PhD-FSTC-2019-07
The Faculty of Sciences, Technology and Communication

DISSERTATION

Defence held on 18/01/2019 in Luxembourg
to obtain the degree of

DOCTEUR DE L'UNIVERSITÉ DU LUXEMBOURG

EN INFORMATIQUE

by

Patrick GLAUNER
Born on 19 January 1989 in Pforzheim (Germany)

Artificial Intelligence for the Detection of Electricity Theft and Irregular Power Usage in Emerging Markets

Dissertation defence committee

Dr Radu State, dissertation supervisor
Professor, Université du Luxembourg

Dr Björn Ottersten, Chairman
Professor, Université du Luxembourg

Dr Djamila Aouada, Vice Chairman
Research Scientist, Université du Luxembourg

Dr Petko Valtchev
Professor, Université du Québec à Montréal

Dr Holger Vogelsang
Professor, Karlsruhe University of Applied Sciences

Declaration

I herewith certify that all material in this thesis which is not my own work has been properly acknowledged.

Patrick Glauner, MBA

Abstract

Power grids are critical infrastructure assets that face non-technical losses (NTL), which include, but are not limited to, electricity theft, broken or malfunctioning meters and arranged false meter readings. In emerging markets, NTL are a prime concern and often range up to 40% of the total electricity distributed. The annual world-wide costs for utilities due to NTL are estimated to be around USD 100 billion. Reducing NTL in order to increase revenue, profit and reliability of the grid is therefore of vital interest to utilities and authorities. In the beginning of this thesis, we provide an in-depth discussion of the causes of NTL and the economic effects thereof.

Industrial NTL detection systems are still largely based on expert knowledge when deciding whether to carry out costly on-site inspections of customers. Electric utilities are reluctant to move to large-scale deployments of automated systems that learn NTL profiles from data. This is due to the latter's propensity to suggest a large number of unnecessary inspections. In this thesis, we compare expert knowledge-based decision making systems to automated statistical decision making. We then branch out our research into different directions: First, in order to allow human experts to feed their knowledge in the decision process, we propose a method for visualizing prediction results at various granularity levels in a spatial hologram. Our approach allows domain experts to put the classification results into the context of the data and to incorporate their knowledge for making the final decisions of which customers to inspect. Second, we propose a machine learning framework that classifies customers into NTL or non-NTL using a variety of features derived from the customers' consumption data as well as a selection of master data. The methodology used is specifically tailored to the level of noise in the data. Last, we discuss the issue of biases in data sets. A bias occurs whenever training sets are not representative of the test data, which results in unreliable models. We show how quantifying and reducing these biases leads to an increased accuracy of the trained NTL detectors.

This thesis has resulted in appreciable results on real-world big data sets of millions customers. Our systems are being deployed in a commercial NTL detection software. We also provide suggestions on how to further reduce NTL by not only carrying out inspections, but by implementing market reforms, increasing efficiency in the organization of utilities and improving communication between utilities, authorities and customers.

Acknowledgements

First and foremost, I would like to offer my sincerest gratitude to my supervisor Professor Radu State for making this opportunity possible, for setting the direction of this thesis and for providing valuable regular feedback. Next thanks must go to Professor Björn Ottersten and Dr. Franck Bettinger for agreeing to join my thesis committee during the last 3 years. I am grateful to Professor Björn Ottersten for chairing my thesis defense. I would like to thank Professor Petko Valtchev for hosting a research visit at his laboratory at UQAM and for having applied his proofreading gurudom to my manuscripts during the last 3 years. Furthermore, I would like to thank Dr. Djamila Aouada and Professor Holger Vogelsang for agreeing to join my thesis defense jury.

I am grateful to the Luxembourg National Research Fund, the Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg and CHOICE Technologies Holding Sàrl for funding my research. Without the generous funding, I would not have been able to travel to a large number of international conferences and summer schools. Further thanks go to Diogo Duarte and his team at CHOICE Technologies Holding Sàrl for providing real-world data and their support.

The last couple of years in SEDAN Lab have been a lot of fun and I would like to thank all the other present and past members for discussions, proof-reading and for having been great company. Special thanks go to Niklas Dahringer, Christian Glaeser, Stanislav Konchenko, Martin Lehmann, Oleksandr Puhachov and Lucian Trestioreanu for helping to implement some of my ideas during their internships.

During my PhD research, I became an adjunct faculty member at the Universities of Applied Sciences in Karlsruhe and Trier. I would like to thank both departments for hiring me. It has been a particularly fruitful experience to teach at both departments. That experience has furthermore motivated me to work even harder on my research in order to lay the groundwork for a potential regular faculty position in the future.

I am eternally obliged to the support from my wife Shengqin Yang during these busy years. I am grateful that she prepared pancakes for me every morning before I went to the lab at 4.30am and returned at 9pm. She has been great company on a number of conference trips all around the globe. In particular, we had a great time together exploring Texas. I am looking forward to spending more time with her after my successful defense.

Contents

List of Publications during PhD Thesis Work

The work done in paper [61] has been featured in the **New Scientist** article “AI could put a stop to electricity theft and meter misreadings"[1]. Paper [60] has been cited in the **McKinsey Global Institute** publication “Artificial intelligence: The next digital frontier?"[2]. The research in paper [72] was presented as both, a talk as well as a poster, at the same conference and subsequently got a **best poster award**.

[59] P. Glauner, “Künstliche Intelligenz - die nächste Revolution (The Artificial Intelligence Revolution)”, in *Innovationsumgebungen gestalten: Impulse für Start-ups und etablierte Unternehmen im globalen Wettbewerb*, P. Plugmann, Ed., Springer, 2018.

[60] P. Glauner, A. Boechat, L. Dolberg, R. State, F. Bettinger, Y. Rangoni and D. Duarte, “Large-scale detection of non-technical losses in imbalanced data sets”, in *Proceedings of the Seventh IEEE Conference on Innovative Smart Grid Technologies (ISGT 2016)*, IEEE, 2016.

[61] P. Glauner, N. Dahringer, O. Puhachov, J. Meira, P. Valtchev, R. State and D. Duarte, “Identifying irregular power usage by turning predictions into holographic spatial visualizations”, in *Proceedings of the 17th IEEE International Conference on Data Mining Workshops (ICDMW 2017)*, IEEE, 2017, pp. 258–265.

[62] P. Glauner, M. Du, V. Paraschiv, A. Boytsov, I. Lopez Andrade, J. Meira, P. Valtchev and R. State, “The top 10 topics in machine learning revisited: A quantitative meta-study”, in *Proceedings of the 25th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2017)*, 2017.

[1] `http://www.newscientist.com/article/2148308-ai-could-put-a-stop-to-electricity-theft-and-meter-misreadings/`

[2] `http://www.mckinsey.com/business-functions/mckinsey-analytics/our-insights/how-artificial-intelligence-can-deliver-real-value-to-companies`

[63] P. Glauner, J. Meira, L. Dolberg, R. State, F. Bettinger, Y. Rangoni and D. Duarte, "Neighborhood features help detecting non-technical losses in big data sets", in *Proceedings of the 3rd IEEE/ACM International Conference on Big Data Computing Applications and Technologies (BDCAT 2016)*, 2016.

[67] P. Glauner, J. Meira, P. Valtchev, R. State and F. Bettinger, "The challenge of non-technical loss detection using artificial intelligence: A survey", *International Journal of Computational Intelligence Systems*, vol. 10, no. 1, pp. 760–775, 2017.

[68] P. Glauner, A. Migliosi, J. Meira, P. Valtchev, R. State and F. Bettinger, "Is big data sufficient for a reliable detection of non-technical losses?", in *Proceedings of the 19th International Conference on Intelligent System Applications to Power Systems (ISAP 2017)*, IEEE, 2017.

[72] P. Glauner, R. State, P. Valtchev and D. Duarte, "On the reduction of biases in big data sets for the detection of irregular power usage", in *Proceedings 13th International FLINS Conference on Data Science and Knowledge Engineering for Sensing Decision Support (FLINS 2018)*, 2018.

[73] P. Glauner, P. Valtchev and R. State, "Impact of biases in big data", in *Proceedings of the 26th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2018)*, 2018.

Further publications not incorporated in this thesis:

[56] M. Galetzka and P. Glauner, "A simple and correct even-odd algorithm for the point-in-polygon problem for complex polygons", in *Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2017), Volume 1: GRAPP*, 2017.

[58] P. Glauner, "Deep learning for smile recognition", in *Uncertainty Modelling in Knowledge Engineering and Decision Making: Proceedings of the 12th International FLINS Conference*, World Scientific, 2016, pp. 319–324.

[112] J. Meira, P. Glauner, R. State, P. Valtchev, L. Dolberg, F. Bettinger and D. Duarte, "Distilling provider-independent data for general detection of non-technical losses", in *Power and Energy Conference at Illinois (PECI)*, IEEE, 2017.

List of Tutorials during PhD Thesis Work

Tutorials [64]–[66], [70], [71] have allowed to present the research carried out in the framework of this thesis in depth to the power engineering community. An **invitation by the Governing Board of the IEEE Power & Energy Society** has initiated tutorial [71].

[64] P. Glauner, J. Meira and R. State, “Introduction to detection of non-technical losses using data analytics”, in *7th IEEE Conference on Innovative Smart Grid Technologies, Europe (ISGT Europe 2017)*, 2017.

[65] P. Glauner, J. Meira and R. State, “Detection of irregular power usage using machine learning”, in *IEEE Conference on Innovative Smart Grid Technologies, Asia (ISGT Asia 2018)*, 2018.

[66] P. Glauner, J. Meira and R. State, “Machine learning for data-driven smart grid applications”, in *IEEE Conference on Innovative Smart Grid Technologies, Asia (ISGT Asia 2018)*, 2018.

[69] P. Glauner and R. State, “Deep learning on big data sets in the cloud with apache spark and google tensorflow”, in *9th IEEE/ACM International Conference on Utility and Cloud Computing (UCC 2016)*, 2016.

[70] P. Glauner and R. State, “Load forecasting with artificial intelligence on big data”, in *Sixth IEEE Conference on Innovative Smart Grid Technologies, Europe (ISGT Europe 2016)*, 2016.

[71] P. Glauner and R. State, “Introduction to machine learning for power engineers”, in *10th IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC 2018)*, 2018.

List of Figures

List of Tables

List of Algorithms

1

Introduction

Modern societies are heavily dependent on electrical energy. This dependency increases with the diminishing reserves of fossil fuels and the resulting changes to energy consumption, such as a shift towards electric mobility as argued in [48]. For emerging markets, energy consumption steadily increases due to increased economic wealth of market participants. However, in those countries, large parts of the generated and transmitted power are not accounted for and therefore bring no contribution to the profit margin of the electric utility producing and distributing the power. In general, these transmission and delivery (T&D) losses can be divided into technical losses and non-technical losses.

Technical losses occur mostly due to power dissipation. They are naturally caused by internal electrical resistance in generators, transformers and transmission lines as well as system use. In general, technical losses are around 1-2% of the total electricity distributed in efficient systems, such as in Western Europe. In less efficient systems, they can range up to 9-12% as reported in [74]. In contrast, [158] estimates technical losses to be in the range of 2-6%.

In turn, non-technical losses (NTL) are caused by external entities. These entities can be separated into customers that have an electrical energy supply contract and irregular users. NTL consist primarily of electricity theft, faulty or broken infrastructure as well as errors in meter reading, accounting and record-keeping as defined in [8].

Electricity theft can be subdivided into:

- Fraud, e.g. by tampering with meters,
- bypassing metering equipment, e.g. by rigging wires, and
- arranged invoicing irregularities.

Examples of electricity theft are depicted in Figures 1.1 and 1.2.

Figure 1.1: Example of meter manipulation[a].

[a] Source: `http://www.bbc.com/news/uk-england-35810183`

In total, the annual world-wide financial losses due to NTL are estimated to be around 100 billion USD as reported in [38], [158]. Unlike technical losses these represent the major share of the overall losses in emerging markets, which are the subject of interest in our research. An example of what the consumption profile of a customer committing electricity theft may look like is depicted in Figure 1.3. Even though the pattern of a sudden drop is common among fraudsters, this drop can also have other causes. For example, tenants can move out of a house or a factory can scale down its production.

1.1 Non-Technical Losses

The first instances of electricity theft go back to the late 19th century when utilities started to deploy electricity to the masses. An example of electricity theft from 1886 in New York City is reported in [133]. The corresponding newspaper article is depicted in Figure 1.4. However, in some jurisdictions, electricity theft was not considered as being a crime. For example, the Imperial Court of Justice of Germany ruled twice in 1896 and 1899 that electricity theft was not included in the criminal law. The reason for that was that the Court did not consider electricity to be an object as such and could thus not actually be

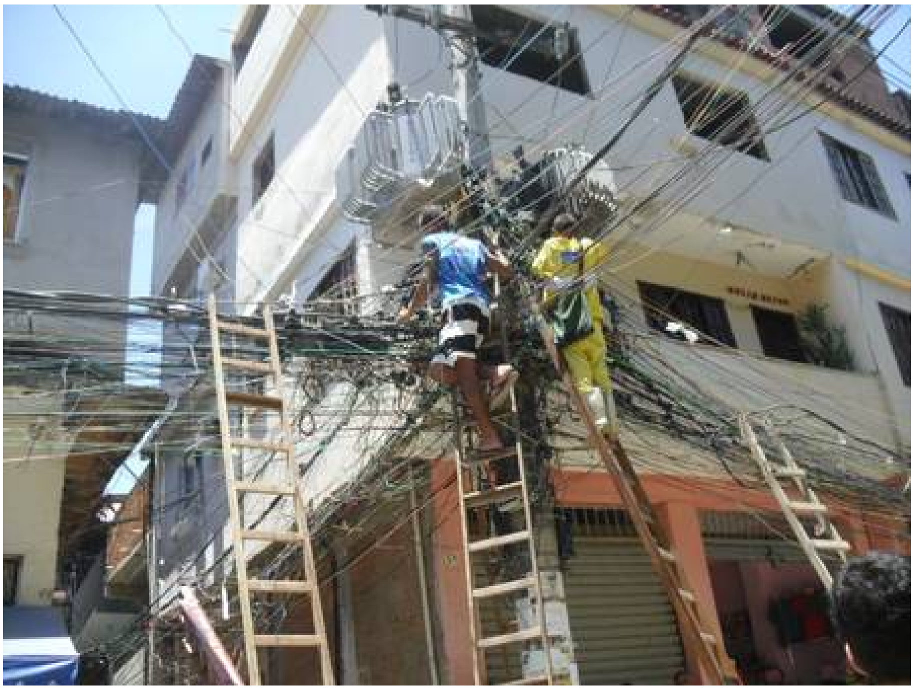


Figure 1.2: Example of power distribution infrastructure manipulation[a].

[a] Source: http://extra.globo.com/casos-de-policia/fornecedoras-de-energia-do-rio-tem-altos-gastos-com-combate-ao-gato-13321228.html

stolen. Both rulings are discussed in [94]. The German Parliament subsequently introduced a law in 1900 making electricity theft punishable. In contrast, the Court of Cassation of France ruled early that electricity theft was already covered by the existing criminal law, as reviewed in [57].

A number of papers in the past have dealt with NTL detection and presented solutions for minimizing NTL. However, they focus on technology, most importantly artificial intelligence methods, and only take a quick glance at the economic side of NTL. An extensive overview of technical approaches for NTL reduction can be found in Chapter 3 and in [113], [170]. In the past decade, the technological means for detecting NTL have improved. However, a significant part of NTL persists as argued in [158]. We therefore take an additional approach as we research the underlying causes of NTL. Based on them, we present further approaches and guidelines to minimizing NTL in Chapter 7. Our recommendations can be used for complementing technical measures of NTL reduction.

Our hypothesis is that a combined approach of the following is necessary for reducing NTL:

- Technical measures against NTL,
- economic analysis of NTL,
- economic measures based on these findings and

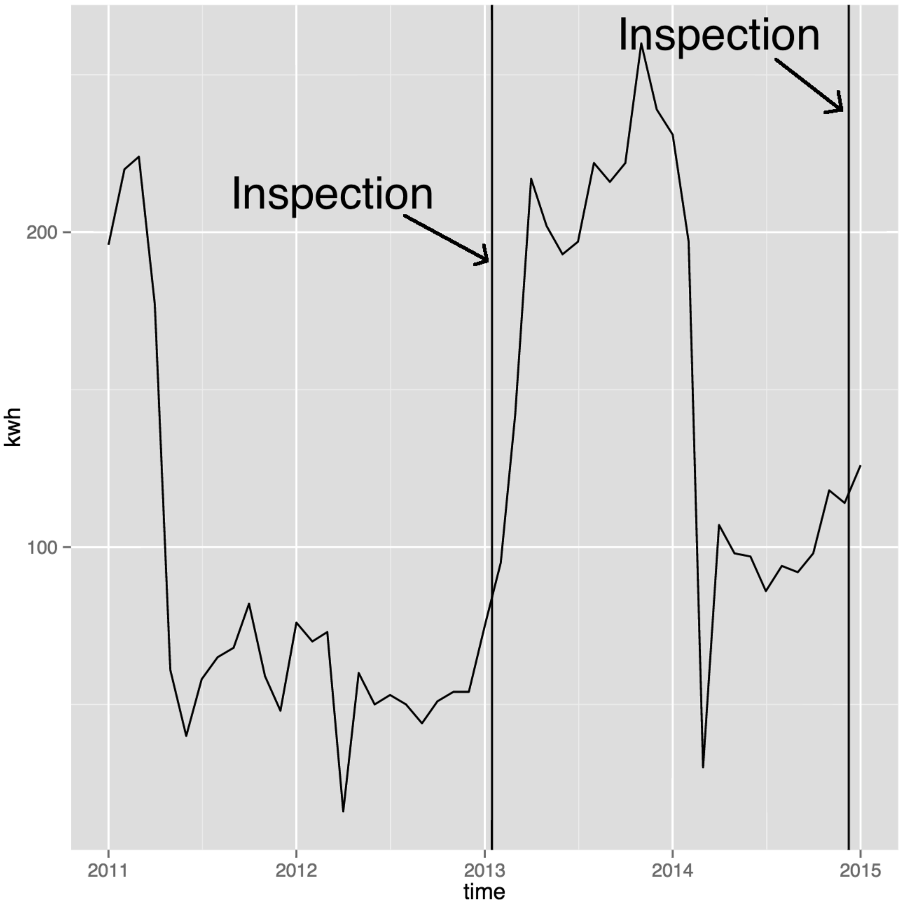
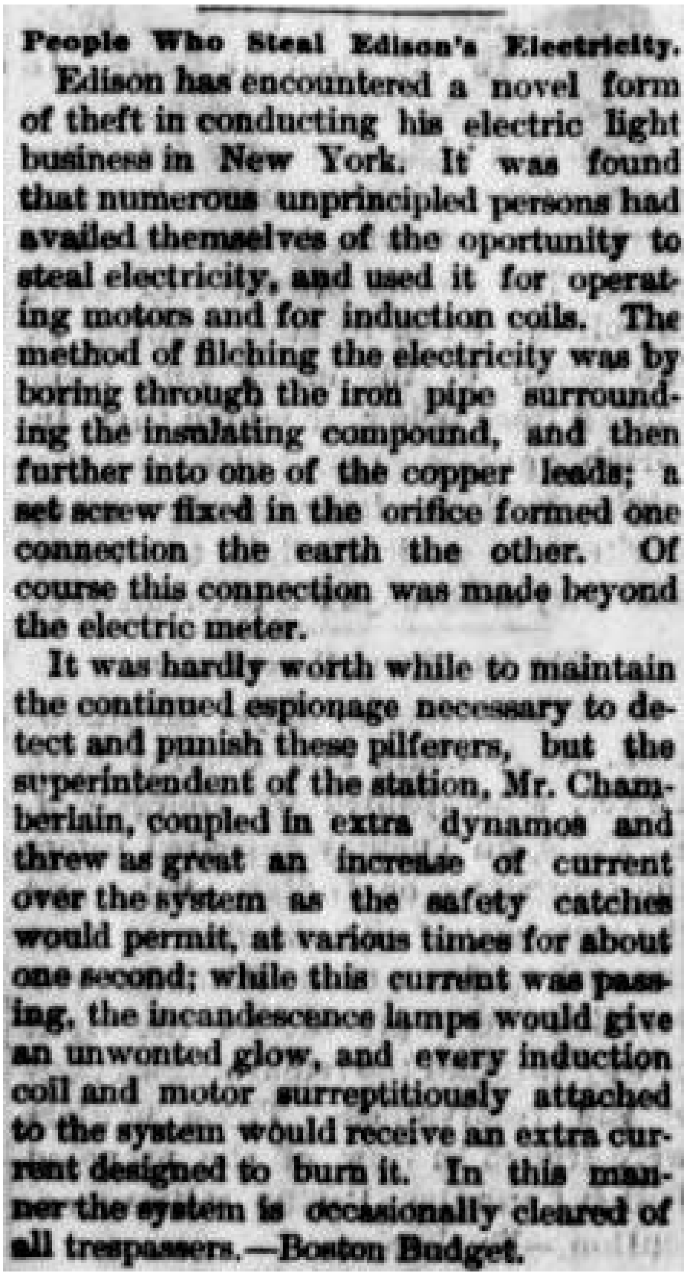


Figure 1.3: Typical example of electricity theft: The consumption time series of the customer undergoes a sudden drop in the beginning of 2011 because the customer's meter was manipulated to record less consumption. This drop then persists over time. Based on this pattern, an inspection was carried out in the beginning of 2013, which detected an instance of electricity theft. This manipulation of the infrastructure was reverted and the electricity consumption resumed to the previous level. One year later, the electricity consumption dropped again to about a third, which led to another inspection a few months later.

- legal and political frameworks that are supportive and consistent.

This combination then yields in an overall higher reduction of NTL compared to focusing solely on technical measures.

The rest of this section is organized as follows. In Chapter 1.1.1, we provide an in-depth review of the causes of NTL. Next, we discuss the global economic effects of NTL in Chapter 1.1.2. These effects vary from country to country depending on the respective proportion of NTL. We discuss this topic further in Chapter 1.1.3.



People Who Steal Edison's Electricity.

Edison has encountered a novel form of theft in conducting his electric light business in New York. It was found that numerous unprincipled persons had availed themselves of the oportunity to steal electricity, and used it for operating motors and for induction coils. The method of filching the electricity was by boring through the iron pipe surrounding the insulating compound, and then further into one of the copper leads; a set screw fixed in the orifice formed one connection the earth the other. Of course this connection was made beyond the electric meter.

It was hardly worth while to maintain the continued espionage necessary to detect and punish these pilferers, but the superintendent of the station, Mr. Chamberlain, coupled in extra dynamos and threw as great an increase of current over the system as the safety catches would permit, at various times for about one second; while this current was passing, the incandescence lamps would give an unwonted glow, and every induction coil and motor surreptitiously attached to the system would receive an extra current designed to burn it. In this manner the system is occasionally cleared of all trespassers.—Boston Budget.

Figure 1.4: Daily Yellowstone newspaper article from March 27, 1886 reporting early instances of electricity theft, see [133].

1.1.1 Causes

NTL are caused both by customers who have a contract and by irregular users without a contract. Customers not only cause NTL for example through fraudulent behavior or faulty or broken infrastructure, but also by having unmetered access. However, unmetered access is not necessarily fraudulent as some customers are allowed to have unmetered access for political or historical reasons. One example of the latter category is discussed in [74] in which Indian farmers do not have meters, but pay a flat rate price in proportion to the consumption required of their farming equipment. However, we would like to add that the opposite may be true, too, though: in Brazil, some customers can buy energy directly from a power plant. Their energy still passes through some distribution company network. These customers are metered but not charged based on the meter value.

There are a number of other reasons why customers of electrical energy may cause NTL. For example, [100], [178] report that some customers have insufficient income to cover the costs, which is a main cause for electricity theft. Other customers may be generally able to pay the energy at market price but are unwilling to do so. This can happen because the customers assume that their non-compliance has no effects due a corrupt local structure as argued in [14]. These assumptions appear to be reasonable, since [74] shows for the Indian state of Uttar Pradesh that a quarter of the members of the state assembly are either under criminal indictment or have been convicted on criminal charges.

Divergent Requirements of Utilities and Customers

In other cases where a customer wants a regular connection, the electric utility may be unable or unwilling to provide access for various reasons. [90] provides the case of Indian farmers who were legally eligible for a substantially subsidized connection. As providing power to them would have decreased the profit of the electric utility, it deferred the connection for up to 15 years. That delay tactic forced the farmers to establish irregular connections. There may also be other reasons such as inefficiency when utilities are not able to respond to demands of customers.

While the previous cases concern potential users that are able to sign a contract with the energy supplier, the others do not fulfil the prerequisites. This could be caused by a missing bank account which may be required as a payment option and a mean to verify the identity of a customer. Also, potential customers may not have the documents required for a contract. Furthermore, a lack of a regular address due to irregular housing could also prevent a customer from getting a regular contract. Concerning irregular housing, areas such as favelas are considered unmanageable due to the high crime rate. While they are usually served by electric utilities, [8] argues that they may not be inspected by utilities at all and thus make electricity theft more likely.

Unpaid Bills

[158] notes that some definitions of NTL also include unpaid bills, a special case when electrical energy is consumed and billed, but the consumption does not increase the earnings of the billing electric utility. Furthermore, unpaid bills impact the financial results of utilities in more than one way. First, they directly reduce the revenue. Second, in some markets, taxes regarding that bill must be paid in advance when it is generated. If the bill is not paid, a complex tax reimbursement process must be followed. Third, a high amount

of unpaid bills may require the electric utility to increase its financial provisions. This may reduce the perceived value of a company listed in a stock market.

Contrary to other cases of theft, unpaid bills can be directly accounted to a specific customer and the amount of billed and unpaid energy can be accurately assessed through accounting. We therefore do not further include those losses in our discussion of NTL as it is mostly a management and accounting issue.

1.1.2 Economic Effects

NTL have direct and indirect economic effects on the price of consumed electrical energy. The direct effects of NTL are loss of revenue and profit for electric utilities. This can lead to a price surcharge for regular customers. Indirect effects result from various causes: electric utilities must regularly check for NTL by carrying out on-site inspections. An inspection is expensive as it requires physical presence of usually two technicians. NTL can also overload generation units, which results in an unstable and unreliable infrastructure. [43], [74] report a number of physical impacts, such as brownouts and blackouts. The lost revenue due to brownouts and blackouts is called value of lost load. [9] shows that these reliability issues have a negative impact on business efficiency and compliance with regulatory targets. Furthermore, NTL affect the overall economy as an unreliable network prevents activities that rely on a stable network. In a similar vein, an unreliable network requires entities to invest in safeguards such as batteries or generators as reasoned in [5].

[101] classifies the effects of electricity interruptions as a whole into direct economic effects, indirect economic effects and social effects. Direct economic effects result from the loss of production, the costs to restart production, equipment costs, repair costs, increased wear and tear, rental equipment and additional use of materials. [162] reports mitigation efforts to compensate for insufficient power quality, for example by using generators or batteries. However, it has been shown in [101] that the extra cost of these actions may be higher than the return on investment. The relationship between the effort spent on NTL detection and the return on investment is sketched in Figure 1.5.

1.1.3 Variation

NTL have a high variation depending on the development status of a country. Typical indicators for the development status are the human development index (HDI) and the gross domestic product (GDP) per capita. These metrics are regularly published by the United Nations, such as in [86].

In developed and economically wealthy countries, such as the United States or Western Europe, NTL are less of an issue. Reasons for this include that the population can afford to pay for electricity as well as the high quality of grid infrastructure as argued in [8]. In developing countries - or emerging markets - such as Brazil, India or Pakistan the situation

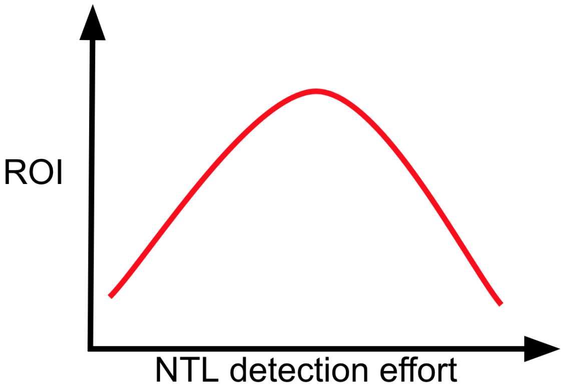


Figure 1.5: Relationship between the effort spent on NTL detection and the return on investment (ROI): Different resources can be spent on the detection of NTL, such as sending technicians for physical on-site inspections, investments in better meters that are less prone to tampering, etc. These actions then reduce NTL and thus lead to higher revenue and profit for the utilities. If an insufficient amount of resources is spent, the ROI is low. In contrast, if too many resources are spent, the ROI is low, too. The optimal amount of resources spent on NTL detection leading to a maximum ROI is in this range.

is different. While some emerging markets have low NTL, other countries have so far not succeeded in significantly reducing NTL. The total annual financial losses in India due to NTL are estimated in [38] to be around USD 16.2 billion and the NTL proportion may range up to 70% of the total electricity distributed. For instance, the overall losses in the Indian state of Uttar Pradesh are estimated to be in the range of 13.6-49.9%. These compose of technical losses of 12% due to an inefficient power grid and of NTL between 1.6-37.9% in [74]. We provide a summary of NTL estimates for selected countries in Table 1.1.

Table 1.1: NTL proportions in different countries.

Country	NTL Proportion	Reference
Brazil	3-40%	[142]
India	Up to 70%	[90]
Rwanda	18%	[119]
Turkey	4-73%	[178]
Uttar Pradesh, India	1.6%-37.9%	[74]

On the level of a single country, NTL can vary significantly as reported in [178]. A study for Turkey shows that the regional distribution varies from 4% in the North West - the European part of Turkey as well as the surrounding areas. The losses increase towards the South East, with the Anatolia region having losses around 73%. However, it must be noted that the total electricity distributed in the North West significantly exceeds the one distributed in the South East. Therefore, the absolute losses in North West should not be underestimated even though the NTL percentage is far less than in the South East. The study in [178] also shows that there is a correlation with the annual income, which is between USD 3,515 in the South East regions and reaches up to USD 13,865 in the North West. The price for electrical energy are the same, though. As a consequence, an annual spending on electricity represents a higher fraction of the annual income in the less developed area. Furthermore, that study shows a correlation between NTL and education, population rate, temperature and agricultural production rate.

1.2 Research Framework

The research project that led to this thesis has been carried out within an innovation ecosystem.

What is an innovation ecosystem? An idea is a problem-solving approach that aims to improve the state of the art. A large part of all innovations in the field of artificial intelligence originally started in academia. Most of this research is funded by third parties, which therefore requires active collaboration with research funding agencies and industrial partners. In order for new research findings to become a reality, and not just to be published in journals or conferences, these results must be exposed early to interaction with industry. In industry, however, there are predominantly practitioners and less scientists. Modern university teaching must thus ensure that today's computer science graduates are prepared for the challenges of tomorrow. Interaction between academia and industry is possible both with existing companies and through spin-offs. A close integration with funding sources such as research funding agencies or venture capital is indispensable for the rapid and competitive transformation of research results into value-adding products. A good innovation ecosystem therefore consists of a functioning and dynamic combination of research, teaching, industry, research funding and venture capital, as depicted in Figure 1.6.

Now we present our research project between the Interdisciplinary Center for Security, Reliability and Trust (SnT)[1], University of Luxembourg and the industrial partner CHOICE Technologies[2]. CHOICE Technologies has been operating in the Latin American market for more than 20 years with the goal of reducing NTL and electricity theft by using AI. In order to remain competitive in the market, the company has chosen to incorporate state-of-the-art AI technology into its products. Today, however, much of the innovation in the field of AI starts at universities. For this reason, the company has decided to work

[1] http://snt.uni.lu

[2] http://www.choiceholding.com

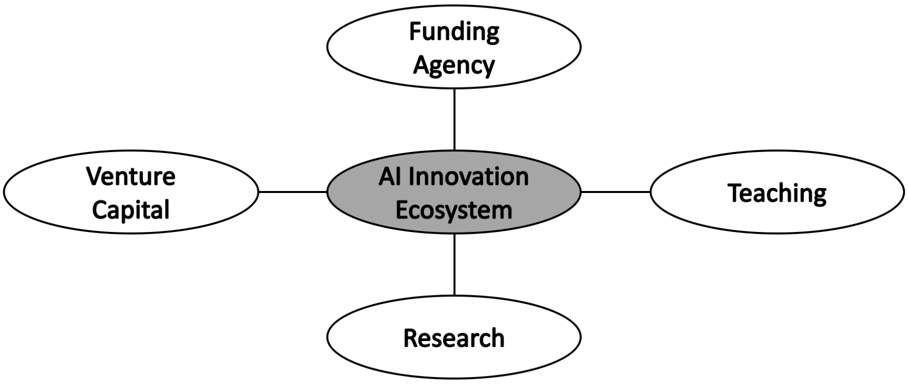


Figure 1.6: Composition of an AI innovation ecosystem.

with SnT, which specializes in conducting hands-on research projects with industrial partners. The aim of these projects is not only to publish research results, but also to develop concrete outcomes that can be used by the industrial partners. The third stakeholder is the Luxembourg National Research Fund (FNR)[3], a research funding agency that contributes to the funding of this research project through a public-private partnership grant under agreement number AFR-PPP 11508593. The activities of this innovation ecosystem are shown in Figure 1.7, which we explain below.

At the beginning of a project iteration, the university staff and the company's employees agree on the requirements to be met. Next, the staff of the university prepare an extensive literature review, which describes in detail the state of the art of research. Based on the literature review and the company's requirements, project goals are agreed on to deliver both new research results and concrete results that the company can exploit. Afterwards, the staff of the university carry out the research tasks and receive data from the company, which consists among other things of electricity consumption measurements and the results of physical on-site inspections. Throughout a project iteration, both sides regularly consult with each other and adjust the requirements as needed. After completing the research, the university staff present the research results to the company, including a software prototype. The use of the results is now divided into two different directions: First, the results are published by the university staff in suitable journals or presented at conferences. The publications also refer to the support of the research funding organization, which can also use these publications for marketing its research funding. In addition, the university staff are able to integrate their new research findings into their courses, preparing the next generation of researchers and developers for future challenges with state-of-the-art lecture content. Second, the company integrates the relevant and usable research results into its products. As a result, it can use the latest research results to not only to maintain its competitiveness, but also to expand their business. After that, the next project iteration begins, in which new requirements are identified. Ideally, these also contain feedback from

[3] `http://www.fnr.lu`

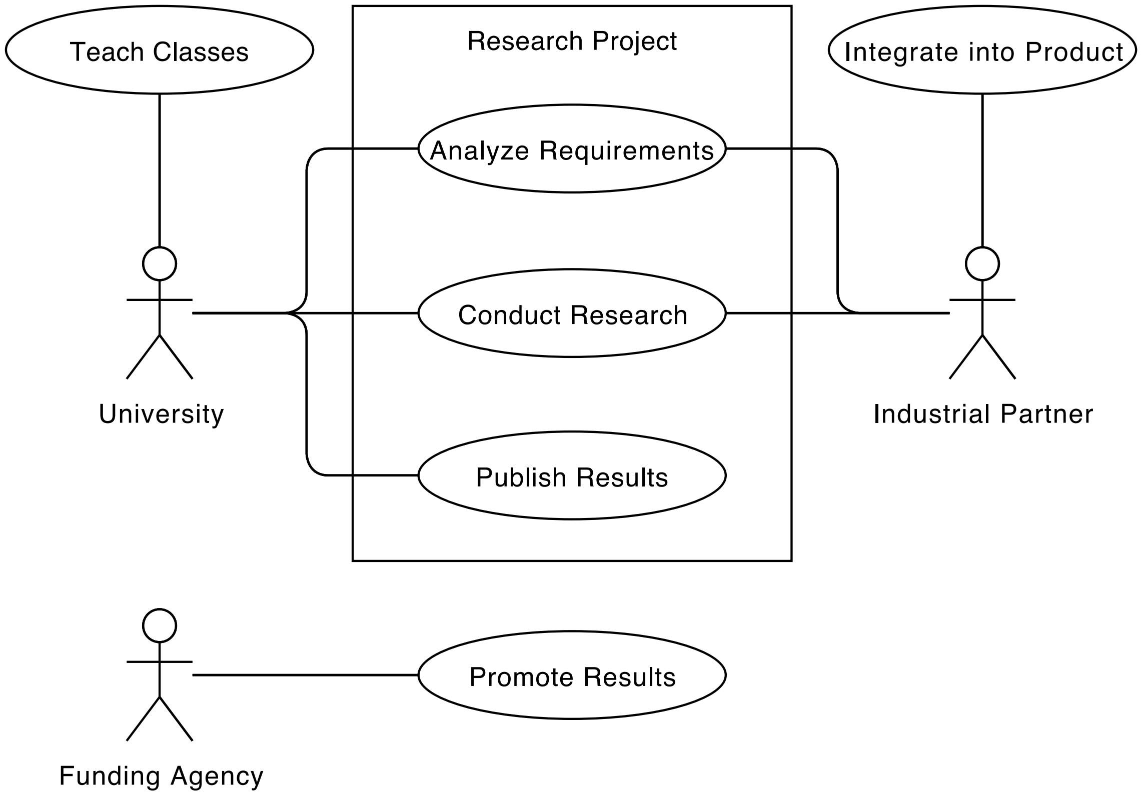


Figure 1.7: Activities of the project in which this thesis research was carried out.

customers that use the new product functions resulting from the research results.

