Identification, Categorisation and Forecasting of Court Decisions

Medvedeva, Masha

DOI:
10.33612/diss.236807643

Identification, Categorisation and Forecasting of Court Decisions

Masha Medvedeva

university of groningen

CLCG
Center for Language and Cognition Groningen

bcn

The work in this thesis has been carried out under the auspices of the Center for Language and Cognition Groningen (CLCG) of the Faculty of Arts, the Department of Legal Methods of the Faculty of Law, the Research School of Behavioural and Cognitive Neurosciences (BCN) and the Young Academy Groningen (YAG) of the University of Groningen.

Groningen Dissertations in Linguistics

Document prepared with LaTeX 2$_\varepsilon$ and typeset by pdfTeX
(Gotham, Droid Serif and Lato fonts)
Cover design: Ekaterina Solomatina & Masha Medvedeva
Illustrations: Ekaterina Solomatina
Printed by Ridderprint on G-print 115g paper.

university of groningen

Identification, Categorisation and Forecasting of Court Decisions

PhD thesis

to obtain the degree of PhD at the
University of Groningen
on the authority of the
Rector Magnificus Prof. C. Wijmenga
and in accordance with
the decision by the College of Deans.

This thesis will be defended in public on

Thursday 8 September 2022 at 16.15 hours

by

Maria Medvedeva

born on 21 December 1993
in Moscow, Russia

Supervisors
Prof. M.B. Wieling
Prof. M. Vols

Assessment Committee
Prof. K.D. Ashley
Prof. J.H. Gerards
Prof. H.B. Verheij

Acknowledgements

I will not pretend that getting this dissertation to where it is today was an easy feat, it definitely was not. Diving into a completely new field, learning to navigate between the two disciplines, figuring out the academic world, decrypting legal Dutch, working from home, it was not without challenge. As overwhelming as it may have been at times, it was an exciting and rewarding endeavour, learning, figuring out new things, publishing, presenting and travelling.

Throughout the process, I've been guided and supported by Martijn Wieling and Michel Vols. I couldn't wish for better supervisors. I am not able to express how much of an impact they both had on my academic and personal life. Thank you for caring, and for being so involved. Working on every project, paper, and finally the dissertation, I've always felt like a part of the team. Though this dissertation has my name on the cover, it has been a true collaborative effort.

I would like to thank my reading committee members Kevin Ashley, Bart Verhij and Janneke Gerards for reading my dissertation and providing such valuable feedback.

Being part of two departments, I've had many colleagues over the years who brought endless insights to my work and life. While I will not be able to mention them all here, in no particular order I'd like to thank Joep, Hessel, Michelle, Malvina, Esmee, Barbara, Teja, Mark, Rik, Wilbert, Iris, Lukas, Emma, Caroline, Laurent, Anna, Nynke, Martijn, Leonie, Noor, Thomas, Marije, Gertjan, Gregory, Hati, Muriël, Harry, Roos, Peter, Adriaan, Stefan, Tommaso, Vincent, Lasha, Ahmet,

Chunliu, Andreas, Lisa, Gosse, Dieke, Gabriele, Huiyuan, Johannes, Gosse, Stephan, Prajit, Raoul, Alex, Rob, Wietse, Gera, Silvia, Marjan, Ana, Monique, Antonio, Arianna, Berend, Johan, Linzy, Mareike and many many more.

I would also like to especially thank the indispensable research assistants working with me on different projects throughout the years Xiao, Thijmen, Nynke and Esmee.

I was very lucky to be able to travel for my work in the first half of my PhD journey. In the second half the quarantine allowed me to present over zoom in places where I wouldn't otherwise have had a chance to. I am very grateful for everybody I met along the way, going to conferences, working at festivals and during meetings and zoom calls.

My time working on this research has been made endlessly more pleasant with the support of my friends. I owe endless gratitude to Teja, my paranymph, for being my tea drinking partner and the greatest of friends; Alex, also my paranymph, for caring, being a good laugh, and always having a positive perspective on things; Stefan, who at the peak of my existential crisis agreed to walk 200km in 4 days with me, because I thought it would make for a good joke; Alex, Stefan and Mareike, for providing me with all the gossip at our Monday drinks; Katya, for being my oldest friend and illustrating this book; Puck, for the emotional support during our walks in the park throughout the pandemic. Shout-out to my dear friends Arnab, Irene, Jidde, Joanna, Mara and Ruben, thank you for being there for me.

I would like to thank my ever-expanding family for being such an inspiration and making me who I am. Look, mom, I wrote a book!

My ultimate gratitude goes to my partner Thom. Thank you for loving, inspiring and feeding me, and for being my rock. I wouldn't be where I am without you.

Contents

Contents **vii**

1 Introduction **1**
1.1 Chapter Guide 2
1.2 Final Remarks 3

I Background and definitions **5**

2 Early Quantitative Analyses of Legal Data **7**
2.1 Introduction 7
2.2 Quantitative Analysis of Case Law 9
2.3 Information Extraction 10
2.4 Statistical Analysis 11
2.5 Citation Analysis 11
2.6 Argument Mining 12
2.7 Machine Learning 13

3 Machine Learning Techniques for Legal Text Classification **15**
3.1 Supervised Machine Learning 16
3.2 Feature Vector Representation 18
3.3 Non-neural Machine Learning 22
3.3.1 Decision Trees, Bootstrap Aggregating (Bagging), and Random Forests 23
3.3.2 K-nearest Neighbour 26
3.3.3 Naive Bayes 26
3.3.4 Logistic Regression 29
3.3.5 Support Vector Machines 31
3.4 Neural Networks 32

3.4.1 Convolutional Neural Networks 34
3.4.2 Hierarchical Attention Networks 34
3.4.3 Transformers 35
3.5 Evaluation 36

4 Rethinking the Field of Automatic Prediction of Court Decisions 41
4.1 Introduction 42
4.2 Scope of the Review 43
4.3 Terminology and Types of Judgement Classification . . . 44
4.3.1 Outcome Identification 47
4.3.2 Outcome-based Judgement Categorisation 50
4.3.3 Outcome Forecasting 56
4.4 Discussion 60
4.5 Conclusion 63

II Experiments 65

5 Automatically Identifying Eviction Cases and Outcomes 67
5.1 Introduction 68
5.2 Related Work 69
5.3 Data 70
5.4 Experiment I: Identifying Eviction Related Judgements . 73
5.4.1 Methodology 73
5.4.2 Results 74
5.5 Experiment II: Identifying the Outcome 75
5.5.1 Methodology 75
5.5.2 Results 79
5.6 Discussion and Conclusion 80

6 Using Machine Learning to Categorise Decisions of the ECtHR 83
6.1 Introduction 84
6.2 Data and Methodology 85
6.2.1 Collecting the Data 85

6.2.2 Balanced Dataset 88
6.3 Experiments 90
6.3.1 Experiment 1: Textual Analysis 92
6.3.2 Experiment 2: Categorising the Future 102
6.3.3 Experiment 3: Judges 106
6.4 Discussion 110
6.5 Conclusion 112

7 Automatic Judgement Forecasting for the ECtHR 115
7.1 Introduction 116
7.2 Data 120
7.2.1 The Court 120
7.2.2 Communicated Cases 120
7.2.3 Data Collection 122
7.2.4 Published Dataset 125
7.3 Methodology 126
7.4 Results 128
7.5 Discussion 130
7.5.1 Future Work 135
7.6 Ethical Considerations 137
7.7 Conclusion 137

8 JURI SAYS 139
8.1 Introduction 139
8.2 JURI SAYS 140
8.2.1 Database 140
8.2.2 Machine Learning System 141
8.2.3 Web Platform 143
8.3 Conclusion 143

III Ethical consideration and conclusions 145

9 Innocent until Predicted Guilty 147

9.1 Introduction 147
9.2 Arguments against Automatic Decision-making 150
9.2.1 Decision-predicting vs. Decision-making Systems 151
9.2.2 Status Quo Bias 152
9.2.3 Dangers of Reverse-engineering 152
9.3 Discussion and Conclusion 153

10 Discussion and Conclusion **155**

Bibliography **159**

Summary **177**

Samenvatting **181**

CHAPTER 1

Introduction

The possibility of being able to predict the future can seem very tempting, especially in today's technologically advanced world. It is therefore no surprise that interest in automatically predicting court decisions has risen considerably in recent years.

Our whole lives are regulated by laws, from buying a house to drinking alcohol. As laws are written down, the legal system is entirely dependent on language. Probably one of the most extreme examples of this was a trial at the US Court of Appeals for the First Circuit, in which the lack of an Oxford comma in a law decided a case to compensate dairy truck drivers for their overtime work.[1]

Nowadays, many courts worldwide publish judgements online. This provides many opportunities for legal research. Nonetheless, legal experts and scientific researchers working in law often analyse data by hand, and are thus limited in the amount of information that they can process.

In this dissertation, we address the potential of using language analysis and automatic information extraction to facilitate statistical research in the legal domain. More specifically, we demonstrate and

[1] `https://cases.justia.com/federal/appellate-courts/ca1/16-1901/16-1901-2017-03-13.pdf`, accessed on 04/04/2022

discuss the possibilities of natural language processing (NLP) techniques for the automatic prediction of judicial decisions, as well as their limitations. Using machine learning, we are able to use a computer to perform quantitative analysis on the basis of words and phrases that were used in a court case. Then, based on that analysis, we can 'teach' the computer to predict the decision of the court. If we can predict the results adequately, we may subsequently analyse which words or phrases made the most impact on this decision, and thus identify the factors that are potentially important for judicial decisions.

1.1 Chapter Guide

This dissertation contains ten chapters that are divided into three parts.

Part I - Background and Definitions

Part I discusses previous work in natural legal language processing. Specifically, Chapter 2 discusses a range of previous quantitative (non-machine learning) research conducted on legal data. In Chapter 3, we explain the machine learning terminology applicable to the field, to support the reader in understanding the methodology presented in previous work and our own experiments. Chapter 4 provides a review of the main research in predicting court decisions. In this chapter, we discuss why existing terminology in the field is problematic, and suggest new terminology that better reflects the tasks that have until now been generalised under the term, 'predicting court decisions'. We then discuss how previous research is distributed between these tasks.

Part II - Experiments

Part II describes our experiments as three sub-tasks: the identification, categorisation, and forecasting of court decisions. Chapter 5 describes a method for identifying eviction-related cases and their outcomes, within all the Dutch judiciary case law available online. This chapter also illustrates a practical method for collecting legal datasets on a specific topic, using machine learning. Chapter 6 focuses on NLP methodology for judgement categorisation, to identify factors that may result in finding patterns, as well as a better understanding of judicial decision-making, using the European Court of Human Rights as an example. Chapters 7 is focused on forecasting the (future) decisions of the European Court of Human Rights, using documents published by the court (sometimes) years before a decision is made. Chapter 8 describes an online platform made specifically for this purpose, which highlights the sentences that have contributed most strongly to the system's predictions.

Part III - Conclusions and Ethical Considerations

In part III we discuss ethical concerns associated with systems that deal with case law (Chapter 9), as well as discuss the overall findings of our work and draw conclusions (Chapter 10). While the ethical concerns regarding predicting court decisions has already been widely discussed in the legal community, we hope to introduce a perspective which acknowledges the technological limitations of such systems.

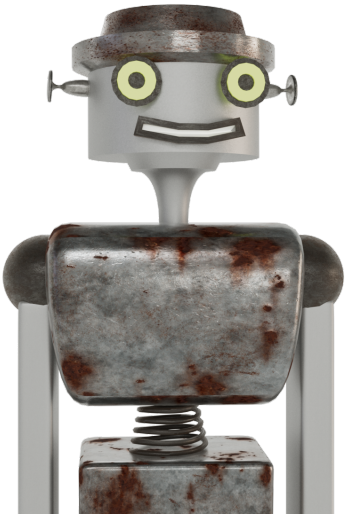
1.2 Final Remarks

We hope that this work is interesting and useful for those interested in legal research, as well as for NLP specialists working with legal data. Consequently, we often go into some detail regarding the algorithms behind various machine learning systems for the benefit

of legal scholars, whereas this information is already familiar to researchers working in NLP. Similarly, for the benefit of NLP specialists, we will try to explain the legal nuances with which legal scholars are already familiar. We choose to do so, because we believe that this interdisciplinary field should always be a collaboration between the two disciplines, where both sides play different roles but are familiar with each other's field. We have noticed throughout our work that, while such collaborations are becoming more common, they are often not considered a prerequisite. Our experience shows that, in the majority of cases, true interdisciplinary collaborations are essential to producing technological systems that provide solutions to existing issues in the legal domain.

PART I

Background and definitions



CHAPTER 2

Early Quantitative Analyses of Legal Data

Chapter based on the introduction of:

Medvedeva, M., Vols, M., and Wieling, M. (2020a). Using machine learning to predict decisions of the European Court of Human Rights. *Artificial Intelligence and Law*, 28:237–266.

2.1 Introduction

Nowadays, when so many courts adhere to the directive to promote accessibility and re-use of public sector information,[1] and publish considered cases online, the door to automatic analysis of legal data stands wide open. However, the idea of automating or semi-automating the legal domain is not new. Search databases for legal data, such as Westlaw and LexisNexis, have existed since the early '90s.

Language analysis also been used in the legal domain and criminology for a long time. For example, text classification has been used

[1] `https://digital-strategy.ec.europa.eu/en/policies/legislation-open-data`, accessed on 11/10/2021

in forensic linguistics. Whereas in earlier times, such as in the Unabomber case,[2] analysis was manual, today we can perform many of these tasks automatically. We now have so-called 'machine learning' software, which is able to identify the gender (Basile et al., 2017), age (op Vollenbroek et al., 2016), personality traits (Golbeck et al., 2011), and even identity of an author almost flawlessly.[3]

For centuries, legal researchers applied doctrinal research methods, which included describing laws, practical problem-solving, and adding interpretative comments to legislation and case law, but also "innovative theory building (systematisation) with the more simple versions of that research being the necessary building blocks for the more sophisticated ones" (Van Hoecke, 2011, p. vi). Doctrinal legal research "provides a systematic exposition of the rules governing a particular legal category, analyses the relationship between rules, explains areas of difficulty and, perhaps, predicts future developments" (Hutchinson and Duncan, 2012, p. 101).

One of the key characteristics of doctrinal analysis of case law is that court decisions are manually collected, read, summarised, commented on, and placed in the overall legal system (Vols, 2021). Historically, quantitative research methods were hardly used to analyse case law (Epstein and Martin, 2010). Nowadays, however, due to the massive amount of published case law, it is physically impossible for legal researchers to read, analyse and systematise all international and national court decisions. In the age of legal big data, more and more researchers are starting to notice that combining traditional doctrinal legal methods and empirical quantitative methods is a promising approach, which will help us make sense of all the available case law (Custers and Leeuw, 2017; Derlén and Lindholm, 2017c; Goanta, 2017;

[2] https://archives.fbi.gov/archives/news/stories/2008/april/unabomber_042408, accessed 04/04/2022

[3] https://github.com/sixhobbits/yelp-dataset-2017, accessed on 04/04/2022

Šadl and Olsen, 2017; Verbruggen, 2021).

2.2 Quantitative Analysis of Case Law

In the United States of America, the quantitative analysis of case law has a longer tradition than in other parts of the world. There are several quantitative studies of datasets consisting of case law from American courts (O'Hear and Wheelock, 2021). Most of these studies use case law which has been manually collected and coded. Many studies use the Supreme Court Database, which contains manually collected and expertly-coded data on the US Supreme Court's behaviour over the last two-hundred years (Spaeth et al., 2014). Many of these studies analyse the relationship between the gender or political background of judges and their decision-making (see Epstein et al., 2013; Rachlinski and Wistrich, 2017; Frankenreiter, 2016).

In countries other than the United States, the use of quantitative methods to analyse case law is not very common (see Vols and Jacobs, 2017). For example, Hunter et al. (2008, 79) state: "This tradition has not been established in the United Kingdom, perhaps because we do not have a sufficient number of judges at the appropriate level who are not male and white to make such statistical analysis worthwhile". Still, researchers have applied quantitative methods to datasets of case law from, for example, Belgium (De Jaeger, 2017), the Czech Republic (Bricker, 2017), France (Sulea et al., 2017), Germany (Dyevre, 2015; Bricker, 2017; Hartung, 2021), Israel (Doron et al., 2015), Japan (Kyo, 2022), Latvia (Bricker, 2017), the Netherlands (Vols et al., 2015; Vols and Jacobs, 2017; van Dijck, 2018; Bruijn et al., 2018; Bruijn, 2021), Poland (Bricker, 2017), Portugal (Rodrigues and Campina, 2021), Slovenia (Bricker, 2017), Spain (Garoupa et al., 2012), and Sweden (Derlén and Lindholm, 2017c).

In addition, a growing body of research exists on the quantitative analysis of case law from international courts. For example,

Behn and Langford (2017) manually collected and coded roughly 800 cases on Investment Treaty Arbitration. Others have applied quantitative methods in the analysis of case law from the International Criminal Court (Holá et al., 2012; Tarissan and Nollez-Goldbach, 2014, 2015), the Court of Justice of the European Union (Lindholm and Derlén, 2012; Derlén and Lindholm, 2014; Tarissan and Nollez-Goldbach, 2016; Derlén and Lindholm, 2017a,b; Frankenreiter, 2017a,b; Zhang et al., 2018; ter Haar, 2020), and the European Court of Human Rights (Bruinsma and De Blois, 1997; Bruinsma, 2007; White and Boussiakou, 2009; Christensen et al., 2016; Olsen and Küçüksu, 2017; Madsen, 2017).

