate reduction: 7
- Loss function: MSE
- Evaluation measure: MAE
- Optimizer: `Adam`
- Number of epochs: 75
- Batch normalization: between every layer

F. The Algorithm

The bootstrapping, model training, and evaluation algorithm is depicted in the algorithm in Figure 1. All individual steps are described above. The algorithm is parallelized.

```
begin
for i=1 to $n$ do
    begin
    Prepare datasets with $seed_i$ (split, manipulate, normalize);
    Train the two benchmark models on $\underline{train}$;
    Optimize RNN's and NNA's hyperparameters (train on $\underline{dev}$, validate on $\underline{val}$ using MAE as performance measure) and retrain the best model in each case on $\underline{train}$;
    Measure all four models' performance on $\underline{test}$;
    Measure RNN's and NNA's detection performance on $\underline{manip}$;
    end
end
```

Figure 1. Procedure for model training and evaluation. Input: the original dataset; a seed vector $(seed_1, \ldots, seed_{100})$; two hyperparameter grids. Output: list of performance data.

As one can see from the algorithm, both approaches (RNN, NNA) are semi-supervised. We use labeled data to train the networks, but only data that is labeled as "correct," i.e., that is not manipulated. Although in the analysis, "correct" and "incorrect," i.e., manipulated, data are used, no incorrect data are used for training—that is, one does not need a data set where "incorrect" data are already identified as incorrect. We use the information about which data entries are really "incorrect" only for the statistical analysis of the results for this paper.

G. Detection Performance

The detection performance is measured as follows: For RNN, we calculate the squared difference of the predicted quality and the given quality (which is possibly manipulated) for each data object (Squared Error; SE). For NNA, we compute for all data objects the sum over all features of the squared differences between the predicted feature and the respective given (possibly manipulated) feature (Sum of Squared Errors; SSE). We sort the data in descending order according to these deviation values (once for RNN and once for NNA): $\underline{manip} \mapsto (\underline{manip_{reg}}, \underline{manip_{auto}})$. Then, we determine the true/false positive rates when marking the first $q_i\%$ of the data objects in the sorted sets $\underline{manip_{reg}}$ and $\underline{manip_{auto}}$ as suspicious for $q_i = i$, $i = 1, 2, \ldots, 99$. The True Positive Rate tpr is defined as $tpr = TP/(TP + FN) = 1 - fnr$ and the False Positive Rate fpr is $fpr = FP/(TN + FP) = 1 - tnr$, where TP is the number of True Positives, i.e., of manipulated objects that are marked suspicious, TN is the number of True Negatives, i.e., of unmanipulated objects that are not marked, and FP and FN are the respective False Positives/Negatives and fnr and tnr the respective Rates. If one would assign the "suspicious marks" randomly with equal probabilities to $q\%$ ($q \in [0, 100]$) of the data, the expected true/false positive rates would equal q, i.e., $\mathbb{E}[tpr] = \mathbb{E}[fpr] = q$, independent of the share of real positives/negatives. The values for $q = 0$ and $q = 100$ are meaningless since in the former case no object would be marked as suspicious and in the latter case all objects would be marked as suspicious. To summarize the results of all runs, we calculate all quartiles of tpr and fpr for every q_i, i.e., minimum, first quartile, median, third quartile, maximum. Before presenting the results of our analysis in Section VI, we describe how the set of possible hyperparameters is found.

V. Hyperparameters

Since basically the set of possible hyperparameters is infinite, it is quite natural that this set has to be shrunk. In doing so, we start with an initial set for possible hyperparameters and with an initial guess for a plausible setting (underlined). This is done based on comparisons to similar problem as well as extensive trail-and-error pre-tests.