1.3 Research Question and Contributions

The main research question of this thesis is:

How can we detect non-technical losses better in the real world?

In Chapter 2, we provide a brief overview about the field of artificial intelligence, its history, recent advances and most relevant questions for its future. We do so in order to provide a larger view on the research field of this thesis. We therefore also lay the groundwork for making this thesis accessible to a larger audience. Next, we provide an overview about modern machine learning methods and then we provide an introduction to the models most relevant to this thesis.

Below we describe in more detail the individual research questions and corresponding contributions of this thesis:

1. What are the Open Challenges of NTL Detection?

The predominant research direction is employing artificial intelligence to predict whether a customer causes NTL. NTL detection methods reported in the literature fall into two categories: expert systems and machine learning. Expert systems incorporate hand-crafted rules for decision making. In contrast, machine learning gives computers the ability to learn from examples without being explicitly programmed. Historically, NTL detection systems were based on domain-specific rules. However, over the years, the field of machine learning has become the predominant research direction of NTL detection. In Chapter 3, we first survey the state-of-the-art research efforts in a up-to-date and comprehensive review of algorithms, features and data sets used. We also compare the various approaches reported in the literature. Next, we identify the key scientific and engineering challenges of NTL detection that have not yet been addressed in scientific works. The challenges identified lay the groundwork for the following detailed research questions. We also put these challenges in the context of AI research as a whole as they are of relevance to many other real-world learning and anomaly detection problems.

2. How can we Compare Industrial NTL Detection Systems based on Expert Knowledge to those based on Machine Learning?

Industrial NTL detection systems are still largely based on expert knowledge when deciding whether to carry out costly on-site inspections of customers. In Chapter 4, we use an industrial NTL detection system based on Boolean logic. We improve it by fuzzifying the rules and compare both to a NTL detection system based on machine learning. We show that the one based on machine learning significantly outperforms the others based on expert knowledge.

3. How can we Combine Industrial Expert Knowledge with Machine Learning for the Decision Making?

Despite the superiority of machine learning-based approaches over expert knowledge for NTL detection, electric utilities are reluctant to move to large-scale deployments of automated systems that learn NTL profiles from data due to the latter's propensity to suggest a large number of unnecessary inspections. In order to allow human experts to feed their knowledge in the decision process, we propose in Chapter 4 a method for visualizing prediction results of a machine learning-based system at various granularity levels in a spatial hologram. Our approach allows domain experts to put the classification results into the context of the data and to incorporate their knowledge for making the final decisions of which customers to inspect.

4. How can we Comprehensively Learn from the Customer Data how to Find Customers with Irregular Behavior?

In Chapter 5, we take full advantage of the customer data in order to detect NTL better. For doing so, we explore two different directions: We first derive features that include information about the neighborhood. We show that the neighborhood of customers contains information about whether a customer may cause NTL or not. We analyze the statistical properties of these features and show why they are useful for NTL detection. By using information of the neighborhood, we can predict NTL better as there are geographic clusters of NTL among the customers. Next, we propose a novel and flexible framework to compute a large number of domain-specific features and generic features from the noisy industrial consumption time series of customers for NTL detection. We retain the statistically meaningful features extracted from the noisy consumption data and optimize different classifiers to predict NTL.

5. How can we Handle the Biases in the Inspection Data?

In machine learning, a bias occurs whenever training sets are not representative of the test data, which results in unreliable models. The most common biases in data are arguably class imbalance and covariate shift. In Chapter 6, we aim to shed light on this topic in order to increase the overall attention to this issue in the field of machine learning. We first provide an intensive introduction to the problem of biased data sets. Next, we propose an algorithm for quantifying covariate shift and show that the location and class of customers have the strongest covariate shift in NTL detection. We then propose a scalable novel framework for reducing multiple biases in high-dimensional data sets in order to train more reliable predictors. We apply our methodology to the detection of NTL and show that reducing these biases increases the accuracy of the trained predictors.

6. What are the Limitations of AI-based NTL Detection Systems?

Despite the advances made in this thesis, just using artificial methods is not enough in order to reliably and sustainably reduce NTL. In addition, legal actions need to be taken, market reforms need to be made and utilities need to make investments into enhanced infrastructure components and streamlined methods of payment. We will discuss these suggestions in Chapter 7.

1.4 References to Publications

This thesis interpolates material from 9 first-author publications by the author [59]–[63], [67], [68], [72], [73]. This introductory chapter uses some material from [59]. Next, Chapter 2 is based on [59], [62]. Chapter 3 is based on [67]. Meanwhile, Chapter 4 uses material from [60], [61]. Chapter 5 combines [61], [63]. Last, Chapter 6 is based on

[68], [72], [73].

2

Artificial Intelligence

There is not a day that passes on which we do not hear news about artificial intelligence (AI): autonomous cars, spam filters, Siri, chess computers, killer robots and much more. What exactly is AI? Peter Norvig, Research Director of Google, describes AI in one sentence:

> *"AI is the science of knowing what to do when you don't know what to do."* [118]

Admittedly, at first glance, this description is somewhat confusing, but secondarily reasonable: the goal of AI is to solve complex computer problems that are often associated with uncertainty.

This chapter first provides a brief review of the history of artificial intelligence, its recent advances and most relevant questions for its future. We do so in order to provide a larger view on the research field of this thesis. We therefore also lay the groundwork for making this thesis accessible to a larger audience. Next, we provide an overview about modern machine learning methods. Last, we provide an introduction to the models most relevant to this thesis.

2.1 History

The first theoretical foundations of AI were laid in the mid-20th century, especially in the works of British mathematician Alan Turing [166]. The actual year of birth of AI is the year 1956, in which the six-week conference Summer Research Project on Artificial Intelligence at Dartmouth College took place. For this purpose, an application for funding was made in the previous year. The research questions contained therein proved to be indicative of many of the long-term research goals of AI [110]. The conference was organized by John McCarthy and was attended by other well-known scientists such as Marvin Minsky, Nathan Rochester and Claude Shannon.

Over the following decades, much of AI research has been divided into two diametrically different areas: expert systems and machine learning. **Expert systems** comprise rule-based descriptions of knowledge and make predictions or decisions based on input/data. In contrast, **machine learning** is based on recognizing patterns from training data. This principle is further outlined in Definitions 2.1 and 2.2.

Definition 2.1. “[Machine learning is the] field of study that gives computers the ability to learn without being explicitly programmed." [153]

Definition 2.2. “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E." [115]

Concretely, a machine learning algorithm learns patterns from examples as depicted in Figure 2.1. These patterns are then used to make decisions based on inputs.

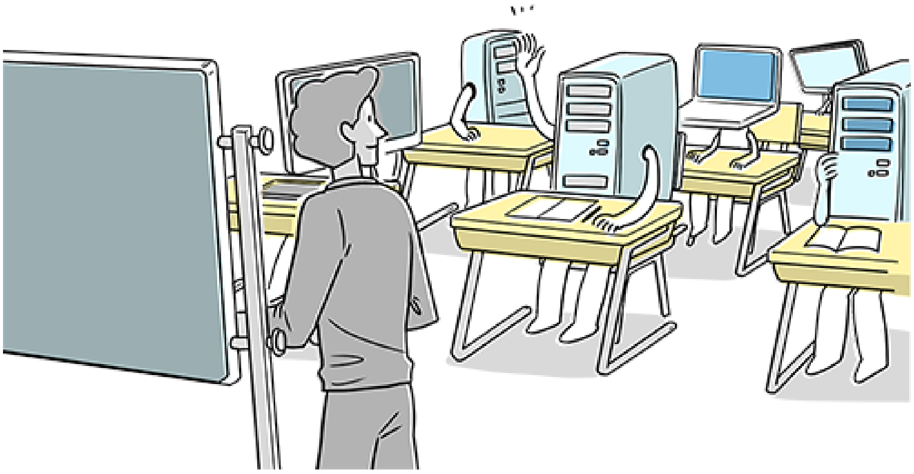


Figure 2.1: Machine learning: learning from examples[a].

[a] Source: `http://blog.capterra.com/what-is-machine-learning/`

Both, expert systems and machine learning, have their respective advantages and disadvantages: Expert systems on one hand have the advantage that they are understandable and interpretable and that their decisions are therefore comprehensible. On the other hand, it often takes a great deal of effort, or sometimes it even turns out to be impossible to understand and describe complex problems in detail.

To illustrate this difficulty, an example of machine translation, the automatic translation from one language to another, is very helpful: First, languages consist of a complex set of words and grammar that are difficult to describe in a mathematical form. Second, one does not necessarily use languages correctly, which can cause inaccuracies and ambiguities. Third, languages are dynamic as they change over decades and centuries. Creating an expert system for machine translation is thus a challenge. The three factors of complexity, inaccuracy and dynamics occur in a variety of fields and prove to be a common limiting factor when building expert systems.

Machine learning has the advantage that often less knowledge about a problem is needed as the algorithms can learn patterns from data. This process is often referred to as “training" an AI. In contrast to expert systems, however, machine learning often leads to a black box whose decisions are often neither explainable nor interpretable. Nonetheless, over the decades, machine learning has gained popularity and largely replaced expert systems.

Of particular historical significance are so-called (artificial) neural networks. These are loosely inspired by the human brain and consist of several layers of units - also called “neurons". An example of a neural network is shown in Figure 2.2.

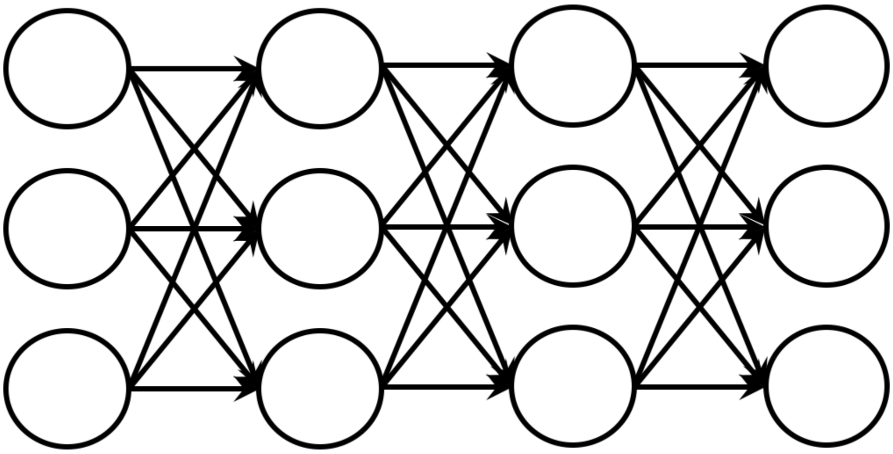


Figure 2.2: Neural network.

The first layer (on the left) is used to enter data and the last layer (on the right) to output predictions or decisions. Between these two layers are zero to several hidden layers, which contribute to the decision-making. Neural networks have experienced several popularity phases over the past 60 years, which are explained in detail in [42]. In addition to neural networks, there are a variety of other methods of machine learning, such as decision trees, support vector machines or regression models, which are discussed in detail in [16].

Over the past decades, a large number of innovative and value-adding applications have emerged, often resulting from AI research results. Autonomously driving cars, speech recognition and autonomous trading systems for example. Nonetheless, there have been many setbacks. These were usually caused by too high and then unfulfilled expectations. In this context, the term of an “AI winter" has been coined, with which periods of major setbacks in recent decades, the loss of optimism and consequent cuts in funding are referred to. Of course, this section can only provide an overview of the history of AI. The interested reader is referred to a detailed discussion in [149].

2.2 Recent Advances

Although AI research has been practiced for over 60 years, many people first heard of AI just a few years ago. This, in addition to the “Terminator" movie series, is largely due to the huge advances made by AI applications over the past few years. Since 2006, there have been a number of significant advances, especially in the field of neural networks, which are now referred to as deep learning [78]. This term aims to ensure that (deep) neural networks have many hidden layers. This type of architecture has proven to be particularly helpful in detecting hidden relationships in input data. Although this was already the case in the 1980s, there was a lack of practical and applicable algorithms for training these networks from data first and, secondly, the lack of adequate computing resources. However, today there is much more powerful computing infrastructure available. In addition, significantly better algorithms for training this type of neural network have been derived since 2006 [78].

As a result, many advances in AI research have been made, some of which are based on deep learning. Examples are autonomously driving cars or the computer program AlphaGo. Go is a board game that is especially popular in Southeast Asia, where players have a much greater number of possible moves than in chess. Traditional methods, with which, for example, the IBM program Deep Blue had beaten the then world chess champion Garry Kasparov in 1997, do not scale to the game Go, since the mere increase of computing capacity is not sufficient due to the high complexity of this problem. It was until a few years ago the prevailing opinion within the AI community that an AI, which plays Go on world level, was still decades away. The UK company Google DeepMind unexpectedly introduced AlphaGo in 2015, which beat South Korean professional Go play Lee Sedol under tournament conditions [157]. This success was partly based on deep learning and led to an increased awareness of AI world-wide. Of course, in addition to the current breakthroughs of AI mentioned in this section, there have been a lot of further success stories and we are sure that more will follow soon.

2.3 Frontiers

We would now like to discuss some current issues concerning AI and provide an outlook on a possible future of AI. While many recent accomplishments are based in part on deep learning, this new kind of neural network is only one of many modern techniques. It is becoming increasingly apparent that there is a hype about deep learning and more and more unrealistic promises are being made about it [97]. It is therefore essential to relate the successes of deep learning and its fundamental limitations. The “no free lunch theorem", which is largely unknown both in industry and academia, states that all methods of machine learning averaged over all possible problems are equally successful [176]. Of course, some methods are better suited to some problems than others, but perform worse on different problems. Deep learning is especially useful for image, audio or video processing problems and when having a lot of training data. By contrast, deep learning, for example, is poorly suited to problems with a small amount of training data.

We have previously introduced the notion of an AI winter - a period of great setbacks, the loss of optimism and consequent reductions in funding. It is to be feared that the current and hype-based promise could trigger a new AI winter. It is therefore essential to better understand deep learning and its potential and not neglect other research methods. A big problem of deep learning - and neural networks in general - is that these are black box models. As a consequence, the decisions made by them are often incomprehensible. Some advances have been made in this area recently, such as local interpretable model-agnostic explanations (LIME) [145] for supervised models. However, there is still great research potential in this direction, as future advances may also likely increase the social acceptance of AI. For example, in the case of autonomously driving cars, the decisions taken by an AI should also be comprehensible for legal as well as software quality reasons.

In addition, the question arises as to how the field of AI will evolve in the long term, whether one day an AI will exceed the intelligence of a human being and thus potentially could make mankind redundant. The point of time when computers become more intelligent than humans is referred to in the literature as the Technological Singularity [155]. There are various predictions as to when - or even if at all - the singularity will occur. They span a wide range, from a period in the next twenty years, to predictions that are realistic about achieving the singularity around the end of the 21st century, to the prediction that the technological singularity may never materialize. Since each of these predictions makes various assumptions, a reliable assessment is difficult to make. Overall, today it is impossible to predict how far away the singularity is. The interested reader is referred to a first-class and extensive analysis on this topic and a discussion of the consequences of the technological singularity in [155].

In recent years, various stakeholders have warned about so-called “killer robots" as a possible unfortunate outcome of AI advances. What about this danger? Andrew Ng, one of the leading scientists in the field of machine learning, has set a much-noticed comparison [175]: Ng's view is that science is still very far away from the potential killer robot threat scenario. In his opinion, the state of the art of AI can be compared with a planned manned trip to Mars, which is currently being prepared by researchers. Ng further states that some researchers are also thinking about how to colonize Mars in the long term, but no researcher has yet tried to explore how to prevent overpopulation on Mars. Ng equates the scenario of overpopulation with the scenario of a killer robot threat. This danger would also be so far into the future that he was not able to work productively to prevent it at the moment, as he first had to do much more fundamental work in AI research. Ng also points to potential job losses as a much more tangible threat to people by AI in the near future.

2.4 The Three Pillars of Machine Learning

The field of machine learning can broadly be separated into three so-called pillars. Supervised learning uses pairs $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), ..., (x^{(m)}, y^{(m)})\}$, where $x^{(i)}$ is the input and $y^{(i)}$ the label, respectively. The goal is to learn a function f: $y^{(i)} = f(x^{(i)})$. This is also called function induction, because rules from examples are derived. Induction is the opposite of deduction, which derives examples from rules. If all possible values in y are a finite set of possible values, f does classification. In contrast, if that set is infinite, we talk about regression. In any case, the labels y give an unambiguous “right answer" for the inputs x.

In many problems, it is difficult to provide such an explicit supervision to a learning problem. Therefore, in reinforcement learning, the algorithm is provided a reward (feedback) function which provides a reward or penalty depending on in which state the learning agent goes. Reinforcement learning is therefore somewhat in between supervised and unsupervised learning, as there is some supervision, but significantly less than in supervised learning.

Unsupervised learning tries to find hidden structures in unlabeled data $\{x^{(1)}, x^{(2)}, ..., x^{(m)}\}$. Common tasks are dimensionality reduction or clustering.

2.5 Most Popular Machine Learning Models in the Literature

In 2007, a paper named “Top 10 algorithms in data mining" identified and presented the top 10 most influential data mining algorithms within the research community [177]. The selection criteria were created by consolidating direct nominations from award winning researchers, the research community opinions and the number of citations in Google Scholar. The top 10 algorithms in that prior work are: C4.5 [138], k-means [104], support vector machine [168], Apriori [3], EM [41], PageRank [129], AdaBoost [83], kNN [4], naive Bayes [146] and CART [19].

In the decade that passed since then, machine learning has expanded, responding to incremental development of computational capabilities and substantial increase of problems in the commercial applications. This section reflects on the top 10 most popular fields of active research in machine learning, as they emerged from the quantitative analysis of leading journals and conferences. This work sees some topics in the broader sense including not only models but also concepts like data sets, features, optimization techniques and evaluation metrics. This wider view on the entire machine learning field is largely ignored in the literature by keeping a strong focus entirely on models [49].

Our core contribution in this section is that we provide a clear view of the active research in machine learning by relying solely on a quantitative methodology without interviewing experts. This attempt aims at reducing bias and looking where the research community puts its focus on. The results of this study allow researchers to put their research into the global context of machine learning. This provides researchers with the opportunity to both conduct research in popular topics and identify topics that have not received sufficient attention in recent research.

2.5.1 Methodology

In this section, we discuss how we determine quantitatively the top 10 topics in machine learning from articles of leading journals and conferences published between January 2007 and June 2016. We selected referenced journals that cover extensively the field of machine learning, neural networks, pattern recognition and data mining both from the theoretical perspective and also with applications to image, video and text processing, inference on networks and graphs, knowledge basis and applications to real data sets.

Data Collection

In the data collection, we focus on the abstracts of publications, as they provide the main results and conclusions of a paper. In contrast, the full text includes details on the research, which also comes with more noise that is not relevant to an overall summary of published work. We have chosen 31 leading journals related to machine learning as summarized in Table 2.1, ranked by their impact factor. For each journal, we have collected as many abstracts as possible of articles published in the time frame of interest. In total, we have collected 39,067 abstracts of those 31 journals, which also include special issues.

Furthermore, we have chosen 7 major international conferences related to machine learning as summarized in Table 2.2, ranked by their average citation count. We have collected as many proceedings as possible of those conferences. In addition, we consider the Journal of Machine Learning Research Workshop and Conference Proceedings series, which includes further conferences, such as International Conference on Artificial Intelligence and Statistics and Asian Conference on Machine Learning among others. We have collected 14,459 abstracts from the proceedings of those conferences in the time frame of interest.

Table 2.1: Source journals.

Name	Impact Factor	#Abstracts
IEEE T. on Sys., Man, and Cybernetics, P. B. (Cyb.)	6.22	1,045
IEEE T. on Pattern Analysis and Machine Intell.	5.781	2,552
IEEE T. on Neural Networks and Learning Systems	4.291	1,518
IEEE T. on Evolutionary Computation	3.654	940
IEEE T. on Medical Imaging	3.39	2,470
Artificial Intelligence	3.371	668
ACM Computing Surveys	3.37	395
Pattern Recognition	3.096	3,016
Knowledge-Based Systems	2.947	1,905
Neural Networks	2.708	1,330
IEEE T. on Neural Networks	2.633	758
IEEE Computational Intelligence Magazine	2.571	574
IEEE T. on Audio, Speech and Language Processing	2.475	1,829
Journal of Machine Learning Research	2.473	986
IEEE Intelligent Systems	2.34	1,049
Neurocomputing	2.083	6,165
IEEE T. on Knowledge and Data Engineering	2.067	2,121
Springer Machine Learning	1.889	571
Computer Speech and Language	1.753	452
Pattern Recognition Letters	1.551	2,380
Computational Statistics & Data Analysis	1.4	3,063
Journal of the ACM	1.39	353
Information Processing & Management	1.265	730
ACM T. on Intelligent Systems and Technology	1.25	396
Data & Knowledge Engineering	1.115	660
ACM T. on Information Systems	1.02	229
ACM T. on Knowledge Discovery from Data	0.93	245
ACM T. on Autonomous and Adaptive Systems	0.92	231
ACM T. on Interactive Intelligent Systems	0.8	117
ACM T. on Applied Perception	0.65	234
ACM T. on Economics and Computation	0.54	85
Total (N=31)	-	39,067

Combining the journals and conference proceedings, we have collected 53,526 abstracts in total.

Table 2.2: Source conferences.

Name	#Avg. Citations	#Abstracts	Years
Inter. Conference on Computer Vision	11.9754	2,092	2007, 2009, 2011, 2013, 2015
Inter. Conference on Machine Learning	9.1862	1,185	2013-2016
Advs. in Neural Inf. Processing Syst.	8.5437	2,416	2007-2015
Conf. on Knowledge Discovery and Data M.	7.7269	1,035	2007-2015
Conf. on Comp. Vision and Pattern Recog.	6.6133	4,471	2007-2015
Conference on Learning Theory	4.2905	347	2011-2016
Inter. Conference on Data Mining	2.137	1,406	2007-2015
J. of Machine Learning Research Conf. Proc.	2.473[a]	1,507	2007-2016
Total (N=8)	-	14,459	-

[a] Computing the average citation count of this mixture of various conferences and workshops has proven to not be feasible. Instead, we use the impact factor of the Journal of Machine Learning Research as the average citation count. We expect the impact of the approximation error to be low since it only concerns 1,507 of the total 53,526 abstracts used in this research.

Key Phrase Extraction

We focus on extracting the most relevant key phrases of each abstract, which we call *topics* in the remainder of this study. First, we apply Porter stemming to an abstract [135]. In stemming, only the stem of a word is retained. For example, “paper" and “papers" have the same stem, which is “paper". For the extraction of key phrases from each abstract, we compare two different methods:

1. We remove the stop words from each abstract. Stop words are the words most frequently used in a language that usually provide very little information. For example, “and" or "of" are typical stop words in the English language. After the stop word removal, we then use all bigrams and trigrams as key phrases.

2. The Rapid Automatic Keyword Extraction Algorithm (RAKE) is an unsupervised, domain-independent and language-independent learning algorithm that generates key

phrases from text [148]. First, RAKE splits each abstract into parts that are separated by signs - such as commas and full stops - and stop words. These parts are then split into n-gram key phrases. In our work, we use $1 \leq n \leq 4$. Next, a word co-occurrence graph is learned from the generated n-grams. Last, each key phrase is ranked by the sum of the ratio of degree to frequency per word.

When merging the key phrases of different journals or conferences, we weight each key phrase by the impact factor or average citation count, respectively. The list of key phrases is then sorted in descending order by their total weighted count. We then manually clean the top 500 key phrases by removing key phrases unrelated to machine learning, such as "propos[ed][1] method" or "experiment[al] result[s] show", but also other irrelevant computer science terms, such as "comput[er] vision". Last, starting with the most popular key phrase, we iteratively skip related key phrases. We continue this merger until we find 10 distinct key phrases of different topics, which are the top 10 topics in machine learning. For example, key phrases related to "data set" are "train[ing] data" and "real data". Our implementation is available as open source: `http://github.com/pglauner/MLtop10`.

2.5.2 Evaluation

Using method 1, which utilizes bigrams and trigrams for extraction, we only get very general topics. Concretely, the top 5 topics are "network pretraining", "supervised classification part", "learn binary representation", "unsupervised [and] supervised learning" and "predict label [from the] input". In contrast, performing method 2, which is machine learning-based key word extraction using RAKE, we get the top 10 topics depicted in Figure 2.3. We notice that after the first three topics, i.e. "support vector machine", "neural network" and "data set", there is a significant drop in terms of popularity. We notice another drop after "objective function". The next 7 topics are vey close in terms of their popularity. "Hidden Markov model" has a popularity only slightly lower than "principal component analysis".

2.5.3 Discussion

Comparing the two key phrase extraction methods, we see that using RAKE we obtain more robust results that reflect frequent keywords and unbalanced terms much better.

Comparing our list of top 10 topics to the list of top 10 algorithms in data mining from 2007 [177], we make the following important observations: Due to their popularity in research, we have expected that support vector machines would appear in the top 10. Also, neural networks have been celebrating a comeback under the umbrella term of "deep learning" since 2006 [97]. We therefore expected them to appear in the top 10 as well under either term. We can also confirm that Hidden Markov models have received significantly less attention in research than neural networks over the last 10 years. We have not expected

[1] Stemmed words are completed to their original form for clarity in this study.

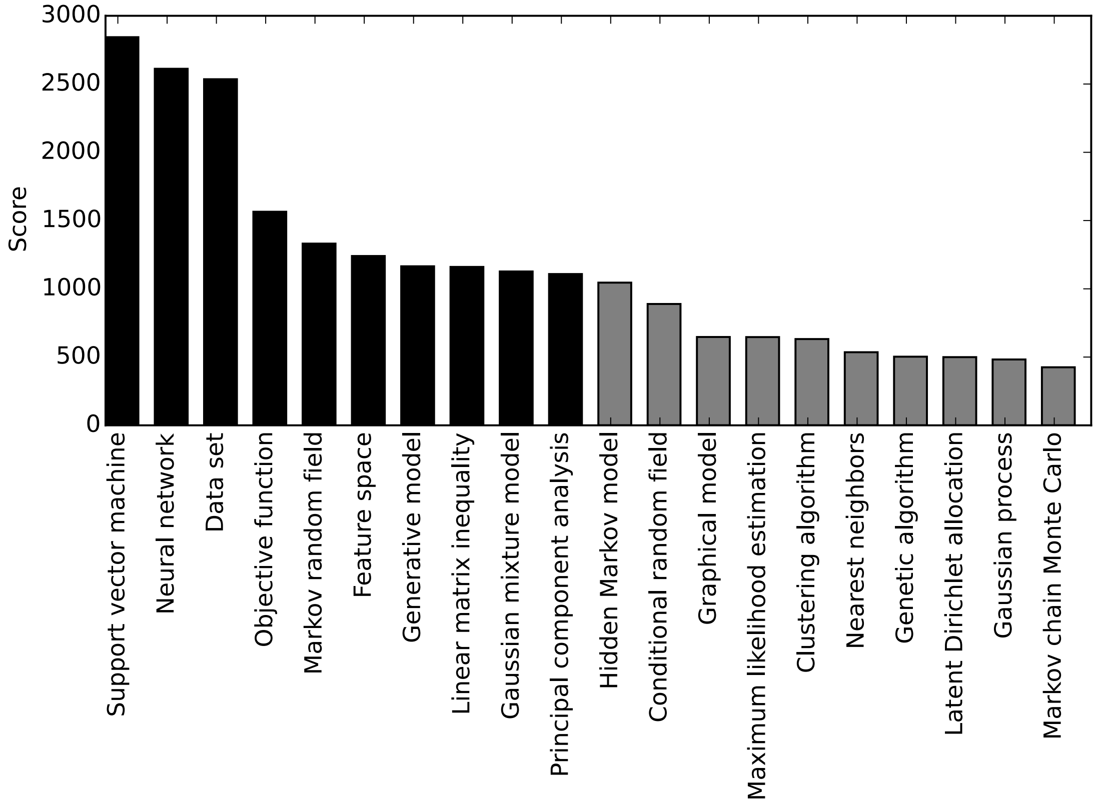


Figure 2.3: The top 10 topics in machine learning are highlighted in black, the top 11-20 topics in grey.

that the linear matrix inequality would appear in the top 10. However, given its importance to the theoretical foundations of the field of machine learning it is absolutely justified to appear in the top 10. Its appearance does not indicate a fallacy in our methodology. Naive Bayes has often been described as a wide-spread baseline model in the literature. Furthermore, tree classifiers such as random forests have become popular in the literature and do not appear in the top 10 either. Both, C4.5 and CART are tree learning algorithms that were found to be among the top 10 data mining algorithms in 2007. In terms of models, we did not expect that Markov random fields and Gaussian mixture models receive more attention than naive Bayes or tree based learning methods in current research publications.

A quantitative approach comes with a potential new bias depending on which data sources are used. Possible factors include the quality of publications and focus of each source (journal/conference). The vast majority of source abstracts are from journals and conferences that have a high impact factor or average citation count. We have made sure to include as many sources as possible that have a wide scope. In return, we have attempted to keep the number of sources with a very narrow scope to a minimum. Also, if the inclusion or omission of a specific source is questioned, this has only very little impact due to the distribution of abstracts: there are in total 39 sources (31 journals + 8 conferences). On average, a source has 1,372 abstracts or 2.56% of all abstracts. The largest source is the Neurocomputing journal, which has 6,165 abstracts or 11.52% of all abstracts.

2.6 Review of Machine Learning Models Relevant to this Thesis

In this section, we provide a brief introduction to the machine learning models that are most relevant to this thesis. We chose these models due to their overall applicability to our problems. Furthermore, we have identified the challenges of NTL detection in Chapter 3.4. As we point out in Chapter 3.4.5, most models reported in the literature on NTL do not scale to large data sets of potentially millions of customers. We therefore also refer to scalability of these models and how these can be trained on big data sets.

2.6.1 Logistic Regression

Logistic regression is a linear classifier that optimizes a convex cross-entropy loss function during the training of the weights [36]. It is related to linear regression, but feeds the continuous output value in the Sigmoid function $\sigma(x) = \frac{1}{1+\exp(-x)}$ in order to predict a probability of binary class membership. Logistic regression scales to big data sets, as the minibatch gradient descent, that is used to optimize the weights, can be parallelized among different cores or nodes.

2.6.2 Support Vector Machine

A support vector machine (SVM) [168] is a maximum margin classifier, i.e. it creates a maximum separation between classes. Support vectors hold up the separating hyperplane. In practice, the support vectors are just a small fraction of the training examples. Therefore, SVMs are reported to tend to be less prone to overfitting than other classifiers, such as neural networks [23]. The training of a SVM can be defined as a Lagrangian dual problem having a convex cost function. In that form, the optimization formulation is only written in terms of the dot product $x^{(i)} \cdot x^{(j)}$ between points in the input space. By default, the separating hyperplane is linear. For complex problems, it is advantageous to map the data set to a higher dimension space, where it is possible to separate them using a linear hyperplane. A kernel is an efficient function that implicitly computes the dot product in the higher dimensional space. A popular kernel is the Gaussian radial basis function: $K(u, v) = \exp(-\gamma \|u - v\|^2)$. Training of SVMs using a kernel to map the input to higher dimension is only feasible for a couple of dozen of thousands of training examples in a realistic amount of time [25]. Therefore, for big data sets only a linear implementation of SVMs is practically usable [131].

2.6.3 Decision Tree

Decision tree learners such as ID3 [137] or C4.5 [138] recursively split the input space by choosing the remaining most discriminative feature of a data set. This process is depicted in Algorithm 2.1. The `IMPORTANCE` function usually utilizes the information gain or the Gini coefficient for choosing the most discriminative feature. To predict, the learned tree is traversed top-down.

Algorithm 2.1 Decision tree learning [149].

```
function DT-LEARNING($examples$, $attributes$, $parent\_examples$)
    if empty($examples$) then return PLURALITY-VAL($parent\_examples$)
    else if all $examples$ have same classification then return the classification
    else if empty($attributes$) then return PLURALITY-VAL($examples$)
    else
        $A \leftarrow \underset{a \in attributes}{\operatorname{argmax}}$ IMPORTANCE($a$, $examples$)
        $tree \leftarrow$ a new decision tree with root test $A$
        for $v_k \in A$ do
            $exs \leftarrow \{e | e \in examples \land e.A = v_k\}$
            $subtree \leftarrow$ DT-LEARNING($exs$, $attributes - A$, $examples$)
            add a branch to $tree$ with label ($A = v_k$) and subtree $subtree$
        end for
        return $tree$
    end if
end function
```

2.6.4 Random Forest

A random forest is an ensemble estimator that comprises a number of decision trees [79]. Each tree is trained on a subsample of the data and feature set in order to control overfitting. In the prediction phase, a majority vote is made of the predictions of the individual trees. Training of the individual trees is independent from each other, so it can be distributed among different cores or nodes. A random forest has been reported to empirically tend to learn a reasonably good model for a large number of problems [53], [169].

2.6.5 Gradient-Boosted Tree

A gradient-boosted tree [28] is also an ensemble of decision trees. The ensemble is boosted by combining weak classifiers (i.e. classifiers that work little better than a random guess) into a strong one. The ensemble is built by optimizing a loss function.

2.6.6 k-Nearest Neighbors

k-nearest neighbors (kNN) is an instance-based or lazy learning method that does not use weights, as there is no training phase as such [4]. During prediction, the class of an example is determined by selecting the majority class of the k nearest training examples. Defining proximity is subject to the selection of a distance function, of which the most popular ones include Euclidean, Manhattan or cosine. k is a smoothing parameter. The larger k, the smoother the output. Since kNN in an instance-based method, predicting is slow and prediction times grow with k. As the prediction of the class of an example in the test set is independent from the other elements, the predictions can be distributed among different cores or nodes.

2.6.7 Comparison

Based on the reviews of the models in this section, we now compare them in Table 2.3 in the context of the properties relevant to this thesis. We notice that decision trees, gradient-boosted trees, logistic regression, random forest and linear support vector machine particularly scale to training on big data sets. This makes them good candidates for further evaluation in this thesis. k-nearest neighbors does not have an explicit training phase as it is a lazy learning method. It somewhat scales to big data sets for a small value of k, the number of nearest training examples.

Table 2.3: Machine learning models relevant to this thesis.

Model	Decision Boundary	Maximum #Examples for Training in Feasible Amount of Time	Reference
Decision tree	Non-linear	>1M	[137]
Gradient-boosted tree	Non-linear	>1M	[28]
k-nearest neighbors	Non-linear	No training phase	[4]
Logistic regression	Linear	>1M	[36]
Random forest	Non-linear	>1M	[79]
Support vector machine	Linear	>1M	[168]
	Non-linear (with kernel)	~20K	[131]

2.7 Conclusions

Artificial intelligence allows computers to solve inherently challenging problems, in particular problems that include uncertainty. The field of artificial intelligence has evolved since the 1950s. While expert system approaches have initially been most popular, there has been a strong shift towards machine learning approaches in the the past decades. In this chapter, we have laid the conceptual foundations in order to use artificial intelligence for the detection of non-technical losses in the remainder of this thesis.

We also used machine learning to find the top 10 topics in machine learning from about 54K abstracts of papers published between 2007 and 2016 in leading machine learning journals and conferences. Concretely, we found support vector machine, neural network, data set, objective function, Markov random field, feature space, generative model, linear matrix inequality, Gaussian mixture model and principal component analysis to be the top 10 topics. Compared to previous work in this field from 2007, support vector machine is the only intersection of both top 10 lists. This intersection is small for the following reasons:

1. We do not only consider models, but span a wider view across the entire field of machine learning also including features, data and optimization.

2. We perform a quantitative study rather than opinion-based surveying of domain experts in order to reduce the bias.

3. The models of interest have significantly changed over the last 10 years, most prominently symbolized by the comeback of neural networks under the term deep learning.