2.3 Information Extraction

In most research projects, case law is manually collected and hand-coded. Nevertheless, a number of researchers are now using computerised techniques to collect case law and automatically generate usable information from it (see Trompper and Winkels, 2016; Livermore et al., 2017; Shulayeva et al., 2017; Law, 2017; Lippi et al., 2019; Kufakwababa, 2021; Ruggeri et al., 2022). For example, Dyevre (2015) discusses the use of automated content analysis techniques in the legal discipline, using tools such as Wordscores[4] and Wordfish,[5] which have traditionally been used to automatically extract political positions, by using word frequencies in text documents. The author applied the two techniques to the analysis of a (relatively small) dataset of 16 judgements on European integration by the German Federal Constitutional Court. He found that both Wordscore and Wordfish are able to generate judicial position estimates that are remarkably reliable, when compared with the accounts appearing in legal schol-

[4] `http://www.tcd.ie/Political_Science/wordscores/`, accessed on 04/04/2022

[5] `http://www.wordfish.org/`, accessed on 04/04/2022

arship. Christensen et al. (2016) used a quantitative network analysis to automatically identify the content of cases from the ECtHR. They exploited the network structure induced by the citations, in order to automatically infer the content of a court judgement. Panagis et al. (2016) used topic modelling techniques to automatically find latent topics in a set of judgements by the Court of Justice of the European Union (CJEU) and the ECtHR. Derlén and Lindholm (2017a) used computer scripts to extract information concerning citations in CJEU case law.

2.4 Statistical Analysis

A large number of studies (especially outside the USA) present basic descriptive statistics for manually collected and coded case law (e.g., Bruinsma and De Blois, 1997; White and Boussiakou, 2009; De Jaeger, 2017; Madsen, 2017; Vols and Jacobs, 2017). Other studies present the results of relatively basic statistical tests, such as correlation analysis (e.g., Doron et al., 2015; Evans et al., 2017; Bruijn et al., 2018). A growing body of papers present the results of more sophisticated statistical analyses, including case law regression analysis (see Dhami and Belton, 2016). Most of these papers focus on case law from the USA (see Chien, 2011; Epstein et al., 2013), but researchers outside the USA have also conducted such analyses (Holá et al., 2012; Behn and Langford, 2017; Bricker, 2017; van Dijck, 2018; Zhang et al., 2018; Frankenreiter, 2017a, 2016; Vols, 2019).

2.5 Citation Analysis

A growing body of research presents the results of citation analysis of case law from US courts (see Whalen, 2016; Matthews, 2017; Shulayeva et al., 2017; Frankenreiter, 2016), in which patterns of citations within case law documents, as well as their number and

impact, are analysed. Other scholars have applied this method to case law from European countries, such as Sweden (Derlén and Lindholm, 2017c), Germany (Arnold et al., 2021), Poland (Górski, 2021) and France (Vazirgiannis et al., 2020). Researchers have also used this method to analyse case law from international courts. Soh (2019) analyse citations of ingapore Court of Appeal. Some have performed a citation analysis of the case law from the CJEU (Lindholm and Derlén, 2012; Derlén and Lindholm, 2014; Tarissan and Nollez-Goldbach, 2016; Derlén and Lindholm, 2017a,b; Frankenreiter, 2017a,b, 2016; Renberg and Tolley, 2021). Derlén and Lindholm (2017a, p. 260) use this method to compare the precedential and persuasive power of key CJEU decisions, using different centrality measurements. A number of studies investigated citation network analyses of case law from the ECtHR (Lupu and Voeten, 2012; Christensen et al., 2016; Olsen and Küçüksu, 2017; Olsen and Esmark, 2020; Renberg and Tolley, 2021). Olsen and Küçüksu (2017, p. 19) posit that citation network analysis enables researchers to note the emergence and establishment of patterns in case law more easily; patterns which might otherwise have been difficult to identify. Some researchers have combined citation network analysis of case law from both European courts into one study (Šadl and Olsen, 2017). Other papers used this method to analyse case law from the International Criminal Court (Tarissan and Nollez-Goldbach, 2014, 2015, 2016).

2.6 Argument Mining

There are studies which focus specifically on extracting the arguments of legal cases (among others, Mochales and Moens, 2008; Wyner et al., 2010; Xu et al., 2020). Being able to identify arguments is essential for the automatic analysis of legal data, and the method can also be used to predict court decisions. However, argument mining is a very hard task, and the majority of known approaches to solving

it require a large amount of manually annotated data.

2.7 Machine Learning

A relatively small number of studies have used machine learning techniques to analyse case law (see Evans et al., 2007; Ashley and Brüninghaus, 2009; Ashley, 2017; Custers and Leeuw, 2017). However, since 2015 the number of papers focussing on predicting court decisions using machine learning has been increasing rapidly, worldwide. We discuss the papers that (claim to) predict court decisions in Chapter 4. First, in the next chapter, we will introduce the necessary terminology.

CHAPTER 3

Machine Learning Techniques for Legal Text Classification

Given the interdisciplinary nature and novelty of the field of predicting court decisions using machine learning, we do not expect the reader to know all the relevant terminology and methodology. In this chapter, we therefore attempt to explain (in simple terms) some terminology and algorithms that are inherent to text classification using machine learning. We limit the described methods to those used in academic works claiming to predict court decisions. Specifically, we will introduce classification, supervised machine learning, features, n-grams, word embeddings, decision trees, random forests, bootstrap aggregating (bagging), k-nearest neighbours, Naive Bayes, logistic regression, support vector machines, neural networks, the multilayer perceptron, hierarchical attention networks, transformers, BERT, convolutional neural networks, cross-validation, and, finally, evaluation metrics, such as accuracy, precision, recall, and f1-score. We will provide a high-level introduction to these concepts, as they appear in our work and in other papers in the field. In the following chapters, we also discuss studies employing these techniques, and report on our studies of machine learning for legal text classification.

Legal information of any sort is largely written in natural, although rather specific, language. For the most part, this information is relatively unstructured. Consequently, to process legal big data automatically one needs to use techniques developed mainly in the field of natural language processing (NLP). NLP allows for many possible ways to process case law, and even though many steps have been taken towards the systematisation of data and the automatisation of processes, the number of choices one can make remains daunting.

In order to predict court decisions, one may employ statistical methods or machine learning (which also involves statistics). Academic research focused on predicting court decisions has predominantly used machine learning. Machine learning is an omnibus term for a computer program or model that uses historical data to make predictions for new (i.e. unseen) data. In more technical terms, machine learning is the process of approximating a function that *maps* the model's input (e.g., the text of legal documents, converted into a computer-suitable numerical representation) to the labels (e.g., a violation or no violation of human rights, or the amount of a fine). There are different types of machine learning, but the most common approach in the field of predicting court decisions is supervised machine learning, which is what we use in this dissertation and will therefore discuss in more detail.

3.1 Supervised Machine Learning

To illustrate how supervised machine learning works, we will consider a non-textual example. Suppose we want to create a program that recognises pictures of cats and dogs. For that we first need a database of images of cats and dogs, where each image has a label: either *cat* or *dog*. Then we show the program the pictures with their respective label, one by one (i.e. the supervision part). If we show enough pictures, eventually the program starts recognising various

characteristics for each animal (e.g., cats have long tails, dogs are generally more furry). This process is called *training* or *fitting the model*. Once the program has *learned* this information, we can show it a picture without a label and ask it to guess which *class* the picture belongs to. Of course, a computer does not know what a tail or fur are, but it can use the pixels of the images to recognise the patterns belonging to each class.

Very similar experiments can be conducted with text. For instance, when categorising texts into those written by men and those written by women, the program can analyse both the text itself, and the style in which it was written. Research conducted on social media data shows that, when training such models, we can observe that men and women generally talk about different things, and in different ways. For example, women use more pronouns than men (Rangel and Rosso, 2013), while men swear more often (Schwartz et al., 2013).

Within supervised machine learning there are two types of predictive models: classification models, and regression models. Classification is the task of predicting a discrete class label (e.g., whether or not a case will result in an eviction) for a specific input, whereas regression is the task of predicting a continuous quantity (e.g., the length of a prison sentence, in months).

The most commonly adopted approach to predicting court decisions has been to treat it as a classification task. However, regression has started to be used in the field more often, particularly for charge prediction, and primarily concentrating on the decisions of Chinese courts (e.g., Cheng et al., 2020; Tan et al., 2020; Huang et al., 2020). In this dissertation we will focus on the classification task, due to the diversity of work written on this topic. Consequently, when we classify court decisions it should be understood as predicting a specific, discrete class (e.g., a violation or no violation of human rights). Nevertheless, the issues and advantages of the different methods discussed throughout this dissertation are, to a large extent, applicable to re-

gression models as well.

In a legal judgement classification task, the model is provided with (textual) information from many court cases or other legal documents, together with the actual outcomes. By being provided numerous examples (in the training phase), the computer is able to identify patterns which are associated with each class of verdict (i.e. violation versus no violation). To evaluate the performance of the machine learning program in the so-called *testing* phase, the model is provided with a new case (without a verdict) that it has not encountered before, for which it has to estimate the most likely outcome. To determine the outcome (or *classification*), the program uses the information which was deemed important during the training phase. The predicted outcome is then compared to the actual outcome to evaluate the model's performance.

3.2 Feature Vector Representation

Features are the input which a machine learning model uses to determine its classification. For example, when classifying pictures of cats and dogs, the position and colour of pixels are features. In the case of legal judgement classification, features are extracted from the text and may be (for example) individual words or sequences of words. In order to make these interpretable for a machine learning algorithm, each feature is converted into a series of numbers (i.e. a vector). The features are most commonly extracted automatically (using a large piece of text), but can also be chosen manually if one wants to predict court decisions based only on specific variables, such as judges, gender, country of origin, etc.

There are many approaches to creating vectors out of textual data. One such approach is *one-hot encoding*. For example, consider the following sentence:

By a decision of 4 March 2003 the Chamber declared the application admissible.

With *one-hot encoding* (see Figure 3.1), we (automatically) create vectors that are as long as our *vocabulary*, which is the collection of unique words in our data. Each number in the vector represents whether or not a certain word in our vocabulary is present (i.e. a binary value). Let us assume the sentence above is all we have. Since we have 13 unique words (note that punctuation is also considered a word in our example), each word is represented by a vector of 13 numbers, where '1' is in a different position each time.

By	-	[1,0,0,0,0,0,0,0,0,0,0,0,0]
a	-	[0,1,0,0,0,0,0,0,0,0,0,0,0]
decision	-	[0,0,1,0,0,0,0,0,0,0,0,0,0]
of	-	[0,0,0,1,0,0,0,0,0,0,0,0,0]
4	-	[0,0,0,0,1,0,0,0,0,0,0,0,0]
March	-	[0,0,0,0,0,1,0,0,0,0,0,0,0]
2003	-	[0,0,0,0,0,0,1,0,0,0,0,0,0]
the	-	[0,0,0,0,0,0,0,1,0,0,0,0,0]
Chamber	-	[0,0,0,0,0,0,0,0,1,0,0,0,0]
declared	-	[0,0,0,0,0,0,0,0,0,1,0,0,0]
the	-	[0,0,0,0,0,0,0,0,1,0,0,0,0]
application	-	[0,0,0,0,0,0,0,0,0,0,1,0,0]
admissible	-	[0,0,0,0,0,0,0,0,0,0,0,1,0]
.	-	[0,0,0,0,0,0,0,0,0,0,0,0,1]

Figure 3.1 | Example of one-hot encoding of the sentence 'By a decision of 4 March 2003 the Chamber declared the application admissible.'

If we want to encode more than just single words, in order to provide the entire text at once, we can do so by encoding all words in the same vector. For instance, the phrase 'application declared admissible' will be represented as the following vector: [0,0,0,0,0,0,0,0,0,1,1,1,0], where the word 'application' is marked in

the 11th position, 'declared' in the 10th, and 'admissible' in the 12th. If we take a much larger vocabulary (commonly, one would use all the available training data to do this), we can encode entire documents using this method. Often, in addition to encoding separate words, we want to take longer (more informative) phrases into account. A sequence of words (or characters) is called an n-gram. Single words are called unigrams, sequences of two words are dubbed bigrams, and sequences of three consecutive words are called trigrams.

If we split the above sentence into bigrams (i.e. two consecutive words) the extracted features consist of:

> *By a, a decision, decision of, of 4, 4 March, March 2003, 2003 the, the Chamber, Chamber declared, declared the, the application, application admissible, admissible .*

For trigrams, the features consist of:

> *By a decision, a decision of, decision of 4, of 4 March, 4 March 2003, March 2003 the, 2003 the Chamber, the Chamber declared, Chamber declared the, declared the application, the application admissible, application admissible .*

The above features can be automatically extracted from the text. However one-hot encoding of n-grams is a rather crude approach, as it only distinguishes between the presence and absence of a feature and does not take into account any other potentially useful information, such as the frequency with which an n-gram occurs in a document. Using the frequency as a feature value is certainly an improvement (e.g., 'By a': 100, '4 March': 1, 'never in': 0). However, some n-grams or words (such as 'the') are simply more common and are therefore used more frequently than others. Less frequent n-grams or words, such as 'torture', may be more informative for classification than common, high-frequency words. In order to take this into

account, the general approach is to normalise the absolute n-gram frequency, by taking into account the number of documents (i.e. cases) in which each n-gram occurs. The underlying idea is that the n-grams characteristic of a certain case will only occur in a few cases, whereas common, uncharacteristic n-grams will occur in many cases. This normalised measure is called *term frequency-inverse document frequency* (or *tf-idf*), and it is defined in the following formula:

$$\text{tf-idf}(d, t) = \text{tf}(t) * \text{idf}(d, t),$$

where $\text{idf}(d, t) = \log(n/\text{df}(d, t)) + 1$, with n being the total number of documents, and $\text{df}(d, t)$ being the document frequency.[1] The document frequency is defined as the number of documents d that contain term t. In our case, the terms are n-grams. For example, when a document of a thousand words contains the unigram 'torture' three times, the term frequency (i.e. tf) of 'torture' is (3 / 1000) = 0.003. Additionally, if the unigram 'torture' occurs in 10 documents out of a total of 10,000, the inverse document frequency (i.e. idf) equals $\log(10000/10) + 1 = 4$. The resulting tf-idf score is $0.003 \times 4 = 0.012$, and this is the score (i.e. the weight) assigned to the n-gram *torture*. It should be noted that this score is higher than simply using the term frequency score, as the tf-idf score reflects that the n-gram does not occur often in other documents. Instead of 0s and 1s, the vectors will consist of values between 0 and 1. A value of 0 indicates that the the n-gram is absent from the document, whereas a value higher than 0 indicates that the n-gram is present, and the number itself represents the tf-idf score.

However, the problem with the above methods is that each word, or n-gram, is seen as a completely unrelated item. In the above approach, for instance, words such as 'outcome' and 'verdict' are as different from each other as the words 'outcome' and 'torture'. This is

[1] This formula follows the one used in the `scikit-learn` Python package (Pedregosa et al., 2011). However, different variations and implementations can be used.

of course problematic, as the first two are clearly much more similar in meaning. To incorporate this similarity in feature vectors, *word embeddings* (see Mikolov et al., 2013) have been developed.

Word embeddings are also vectors, although the values or *weights* have been learned using supervised machine learning. Word embeddings aim to capture the semantic, syntactic, and contextual meaning of each word in the vocabulary, based on the context (i.e. the words surrounding any particular word in the text) in which it appears. Words that appear in a similar context will be represented by similar vectors. As 'outcome' and 'verdict' are more likely to appear in a similar context than 'outcome' and 'torture', their feature vectors will also be similar. Such an approach allows much more meaningful information to be captured. However, as the individual (feature) weights in a word embedding are not interpretable, this method may not be appropriate if the goal is to obtain a fully explainable system (as is often the case in the legal domain).

While one can train one's own word embeddings, some of the best-performing embeddings have already been trained using enormous amounts of data (i.e. *pre-trained*), and are available for others to use as off-the-shelf vector representations. These pre-trained models have often been created by very large commercial companies (such as Google or Facebook), which have many computational resources available to them. Some of the most commonly used pre-trained word embeddings are *Word2Vec*, *Glove*, *ELMo*, *BERT*, and *GPT*. Some of these models are available in multiple languages.

3.3 Non-neural Machine Learning

Traditional (or classic) machine learning is generally used to denote all machine learning approaches that are not based on so-called neural networks (i.e. neural machine learning). To make the distinction clear, in this dissertation we refer to such algorithms as 'non-neural

machine learning'. We will discuss seven algorithms in this section, and then proceed to discuss a number of neural methods in the subsequent section.

3.3.1 Decision Trees, Bootstrap Aggregating (Bagging), and Random Forests

A *decision tree* is a tree where each node represents a feature, each branch represents a decision rule, and each leaf represents the predicted outcome. Consider an eviction case as an illustration. Figure 3.2 shows a very simple example of a decision tree that predicts whether a case regarding eviction will result in an eviction or not. This decision tree shows that, if the tenant does not show up in court, or if the tenant does show up in court and the reason for their eviction is drugs or prostitution, they are evicted. If the tenant does show up, and the reason the landlord wants to evict the tenant is different, the tenant is not evicted. While this is a toy example, the machine learning algorithm will normally determine the nodes and branches of the tree during the training phase. The goal of the algorithm is to decide on the best nodes and branches, in order to yield the correct outcome as often as possible. To do so, it inspects features or variables one by one (in random order) and compares them against the label. For instance, it uses 'Did the tenant show up in court?' as the first variable, and calculates how many times the defendant was evicted after showing up in court, and how many times they were not. It does the same for the variables 'Are drugs the reason for eviction?', and 'Is prostitution the reason for eviction?' It then chooses the variable that is the best predictor of the correct label, and that node is chosen as the top (i.e. the *root* of the tree). Let us assume our toy example contains 100 judgements, with 50 cases that resulted in eviction and 50 cases that did not. The variable 'Did tenant show up in court?' separates the data in the following way: out of 30 tenants that did not show up in court, all were evicted. However, out of 70 cases where the tenant

did appear in front of the judge, 40 resulted in eviction. Subsequently, the algorithm investigates all the variables for the remaining 70 cases in the new node, compares them against eviction and non-eviction labels, and selects the variable that separates the cases between labels in the best way. This process is then repeated for every node, for as long as using the variable to split the data results in an improvement.