The initial hyperparameter grid for RNN is:

- Activation function: `linear`, `softplus`, $\underline{\texttt{ReLU}}$, `tanh`, `sigmoid`
- Number of hidden layers: 0, 1, 3, 5, $\underline{7}$
- Dropout rate: 0%, $\underline{5\%}$, 10%
- Number of neurons in each hidden layer: 32, 64, $\underline{128}$
- Batch size: $\underline{32}$, 64

- Learning rate: 5%, $\underline{10\%}$

The initial grid for NNA is:

- Activation function: `linear`, `softplus`, `ReLU`, `tanh`, `sigmoid`
- Number of hidden layers (excluding the code layer): 0, 2, 4, $\underline{6}$
- Dropout rate: 0, $\underline{0.05}$, 0.1
- Number of neurons in each hidden layers (except the code layer): 32, 64, $\underline{128}$
- Batch size: $\underline{32}$, 64
- Learning rate: 0.05, $\underline{0.1}$

All other parameters are fixed to the values of Section IV-E. Note that we intentionally did not include varying numbers of neurons for the code layer (in the autoencoder case). This is because higher numbers of neurons in the code lead to a higher performance, but a lower compression. Since both values are important for outlier detection, based on comparisons to similar examples, we chose four as a promising tradeoff.

Using the heuristic strategy of *sequential accumulative selection,* this set is further shrunk so that the hyperparameter optimization in the algorithm in Figure 1 (in Section IV) performs within a reasonable runtime. Next, we explain the sequential accumulative selection: We started with performing 50 runs with the hyperparameters fixed to the underlined, plausible values except for the activation function, which was allowed to be any of the given possibilities. All activation functions that were taken at least once in the hyperparameter optimization in the 50 runs were declared to be also plausible, all others were deleted. In the same fashion, next, the number of hidden layers was analyzed, i.e., the hyperparameter optimizer had to optimize over the set of the plausible activation functions (due to step one there is possibly more than one plausible activation function) and the number of hidden layers. All values for the hidden layers that were chosen at least once were declared to be also plausible, all others deleted. The plausible activation functions remain the same. This procedure is repeated in the following order with number of neurons, dropout rate, batch size, learning rate. The results of the sequential accumulative selection, i.e., of the diminution of the possible hyperparameters can be found in Section IV-E. For clear, the procedure of sequential accumulative selection is done separately for RNN and NNA.

VI. Results

Here, we do set neither an explicit threshold for the share of data objects to be marked as suspicious nor an explicit threshold for the SE resp. SSE beyond which the data objects have to be marked as suspicious since the aim of this work is not to find a classifier for manipulated wine data quality, but the comparison of the two neural network based models. How a threshold can be found is, e.g., outlined in [42]. To illustrate the detection performance of RNN and NNA, we calculate tpr and fpr for all Monte-Carlo-like runs and for all $q_i = 1, \ldots, 99$. For all q_i, we calculate the five quartiles of tpr and fpr for RNN and NNA and plot these values against q. The results are depicted in Figures 2 (tpr) and 3 (fpr). The respective quartiles of NNA are drawn solid and of RNN dashed.

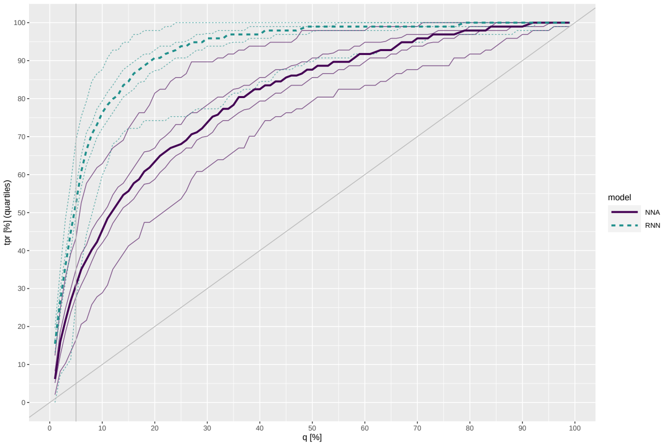

Figure 2. The five quartiles of tpr for varying q and RNN (dashed), NNA (solid), resp. Additionally, the diagonal and the 5% line are depicted. RNN outperforms NNA in median, but not in all cases and for all q.