Overall, we are confident that our quantitative study provides a comprehensive view on the ground truth of current machine learning topics of interest in order to strengthen and streamline future research activities.

Readers interested in learning more about artificial intelligence and machine learning are referred to [16], [149], two text books that are used in many graduate schools all around the globe.

3

The State of the Art

Most contemporary approaches for the detection of non-technical losses (NTL) utilize artificial intelligence methods. Early NTL detection systems were mainly based on rule-based expert systems. Over the years, there has been a shift to machine learning - also called data mining - methods. These employ statistical methods to learn fraudulent patterns from the data of customers and known irregular behavior. Once these patterns are learned, they predict for customers whether they should be inspected for NTL or not. Then, for some of these customers, on-site inspections are carried out. This process is depicted in Figure 3.1. This chapter provides an in-depth review of NTL detection systems based on artificial intelligence, in particular machine learning methods.

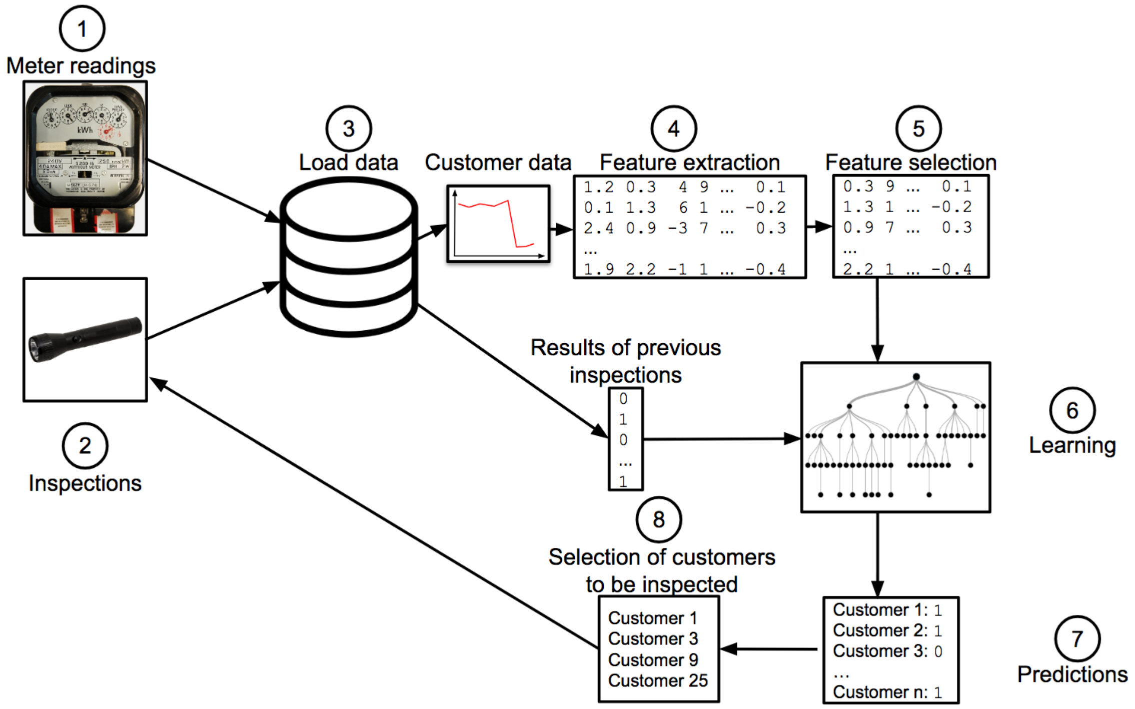


Figure 3.1: NTL detection system based on machine learning: First, meter readings and other customer data are collected. Second, inspections of customers are carried out by technicians. Third, the data of previously inspected customers is loaded, which consists for example of their consumption data as well the inspection result. Fourth, features are extracted from the customer data. Fifth, these features are reduced in order to only retain the statistically meaningful ones. Sixth, using the reduced set of features and the results of previously carried out inspections, classifiers are trained in order to detect NTL. Seventh, these classifiers are then used to predict for customers whether they should be inspected for NTL or not. Eighth, domain expert at the utilities choose the customers for which an inspection appears to be justified from an economic point of view. Last, the inspections are carried out by technicians. See for example Figure 4.2 for a more complex process in a large NTL detection system.

3.1 Scope

NTL detection can be treated as a special case of fraud detection, for which general surveys are provided in [17], [92]. Both highlight expert systems and machine learning as key methods to detect fraudulent behavior in applications such as credit card fraud, computer intrusion and telecommunications fraud. This section is focused on an overview of the existing AI methods for detecting NTL. Existing short surveys of the past efforts in this field, such as in [26], [89], [91], [111] only provide a narrow comparison of the entire range of relevant publications. Two surveys [113], [170] provide broad reviews of the field of NTL detection. Both works not only review AI methods, but also other approaches such as state estimation and network analysis methods. However, both works lack detailed discussions of the works reviewed.

The novelty of our survey is to not only provide an extensive review of AI methods for NTL detection and compare a wide range of results reported in the literature, but also to derive the unsolved challenges of NTL detection.

3.2 Features

In this section, we summarize and group the features reported in the literature.

3.2.1 Monthly Consumption

Many works on NTL detection use traditional meters, which are read monthly or annually by meter readers. Based on this data, average consumption features are used in [120], [122]–[124]. In those works, the feature computation used can be summarized as follows: For M customers $\{0, 1, ..., M-1\}$ over the last N months $\{0, 1, ..., N-1\}$, a feature matrix X is computed, in which element $X_{m,d}$ is a daily average kWh consumption feature during that month:

$$x_d^{(m)} = \frac{L_d^{(m)}}{R_d^{(m)} - R_{d-1}^{(m)}}, \tag{3.1}$$

where, for customer m, $L_d^{(m)}$ is the kWh consumption increase between the meter reading to date $R_d^{(m)}$ and the previous one $R_{d-1}^{(m)}$. $R_d^{(m)} - R_{d-1}^{(m)}$ is the number of days between both meter readings of customer m.

The previous 24 monthly meter readings are used in [116], [117]. The features computed are the monthly consumption before the inspection, the consumption in the same month in the year before the consumption in the past three months and the customer's consumption over the past 24 months. Using the previous six monthly meter readings, the following features are derived in [6]: average consumption, maximum consumption, standard deviation, number of inspections and the average consumption of the residential area. The average consumption is also used as a feature in [34], [159].

3.2.2 Smart Meter Consumption

With the increasing availability of smart meter devices, consumption of electrical energy in short intervals can be recorded. Consumption features of intervals of 15 minutes are used in [22], [44], whereas intervals of 30 minutes are used in [121], [151]. The $4 \times 24 = 96$ measurements of [44] are encoded into a 32-dimensional space in [45], [46]. Each measurement is 0 or positive. Next, it is mapped to 0 or 1, respectively. Last, the final 32 features are computed. A feature is the weighted sum of three subsequent values, in which the first value is multiplied by 4, the second by 2 and the third by 1. The maximum consumption in any 15-minute period is used as a feature in [140], [141], [143], [144]. The load factor is computed by dividing the demand contracted by the maximum consumption. Features from the consumption time series called shape factors are derived from the consumption time series including the impact of lunch times, nights and weekends in [127].

3.2.3 Master Data

Master data represents customer reference data such as name or address, which typically changes less frequently with respect to the consumption data. The work in [34] uses the following features from the master data for classification: location (city and neighborhood), business class (e.g. residential or industrial), activity type (e.g. residence or drugstore), voltage (110V or 200V), number of phases (1, 2 or 3) and meter type. The demand contracted, which is the number of kW of continuous availability requested from the energy company and the total demand in kW of installed equipment of the customer are used in [141], [143], [144]. In addition, information about the power transformer to which the customer is connected to is used in [140]. The town or village in which the customer is located, the type of voltage (low, median or high), the electricity tariff, the contracted power as well as the number of phases (1 or 3) are used in [159]. Related master data features are used in [127], including the type of customer, location, voltage level, type of climate (rainy or hot), weather conditions and type of day.

3.2.4 Credit Worthiness

The works in [120], [122], [123] use the credit worthiness ranking (CWR) of each customer as a feature. It is computed from the electric utility's billing system and reflects if a customer delays or avoids payments of bills. CWR ranges from 0 to 5 where 5 represents the maximum score. It reflects different information about a customer such as payment performance, income and prosperity of the neighborhood in a single feature.

3.3 Models

In this section, we summarize and group the models reported in the literature.

3.3.1 Expert Systems and Fuzzy Systems

Profiles of 80K low-voltage and 6K high-voltage customers in Malaysia having meter readings every 30 minutes over a period of 30 days are used in [121] for electricity theft and abnormality detection. A test recall of 0.55 is reported. This work is related to features of [124], however, it uses entirely fuzzy logic incorporating human expert knowledge for detection.

The work in [123] is combined with a fuzzy logic expert system postprocessing the output of the SVM in [124] for ∼100K customers. The motivation of that work is to integrate human expert knowledge into the decision making process in order to identify fraudulent behavior. A test recall of 0.72 is reported.

Five features of customers' consumption of the previous six months are derived in [6]: average consumption, maximum consumption, standard deviation, number of inspections and the average consumption of the residential area. These features are then used in a fuzzy c-means clustering algorithm to group the customers into c classes. Subsequently, the fuzzy membership values are used to classify customers into NTL and non-NTL using the Euclidean distance measure. On the test set, an average precision (called average assertiveness) of 0.745 is reported.

An ensemble pre-filters the customers to select irregular and regular customers in [117]. These customers are then used for training as they represent well the two different classes. This is done because of noise in the inspection labels. In the classification step, a neuro-fuzzy hierarchical system is used. In this setting, a neural network is used to optimize the fuzzy membership parameters. A precision of 0.512 and an accuracy of 0.682 on the test set are obtained.

3.3.2 Neural Networks

Extreme learning machines (ELM) are one-hidden layer neural networks in which the weights from the inputs to the hidden layer are randomly set and never updated. Only the weights from the hidden to output layer are learned. The ELM algorithm is applied to NTL detection in meter readings of 30 minutes in [126], for which a test accuracy of 0.5461 is reported.

An ensemble of five neural networks (NN) is trained on samples of a data set containing ∼20K customers in [116]. Each neural network uses features calculated from the consumption time series plus customer-specific pre-computed attributes. A precision of 0.626 and an accuracy of 0.686 are obtained on the test set.

A self-organizing map (SOM) is a type of unsupervised neural network training algorithm that is used for clustering. SOMs are applied to weekly customer data of 2K customers consisting of meter readings of 15 minutes in [22]. This allows to cluster customers' behavior into fraud or non-fraud. Inspections are only carried out if certain hand-crafted criteria are satisfied including how well a week fits into a cluster and if no contractual changes of the customer have taken place. A test accuracy of 0.9267, a test precision of 0.8526, and test recall of 0.9779 are reported.

A data set of ∼22K customers is used in [34] for training a neural network. It uses the average consumption of the previous 12 months and other customer features such as location, type of customer, voltage and whether there are meter reading notes during that period. On the test set, an accuracy of 0.8717, a precision of 0.6503 and a recall of 0.2947 are reported.

3.3.3 Support Vector Machines

Electricity customer consumption data of less than 400 highly imbalanced out of ~260K customers in Kuala Lumpur, Malaysia are used in [120]. Each customer has 25 monthly meter readings in the period from June 2006 to June 2008. From these meter readings, daily average consumption features per month are computed. Those features are then normalized and used for training in a SVM with a Gaussian kernel. In addition, credit worthiness ranking (CWR) is used. It is computed from the electric utility's billing system and reflects if a customer delays or avoids payments of bills. CWR ranges from 0 to 5 where 5 represents the maximum score. It was observed that CWR is a significant indicator of whether customers commit electricity theft. For this setting, a recall of 0.53 is achieved on the test set. A related setting is used in [123], where a test accuracy of 0.86 and a test recall of 0.77 are reported on a different data set.

SVMs are also applied to 1,350 Indian customer profiles in [44]. They are split into 135 different daily average consumption patterns, each having 10 customers. For each customer, meters are read every 15 minutes. A test accuracy of 0.984 is reported. That work is extended in [45] by encoding the $4 \times 24 = 96$-dimensional input in a lower dimension indicating possible irregularities. This encoding technique results in a simpler model that is faster to train while not losing the expressiveness of the data and results in a test accuracy of 0.92.

Consumption profiles of 5K Brazilian industrial customer profiles are analyzed in [140]. Each customer profile contains 10 features including the demand billed, maximum demand, installed power, etc. In this setting, a SVM slightly outperforms k-nearest neighbors (kNN) and a neural network, for which test accuracies of 0.9628, 0.9620 and 0.9448, respectively, are reported.

The work of [45] is extended in [46] by introducing high performance computing algorithms in order to enhance the performance of the previously developed algorithms. This faster model has a test accuracy of 0.89.

3.3.4 Genetic Algorithms

The work in [120], [123] is extended by using a genetic SVM for 1,171 customers in [122]. It uses a genetic algorithm in order to globally optimize the hyperparameters of the SVM. Each chromosome contains the Lagrangian multipliers $(\alpha_1, ..., \alpha_i)$, regularization factor C and Gaussian kernel parameter γ. This model achieves a test recall of 0.62.

A data set of ~1.1M customers is used in [35]. The paper identifies the much smaller class of inspected customers as the main challenge in NTL detection. It then proposes stratified sampling in order to increase the number of inspections and to minimize the statistical variance between them. The stratified sampling procedure is posed as a non-linear restricted optimization problem of minimizing the overall energy loss due to electricity theft. This minimization problem is solved using two methods: (1) genetic algorithm and (2) simulated annealing. The first approach outperforms the second one. Only the reduced variance is reported, which is not comparable to the other research and therefore left out of this survey.

3.3.5 Rough Sets

Rough sets give lower and upper approximations of an original conventional or crisp set. The first application of rough set analysis applied to NTL detection is described in [21] on 40K customers, but lacks details on the attributes used per customer, for which a test accuracy of 0.2 is achieved. Rough set analysis is also applied to NTL detection in [159] on features related to [34]. This supervised learning technique allows to approximate concepts that describe fraud and regular use. A test accuracy of 0.9322 is reported.

3.3.6 Miscellaneous

Different feature selection techniques for customer master data and consumption data are assessed in [127]. Those methods include complete search, best-first search, genetic search and greedy search algorithms for the master data. Other features called shape factors are derived from the consumption time series including the impact of lunch times, nights and weekends on the consumption. These features are used in k-means for clustering similar consumption time series. In the classification step, a decision tree is used to predict whether a customer causes NTL or not. An overall test accuracy of 0.9997 is reported.

Optimum path forests (OPF), a graph-based classifier, is used in [141]. It builds a graph in the feature space and uses so-called “prototypes" or training samples. Those become roots of their optimum-path tree node. Each graph node is classified based on its most strongly connected prototype. This approach is fundamentally different to most other learning algorithms such as SVMs or neural networks which learn hyperplanes. Optimum path forests do not learn parameters, thus making training faster, but predicting slower compared to parametric methods. They are used in [143] for 736 customers and achieved a test accuracy of 0.9021, outperforming SMVs with Gaussian and linear kernels and a neural network which achieved test accuracies of 0.8893, 0.4540 and 0.5301, respectively. Related results and differences between these classifiers are also reported in [144].

A different method is to estimate NTL by subtracting an estimate of the technical losses from the overall losses reported in [151]. It models the resistance of the infrastructure in a temperature-dependent model using regression which approximates the technical losses. It applies the model to a data set of 30 customers for which the consumption was recorded for six days with meter readings every 30 minutes for theft levels of 1, 2, 3, 4, 6, 8 and

10%. The respective test recalls in linear circuits are 0.2211, 0.7789, 0.9789, 1, 1, 1 and 1, respectively.

3.3.7 Summary

A summary and comparison of models, data sets and performance measures of selected works discussed in this review is reported in Table 3.1. The most commonly used models comprise SVMs and neural networks. In addition, genetic methods, OPF and regression methods are used. Data set sizes have a wide range from 30 up to 700K customers. However, the largest data set of 1.1M customers in [35] is not included in the table because only the variance is reduced and no other performance measure is provided. Accuracy and recall are the most popular performance measures in the literature, ranging from 0.45 to 0.99 and from 0.29 to 1, respectively. Only very few publications report the precision of their models, ranging from 0.51 to 0.85.

3.4 Challenges

The research reviewed in the previous section indicates multiple open challenges. These challenges do not apply to single contributions, rather they spread across different ones. In this section, we discuss these challenges, which must be addressed in order to advance in NTL detection. Concretely, we discuss common topics that have not yet received the necessary attention in previous research and put them in the context of AI research as a whole.

3.4.1 Class Imbalance and Evaluation Metric

Imbalanced classes appear frequently in machine learning, which also affects the choice of evaluation metrics as discussed in [87], [161]. Most NTL detection research do not address this property. Therefore, in many cases, high accuracies or high recalls are reported, such as in [34], [35], [120], [141], [159]. Example 3.1 demonstrates why those performance measures are not suitable for NTL detection in imbalanced data sets.

Example 3.1. Anomaly detection problems often work on particularly imbalanced data sets. A test set containing 1K customers of which 999 have regular behavior and 1 has irregular behavior, (1) a classifier always predicting regular behavior has an accuracy of 99.9%, whereas in contrast, (2) a classifier always predicting irregular behavior has a recall of 100%. While the classifier of the first example has a very high accuracy and intuitively seems to perform very well, it will never find any irregular behavior. In contrast, the classifier of the second example will find all customers that have irregular behavior, but may potentially trigger many costly and unnecessary interventions for customers that have a regular behavior.

Table 3.1: Summary of models, data sets and performance measures (two-decimal precision).

Reference	Model	#Customers	Accuracy	Precision	Recall	NTL/Theft Proportion
[22]	SOM	2K	0.93	0.85	0.98	-
[34]	NN	22K	0.87	0.65	0.29	-
[44]	SVM (Gauss)	1,350	0.98	-	-	-
[117]	Neuro-fuzzy	20K	0.68	0.51	-	-
[120]	SVM	< 400	-	-	0.53	-
[122]	Genetic SVM	1,171	-	-	0.62	-
[123]	SVM (Gauss)	< 400	0.86	-	0.77	-
[124]	SVM + fuzzy	100K	-	-	0.72	-
[127]	Decision tree	N/A	0.99	-	-	-
[140]	SVM	5K	0.96	-	-	-
[140]	kNN	5K	0.96	-	-	-
[140]	NN	5K	0.94	-	-	-
[143]	OPF	736	0.90	-	-	-
[143]	SVM (Gauss)	736	0.89	-	-	-
[143]	SVM (linear)	736	0.45	-	-	-
[143]	NN	736	0.53	-	-	-
[151]	Regression	30	-	-	0.22	1%
[151]	Regression	30	-	-	0.78	2%
[151]	Regression	30	-	-	0.98	3%
[151]	Regression	30	-	-	1	4-10%
[159]	Rough sets	N/A	0.93	-	-	-

This topic is rarely addressed in NTL literature, such as in [47], [116], and these contributions do not use a proper single measure of performance of a classifier when applied to an imbalanced data set.

3.4.2 Feature Description

Generally, hand-crafting features from raw data is a long-standing issue in machine learning having significant impact on the performance of a classifier, as discussed in [49]. Different feature description methods have been reviewed in the previous section. They fall into two main categories: features computed from the consumption profile of customers, which are from monthly meter readings, for example in [6], [34], [116], [117], [120], [122]–[124], [159], or smart meter readings, for example in [22], [44]–[46], [121], [127], [140], [141], [143], [144], [151], and features from the customer master data in [34], [127], [140], [141], [143], [144], [159]. The features computed from the time series are very different for monthly meter readings and smart meter readings. The results of those works are not easily interchangeable. While electric utilities continuously upgrade their infrastructure to smart metering, there will be many remaining traditional meters. In particular, this applies to emerging countries.

Also, almost all works on NTL detection define features and subsequently report improved models that were mostly found experimentally without having a strong theoretical foundation.

3.4.3 Data Quality

In the preliminary work of this thesis, we noticed that the inspection result labels in the training set may not always be correct and that some fraudsters may be labelled as non-fraudulent. The reasons for this may include bribing, blackmailing or threatening of the technician performing the inspection. Also, the fraud may be hidden well and is therefore not observable by technicians. Another reason may be incorrect processing of the data. It must be noted that the latter reason may, however, also label non-fraudulent behavior as fraudulent. Handling noise is a common challenge in machine learning. In supervised machine learning settings, most existing methods address handling noise in the input data. There are different regularization methods such as L_1 or L_2 regularization discussed in [125] or learning of invariances allowing learning algorithms to better handle noise in the input data discussed in [16], [96]. However, handling noise in the training labels is less commonly addressed in the machine learning literature. Most NTL detection research use supervised methods. This shortcoming of the training data and potential wrong labels in particular are only rarely reported in the literature, such as in [117], which uses an ensemble to pre-filter the training data.

3.4.4 Covariate Shift

Covariate shift refers to the problem of training data (i.e. the set of inspection results) and production data (i.e. the set of customers to generate inspections for) having different distributions. This fact leads to unreliable NTL predictors when learning from the training data. Historically, covariate shift has been a long-standing issue in statistics, as surveyed in [75]. For example, The Literary Digest sent out 10M questionnaires in order to predict the outcome of the 1936 US Presidential election. They received 2.3M returns. Nonetheless, the predicted result proved to be wrong because the voters contacted by the Literary Digest represented a biased sample of the overall population. In contrast, George Gallup only interviewed 3K handpicked people, which were an unbiased sample of the population. As a consequence, Gallup could predict the outcome of the election very well. We discuss this historical artifact in detail in Example 6.2.

For about the last fifteen years, the big data paradigm followed in machine learning has been to gather more data rather than improving models. Hence, one may assume that having simply more customers and inspection data would help to detect NTL more accurately. However, in many cases, the data may be biased as depicted in Figure 3.2.

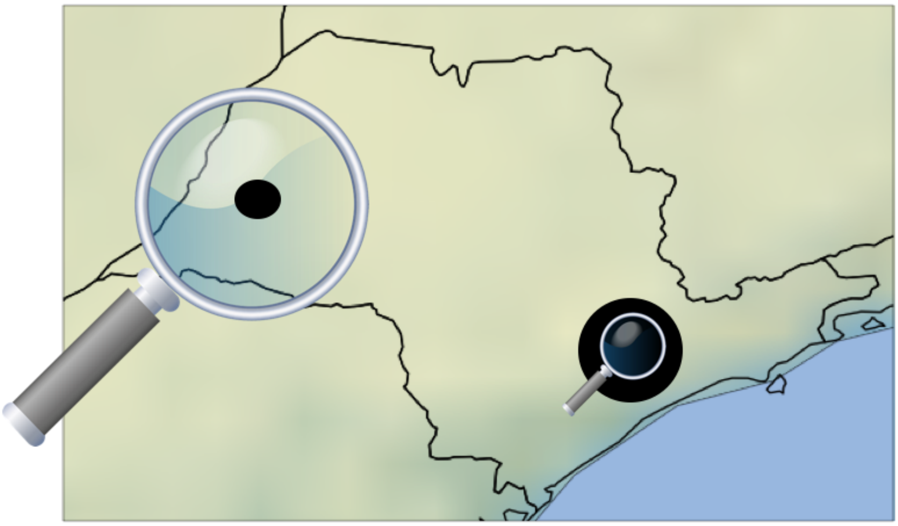


Figure 3.2: Example of spatial bias: The large city is close to the sea, whereas the small city is located in the interior of the country. The weather in the small city undergoes stronger changes during the year. The subsequent change of electricity consumption during the year triggers many inspections. As a consequence, most inspections are carried out in the small city. Therefore, the sample of customers inspected does not represent the overall population of customers.

One reason is, for example, that electricity suppliers previously may have focused on certain neighborhoods for inspections. Concretely, the set of customers inspected is a sample of the overall population of customers. In this example, there is a spatial bias. Hence, the inspections do not represent the overall population of customers. As a consequence, when learning from the inspection results, a bias is learned, making predictions less reliable. Aside from spatial covariate shift, there may be other types of covariate shift in the data, such as the meter type, connection type, etc.

To the best of our knowledge, the issue of covariate shift has not been addressed in the literature on NTL detection. However, in many cases it may lead to unreliable NTL detection models. Therefore, we consider it important to derive methods for quantifying and reducing the covariate shift in data sets relevant to NTL detection. This will allow to build more reliable NTL detection models.

3.4.5 Scalability

The number of customers used throughout the research reviewed significantly varies. For example, [120], [151] only use less than a few hundred customers in the training. A SVM with a Gaussian kernel is used in [120]. In that setting, training is only feasible in a realistic amount of time for up to a couple of tens of thousands of customers in current implementations as discussed in [25]. A regression model using the Moore-Penrose pseudoinverse introduced in [132], [151]. This model is also only able to scale to up to a couple of tens of thousands of customers. Neural networks are trained on up to a couple of tens of thousands of customers in [34], [116]. The training methods used in prior work usually do not scale to significantly larger customer data sets. A large data set using up to hundreds of thousands in [35] uses genetic algorithms. An important property of NTL detection methods is that their computational time must scale to large data sets of hundreds of thousands or millions of customers. Most works reported in the literature do not satisfy this requirement.

3.4.6 Comparison of Different Methods

Comparing the different methods reviewed in this chapter is challenging because they are tested on different data sets, as summarized in Table 3.1. In many cases, the description of the data lacks fundamental properties such as the number of meter readings per customer, NTL proportion, etc. In order to increase the reliability of a comparison, joint efforts of different research groups are necessary. These efforts need to address the benchmarking and comparability of NTL detection systems based on a comprehensive freely available data set.

3.5 Conclusions

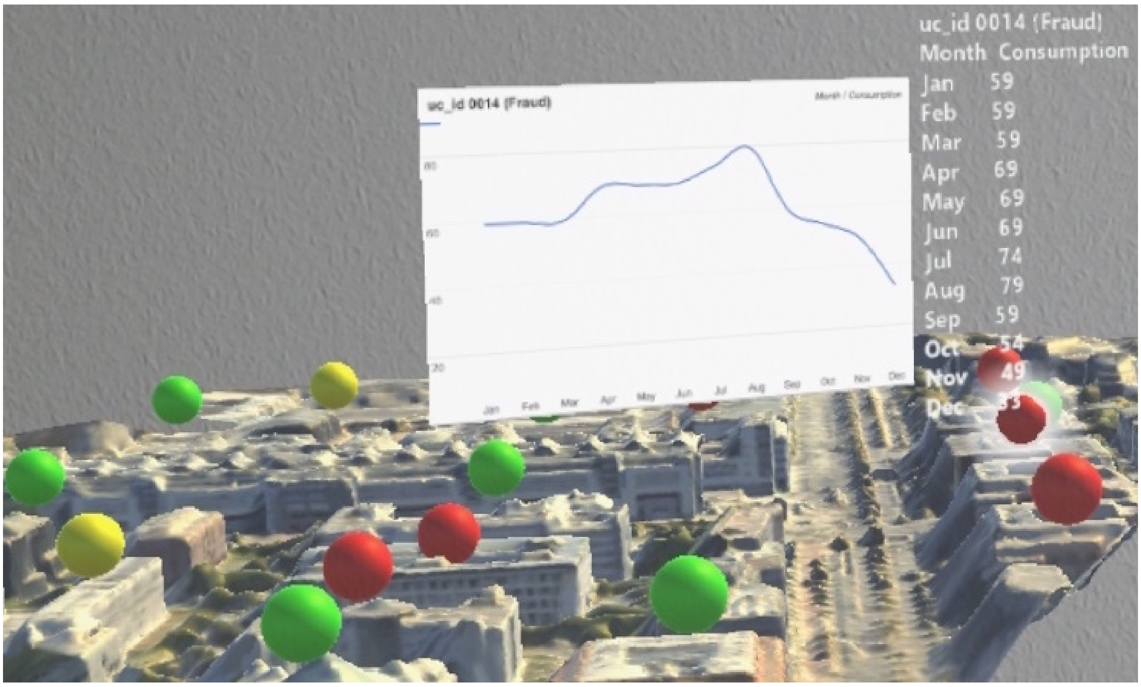
Non-technical losses (NTL) are the predominant type of losses in electricity power grids. In the literature, a vast variety of NTL detection methods employing artificial intelligence methods are reported. Expert systems and fuzzy systems are traditional detection models. Over the past years, machine learning methods have become more popular. The most commonly used methods are support vector machines and neural networks, which outperform expert systems in most settings. These models are typically applied to features computed from customer consumption profiles such as average consumption, maximum consumption and change of consumption in addition to customer master data features such as type of customer and connection types. Sizes of data sets used in the literature have a large range from less than 100 to more than one million. In this chapter, we have also identified the six main open challenges of NTL detection: handling imbalanced classes in the training data and choosing appropriate evaluation metrics, describing features from the data, handling incorrect inspection results, correcting the covariate shift in the inspection results, building models scalable to big data sets and making results obtained through different methods comparable. We believe that these need to be accurately addressed in future research in order to advance in NTL detection methods. This will allow to share sound, assessable, understandable, replicable and scalable results with the research community. We are confident that this comprehensive survey of challenges will allow other research groups to not only advance in NTL detection, but in anomaly detection as a whole.

4

Expert Knowledge, Machine Learning and Visualization of Predictions

To date, most NTL detection systems deployed in industry are based on expert knowledge rules. In contrast, the predominant research direction reported in the recent research literature is the use of machine learning/data mining methods, which learn from customer data and known irregular behavior that was reported through inspection results. Due to the high costs per inspection and the limited number of possible inspections, electric utilities aim to maximize the return on investment (ROI) of inspections. In this chapter, we compare expert systems to machine learning for NTL detection. We specifically address the challenges of class imbalance and evaluation metric, feature description and scalability that we identified in Chapters 3.4.1, 3.4.2 and 3.4.5, respectively.

As electric utilities are keen to include their expert knowledge when deciding which customers to inspect, we then combine both worlds in a new approach that allows domain experts to visualize the prediction results of NTL classifiers in a holographic spatial visualization. An example of this holographic visualization is depicted in Figure 4.1. Using this hologram, domain experts can then review and amend the suggestions of which customers to inspect in order to increase the return on investment of inspections. The entire NTL detection process proposed and evaluated in this chapter is depicted in Figure 4.2.



uc_id 0014 (Fraud)

Month	Consumption
Jan	59
Feb	59
Mar	59
Apr	69
May	69
Jun	69
Jul	74
Aug	79
Sep	59
Oct	54
Nov	49
Dec	33

Figure 4.1: Example usage of our NTL detection system: Customers are classified as either regular (green), irregular (red) or suspicious (yellow) by a machine learning system. Holographic spatial visualization of customers allows domain experts at the electric utilities to gather information about the customers as well as their neighborhood in order to decide which customers to inspect. The figure depicts the profile of an irregular customer whose consumption has significantly dropped in the last few months.

4.1 Detecting Irregular Power Usage

The data used in this chapter is from an electric utility in Brazil. It consists of three parts: (i) ∼700K customer data, such as location, type, etc., (ii) ∼31M monthly consumption data from January 2011 to January 2015 such as consumption in kWh, date of meter reading and number of days between meter readings and (iii) ∼400K inspection data such as presence of fraud or irregularity, type of NTL and inspection notes.

Most inspections did not find NTL, making the classes highly imbalanced. In order for the models to be applied to other regions or countries, they must be assessed on different NTL proportions. Therefore, the data was subsampled using 17 different NTL proportion levels: 0%, 0.1%, 1%, 2%, 3%, 4%, 5%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% and 100%. Each sample contains ∼100K inspection results.

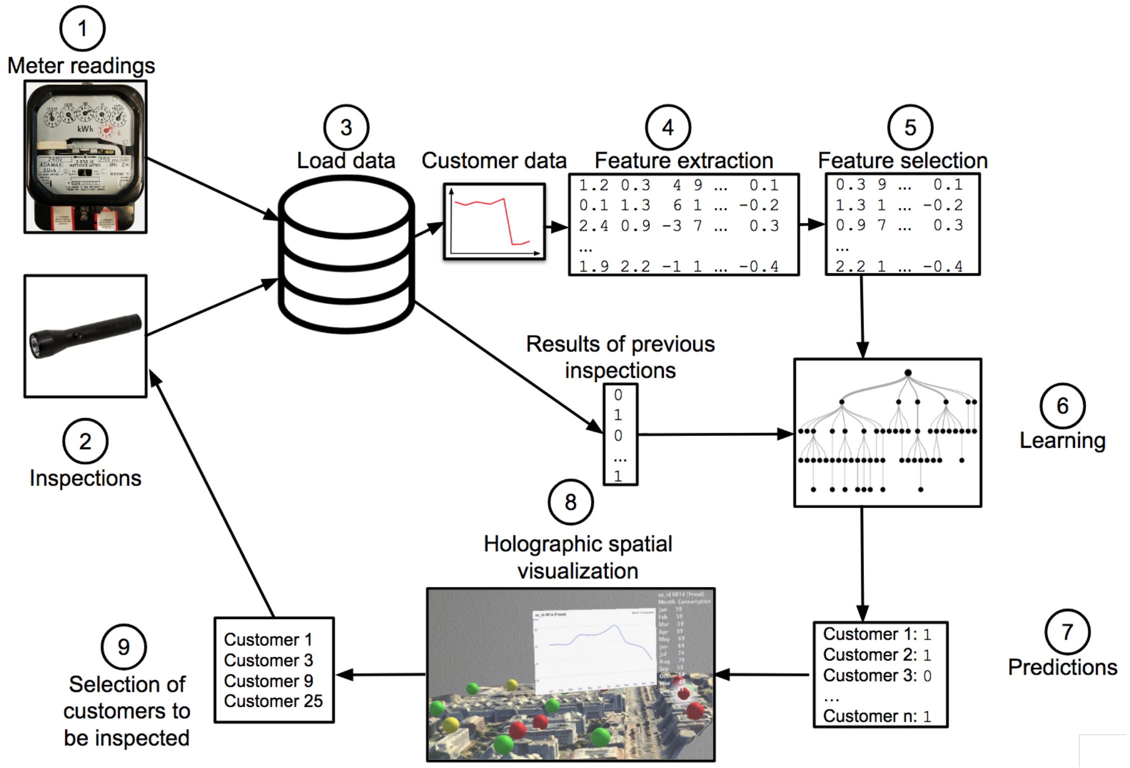


Figure 4.2: NTL detection system combining machine learning and expert knowledge extending the general NTL detection process depicted in Figure 3.1: First, meter readings and other customer data are collected. Second, inspections of customers are carried out by technicians. Third, the data of previously inspected customers is loaded, which consists for example of their consumption data as well the inspection result. Fourth, features are extracted from the customer data. Fifth, these features are reduced in order to only retain the statistically meaningful ones. Sixth, using the reduced set of features and the results of previously carried out inspections, classifiers are trained in order to detect NTL. Seventh, these classifiers are then used to predict for customers whether they should be inspected for NTL or not. Eighth, domain experts visualize the customers, their neighbors, inspection results and other data such as the consumption data in a spatial 3D hologram. Ninth, domain expert at the utilities choose the customers for which an inspection appears to be justified from an economic point of view. Last, the inspections are carried out by technicians. Please note that the fifth step is not performed in this chapter. Instead, it is extensively performed in Chapters 5 and onwards.

4.1.1 Models

We now describe three different models for NTL detection. The first model is a CHOICE Technologies product based on Boolean logic and is used as a baseline. It is extended to fuzzy logic in the second model in order to smoothen the decision making process. The third model is a Support Vector Machine, a state-of-the-art machine learning algorithm.

Expert System

This model is an expert system, it consists of hand-crafted rules created by the CHOICE Technologies expert team which are conjunctions of (in)equality terms, such as:

$$(N_1 > v_1) \wedge (N_1 < v_2) \wedge (N_2 < v_3) \wedge (N_3 = v_4)... \tag{4.1}$$

N_x are so-called attributes. Possible attributes are change of consumption over the last 3 months, slope of consumption curves. v_x are numeric values. In total, 42 attributes are used in 14 rules. If at least one rule outcome is true, that customer is considered to potentially cause NTL.

Fuzzy systems [7] have a long tradition in control applications allowing to implement expert knowledge in a softer decision making process. They can be used to relate to classes of objects, breaking up boundaries, making membership a matter of degree. In this chapter, the 14 Boolean rules were fuzzified and incorporated in a Mamdani fuzzy system using the centroid defuzzification method [7]. Fuzzy rules rely on membership functions. The number of membership functions for each attribute depends on the ranges of values found in the rules among which 1 attribute has 1 function, 32 attributes have 2 functions and 9 attributes have 4 functions. In most cases, trapezoid membership functions are used to keep the model simple. The exact parameters, such as membership function boundaries or the mean of sigmoid membership functions were determined from the distribution of attribute values.