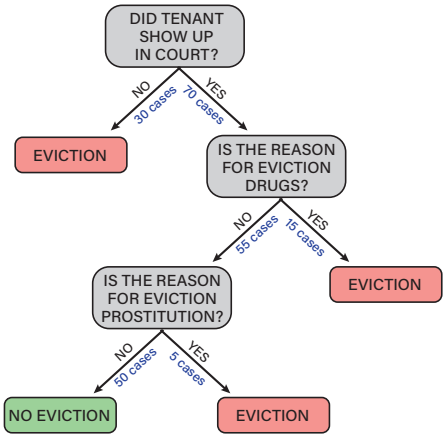


Figure 3.2|A toy example of a decision tree that predicts whether a case results in an eviction

There are also extensions to this system that use a more sophisticated way of determining the nodes and branches. For instance, bootstrap aggregating (commonly referred to as *bagging*) is an algorithm that fits multiple models at once, by randomly sampling the training set over and over again, so the same judgements may be sampled multiple times for the same tree, or for multiple trees. The model then combines all results to make the final prediction. A further extension of bagging is a *random forest*, which, in addition to using different samples of the data, randomly selects the features that will be used for prediction. It creates many separate random decision trees

(to form a forest), and determines the final label on the basis of the most common label assigned by its trees. A further extension of the random forest is an *extremely randomised trees classifier* (Extra Tree). Like the random forest, it creates a large number of trees based on the training data. However, it does not sample the same data points twice. Additionally, instead of trying to find the best split for the nodes in order to fit the data in the best possible way, the model splits the nodes randomly. Such randomisation allows for better generalisation, and it sometimes results in better predictions on unseen data than those resulting from decision trees and random forests.

To illustrate how a random forest (and a decision tree) works, consider the following example. A lawyer tries to estimate whether the firm can win a case. However, it is a very complex case that involves multiple areas of expertise. For this reason, she asks her colleagues for advice. The colleague asks her a range of questions about the outcomes (i.e. a selection of features and associated labels) of (similar) cases she has dealt with previously. Based on the lawyer's responses and the features of the current case, her colleague estimates the potential outcome. This is representative of how a decision tree works. As the lawyer is still nervous about the case, she asks more of her colleagues for advice, they all ask her different questions according to their expertise, and they all provide a prediction on the basis of her answers. In the end, the lawyer decides to rely on the prediction she received most often. This procedure is representative of how a random forest works.

Other variants of the decision tree algorithm include *AdaBoost* (adaptive boosting), which creates small decision trees, and *gradient boosting*, which uses different mechanisms to build small trees and prioritise some over the others, taking the mistakes of previously created trees into account.

3.3.2 K-nearest Neighbour

The *k-nearest neighbour* (k-NN) algorithm proceeds from the assumption that similar data points are near to each other. For example, we would like to predict whether or not a case will result in an eviction, given the submissions by the parties. The k-NN algorithm assumes that the information (i.e. words, phrases, and other specific variables) about cases that end up in eviction is somewhat similar in all of them, and that the cases which end in no eviction are also assumed to be similar to each other, but different from eviction cases. To classify a new case, the algorithm determines the *k* (i.e. a number) cases most similar to the example and checks what the outcome was for the majority of these. The value of *k* is determined during training, and is based on which value produces the best results. The choice of *k* may affect the results, as is shown in Figure 3.3. In the example, when *k* is set to 3, the predicted outcome will be eviction (following the majority), whereas when *k* is set to 7, the predicted outcome will be no eviction.

3.3.3 Naive Bayes

Naive Bayes (NB) is a frequently-used classification technique in NLP. The NB classifier is based on Bayes' rule:

$$P(O|W) = \frac{P(W|O) * P(O)}{P(W)}$$

This rule states that the posterior probability of an outcome O given a word W, is equal to the likelihood of a word being present in the text of a certain outcome, multiplied by the (prior) probability of that outcome. This value then is divided by the probability of the word. However, this latter probability is usually held constant, as the goal is to determine the optimal outcome given a predetermined set of words.

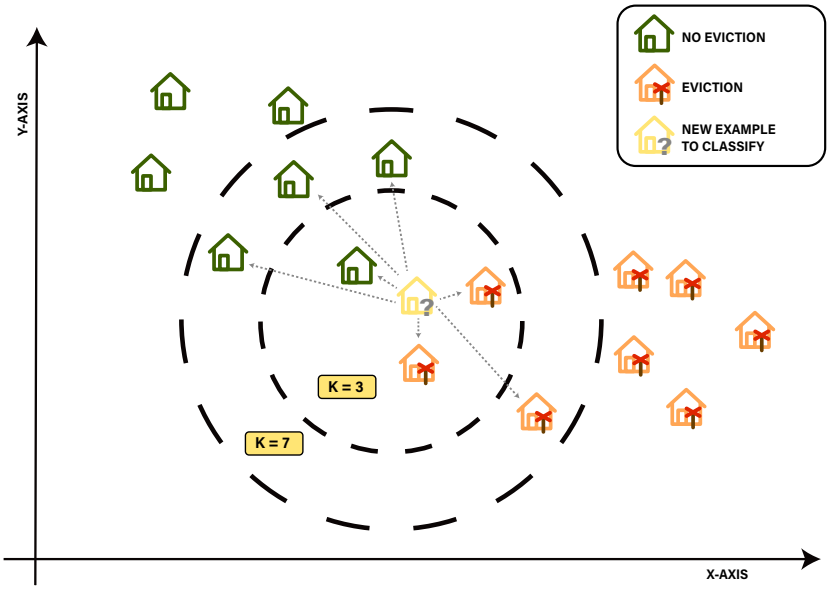


Figure 3.3 | A toy example of a k-nearest neighbour algorithm for classifying case outcomes (eviction or no eviction)

The NB classifier calculates the probability of each feature using Bayes rule, but under the (naive) assumption that all features are independent. This means that the NB algorithm does not take the word order into account. To illustrate how the algorithm works, we will try to forecast whether a particular Dutch case results in eviction, using short summaries of submissions by the parties. Our example contains 12 cases, 8 of which are eviction cases, and 4 of which are non-eviction cases. To make a prediction using Naive Bayes, we first count how many times each word appears in the text of eviction cases ($P(W)$). Using this information, we can calculate the probability of seeing each word, given that it has appeared in cases with an eviction

outcome ($P(W|O)$). For instance the word 'drugs' appeared in the set of eviction cases 18 times, and the total number of words in all the eviction cases is 100, which gives us a probability of $P(W|O) = 0.18$ that a word in that set of eviction cases equals 'drugs'. Likewise, we can calculate the probability for all words in eviction cases. For instance, in our toy dataset the word 'prostitution' has a 0.09 probability of being a word in an eviction case, while the word 'child' has a much lower probability of $P(W|O) = 0.02$. We can use the same method to calculate the probability of observing each word, given that the words have appeared in non-eviction cases. It should be noted that non-eviction cases do not necessarily have to have the same number of words. Assuming that non-eviction cases only have a total of 50 words and the word 'drugs' appears three times, the probability of encountering the word 'drugs' in a non-eviction case is $P(W|O) = 0.06$. Similarly, the word 'prostitution' may have a probability of $P(W|O) = 0.1$, whereas the word 'child' may have a probability of 0.2. These probabilities are called *likelihoods* (i.e. the probability of observing a word, given an outcome).

If we want to use Naive Bayes to determine the outcome for a new case that (for simplicity) only contains two words, 'drugs' and 'child', we calculate the probability for both cases (eviction versus non-eviction). For an eviction case, the prior probability is $P(O) = 8/12 = 0.67$, as 8 out of 12 cases are eviction cases. This prior probability is then multiplied by the probability of the word 'drugs' appearing in the eviction case ($P(W|O) = 0.18$, as indicated above), and again multiplied by the probability of 'child' appearing in the eviction case ($P(W|O) = 0.02$, as indicated above). This results in $P(O|W) = 0.67 \times 0.18 \times 0.02 = 0.002$. We compare this value to that calculated under the prior assumption of the case being a non-eviction case. In this situation, the prior probability is $P(O) = 4/12 = 0.33$. This value is multiplied by the likelihood of the word 'drugs' occurring in a non-eviction case, and by the likelihood of the word 'child' occurring in

non-eviction cases. This yields $P(O|W) = 0.33 \times 0.1 \times 0.2 = 0.007$. As this value is higher than that associated with assuming it would be an eviction case, the prediction of the Naive Bayes system will be that the outcome of the case is non-eviction.

The issue with this approach is that some words may be absent from previous cases. For these new words we end up multiplying the prior probability and likelihoods of other words by zero. This will always result in zero, which is problematic. To combat this issue, the typical solution is to pretend that each word occurs at least once in every class (i.e. by increasing all frequencies with one).

3.3.4 Logistic Regression

Another popular approach for classification tasks is *logistic regression* (LR). The word regression may be confusing here, since it is, in fact, a classification method. Instead of calculating the probability of a certain class given a particular feature, as in Naive Bayes, logistic regression learns the probability of a sample belonging to a certain class. It then tries to find the optimal decision boundary to best separate the classes, using a sigmoid (i.e. s-shaped) function.

To illustrate logistic regression, we will look at another example. Here, we are trying to predict whether the defendant in a theft case will not be granted bail, given the amount of money they stole. Figure 3.4 provides an illustration for this example. After placing cases where the defendant did not make bail, according to how much was stolen at 100% probability (we know they ended up not making bail, since we use data where the court has already made the decision), and cases that did make bail at 0% probability, the algorithm tries to find the best way to place the s-shaped decision boundary between them, in order to make the best separation possible. After this process, also called *fitting*, we can test the model by taking new examples to classify. In Figure 3.4, we can see two new examples (in yellow) close to each other. The left-most example was assigned a 25% proba-

bility of not making bail (and therefore, likely to make bail), whereas the right-most example was assigned a 75% probability of not making bail.

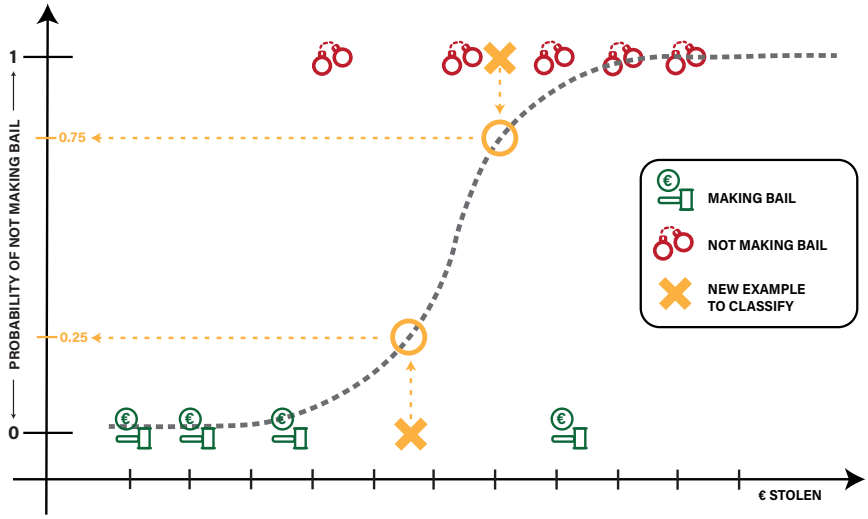


Figure 3.4 | An example of logistic regression

In the example in the figure we have only used one feature, so that it could be visualised in a two-dimensional graph. It is possible to do a classification using more variables. Each variable introduces one additional dimension to the data. For instance, we can introduce an additional variable of whether or not the defendant has previously been convicted of stealing, and how many times. Two predictor variables could be visualised with a 3D graph and, instead of the line (as in Figure 3.4), the data points would be separated by a curved plane. However, having more variables does not allow for a clear visualisation anymore. Nevertheless, the principle remains the same: the model separates the data points in the multi-dimensional space, using a curved hyperplane. Logistic regression also allows us to evaluate which predictor variables are best for predicting a particular

outcome, by inspecting the associated coefficients for each variable.

3.3.5 Support Vector Machines

One of the non-neural machine learning algorithms most commonly used in NLP is the *support vector machine* (SVM). Even though neural machine learning systems are more powerful, in some NLP tasks, such as dialect identification and identifying an author's gender (Basile et al., 2018; Medvedeva et al., 2017), SVMs still perform very well, while also having the advantage of requiring less computational resources than neural machine learning algorithms.

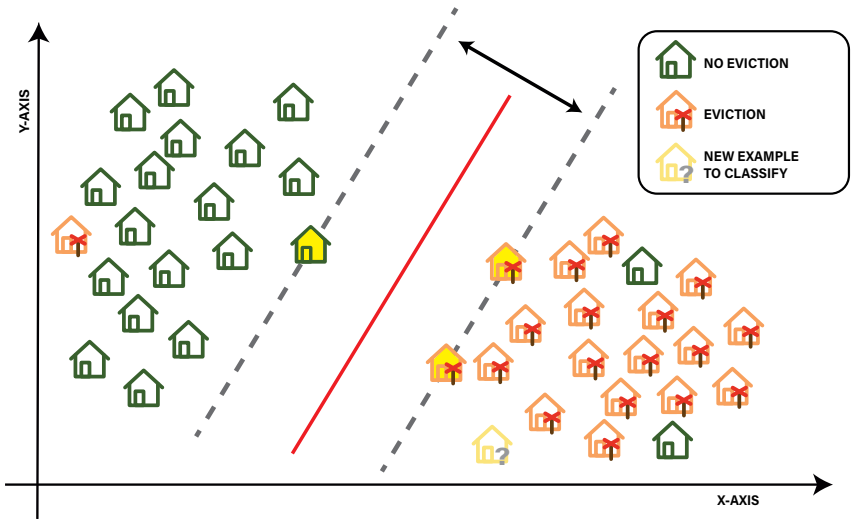


Figure 3.5 | Illustration of an SVM dividing data into classes

Figure 3.5 shows an example of a (linear) support vector machine classifier. The SVM tries to split the data points, based on their labels in the dataset (i.e. the *training data*). Specifically, it will determine the simplest (linear) equation that separates differently-labelled data

points from each other with the least amount of error.

As we can see in Figure 3.5, the algorithm decides on the best hyperplane (i.e. a line in multiple dimensions) to separate the data. In the figure this is the middle line, separating the eviction and no eviction cases. The support vectors are the data points nearest to this line. The goal of the SVM algorithm is to choose the position of the hyperplane in such a way that the largest possible margin with respect to the support vectors is achieved. This allows for a greater chance to classify new (i.e. unseen) data correctly. While non-linear SVMs are possible,[2] linear SVMs are most commonly used in text classification.

3.4 Neural Networks

For many text classification tasks, current state-of-the-art systems consist of so-called neural networks (NN). The reason they have been given this name is that their architecture was originally *inspired* by the network of human neurons. A neural network consists of nodes, which together form layers, and weights (connecting the nodes), which are learned during training. Figure 3.6 shows an example of a relatively simple neural network, called a *multilayer perceptron*.

The example network shown in Figure 3.6 contains an input layer, an output layer, and a hidden layer. However, neural networks can have many more hidden layers; the more layers, the *deeper* the network. At present, very deep networks have shown to be successful in a range of NLP tasks (Deng and Liu, 2018). Consequently, neural networks with more than three layers are often referred to as *deep learning*. Figure 3.6 visualises a neural network for the task of classifying whether or not a case is judged to show a violation of human rights. The feature vectors from the training dataset are fed into the model. They are then adjusted in each layer of the model, using weights (i.e. values by which the input from the previous layer is mul-

[2] See `https://scikit-learn.org/stable/modules/svm.html` for a description.

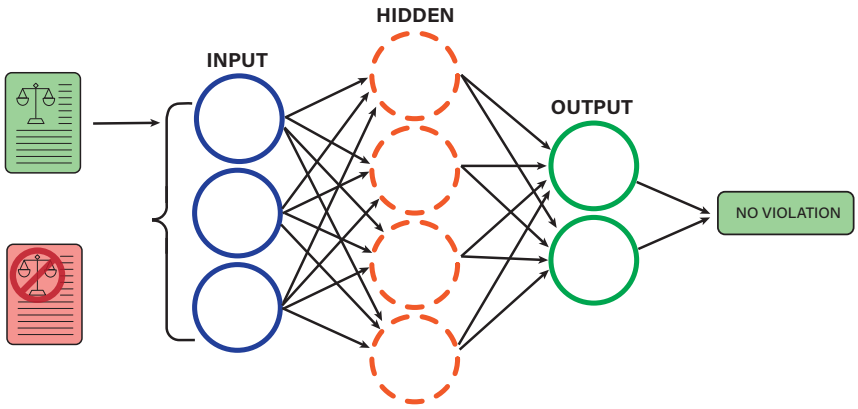


Figure 3.6 | An example of multilayer perceptron neural network

tiplied) and *activation functions*, which are mathematical functions that convert the nodes input into a new value. The activation function may differ, depending on the specific architecture of the model. Moreover, the size of the hidden layer does not have to be the same as that of the input layer, and the amount of nodes in the hidden layer is chosen before training. Each value from the input layer is adjusted according to the weights, which contributes to each node in the hidden layer. This is shown schematically by the arrows in Figure 3.6. Simply put, the model receives the vector representation as an input, then multiplies each value in it by (usually) randomly initialised weights. The new values are then given as an input to each node of the next (hidden) layer, where they are converted into a new numerical value by the activation function. The new values are once again multiplied by randomly initialised weights and passed onto the activation function in the next layer, and so on. If the (random) weights result in an incorrect prediction, the model goes back through the layers (in the

opposite direction to the arrows), adjusting the weights according to the degree of prediction error in the previous iteration. After that, the process repeats a number of times (i.e. *epochs*). The process of adjusting the weights is called *back propagation*, and it is an essential part of neural networks.