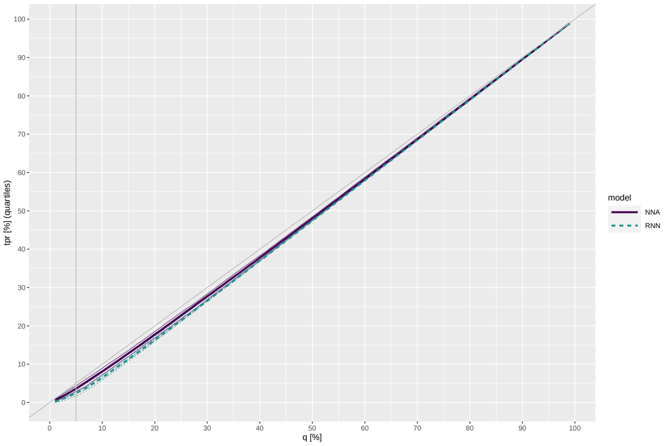

Figure 3. The five quartiles of fpr for varying q and RNN (dashed), NNA (solid), resp. Additionally, the diagonal and the 5% line are depicted. RNN outperforms NNA in median, but not in all cases and for all q.

As we can easily observe, the regression network outperforms the autoencoder in most of the cases (remember that an autoencoder does not use the information about the assumed dependency) concerning tpr and fpr. Both models are better than randomly guessing (cf. the diagonals in the figures). The average runtime of RNN was with ca. 2h14'06" much larger than those of NNA (ca. 15'40.8"). The performance of all four models on the unmanipulated test data is depicted in Figure 4.

We see that NNA is best (in median), while autoencoders are better than regressions (in median). RNN is (in median) a little better than LM, however, its variability is the largest among all models, while those of LM is the smallest.

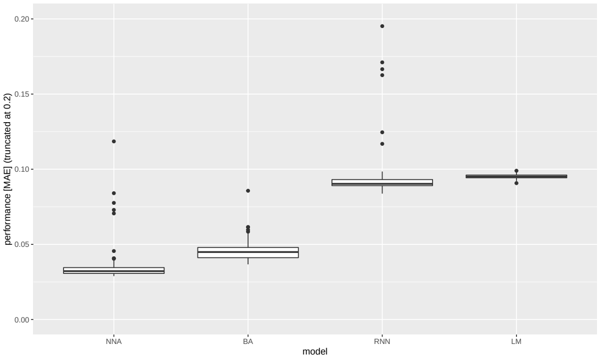

Figure 4. Boxplot of the performance (MAE) of NNA, BA, RNN, LM. There are some outliers that are not depicted. In median, NNA is best, whilst the interquartile distance is the smallest for LM.

VII. Conclusion

We analyzed four models for predicting resp. reconstructing wine quality, two benchmark models and two deep neural network based models (regression neural network and autoencoder network). We considered one resp. two specific datasets, the vinho verde data. We find that a neural network based autoencoder performs best on unmanipulated test data while a linear regression shows the smallest variability in the results.

We then analyzed the ability of the two deep neural network models for detecting manipulated wine ratings. It turns out that in our study regressions outperform autoencoders on this task although autoencoders are generally used for outlier detection. There is a lot of literature concerning regressions from measurable data to wine quality. Interestingly, such regressions seem to work well also in our case for semi-supervised manipulation identification. Further, we established the procedure of sequential accumulative selection for finding appropriate hyperparameters.

VIII. Future Work

In this paper, we assumed that it is reasonable that manipulations are applied to low rated wines to make them appear better to increase sales numbers. It would be interesting to test our approach also on other manipulation strategies, including, e.g., intentional and unjustified down ratings. Future work could also deal with the detection of faked ratings when there are multiple ratings per product as it is typical for many online stores or rating portals. Are there ways to detect the faked/manipulated ratings (whether better or worse) when there are many ratings for the same product? In this context, many stores and portals offer the possibility to write a review in addition to the plain rating. The processing of such information (via Natural Language Processing; NLP) is likely to be useful here.

Of course, other application areas apart from wine can be investigated with our approach, for example, ratings for products in online stores, restaurants, hotels. The detection of fraud in telecommunication, insurance, etc. [43] is also closely related. It could be of interest to identify the similarities and differences between these applications and how they should be addressed. When analyzing wine ratings, in addition to extending our approach to other, larger datasets with more features, such as countries, producing regions, price segments, etc., it is also worthwhile to apply other models, e.g., SVMs [14], and compare the results to the neural network based models. Further, an extensive comparison with other methodologies concerning the topic of manipulation detection for wine ratings could be done in future work.