However, these parameters could be optimized using: (i) gradient techniques [147], (ii) genetic algorithms [147] or (iii) neuro-fuzzy systems [2]. Techniques (i) and (ii) are highly constrained optimization problems due to dependence among parameter values to keep the fuzzy system valid. Technique (iii) requires a Sugeno instead of a Mamdani fuzzy system, in which the number of rules equals the number of output membership functions, which is not applicable to the rules used in this chapter. Technique (i) was implemented and studied further. Its results are reported in this chapter.

Machine Learning

Inspired by [123], we compute for M customers $\{0, 1, ..., M-1\}$ over the last N months $\{0, 1, ..., N-1\}$, a feature matrix X. Element $X_{m,d}$ is the daily average kWh consumption feature during that month d of customer m that is computed using Equation 3.1. Similarly, a binary target vector y is created in which element $y^{(m)}$ is the most recent inspection result for customer m in the respective period of time. NTL are encoded by 1 if they are detected and 0 if not.

4.1.2 Evaluation

Metric

In many classification problems, the classification rate, or accuracy is used as a performance measure. Given the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN):

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}. \tag{4.2}$$

However, many publications ignore that it is only of minor expressiveness for imbalanced classes as discussed in Example 3.1. It clearly demonstrates that other performance measures must be used for NTL detection.

The recall is a measure of the proportion of the true positives found. It is also named true positive rate (TPR) or sensitivity:

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \tag{4.3}$$

The specificity is a measure of the proportion of the true negatives classified as negative. It is also named true negative rate (TNR):

$$\mathrm{Specificity} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}. \tag{4.4}$$

The false positive rate (FPR) is $1 - TNR$.

A receiver operating characteristic (ROC) curve plots the TPR against the FNR. The area under the curve (AUC) is a performance measure between 0 and 1, where any binary classifier with an AUC > 0.5 performs better than random guessing. While in many applications multiple thresholds are used to generate points plotted in a ROC curve, the AUC can also be computed for a single point, when connecting it with straight lines to $(0,0)$ and $(1,1)$ as shown in [82]:

$$\mathrm{AUC} = \frac{\mathrm{Recall} + \mathrm{Specificity}}{2}. \tag{4.5}$$

For NTL detection, the goal is to reduce the FPR to decrease the number of costly inspections, while increasing the TPR to find as many NTL occurrences as possible. In order to assess a NTL prediction model using a single performance measure, the AUC is the most suitable.

Methodology

Throughout the experiments, consumption readings and inspection result data are used. Further data, such as location of customers are not used. In the comparison of the three classifiers, the AUC performance measure is used for the different levels of NTL proportion mentioned in the beginning of this chapter. We assessed different values for the number of the most recent meter readings N. Only customers with a complete time series of the last N months before the respective inspection are considered. The larger N, the less data is available. At least 12 months should be considered in order to represent seasonality effects. Experiments for the last 12, 18 and 24 months were carried out, for which 12 months have proven to lead to the best results as the other experiments lead to more overfitting.

The SVM is the only classifier that requires training in our experiments. However, since it is a binary classifier, it could not be trained on NTL proportions of 0% and 100%. For the NTL proportions used for training, 10-fold cross-validation is performed for every NTL proportion, splitting the data into a 60%/20%/20% training/validation/test ratio. The AUC score is used as the validation measure to pick the best classifier fold. Throughout the experiments, a linear SVM is used. The same experiments were repeated using a Gaussian Kernel, which proved to overfit for all NTL proportions.

Implementation

The Boolean and fuzzy classifiers were implemented in `MATLAB`, the latter using the Fuzzy Logic Toolbox [108]. The SVM classifier was implemented in Python using `scikit-learn` [131], which builds on top of `LIBSVM` [25]. The regularization parameter and the inverse variance parameter γ of the Gaussian kernel were optimized using `scikit-learn`. Using 10-fold cross-validation to train 10 SVMs and to select the best one takes about 2 minutes per NTL proportion on a state-of-the-art i5 notebook. Using the Boolean or fuzzy systems to classify the same amount of data takes about 1 second. However, both classifiers use pre-computed customer-specific attributes. Computing those takes a couple of hours in a cloud infrastructure.

Comparison of Classifier Performance

For different NTL proportions, the change of test AUC for the Boolean and fuzzy systems and the SVM can be observed in Figure 4.3. The Boolean classifier has an AUC < 0.5 for all NTL proportions and therefore performs worse than random guessing. The same applies for the fuzzy system, except for a NTL proportion of 0.1%. The SVM performs only (noticeably) better than random guessing for NTL proportions between 50% and 80%.

Given the theory of fuzzy systems and their potential, the parameters of the fuzzy system were optimized using stochastic gradient descent (SGD) for each of the 15 binary NTL proportions: 0.1% to 90%. Out of the 15 optimized fuzzy systems, the one with

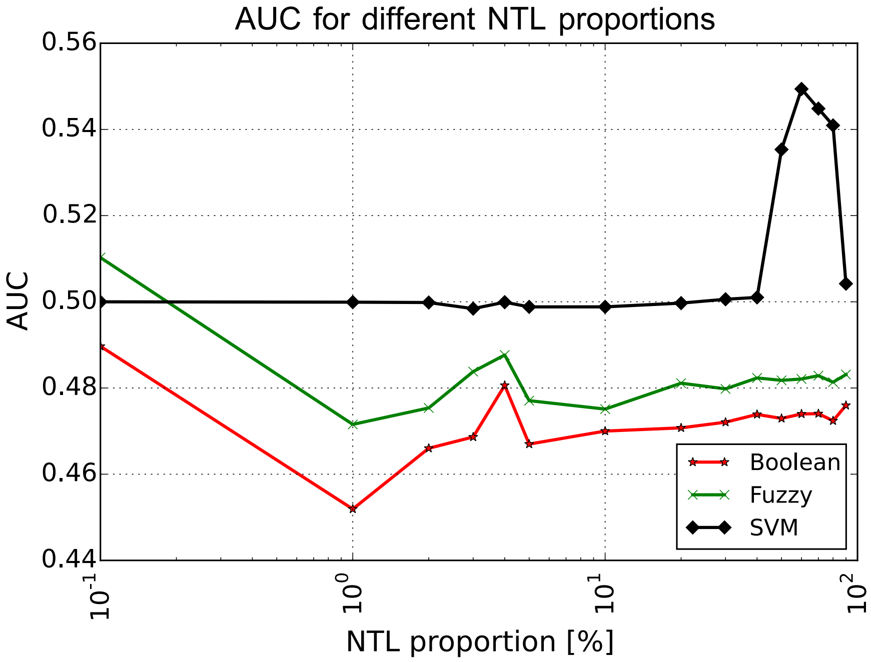


Figure 4.3: Comparison of classifiers tested on different NTL proportions.

the greatest AUC test score is picked and tested on all NTL proportions. The fuzzy system trained on 30% and tested on all NTL proportions - Fuzzy SGD 30% - significantly outperforms both, the Boolean and fuzzy systems, as shown in Figure 4.4.

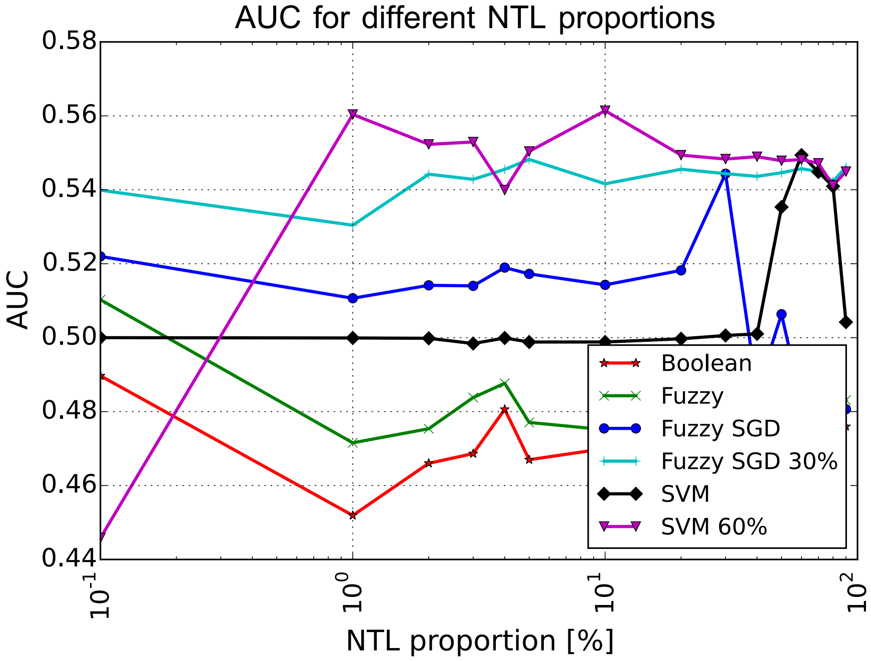


Figure 4.4: Comparison of optimized classifiers tested on different NTL proportions.

However, comparing the confusion matrices of both classifiers, they perform very differently as shown in Tables 4.1 and 4.2 for selected NTL levels of 5% and 20%, respectively. The optimized fuzzy system has a higher TNR, but lower TPR compared to the optimized SVM. In return, the SVM has a higher TPR, but a lower FNR.

Table 4.1: Normalized confusion matrices for test on 5% NTL proportion.

		Predicted	
Classifier	Actual	TNR	FPR
		FNR	TPR
Boolean	Actual	0.53	0.47
		0.60	0.40
Fuzzy SGD 30%	Actual	0.87	0.13
		0.77	0.23
SVM 60%	Actual	0.36	0.64
		0.26	0.74

Table 4.2: Normalized confusion matrices for test on 20% NTL proportion.

		Predicted	
Classifier	Actual	TNR	FPR
		FNR	TPR
Boolean	Actual	0.53	0.47
		0.58	0.42
Fuzzy SGD 30%	Actual	0.87	0.13
		0.78	0.22
SVM 60%	Actual	0.35	0.65
		0.25	0.75

4.1.3 Discussion

The initial industrial Boolean and fuzzy models perform worse than random guessing and are therefore not suitable for real data, as they trigger too many inspections while not many of them will lead to NTL detection. Optimized fuzzy and SVM models trained on 30% and 60% NTL proportion, respectively, result in significantly greater AUC scores. However, both perform very differently, as the optimized fuzzy system is more conservative in NTL detection. In contrast, the optimized SVM is more optimistic, leading also to a higher FPR. In general, neither can be named better than the other one, as picking the appropriate model from these two is subject to business decisions.

However, this work also demonstrates that for real data, NTL classifiers using only the consumption profile are limited. Therefore, it is desirable to use more features like location, inspection notes, etc. Another issue with the real data is the potential bias of inspections so that this sample of customers does not represent the overall population of customers. We expect a correction of the bias to lead to better predictions, too.

4.2 Holographic Visualization of Irregular Power Usage

The NTL detection approach based on machine learning presented in this chapter allows to predict whether customers cause NTL or not. It can then be used to trigger possible inspections of customers that have irregular electricity consumption patterns. Subsequently, technicians carry out inspections, which allow them to remove possible manipulations or malfunctions of the power distribution infrastructure. Furthermore, the fraudulent customers can be charged for the additional electricity consumed. Generally, carrying out inspections is costly, as it requires physical presence of technicians. In order to increase both the return on investment (ROI) of the limited number of inspections and the reliability and stability of the power grid, electric utilities in practice strongly rely on expert knowledge for making the decision of whether to inspect a customer or not. As a consequence, electric utilities are reluctant to move to large-deployments of NTL detection systems based on machine learning. Our goal is to visualize customers, their neighborhood and predictions whether customers cause NTL in 3D spatial holograms that are for example depicted in Figures 4.1 and 4.5. We therefore aim to combine automated statistical decision making for generating inspection proposals with incorporating knowledge of the domain experts at the electric utilities for making the final decisions of which customers to inspect as depicted in Figure 4.2.

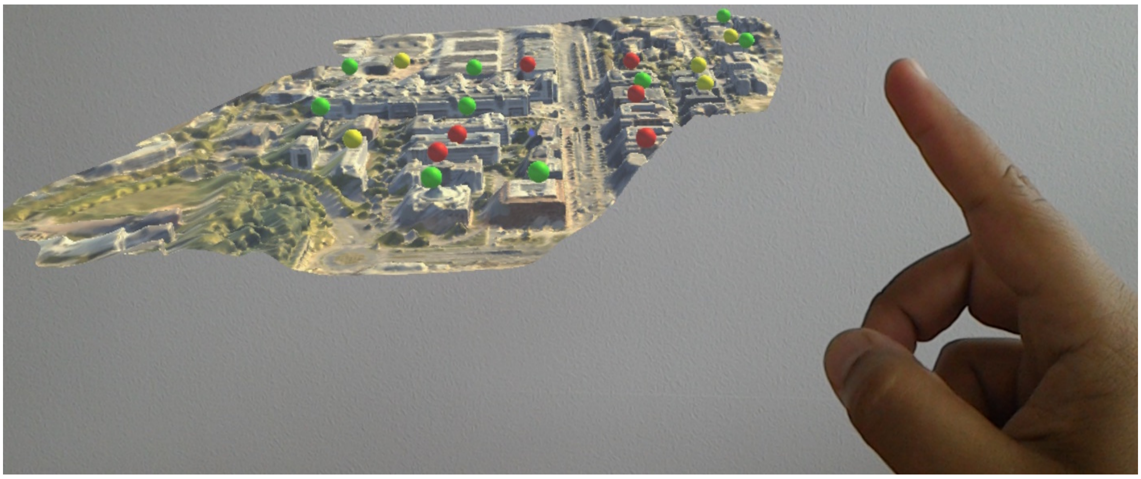


Figure 4.5: Gesture interactions with the spatial hologram allow to select customers as well as to zoom into or rotate holograms. We also provide a future yellow label that depicts a borderline case, which requires a manual check by domain experts.

4.2.1 Related Work

In the literature, different approaches for visualization of NTL are reported. In order to support the decision making, the visualization of the network topology on feeder level as well as load curves on transformer level is proposed in [1]. In addition, the density of NTL in a 2D map is visualized in [134]. For analytics in power grids as a whole, the need for novel and more powerful visualization techniques is argued in [165]. The proposed approaches include heat maps and risk maps. All methods for visualization of NTL proposed in the literature focus only on 2D representations.

We are currently undergoing a paradigm shift in data visualization from not only 2D to 3D, but rather to augmented reality using holographic projections [128]. This shift allows to better understand and experience data [81]. Users are not constrained to looking at data on a screen, as they can interact with the data, e.g. walking around holograms to get a better understanding of big data sets. This comes with the benefit of increased productivity as users can use their hands to turn and manipulate objects rather than getting distracted caused by a change of focus from the screen to the input devices such as keyboards or mice [103]. A number of successful applications of holographic projections have been described in the literature including guided assembly instructions [51] as well as a combination of different geographical information data sources in city management [103]. The literature also discusses the limitations of 3D visualizations, such as that users mistakenly may have greater confidence in the quality of the data [181].

4.2.2 Microsoft HoloLens

Mixed reality smart glasses such as the Microsoft HoloLens [31] depicted in Figure 4.6 allow users to combine holographic projections with the real world. The HoloLens offers their user a new perception of 3D models and, perhaps, can provide a new meaning to it. Visualization of data through holograms has found its application in many areas. In medicine, future doctors can study human anatomy by looking at a representation of the human body and navigate through muscles, organs and skeletons [163]. The HoloLens has the ability to perform the so-called holoportation. It allows to virtually place users to remote locations to see, hear and interact with others. Users can walk around holograms and interact with them using gaze, gestures or voice in the most natural way. Spatial sound allows hearing holograms even if they are behind the user, considering its position and direction of the sound. Spatial mapping features provide a real-world representation of surfaces, creating convincing holograms in augmented reality.

4.2.3 Architecture

We now describe the architecture of our visualization in the HoloLens. First, we create a 3D model of the map. A movie is recorded to capture the scene and all its objects from the different angles through Google Earth Pro. Afterwards, images are extracted in Windows Movie Maker from that movie at the best experimentally determined rate of 1 frame/sec. Then, those images are loaded in Blender, which in turn creates a 3D FBX model. This model is exported to Unity. Holographic effects are implemented through

Figure 4.6: Microsoft HoloLens[a].

[a] Source: `http://www.microsoft.com/en-us/hololens`

`HoloToolkit-Unity` [32]. Second, we load the customer data from the database. For example, we load the customers' locations as well as their past consumption profiles. Third, we use a machine learning system that predicts whether a customer causes NTL or not. We visualize each prediction at the location of the respective customer. The entire architecture is depicted in Figure 4.7.

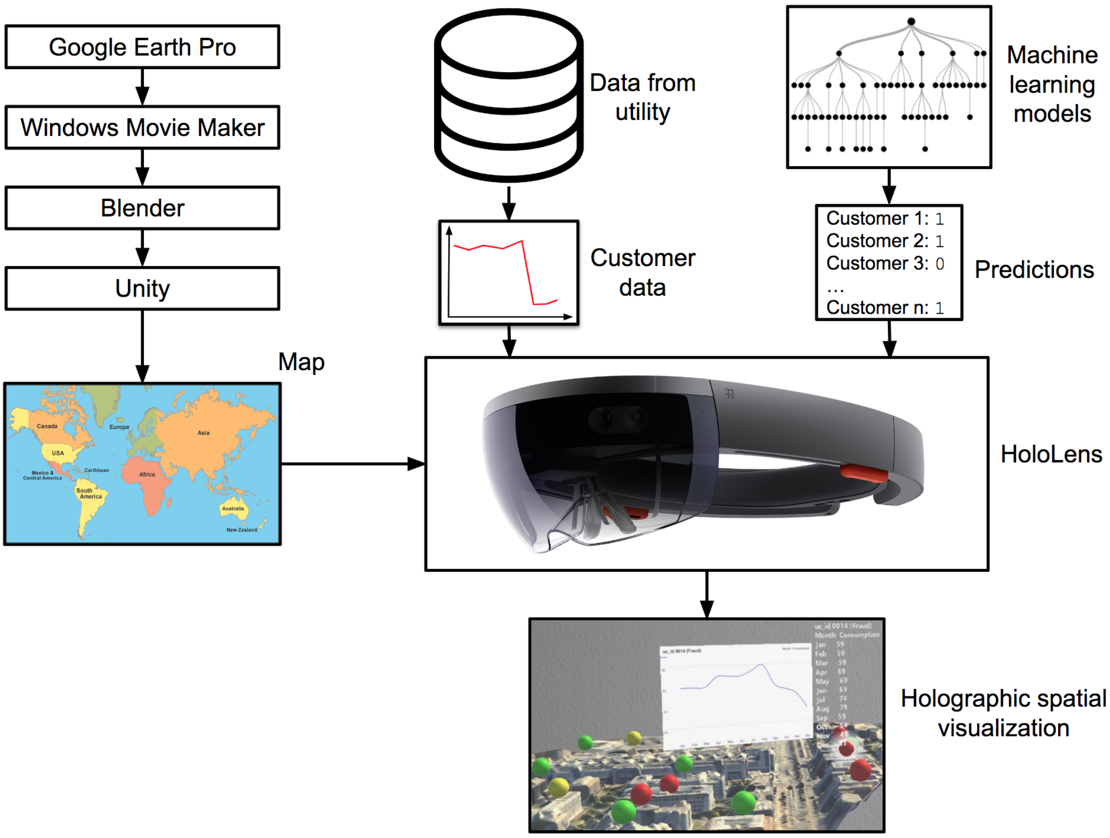


Figure 4.7: Visualization architecture: Map data, customer data and predictions are fed into the HoloLens in order to visualize a holographic spatial visualization.

4.2.4 Evaluation

This application is used by domain experts at the electric utilities and perceive that customers are classified as either regular (green) or irregular (red). Domain experts can walk around a spatial hologram and observe the data from different directions. Using their hands, they can also interact with the hologram in different ways, such as zooming into or rotating the hologram as depicted in Figure 4.8.

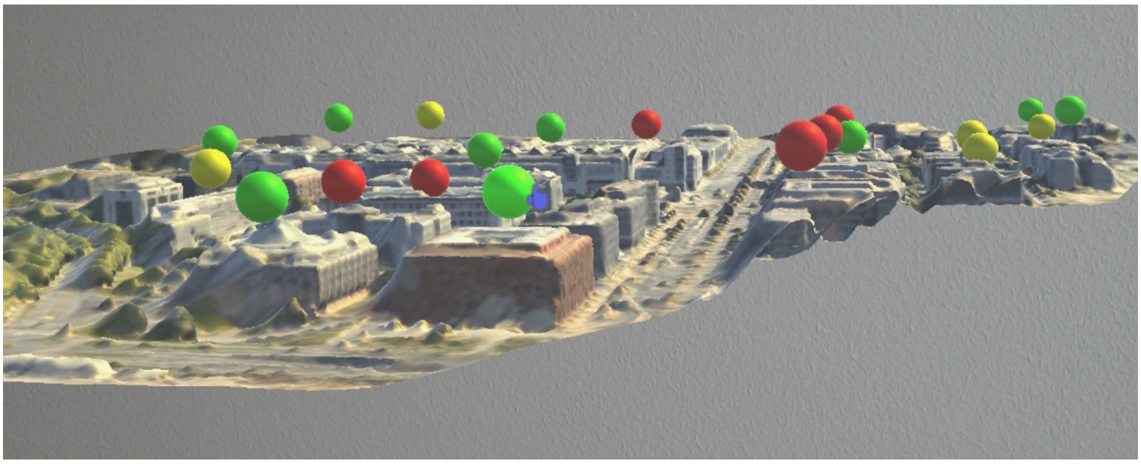
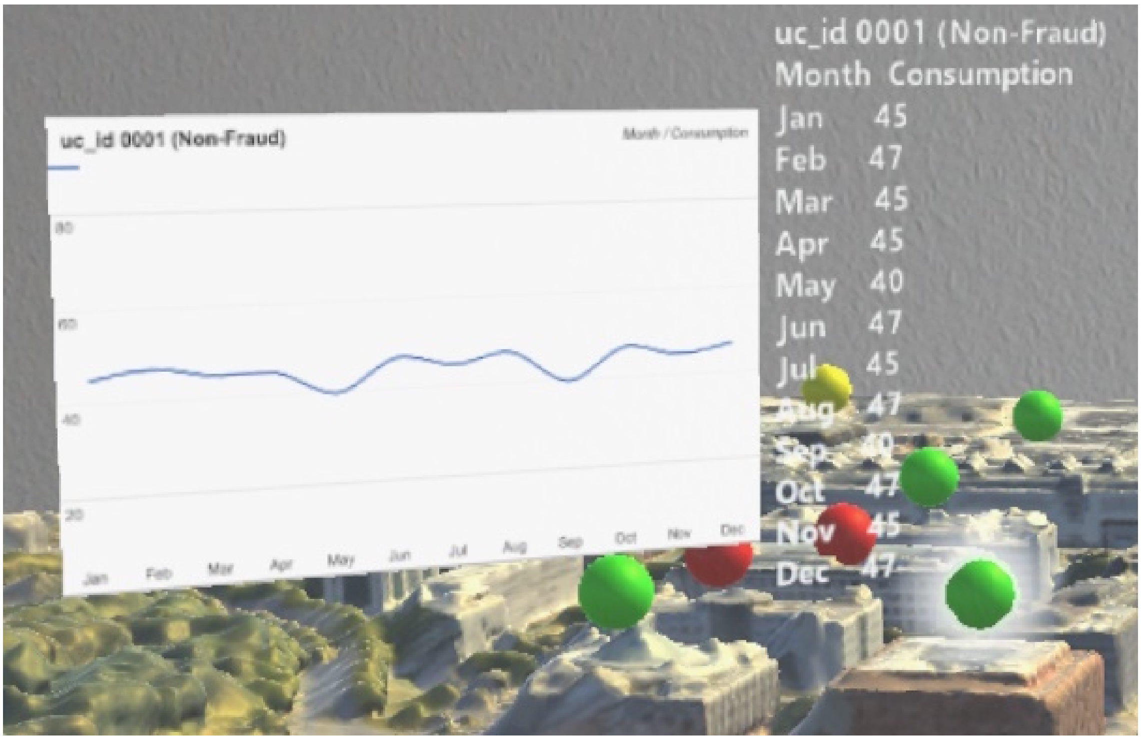


Figure 4.8: Zoomed and rotated view on the spatial hologram.

Domain experts can also learn more about a customer by tapping on it with their finger. The spatial hologram then also depicts the consumption profile of the respective customer over a selected period of time such as the previous 12 months. A customer with a predicted regular consumption profile is depicted in Figure 4.9. This customer's consumption has only changed very little in the last 12 months. As a consequence, the machine learning system classifies this customer as non-NTL (green). A customer with an irregular consumption profile is depicted in Figure 4.1. This customer's consumption has undergone a significant drop over the last few months. Therefore, the machine learning system classifies this customer as NTL (red). In both cases, domain experts can compare their observations with the prediction made by the machine learning system. If the prediction is not plausible, domain experts can choose not to follow the recommendation and therefore decide whether to inspect a customer. Our visualization allows domain experts to take the neighborhood of customers into account in order to decide which customers to inspect. Aside from the actual spatial visualization of satellite images of a neighborhood, domain experts can also visualize the consumption profile of neighbors as visualized in Figure 4.10 for comparing customers in order to decide whether to inspect a customer.



uc_id 0001 (Non-Fraud)

Month	Consumption
Jan	45
Feb	47
Mar	45
Apr	45
May	40
Jun	47
Jul	45
Aug	47
Sep	40
Oct	47
Nov	45
Dec	47

Figure 4.9: Detailed view of a customer depicted by a green dot predicted to have a regular power consumption pattern.

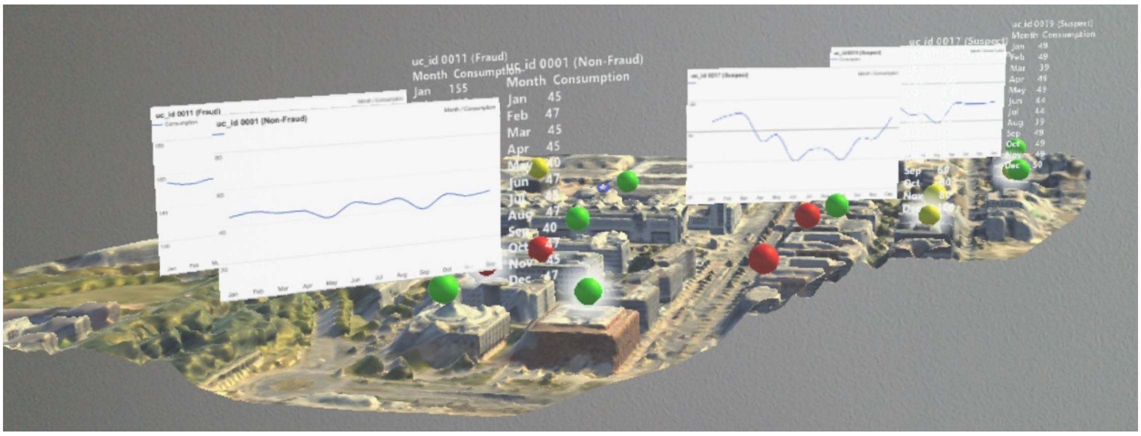


Figure 4.10: Multi-view on multiple customers' power consumption history.

4.2.5 Discussion

Our holographic spatial visualization of customers and their neighborhood comes with the benefit of increased productivity. We will show in Chapter 5 that the neighborhoods of customers yield significant information in order to decide whether a customer causes NTL or not. There are many interpretations of this fact. For example, fraudulent customers may share their knowledge with neighbors or there may be a correlation between electricity theft and the level of prosperity of a neighborhood. Our system allows to increase the ROI of inspections as well as to increase both the reliability and stability of the power grid by incorporating expert knowledge in the decision making process. Also, domain experts can use their hands to turn and manipulate objects rather than getting distracted by a change of focus from the screen to the input devices such as a keyboard or mouse.

4.3 Conclusions

In this chapter, we have proposed three models for NTL detection for large data sets of 100K customers: Boolean, fuzzy and machine learning using a Support Vector Machine. In contrast to other results reported in the literature, the optimized fuzzy and SVM models were assessed for varying NTL proportions on imbalanced real-world consumption data. Both have an AUC > 0.5 for all NTL proportions $> 0.1\%$ and significantly outperform simple Boolean or optimized fuzzy models. The improved models are about to be deployed in a CHOICE Technologies product. The issue of class imbalance will be discussed from a more generic and unified perspectives on biases in data sets in Chapter 6.

However, electric utilities are still keen to include expert knowledge in their decision making process when identifying the customers to be inspected. We have therefore proposed a novel system for detecting non-technical losses (NTL) for a real-world data set, depicted in Figure 4.2. In the first stage, the machine learning system learns to predict whether a customer causes NTL or not. Our machine learning system allows to detect NTL better than using an expert system. In the second stage, we put the prediction results into context by visualizing further data of the customers and their neighborhoods in a spatial hologram using a Microsoft HoloLens. Using this hologram, domain experts can then review and amend the suggestions of which customers to inspect. As a result, they can make the final decisions of which customers to inspect in order to increase the return on investment (ROI) of the limited number of inspections. We are confident that this approach will lead to an increase of both stability and reliability of power grids by making better use of the limited number of inspections as well as lead to a greater ROI of the limited number of inspections. We are also planning to evaluate our visualization approach through a future user study.

5

Comprehensive Learning from Customer Data

We have proposed a machine learning system based on average consumption features for detecting NTL in Chapter 4. In this chapter, we further address the challenge of feature description that we identified in Chapter 3.4.2. We thus take full advantage of the customer data in order to detect NTL better by exploring two different directions motivated below. Both approaches also address the challenge of scalability that we identified in Chapter 3.4.5.

The main idea of the first approach is to derive features that include information about the neighborhood. We show that the neighborhood of customers contains information about whether a customer may cause NTL or not. We analyze the statistical properties of these features and show why they are useful for NTL detection. By using information of the neighborhood, we can predict NTL better as there are geographic clusters of NTL among the customers. To the best of our knowledge, we are not aware of any previously published research that addressed this topic.

In the second approach, we propose a novel and flexible framework to compute a large number of domain-specific features and generic features from the noisy industrial consumption time series of customers for NTL detection. We retain the statistically meaningful features extracted from the noisy consumption data and optimize different classifiers to predict NTL better.

5.1 Information in Spatial Data

We use the same data set as in Chapter 4. About one third of the inspections found NTL. However, the models of this chapter must also work in other regions which have different NTL proportions. Therefore, the 14 samples each having 100K inspects results with the following NTL proportions are used: 1%, 2%, 3%, 4%, 5%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80% and 90%.

5.1.1 Features

We derive two new types of features from the customer data: features from the neighborhood as well as features from the master data of the customers. In addition, we also use the daily average consumption features derived in Equation 3.1 that we have previously used in Chapter 4. We also analyze the statistical properties of features derived from the neighborhood and explain why they are useful for NTL detection.

Neighborhood

Certain areas are more likely to cause NTL than others. Therefore, features based on the neighborhood are interesting in order to improve predictions. The data includes invalid coordinates of customers, such as coordinates in the ocean. For this, all customers outside a deviation from the mean coordinates are removed. We empirically found that removing the 1K customers that are not within five standard deviations from the mean coordinates worked the best. The bounding box around the remaining valid coordinates is about 200 km along the longitude and about 500 km along the latitude. Therefore, the bounding box has an area of approximately 100,000 km^2. This bounding box is split into a grid along the longitude and latitude.

In each cell$_{ij}$, the proportion of inspected customers and the proportion of NTL found among the inspected customers are computed:

$$\text{inspected\_ratio}_{ij} = \frac{\#\text{inspected}_{ij}}{\#\text{customers}_{ij}}, \tag{5.1}$$

$$\text{NTL\_ratio}_{ij} = \frac{\#\text{NTL}_{ij}}{\#\text{inspected}_{ij}}. \tag{5.2}$$

An example cell is provided in Figure 5.1.

The grid sizes used are 50, 100, 200 and 400 cells along the longitude and latitude, respectively. For each grid size, both features are assigned to each customer registered in the respective cell. The area per cell is depicted in Table 5.1 for each grid size.

As four grid sizes are used, a total of $4 \times 2 = 8$ neighborhood features are computed per customer. For both classes, the distributions of the values of both features for these four

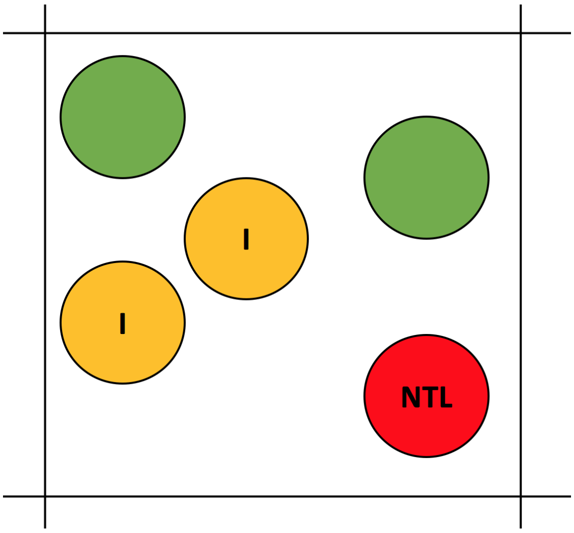


Figure 5.1: Example cell with 5 customers, 3 out of 5 were inspected (I) and 1 out of 3 inspected customers caused NTL.

Table 5.1: Area per cell for all grid sizes.

Grid Size	Area per Cell [km^2]
50 × 50	40
100 × 100	10
200 × 200	2.5
400 × 400	0.625

grid sizes are depicted in Figure 5.2 for a NTL proportion of 20% and in Figure 5.3 for a balanced NTL proportion.

The distributions of both neighborhood features represent the prior distributions of a Bayesian approach. However, none of the distributions is Gaussian, and it is therefore interesting to study how their properties change for varying NTL proportions of the data set and how they allow to separate between no NTL found and NTL found.

The mean of each feature distribution is depicted in Figure 5.4. The means of the inspected ratio distributions are expected to be around 0.14 because there are 700K customers and each NTL proportion file contains 100K inspections. However, the means slightly decrease for greater NTL proportions for customers for which no NTL was found and slightly increase for customers for which NTL was found. We have not found any cause of this in our experiments, however, we believe that this is caused by the sampling of the data. However, this helps to separate both classes. The means of the NTL found ratio distributions are approximately the NTL proportion as expected. For all grid sizes, the distributions of

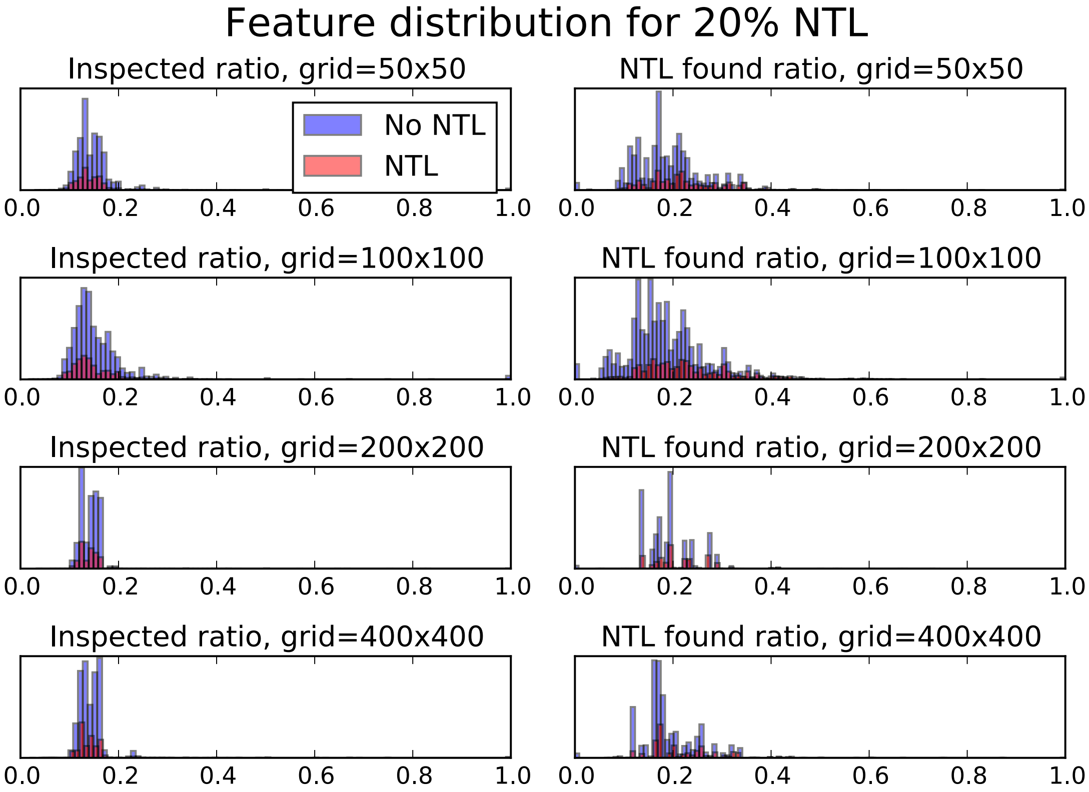


Figure 5.2: Distributions of both neighborhood features for varying grid sizes for 20% NTL.

means are approximately the same for the inspected ratio and NTL found ratio features, respectively.