3.4.1 Convolutional Neural Networks

A convolutional neural network (CNN) is a specific variant of a multilayer perceptron, but with many different hidden layers (i.e. deep learning). CNNs were developed and traditionally used for image (pixel) analysis, but have also been used successfully in several text classification tasks (e.g., sentiment analysis, in which a sentence is classified as being positive, negative or neutral; Sun and Gu, 2017). In the same way that vectors representing pixels are the input for a model in image analysis, the algorithm takes vector representation of text as input, and uses it to determine patterns in text classification.

3.4.2 Hierarchical Attention Networks

In addition to making the networks deeper or wider (i.e. with more nodes in the layers), there are many more complex techniques to improve the performance of a neural network. One such technique, which can be included in a variety of neural network models, is *attention*. Attention allows the neural network to focus on, or *pay attention* to, more important parts of the input sequence by assigning higher weights, and thereby suppressing other (less important) parts. A hierarchical attention network uses its own neural network to determine the weights that correspond to each word vector. It then calculates the weighted sum of every vector, to create a vector representation of a whole sentence. This procedure is then repeated at sentence level and document level, thereby creating *hierarchical attention*.

3.4.3 Transformers

Another more recent architecture that uses attention is a transformer. For the last few years these new deep learning models have dominated NLP (Wolf et al., 2020). Transformers consist of a sequence-to-sequence architecture combined with attention. This sequence-to-sequence approach is fundamental for tasks such as machine translation, since both the input and output must be a sequence (of words). An example of a popular legal domain task that often uses a sequence-to-sequence approach is legal text summarisation.

One of the most well-known transformer models is BERT (Devlin et al., 2019), which was also mentioned in Section 3.2. BERT's popularity has risen due to its unprecedented performance in many NLP tasks (Otter et al., 2020). The system is essentially a very large language model, pre-trained on an enormous corpus of texts from the web. A language model is a probability distribution over words in a text. It calculates these probabilities by using two strategies at the same time. The first strategy is masking words (i.e. replacing random words with a placeholder, such as [MASK]), trying to predict the missing word based on other words in the sentence, and then calculating its probability. The second strategy uses two sentences, and tries to predict whether the second one follows the first in the text. The advantage of this model is that it can be fine-tuned (i.e. adapted) for a specific task, such as classifying court decisions, by adding an additional layer on top. Consequently, other than being able to produce vector representations for words and sentences, it is also able to make predictions. Due to BERT's high performance, many variations on this model have been developed. These are trained similarly, but on different data, and often in different languages. Two models commonly used to classify court decisions are RoBERTa (Liu et al., 2019) and XL-Net (Yang et al., 2019), which are both much larger and trained on up to 10 times more data than BERT. Another variant is DistillBERT (Sanh et al., 2019), which is a much lighter version of BERT, although with

comparable performance. Yet another variant is H-BERT (Chalkidis et al., 2019), which uses a hierarchical structure similar to hierarchical attention networks. Finally, FlauBERT (Le et al., 2019), which was trained on the French language, has also been used.

3.5 Evaluation

As mentioned before, supervised machine learning involves both training and testing phases. During the training phase, the model parameters are learned on the basis of providing the model with input data and the associated labels. During the testing phase, the model assigns the labels (e.g., cats and dogs, or eviction and non-eviction) to a separate held-out (i.e. unseen) test set, and performance is evaluated by comparing the assigned labels to the correct labels.

Another approach for evaluating the model's performance is to use *k-fold cross-validation*. For this, we take all the data available for training the model, and split the set into *k* parts. Then, we take one part out and train the model using the remaining $k-1$ parts. Once the model is trained, we evaluate it using the held-out part, by comparing the assigned labels to the actual labels. We then repeat this procedure, except that we take a different part out (i.e. train on the rest, evaluate on the held-out part). We repeat this procedure *k* times, until we have evaluated the model using each of the *k* withheld parts. For instance, by setting *k = 5* we will perform five-fold cross-validation, and train and test the model five times (see Figure 3.7). Each time, the part used for testing consists of 20% (1/5th) of the data, whereas the training uses the remaining 80% of the data. The average cross-validation performance is then determined by averaging the testing performance across all folds.

Cross-validation allows one to simultaneously determine the optimal parameters of the machine learning system and evaluate how well the system performs while it is being evaluated, using different

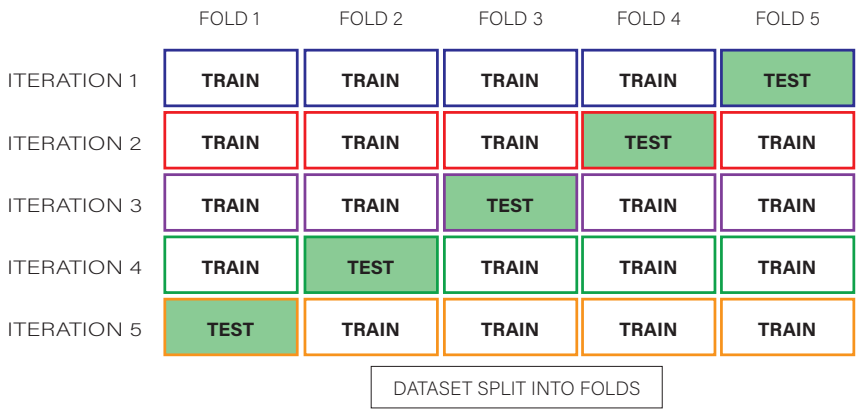


Figure 3.7 | Example of 5-fold cross-validation

samples of data. In this way, the model is more likely to perform better for unseen cases. Nevertheless, a held-out test set is often used for the final evaluation. If this is the case, the cross-validation stage is (for example) used to determine the optimal parameter choices or the best algorithm, after which the test set is used to determine the performance of the final model.

The conventional way to report the performance of a classification system is to use either accuracy or the so-called *f1-score*. Accuracy is the percentage of correctly identified labels. However, accuracy is not always the best way to evaluate performance, especially when a test set is not balanced regarding the class labels (i.e. the number of cases per class label differs). For example, consider a test set where 80% of cases have the label 'violation' (of human rights) and 20% have the label 'no violation'. If a model predicts everything as 'violation', its accuracy will be 80%, despite it not being useful at all. This issue is exacerbated when dealing with more than two classes,

as the measure does not indicate which classes were predicted better or worse than the others. Therefore, the f1-score is commonly used to evaluate performance. This is the harmonic mean (a type of average) of the two measures *precision* and *recall*. For a class, precision is the number of judgements for which the assigned outcome is correct. Recall is the percentage of cases with a specific outcome, which are classified correctly by the system.

To illustrate the terms above, consider the so-called confusion matrix in Table 3.1, which gives an example of prediction results for a supreme court, with labels being the supreme court either affirming or reversing the decision of the lower court.

		Predicted label	
		Affirmed	Reversed
Actual label	Affirmed	30 (TP)	10 (FN)
	Reversed	40 (FP)	20 (TN)

Table 3.1 | An example of a confusion matrix

In the table, the columns show predicted labels and the rows show actual labels. For instance, we can see that out of 40 cases that were affirmed, the model predicted 30 as affirmed and 10 as reversed, and out of 60 cases that were reversed it predicted 40 as affirmed and 20 as reversed. In order to explain the metrics better, we will assume one class as positive (e.g., affirmed), and the other as negative (e.g., reversed). Then, the judgements that were predicted as affirmed and were actually affirmed are *true positives* (TP) and those which were correctly predicted as reversed are *true negatives* (TN). Additionally, judgements that were predicted as affirmed, but were actually reversed are *false positives* (FP), and judgements that that were predicted as reversed, but are actually affirmed are *false negatives* (FN).

Accuracy is calculated by dividing all the correctly predicted outcomes (TP+TN) by all the cases (TP+TN+FP+FN), so $(30 + 20)/(30 +$

$20 + 40 + 10) = 0.5$, yielding an accuracy of 50%. According to the table however, the model most often predicts the affirmed label, although accuracy does not reveal this. Consequently, we can use the other measures to evaluate specific issues with the system. Precision shows how many judgements, out of all the cases predicted as affirmed, were correctly identified as affirmed. This equals the number of true positives divided by the sum of true positives and false positives, so $30/(30 + 40) = 0.43$. Recall is a metric which evaluates how many of the actual affirmed cases are identified as such. This equals the number of true positives, divided by the sum of true positives and false negatives, so $30/(30 + 10) = 0.75$. The f1-score is calculated according to the formula below:

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = 2 \times \frac{0.43 \times 0.75}{0.43 + 0.75} = 0.55$$

In addition, one may want to calculate and report scores for the reverse label. However, in our case, both the scores and the confusion matrix clearly show that the model does not perform well.

CHAPTER 4

Rethinking the Field of Automatic Prediction of Court Decisions

In this chapter, we discuss previous research in the field of automatic prediction of court decisions. We define the differences between outcome identification, outcome-based judgement categorisation, and outcome forecasting, and we review how various studies fall into these categories. We also discuss how important it is to understand the legal data one works with, in order to determine which task can be performed. Finally, we reflect on the needs of the legal discipline regarding the analysis of court judgements.

Chapter adapted from:

Medvedeva, M., Wieling, M., and Vols, M. (2022). Rethinking the field of automatic prediction of court decisions. *Artificial Intelligence and Law*, pages 1–18.

4.1 Introduction

Automatic analysis of legal documents is a useful, if not necessary, task in contemporary legal practice and research. Of course, data analysis should be conducted in a methodologically sound, transparent and thorough way. These requirements are extra important with regard to legal data. The stakes that legal professionals such as lawyers, judges and other legal decision-makers deal with, and the cost of error in this field, make it very important that automatic processing and analysis are done well. This means that it is essential to understand how the automated systems being used in the analysis work, as well as exactly which legal data are being analysed, and for what purpose.

The need for established practices and methodologies is becoming more urgent with the growing availability of data. Striving for transparency, many national and international courts in Europe adhere to the directive to promote accessibility and the reuse of public sector information,[1] by publishing their documents online (Marković and Gostojić, 2018). This is also the case for many other courts around the world.[2] Digital access to a large amount of published case law provides a unique opportunity to process this data automatically, and on a large scale, using NLP techniques.

In this chapter, we review previous work on applying NLP techniques to court decisions, and discuss the methodological issues, as well as good practice. While automatic legal analysis is an enormous field, which has been around for some time (see Chapters 2 and 3), in this chapter, we focus solely on the recent development of using machine learning techniques to classify court decisions. This sub-field has expanded drastically over the past six years, with papers attempt-

[1] `https://digital-strategy.ec.europa.eu/en/policies/legislation-open-data`, accessed on 11/10/2021

[2] See, for instance, case law from the Constitutional Court of South Africa, available at: `https://collections.concourt.org.za`, accessed on 04/04/2022

ing to predict the decisions of various courts around the world. We subsequently discuss whether it is fair to say that such attempts have indeed succeeded. Our main finding is that many of the papers under review, which claim to predict the decisions of courts using machine learning, actually perform one of three different tasks.

In the following section, we define the scope of the review we conducted. Next, in Section 4.3 we discuss (our terminology for) the three different types of tasks within the field of automatic analysis of court decisions, and how previous research falls within those categories. We examine the purpose of such research for each task, as well as good practice and potential pitfalls. In Section 4.4, we discuss our survey. In Section 4.5, we summarise and conclude our work.

4.2 Scope of the Review

We limit our review to the papers that use machine learning techniques and claim to be predicting court decisions. The publication dates range from 2015 to (June) 2021.[3] We specifically chose this time range, as this is when machine learning in this field became popular. If a paper included in our review attempts multiple tasks, we focus only on its experiment(s) regarding the prediction of judicial decisions. While our survey is meant to provide an exhaustive overview, we may have inadvertently missed some research in the field.

While we already mentioned that research in the field is growing, not all courts share (all of) their case law online. Furthermore, the majority of available case law is extremely varied in its outcomes, which may make it harder to set up an outcome prediction task. For this reason, the research often focuses on a relatively restricted set of courts. For this dissertation, we surveyed publica-

[3] For descriptions of earlier approaches to the automatic prediction of court decisions, with and without using machine learning, we refer to Ashley and Brüninghaus (2009), Chapter 4 of Ashley (2017) and Chapter 2 of this dissertation.

tions which use machine learning approaches and focus on the case law of: the US Supreme Court (Sharma et al., 2015; Katz et al., 2017; Kaufman et al., 2019); the French court of Cassation (Şulea et al., 2017; Sulea et al., 2017); the European Court of Human Rights (Aletras et al., 2016; Liu and Chen, 2017; Chalkidis et al., 2019; Kaur and Bozic, 2019; O'Sullivan and Beel, 2019; Visentin et al., 2019; Chalkidis et al., 2020; Condevaux, 2020; Medvedeva et al., 2020a,b; Quemy and Wrembel, 2020; Medvedeva et al., 2021b); Brazilian courts (Bertalan and Ruiz, 2020; Lage-Freitas et al., 2019); Indian courts (Bhilare et al., 2019; Shaikh et al., 2020; Malik et al., 2021); UK courts (Strickson and De La Iglesia, 2020); German courts (Waltl et al., 2017); the Quebec Rental Tribunal (Salaün et al., 2020) (Canada); the Philippine Supreme Court (Virtucio et al., 2018); the Thai Supreme Court (Kowsrihawat et al., 2018); and the Turkish Constitutional Court (Sert et al., 2021). Many of these papers achieve a relatively high performance using various machine learning techniques.

The distinction between different tasks in this thesis is conditional on the data, but not on the algorithms used. Consequently, we discuss the papers from the perspective of which data was used, how it was processed, and the general performance of the systems when using particular data for a particular task. We do not go into detail regarding the algorithms used to achieve that performance. For the specifics of different systems, we therefore refer interested readers to the papers at hand. For a more detailed explanation of machine learning classification for legal texts in general, see Chapter 3 and Dyevre (2020).

4.3 Terminology and Types of Judgement Classification

In papers that use machine learning to classify court decisions, different terms and types of tasks are often used interchangeably. For the field to move forward, we therefore argue for a stricter use of ter-

minology. Consequently, in this dissertation, we use 'judgement' to mean the text of a published judgement. While the word 'outcome' is a very general term, for the purpose of distinguishing between different tasks in the legal context, we define outcome as a specific, closed class of labels for verdicts (i.e. with a pre-defined, limited number of verdicts). For example, in the context of case law concerning the European Convention on Human Rights (ECHR), the outcome will be either a *violation* or a *non-violation* of a specific human right. Other examples of outcomes are *eviction* or *non-eviction* in a housing law context (Vols, 2019), or the US Supreme Court affirming or reversing the decision of a lower court. We use 'verdict' and 'decision' as synonyms of 'outcome'.

In this chapter, we will distinguish between three types of tasks: *outcome identification*, *outcome-based judgement categorisation*, and *outcome forecasting*.[4] In simple terms, outcome identification is the task of identifying a verdict in the full text of a published judgement, judgement categorisation is the task of categorising documents based on the outcome, and outcome forecasting is the task of predicting the future decisions of a particular court. At present, these tasks are not distinguished clearly in the literature, even by ourselves (Medvedeva et al., 2020a). This is potentially problematic, as the different tasks have specific uses, which we will discuss below.

The most likely reason for ambiguity in the terminology is the

[4]In principle, there are three additional tasks, namely *charge identification*, *charge-based judgement categorisation* and *charge forecasting*. These tasks involve determining the specific sentence or charge. For example, the number of years someone was sentenced to go to prison in criminal court proceedings. These tasks have most often been investigated for various courts in China (Luo et al., 2017; Ye et al., 2018; Jiang et al., 2018; Liu and Chen, 2018; Zhong et al., 2018a,b; Li et al., 2019; Chen et al., 2019; Long et al., 2019; Chao et al., 2019; Fan et al., 2020; Cheng et al., 2020; Tan et al., 2020; Huang et al., 2020). However, the distinctions we make in this chapter between identification, categorisation and forecasting (and the pitfalls and suggestions regarding this distinction) hold for these cases as well.

cross-disciplinary nature of the field, combining law with NLP. When using machine learning in the field of NLP, all three tasks are so-called classification tasks. The most commonly used approach in machine learning, and the one which all of the reviewed papers have used, is *supervised learning*. This means that the system is trained on some input data (e.g., facts extracted from a criminal case) that are connected to the labels (outcomes); for instance, whether or not the case was won by the defendant or the prosecution. During the training phase, the model is presented with input data and their labels, so that patterns characterising the relationship between the two can be inferred. To evaluate the system after training, it is provided with similar data (*not* used during the training phase), such as other criminal cases, and it then *predicts* the label for each document. Since the label in each task is the outcome, identifying the purpose of these systems within NLP as 'predicting court decisions' is appropriate. However, outside of the NLP domain, this phrase does not translate in the same way. Specifically, the word *predict* in the legal domain suggests that one can forecast a decision (of a judge) that has not been made yet, whereas in NLP *predict* merely refers to the methodology and terminology of machine learning. Today, the majority of published papers on *predicting* court decisions do not attempt to predict decisions for cases that have not yet been judged. Furthermore, the majority of the work in this interdisciplinary field suggests a benefit for legal professionals, but it does not explicitly specify the potential application(s) of the models that were introduced.