The procedure of *sequential accumulative selection* (as explained in Section V) can be further analyzed. One might investigate whether and how the order of the features is important. Comparisons to other hyperparameter selection methods are also possible (cf. [26]). Last but not least, it should be noted that the topic of explainable AI and responsible AI is rapidly growing in importance [44]. For example, one can ask how to explain which data sets are marked as suspicious. As few as possible false positives are to be marked, whereas all manipulated ones are to be recognized if possible. So how can the decisions of the recognition algorithms be (understandably) explained?

Acknowledgment

Michaela Baumann is with NÜRNBERGER Versicherung, Germany. The opinions expressed here are her own and not necessarily those of her employer.

The authors thank Lars Grüne and Bernhard Herz, both with University of Bayreuth, Germany, as well as Claire Pfeil and Heinrich Fritzlar, both with NÜRNBERGER Versicherung, Germany.

References

[1] G. Rosso, Italian Wines 2021 *(English Edition),* Gambero Rosso, 2021

[2] R. Parker, The Wine Advocate, https://www.robertparker.com/articles/the-wine-advocate, accessed: 2022-02-02

[3] Gault&Millau, https://www.gaultmillau.com/, accessed: 2022-02-02

[4] Guide Michelin, https://guide.michelin.com/en, accessed: 2022-02-02

[5] D. Freedman, R. Pisani, and R. Purves, Statistics, 4th ed., W. W. Norton & Company, Inc., New York, London, 2007, Chapters 10-12

[6] E. Gelenbe, Z. H. Mao, and Y. D. Li, “Function approximation with spiked random networks,” in IEEE Transactions on Neural Networks, vol. 10, no. 1, 1999, pp. 3-9

[7] E. Gelenbe, “Random neural networks with negative and positive signals and product form solution,” in Neural Computataion, vol. 1, no. 4, 1989, pp. 502-510

[8] T. Poggio, H. Mhaskar, L. Rosasco, B. Miranda, and Q. Liao, “Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review,” in International Journal of Automation and Computing, vol. 14, no. 5, 2017, 503-519

[9] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” in nature, vol. 521, no. 7553, 2015, pp. 436-444

[10] S. Hawkins, H. He, G. Williams, and R. Baxter, “Outlier Detection Using Replicator Neural Networks,” Data Warehousing and Knowledge Discovery, 2002, pp. 170-180

[11] M. Sakurada and T. Yairi, “Anomaly Detection Using Autoencoders with Nonlinear Dimensionality Reduction,” Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis, 2014, pp. 4-11

[12] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” in science, vol. 313, no. 5786, 2006, pp. 504-507

[13] J. D. Kelleher, Deep learning, MIT press, 2019

[14] I. Steinwart and A. Christmann, Support Vector Machines, Springer, 2008

[15] D. L. Donoho, “High-dimensional data analysis: The curses and blessings of dimensionality,” in AMS math challenges lecture, 2000

[16] J. W. Tukey, “The future of data analysis,” in The Annals of Mathematical Statistics, vol. 33, no. 1, Institute of Mathematical Statistics, 1962, pp. 1-67

[17] P. Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis, “Modeling wine preferences by data mining from physicochemical properties,” in Decision Support Systems, vol. 47, no. 4, 2009, pp. 547-553

[18] P. Cortez et al., “Using data mining for wine quality assessment,” in International Conference on Discovery Science, Springer, Berlin, Heidelberg, 2009, pp. 66-79

[19] Y. Gupta, “Selection of important features and predicting wine quality using machine learning techniques,” in Procedia Computer Science, vol. 125, 2018, pp. 305-312

[20] À. Nebot, F. Mugica, and A. Escobet, “Modeling wine preferences from physicochemical properties using fuzzy techniques,” in SIMULTECH, 2015, pp. 501-507

[21] S. Kumar, Y. Kraeva, R. Kraleva, and M. Zymbler, “A deep neural network approach to predict the wine taste preferences,” in Intelligent Computing in Engineering, Springer, Singapore, 2020, pp. 1165-1173

[22] P. Abbal, J. M. Sablayrolles, E. Matzner-Lober, and A. Carbonneau, “A model for predicting wine quality in a rhône valley vineyard,” in Agronomy Journal, vol. 111, no. 2, 2019, 545-554