The variance of each feature distribution is depicted in Figure 5.5. For the inspected ratio feature, we see that the variance is lower for the customers for which NTL was found than for those for which no NTL was found. There is only an exception for the grid size of 100 for NTL proportions $> 70\%$. The variance of the NTL found ratio feature is greater for the customers for which NTL was found than those for which no NTL was found for NTL proportions $< 50\%$ and then flips around 50% for all grid sizes. This demonstrates an inverse relationship between the distributions of variances of both features for NTL proportions $< 50\%$. For both features, the variances are in different ranges for each grid sizes, which helps to separate between both classes.

Skewness is the extent to which the data are not symmetrical [40]. It is the third standardized moment, defined as:

$$\gamma_1 = \mathrm{E}\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] = \frac{\mu_3}{\sigma^3} = \frac{\mathrm{E}\left[(X-\mu)^3\right]}{(\mathrm{E}\left[(X-\mu)^2\right])^{3/2}}, \tag{5.3}$$

where μ_3 is the third central moment, μ is the mean, σ is the standard deviation and E is

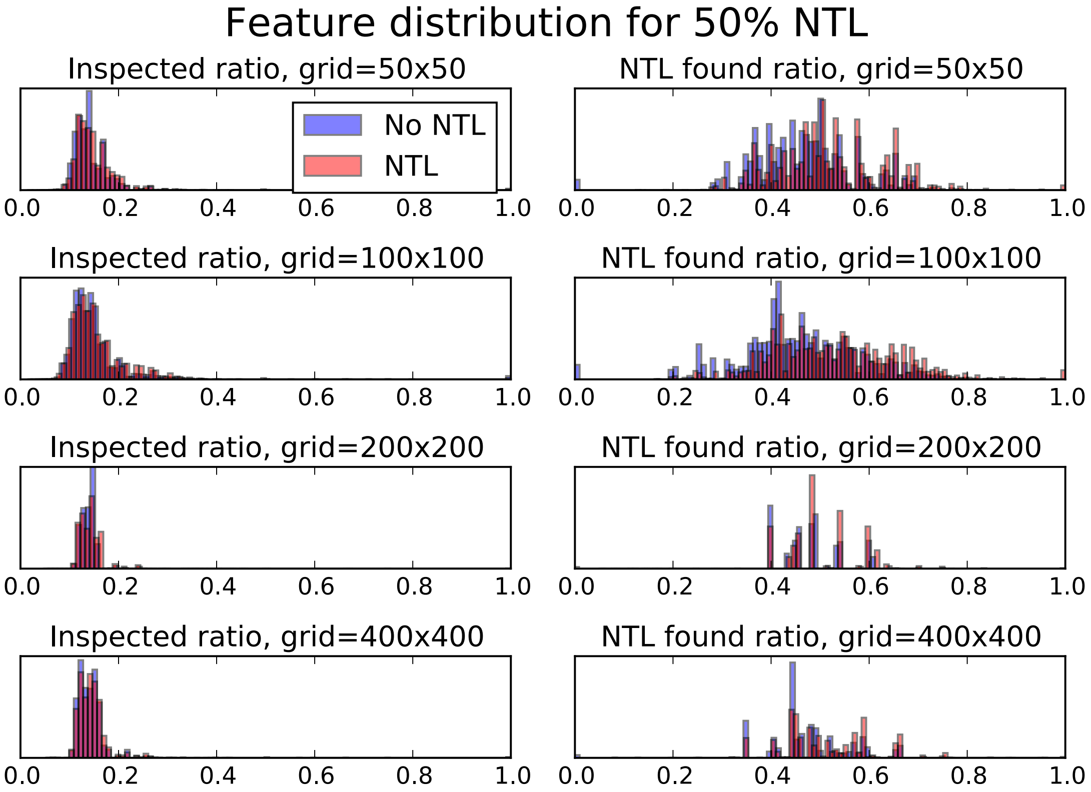


Figure 5.3: Distributions of both neighborhood features for varying grid sizes for 50% NTL. Example: the red NTL peak around 0.5 in the NTL found ratio, grid=200x200 plot represents a type of favela neighborhood, in which every second customer causes NTL.

the expectation operator. Positively skewed data have a tail that points to the right. In contrast, negatively skewed data have a tail that points to the left. The skewness of each feature distribution is depicted in Figure 5.6.

All inspected ratio distributions are positively skewed. This skewness means that there are more grid cells with very high inspected ratios than cells with very low inspection ratios. There is no significant difference between both classes for most NTL proportions and therefore this property does not help much to separate between both. All NTL found ratio distributions are positively skewed for NTL proportions $\leq 50\%$. For NTL proportions $> 50\%$, the distributions are negatively skewed for the non-NTL class. The change of sign in the skewness distributions for samples with low NTL proportions shows the existence of clusters of low NTL of different sizes. The skewness of this feature is generally greater for the NTL class than the non-NTL class for all grid sizes, which allows to separate both classes better.

Kurtosis indicates how the peak and tails of a distribution differ from the normal distri-

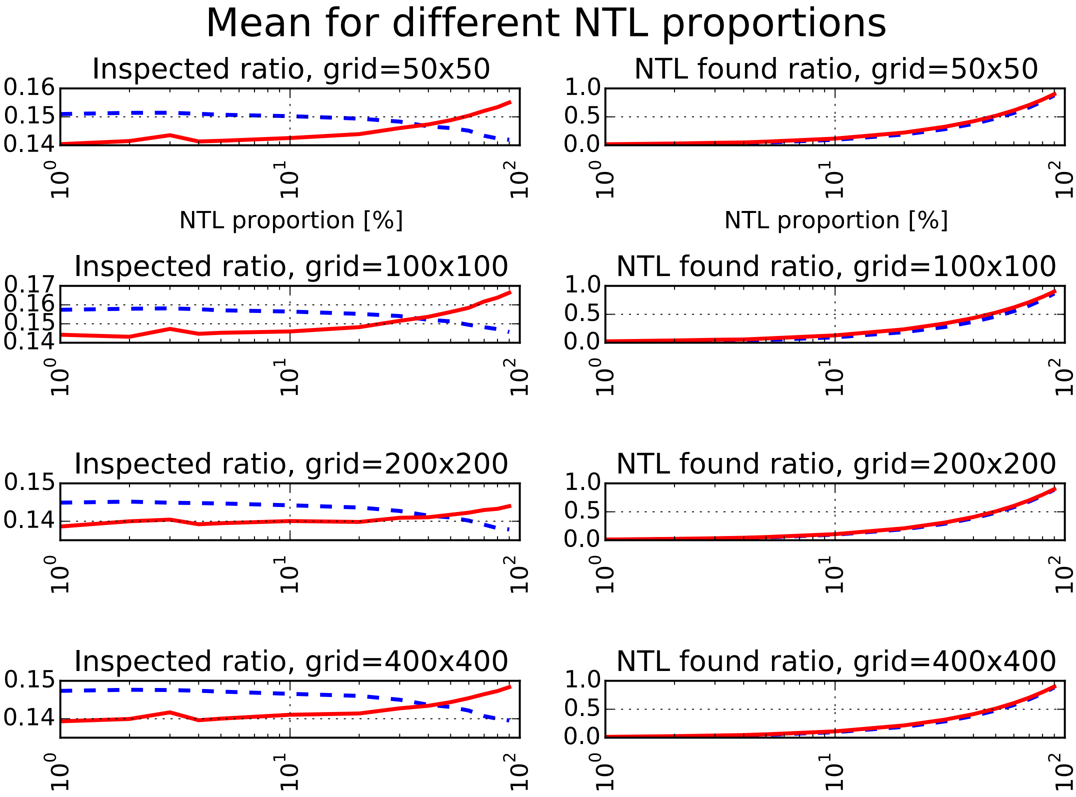


Figure 5.4: Mean of each feature distribution for different NTL proportions. Legend: the blue dashed curve represents the non-NTL class and the red solid curve represents the NTL class.

bution [40]. It is the fourth standardized moment, defined as:

$$\mathrm{Kurt}[X] = \frac{\mu_4}{\sigma^4} = \frac{\mathrm{E}[(X-\mu)^4]}{(\mathrm{E}[(X-\mu)^2])^2} - 3, \tag{5.4}$$

where μ_4 is the fourth moment about the mean and σ is the standard deviation. A distribution with a positive kurtosis value has heavier tails and a sharper peak than the normal distribution. In contrast, a distribution with a negative kurtosis value indicates that the distribution has lighter tails and a flatter peak than the normal distribution. The kurtosis of each feature distribution is depicted in Figure 5.7.

The kurtosis values of all distributions of both features are positive and therefore have sharper peaks than the normal distribution. For the inspection ratio features, the kurtosis is greater for the NTL class for most NTL proportions, meaning these features are less Gaussian than for the NTL class, which helps to separate both classes. The same applies to the NTL found ratio feature for NTL proportions $< 50\%$.

Overall, the plots of variance, skewness and kurtosis of both classes show that for both

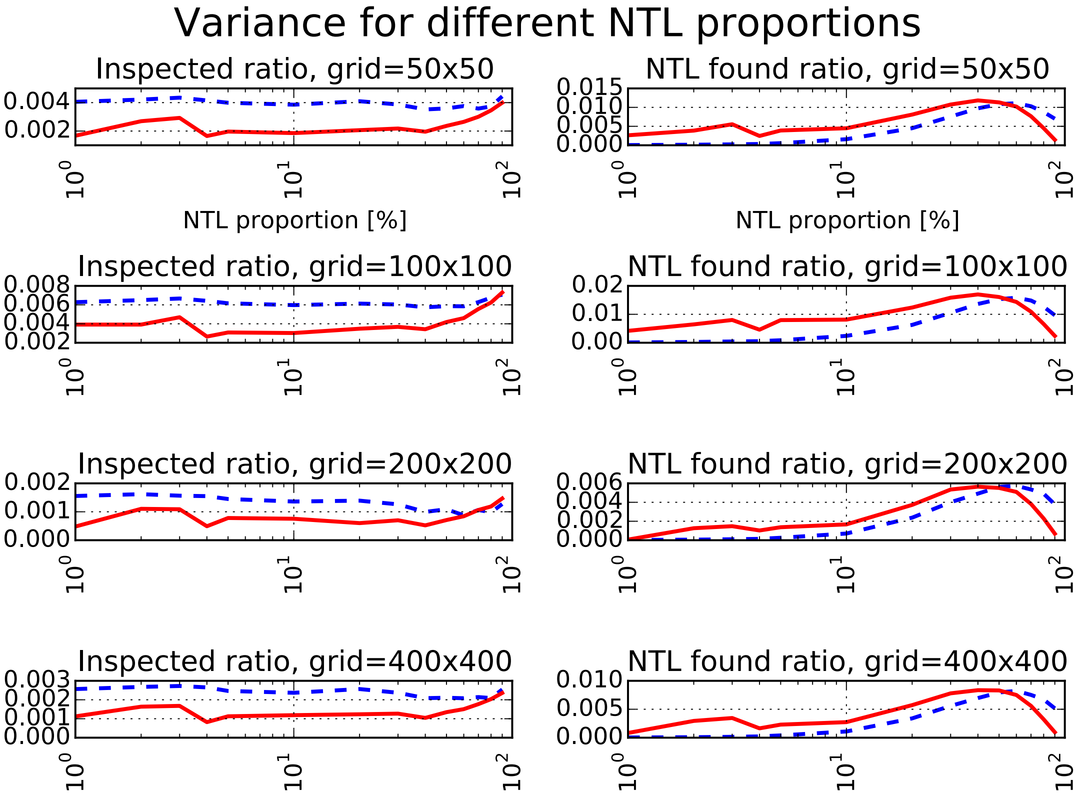


Figure 5.5: Variance of each feature distribution for different NTL proportions. Legend: the blue dashed curve represents the non-NTL class and the red solid curve represents the NTL class.

features the values of the distributions for the different grid sizes have different ranges. This is helpful in order to discriminate between NTL and no NTL.

Categorial Master Data

In addition, more information about the customer should be considered in the prediction. The categorial master data available for each customer is summarized in Table 5.2. Each feature is converted to one-hot coding. Therefore, there are $8 + 3 + 3 + 2 = 16$ binary features per customer.

In order to reduce overfitting, only representative binary features are kept. These could be found using the principal component analysis (PCA) [130]. However, PCA is not able to handle noise in the data well. Since this real data set is noisy, PCA is not used for the reduction of the binary features. Instead the dimensionality reduction approach is as follows: All features that are either one or zero in more than $p \times 100\%$ of each proportion sample are removed. These binary features are Bernoulli random variables,

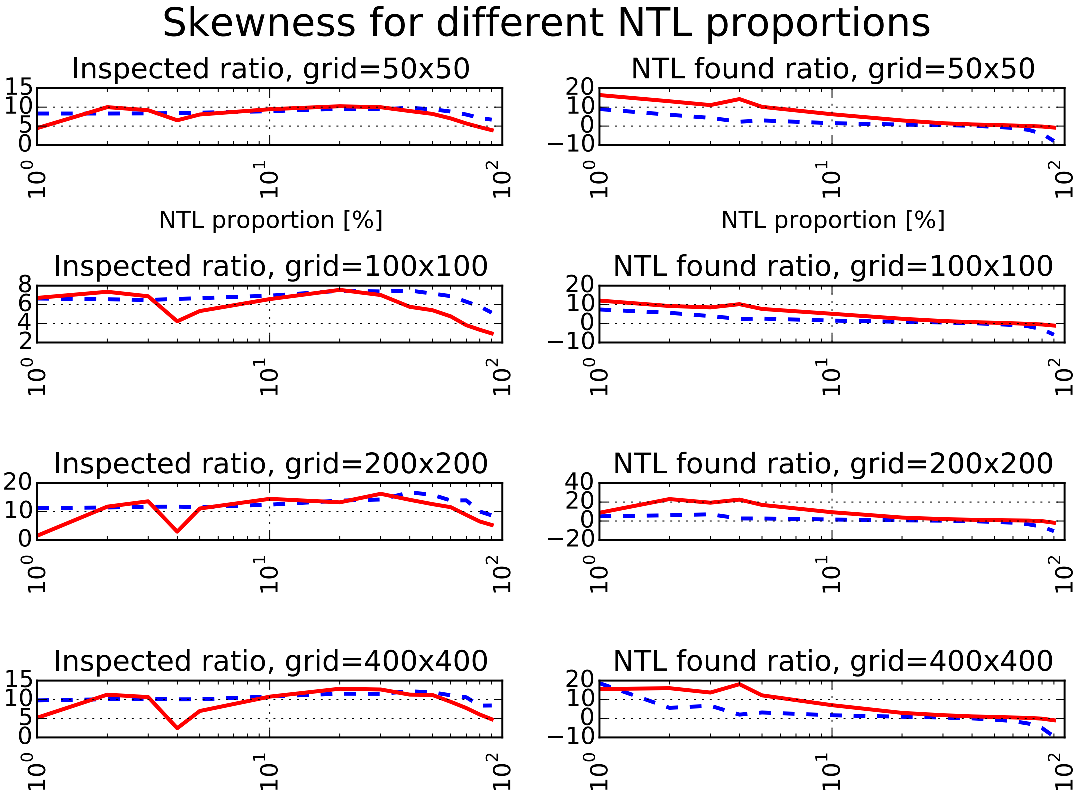


Figure 5.6: Skewness of each feature distribution for different NTL proportions. Legend: the blue dashed curve represents the non-NTL class and the red solid curve represents the NTL class.

Table 5.2: Available master data.

Name	Possible Values
Class	Residential, commercial, industrial, public illumination, rural, public, public service, power generation infrastructure
Contract status	Active, suspended, inactive
Number of wires	1, 2, 3
Voltage	>2.3kV, ≤2.3kV

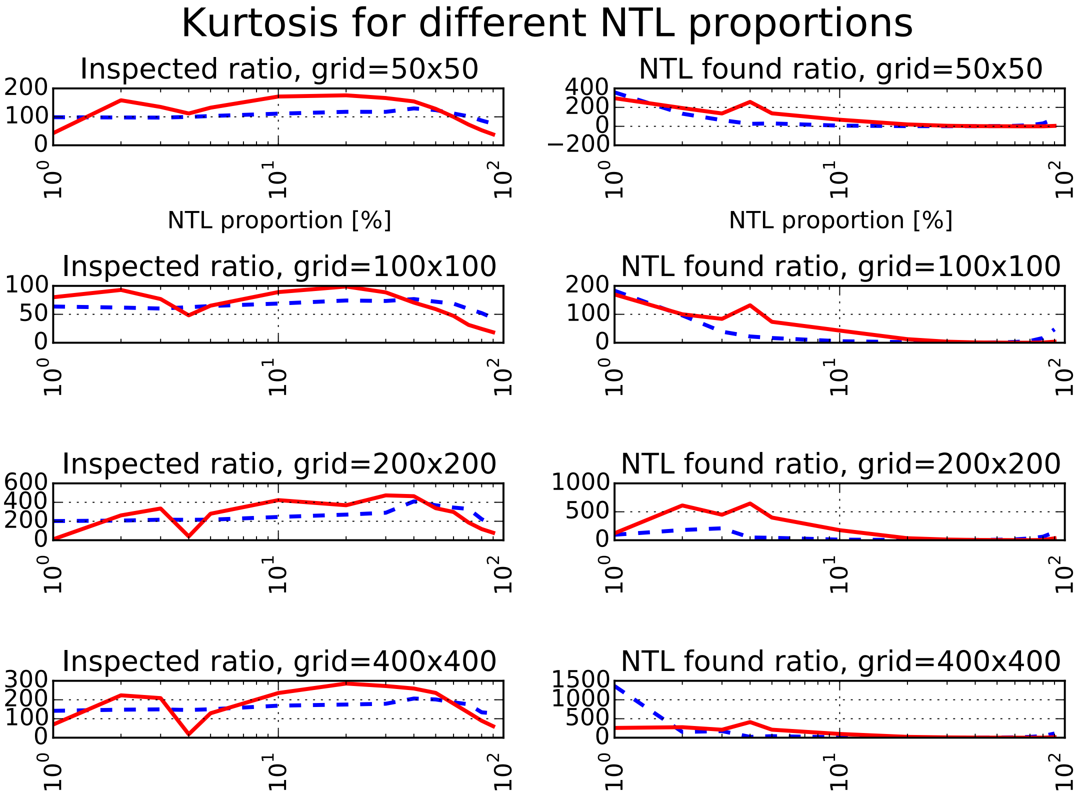


Figure 5.7: Kurtosis of each feature distribution for different NTL proportions. Legend: the blue dashed curve represents the non-NTL class and the red solid curve represents the NTL class.

and the variance of such variables is given by:

$$\mathrm{Var}[X] = p(1 - p). \tag{5.5}$$

For the following experiments in this chapter, $p = 0.9$ was experimentally determined to work the best.

Final Feature Set

For each NTL proportion, the feature matrix has at least 20 features, which are the 8 neighborhood features combined with the 12 daily average consumption features. Depending on the distribution of customers in each NTL proportion, up to 16 binary master data features are added. However, only a fraction of them is expressive enough to improve the prediction results. The number of retained and number of total features per NTL proportion sample are summarized in Table 5.3.

In order to optimize the training, each of the 8 neighborhood features and 12 daily

Table 5.3: Number of features used per NTL proportion.

NTL Proportion	#Retained Binary Features	#Total Features
1% - 10%	5	25
30% - 70%	4	24
20%, 80%, 90%	6	26

average consumption features is normalized:

$$x_j\prime = \frac{x_j - \bar{x}_j}{\sigma_j}. \tag{5.6}$$

This normalization makes the values of each future in the data have zero mean and unit variance. This allows to reduce the impact of features with a broad range of values. As an outcome, each feature contributes approximately proportionally to the classification.

5.1.2 Evaluation

In this chapter, we train logistic regression, k-nearest neighbors, support vector machine and random forest classifiers.

Implementation

All computations were run on a server with 24 cores and 128 GB of RAM. The entire code was written in Python. The neighborhood features were computed using `Spark` [180]. For all experiments, `scikit-learn` [131] was used, which allows to distribute the training and evaluation of each of the four classifiers among all cores.

Experimental Setup

For every NTL proportion, the data set is split into training, validation and test sets with a ratio of 80%, 10% and 10%, respectively. Each of the four models is trained using 10-fold cross-validation. For each of the four models the trained classifier that performed the best on the validation set in any of the 10 folds is selected and tested on the test set to report the test AUC. This methodology is related to Chapter 4. For each of the four models, the following parameter values were determined empirically as a compromise between expressiveness, generalization of models and training time. For logistic regression and SVM, the inverse regularization factor C is set to 1.0. $K = 100$ neighbors are visited in kNN. The random forest consists of 1K trees. Running the following experiments including cross-validation takes about 4 hours on this computing infrastructure.

Results

For different NTL proportions, the test AUC of the logistic regression (LR) classifiers is depicted in Figure 5.8. Using only the time series daily average consumption features of the last 12 months results in a classifier that performs like chance for most NTL proportions. It only performs better than chance for NTL proportions of 50%-80% with a maximum AUC of 0.525 for a NTL proportion of 50%. However, by adding the neighborhood and selected categorial features, the classifier performs noticeably better than chance for all NTL proportions and significantly better than time series features only for NTL proportions of 30%-70%.

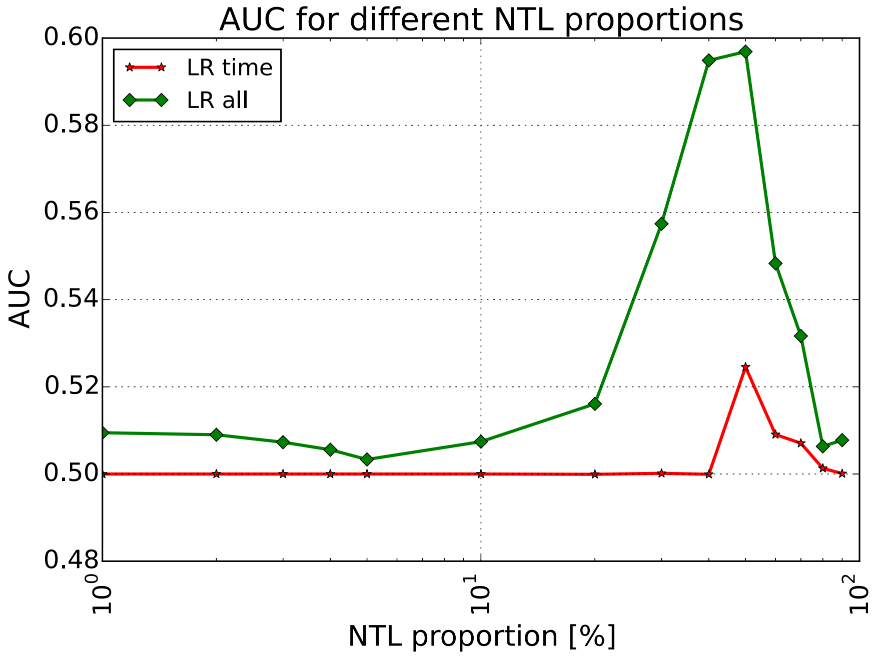


Figure 5.8: Test performance of logistic regression classifier on different NTL proportions for time series and all features.

Similar experiments are run for the kNN, SVM and random forest (RF) classifiers and summarized in Table 5.4. It can be observed that the extra features help all classifiers to maximize the overall AUC scores and that the classifiers perform noticeably better than chance for more NTL proportions.

The LR, kNN and SVM classifiers perform the best for a balanced data set of 50%. The RF classifiers perform the best for 60% and 40% using only the time series or all features, respectively. However, it must be noted that for 50%, both RF classifiers perform close to the optimal AUC scores achieved. This is most likely due to the ensemble, which allows to better adopt to variations in the data set. The four models that performed the best on all features are then tested on all proportions. The results are summarized in Table 5.5 and visualized in Figure 5.9.

Table 5.4: Comparison of classifiers trained on time series and all features. $_t$ denotes that only the time series is used in the models. $_a$ denotes that all features are used: time series, neighborhood features and selected master data. Best proportion per model in **bold**.

NTL Proportion	LR_t	LR_a	kNN_t	kNN_a	SVM_t	SVM_a	RF_t	RF_a
1%	0.5	0.51	0.5	0.5	0.5	0.5	0.5	0.505
2%	0.5	0.509	0.5	0.5	0.5	0.5	0.5	0.505
3%	0.5	0.507	0.5	0.5	0.5	0.505	0.5	0.511
4%	0.5	0.506	0.5	0.5	0.5	0.503	0.502	0.509
5%	0.5	0.503	0.5	0.5	0.5	0.504	0.5	0.511
10%	0.5	0.507	0.504	0.5	0.5	0.505	0.504	0.519
20%	0.5	0.516	0.523	0.506	0.5	0.511	0.509	0.539
30%	0.5	0.557	0.53	0.549	0.5	0.552	0.535	0.578
40%	0.5	0.595	0.546	0.587	0.5	0.592	0.55	**0.619**
50%	**0.525**	**0.597**	**0.57**	**0.596**	**0.521**	**0.6**	0.572	0.618
60%	0.509	0.548	0.545	0.556	0.509	0.546	**0.579**	0.582
70%	0.507	0.532	0.526	0.53	0.507	0.529	0.55	0.553
80%	0.501	0.506	0.508	0.505	0.502	0.51	0.527	0.514
90%	0.5	0.508	0.5	0.5	0.502	0.506	0.507	0.506

The RF classifier achieves the greatest AUC throughout the experiments of 0.628 for a NTL proportion of 3% and achieves the best AUC among all classifiers for 7 of the 14 classifiers. The SVM performs the best on 4 proportions, the LR performs performs the best on 2 proportions. Both classifiers perform similarly well on the NTL proportion of 10%. The kNN classifier only performs the best on one proportion. Even though the RF achieved the maximum AUC, it also has the lowest AUC throughout the experiments. Furthermore, it has the greatest standard deviation of all classifiers.

Table 5.5: Performance of optimized models on all NTL proportions. Model XY% stands for a model that was trained on a NTL proportion of XY% and tested on all proportions. Best model per proportion in **bold**.

NTL Proportion	LR 50%	kNN 50%	SVM 50%	RF 40%
1%	0.601	0.585	0.602	**0.62**
2%	0.611	**0.614**	0.611	0.606
3%	0.596	0.566	0.598	**0.628**
4%	0.593	0.587	**0.604**	0.565
5%	0.588	0.581	0.588	**0.596**
10%	**0.585**	0.583	**0.585**	0.561
20%	0.585	0.576	0.583	**0.6**
30%	0.596	0.581	0.594	**0.603**
40%	0.598	0.586	0.601	**0.619**
50%	0.597	0.596	**0.6**	0.59
60%	**0.6**	0.591	0.598	0.598
70%	0.596	0.595	0.597	**0.598**
80%	**0.606**	0.591	0.588	0.583
90%	0.591	0.596	**0.605**	0.596
Max	0.611	0.614	0.611	0.628
Min	0.585	0.566	0.583	0.561
$\overline{AUC}$	0.5959	0.5877	0.5967	0.5973
σ_{AUC}	0.0071	0.0108	0.0079	0.0183

5.1.3 Discussion

Overall, all four classifiers perform in the same regime, as their mean AUC scores over all NTL proportions are very close. This observation is often made in machine learning, as the actual algorithm is less important, but having more and representative data is generally considered to be more important [11]. This can also be justified by the "no free lunch

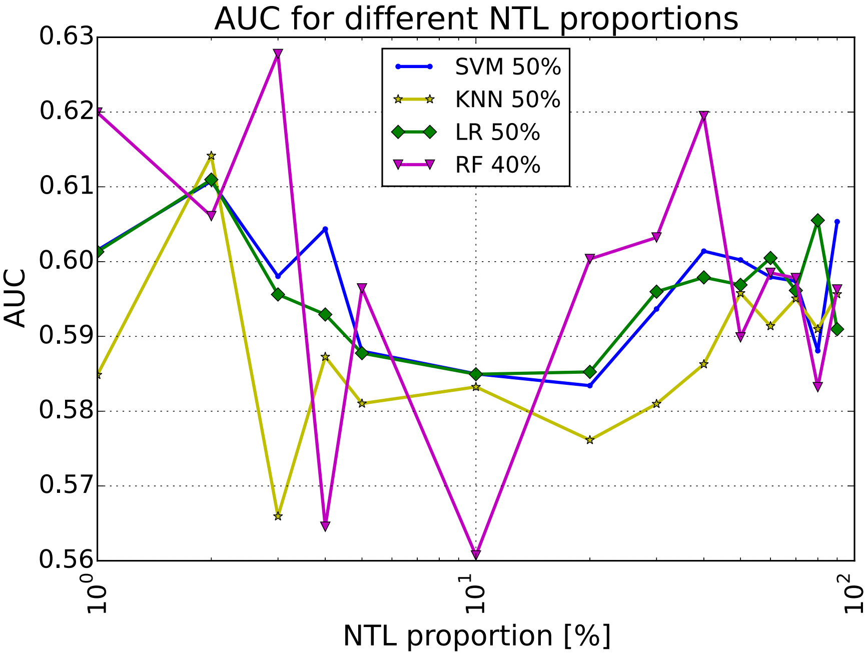


Figure 5.9: Test performance of optimized classifiers on different NTL proportions.

theorem", which states that no learning algorithm is generally better than others [176]. We only used the features derived from the consumption time series in Chapter 4. Using also the neighborhood information and categorial customer master data, each of the four classifiers consistently performs better than the classifiers in our previous work for all NTL proportions.

5.2 Information in Consumption Records

The data used in this NTL detection method comes from an electric utility in Brazil and consists of 3.6M customers. The data contains 820K inspection results, such as inspection date, presence of fraud or irregularity, type of NTL and inspection notes. 620K customers have been inspected at least once and the remaining ∼3M customers have never been inspected. Third, there are 195M meter readings from 2011 to 2016 such as consumption in kWh, date of meter reading and number of days between meter readings. From the 620K customers for which an inspection result is available, only the most recent inspection result is used in the following experiments.

The available data per customer m is a complete time series of monthly meter readings of electricity consumption in kWh over the last N months before the most recent inspection, described as follows:

$$C^{(m)} = [C_0^{(m)}, ..., C_{N-1}^{(m)}], \tag{5.7}$$

where $C_{N-1}^{(m)}$ is the most recent meter reading before the inspection. For greater N, less customers with a complete time series are available. In contrast, for smaller N, less information per customer is available.

5.2.1 Features

In this section, we describe the features that we compute from a customer's consumption time series $C^{(m)}$ for the detection of NTL.

Difference Features

The intra year difference

$$\text{intra\_year}_d^{(m)} = C_d^{(m)} - C_{d-K}^{(m)}, \tag{5.8}$$

for $K = 12$, is the change of consumption to the consumption in the same month of the previous year. In total, there are $N - 12$ intra year difference features.

The intra year seasonal difference

$$\text{intra\_year\_seasonal}_d^{(m)} = C_d^{(m)} - \frac{1}{3} \times \sum_{k=d-K-1}^{d-K+1} C_k^{(m)}, \tag{5.9}$$

for $K = 12$, is the change of consumption to the mean of the same season in the previous year. In total, there are $N - 13$ intra year seasonal difference features.

The fixed interval

$$\text{fixed\_interval}_d^{(m)} = C_d^{(m)} - \frac{1}{K} \times \sum_{k=d-K}^{d-1} C_k^{(m)}, \tag{5.10}$$

for $K \in \{3, 6, 12\}$, is the change of consumption to the mean consumption in a period of time directly before a meter reading. In total, there are $3 \times (N - 12)$ fixed interval features. These features are inspired by our previous work [112], in which we have proposed them only for the most recent meter reading. Instead, we now compute these features for the entire consumption time series.

Daily Averages

We again use the daily average features derived in Equation 3.1.

Generic Time Series Features

In order to catch more characteristics of the consumption time series, we compute 222 generic time series features from it, comprising:

- Summary statistics, such as maximum, variance or kurtosis.
- Characteristics from sample distribution, such as absolute energy, whether a distribution is symmetric or the number of data points above the median.
- Observed dynamics, such as fast Fourier transformation coefficients, autocorrelation lags or mean value of the second derivative.

The full list of features is provided in [29].

5.2.2 Feature Selection

In total, 304 features are computed. In the subsequent learning phase, only the meaningful features should be used. One common dimensionality reduction method is the principal component analysis (PCA). However, time series, and in particular real-world data sets, are noisy, which can lead to poor performance of PCA [55]. It is for that reason that we do not use PCA for the feature selection. Instead, we employ hypothesis tests to the features in order to retain the ones that are statistically relevant [30]. These tests are based on the assumption that a feature x_k is meaningful for the prediction of the binary label vector y if x_k and y are not statistically independent [139]. For binary features, we use Fisher's exact test [54]. In contrast, for continuous features, we use the Kolmogorov-Smirnov test [107].

5.2.3 Evaluation

As in our previous experiments, we use the AUC metric for all experiments.

Experimental Setup

We experimentally determined $N = 24$ months to work the best for the following experiments. Using $N = 24$ allows the consumption data to reflect seasonality in the experiments. As a consequence, $M = 150,700$ customers are retained for the experiments. This data set is imbalanced: 100,471 have a negative label (non-NTL), whereas 50,229 have a positive one (NTL). Therefore, 33.33% of the customers used in the following experiments have been found to cause NTL.

We train the decision tree (DT), random forest (RF), gradient-boosted tree (GBT) and linear support vector machine (LSVM) classifiers as follows:

- Handling class imbalance: We handle the class imbalance during training by assigning class weights to the examples of both classes in the training set:

$$w_0 = \frac{\#\,\text{examples}}{\#\,\text{examples}_{C=0}}, \tag{5.11}$$

$$w_1 = \frac{\#\,\text{examples}}{\#\,\text{examples}_{C=1}}. \tag{5.12}$$

- Performing model selection: We want to find the model which is able to distinguish between NTL and non-NTL customers the best. For this, we optimize various parameters for every classifier. The complete list of parameters and considered values per classifier is depicted in Table 5.6. We use randomized grid search, which samples from the joint distribution of model parameters. In contrast to grid search, randomized grid search does not try out all parameter values. We use 100 sampled models in every model selection.

- Handling overfitting: We also employ model selection that splits the data set into $k = 10$ folds. This leads to a more reliable model for NTL detection. The AUC reported per model is the average of the AUCs of the k test sets.

Table 5.6: Model parameters.

Parameter	Values	DT	RF	GBT	LSVM
Learning rate	$[0.0001, 1]$ (log space)			✓	
Loss function	{AdaBoost, deviance}			✓	
Max. number of leaves	$[2, 1000)$	✓	✓	✓	
Max. number of levels	$[1, 50)$	✓	✓	✓	
Measure of the purity of a split	{entropy, gini}	✓	✓		
Min. number of samples at leaf	$[1, 1000)$	✓	✓	✓	
Min. number of samples to split node	$[2, 50)$	✓	✓	✓	
Number of estimators	20		✓	✓	
L_2 regularization	[0.001, 10] (log space)				✓

Implementation

All computations were run on a server with 80 cores and 128 GB of RAM. The entire code was implemented in Python using `scikit-learn` [131] for machine learning. `scikit-learn` allows to distribute the training of the numerous classifiers among all cores. Using this infrastructure, the extraction of features took 6 hours. The feature selection took only 1 minute. The extensive model selection of classifiers took 4 days. In deployment, the training of classifiers will perform significantly faster as the extensive model selection needs to be performed only when a new data set is used. We have also noticed that about 90% of the training time was spent on the gradient-boosted tree. Therefore, a significant speedup can be achieved in deployment when skipping the training of this classifier.

Feature Selection

We first compute the features defined previously and then perform the feature selection. In summary, there are three types of features: (1) generic time series (GTS) features, (2) daily average features (AVG) and (3) difference features (DIF) composed of fixed interval, intra year difference and intra year seasonal difference features. The numbers of features before and after selection are depicted in Table 5.7.

Table 5.7: Number of features before and after selection.