To circumvent the use of the ambiguous word, *predict*, we therefore suggest using terminology that better reflects the different tasks, and thereby also differentiates between objectives. In order to distinguish between outcome identification, outcome-based judgement categorisation, and outcome forecasting, it is important to carefully assess the data used in the experiments conducted.

When discussing different papers, we will also refer to their per-

formance scores. The conventional way to report the performance of a classification system is by using accuracy, or the f1-score (see Chapter 3).

In the following subsections we make the definitions of the three tasks more explicit and give examples, from published research, of each task. We also highlight how legal professionals can make specific use of the different tasks.

4.3.1 Outcome Identification

Outcome identification is defined as the task of identifying the verdict within the full text of a judgement, including (references to) the verdict itself. In principle, a machine learning system is often not necessary for such a task, as a keyword search (or using simple regular expressions) might suffice.

Outcome identification falls under the field of information extraction and, when not confused with predicting court decisions, is often also referred to as outcome extraction (e.g., Petrova et al., 2020). Given the growing body of published case law across the world, the automation of this task may be very useful, since many courts publish case law without any structured information (i.e. metadata), other than the judgements themselves. Often, in order to conduct research, a database where the judgements are connected to the verdicts is required. At present, and to our knowledge, most of such work is generally done manually, as a human can do this task with 100% accuracy (by simply reading the case and finding the verdict within it).

The automation of outcome identification allows one to save time when collecting this information. While the task is not necessarily always trivial for a machine, and it depends on how the verdict is formulated (see, for instance, Vacek and Schilder, 2017; Petrova et al., 2020; and Tagny-Ngompé et al., 2020), there is nonetheless an expectation that, to justify automation, automated systems should achieve (almost) perfect performance. However, the approach to outcome

Paper	Court	Max. performance
Aletras et al. (2016)	ECtHR	79%
Liu and Chen (2017)	ECtHR	88%
Sulea et al. (2017); Şulea et al. (2017)	French Court of Cassation	99%
Virtucio et al. (2018)	Philippine Supreme Court	59%
Lage-Freitas et al. (2019)	Brazilian courts (appeal)	79% (F1)
Visentin et al. (2019)	ECtHR	79%
Bertalan and Ruiz (2020)	São Paolo Justice Court	98% (F1)
Quemy and Wrembel (2020)	ECtHR	96%

Table 4.1 | Research that falls under the category of outcome identification, including relevant court, and the (best) performance. When instead of accuracy, the f1-score (the average between precision and recall) is used as a performance indicator, this is indicated.

identification is highly dependent on both the structure of judgements in a particular legal domain or jurisdiction, and the language of the case law. As a result, a system that automatically identifies a verdict in a particular set of judgements cannot be applied easily to the case law of courts in other legal domains or jurisdictions.

4.3.1.1 Research in Outcome Identification

Eight papers trying to predict court decisions (see Table 4.1) performed the outcome identification task. These papers used text from final judgements published by the court, which contained either references to the verdict, or the verdict itself.

One of the earliest papers that tried to predict court decisions using text from judgements is Aletras et al. (2016). The authors used

a popular machine learning algorithm, a support vector machine (SVM) to predict the decisions of the European Court of Human Rights (ECtHR). Their model aimed to predict the court's decisions by extracting the available textual information from relevant sections of the ECtHR judgements, and it reached an average accuracy of 79% for 3 separate ECHR articles. While the authors excluded the verdict itself (or the complete section containing the verdict), they did use the remaining text of the judgements, which often included specific references to the final verdict (e.g., 'Therefore there is a violation of Article 3'). While their work was positioned as predicting the outcome of court cases, the task conducted was restricted to outcome identification.

Other studies focussing on the ECtHR included Liu and Chen (2017), Visentin et al. (2019), and Quemy and Wrembel (2020). Since Liu and Chen (2017) and Visentin et al. (2019) used the same dataset as Aletras et al. (2016), they also conducted outcome identification. Liu and Chen (2017) used similar statistical methods as Aletras et al. (2016) and achieved an 88% accuracy using an SVM, whereas Visentin et al. (2019) achieved an accuracy of 79%, using an SVM ensemble. Whereas Quemy and Wrembel (2020) collected a larger dataset for the same court, and performed a binary classification task (violation of any article of the ECHR vs. no violation) using neural models (e.g., AdaBoost), they did not appear to exclude any part of the judgement, thereby also restricting their task to outcome identification (with a concomitant high accuracy of 96%, using a range of statistical methods). These studies show that automatic outcome identification is, to a large extent, possible for the ECtHR. However, from a legal perspective, this task is not very useful, as the verdict has already been categorised on the ECtHR website.

These ECtHR studies illustrate two broad categories of papers which aim to predict court judgements, but are instead outcome identification tasks. The first category consists of studies which were only partially successful in removing the information about (references

to) the verdict. In addition to the aforementioned studies of Aletras et al. (2016), Liu and Chen (2017) and Visentin et al. (2019), the studies of Şulea et al. (2017) and Sulea et al. (2017) suffer from the same problem. They focus on the French Court of Cassation, and reach an accuracy of up to 96% using an SVM ensemble (a set of multiple SVMs). While the studies masked the words containing verdicts, various words that were found to be important for the prediction of their model appeared to be closely related to the outcome description. Consequently, they were not completely successful in filtering out information about the outcome.

The second category consists of studies which do not filter any information out of the judgement at all (or do not mention filtering out this type of information), such as Quemy and Wrembel (2020). Virtucio et al. (2018) are explicit about not filtering out the actual court decision of the Philippine Supreme Court (due to a lack of consistent sectioning in the judgement descriptions), when predicting its judgement. Nevertheless, their accuracy was rather low, at only 59%. In addition, a number of papers do not specify any pre-processing steps for removing information that may contain the verdict. Examples are Lage-Freitas et al. (2019), who deal with the appeal cases of Brazilian courts (with an f1-score of 79%), and Bertalan and Ruiz (2020), who worked on second-degree murder and corruption cases tried in the São Paolo Justice Court (with an f1-score of up to 98%).

4.3.2 Outcome-based Judgement Categorisation

Outcome-based judgement categorisation is defined as categorising court judgements based on their outcome, by using textual or any other information published with the final judgement, but excluding (references to) the verdict in the judgement. Since the outcomes of such cases are published and no longer need to be 'predicted', this task is mainly useful for identifying predictors (facts, arguments, judges, etc.) of court decisions within judgement texts. To avoid the

Paper	Court	FI	Max. performance
Kowsrihawat et al. (2018)	Thai Supreme Court	✗	67%
Chalkidis et al. (2019)	ECtHR	✓	82% (F1)
Kaufman et al. (2019)	SCOTUS	✗	77%
Kaur and Bozic (2019)	ECtHR	✗	82%
O'Sullivan and Beel (2019)	ECtHR	✗	69%
Chalkidis et al. (2020)	ECtHR	✗	83% (F1)
Condevaux (2020)	ECtHR	✗	88%
Medvedeva et al. (2018, 2020a)	ECtHR	✓	75%
Salaün et al. (2020)	Québec Rental Tribunal	✗	85%
Shaikh et al. (2020)	Delhi District Court	✗	92%
Strickson and De La Iglesia (2020)	UK highest Court of Appeal	✓	69%
Sert et al. (2021)	Turkish Constitutional Court	✗	98% (F1)
Malik et al. (2021)	Indian Supreme Court Court	✗	77%
Medvedeva et al. (2021b)	ECtHR	✗	92% (F1)

Table 4.2|Research that falls under the category of outcome-based judgement categorisation, including the relevant court, whether or not the most important features were extracted (FI), and the best performance achieved. When the f1-score (the average between precision and recall) is used as a performance indicator, this is indicated.

system *identifying* the outcome within the text of the judgement, and in order for it to learn new information, any references to the verdict need to be removed.

While an algorithm may perform very well on the categorisation task, the categories obtained are not useful on their own. As the documents used by the system are only available when judgements have been made public, outcome categorisation does not contribute any new information (one can simply extract the verdict from the published judgement). This view is also supported by Bex and Prakken (2021), who insist that the ability to categorise decisions without explaining why the categorisation was made does not provide any useful information, and it may even be misleading. The performance of a machine learning model for judgement categorisation, however, may provide useful information about how informative the characteristic features are. To enable feature extraction, it is important that the system is not a 'black box' (such as many of the more recent neural classification models). Therefore, rather than 'predicting court decisions', the main objective of the outcome-based judgement categorisation task should be to identify *predictors* underlying the categorisations.

As we only discuss publications that categorise judgements on the basis of the outcome of the case, we will refer to outcome-based judgement categorisation simply as 'judgement categorisation'.

4.3.2.1 Research in Outcome-based Judgement Categorisation

Most of the papers in the field categorise judgements. The papers surveyed that involve judgement categorisation can be found in Table 4.2. For all fifteen papers, we indicate the paper itself, the court, whether or not the authors provide a method of analysing feature importance (FI) and (consequently) identify specific predictors of the outcome within the text, and the maximum performance.

Within these studies two broad categories can be distinguished, according to the type of data they use. On the one hand, most studies

use the raw text, explicitly selecting parts of the judgement which do not include (references to) the verdict. On the other hand, there are (fewer) studies which manually annotate data to use as a basis for categorisation.

Kowsrihawat et al. (2018) used raw text to categorise (with an accuracy of 67%) Thai Supreme Court documents using attention, on the basis of text relating to the facts and legal provisions (such as murder, assault, theft, fraud and defamation) of each case, using a range of statistical and neural methods. In our work (Medvedeva et al., 2018, 2020a; see also Chapter 6), we categorised (with an accuracy of up to 75%) decisions of the ECtHR, using only the facts of each case (i.e. a separate section in each ECtHR judgement). Notably, in Medvedeva et al. (2020a), we identified the top predictors (i.e. sequences of one or more words) for each category, which was possible due to the (support vector machine) approach we used.[5] Strickson and De La Iglesia (2020) worked on categorising UK Supreme Court judgements, and compared several systems trained on the raw text of each judgement (without the verdict), reporting an accuracy of 69% when using logistic regression, while presenting the top predictors for each class. Sert et al. (2021) categorised Turkish Constitutional Court cases related to public morality and freedom of expression, using a traditional neural multi-layer perceptron approach with an average accuracy of 90%. Similarly to Medvedeva et al. (2020a), Chalkidis et al. (2019) also investigated the ECtHR using the facts of each case, proposing several neural methods to improve categorisation performance (up to 82%). They also proposed an approach (a hierarchical attention network) for identifying which words and facts were most important for the classification of their systems. In their subsequent study, Chalkidis et al. (2020) used a more sophisticated neural categorisation algorithm that was specifically tailored for legal data (LEGAL-BERT). Unfortunately, while their approach did show an improved perfor-

[5] See also Chapter 6 for more details on this study.

mance (with an f1-score of 83%), it was not possible to determine the best predictors of the outcome, due to the system's complexity. In our later work, (Medvedeva et al., 2021b), we reproduced the algorithms in Chalkidis et al. (2019) and Chalkidis et al. (2020), in order to compare their categorisation and forecasting task performance (see Section 4.3.3) for a smaller subset of ECtHR cases, and an f1-score of up to 92% was achieved for the categorisation of judgements from 2019, although the score varied throughout the years. For example, categorisation of cases from 2020 did not surpass 62%. Several other categorisation studies (with accuracies ranging between 69% and 88%) focused on the facts of the ECtHR, but likewise did not investigate the best predictors (Kaur and Bozic, 2019; O'Sullivan and Beel, 2019; Condevaux, 2020). Kaur and Bozic (2019) use convolutional neural networks to achieve their results, O'Sullivan and Beel (2019) experimented with a variety of different neural and non-neural methods (i.e. SVM, Gradient Boosting, RF, AdaBoost, Decision Tree, among others), where each algorithm yielded better results for some of the data, but not for the other. Condevaux (2020) achieved their best performance using attention networks. Malik et al. (2021) used neural methods to develop a system that categorised Indian Supreme Court decisions, achieving 77% accuracy. As their main focus was to develop an explainable system, they used an approach which allowed them to investigate the importance of their features, using attention, somewhat similar to the approach of Chalkidis et al. (2020).

Manually annotated data was used by Kaufman et al. (2019), who focused on data from the US Supreme Court (SCOTUS) Database (Spaeth et al., 2014) and achieved an accuracy of 75% using statistical methods (i.e. AdaBoosted decision trees). However, they did not investigate the most informative predictors. Shaikh et al. (2020) also used manually annotated data to categorise murder case decisions from Delhi District Court with an accuracy of up to 92%, using classification and regression trees. The authors manually annotated 18

features, including whether the injured person was dead or alive, the type of evidence, the number of witnesses, etc. Importantly, they analysed the impact of each type of feature, for each type of outcome.

Finally, Salaün et al. (2020) essentially combined the two types of predictors, by not only extracting a number of characteristics from cases from the Rental Tribunal of Quebec (including the court location, judges, types of parties, etc.), but also by using the raw text of the facts (as well as the complete text, excluding the verdict. This achieved a performance of up to 85% with the French BERT model, FlauBERT.

Notably, the performance of Sert et al. (2021) was very high (f1-score 98%) when using a multi-layer perceptron. Despite the high success rate of their system, however, the authors warn against using it for decision-making. Nevertheless, they do suggest that their system could be used to prioritise cases that have a higher likelihood of resulting in a violation. This suggestion mirrors the proposition made by Aletras et al. (2016), that their system could be used to prioritise cases with human rights violations. In both cases, however, the experiments were conducted using data extracted from the courts' final judgements, so the performance of these systems using data compiled before the verdict was reached (i.e. information necessary to prioritise cases) is unknown. Making these types of recommendations is therefore potentially problematic.

Many categorisation papers shown in Table 4.2 claim to be useful for legal aid. However, as we argued before, categorisation as such is not useful, because the verdict can simply be read in the judgement text. To be useful, it is essential that categorisation performance is supplemented with the most characteristic features (i.e. predictors). Unfortunately, only a small number of studies provide this information. Even then, the resulting features may not be particularly meaningful, especially when using the raw text (i.e. characteristic words or phrases).

Aiming for maximum explanation, Collenette et al. (2020) suggest using an Abstract Dialectical Framework, instead of machine learning. They apply this framework to deducing the verdict from the text of ECtHR judgements, regarding Article 6 of the ECHR (the right to a fair trial). The system requires the user to answer a range of questions and, on the basis of their answers, the model determines whether or not there was a violation of the right to a fair trial. Questions for the system were derived by legal experts, and legal expertise is also required to answer these questions (Collenette et al., 2020). While the Abstract Dialectical Framework system seemed to perform flawlessly when tested on ten cases, it faces the same issue as the machine learning systems. Specifically, the main input data is based on the final decision, which has already been made by the judge. For instance, one of the questions that the model requires to be answered is whether or not the trial was independent and impartial, a point which must be decided by the judge. While this type of tool could be used for judicial support (for example, as a checklist for a judge when making a specific decision), it is unable to actually either forecast decisions in advance, or point to external factors that have not been identified by legal experts.

4.3.3 Outcome Forecasting

Outcome forecasting is defined as determining a court verdict by using textual information from a court case, which was available *before* the verdict was made (public). This textual information can (for instance) be submissions by the parties, or information (including judgements) provided by lower courts to predict the decisions of a higher court, such as the US Supreme Court. Forecasting thereby entails an essential assumption that the input for the system was not influenced in any way by the final outcome that it forecasts. In contrast to *outcome-based judgement categorisation*, it is useful to evaluate how well the algorithm predicts the outcome of cases. For exam-

ple, individuals may use such algorithms to evaluate the likelihood of them winning a court case. Similarly to *judgement categorisation*, determining the factors underlying a well-performing model is also useful. While identification and categorisation tasks only allow for the extraction of information and the analysis of past court decisions, forecasting allows for the prediction of future decisions. It should be noted that it does not matter if a model was trained on older cases than those on which it was evaluated (e.g., the 'predicting the future' experiment conducted by us in Medvedeva et al., 2020a), because this does not affect its classification as a judgement categorisation, as opposed to a judgement forecasting task. The type of data is the only factor affecting task classification. Since we uses data extracted from court judgements in Medvedeva et al. (2020a), the task was still an outcome-based judgement categorisation task.

4.3.3.1 Research in Outcome Forecasting

Table 4.3 lists the papers that focus on forecasting court verdicts. While many publications focus on 'predicting court decisions', only five papers satisfy our criteria for outcome forecasting. We can observe that the performance of these studies is lower than for the categorisation and identification tasks. This is not surprising, as forecasting can be expected to be a harder task. As there are only a small number of papers, we discuss each of them in some detail.

The advantage of working with the US Supreme Court databases is that it attracts much attention. Consequently, the trial data are always systematically and manually annotated by legal experts with many variables, immediately after a case has been tried. Sharma et al. (2015) and Katz et al. (2017) both use variables that were available to the public once the case had been moved to the Supreme Court, but before the decision was made by the SCOTUS. Sharma et al. (2015) use neural methods, whereas Katz et al. (2017) use the more traditional technique of random forests. Both approaches resulted in forecast-

Paper	Court	Data	Max. performance
Sharma et al. (2015)	SCOTUS	Court of Appeal info	70%
Katz et al. (2017)	SCOTUS	Court of Appeal info	70%
Waltl et al. (2017)	German Court of Appeal (Tax Law)	Decision of the lower (fiscal) courts	57% (F1)
Medvedeva et al. (2020b)	ECtHR	Facts as communicated to the parties	75%
Medvedeva et al. (2021b)	ECtHR	Facts as communicated to the parties	66% (F1)

Table 4.3 | Research that falls under the category of outcome forecasting, including relevant court, the data used for forecasting, the best performance. When instead of accuracy, the f1-score (the average between precision and recall) is used as a performance indicator, this is indicated.

ing 70% of the outcomes correctly, which was a small improvement on the 68% baseline accuracy, in which the petitioner always wins (suggested by Kaufman et al., 2019). Moreover, Sharma et al. (2015) present the importance of different variables in their model, potentially enabling a more thorough legal analysis of the data. The variables used in both studies contained information about the courts and proceedings, but hardly any variables pertaining to the facts of the case.