[23] O. Ashenfelter, “Predicting the quality and prices of Bordeaux wine,” in The Economic Journal, vol. 118, no. 529, 2008, F174-F184

[24] O. Ashenfelter, “Predicting the quality and prices of Bordeaux wine,” in Journal of Wine Economics, vol. 5, no. 1, 2010, 40-52

[25] R. Schwarz, “Predicting wine quality from terrain characteristics with regression trees,” in Cybergeo: European Journal of Geography, 1997

[26] T. H. Y. Chiu, C. Wu, and C. H. Chen, A generalized wine quality prediction framework by evolutionary algorithms,” in International Journal of Interactive Multimedia & Artificial Intelligence, vol. 6, no. 7, 2021, pp. 60-70

[27] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly Detection: A Survey”, ACM Comput. Surv., vol. 41, no. 3, 2009, article no. 15, pp. 1-15

[28] V. Hodge and J. Austin, “A Survey of Outlier Detection Methodologies.” Artificial Intelligence Review, vol. 22, 2004, pp. 85-126

[29] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, “Network Anomaly Detection: Methods, Systems and Tools,” in IEEE Communications Surveys & Tutorials, vol. 16, no. 1, 2014, pp. 303-336

[30] A. Patcha and J.-M. Park, “An overview of anomaly detection techniques: Existing solutions and latest technological trends,” in Computer Networks, vol. 51, no. 12, 2007, pp. 3448-3470

[31] R. Chalapathy and S. Chawla, “Deep Learning for Anomaly Detection: A Survey,” preprint on arXiv, https://arxiv.org/abs/1901.03407, 2019

[32] C. C. Noble and D. J. Cook, “Graph-Based Anomaly Detection,” Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003, pp. 631-636

[33] Wine Quality Datasets, Universidade do Minho, http://www3.dsi.uminho.pt/pcortez/wine/, accessed: 2022-02-02

[34] kaggle (Piyush Agnihotri), White Wine Quality—Simple and clean practice dataset for regression or classification modelling, https://www.kaggle.com/piyushagni5/white-wine-quality, accessed: 2022-01-19

[35] kaggle (UCI Machine Learning), Red Wine Quality—Simple and clean practice dataset for regression or classification modelling, https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009, accessed: 2022-01-17

[36] Open Data Commons—Legal tools for Open Data, Database Contents License (DbCL) v1.0, https://opendatacommons.org/licenses/dbcl/1-0/, accessed: 2022-01-19

[37] T. Shin, “Predicting Wine Quality with Several Classification Techniques” towards data science, 2020, https://towardsdatascience.com/predicting-wine-quality-with-several-classification-techniques-179038ea6434, accessed: 2022-02-02

[38] D. Nguyen, “Red Wine Quality Prediction Using Regression Modeling and Machine Learning,” Towards Data Science, 2020, https://towardsdatascience.com/red-wine-quality-prediction-using-regression-modeling-and-machine-learning-7a3e2c3e1f46, accessed: 2022-02-02

[39] F. Rodríguez Mir, “Red Wine Quality,” Data UAB, 2019, https://datauab.github.io/red_wine_quality/, accessed: 2022-02-02

[40] *unknown* “Wine Quality Prediction,” cppsecrets.com, 2021, https://cppsecrets.com/users/101261001041051149710611211411110610199641031099710510846991111109/WINE-QUALITY-PREDICTION.php, accessed: 2022-02-02

[41] D. Alekseeva, “Red and White Wine Quality,” RPubs, https://rpubs.com/Daria/57835, accessed: 2022-02-02

[42] N. Japkowicz, C. Myers, and M. Gluck, “A Novelty Detection Approach to Classification,” IJCAI, vol. 1, 1995, pp. 518-523

[43] M. Baumann, “Improving a rule-based fraud detection system with classification based on association rule mining,” INFORMATIK 2021, 2021, pp. 1121-1134

[44] M. Baumann, “Data science challenge 2021: explainable machine learning,” https://github.com/DeutscheAktuarvereinigung/Data-Science-Challenge2021_Explainable-Machine-Learning, accessed: 2022-02-04