Name	#Features	#Retained Features
Daily average (AVG)	23	18
Fixed interval	36	34
Generic time series (GTS)	222	162
Intra year difference	12	12
Intra year seasonal difference	11	11
Total	304	237

In total, 237 out of the 304 features are retained. The relevance of our hand-crafted difference features is confirmed: all intra year difference and intra year seasonal difference features are retained. In addition, 34 out of 36 fixed interval features are retained. The 2 features are not retained for $K = 3$, which is most likely due to the too short span of time they reflect. As a matter of fact, daily average features are widely used in the research literature on NTL detection. However, only 18 out of 23 daily average consumption features (i.e. 78%) are retained. The 5 daily average consumption features that are not retained are the ones for the first - i.e. the oldest - 6 months of the 24-month window. The statistical feature check leads to the conclusion that this type of feature is only useful for about 1.5 years of our data for NTL detection. In addition, 73% of the generic time series features are retained after the statistical relevance check. As these features are generic and not particularly made for NTL detection, it is to no surprise that the retention rate for these features is the lowest.

Classification Results

We train the four classifiers on each of the GTS, AVG and DIF feature sets as well as on all combinations thereof. The test performance of the best model per experiment returned by the model selection is depicted in Tables 5.8, 5.9 and 5.10.

The best test AUC of 0.65977 is achieved for training the random forest classifier on the

Table 5.8: Test performance of classifiers on features from measured consumption data. Test AUC for combinations of decision tree (DT), random forest (RF), gradient-boosted tree (GBT) and linear support vector machine (LSVM) classifiers trained on sets composed of general time series (GTS), daily average (AVG) and difference (DIF) features. Per combination of classifier and feature set, the better result on either a full feature set (X_{all}) or retained feature set (X_{ret}) is highlighted . ^{c} denotes the best classifier per feature set.

Classifier	GTS		AVG		DIF	
	X_{all}	X_{ret}	X_{all}	X_{ret}	X_{all}	X_{ret}
DT	0.64544	0.64625	0.64037	0.63985	0.63730	0.63792
RF	0.65665^c	0.65726^c	0.65083^c	0.65248^c	0.65529^c	0.65459^c
GBT	0.63149	0.63125	0.63234	0.63186	0.62869	0.63019
LSVM	0.63696	0.63656	0.54982	0.54933	0.55749	0.55843

Table 5.9: Test performance of classifiers on combined featured sets from measured consumption data. ^{f} denotes the best feature set per classifier.

Classifier	GTS+AVG		GTS+DIF		AVG+DIF	
	X_{all}	X_{ret}	X_{all}	X_{ret}	X_{all}	X_{ret}
DT	0.64712	0.64705	0.64638	0.64647	0.64348	0.64312
RF	0.65800^c	0.65835^c	0.65911^c	0.65896^c	0.65858^c	0.65755^c
GBT	0.63262	0.63322	0.63319	0.63358^f	0.63261	0.63245
LSVM	0.63725	0.63689	0.63731	0.63693	0.57173	0.57432

combination of the retained GTS, AVG and DIF features in Table 5.10. In general, the random forest classifier works the best for every feature set. In total, we report the results of 28 experiments in the three tables, both for the full feature sets as well as the retained feature sets. In 16 experiments, the feature selection leads to better results over using all features. Our observation can be explained by the “no free lunch theorem", which states that no model is generally better than others [176]. However, our best result of 0.65977 is achieved for the retained feature set.

Table 5.10: Test performance of classifiers on full feature set from measured consumption data. The best overall combination of classifier and feature set is in **bold**.

Classifier	GTS+AVG+DIF	
	X_{all}	X_{ret}
DT	0.64646	0.64765^{f}
RF	0.65747^{c}	$\mathbf{0.65977}^{cf}$
GBT	0.63354	0.63355
LSVM	0.63728	0.63760^{f}

Generally, we observe that a combination of two or three feature sets leads to a better test result than for any of the respective single feature sets. An example to demonstrate this observation is as follows: The random forest classifier achieves test AUCs of 0.65726, 0.65248 and 0.65459 for the retained GTS, AVG and DIF features, respectively. It then achieves test AUCs of 0.65835, 0.65896, 0.65755 and 0.65977 for the retained GTS+AVG, GTS+DIF, AVG+DIF and GTS+AVG+DIF feature sets, respectively. Therefore, the test AUCs for each of the combined feature sets are greater than the test AUCs for any of the single feature sets.

5.2.4 Discussion

Our previous works in Chapter 4 that employ the widely-used daily average features established a baseline that only achieved an AUC of slightly above 0.5, i.e. slightly above chance, on real-world NTL detection data sets using linear classifiers. First and foremost, we want to highlight that increasing the performance of machine learning models on noisy real-world data sets is far more challenging than doing so on academic data sets that were created and curated in controlled environments. Furthermore, a small increase of the performance of a real-world model can lead to a major increase of the market value of a company. Our framework presented in this chapter significantly outperforms the baselines established in the literature. As a consequence, our models lead to a better detection of NTL and thus to an increase of revenue and profit for electric utilties as well as an increase of stability and reliability in their critical infrastructure. Our NTL detection framework allows other electric utilities to apply our extensive feature extraction, feature selection and model selection techniques to their data sets, which can lead to potentially greater improvements of NTL detection in their power networks.

It is to our surprise that the gradient-boosted tree classifier performs consistently worse than the random forest classifier in our experiments. In the literature, the gradient-boosted tree is reported to often lead in a wide range of classification problems [28]. However, our observation can also be explained by the “no free lunch theorem".

5.3 Conclusions

In this chapter, we have proposed two neighborhood features for detecting non-technical losses (NTL) of a big data set of 700K customers and 400K inspection results by splitting the area into a grid: the ratio of customers inspected and ratio of inspected customers for which NTL was detected. We generated these features for four different grid sizes. We have analyzed the statistical properties of their distributions and showed why they are useful for predicting NTL. These features were combined with daily average consumption features of the last 12 months before the most recent inspection of a customer from a big data set, which contains 32M meter readings in total. Furthermore, we also used selected customer master data, such as the customer class and voltage of the connection of the customer. We used four machine learning algorithms that are particularly suitable for big data sets to predict if a customer causes NTL or not: logistic regression, k-nearest neighbors, linear support vector machine and random forest. We observed that all models significantly perform better when using the neighborhood and customer master data features compared to using only the time series features. All models perform in the same regime measured by the AUC score. In total, the random forest classifier slightly outperforms the other classifiers.

In a different vein, we have also proposed a novel system for detecting NTL for a real-world data set of 3.6M customers that extracts a number of domain-specific features from the noisy consumption data. We have shown the statistical relevance of these features over generic time series features. As a consequence, our machine learning system allows to detect NTL better than previous works described in the literature.

6

Biases in Inspection Data

The underlying paradigm of big data-driven machine learning reflects the desire of deriving better conclusions from analyzing more data, without the necessity of looking at theory and models. Is having simply more data always helpful? In 1936, The Literary Digest collected 2.3M filled in questionnaires to predict the outcome of that year's US presidential election. The outcome of that big data prediction proved to be entirely wrong, whereas George Gallup only needed 3K handpicked people to make an accurate prediction.

Generally, biases occur in machine learning whenever the distributions of training set and test set are different, for which an example is depicted in Figure 6.1. In this work, we provide a review of different sorts of biases in (big) data sets in machine learning. We aim to shed light on this topic in order to increase the overall attention to this issue in the field of machine learning. We thus provide definitions and discussions of the most commonly appearing biases in machine learning: class imbalance and covariate shift. We also show how these biases can be quantified and corrected.

In Chapter 3 we have identified the open challenges of NTL detection. Two of them address biases: class imbalance and covariate shift that we motivate in Chapters 3.4.1 and 3.4.4, respectively. While we have previously addressed class imbalance in Chapters 4 and 5, we will re-discuss this topic from a more general and more holistic perspective. We have also discussed the issue of covariate shift and provided an example in Figure 3.2. The customers inspected are a sample of the overall population of customers. However, that sample may be biased as electricity suppliers previously may have focused on certain

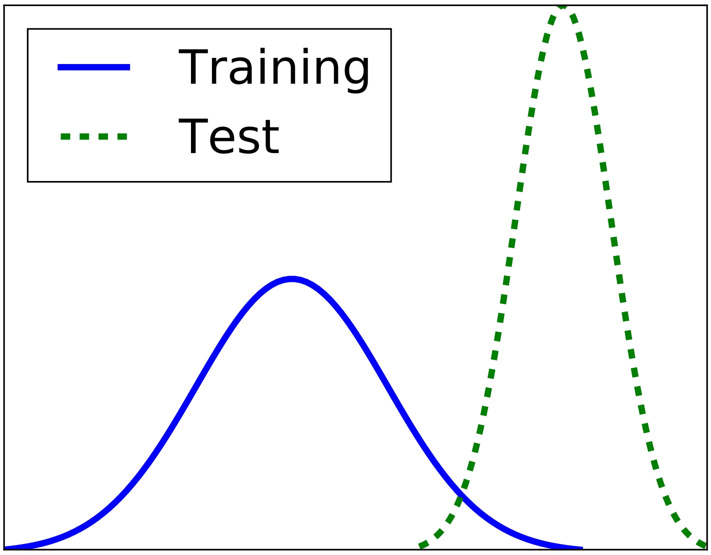


Figure 6.1: Bias: Training and test data sets are drawn from different distributions.

criteria, such as neighborhoods, for inspections. We propose a novel method for quantifying covariate shift and show that some features have a stronger covariate shift than others, making predictions less reliable. In particular, previous inspections were focused on certain neighborhoods or customer classes and that they were not sufficiently spread among the population of customers. We then propose a scalable novel framework for reducing multiple biases in high-dimensional data sets in order to train more reliable predictors. We apply our methodology to the detection of NTL and show that reducing these biases increases the accuracy of the trained predictors.

6.1 The More Data, the Better?

For about the last decade, the big data paradigm that has dominated research in machine learning can be summarized as follows: “It’s not who has the best algorithm that wins. It’s who has the most data." [11] In practice, however, most data sets are (systematically) biased. One example we deal with every day is described in Example 6.1.

Example 6.1. A spam filter is trained on a data set that consists of positive and negative examples. However, that training set was created a few years ago. Recent spam emails are different in two ways: the content of spam emails is different and the proportion of spam among all emails sent out has changed. As an outcome, the spam filter does not detect spam reliably and becomes even less reliable over time.

The appearance of biases in data sets imply a number of severe consequences including, but not limited to, the following: First, conclusions derived from biased - and therefore unrepresentative - data sets could simply be wrong due to lack of reproducibility and lack of generalizability. This is a common issue in research as a whole, as it has been argued that most research published may actually be wrong [84]. Second, these machine learning models may discriminate against subjects of under-represented categories [37], [172].

From a technical perspective, the most commonly appearing biases include *class imbalance* and *covariate shift*. Class imbalance is the case where classes are unequally represented in the data. An example is visualized in Figure 6.2

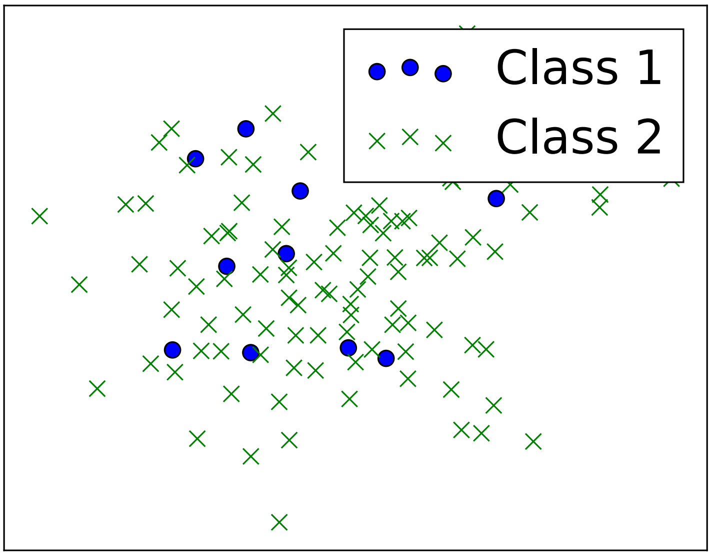


Figure 6.2: Class imbalance: Classes are unequally represented in the data.

Covariate shift is the problem of drawing training and test data sets from different distributions. An example is visualized in Figure 6.3.

These biases are often ignored in both research and practical applications. In part of the statistical literature, the phenomenon of biased data sets is called non-stationarity. In essence, this term indicates different statistics at a different time of collection of the training and test data sets, respectively [154].

More generally, however, the term *bias* is multifaceted in the field of machine learning and describes different matters: The inductive bias of a learning algorithm refers to the set of assumptions a learner makes [115]. For example, logistic regression assumes that the training data is linearly separable. In contrast, the term bias is often used as a synonym

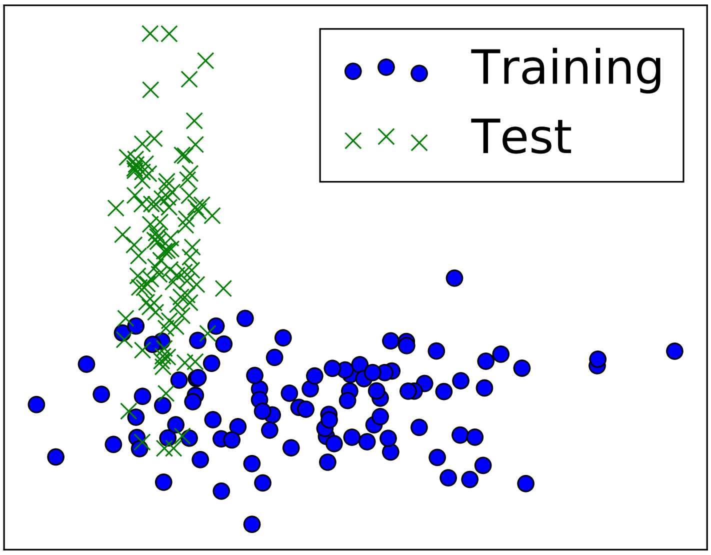


Figure 6.3: Covariate shift: Training and test data sets are drawn from different distributions.

for underfitting in the literature [16]. Moreover, the parameter w_0 of a hypothesis

$$h(x) = w_0 + w_1 x_1 + ... + w_n x_n \tag{6.1}$$

is sometimes called bias as it allows to shift a hypothesis by a fixed offset [16].

Historically, biased data sets have been a long-standing issue in statistics. The failed prediction of the outcome of the 1936 US presidential election is described in Example 6.2. It is often cited in the statistics literature in order to illustrate the impact of biases in data. This example is discussed in detail in [20].

Example 6.2. The Democratic candidate Franklin D. Roosevelt was elected President in 1932 and ran for a second term in 1936. Roosevelt's Republican opponent was Kansas Governor Alfred Landon. *The Literary Digest*, a general interest weekly magazine, had correctly predicted the outcomes of the elections in 1916, 1920, 1924, 1928 and 1932 based on straw polls. In 1936, The Literary Digest sent out 10M questionnaires in order to predict the outcome of the presidential election. The Literary Digest received 2.3M returns and predicted Landon to win by a landslide. However, the predicted result proved to be wrong, as quite the opposite happened: Roosevelt won by a landslide, as depicted in Figure 6.4.

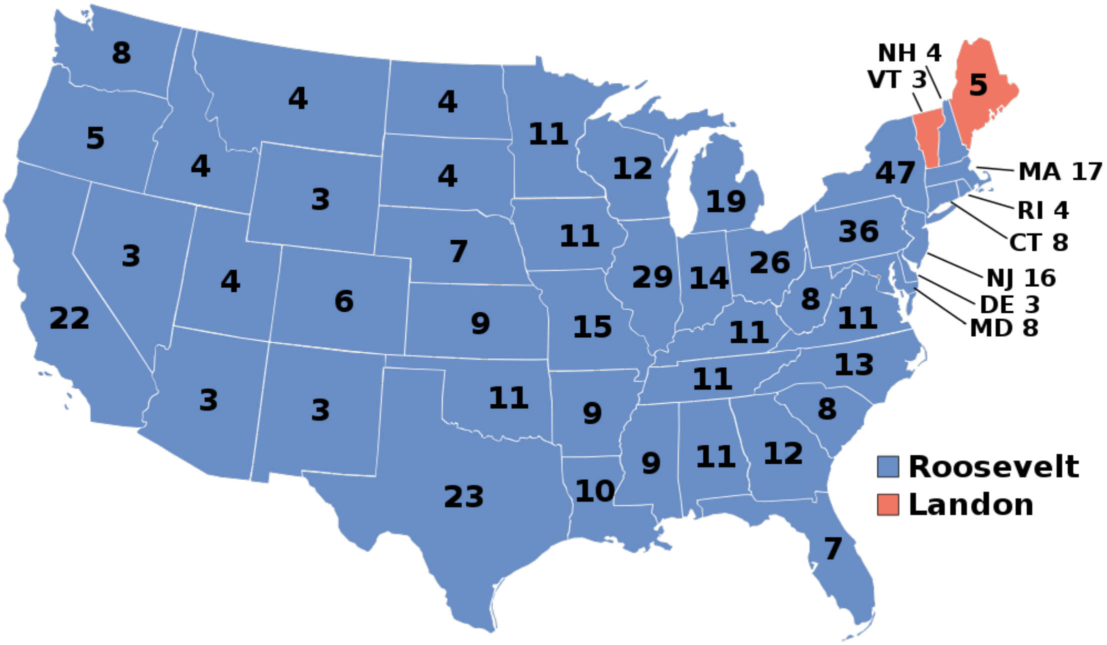


Figure 6.4: 1936 US presidential election results map[a].

This leads to the following questions:

1. How could the prediction turn out to be completely wrong despite the 2.3M participants?
2. How could The Literary Digest actually collect 10M addresses in 1936?

The Literary Digest compiled their data set of 10M recipients mainly from car registrations and phone directories. In that time, the households that had a car or a phone represented a disproportionally rich, and thus biased, sample of the overall population that particularly favored the Republican candidate Landon. In contrast, George Gallup only interviewed 3K handpicked people, which were an unbiased sample of the population. As a consequence, Gallup could predict the outcome of the election very accurately [75].

Even though this historic example is well understood in statistics nowadays, similar or related issues happened for the elections in 1948 and 2016, for which examples are depicted in Figures 6.5 and 6.6, respectively.

Furthermore, in modern big data-oriented machine learning, biases may cause severe impact every day dozens of times, such as in Example 6.3.

Example 6.3. It has been argued that most data on humans may be on white people and thus may not represent the overall population [136]. As a consequence, the predictions of models trained on such biased data may cause infamous news. For example, in 2015,

[a] Source: `http://en.wikipedia.org/United_States_presidential_election,_1936`

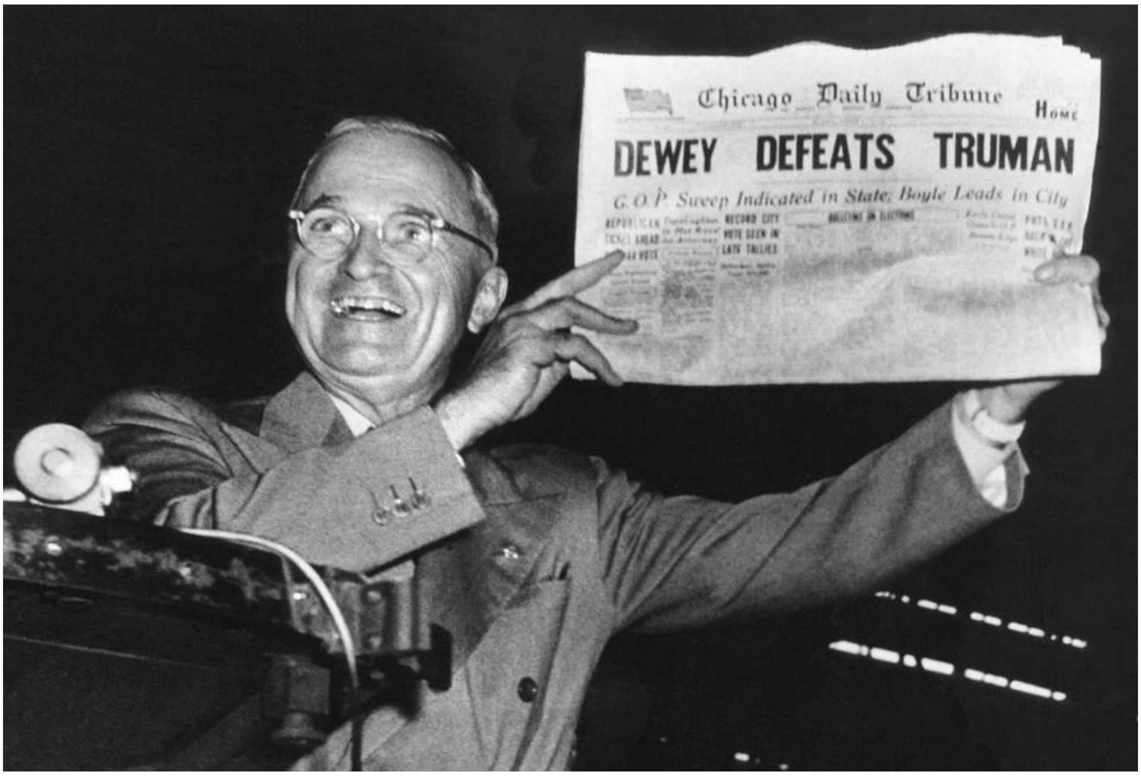
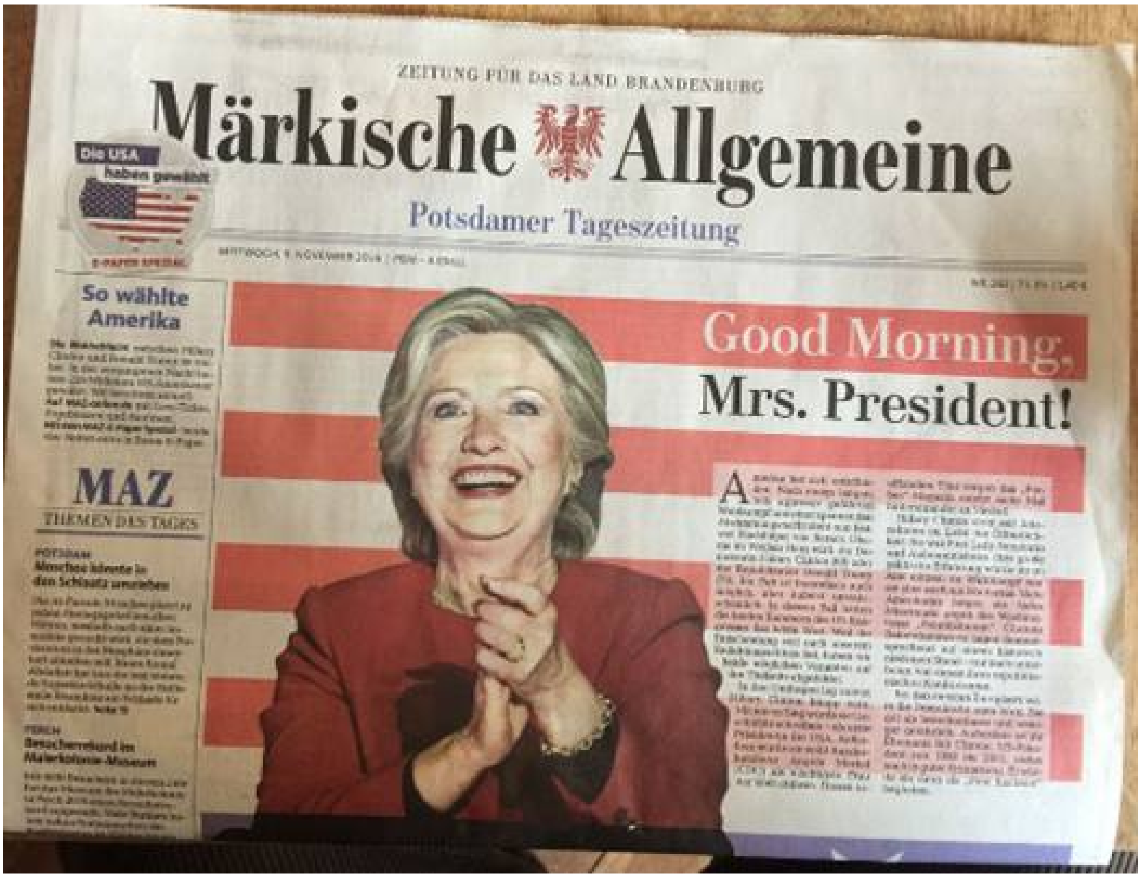


Figure 6.5: 1948 US presidential election result: President Truman was not defeated by Governor Thomas Dewey[a].

[a] Source: http://www.gettyimages.ca/event/the-dewey-truman-election-81078980#victorious-cand-pres-harry-truman-jubilantly-displaying-chicago-daily-picture-id50606875

Google added an auto-tagging feature to its Photos app. This new feature automatically assignes tags to photos, such as bicycle, dog, etc. However, some black users reported that they were tagged as “gorillas", which led to major criticism of Google [37]. Most likely, this mishap was caused by a biased training set, in which black people were largely underrepresented.

The examples provided in this section show that having simply more data is not always helpful in training reliable models, as the data sets used may be biased. In the following sections, we discuss the most commonly appearing biases in data sets. We also present different strategies for assessing biased models and how to correct biases. These techniques include weighting training examples as well as subsampling methods. As a consequence, having data that is more representative is favorable, even if the amount of data used is less than just using the examples from a strongly biased data set.



ZEITUNG FÜR DAS LAND BRANDENBURG

Märkische Allgemeine

Potsdamer Tageszeitung

Die USA haben gewählt

So wählte Amerika

MAZ

THEMEN DES TAGES

Good Morning, Mrs. President!

Figure 6.6: 2016 US presidential election result: Secretary Hillary Clinton did not defeat Donald Trump[a].

[a] Source: Märkische Allgemeine newspaper cover from November 9, 2016.

6.2 Biases in Data Sets

In supervised learning, training examples $(x^{(i)}, y^{(i)})$ are drawn from a training distribution $P_{train}(X, Y)$, where X denotes the data and Y the label, respectively. The training set is biased if the following inequality holds true:

$$P_{train}(X, Y) \neq P_{test}(X, Y). \tag{6.2}$$

Different biases are visualized in Figures 6.2 and 6.3. In order to reduce a bias, it has been shown that example $(x^{(i)}, y^{(i)})$ can be weighted during training as follows [87]:

$$w_i = \frac{P_{test}(x^{(i)}, y^{(i)})}{P_{train}(x^{(i)}, y^{(i)})}. \tag{6.3}$$

However, computing $P_{train}(x^{(i)}, y^{(i)})$ may be impractical in many cases because of the limited amount of data in the training domain. In the following sections, we discuss different biases for which specific assumptions about $P_{train}(X, Y)$ and $P_{test}(X, Y)$ are made.

6.2.1 Class Imbalance

Class imbalance refers to the case where classes are unequally represented in the data. When comparing training set and test set, respectively, we assume [88]:

$$P_{train}(Y) \neq P_{test}(Y), \tag{6.4}$$

$$P_{train}(X|Y) = P_{test}(X|Y). \tag{6.5}$$

An example is depicted in Figure 6.2. Imbalanced classes appear frequently in machine learning, such as in Example 6.4.

Example 6.4. The Modified National Institute of Standards and Technology (MNIST) database consists of 60K training images and 10K test images used for recognition of hand-written digits [98], for which examples are depicted in Figure 6.7.

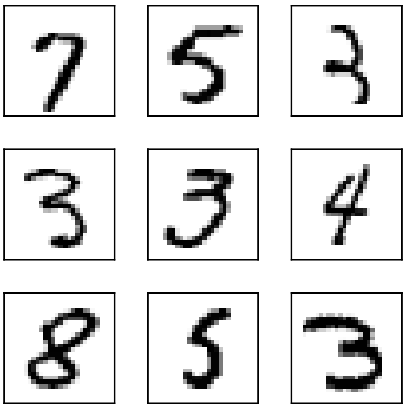


Figure 6.7: MNIST example images.

MNIST has been used in the fields of computer vision and machine learning for the last 20 years. The test accuracies reported in recent research are above 99.6% [150], [171]. The distribution of test labels is depicted in Figure 6.8. We notice that this data set is mainly imbalanced between the different classes. As a consequence, the accuracy is not the right metric for MNIST, as an increase of this metric does not necessarily imply an increased predictive power of a model. We would like to add that the distribution of labels is nearly the same for the training set. Furthermore, there is another imbalance between the training set and test set, respectively. However, that imbalance is less noticeable and we have therefore focused on the imbalance between the labels in each set.

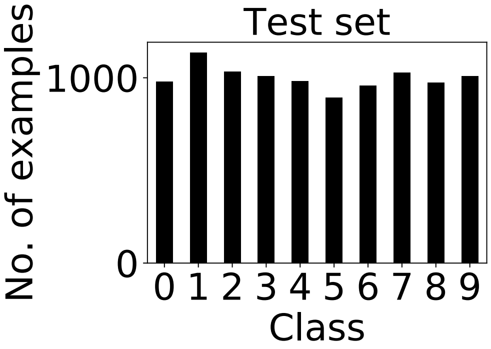


Figure 6.8: Distribution of test labels in MNIST.

Impact on Metrics

Machine learning models trained on imbalanced data sets often tend to predict the majority class. The appearance of imbalanced classes also affects the choice of evaluation metric. Accuracy and recall are the most commonly used metrics in contemporary research works in machine learning [87], [161]. However, both metrics are affected by class imbalance, as discussed in Example 3.1. As a consequence, in many machine learning works, overly high accuracies or recalls are reported for NTL detection in Chapter 3.

A number of metrics that are insensitive to class imbalance can be found in the literature. One common metric is to use a receiver operating characteristic (ROC) curve, which plots the true positive rate against the false positive rate for varying decision threshold values. An example is depicted in Figure 6.9.

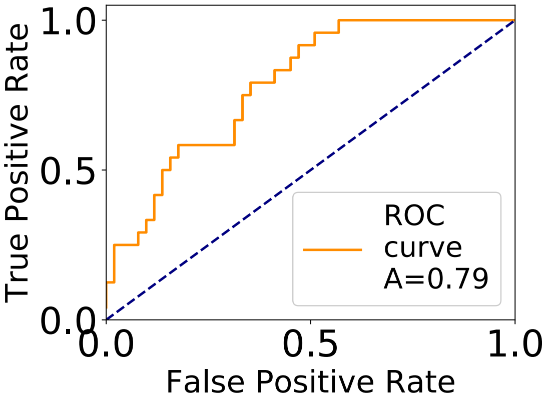


Figure 6.9: Example of receiver operating characteristic (ROC) curve.

The area under the curve (AUC) is a performance measure between 0 and 1, where any binary classifier with an AUC > 0.5 performs better than random guessing [52].

Another metric that is insensitive to class imbalance is the Matthews correlation coefficient (MCC):

$$\frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}, \tag{6.6}$$

which measures the accuracy of binary classifiers taking into account the imbalance of both classes, ranging from -1 to $+1$ [109].

Furthermore, for multi-class problems the intraclass correlation coefficient (ICC) has been proposed [160]. It can be interpreted as the fraction of the total variance that is between the different classes. It has been successfully applied to imbalanced multi-class learning problems [174].

Correction

In order to correct the class imbalance during training, a number of methods are proposed in the literature. First, weighting examples by the inverse proportion of examples per class using Equation 6.3 is proposed in the literature [88]. On the one hand, one intuitive method is undersampling the majority classes by dropping training examples, either randomly or by specific criteria [105], [164]. This approach leads to smaller data sets, but may lack variation, as important examples could have been dropped. On the other hand, oversampling the minority classes by creating more training examples is proposed in the literature. Most trivially, training examples can simply be randomly copied. However, there are also more sophisticated algorithms, such as the synthetic minority over-sampling technique (SMOTE), which attempts to create synthetic examples representing the minority class by interpolating between neighboring data points [27]. Generally, adding more examples leads to larger training sets, which, in turn, leads to increased training time. Therefore, combinations of oversampling and undersampling were proposed [12], [102].

6.2.2 Covariate Shift

The problem of training data and production data having different distributions has initially been addressed in the field of computational learning theory [33], which also calls it covariate shift, sampling bias or sample selection bias. We assume [88]:

$$P_{train}(X) \neq P_{test}(X), \tag{6.7}$$

$$P_{train}(Y|X) = P_{test}(Y|X). \tag{6.8}$$

Covariate shift appears frequently in machine learning as discussed in Example 6.1. Machine learning models trained on biased training sets tend not to generalize on test data that is from the true underlying distribution of the population. An example of covariate shift is depicted in Figure 6.3.

However, there are alternate definitions, such as in [179]:

- Assume that all examples are drawn from a distribution D with domain $X \times Y \times S$,
- where X is the feature space,
- Y is the label space and
- S is $\{0, 1\}$.

Examples (x, y, s) are drawn independently from D. $s = 1$ denotes a selected example, whereas $s = 0$ denotes the opposite. The training is performed on a sample that comprises all examples that have $s = 1$. $P(s|x, y) = P(s|x)$ implies that s is independent of y given x. In this case, the selected sample is biased but the bias only depends on the feature vector x [179].

The literature distinguishes classifiers into local learners and global learners, respectively [179]. The terms “global" and “local”, respectively, have been established as follows: A global learner also uses $P(X)$, which is a (global) distribution over the entire input data. In contrast, a local learner uses $P(Y|X)$, which refers for every $x^{(i)} \in X$ to a local distribution $P(Y|x^{(i)})$.

Local Learners

For a local learner, the prediction of the learner depends only on $P(Y|X)$ for an increasing number of training examples. Previous research addresses this behavior with the term “asymptotically". We assume $P_{train}(Y|X) = P_{test}(Y|X)$ in Equation 6.8. Hence, a local learner is not affected by covariate shift. Examples include logistic regression and hard-margin support vector machine (SVM) [179].

Global Learners

In contrast, the prediction of a global learner depends asymptotically on both, $P(Y|X)$ and $P(X)$. We assume $P_{train}(X) \neq P_{test}(X)$ in Equation 6.7. Hence, a global learner is affected by covariate shift. Examples include decision tree learners such as ID3 or C4.5, naive Bayes and soft-margin SVM [179].

Reduction

Instance weighting using density estimation has been proposed for correcting covariate shift [156]. Examples can either be weighted during training [33] or the weights can be used for rejection sampling [179]. Historically, the Heckman method has been proposed to correct covariate shift by estimating the probability of an example being selected into the training sample [76]. However, the Heckman method only applies to linear regression models.

6.2.3 Other Biases

Below we list other types of biases that have been investigated. Without any pretension for exhaustivity, we define those biases and refer the reader to the corresponding literature for further details. For instance, a change of functional relations can create a new bias and thus lead to $P_{train}(Y|X) \neq P_{test}(Y|X)$ [88]. Also, it has been shown that biases can be created by transforming the feature space [13]. Furthermore, a bias specific to neural networks has been reported: During training, a change of the weights in one layer may alter the distribution of the input to the following layer. This so-called internal covariate shift slows down convergence of training a neural network and may result in a neural network that overfits [85]. Internal covariate shift can be compensated by normalizing the input of every layer. By doing so, it has been reported that the training can be significantly accelerated. The resulting neural network is also less likely to overfit. This approach is radically different to regularization [15], as it addresses the cause of overfitting rather than trying to improve a model that overfits.

6.3 Quantifying Covariate Shift

When using a data set, we need to assess whether the training set actually has actually a covariate shift before thinking about reducing the later. The Kullback-Leibler divergence [93] is a measure of the difference of two probability distributions P and Q:

$$D_{\mathrm{KL}}(P\|Q) = -\sum_i P(i)\,\log\frac{Q(i)}{P(i)}, \tag{6.9}$$

which is equivalent to

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P(i)\,\log\frac{P(i)}{Q(i)}. \tag{6.10}$$

However, it is challenging (1) to adapt this measure to multi-dimensional data that is a combination of discrete and continuous features, which is common in machine learning, and (2) to define criteria from what values on a distance is an indicator for a covariate shift.

6.3.1 Methodology

Instead, a preferred methodology for quantifying covariate shift is:

1. First, we add a feature `s` and assign the values `1` or `0` to the training data ($s = 1$) or production data ($s = 0$), respectively.

2. These data sets are furthermore merged into one data set. This latter is split into a training set `X1` (with no relation to the original training set) and a test set `X2`.

3. The objective is to develop a supervised learning method capable of predicting the feature `s` using `X1`.

4. The performance of the classifier on `X2` is then quantified using the Matthews correlation coefficient (MCC) defined in Equation 6.6. which measures the accuracy of binary classifiers taking into account the imbalance of both classes, ranging from -1 to $+1$ [109].