Waltl et al. (2017) attempted to forecast German appeal court decisions on matters of tax law (the Federal Fiscal Court). The authors used the documents and meta-data for the case (e.g., year of dispute,

court, Chamber, duration of the case, etc.) from the court of first instance. They extracted keywords from the facts and (lower) court reasoning, in order to forecast decisions. They tried a range of methods, but selected the best-performing naive Bayes classifier as their final model. However, their relatively low f1-score (0.57) indicates that it may have been a rather difficult task.

Medvedeva et al. (2020b)[6] used raw text and facts within documents that were published by the ECtHR, (sometimes) years before a final judgement was made. These documents are known as 'communicated cases'. Specifically, facts presented by the applicant, then communicated by the court to the state as a potential violator of human rights, were used. Communicated cases reflect the side of the potential victim, and are only communicated when no similar cases have been processed by the court before. Consequently, these documents include a very diverse set of facts, and they cover different issues (although they are all within the scope of the European Convention on Human Rights). Medvedeva et al. (2020b) reported an accuracy of 75%, using SVMs on their dataset (the model is automatically re-trained and run again every month). This system is integrated in an online platform that also highlights sentences or facts within the text of the (communicated) cases that are most important for the model's decision.[7] Medvedeva et al. (2021b)[8] used a slightly different dataset of the same documents (i.e. only cases with the English judgement were included, but the dataset was expanded by adding cases that resulted in inadmissibility based on merit), and retrained the model every year (as opposed to every month, as in Medvedeva et al., 2020b). The authors compared how the state-of-the-art algorithms for this court, BERT (Chalkidis et al., 2019), LegalBERT (Chalkidis et al., 2020), and SVMs (Medvedeva et al., 2020a,b) performed on data avail-

[6] See also Chapter 8 for all details of this study.

[7] `https://jurisays.com`

[8] See also Chapter 7 for all details of this study.

able *before* the final judgement and the final judgements themselves. The results showed that forecasting is indeed a much harder task, as the forecasting models achieved a maximum f1-score of 66%, as opposed to 92% for categorisation of the same cases.

4.4 Discussion

It is clear that 'predicting court decisions' is not an unambiguous task. Therefore, there is a clear need to carefully identify the *objective* of the experiments before conducting them. We believe such an objective has to be rooted within the specific needs of the legal community, in order to prevent systems being developed which authors believe to be useful, but which have no meaningful application in the legal field. The purpose of this chapter has been to provide some terminology which may be helpful in this endeavour.

While researchers may believe they are 'predicting court decisions', this very seldom involves actually being able to predict the outcome of *future* judgements. In fact, the prediction of court decisions sometimes (likely inadvertently, due to sub-optimal filtering or insufficient knowledge about the exact dataset) only resulted in identifying an outcome from a judgement text. While sophisticated approaches were often put forward in such cases, a simple keyword search might have resulted in higher performance for this identification task. Most often, however, predicting court decisions was found to be equal to the task of categorising the judgements according to the verdicts. This is not so surprising, given the available legal datasets, which are more likely to contain complete judgements than documents which were produced before the verdict was known.

In sum, to identify the exact task and the concomitant goals which are useful from a legal perspective, it is essential that researchers are well aware of the type of data they are analysing. Unfortunately, this is frequently not the case. For example, several researchers

(Chalkidis et al., 2019; Quemy and Wrembel, 2020; Condevaux, 2020) have recently started to develop (multilabel classification) systems, which are able to predict the articles which would be invoked in an ECtHR case. However, this task is not relevant from a legal perspective, as articles which are potentially violated have to be specified when petitioning the ECtHR.

Therefore, when creating a new application (for instance, using data from another court), the goal of such a system should first be clearly determined, after which a review of the data is necessary to establish if the data for the established task is available. Specifically, one needs full judgements for the outcome identification task. In the case of a judgement categorisation task, full judgements from which the outcomes can be removed are necessary. If the system needs to perform a forecasting task, it requires data that are available before the judgement is made.

For all of the above tasks, explainability (i.e. being able to determine the importance of various features, when determining the model's outcome) helps to improve performance analysis and gain insight into the workings of the system. However, explainability is *essential* for judgement categorisation, as this task is reliant on the ability to investigate which features are related to the outcome.

As we mentioned before, the identification task does not always require the use of machine learning techniques. This task can often be solved with a keyword search, which does not require any annotated data. The use of machine learning is necessary when a judgement text is not very structured, and when more complex descriptions of the outcome need to be extracted. For both the judgement categorisation task and the forecasting task, statistics may be useful to assess the relationship between predetermined factors and the outcome. However, for the categorisation task, machine learning techniques may allow for the discovery of new patterns and factors within the judgements that may not have been previously considered.

Similarly, machine learning techniques can be used to forecast future court decisions, by training the system on decisions that the court has made in the past. Figure 4.1 illustrates these three tasks, their goals, and their requirements.

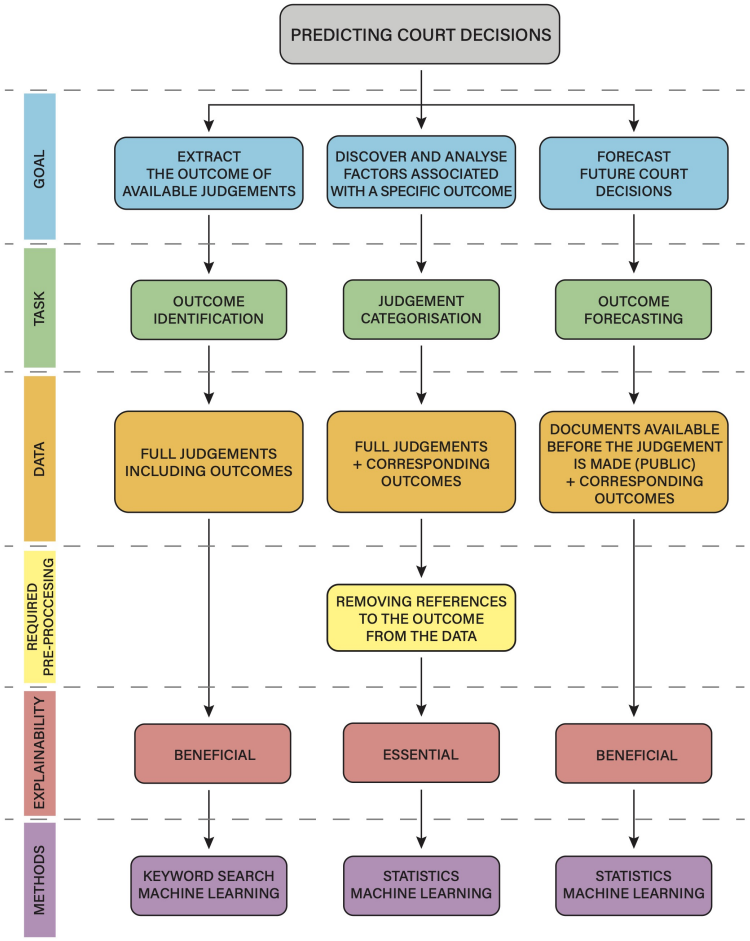


Figure 4.1 | Flowchart illustrating the goals and requirements for the three court decision prediction tasks

Finally, we would like to emphasise that, while the approaches discussed in this chapter are suitable for use in legal analysis (for example, to try to understand past court decisions), none of the systems capable of solving any of the discussed tasks are appropriate for *making* court decisions. Judicial decision-making requires (amongst other things) knowledge of the law, knowledge of our ever-changing world, and the skilled weighing of arguments. This is very different from the (sometimes very sophisticated) pattern-matching capabilities of the systems discussed in this chapter.

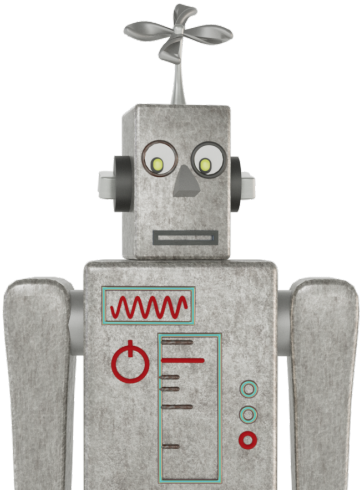
4.5 Conclusion

In this chapter, we have proposed several definitions for analysing court decisions using computational techniques. Specifically, we discussed the difference between forecasting decisions, categorising judgements according to the verdict, and identifying the outcome based on the text of a judgement. We also highlighted the specific goals potentially associated with each task, and illustrated how each task is strongly dependent on the type of data used.

The availability of enormous amounts of legal (textual) data, in combination with the legal discipline being relatively methodologically conservative (Vols, 2021), has enabled researchers from various other fields to attempt to analyse these data. However, to conduct meaningful tasks, we argue for more interdisciplinary collaborations, not only involving technically skilled researchers, but also legal scholars. This should ensure that meaningful legal questions are answered, while enabling this new and interesting field to be propelled forwards.

PART II

Experiments



CHAPTER 5

Automatically Identifying Eviction Cases and Outcomes within the Case Law of Dutch Courts of First Instance

In this chapter, we attempt to identify eviction judgements within case law published by Dutch courts, in order to automate data collection, which was previously undertaken manually. To do so, we performed two experiments. The first focused on identifying judgements related to eviction, while the second focused on identifying the outcome of the cases in the judgements (eviction vs. dismissal of the landlord's claim). In the process of conducting the experiments for this study, we have created a manually annotated dataset of eviction related judgements and their outcomes.

Chapter adapted from:

Medvedeva, M., Dam, T., Wieling, M., and Vols, M. (2021a). Automatically identifying eviction cases and outcomes within case law of Dutch courts of first instance. In *Legal Knowledge and Information Systems*, pages 13–22. IOS Press.

5.1 Introduction

Legal scholars and practitioners are confronted with an enormous and expanding body of case law. For example, in the Netherlands the judiciary dealt with over 1.3 million cases in 2020 alone.[1] Many of these cases involve bulk cases on, for example, family law or labour law. Another area that results in a considerable number of bulk cases is landlord-tenant law. It is estimated that courts have to decide whether or not a tenant needs to be evicted in nearly 20,000 cases every year (Vols, 2018). The Dutch judiciary does not publish all its judgements online, but a significant number of cases can be found online, in the *Open Data van de Rechtspraak* dataset.[2] Traditionally, legal scholars and practitioners collect and analyse these cases manually (Vols, 2021). Of course, this is time-consuming, and it will eventually become impossible, due to the increasing amount of case law published online.[3]

In this chapter, we are trying to provide a solution for this legal research issue. Specifically, we want to identify judgements concerning eviction within all the judgements published by the Dutch judiciary, and extract their outcome from the text (i.e. eviction/non-eviction). This work builds upon existing research that until now has been done manually (Vols, 2019), and our goal is to test how much of the data collection and outcome extraction can be automated. Some of the case law under review has already been annotated by hand, and it can be used to train machine learning systems.

In this chapter, we use 'judgement' to denote the text of a pub-

[1] `https://www.rechtspraak.nl/Organisatie-en-contact/Rechtspraak-in-Nederland/Rechtspraak-in-cijfers` (in Dutch), accessed on 04/04/2022

[2] `https://www.rechtspraak.nl/Uitspraken/Paginas/Selectiecriteria.aspx` (in Dutch), accessed on 04/04/2022

[3] `https://www.volkskrant.nl/nieuws-achtergrond/raad-voor-de-rechtspraak-meer-vonnissen-online-publiceren~bf045df7` (in Dutch), accessed on 04/04/2022

lished judgement. The word 'outcome', and its synonyms 'verdict' and 'decision', are used to define a specific closed class (i.e. a limited number) of labels for verdicts. An example of an outcome, in the landlord-tenant law context, is *eviction* or *non-eviction*.

5.2 Related Work

This chapter deals with the identification of a judgement topic (i.e. an eviction case or a non-eviction case). To our knowledge, the number of publications dealing with the automatic identification of a case topic for dataset creation is limited. Similar work involves topic modelling techniques which allow for the identification and clustering of multiple topics at once (Dyevre, 2020; Silveira et al., 2021; Remmits, 2017), and the use of document similarity to find documents dealing with similar issues (Novotná et al., 2020; Barco Ranera et al., 2019), both of which can be particularly hard to evaluate.

Besides the identification of a judgement topic, this chapter concerns outcome identification (i.e. extraction of the outcome from the full text). This identification task can be useful in itself, for instance, if one wants to know the statistics for cases concerning eviction that actually resulted in a tenant being evicted.

Depending on the court, identifying the outcome can be more or less complicated. Some courts publish their judgements with metadata stating the outcome (e.g., the European Court of Human Rights). As a result, one simply needs to extract this information to obtain outcomes. In other judgements, the wording of the outcome may be standardised and therefore easy to extract (e.g., 'The Court of Appeal therefore affirms the decision of the Court of First Instance'). However, the majority of courts seem to formulate their decisions in free-form natural language, making the task of extracting a specific outcome a more complex task.

A small number of studies focus specifically on identifying the out-

come within judgements. Recent papers extracted outcomes from appellate decisions in US State courts (Petrova et al., 2020; f1-score: 0.82), and US federal court dockets (Vacek and Schilder, 2017; recall up to 0.96), as well as from French courts (Tagny-Ngompé et al., 2020; f1-scores: 0.8 up to 1.0), using various machine learning methods. In this chapter, we compare the performance of a (simpler) keyword search approach (not requiring any annotated data) with that of a simple machine learning system.

5.3 Data

For our dataset we rely on the *Open Data van de Rechtspraak*,[4] which is the official, publicly available, database of the Dutch judiciary (De Rechtspraak). Not all Dutch case law is published online, but merely a subset of judgements that *De Rechtspraak* permits to be published. Unfortunately, the exact criteria for this process are not available to the public, although some guidance is provided on a dedicated page on the Dutch judiciary website.[5] The *Open Data van de Rechtspraak* dataset can be downloaded as one large archive (>4GB) of XML files, containing the texts of the judgements and some basic meta-data (e.g., court, date).

For this work, we are specifically interested in the cases of the courts of first instance (*rechtbanken*). A collection of 591 eviction cases between 2000 and 2020 (manually collected and annotated, including the verdicts: eviction or non-eviction) was already available from the courts of first instance, based on existing research from our lab. This dataset was compiled with the aim of including the majority of all published eviction cases between 2000 and 2020. As this dataset only contains a relatively limited number of eviction cases,

[4] `https://www.rechtspraak.nl` (in Dutch)

[5] `https://www.rechtspraak.nl/Uitspraken/Paginas/Selectiecriteria.aspx` (in Dutch), accessed on 04/04/2022

and no non-eviction cases, we aimed to supplement it by including cases about eviction and cases on other topics (still somewhat related to the subject matter). This was to ensure that the task was useful (not trivial), and that we had a larger dataset for training the system.

To increase the likelihood of identifying eviction cases, we used the following procedure. We extracted all (2,641,946) judgements between 2000 and 2020 from the *Open Data van de Rechtspraak* dataset. From this set, we included only judgements from the courts of first instance, and furthermore selected the judgements that contained at least one of the following words: *huurovereenkomst* (rental agreement), *ontruiming* (eviction), or *woning* (home). Subsequently, we narrowed down the selection by only retaining judgements with the label 'private law', which is the appropriate label for eviction cases. These relatively simple filters allowed us to reduce the amount of judgements to 24,268 cases. Unfortunately, this number was still rather large. Consequently, we made a further reduction by only including cases from 2016 to 2018, and excluding cases already included in the original set of 591 cases, yielding a set of 4,795 judgements. From this set, we randomly sampled 69 judgements (1 hour of manual annotation) to assess the proportion of cases related to eviction. A manual inspection showed that more than half of the judgements (37) were eviction cases. This suggests that our manually curated dataset of 591 eviction cases was missing a substantial amount of eviction related cases.

To increase the amount of data, we took all 591 manually annotated eviction judgements and, once again, randomly sampled from the 2016 to 2018 judgements, extracting twice the amount (1,182) of manually annotated eviction judgements. We then built a simple three-fold cross-validation support vector machine (SVM), only using 1-3 n-grams (i.e. sequences of one to three words from the text of a judgement) as features. When training the model, we treated the 591 judgements as eviction cases and the 1,182 judgements as non-

eviction cases.[6] Of course, this is a sub-optimal class distinction, as potentially many of the 1,182 judgements may, in fact, be eviction cases. Consequently, out of all the cases that were classified as non-eviction cases, we only retained those which were (when included in the test set during the three-fold cross-validation procedure) assigned the non-eviction label with over 99% confidence (using Platt Scaling; Platt et al., 1999). This reduced the number of non-eviction judgements in our training set to 809. We then trained the system again (using 809 non-eviction cases, and 591 eviction cases), and evaluated it using the rest of the judgements from between 2000 and 2020. Out of 22,868 judgements, 3,277 (14%) were predicted as eviction related.

Of course, not all predictions will be correct. To supplement our final, correct training dataset, we did not use these predictions. Instead, we used them simply to guide two subsequent manual annotation rounds (the first annotation round included the 69 aforementioned cases). Specifically, we asked two legal experts to spend a total of eight hours annotating judgements that our model had predicted as eviction related in the second annotation round (under the assumption that many would *not* be eviction related), and an additional total of four hours on a third annotation round, focussing on judgements that our model had predicted as *non-eviction*. The annotators were provided with the full text of a randomly selected judgement and they simply had to identify whether the judgement concerned an eviction or not. In the allocated time, 716 judgements were annotated. Out of the predicted eviction judgements, 298 (55%) turned out to be eviction related, while 243 judgements (45%) were not. In addition, the vast majority of non-eviction cases – 161 out 175 (92%) – indeed turned out to be non-eviction related. The manual annotation rounds left us with a dataset of 940 eviction judgements, and 436

[6]For a more detailed explanation of machine learning classification and its evaluation (i.e. precision, recall, f1-score, accuracy), as applied to legal texts, see Chapter 3.

non-eviction judgements. Table 5.1 provides an overview of our final dataset.

	Eviction	**Non-eviction**
Initial dataset	591	0
First annotation round	37	32
Second annotation round (predicted as eviction)	298	243
Third annotation round (predicted as non-eviction)	14	161
Total	940	436

Table 5.1|Number of available data in the initial dataset and after three rounds of annotation

After identifying the eviction related judgements, we were also interested in their outcome. In judgements concerning evictions, the courts of first instance can decide either to evict the resident and/or cancel the lease (labelled as *eviction*), or to reject the property owner's claim (labelled as *non-eviction*). The cases are decided on by a single judge. All of the eviction cases in the court of first instance are property owner vs. resident, with the latter being the defendant.

5.4 Experiment I: Identifying Eviction Related Judgements

5.4.1 Methodology

We used 200 judgements (100 eviction related and 100 non-related) from the final dataset (see Table 5.1) to test and evaluate the model, which left us with 840 eviction and 336 non-eviction judgements to use to train and fine-tune the system. We then balanced this dataset for training, leaving us with 336 eviction related judgements, and the same number of non-related judgements. We used three-fold cross-validation to fine-tune the parameters, and ended up using a linear support vector machine, using the frequencies of 1-6 character n-

grams (i.e. sequences of one to six characters) as features.[7] The results of the best model can be found in Tables 5.2 and 5.3.

	Precision	**Recall**	**F1-score**	**Support**
Non-eviction	0.90	0.88	0.89	336
Eviction	0.88	0.90	0.89	336
Accuracy			0.89	672
Macro avg.	0.89	0.89	0.89	672
Weighted avg.	0.89	0.89	0.89	672

Table 5.2 | Results (precision, recall, f1-score and accuracy) for identifying eviction related judgements using three-fold cross-validation

		Predicted topic	
		Non-eviction	Eviction
Actual topic	Non-eviction	294	42
	Eviction	32	304

Table 5.3 | Results (confusion matrix) for identifying eviction related judgements using three-fold cross-validation

5.4.2 Results

The final results, when evaluating our model on the held-out test set, are shown in Tables 5.4 and 5.5.

The results suggest that, when there is a reasonable amount of annotated data, it is possible to identify eviction related cases with a relatively high accuracy of about 88%. Consequently, this automatic procedure is suitable to speeding up the process of finding relevant (eviction related) case law.

[7] The following command, showing all used parameters, was used to fit our final model: `CountVectorizer(analyzer = 'char', ngram_range = (1,6), max_features = None, max_df = 0.7, lowercase = False, binary = True); LinearSVC(C = 0.01)`. For more details on each parameter, see the `sklearn` documentation available at `https://scikit-learn.org/`. The full set of parameters we experimented with can be found in our code and data, available at `https://github.com/masha-medvedeva/EVICT`.

	Precision	Recall	F1-score	Support
Non-eviction	0.92	0.81	0.87	100
Eviction	0.83	0.95	0.89	100
Accuracy			0.88	200
Macro avg.	0.89	0.88	0.88	200
Weighted avg.	0.89	0.88	0.88	200

Table 5.4 | Results (precision, recall, f1-score and accuracy) on the test set for identifying eviction related judgements

		Predicted topic	
		Non-eviction	Eviction
Actual topic	Non-eviction	81	19
	Eviction	5	95

Table 5.5 | Results (confusion matrix) on the test set for identifying eviction related judgements

When we evaluated the model on all of the (filtered) judgements published between 2000 and 2020, which were not included in our dataset, a total of 3,248 out of 22,872 cases (all original judgements between 2000 and 2020, excluding annotated judgements) were classified as eviction related judgements. With an estimated precision of 83%, we expect about 2,695 cases to be actual eviction related judgements. Similarly, with an estimated precision of 92% in identifying non-eviction related judgements, we expect an additional 8% of these (i.e. 1,569 judgements) to be eviction related.

5.5 Experiment II: Identifying the Outcome

5.5.1 Methodology

Once we had identified the eviction related judgements, we became interested in identifying how many of them actually resulted in the

eviction of a resident. Identifying the verdict should not necessarily always be a machine learning task. A simple keyword search could potentially be sufficient. Therefore, we first tried determining words that may be characteristic of a specific outcome, because, while courts of first instance judgements do not have a clear structure, they potentially use the same wording for different verdicts. We compared these results to those produced by a more a more sophisticated machine learning system, which is able to take more advanced features into account.

For this experiment, we used the full set of 940 eviction related cases shown in Table 5.1. Except for the cases included in the initial dataset, which already included an annotated outcome, we asked two legal experts to annotate the outcome of each case: *eviction* or *non-eviction*. We excluded 28 cases that had other types of verdicts, such as only cancellation of the lease, but no eviction, etc. The final dataset for this task contained 912 judgements (620 having an eviction outcome, and 292 having a non-eviction outcome).

5.5.1.1 Keyword-based System

For the keyword-based system, we determined (via manual inspection of several cases) a number of terms that relate to each specific outcome. We then automatically searched for these terms in the *decision* section of the published judgement, and in cases where the decision section was not specified, in the bottom part (2,500 characters) of the text. The keywords that we chose as representative of an *eviction* outcome were (including different forms of the same words): *ontbinding* (cancellation), *ontruiming* (eviction), and *verlaten* (leave). If none of these words were found, our keyword-based system determined that no eviction had been ordered by the court.

We tested the method on all 912 judgements for which we had labels. The results of this system can be found in Tables 5.6 and 5.7.

	Precision	Recall	F1-score	Support
Non-eviction	0.88	0.65	0.75	292
Eviction	0.85	0.96	0.90	620
Accuracy			0.86	912
Macro avg.	0.87	0.80	0.82	912
Weighted avg.	0.86	0.86	0.85	912

Table 5.6 | Results (precision, recall, f1-score and accuracy) for identifying the outcome of eviction cases using keyword extraction

		Predicted outcome	
		Non-eviction	Eviction
Actual outcome	Non-eviction	189	103
	Eviction	25	595

Table 5.7 | Results (confusion matrix) on the test set of identifying the outcome of eviction cases using keyword extraction

This simple system achieved reasonably good results, although we can see that non-eviction is not categorised very well: 103 (35%) out of 292 non-eviction cases were misclassified. However, the issue with a keyword-based system is that it is very hard to improve upon, unless one can come up with more specific keywords. Moreover, if the keywords from one type of outcome are found in a judgement with a different outcome, this is hard to correct. For instance, a judgement may contain the phrase, 'at this point, eviction is not necessary'. While the word 'eviction' is present in this judgement, the case clearly resulted in no eviction. However, since we are dealing with individual words only, it is hard to incorporate all the possible nuances.

Nonetheless, as opposed to a system using machine learning, which we will discuss in the next subsection, this system does not require any prior annotation, other than to determine the keywords.

5.5.1.2 Machine Learning System

During the keyword-based experiment, we determined that the outcome usually appears within the last 2,500 characters of the judgement. While we experimented with shorter and longer fragments, this subset seemed to work best in identifying the verdict, for both experiments. We used the initial dataset for training, and cases from the first, second and third rounds of annotation for testing. We built a linear SVM which uses character (1-7) n-grams, and optimised it for a number of other parameters.[8] The results of the model during the cross-validation stage can be found in Tables 5.8 and 5.9.

	Precision	**Recall**	**F1-score**	**Support**
Non-eviction	0.97	0.96	0.96	183
Eviction	0.98	0.99	0.98	379
Accuracy			0.98	562
Macro avg.	0.98	0.97	0.97	562
Weighted avg.	0.98	0.98	0.98	562

Table 5.8 | Results (precision, recall, f1-score and accuracy) for identifying the outcome of eviction cases using three-fold cross-validation

		Predicted outcome	
		Non-eviction	Eviction
Actual outcome	Non-eviction	175	8
	Eviction	5	374

Table 5.9 | Results (confusion matrix) for identifying the outcome of eviction cases using three-fold cross-validation

[8] The following command, showing all used parameters, was used to fit our final model: `CountVectorizer(analyzer = 'char', ngram_range = (1,7), max_features = 2000, max_df = 0.9, lowercase = True, binary = True); LinearSVC(C = 0.001)` The full set of parameters we experimented with can be found in our code and data available at `https://github.com/masha-medvedeva/EVICT`.

5.5.2 Results

We then tested the model on the cases that we were able to extract in the previous experiment. The performance on the test set can be found in Tables 5.10 and 5.11.

	Precision	**Recall**	**F1-score**	**Support**
Non-eviction	0.82	0.94	0.88	109
Eviction	0.97	0.91	0.94	241
Accuracy			0.92	350
Macro avg.	0.90	0.92	0.91	350
Weighted avg.	0.92	0.92	0.92	350

Table 5.10 | Results (precision, recall, f1-score and accuracy) on a test set for identifying the verdict of eviction cases

		Predicted outcome	
		Non-eviction	Eviction
Actual outcome	Non-eviction	102	7
	Eviction	22	219

Table 5.11 | Results (confusion matrix) on a test set for identifying the verdict of eviction cases

As we can see from the results, we were able to achieve a very high performance, especially for the eviction class. When inspecting the cases manually, it is clear that the phrasing of the judgement outcome varies to a large extent from case to case. As in many other natural language processing tasks, the best-performing model did not include word n-grams, but only character n-grams (Basile et al., 2017; Medvedeva et al., 2017). While we did try using word n-grams for this experiment, in the hope of identifying additional keywords for the keyword-based approach, we did not identify any additional unique words for both outcomes. The performance of the machine learning approach was substantially higher than that of the keyword-based approach. However, whereas the machine learning approach requires

annotated data, the keyword-based method does not.

5.6 Discussion and Conclusion

In this chapter, we have presented two experiments, one to identify case law concerning a certain topic (judgements concerning evictions), and one to subsequently identify the outcomes of these eviction judgements. Both tasks have shown a high performance. We were able to identify eviction related cases with 88% accuracy, whereas we were able to correctly identify the outcome in eviction related cases with 92% accuracy. While identifying this type of information may seem easy (as the information is available when reading the document), a keyword-based approach showed that it is not straightforward when the information is provided as natural text. While in this chapter we were not able to identify *all* eviction cases perfectly, our machine learning approach is suitable to use for identifying cases which are *likely* to be eviction cases. Manually checking this smaller set of cases (at a rate of about one case per minute) is feasible, whereas checking the full set is not. With relatively little effort, a new database containing thousands of cases can therefore easily be created.

A more restricted, subject-specific database is also useful in the context of an increasing body of research focussing on categorising or forecasting court decisions (Medvedeva et al., 2020a; Chalkidis et al., 2019; Katz et al., 2017; Waltl et al., 2017; Strickson and De La Iglesia, 2020). This type of research is mostly limited to a few courts, such as the US Supreme Court or the European Court of Human Rights. This is partly due to the courts' publishing policies, although more and more courts are now publishing their case law. The dominant focus on a few courts, however, is also caused by the relatively wide diversity of uncategorised cases in other courts. Therefore, narrowing down the task, as we have done here, will likely help subject-specific ma-

chine learning systems (e.g., to distinguish bankruptcy cases) to be developed for these courts.

CHAPTER 6

Using Machine Learning to Categorise Decisions of the European Court of Human Rights

When courts started publishing judgements, big data analysis (i.e. large-scale statistical analysis of case law and machine learning) became possible within the legal domain. Taking data from the European Court of Human Rights as an example, we investigate how natural language processing tools can be used to analyse the texts of court proceedings, in order to automatically categorise court judgements. With an average accuracy of 75% in categorising the judgements, according to whether or not there was a violation of 9 articles of the European Convention on Human Rights, our (relatively simple) approach highlights the potential of machine learning within the legal domain. We also show, however, that categorising future cases based on past cases negatively impacts performance (with an average accuracy range from 58% to 68%). Furthermore, we demonstrate that a relatively high classification performance (average accuracy of 65%) can be achieved, when categorising judgements based only on the surnames of judges trying cases.

Chapter adapted from:

Medvedeva, M., Vols, M., and Wieling, M. (2020a). Using machine learning to predict decisions of the European Court of Human Rights. *Artificial Intelligence and Law*, 28:237–266.

6.1 Introduction

Research presented in this chapter is aimed at understanding whether or not it is possible to categorise ECtHR judgements. If we can do so adequately, we may subsequently analyse which words made the most impact on this decision, and thus identify what factors may be important for making judicial decisions.

As we discussed in Chapter 4, categorising court decisions is a task that many researchers are trying to undertake. However, as we already mentioned, the majority of judgement categorisation papers seem under the impression that they are able to predict future decisions. Consequently, they are not set up to analyse the results appropriately, even though this is the most essential part of the categorisation task.

This chapter is aimed at providing a simple but strong baseline for categorising ECtHR judgements. Our goal is to build a high-performing system, with which we can analyse the predictors. To do so, we take Aletras et al. (2016) as the base model to improve upon. Their model aimed at predicting (but actually only identified; see Chapter 4) the court's decision by extracting the available textual information from relevant sections of other ECtHR judgements. They derived two types of textual features from the texts, n-gram features (i.e. contiguous word sequences) and word clusters (i.e. abstract semantic topics). Their model achieved an accuracy of 79%, at case outcome level.

Aletras et al. (2016), however, used only a limited number of cases in their work. Further, due to the unavailability of case application

numbers for the cases they used for their classifications, we were unable to reproduce their results. However, when using their methods with the same and larger amount(s) of data, we consistently achieved lower results than were reported in their paper. Therefore, we begin our research using similar methods, and all the available data. We then explore how we can gradually improve on these methods.

The following section is dedicated to describing the data and methodology we used for our experiments. In Section 6.3, we describe three experiments conducted for this study, and we report the results. In Sections 6.4 and 6.5, respectively, we discuss the results and draw conclusions.

6.2 Data and Methodology

6.2.1 Collecting the Data

As we build on the results by Aletras et al. (2016), we will use publicly available data published by the ECtHR. In order to understand the data we are going to be working on, it is important to have some idea about the composition of the court and the structure of its documents.

The European Court of Human Rights is an international court that was established in 1959. It deals with individual and State applications that claim the violation of various rights laid out in the European Convention on Human Rights. Applications are always brought against a State (or multiple States) which has ratified the ECHR, not against individuals.

The number of judges in the court is equal to the number of States Parties to the Convention, which is 47 at the time of writing. The judges are currently elected for nine-year terms, with no possibility of re-election. The cases are tried either in sections containing seven-member Chambers, or in a 17-member Grand Chamber, for which the judge from the State accused of the violation (the 'national judge') is always present.

To warrant a full judgement the cases have to be admissible, some of the admissibility cases (especially when the entire case is found inadmissible) can be published separately from the judgement. Those can be decided by a single judge (who is not the 'national judge'), a Committee or a Chamber. Since '00s the majority admissibility decisions are published as part of the judgement. Most often the case will be judged based on merit by a Chamber within one of the five sections of the court or, in exceptional circumstances, by the Grand Chamber.

The rulings of the court are available online and have a relatively consistent structure. An ECtHR judicial decision contains the following main parts:

- *Introduction*, consisting of the title (e.g., Lawless vs. Ireland), date, Chamber, and section of the court and its constitution (i.e. judges, president, registrar);
- *Procedure*, containing the procedure that took place, from lodging and application until the court judgement;
- *Facts*, consisting of two parts:
  - *Circumstances*, containing relevant background information on the applicant, and the events and circumstances which led them to seek justice due to the alleged violation of their rights in accordance with the ECHR;
  - *Relevant Law*, containing relevant provisions from legal documents other than the ECHR (these are typically domestic laws, as well as European and international treaties);
- *Law*, containing the legal arguments of the court, with each alleged violation discussed separately;
- *Judgement*, containing the decision of the court regarding the alleged violation;

- *Dissenting/Concurring Opinions*, containing judges' additional opinions, explaining why they voted with the majority (concurring opinion), or why they did not agree with the majority (dissenting opinion).

There is often no dissenting or concurring opinions part, but the other parts are typically included, and they can be of varying length and detail.

In order to create a database which we could use for our experiments, we had to automatically collect all the data online. We therefore created a program that automatically downloaded all the English documents from the HUDOC website.[1] Our database[2] contains all texts of admissible cases available via HUDOC, as of September 11, 2017. Cases which were only available in French or another language were excluded. We used a rather crude automatic extraction method, so it is possible that a few cases might be missing from our dataset. However, this should not matter, because our sample size is very large. For reproducibility, all of the documents that we obtained are available online, together with the code we used to process the data.

In this study, our goal was to categorise the judgements, based on whether there were any violations of each separate article of the European Convention on Human Rights. We therefore created separate data collections, with cases that involved specific articles and whether or not the court ruled that there had been a violation. As many of the cases consider multiple violations at once, some of the cases appear in multiple collections. The information about whether or not a case was a violation of a specific article was automatically extracted from the metadata available on the HUDOC website.