5. The greater the MCC, the greater the covariate shift. A concrete threshold for covariate shift depends on the problem, however 0.2 has been proposed [106]. Though a low MCC does not automatically imply the lack of a covariate shift, a significant MCC value is an indicator of covariate shift.

We propose to add the following following novelties to this approach:

1. Tree classifier: Decision tree learning is affected by covariate shift. Decision trees scale to very large data sets while they allow to learn non-linearities. Soft-margin SVMs are also global learners, however, for large data sets only a linear kernel is learnable in a feasible amount of time.

2. Model selection: We want to find a model which is able to distinguish between both distributions. Thus maximizing the MCC on the test set is equivalent to finding the best two-class classification between production data and original training data. For this, we optimize the five most important tree model parameters by randomly drawing from probability distributions: Max. number of leaves, max. number of levels, measure of the purity of a split, min. number of samples required to be at a leaf and min. number of samples required to split a node.

3. Cross-validation: We also split the data set into k folds in order to reduce the overfitting. This leads to a more reliable model for covariate shift quantification. The MCC per model, denoted by $\overline{MCC}$, is the average of the MCCs of the k test sets. The standard deviation of the k test MCCs serves as the reliability of $\overline{MCC}$. The lower the standard deviation, the more reliable $\overline{MCC}$.

Our proposed methodology is depicted in Algorithm 6.1.

Note: the inspection results are not taken into account as covariate shift only concerns the distributions of the inputs.

6.3.2 Evaluation

We use the same data as in Chapter 5.2. A complete list of the customer master data used in the following experiments is depicted in Table 6.1.

Algorithm 6.1 Quantifying covariate shift.

```
$result \leftarrow 0$
$reliability \leftarrow 0$
$selected \leftarrow train\_data.add\_feature(s, 1)$
$not\_selected \leftarrow prod\_data.add\_feature(s, 0)$
$data \leftarrow selected \cup not\_selected$
$folds \leftarrow cv\_folds(data, k)$
for $model$ in $get\_model\_candidates()$ do
    $mccs \leftarrow list()$
    for $fold$ in $folds$ do
        $X_{train}, X_{test}, y_{train}, y_{test} \leftarrow fold$
        $classifier \leftarrow DecisionTree(model)$
        $classifier.train(X_{train}, y_{train})$
        $y_{pred} \leftarrow classifier.predict(X_{test})$
        $mccs.append(MCC(y_{test}, y_{pred}))$
    end for
    $mcc\_mean \leftarrow mean(mccs)$
    if $mcc\_mean > result$ then
        $result \leftarrow mcc\_mean$
        $reliability \leftarrow std(mccs)$
    end if
end for
return $result, reliability$
```

Table 6.1: Assessed features.

Feature	Possible Values
Class	Power generation infrastructure, residential, commercial, industrial, public, public illumination, rural, public service, reseller
Contract status	Active, suspended
Location	Longitude and latitude
Meter type	22 different meter types
Number of wires	1, 2, 3
Voltage	≤2.3kV, >2.3kV

Implementation

All computations were run on a server with 24 cores and 128 GB of RAM. The entire code was implemented in Python using `scikit-learn` [131] for machine learning. `scikit-learn` allows to distribute the training of the cross-validated classifiers among all cores. The maps were plotted using `cartopy` [114]. In total, all results and plots reported in this chapter were computed in 12 hours using this infrastructure. Our implementation is available as open source: `http://github.com/pglauner/SpatialBiasNTL`.

Model Parameters

In the following experiments, we use $k = 10$-fold cross-validation. In each experiment, we train 1K trees, which are 100 different tree models trained on each of the 10 folds. We optimize the five tree model parameters by randomly drawing from predefined uniform probability distributions depicted in Table 6.2.

Table 6.2: Tree model parameters.

Parameter	Range
Max. number of leaves	$[2, 20)$
Max. number of levels	$[1, 20)$
Measure of the purity of a split	{entropy, gini}
Min. number of samples required to be at a leaf	$[1, 20)$
Min. number of samples required to split a node	$[2, 20)$

We have chosen these ranges based on best practice recommendations and our own experience. Furthermore, the two classes ($s = 1$ and $s = 0$) are imbalanced, i.e. there are more examples of the non-inspected customers than the inspected ones. In order to take this into account during training, we associate weights with the classes such that the examples of the minority class have stronger impact.

Global Covariate Shifts

In the following experiments, we compute different global types of covariate shift by using all customers in each experiment. We therefore do not split the customers into different geographical areas. We have previously presented the customer master data features available in Table 6.1. We compute the global covariate shift of each of these features. We report our results in Table 6.3.

Overall, the strongest covariate shift is in the location with a MCC value of 0.22367. This means that previous inspections are mostly biased towards the location of customers. The location is also the only feature that is beyond the threshold of 0.2 mentioned before. The features class, number of wires and meter type are below the threshold but are greater

Table 6.3: Global covariate shift of single features. $\overline{MCC}_{max}$ denotes the maximum average of the MCCs of the $k = 10$ test sets among all 100 tree models trained on a feature. σ denotes the standard deviation of those $k = 10$ MCC test scores, which is a reliability measure of $\overline{MCC}_{max}$.

Feature	$\overline{MCC}_{max}$	σ
Location	**0.22367**	0.03453
Class	0.16255	0.01371
Number of wires	0.14111	0.00794
Meter type	0.13158	0.00382
Voltage	0.07092	0.02375
Contract status	0.03744	**0.09183**

than 0.1. There is almost no covariate shift of previous inspections towards the voltage and contract status features. The standard deviation of the MCCs is the greatest for the contract status. The reason for this is a strong overfit in one of the folds. All other MCCs have a much lower standard deviation, making them more reliable.

Next, we create compound features that are composed of multiple features. Due to the great number of possible combinations, we assess all 2-combinations as well as the 6-combination of all features. We visualize the MCCs for all 2-combinations in Figure 6.10 and report the MCCs in Table 6.4.

For the 6-combination comprising all features, we computed $\overline{MCC}_{max} = 0.27325$, which is the maximum covariate shift of all compound features. Therefore, the spatial covariate shift contributes to this covariate shift the most, however, the other covariate shifts contribute a fraction as well.

Local Covariate Shifts

We now entirely focus on spatial covariate shift since it is the strongest one among the different types of covariate shift. In the following experiments we compute local covariate shifts by splitting the customers in different locations. The data set provides the following divisions in the following hierarchical order:

1. 9 regions
2. 261 municipalities
3. 1,380 localities

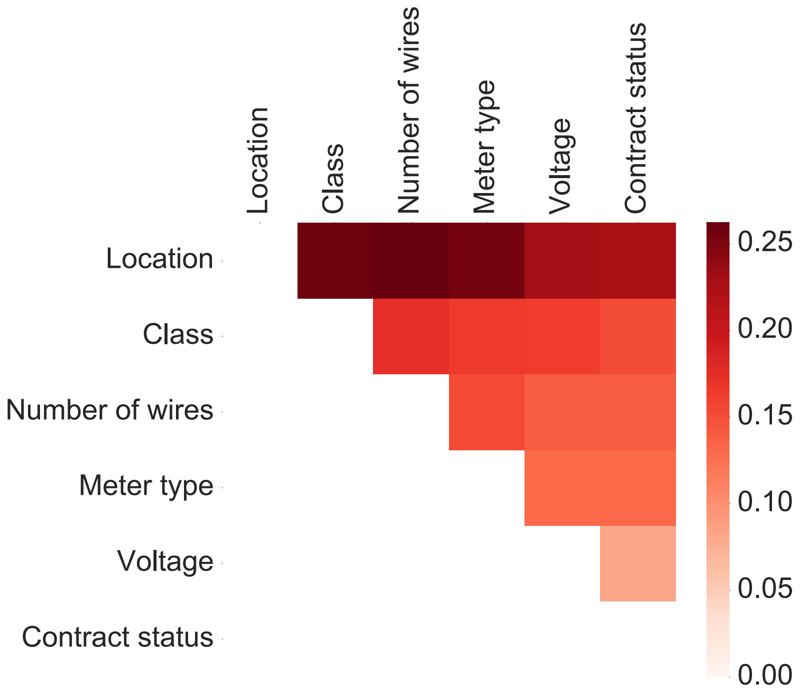


Figure 6.10: Global covariate shift of compound features.

4. 19,026 neighborhoods

All customers are located in one Brazilian state. We observe that spatial covariate shift is smoothened for regional level in Figure 6.11. It becomes increasingly more granular at municipal, local and neighborhood levels in Figures 6.12 through 6.14, respectively. We also notice that the spatial covariate shifts at lower levels tend to increase, which is depicted by increasing upper limits of the color bars.

6.3.3 Discussion

We have shown that covariate shift exists in our real-world data set. The features of the customer data that are most affected by covariate shift are the location, followed by class, number of wires and meter type. Classifiers that use other features such as the voltage or contract status instead tend to be more reliable. We have also shown that the spatial covariate shift exists on different levels of granularity and that municipalities and localities with very strong covariate shifts exist. Subsequently, these local covariate shifts have significant impact on the covariate shifts on higher levels or even globally on the entire data set. Therefore, using all inspection results of the data set for training a NTL predictor from a big data perspective leads to biased models that may not reliably detect NTL. As a consequence, reducing the spatial covariate shift in the data set must be a priority in order to learn reliable NTL predictors.

Table 6.4: Global covariate shift of compound features.

Feature	$\overline{MCC}_{max}$	σ
All	**0.27325**	0.03014
Location + number of wires	**0.26206**	0.03676
Location + class	0.25796	0.03540
Location + meter type	0.25479	0.03884
Location + voltage	0.22944	0.03544
Location + contract status	0.22335	0.03454
Class + number of wires	0.17501	0.00468
Class + meter type	0.16472	0.00309
Class + voltage	0.16322	0.01400
Number of wires + meter type	0.15283	0.00274
Class + contract status	0.15158	0.00992
Number of wires + voltage	0.14156	0.00800
Number of wires + contract status	0.14111	0.00794
Meter type + voltage	0.13165	0.00381
Meter type + contract status	0.13155	0.00382
Voltage + contract status	0.08213	**0.08301**

However, the finer the hierarchical granularity, the more divisions cannot be used for the computations for the following reasons. First, using $k = 10$-fold cross-validation, training is only possible if a division has at least k customers. Second, the $k-1$ folds used for training must have examples of both classes. Third, the MCC can only be computed for denominator $\neq 0$, which is the case for $(TP > 0 \wedge TN > 0) \vee (FP > 0 \wedge FN > 0)$. If the test MCC cannot be computed for a fold of a model, only the MCCs of the remaining folds are used in cross-validation. If no MCCs can be computed for a division, we skip it in the plotting. For instance, this effect has become most apparent at neighborhood level in the west of that state due to the low population density.

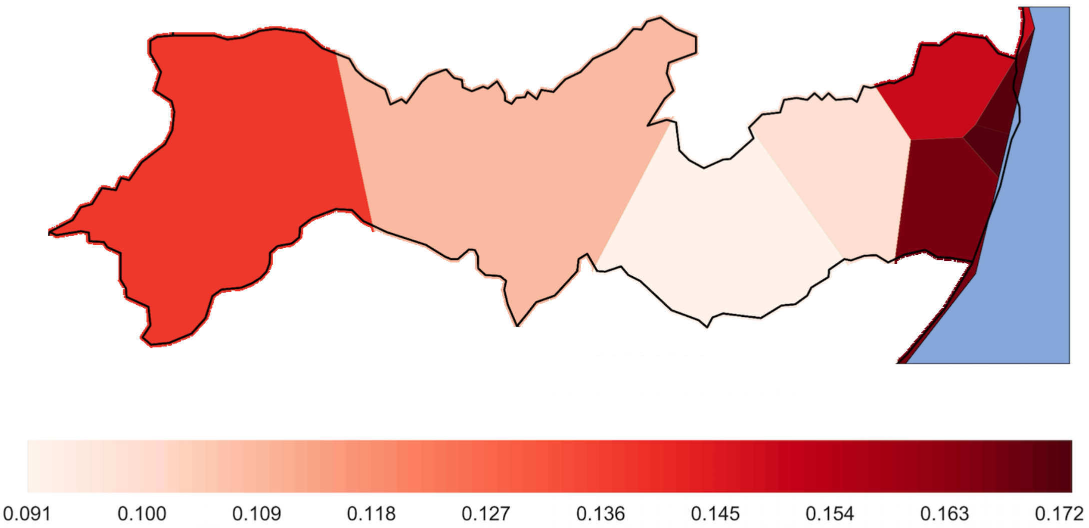


Figure 6.11: Spatial covariate shift at regional level. For each division, we compute the median location of the respective customers and assign $\overline{MCC}_{max}$ to it. We then use nearest interpolation to generate the local covariate shift maps.

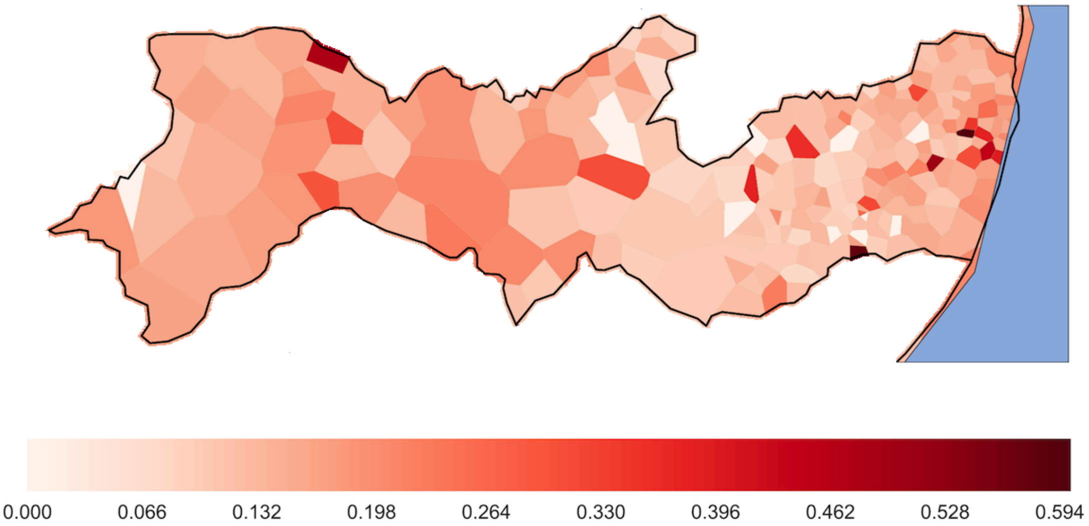


Figure 6.12: Spatial covariate shift at municipal level.

6.4 Reducing Multiple Biases

We propose the following methodology employing instance weighting that is defined in Equation 6.3.

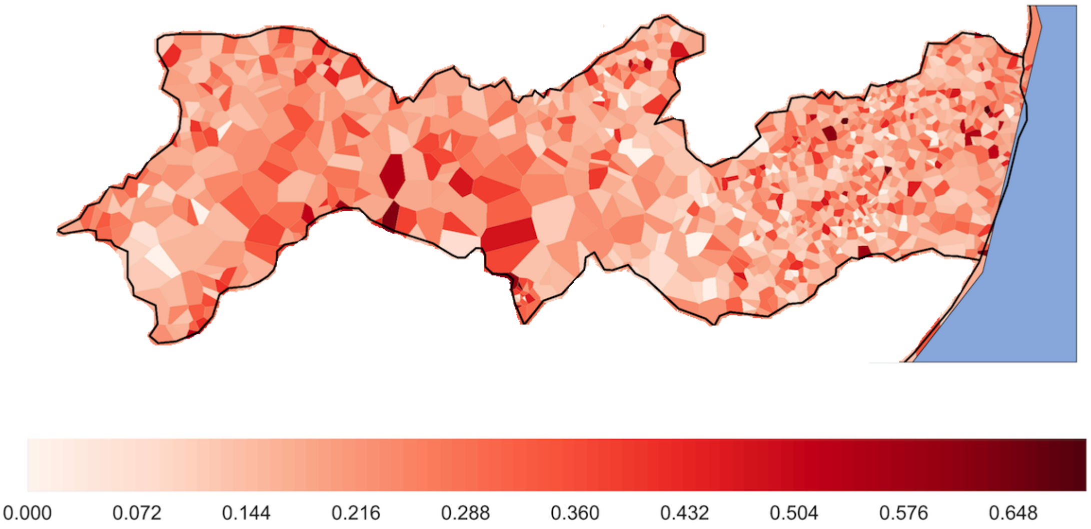


Figure 6.13: Spatial covariate shift at local level.

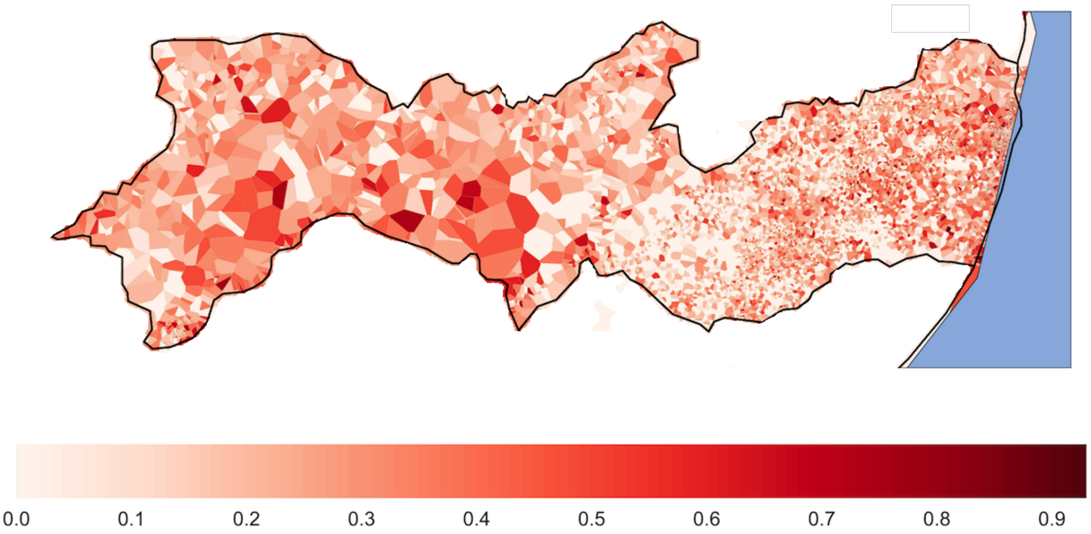


Figure 6.14: Spatial covariate shift at neighborhood level.

6.4.1 Methodology

Given the assumptions made for class imbalance in Equations 6.4 and 6.5, we compute the corresponding weight for example i having a label of class k as follows:

$$w_{i,k} = \frac{P_{test}(x^{(i)}, y_k^{(i)})}{P_{train}(x^{(i)}, y_k^{(i)})} = \frac{P_{test}(x^{(i)}|y_k^{(i)})P_{test}(y_k^{(i)})}{P_{train}(x^{(i)}|y_k^{(i)})P_{train}(y_k^{(i)})} = \frac{P_{test}(y_k^{(i)})}{P_{train}(y_k^{(i)})}. \tag{6.11}$$

We use the empirical counts of classes for computing $P_{<dist>}(y_k)$.

Given the assumptions made for covariate shift Equations 6.7 and 6.8, we compute the corresponding weight for the bias in feature k of example i as follows:

$$w_{i,k} = \frac{P_{test}(x_k^{(i)}, y^{(i)})}{P_{train}(x_k^{(i)}, y^{(i)})} = \frac{P_{test}(y^{(i)}|x_k^{(i)})P_{test}(x_k^{(i)})}{P_{train}(y^{(i)}|x_k^{(i)})P_{train}(x_k^{(i)})} = \frac{P_{test}(x_k^{(i)})}{P_{train}(x_k^{(i)})}. \tag{6.12}$$

We use density estimation for computing $P_{<dist>}(x_k^{(i)})$ [131].

There may be a variety of biases in a learning problem that are far more than just class imbalance and covariate shift on a single dimension. We have shown previously that there may be multiple types of covariate shift, for example spatial covariate shifts on different hierarchical levels. As we have shown in Chapter 6.3, there may be also covariate shifts for other master data, such as for the customer class or for the contract status.

We now aim to correct n different biases at a same time, e.g. for class imbalance as well as different types of covariate shift. As $x^{(i)}$ has potentially many dimensions with a considerable covariate shift, computing the joint $P_{<dist>}(x^{(i)})$ becomes impractical for an increasing number of dimensions. We propose a uniformed and scalable solution to combine weights for correcting the n different biases, comprising for example of class imbalance and different types of covariate shift. The corresponding weights per bias of an example are $w_{i,1}, w_{i,2}, ..., w_{i,n}$. The example weight w_i is the harmonic mean of the weights of the biases considered is computed as follows:

$$w_i = \frac{n}{\frac{1}{w_{i,1}} + \frac{1}{w_{i,2}} + \cdots + \frac{1}{w_{i,n}}} = \frac{n}{\sum\limits_{k=1}^{n} \frac{1}{w_{i,k}}}. \tag{6.13}$$

As the different $w_{i,k}$ are computed from noisy, real-world data, special care needs to be paid to outliers. Outliers can potentially lead to very large values $w_{i,k}$ for the density estimation proposed above. It is for that reason that we choose the harmonic mean, as it allows to penalizes extreme values and give preference to smaller values.

6.4.2 Evaluation

We use the same data as in Chapter 5.2. We retain $M = 150,700$ customers. For these customers, we have a complete time series of 24 monthly meter readings before the most recent inspection. From each time series, we compute 304 features comprising generic time series features, daily average features and difference features, as detailed in Table 5.7. We employ hypothesis tests to the features in order to retain the ones that are statistically relevant. These tests are based on the assumption that a feature x_k is meaningful for the prediction of the binary label vector y if x_k and y are not statistically independent [139]. For binary features, we use Fisher's exact test [54]. In contrast, for continuous features, we use the Kolmogorov-Smirnov test [107]. We retain 237 of the 304 features.

We previously found a random forest (RF) classifier to perform the best on this data compared to decision tree, gradient-boosted tree and support vector machine classifiers.

It is for this reason that in the following experiments, we only train RF classifiers. When training a RF, we perform model selection by doing randomized grid search, for which the parameters are detailed in Table 6.5. We use 100 sampled models and perform 10-fold cross-validation for each model.

Table 6.5: Model parameters for random forest.

Parameter	Values
Max. number of leaves	$[2, 1000)$
Max. number of levels	$[1, 50)$
Measure of the purity of a split	{entropy, gini}
Min. number of samples required to be at a leaf	$[1, 1000)$
Min. number of samples required to split a node	$[2, 50)$
Number of estimators	20

6.4.3 Discussion

We have previously shown in Chapter 6.3 that the location and class of customers have the strongest covariate shift. When reducing these, we first compute the weights for the class imbalance, the spatial covariate shift and customer class covariate shift, respectively, as defined in Chapter 6.2. For covariate shift, we use randomized grid search for a model selection of the density estimator that is composed of the kernel type and kernel bandwidth. The complete list of parameters and considered values is depicted in Table 6.6.

Table 6.6: Density estimation parameters.

Parameter	Values
Kernel	{gaussian, tophat, epanechnikov, exponential, linear, cosine}
Bandwidth	$[0.001, 10]$ (log space)

Next, we use Equation 6.13 to combine these weights step by step. For each step, we report the test performance of the NTL classifier in Table 6.7. It clearly shows that the larger the number of addressed biases, the higher the reliability of the learned predictor.

Table 6.7: Test performance of random forest. $\overline{AUC}$ denotes the mean test AUC of the 10 folds of cross-validation for the best model.

Biases Reduced	$\overline{AUC}$
None	0.59535
Class imbalance	0.64445
Class imbalance + spatial covariate shift	0.71431
Class imbalance + spatial covariate shift + customer class covariate shift	**0.73980**

6.5 Conclusions

In this chapter, we first have presented a number of historic and modern examples of biased data sets that resulted in unreliable models. Biases occur in machine learning whenever training sets are not representative of the test data. Even though biases have been recognized as an issue in statistics since the mid-20th century, they only recently started to get more attention in machine learning, yet the situation is evolving. We then provided an extensive review of biases in machine learning, with a (special) focus on the most common ones: class imbalance and covariate shift. As a consequence, in many cases it may not be helpful to simply have more data, but rather to have (possibly less) data that is more representative.

We have then proposed a novel framework for quantifying and visualizing covariate shift in data sets, with a particular focus on spatial data sets. In the context of non-technical loss (NTL) detection, we showed that there is a covariate shift between the inspected customers and the overall population of customers. We showed that some features have a stronger covariate shift than others. In particular, the spatial covariate shift is the strongest and appears in different hierarchical levels. Subsequently, machine learning models trained on this data will lead to unreliable NTL predictions. Last, we proposed a scalable model for reducing multiple biases in high-dimensional data at the same time. We applied our methodology to NTL detection. Our model leads to more reliable predictors, thus allowing to better detect customers that have an irregular power usage.

7

Conclusions and Prospects

In emerging markets, non-technical losses (NTL) constitute the dominant part of losses in power grids. Concretely, they may range up to 40% of the total electricity distributed in countries such as Brazil, India or Pakistan. The economic effects thereof for utilities are enormous as the world-wide total financial losses are about USD 100 billion per year.

The main research question of this thesis is:

How can we detect non-technical losses better in the real world?

In this research and implementation of machine learning for NTL detection using real-world data, we have shown that our solution has a large part to play in the future of NTL detection. Our models have the potential to generate significant economic value in real-world applications, as they are being deployed in a commercial NTL detection software. To conclude this thesis, we will review what we consider are the main contributions made and discuss plans for future related work.

7.1 Contributions

The main achievements of this thesis can be outlined as follows:

1. Review of Causes and Economic Impact of NTL
In Chapter 1, we have provided an in-depth discussion of the causes of NTL. Our review has shown that NTL are a prime concern and often range up to 40% of the total electricity distributed. The annual world-wide costs for utilities due to NTL are estimated to be around USD 100 billion. Reducing NTL in order to increase revenue, profit and reliability of the grid is therefore a vital interest to utilities and authorities.

2. Identification of the Open Challenges of NTL Detection
In Chapter 3, we first surveyed the state-of-the-art research efforts in a up-to-date and comprehensive review of algorithms, features and data sets used. We also compared the various approaches reported in the literature. Next, we identified the key scientific and engineering challenges of NTL detection that have not yet been addressed in scientific works. We put these challenges in the context of AI research as a whole as they are of relevance to many other real-world learning and anomaly detection problems.

3. Comparing Industrial NTL Detection Systems based on Expert Knowledge to those based on Machine Learning
In Chapter 4, we used an industrial NTL detection system based on Boolean logic. We improved it by fuzzifying the rules and compared both to a NTL detection system based on machine learning. We showed that the one based on machine learning significantly outperforms the others based on expert knowledge.

4. Combining Industrial Expert Knowledge with Machine Learning for the Decision Making
Despite the superiority of machine learning-based approaches over expert knowledge for NTL detection, electric utilities are reluctant to move to large-scale deployments of automated systems that learn NTL profiles from data due to the latter's propensity to suggest a large number of unnecessary inspections. In order to allow human experts to feed their knowledge in the decision process, we proposed in Chapter 4 a method for visualizing prediction results of a machine learning-based system at various granularity levels in a spatial hologram. Our approach allows domain experts to put the classification results into the context of the data and to incorporate their knowledge for making the final decisions of which customers to inspect.

5. Comprehensive Learning from the Customer Data how to Find Customers with Irregular Behavior
In Chapter 5, we took full advantage of the customer data in order to detect NTL better. We derived features that include information about the neighborhood. We showed that the neighborhood of customers contains information about whether a customer may cause NTL or not. We analyzed the statistical properties of these features and showed why they are useful for NTL detection. By using information of the neighborhood, we can predict NTL better as there are geographic clusters of NTL among the customers. Next, we proposed a novel and flexible framework to compute a large number of domain-specific features and generic features from the noisy industrial consumption time series of customers for NTL detection. We retained the statistically meaningful features extracted from the noisy consumption data and optimized different classifiers to predict NTL.

6. Handling the Biases in the Inspection Data
In Chapter 6, we provided an unified and holistic introduction to the problem of biased data sets. We have demonstrated its importance not only to NTL detection, but to machine learning as a whole. We proposed an algorithm for quantifying covariate shift and showed that the location and class of customers have the strongest covariate shift in NTL detection. We then proposed a scalable novel framework for reducing multiple biases in high-dimensional data sets in order to train more reliable predictors. We applied our methodology to the detection of NTL and showed that reducing these biases increases the accuracy of the trained predictors.

7.2 Future Work

The framework put in place by the work in this thesis gives a strong basis for several interesting directions of future work, which are likely to lead to increasing levels of NTL detection in useful applications. In order to advance the field of NTL detection, we feel that work should be concentrated on four main areas:

1. Creation of a Publicly Available Real-World Data Set
How can we compare different models?

The works reported in the literature describe a wide variety of different approaches for NTL detection. Most works only use one type of classifier, such as in [22], [34], [123], [151], whereas some works compare different classifiers on the same features, such as in [126], [140], [141]. However, in many cases, the actual choice of classification algorithm is less important. This can also be justified by the “no free lunch theorem" introduced in [176], which states that no learning algorithm is generally better than others.

We are interested in not only comparing classification algorithms on the same features, but instead in comparing totally different NTL detection models as argued in Chapter 3.4.6. We suggest to create a publicly available data set for NTL detection. Generally, the more data, the better for this data set. However, acquiring more data is costly. Therefore, a tradeoff between the amount of data and the data acquisition costs must be found. The data set must be based on real-world customer data, including meter readings and inspection results. This will allow to compare various models reported in the literature. For these reasons, it should reflect at least the following properties:

- Different types of customers: the most common types are residential and industrial customers. Both have very different consumption profiles. For example, the consumption of industrial customers often peaks during the weekdays, whereas residential customers consume most electricity on the weekends.

- Number of customers and inspections: the number of customers and inspections must be in the hundreds of thousands in order to make sure that the models assessed scale to big data sets.

- Spread of customers across geographical area: the customers of the data set must be spread in order to reflect different levels of prosperity as well as changes of the climate. Both factors affect electricity consumption and NTL occurrence.

- Sufficiently long period of meter readings: due to seasonality, the data set must contain at least one year of data. More years are better to reflect changes in the consumption profile as well as to become less prone to weather anomalies.

We had initial discussions with the IEEE Power & Energy Society (PES) on this opportunity in autumn 2018. As a result, senior PES members have expressed interest in this topic. They are confident that some utilities will be open to sharing some anonymized data with a future PES working group. We are also currently involved in a respective grant application together with a university from Uruguay. Our collaborators have also already started initial negotiations with a local utility.

2. Deep Learning for Smart Meter Recordings

In order to improve the predictive power of the process depicted in Figure 3.1, [38], [142] suggest to roll out more smart meters in order to have more data available for better decision making when predicting NTL for individual customers. However, to date, many customers in emerging markets do not have smart meters yet due to the high rollout costs. Nonetheless, over time more smart meters will be deployed.

Deep learning allows to self-learn hidden correlations and increasingly more complex feature hierarchies from the raw data input as discussed in [97]. This approach has led to breakthroughs in image analysis and speech recognition as presented in [77].

Smart meters provide meter readings every 15 or 30 minutes instead of every 1 month. The following advances in deep learning have the potential to take advantage of this information in that data and thus to predict NTL more accurately:

- A long short-term memory (LSTM) defined in [80] is a modular recurrent neural network composed of LSTM cells. Training LSTMs takes advantage of backpropagation through time, a variant of backpropagation. Its goal is to minimize the LSTM's total cost on a training set. LSTMs have been reported to outperform regular recurrent neural networks (RNN) and hidden Markov models (HMM) in classification and time series prediction tasks [152].
- Convolutional neural networks (CNN) implement invariance in neural networks and are inspired by biological processes [99]. Traditionally, they have been mostly applied to computer vision problems, such as hand-written digit recognition. However, they have recently been used for time series analysis. They have in particular demonstrated to be easier to be trained for this task [39], [167].
- The literature also proposes to transform a time series into an image using different transformations, such as gramian angular summation fields, gramian angular difference fields and Markov transition fields [173]. The images generated from these transformations can then be analyzed using a CNN.

Furthermore, deep learning also has the potential to (partially) overcome the challenge of feature description for NTL detection, see Chapter 3.4.2, as a whole.

3. Explainability of Automated Decision Making

We have previously discussed in Chapter 4 that electric utilities are reluctant to move to machine learning-based detection of customers that potentially cause NTL. Due to the high costs of physical on-site inspections, utilities want to understand why a certain customer may cause NTL. We have therefore proposed a combination of machine learning with augmented reality such that decision makers with the utilities can understand the predictions in the context of the data. The field of explainable machine learning has got more attention in the last few years, in particular for decisions that involve significant costs or may cause damage or loss to human life [10], [18], [145]. We are therefore interested in building machine learning models for NTL detection that automatically reason why they made a certain decision. This would help to increase acceptance of machine learning-based solutions with the utilities as well as to further increase the return on investment of on-site physical inspections.

4. Modeling of Spatio-Temporal Behavior

A temporal process, such as a Hawkes process [95], models the occurrence of an event that depends on previous events. Hawkes processes include self-excitement, meaning that once an event happens, that event is more likely to happen in the near future again and decays over time. In other words, the further back the event in the process, the less impact it has on future events. The dynamics of Hawkes processes look promising for modeling NTL: Our first hypothesis is that once customers were found to steal electricity, finding them or their neighbors to commit theft again is more likely in the near future and decays over time. A Hawkes process allows to model this first hypothesis. Our second hypothesis is that once customers were found to steal electricity, they are aware of inspections and subsequently are less likely to commit further electricity theft. Therefore, finding them or their neighbors to commit theft again is more likely in the far future and increases over time as they become less risk-aware. As a consequence, we need to extend a Hawkes process by incorporating both, self-excitement in order to model the first hypothesis, as well as self-regulation in order to model the second hypothesis. Only few works have been reported on modeling anomaly detection using self-excitement and self-regulation, such as finding faulty electrical equipment in subway systems in [50].

We have shown in Chapter 5 that the neighborhood is essential from our point of view as neighbors are likely to share their knowledge of NTL as well as the outcome of inspections with their neighbors. We therefore want to extend that model by optimizing the number of temporal processes to be used. In the most trivial case, one temporal process could be used for all customers combined. However, this would lead to a model that underfits, meaning it would not be able to distinguish among the different fraudulent behaviors. In contrast, each customer could be modeled by a dedicated temporal process. However, this would not allow to catch the relevant dynamics, as most fraudulent customers were only found to steal once. Furthermore, the computational costs of this approach would not be feasible. Therefore, we suggest to cluster customers based on their location and then to train one temporal process on the customers of each cluster. Finally, for each cluster, the conditional intensity of its temporal process at a given time can then be used as a feature for the respective customers. In order to find reasonable clusters, we suggest to pose an optimization problem which includes the number of clusters, i.e. the number of temporal processes to train, as well as the sum of prediction errors of all customers.

7.3 Further Measures beyond Artificial Intelligence

In this thesis, we have discussed a broad number of causes of NTL. The predominant approach to detect NTL is to employ artificial intelligence methods for determining the most suspicious customers for inspection. However, only carrying out inspections is not enough. In this section, we review and discuss a number of possible measures for further reducing NTL. We are confident that once electric utilities take these measures into account, they can further reduce NTL and thus increase revenue, profit and reliability of the grid.

1. Legal Actions and Market Reforms

First of all, it is important to make sure that the criminal law of a jurisdiction includes electricity theft. In order to emphasize this issue, we have discussed historical examples from Germany and France in Chapter 1. However, even if electricity theft is included, not all customers in emerging markets may be aware of the legal framework of a country. [43] reasons that it therefore may be important to make them aware of electricity theft being illegal. Next, countries should enforce legal actions and sentences. Prohibitive actions are also recommended in [8], including making major frauds public.

Furthermore, market reforms have been suggested, including privatization. It is argued in [8] that private utilities need to generate profit and that they are thus more willing to reduce NTL. Other studies, such as [90], [178], indicate that an income-aware approach to pricing might offset differences within a country and may thus reduce NTL. Typically, NTL reduction efforts face the problem of having to find irregular customers of electrical energy and correctly assessing the scale of the losses incurred. In case of correct, yet unpaid bills, these problems do not exist. Instead, a regular customer who has been charged correctly does not pay the bills. As a measure, the electric utility can limit the losses by denying further delivery of electrical energy. However, [158] notes that this option is not legally available in some countries or can be dangerous for the employees involved. Additionally, if the customer is in an area where NTL are prevalent, this approach bears the risk that the customer mitigates the power cut-off through bypassing it. Furthermore, the cancellation of power supply comes with additional costs for disconnecting the customer. Electric utilities therefore typically attempt to reach an agreement with the customer on a payment plan.

2. Enhanced Infrastructure Components and Streamlined Methods of Payment

The literature also proposes improvements to the infrastructure in order to reduce NTL. For example, meters that are less prone to tampering may be installed. However, [43] explains that these do not allow to prevent all types of theft as customers may still rig wires before the actual meter from the power source. The literature also suggests further improvements such as addition of central observer meters that allow to calculate energy balances and thus to delimitate customers that cause NTL. Alternatively, prepaid meters should be installed for these customers. Prepaid meters are discussed controversially and have been banned in some countries as discussed in [24]. However, it has been shown that over longer periods - taking the high costs for implementation into consideration - prepaid meters have an aggregate positive welfare effect. It is reported in [119] that in Rwanda, where a prepaid meter system rollout was done alongside harsher penalties, NTL was reduced from 40% in 1998 to 2% in 2008. Additionally, [158] suggests the use of factoring, a process in which a business sells its bills receivable to a third party at a discount. Factoring allows electric utilities to increase effectiveness in payment collection. Furthermore, they should offer additional methods and places for bill payment.

3. Improvement of Data Quality

During the research carried out for this thesis and discussions with the partner company and as argued in Chapter 3.4.3, we noticed that the inspection result labels in the training may not always be correct and that some fraudsters may be labelled as non-fraudulent. The reasons for this may include bribing, blackmailing or threatening of the technicians performing the inspection. Also, the fraud may be done too well and is therefore not observable by technicians. Contrary, discussions with the partner company have revealed that inspectors may also have to satisfy a certain number of NTL found through inspections in a given time frame. Therefore, after careful review of some of the data, it has been assumed that some inspectors may manipulate or break infrastructure on purpose in order to incorrectly report NTL and thus satisfy the quota required. It must be noted that another reason for causing both false positive and false negatives may be incorrect processing of the data. The latter reason may, therefore, also label non-fraudulent behavior as fraudulent.

As a consequence, the utilities need to improve their data collection, both the inspection results, but also the meter readings, which may be incorrect, too. Furthermore, the utilities need to improve their extract, transform, load (ETL) processes in order to reduce erroneous data processing to a minimum.

Bibliography

[1] A. Abaide, L. Canha, A Barin and G Cassel, “Assessment of the smart grids applied in reducing the cost of distribution system losses”, in *Energy Market (EEM), 2010 7th International Conference on the European*, IEEE, 2010, pp. 1–6.

[2] A. Abraham, “Adaptation of fuzzy inference system using neural learning”, in *Fuzzy systems engineering*, Springer, 2005, pp. 53–83.

[3] R. Agrawal and R. Srikant, “Fast algorithms for mining association rules”, in *Proc. 20th int. conf. very large data bases, VLDB*, vol. 1215, 1994, pp. 487–499.

[4] N. S. Altman, “An introduction to kernel and nearest-neighbor nonparametric regression”, *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[5] T. B. Andersen and C.-J. Dalgaard, “Power outages and economic growth in Africa”, *Energy Economics*, vol. 38, pp. 19–23, 2013.

[6] E. W. S. Angelos, O. R. Saavedra, O. A. C. Cortés and A. N. de Souza, “Detection and identification of abnormalities in customer consumptions in power distribution systems”, *IEEE Transactions on Power Delivery*, vol. 26, no. 4, pp. 2436–2442, 2011.

[7] P. Angelov, *Autonomous learning systems: From data streams to knowledge in real-time.* John Wiley & Sons, 2012.

[8] P. Antmann, *Reducing technical and non-technical losses in the power sector.* World Bank, Washington, DC, 2009.

[9] L. Arango, E. Deccache, B. Bonatto, H. Arango and E. Pamplona, “Study of electricity theft impact on the economy of a regulated electricity company”, *Journal of Control, Automation and Electrical Systems*, vol. 28, no. 4, pp. 567–575, 2017.

[10] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller and W. Samek, “On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation”, *PloS one*, vol. 10, no. 7, e0130140, 2015.

[11] M. Banko and E. Brill,
"Scaling to very very large corpora for natural language disambiguation",
in *Proceedings of the 39th annual meeting on association for computational linguistics*, Association for Computational Linguistics, 2001, pp. 26–33.

[12] G. E. Batista, A. L. Bazzan and M. C. Monard,
"Balancing training data for automated annotation of keywords: A case study.",
in *WOB*, 2003, pp. 10–18.

[13] S. Ben-David, J. Blitzer, K. Crammer and F. Pereira,
"Analysis of representations for domain adaptation",
in *Advances in neural information processing systems*, 2007, pp. 137–144.

[14] B. Bhatia and M. Gulati, "Reforming the power sector, controlling electricity theft and improving revenue", *Public Policy for the Private Sector*, 2004.

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

[16] C. M. Bishop, "Pattern recognition and machine learning", 2006.

[17] R. J. Bolton and D. J. Hand, "Statistical fraud detection: A review",
Statistical science, pp. 235–249, 2002.

[18] N. Bostrom and E. Yudkowsky, "The ethics of artificial intelligence",
The Cambridge handbook of artificial intelligence, vol. 316, p. 334, 2014.

[19] L. Breiman, *Classification and regression trees.* Routledge, 2017.

[20] M. C. Bryson, "The literary digest poll: Making of a statistical myth",
The American Statistician, vol. 30, no. 4, pp. 184–185, 1976.

[21] J. Cabral, J. Pinto, E. Gontijo and J Reis,
"Rough sets based fraud detection in electrical energy consumers",
in *WSEAS International Conference on MATHEMATICS AND COMPUTERS IN PHYSICS, Cancun, Mexico*, vol. 2, 2004, pp. 413–416.

[22] J. E. Cabral, J. O. Pinto and A. M. Pinto, "Fraud detection system for high and low voltage electricity consumers based on data mining",
in *2009 IEEE Power & Energy Society General Meeting*, IEEE, 2009, pp. 1–5.

[23] L.-J. Cao and F. E. H. Tay, "Support vector machine with adaptive parameters in financial time series forecasting",
IEEE Transactions on neural networks, vol. 14, no. 6, pp. 1506–1518, 2003.

[24] A. A. Casarin and L. Nicollier, "Prepaid meters in electricity. A cost-benefit analysis", in *Private utilities and poverty alleviation: Market initiatives at the base of the pyramid*, P. C. Marquez and C. Rufin, Eds., Edward Elgar Publishing, 2011, ch. 6, pp. 108–133.

[25] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines", *ACM Transactions on Intelligent Systems and Technology*, vol. 2, 27:1–27:27, 3 2011.

[26] A. Chauhan and S. Rajvanshi, "Non-technical losses in power system: A review", in *Power, Energy and Control (ICPEC), 2013 International Conference on*, IEEE, 2013, pp. 558–561.

[27] N. V. Chawla, K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique", *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[28] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system", in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, ACM, 2016, pp. 785–794.

[29] M. Christ, N. Braun, J. Neuffer and A. W. Kempa-Liehr, "Time series feature extraction on basis of scalable hypothesis tests (tsfresh–a python package)", *Neurocomputing*, 2018.

[30] M. Christ, A. W. Kempa-Liehr and M. Feindt, "Distributed and parallel time series feature extraction for industrial big data applications", *ArXiv preprint arXiv:1610.07717*, 2016.

[31] M. Corporation, *Microsoft hololens*, `http://www.microsoft.com/en-us/hololens`, [Online; accessed July 3, 2017], 2016.

[32] ——, *Holotoolkit-unity*, `http://github.com/Microsoft/HoloToolkit-Unity`, 2017.

[33] C. Cortes and M. Mohri, "Domain adaptation and sample bias correction theory and algorithm for regression", *Theoretical Computer Science*, vol. 519, pp. 103–126, 2014.

[34] B. C. Costa, B. L. Alberto, A. M. Portela, W Maduro and E. O. Eler, "Fraud detection in electric power distribution networks using an ann-based knowledge-discovery process", *International Journal of Artificial Intelligence & Applications*, vol. 4, no. 6, p. 17, 2013.

[35] E. Costa, F. Fabris, A. R. Loureiros, H. Ahonen and F. M. Varejão, “Optimization metaheuristics for minimizing variance in a real-world statistical application”, in *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, ACM, 2013, pp. 206–207.

[36] D. R. Cox, “The regression analysis of binary sequences”, *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 215–242, 1958.

[37] S. Curtis, *Google photos labels black people as gorillas. the telegraph*, http://www.telegraph.co.uk/technology/google/11710136/Google-Photos-assigns-gorilla-tag-to-photos-of-black-people.html, [Online; accessed December 28, 2017], 2015.

[38] K. Dasgupta, M. Padmanaban and J. Hazra, “Power theft localisation using voltage measurements from distribution feeder nodes”, *IET Generation, Transmission & Distribution*, vol. 11, no. 11, pp. 2831–2839, 2017.

[39] A. De Brébisson, E. Simon, A. Auvolat, P. Vincent and Y. Bengio, “Artificial neural networks applied to taxi destination prediction”, in *Proceedings of the 2015th International Conference on ECML PKDD Discovery Challenge - Volume 1526*, ser. ECMLPKDDDC’15, Porto, Portugal: CEUR-WS.org, 2015, pp. 40–51. [Online]. Available: `http://dl.acm.org/citation.cfm?id=3056172.3056178`.

[40] L. T. DeCarlo, “On the meaning and use of kurtosis.”, *Psychological methods*, vol. 2, no. 3, p. 292, 1997.

[41] A. P. Dempster, N. M. Laird and D. B. Rubin, “Maximum likelihood from incomplete data via the em algorithm”, *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[42] L. Deng and D. Yu, “Deep learning: Methods and applications”, *Foundations and Trends® in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, 2014.

[43] S. S. S. R. Depuru, L. Wang and V. Devabhaktuni, “Electricity theft: Overview, issues, prevention and a smart meter based approach to control theft”, *Energy Policy*, vol. 39, no. 2, pp. 1007–1015, 2011.

[44] ——, “Support vector machine based data classification for detection of electricity theft”, in *Power Systems Conference and Exposition (PSCE), 2011 IEEE/PES*, IEEE, 2011, pp. 1–8.

[45] ——, “Enhanced encoding technique for identifying abnormal energy usage pattern”, in *North American Power Symposium (NAPS), 2012*, IEEE, 2012, pp. 1–6.

[46] S. S. S. R. Depuru, L. Wang, V. Devabhaktuni and R. C. Green, "High performance computing for detection of electricity theft", *International Journal of Electrical Power & Energy Systems*, vol. 47, pp. 21–30, 2013.

[47] M. Di Martino, F. Decia, J. Molinelli and A. Fernández, "Improving electric fraud detection using class imbalance strategies.", in *ICPRAM (2)*, 2012, pp. 135–141.

[48] M. Dijk, R. J. Orsato and R. Kemp, "The emergence of an electric mobility trajectory", *Energy Policy*, vol. 52, pp. 135–145, 2013.

[49] P. Domingos, "A few useful things to know about machine learning", *Communications of the ACM*, vol. 55, no. 10, pp. 78–87, 2012.

[50] Ş. Ertekin, C. Rudin and T. H. McCormick, "Reactive point processes: A new approach to predicting power failures in underground electrical systems", *The Annals of Applied Statistics*, vol. 9, no. 1, pp. 122–144, 2015.

[51] G. Evans, J. Miller, M. I. Pena, A. MacAllister and E. Winer, "Evaluating the microsoft hololens through an augmented reality assembly application", in *SPIE Defense+ Security*, International Society for Optics and Photonics, 2017, pp. 101970V–101970V.

[52] T. Fawcett, "An introduction to roc analysis", *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.

[53] M. Fernández-Delgado, E. Cernadas, S. Barro and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?", *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3133–3181, 2014.

[54] R. A. Fisher, "On the interpretation of chi 2 from contingency tables, and the calculation of p", *Journal of the Royal Statistical Society*, vol. 85, no. 1, pp. 87–94, 1922.

[55] T.-c. Fu, "A review on time series data mining", *Engineering Applications of Artificial Intelligence*, vol. 24, no. 1, pp. 164–181, 2011.

[56] M. Galetzka and P. Glauner, "A simple and correct even-odd algorithm for the point-in-polygon problem for complex polygons", in *Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2017), Volume 1: GRAPP*, 2017.

[57] P. Garraud, *La preuve par indices: Dans le procés penal.* Recuil Sirey, 1913.

[58] P. Glauner, "Deep learning for smile recognition", in *Uncertainty Modelling in Knowledge Engineering and Decision Making: Proceedings of the 12th International FLINS Conference*, World Scientific, 2016, pp. 319–324.

[59] ——, “Künstliche Intelligenz - die nächste Revolution (The Artificial Intelligence Revolution)”, in *Innovationsumgebungen gestalten: Impulse für Start-ups und etablierte Unternehmen im globalen Wettbewerb*, P. Plugmann, Ed., Springer, 2018.

[60] P. Glauner, A. Boechat, L. Dolberg, R. State, F. Bettinger, Y. Rangoni and D. Duarte, “Large-scale detection of non-technical losses in imbalanced data sets”, in *Proceedings of the Seventh IEEE Conference on Innovative Smart Grid Technologies (ISGT 2016)*, IEEE, 2016.

[61] P. Glauner, N. Dahringer, O. Puhachov, J. Meira, P. Valtchev, R. State and D. Duarte, “Identifying irregular power usage by turning predictions into holographic spatial visualizations”, in *Proceedings of the 17th IEEE International Conference on Data Mining Workshops (ICDMW 2017)*, IEEE, 2017, pp. 258–265.

[62] P. Glauner, M. Du, V. Paraschiv, A. Boytsov, I. Lopez Andrade, J. Meira, P. Valtchev and R. State, “The top 10 topics in machine learning revisited: A quantitative meta-study”, in *Proceedings of the 25th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2017)*, 2017.

[63] P. Glauner, J. Meira, L. Dolberg, R. State, F. Bettinger, Y. Rangoni and D. Duarte, “Neighborhood features help detecting non-technical losses in big data sets”, in *Proceedings of the 3rd IEEE/ACM International Conference on Big Data Computing Applications and Technologies (BDCAT 2016)*, 2016.

[64] P. Glauner, J. Meira and R. State, “Introduction to detection of non-technical losses using data analytics”, in *7th IEEE Conference on Innovative Smart Grid Technologies, Europe (ISGT Europe 2017)*, 2017.

[65] ——, “Detection of irregular power usage using machine learning”, in *IEEE Conference on Innovative Smart Grid Technologies, Asia (ISGT Asia 2018)*, 2018.

[66] ——, “Machine learning for data-driven smart grid applications”, in *IEEE Conference on Innovative Smart Grid Technologies, Asia (ISGT Asia 2018)*, 2018.

[67] P. Glauner, J. Meira, P. Valtchev, R. State and F. Bettinger, “The challenge of non-technical loss detection using artificial intelligence: A survey”, *International Journal of Computational Intelligence Systems*, vol. 10, no. 1, pp. 760–775, 2017.

[68] P. Glauner, A. Migliosi, J. Meira, P. Valtchev, R. State and F. Bettinger, “Is big data sufficient for a reliable detection of non-technical losses?”, in *Proceedings of the 19th International Conference on Intelligent System Applications to Power Systems (ISAP 2017)*, IEEE, 2017.

[69] P. Glauner and R. State, “Deep learning on big data sets in the cloud with apache spark and google tensorflow”, in *9th IEEE/ACM International Conference on Utility and Cloud Computing (UCC 2016)*, 2016.

[70] ——, “Load forecasting with artificial intelligence on big data”, in *Sixth IEEE Conference on Innovative Smart Grid Technologies, Europe (ISGT Europe 2016)*, 2016.

[71] ——, “Introduction to machine learning for power engineers”, in *10th IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC 2018)*, 2018.

[72] P. Glauner, R. State, P. Valtchev and D. Duarte, “On the reduction of biases in big data sets for the detection of irregular power usage”, in *Proceedings 13th International FLINS Conference on Data Science and Knowledge Engineering for Sensing Decision Support (FLINS 2018)*, 2018.

[73] P. Glauner, P. Valtchev and R. State, “Impact of biases in big data”, in *Proceedings of the 26th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2018)*, 2018.

[74] M. Golden and B. Min, “Theft and loss of electricity in an Indian state”, International Growth Centre, Tech. Rep., 2012.

[75] T. Harford, *Big data: Are we making a big mistake? ft magazine*, http://www.ft.com/intl/cms/s/2/21a6e7d8-b479-11e3-a09a-00144feabdc0.html, [Online; accessed January 15, 2016], 2014.

[76] J. J. Heckman, “Sample selection bias as a specification error”, *Econometrica*, vol. 47, no. 1, pp. 153–161, 1979, ISSN: 00129682, 14680262. [Online]. Available: `http://www.jstor.org/stable/1912352`.

[77] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups”, *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[78] G. E. Hinton, S. Osindero and Y.-W. Teh, “A fast learning algorithm for deep belief nets”, *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[79] T. K. Ho, “Random decision forests”, in *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*, IEEE, vol. 1, 1995, pp. 278–282.

[80] S. Hochreiter and J. Schmidhuber, “Long short-term memory”, *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[81] M. A. Hoffman, “The future of three-dimensional thinking”, *Science*, vol. 353, no. 6302, pp. 876–876, 2016.

[82] W. B. van den Hout, “The area under an roc curve with limited information”, *Medical decision making*, vol. 23, no. 2, pp. 160–166, 2003.

[83] G. Hughes, “On the mean accuracy of statistical pattern recognizers”, *IEEE transactions on information theory*, vol. 14, no. 1, pp. 55–63, 1968.

[84] J. P. Ioannidis, “Why most published research findings are false”, *PLoS medicine*, vol. 2, no. 8, e124, 2005.

[85] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift”, in *International Conference on Machine Learning*, 2015, pp. 448–456.

[86] S. Jahan *et al.*, *Human Development Report 2016: Human Development for Everyone.* United Nations Publications, 2016, ISBN: 9789211264135.

[87] N. Japkowicz and S. Stephen, “The class imbalance problem: A systematic study”, *Intelligent data analysis*, vol. 6, no. 5, pp. 429–449, 2002.

[88] J. Jiang, “A literature survey on domain adaptation of statistical classifiers”, 2008.

[89] R. Jiang, R. Lu, Y. Wang, J. Luo, C. Shen and X. S. Shen, “Energy-theft detection issues for advanced metering infrastructure in smart grid”, *Tsinghua Science and Technology*, vol. 19, no. 2, pp. 105–120, 2014.

[90] S. K. Katiyar, “Political economy of electricity theft in rural areas: A case study from Rajasthan”, *Economic and Political Weekly*, pp. 644–648, 2005.

[91] M. Kazerooni, H. Zhu and T. J. Overbye, “Literature review on the applications of data mining in power systems”, in *Power and Energy Conference at Illinois (PECI)*, IEEE, 2014, pp. 1–8.

[92] Y. Kou, C.-T. Lu, S. Sirwongwattana and Y.-P. Huang, “Survey of fraud detection techniques”, in *Networking, sensing and control, 2004 IEEE international conference on*, IEEE, vol. 2, 2004, pp. 749–754.

[93] S. Kullback, “Letter to the editor: The kullback-leibler distance”, *The American Statistician*, 1987.

[94] K. Lackner, K. Kühl, E. Dreher and H. Maassen, *Strafgesetzbuch (StGB): Kommentar.* C.H. Beck, 2014.

[95] P. J. Laub, T. Taimre and P. K. Pollett, “Hawkes processes”, *ArXiv preprint arXiv:1507.02822*, 2015.

[96] B. B. Le Cun, J. S. Denker, D Henderson, R. E. Howard, W Hubbard and L. D. Jackel, “Handwritten digit recognition with a back-propagation network”, in *Advances in neural information processing systems*, Citeseer, 1990.

[97] Y. LeCun, Y. Bengio and G. Hinton, “Deep learning”, *Nature*, vol. 521, no. 7553, p. 436, 2015.

[98] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, “Gradient-based learning applied to document recognition”, *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[99] Y. LeCun, P. Haffner, L. Bottou and Y. Bengio, “Object recognition with gradient-based learning”, in *Shape, contour and grouping in computer vision*, Springer, 1999, pp. 319–345.

[100] F. B. Lewis, “Costly ’throw-ups’: Electricity theft and power disruptions”, *The Electricity Journal*, vol. 28, no. 7, pp. 118–135, 2015.

[101] P. Linares and L. Rey, “The costs of electricity interruptions in Spain. Are we sending the right signals?”, *Energy Policy*, vol. 61, pp. 751–760, 2013.

[102] X.-Y. Liu, J. Wu and Z.-H. Zhou, “Exploratory undersampling for class-imbalance learning”, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp. 539–550, 2009.

[103] Z. Lv, X. Li, B. Zhang, W. Wang, Y. Zhu, J. Hu and S. Feng, “Managing big city information based on webvrgis”, *IEEE Access*, vol. 4, pp. 407–415, 2016.

[104] J. MacQueen, “Some methods for classification and analysis of multivariate observations”, in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Oakland, CA, USA, vol. 1, 1967, pp. 281–297.

[105] I. Mani and I Zhang, “Knn approach to unbalanced data distributions: A case study involving information extraction”, in *Proceedings of workshop on learning from imbalanced datasets*, vol. 126, 2003.

[106] F. J. Martin, *A simple machine learning method to detect covariate shift*, `http://blog.bigml.com/2014/01/03/simple-machine-learning-to-detect-covariate-shift/`, [Online; accessed January 15, 2016], 2014.

[107] F. J. Massey Jr, “The kolmogorov-smirnov test for goodness of fit”, *Journal of the American statistical Association*, vol. 46, no. 253, pp. 68–78, 1951.

[108] *Matlab Fuzzy Logic Toolbox User’s Guide*. Mat Works, 2015.

[109] B. W. Matthews, “Comparison of the predicted and observed secondary structure of t4 phage lysozyme”, *Biochimica et Biophysica Acta (BBA)-Protein Structure*, vol. 405, no. 2, pp. 442–451, 1975.

[110] J. McCarthy, M. L. Minsky, N. Rochester and C. E. Shannon, “A proposal for the dartmouth summer research project on artificial intelligence”, 1955.

[111] S. McLaughlin, D. Podkuiko and P. McDaniel, "Energy theft in the advanced metering infrastructure", in *International Workshop on Critical Information Infrastructures Security*, Springer, 2009, pp. 176–187.

[112] J. Meira, P. Glauner, R. State, P. Valtchev, L. Dolberg, F. Bettinger and D. Duarte, "Distilling provider-independent data for general detection of non-technical losses", in *Power and Energy Conference at Illinois (PECI)*, IEEE, 2017.

[113] G. M. Messinis and N. D. Hatziargyriou, "Review of non-technical loss detection methods", *Electric Power Systems Research*, vol. 158, pp. 250 –266, 2018, ISSN: 0378-7796. DOI: `http://doi.org/10.1016/j.epsr.2018.01.005`. [Online]. Available: `http://www.sciencedirect.com/science/article/pii/S0378779618300051`.

[114] Met Office, *Cartopy: A cartographic python library with a matplotlib interface*, Exeter, Devon, 2010 - 2015. [Online]. Available: `http://scitools.org.uk/cartopy`.

[115] T. M. Mitchell, "Machine learning. 1997", *Burr Ridge, IL: McGraw Hill*, vol. 45, no. 37, pp. 870–877, 1997.

[116] C. Muniz, K. Figueiredo, M. Vellasco, G. Chavez and M. Pacheco, "Irregularity detection on low tension electric installations by neural network ensembles", in *2009 International Joint Conference on Neural Networks*, IEEE, 2009, pp. 2176–2182.

[117] C. Muniz, M. M. B. R. Vellasco, R. Tanscheit and K. Figueiredo, "A neuro-fuzzy system for fraud detection in electricity distribution.", in *IFSA/EUSFLAT Conf.*, Citeseer, 2009, pp. 1096–1101.

[118] C. H. Museum and K. television, *Chm revolutionaries: the challenge & promise of artificial intelligence*, http://www.youtube.com/watch?v=rtmQ3xlt-4A, [Online; accessed July 18, 2018], 2016.

[119] F. M. Mwaura, "Adopting electricity prepayment billing system to reduce non-technical energy losses in Uganda: Lesson from Rwanda", *Utilities Policy*, vol. 23, pp. 72–79, 2012.

[120] J Nagi, A. Mohammad, K. Yap, S. Tiong and S. Ahmed, "Non-technical loss analysis for detection of electricity theft using support vector machines", in *Power and Energy Conference, 2008. PECon 2008. IEEE 2nd International*, IEEE, 2008, pp. 907–912.

[121] J Nagi, K. Yap, F Nagi, S. Tiong, S. Koh and S. Ahmed, “Ntl detection of electricity theft and abnormalities for large power consumers in tnb malaysia”, in *Research and Development (SCOReD), 2010 IEEE Student Conference on*, IEEE, 2010, pp. 202–206.

[122] J Nagi, K. Yap, S. Tiong, S. Ahmed and A. Mohammad, “Detection of abnormalities and electricity theft using genetic support vector machines”, in *TENCON 2008-2008 IEEE Region 10 Conference*, IEEE, 2008, pp. 1–6.

[123] J. Nagi, K. S. Yap, S. K. Tiong, S. K. Ahmed and M. Mohamad, “Nontechnical loss detection for metered customers in power utility using support vector machines”,
IEEE transactions on Power Delivery, vol. 25, no. 2, pp. 1162–1171, 2010.

[124] J. Nagi, K. S. Yap, S. K. Tiong, S. K. Ahmed and F. Nagi, “Improving svm-based nontechnical loss detection in power utility using the fuzzy inference system”, *IEEE Transactions on power delivery*, vol. 26, no. 2, pp. 1284–1285, 2011.

[125] A. Y. Ng, “Feature selection, l 1 vs. l 2 regularization, and rotational invariance”, in *Proceedings of the twenty-first international conference on Machine learning*, ACM, 2004, p. 78.

[126] A. Nizar, Z. Dong and Y Wang, “Power utility nontechnical loss analysis with extreme learning machine method”,
IEEE Transactions on Power Systems, vol. 23, no. 3, pp. 946–955, 2008.

[127] A. H. Nizar, J. H. Zhao and Z. Y. Dong,
“Customer information system data pre-processing with feature selection techniques for non-technical losses prediction in an electricity market”, in *2006 International Conference on Power System Technology*, IEEE, 2006, pp. 1–7.

[128] E. Olshannikova, A. Ometov, Y. Koucheryavy and T. Olsson, “Visualizing big data with augmented and virtual reality: Challenges and research agenda”, *Journal of Big Data*, vol. 2, no. 1, p. 22, 2015.

[129] L. Page, S. Brin, R. Motwani and T. Winograd,
“The pagerank citation ranking: Bringing order to the web.”, Stanford InfoLab, Tech. Rep., 1999.

[130] K. Pearson, “Liii. on lines and planes of closest fit to systems of points in space”, *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.

[131] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in Python", *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[132] R. Penrose, "A generalized inverse for matrices", in *Mathematical proceedings of the Cambridge philosophical society*, Cambridge Univ Press, vol. 51, 1955, pp. 406–413.

[133] "People who steal Edison's electricity", *Daily Yellowstone Journal*, p. 2, 1886, [March 27 edition].

[134] J. Porras, H. Rivera, F. Giraldo and B. Correa, "Identification of non-technical electricity losses in power distribution systems by applying techniques of information analysis and visualization", *IEEE Latin America Transactions*, vol. 13, no. 3, pp. 659–664, 2015.

[135] M. F. Porter, "Readings in information retrieval", in, K. Sparck Jones and P. Willett, Eds., San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1997, ch. An Algorithm for Suffix Stripping, pp. 313–316, ISBN: 1-55860-454-5. [Online]. Available: `http://dl.acm.org/citation.cfm?id=275537.275705`.

[136] U. S. E. O. of the President and J. Podesta, *Big data: Seizing opportunities, preserving values.* White House, Executive Office of the President, 2014.

[137] J. R. Quinlan, "Induction of decision trees", *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[138] J. R. Quinlan, "C4.5: Programming for machine learning", *Morgan Kauffmann*, p. 38, 1993.

[139] P. Radivojac, Z. Obradovic, A. K. Dunker and S. Vucetic, "Feature selection filters based on the permutation test", in *ECML*, Springer, 2004, pp. 334–346.

[140] C. C. O. Ramos, A. N. De Souza, D. S. Gastaldello and J. P. Papa, "Identification and feature selection of non-technical losses for industrial consumers using the software weka", in *Industry Applications (INDUSCON), 2012 10th IEEE/IAS International Conference on*, IEEE, 2012, pp. 1–6.

[141] C. C. O. Ramos, A. N. de Sousa, J. P. Papa and A. X. Falcao, "A new approach for nontechnical losses detection based on optimum-path forest", *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 181–189, 2011.

[142] C. C. Ramos, D. Rodrigues, A. N. de Souza and J. P. Papa, "On the study of commercial losses in Brazil: A binary black hole algorithm for theft characterization", *IEEE Transactions on Smart Grid*, vol. 9, no. 2, pp. 676–683, 2018.

[143] C. C. Ramos, A. N. Souza, J. P. Papa and A. X. Falcao, "Fast non-technical losses identification through optimum-path forest", in *Intelligent System Applications to Power Systems, 2009. ISAP'09. 15th International Conference on*, IEEE, 2009, pp. 1–5.

[144] C. C. Ramos, A. N. Souza, J. P. Papa and A. X. Falcão, "Learning to identify non-technical losses with optimum-path forest", in *Proceedings of the 17th International Conference on Systems, Signals and Image Processing (IWSSIP 2010)*, 2010, pp. 154–157.

[145] M. T. Ribeiro, S. Singh and C. Guestrin, "Why should i trust you?: Explaining the predictions of any classifier", in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, ACM, 2016, pp. 1135–1144.

[146] I. Rish, "An empirical study of the naive bayes classifier", in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, IBM New York, vol. 3, 2001, pp. 41–46.

[147] L. S. Riza, C. N. Bergmeir, F. Herrera and J. M. Benítez Sánchez, "Frbs: Fuzzy rule-based systems for classification and regression in r", American Statistical Association, 2015.

[148] S. Rose, D. Engel, N. Cramer and W. Cowley, "Automatic keyword extraction from individual documents", in *Text Mining.* John Wiley & Sons, Ltd, 2010, pp. 1–20, ISBN: 9780470689646. DOI: `10.1002/9780470689646.ch1`. [Online]. Available: `http://dx.doi.org/10.1002/9780470689646.ch1`.

[149] S. J. Russell and P. Norvig, *Artificial intelligence: A modern approach (3rd edition)*, 2009.

[150] S. Sabour, N. Frosst and G. E. Hinton, "Dynamic routing between capsules", in *Advances in Neural Information Processing Systems*, 2017, pp. 3859–3869.

[151] S. Sahoo, D. Nikovski, T. Muso and K. Tsuru, "Electricity theft detection using smart meter data", in *Innovative Smart Grid Technologies Conference (ISGT), 2015 IEEE Power & Energy Society*, IEEE, 2015, pp. 1–5.

[152] H. Sak, A. Senior and F. Beaufays, “Long short-term memory recurrent neural network architectures for large scale acoustic modeling”, in *Fifteenth annual conference of the international speech communication association*, 2014.

[153] A. L. Samuel, “Some studies in machine learning using the game of checkers”, *IBM Journal of research and development*, vol. 3, no. 3, pp. 210–229, 1959.

[154] M. Sayed-Mouchaweh and E. Lughofer, *Learning in non-stationary environments: Methods and applications.* Springer Science & Business Media, 2012.

[155] M. Shanahan, *The technological singularity.* MIT Press, 2015.

[156] H. Shimodaira, “Improving predictive inference under covariate shift by weighting the log-likelihood function”, *Journal of statistical planning and inference*, vol. 90, no. 2, pp. 227–244, 2000.

[157] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, “Mastering the game of go with deep neural networks and tree search”, *Nature*, vol. 529, no. 7587, p. 484, 2016.

[158] T. B. Smith, “Electricity theft: A comparative analysis”, *Energy Policy*, vol. 32, no. 18, pp. 2067–2076, 2004.

[159] J. V. Spirić, S. S. Stanković, M. B. Dočić and T. D. Popović, “Using the rough set theory to detect fraud committed by electricity customers”, *International Journal of Electrical Power & Energy Systems*, vol. 62, pp. 727–734, 2014.

[160] W. M. Stanish and N. Taylor, “Estimation of the intraclass correlation coefficient for the analysis of covariance model”, *The American Statistician*, vol. 37, no. 3, pp. 221–224, 1983.

[161] Y. Tang, Y.-Q. Zhang, N. V. Chawla and S. Krasser, “Svms modeling for highly imbalanced classification”, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 1, pp. 281–288, 2009.

[162] R. Targosz and J. Manson, “Pan-European power quality survey”, in *Electrical Power Quality and Utilisation, 2007. EPQU 2007. 9th International Conference on*, IEEE, 2007, pp. 1–6.

[163] A. G. Taylor, “Creating a holographic teaching tool”, in *Develop Microsoft HoloLens Apps Now*, Springer, 2016, pp. 185–193.

[164] I. Tomek, “Two modifications of cnn”, *IEEE Trans. Systems, Man and Cybernetics*, vol. 6, pp. 769–772, 1976.

[165] F. C. Trindade, L. F. Ochoa and W. Freitas, "Data analytics in smart distribution networks: Applications and challenges", in *Innovative Smart Grid Technologies-Asia (ISGT-Asia), 2016 IEEE*, IEEE, 2016, pp. 574–579.

[166] A. Turing, "Computing machinery and intelligence", *Mind*, vol. 59, no. 236, pp. 433–460, 1950.

[167] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior and K. Kavukcuoglu, "Wavenet: A generative model for raw audio.", in *SSW*, 2016, p. 125.

[168] V. N. Vapnik, "An overview of statistical learning theory", *IEEE transactions on neural networks*, vol. 10, no. 5, pp. 988–999, 1999.

[169] A. Verikas, E. Vaiciukynas, A. Gelzinis, J. Parker and M. C. Olsson, "Electromyographic patterns during golf swing: Activation sequence profiling and prediction of shot effectiveness", *Sensors*, vol. 16, no. 4, p. 592, 2016.

[170] J. L. Viegas, P. R. Esteves, R Melício, V. Mendes and S. M. Vieira, "Solutions for detection of non-technical losses in the electricity grid: A review", *Renewable and Sustainable Energy Reviews*, vol. 80, pp. 1256–1268, 2017.

[171] L. Wan, M. Zeiler, S. Zhang, Y. Le Cun and R. Fergus, "Regularization of neural networks using dropconnect", in *International Conference on Machine Learning*, 2013, pp. 1058–1066.

[172] Y. Wang and M. Kosinski, "Deep neural networks are more accurate than humans at detecting sexual orientation from facial images", *Journal of Personality and Social Psychology*, 2017.

[173] Z. Wang and T. Oates, "Imaging time-series to improve classification and imputation", in *Proceedings of the 24th International Conference on Artificial Intelligence*, AAAI Press, 2015, pp. 3939–3945.

[174] P. Werner, F. Saxen and A. Al-Hamadi, "Handling data imbalance in automatic facial action intensity estimation", *FERA*, p. 26, 2015.

[175] C. Williams, *Ai guru ng: Fearing a rise of killer robots is like worrying about overpopulation on mars*, http://www.theregister.co.uk/2015/03/19/andrew_ng_baidu_ai/, [Online; accessed August 1, 2018], 2015.

[176] D. H. Wolpert, "The lack of a priori distinctions between learning algorithms", *Neural computation*, vol. 8, no. 7, pp. 1341–1390, 1996.

[177] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip, Z.-H. Zhou, M. Steinbach, D. J. Hand and D. Steinberg, "Top 10 algorithms in data mining", *Knowledge and Information Systems*, vol. 14, no. 1, pp. 1–37, 2008, Published online: 4 December 2007, ISSN: 0219-3116. DOI: `10.1007/s10115-007-0114-2`. [Online]. Available: `http://dx.doi.org/10.1007/s10115-007-0114-2`.

[178] Ç. Yurtseven, "The causes of electricity theft: An econometric analysis of the case of Turkey", *Utilities Policy*, vol. 37, pp. 70–78, 2015.

[179] B. Zadrozny, "Learning and evaluating classifiers under sample selection bias", in *Proceedings of the twenty-first international conference on Machine learning*, ACM, 2004, p. 114.

[180] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker and I. Stoica, "Spark: Cluster computing with working sets.", *HotCloud*, vol. 10, pp. 10–10, 2010.

[181] S. Zanola, S. I. Fabrikant and A. Çöltekin, "The effect of realism on the confidence in spatial data quality in stereoscopic 3d displays", in *Proceedings of the 24th international cartography conference (ICC 2009), Santiago, Chile*, 2009, pp. 15–21.