From the data (see Table 6.1), we can see that most of the admissible cases considered by the European Court of Human Rights result

[1] `https://hudoc.echr.coe.int/`

[2] `https://www.dropbox.com/s/lxpvvqdwby30157/crystal_ball_data.tar.gz`

Art.	Title	Violation cases	Non-violation cases
2	Right to life	559	161
3	Prohibition of torture	1446	595
4	Prohibition of slavery and forced labour	7	10
5	Right to liberty and security	1511	393
6	Right to a fair trial	4828	736
7	No punishment without law	35	47
8	Right to respect for private and family life	854	358
9	Freedom of thought, conscience and religion	65	31
10	Freedom of expression	394	142
11	Freedom of assembly and association	131	42
12	Right to marry	9	8
13	Right to an effective remedy	1230	170
14	Prohibition of discrimination	195	239
18	Limitation on use of restrictions on rights	7	32

Table 6.1|Initial distribution of admissible cases (in English) obtained from HUDOC on September 11, 2017

in a decision of violation by the state. The specific distribution, however, depends on the article that is being considered.

6.2.2 Balanced Dataset

The machine learning algorithm we use learns the characteristics of cases, based on the text it is presented with as input. The European Court of Human Rights often considers multiple complaints within a

case, even though they might relate to the same article of the ECHR. However, we conduct this experiment as a binary task, only categorising the judgements according to two possible decisions: violation of an article, and non-violation of an article. While some cases may feature both decisions for one article (if there are multiple offences), we focus here on cases in which there is a single ruling (violation, or non-violation). We do this to obtain a clearer picture of what influences the two separate decisions of the court.

While excluding cases which have both decisions makes the task more limited, the goal of our study is to determine the patterns that are specific to violation or no violation of a particular article of the convention. Limiting our task helps us to obtain a clear picture.

Crudely speaking, up to a certain amount, the more data that is available for the training phase, the better the program will perform. However, it is important to control what sort of information the program is provided with during the training phase. If we blindly provide it with all cases, it might only learn the distribution of violation/non-violation cases, rather than more specific characteristics. For example, we might want to train a program that categorises judgements according to whether or not there is a violation of Article 13, by feeding it all 170 non-violation cases, together with all 1,230 violation cases. With such a clear imbalance in the number of cases per type, it is likely that the program will learn that most of the cases feature a violation and then simply will assign the violation label to every new case (the performance will be quite high: 88% correct). In order to avoid this problem, we created a *balanced* dataset which included the same number of violation cases as non-violation cases. We randomly removed the violation cases, so that the distribution of both classes was balanced (i.e. 170 violation cases vs. 170 non-violation cases). The excluded violation cases were subsequently used to test the system.

We decided to withhold 20% of the data, in order to use it in future

research (i.e. as a test set, after several different systems have been developed, one of which is discussed in this chapter). These cases were randomly selected and removed from the dataset. The removed cases are available online.[3]

The results of the present study are evaluated using the violation cases that were not used to train the system. The number of cases can be found in Table 6.2 (column 'Test set'). Article 14 was the only article for which there were more non-violation cases than violation cases. Consequently, the test set used here consists of non-violation cases.

For example, for Article 2 we had 559 cases with violation and 161 with non-violation. Ninety of these cases had both at the same time. After removing those cases, we were left with 469 cases with only violation, and 71 with only non-violation. As we wanted to have the same number of cases with each verdict, we had to reduce the number of cases with violation to 71, leaving us with 142 cases in total, and a test set of 398 violation cases for Article 2. We then removed 20% of the cases (14 violation cases, and 14 non-violation), leaving us with 114 cases for the training phase.

A machine learning algorithm requires a substantial amount of data to be trained with. For this reason, we excluded articles with too few cases. We included only articles having at least 100 cases, but also included Article 11, as an estimate of how well the model performs when only very few cases are available. The final distribution of cases can be seen in Table 6.2.

6.3 Experiments

In this section, we describe the experiments that we conducted in this study. In Experiment 1, we investigate the possibility of using

[3] See `test20` at `https://www.dropbox.com/s/lxpvvqdwby30157/crystal_ball_data.tar.gz`

Article	Violation cases	Non-violation cases	Total	Test set
Article 2	57	57	114	398
Article 3	284	284	568	851
Article 5	150	150	300	1118
Article 6	458	458	916	4092
Article 8	229	229	458	496
Article 10	106	106	212	252
Article 11	32	32	64	89
Article 13	106	106	212	1060
Article 14	144	144	288	44*

Table 6.2 | Final number of cases per ECHR article. The asterisk marks the test set consisting of non-violation cases instead of violation cases

words and phrases extracted from the case text in order to categorise the outcomes of judicial decisions. In Experiment 2, we use the approaches from the first experiment to estimate the potential of categorising future cases. Finally, in Experiment 3, we test whether we can categorise the judgements based solely on objective (although limited) information. Specifically, we evaluate how well we are able to categorise the court's judgements by using only the surnames of the judges involved. In all experiments, we use a (linear) support vector machine (SVM; see Section 3.3.5) as our machine learning system.

During the training phase of the SVM, different weights are assigned to the various bits of information given (i.e. n-grams), and a hyperplane is created which uses support vectors to maximise the distance between the two classes. After training the model, we inspect the weights to see what information had the most impact on the model's decision to determine a certain ruling. The weights represent the coordinates of the data points. The further the data point

is from the hyperplane, the more positive the weight is for the violation class, or the more negative the weight is for the non-violation class. These weights can then be used to determine how important a particular n-gram was to the separation, which is an essential result of a categorisation task (see Chapter 4). The n-grams that were most important will hopefully yield some insight into what influences the court's decision making. The idea behind the approach is that we will be able to determine certain keywords and phrases that are indicative of specific violations (e.g., whether or not a case mentions a minority group or children). Based on previous cases with the same keywords, the machine learning algorithm will then be able to determine the verdict more effectively.

6.3.1 Experiment 1: Textual Analysis

6.3.1.1 Set-up

The data we provided to the machine learning program did not include the entire text of the court decision. Specifically, we removed decisions and dissenting/concurring opinions from the judgement text. We also removed the *Law* part of the judgement, as it includes the judges' arguments and discussions which partially contain their final decisions. See, for instance, the statement (included in the *Law* part) from the Case of Palau-Martinez vs. France (16 December 2003):

> 50. The Court found a violation of Articles 8 and 14, taken together, on account of discrimination suffered by the applicant in the context of interference with the right to respect for their family life.

This sentence makes it clear that, in this case, the court ruled for a violation of Articles 8 and 14. Consequently, if we let our program determine the decision based on this information it will be unfair, as the text already shows the decision ('found a violation'). We

also removed the information at the beginning of the case description, which contains the names of the judges. We do, however, use this data in Experiment 3. The data we used can be grouped into five parts: *Procedure*, *Circumstances*, *Relevant Law*, the latter two together (*Facts*), and all three together (*Procedure + Facts*).

Until now we have ignored one important detail, namely how the text of a case is represented as input for the machine learning program. For this, we need to define the features (i.e. an observable characteristic) of each case. An essential question then becomes how to identify useful features (and their values for each case). While it is possible to use manually created features, such as particular types of issues that were raised in the case, we may also use automatically selected features, such as those which simply contain all separate words or short consecutive sequences of words. The machine learning program will then determine which of these words or word sequences are most characteristic for either a violation or a non-violation. As indicated in Chapter 3, a contiguous sequence of one or more words in a text is formally called a word *n-gram*. In this study, we use n-gram features and the *tf-idf* score (see Section 3.2) as the associated value for each n-gram.

In order to identify which sets of features we should include (e.g., only unigrams, only bigrams, only trigrams, a combination of these, or even longer n-grams), we evaluate all the possible combinations. It is important to realise that longer n-grams are less likely to occur (i.e. it is unlikely that one full sentence occurs exactly the same in multiple case descriptions), and are therefore less useful to include. For this reason, we limit the maximum word sequence (i.e. n-gram length) to four in this study.

However, there are also other choices to make (i.e. parameters to set), such as if all words should be converted to lowercase, or if capitalisation is important. For these parameters we take a similar approach and evaluate all possible combinations. All parameters we

have evaluated are listed in Table 6.3.[4] Because we had to evaluate all possible combinations, there was a total of 4,320 different possibilities. As indicated above, cross-validation is a useful technique to assess (on the basis of training data only) which parameters are best. To limit the computation time, we only used three-fold cross-validation for each article. The program therefore trained 12,960 models. Given that we trained separate models for five parts of the case descriptions (*Facts*, *Circumstances*, etc.), the total number of models was 64,800 for each article, and 583,200 models for all nine articles of the ECHR. Of course we did not run all these programs manually, but instead created a computer program to conduct this so-called grid-search automatically. For each article, the best combination of parameters was used to evaluate the final performance (on the test set). Table 6.4 shows the best settings for each article.[5]

During the grid-search, we identify which parameter setting performs best, by testing each combination of parameters three times (using random splits to determine the data used for cross-validation training and testing) and selecting the setting which achieves the highest average performance. We use this approach to make sure that we did not just get 'lucky', but that the model performs well overall. Of course, it is still possible that the model will perform worse (or better) on the test data.

[4]For a more detailed description of the parameters, see `http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html` and `http://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html`

[5]The choice of all parameters per article can be found online: `https://github.com/masha-medvedeva/ECtHR_crystal_ball`

Name	Values	Description
ngram_range	(1,1), (1,2), (1,3), (1,4), (2,2), (2,3), (2,4), (3,3), (3,4), (4,4)	Length of the n-grams; e.g., (2,4) contains bigrams, trigrams and fourgrams
lowercase	True, False	Lowercase all words (remove capitalization for all words)
min_df	1, 2, 3	Exclude terms that appear in fewer than n documents
use_idf	True, False	Use Inverse Document Frequency weighting
binary	True, False	Set Term Frequency to binary (all non-zero terms are set to 1)
norm	None, 'l1', 'l2'	Norm used to normalise term vectors[6]
stop_words	None, 'english'	Remove most frequent English words from the documents; *None* to keep all words
C	0.1, 1, 5	Penalty parameter for the SVM[7]

Table 6.3 | List of evaluated parameter values

Article	Parts	N-grams	Remove capitalisation	Remove stopwords
Article 2	Procedure + Facts	3-4	✓	✗
Article 3	Facts	1	✓	✗
Article 5	Facts	1	✓	✗
Article 6	Procedure + Facts	2-4	✓	✗
Article 8	Procedure + Facts	3	✓	✗
Article 10	Procedure + Facts	1	✗	✗
Article 11	Procedure	1	✗	✓
Article 13	Facts	1-2	✗	✗
Article 14	Procedure + Facts	1	✓	✓

Table 6.4 | Selected parameters used for the best model

Unigrams achieved the best results for most articles, but longer word sequences were better for some longer articles. As we expected, the *Facts* section of the case was the most informative, and it was selected for eight out of nine articles. For many articles, the *Procedure* section was also informative. This is not surprising, as this section contains important information on the alleged violations. See, for instance, a fragment from the *Procedure* part of the Case of Abubakarova and Midalishova vs. Russia (4 April 2017):

[6] We can use normalization to account for bias towards high frequencies of certain words as well as the length of the texts. For more information on the differences between L1- and L2-norms see `http://blog.christianperone.com/2011/10/machine-learning-text-feature-extraction-tf-idf-part-ii/`, accessed on 04/04/2022.

[7] The C-parameter determines the trade-off between training error and model complexity. If C is too small, it will increase the number of training errors, while a large value for C might lead to a model that cannot generalise and is thus unable to determine the decisions of the cases it has never seen before adequately (Joachims, 2002).

3. The applicants alleged that on 30 September 2002 their husbands had been killed by military servicemen in Chechnya, and that the authorities had failed to investigate the matter effectively.

6.3.1.2 Results

After investigating which combinations of parameters worked best, we used these parameter settings and 10-fold cross-validation to ensure that the model performed well in general, and that it was not overly sensitive to the specific set of cases on which it was trained. When performing 10-fold cross-validation, instead of three-fold cross-validation, there is more data available to use for training in each fold (i.e. 90%, rather than 66.7%). Note that, as we used a balanced dataset during cross-validation, the number of violation cases is equal to the number of non-violation cases. Consequently, if we were to randomly guess the outcome, we would be correct in about 50% of the cases. Percentages substantially higher than 50% indicate that the model is able to use (simplified) textual information present in the case to improve the performance. The first row of Table 6.5 shows the results of the cross-validation procedure.

	Art. 2	**Art. 3**	**Art. 5**	**Art. 6**	**Art. 8**	**Art. 10**	**Art. 11**	**Art. 13**	**Art. 14**	**Avg.**
cross-val	0.73	0.80	0.71	0.80	0.72	0.61	0.83	0.83	0.75	0.75
test	0.82	0.81	0.75	0.75	0.65	0.52	0.66	0.82	0.84	0.74

Table 6.5 | Cross-validation (10-fold) and test results for Experiment 1

In order to evaluate whether the model categorises judgements into both classes well, we use precision, recall, and the f1-score to estimate performance. *Precision* is the percentage of cases for which the assigned label is correct (i.e. violation or non-violation). *Recall* is the percentage of cases with a certain label which are identified correctly. The f1-score can be described as the harmonic mean of preci-

sion and recall.[8] More details about these measures can be found in Chapter 3. Table 6.6 shows the values of these measures, per class, for each model. As we can see from the table, violation and non-violation are predicted very similarly for each article.

Article	Class	Precision	Recall	F1-score
Art. 2	non-violation	0.72	0.68	0.70
Art. 2	violation	0.70	0.74	0.72
Art. 3	non-violation	0.80	0.77	0.79
Art. 3	violation	0.78	0.81	0.80
Art. 5	non-violation	0.77	0.75	0.76
Art. 5	violation	0.76	0.77	0.77
Art. 6	non-violation	0.78	0.87	0.82
Art. 6	violation	0.85	0.76	0.80
Art. 8	non-violation	0.69	0.76	0.72
Art. 8	violation	0.73	0.66	0.69
Art. 10	non-violation	0.63	0.66	0.65
Art. 10	violation	0.64	0.61	0.63
Art. 11	non-violation	0.86	0.78	0.82
Art. 11	violation	0.80	0.88	0.8
Art. 13	non-violation	0.83	0.86	0.85
Art. 13	violation	0.85	0.83	0.84
Art. 14	non-violation	0.77	0.76	0.77
Art. 14	violation	0.77	0.77	0.77

Table 6.6 | Precision, recall and f-score, per class, per article, obtained during 10-fold cross-validation

In Figure 6.1 we visualise the phrases (i.e. trigrams and four-grams) that ranked highest when identifying a case as a violation (blue, on the right) or a non-violation (red, on the left) for Article 2. This figure shows that the Chechen Republic is an important feature in relation to violation cases, while Bosnia and Herzegovina are more highly weighted on the non-violation side.

[8]The exact description of the metric used can be found here: `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html`

We can also observe many relatively 'meaningless' n-grams, such as *in case no* or *the court of*. This can be explained by the simplicity of our model, which does not filter unnecessary information. As we have already mentioned, we only process the text by lowering the case of the words, or filtering out more common words (such as *the*, *a*, *these*, *on*) for some of the articles.

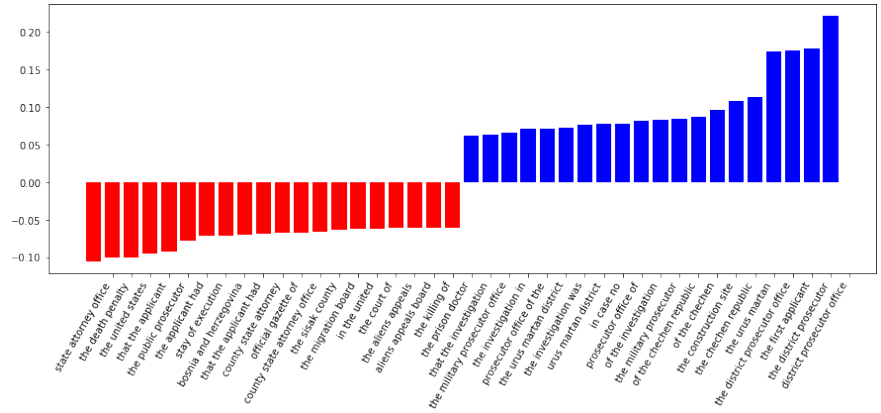


Figure 6.1 | Coefficients (weights) assigned to different n-grams for categorising violations of Article 2 of ECHR. The top-20 violation predictors (blue on the right) and the top-20 non-violation predictions (red on the left) are shown.

After tuning and evaluating the results using cross-validation, we tested the system using the violation cases (for Article 14: non-violation cases) that had not already been used to determine the best system parameters. These are the 'extra' cases. The results can be found in Table 6.5 (the second line). For several articles (6, 8, 10, 11, and 13) the performance on the test set was worse, but for a few articles it performed substantially better (e.g., Articles 2, 5, and 14). Discrepancies in the results might be explained by the fact that sometimes the model learns to categorise non-violation cases better than violation cases. By testing the system on cases that only contain vio-

lations, performance may appear to be worse. The opposite happens when the model learns to categorise violations better. In that case, the results for a violation-only test set appear to be better. It should be noted that the test set for Article 14 contains non-violations only, and an increase in performance here indicates that the model has probably learned to categorise non-violations better. Nevertheless, the test set results are similar to the cross-validation results, suggesting that the model is performing well with only very simple textual features.

6.3.1.3 Discussion

The results, with an average performance of 0.75, show substantial variability across different articles. It is likely that the differences are caused, to a large extent, by differences in the amount of training data. The lower the amount of training data, the less the model is able to learn from the data.

To analyse how well the model performs, it is useful to investigate the *confusion matrix*. This matrix shows how cases were classified correctly and incorrectly. For example, Table 6.7 shows the confusion matrix for Article 6. There were 916 cases in the training set for Article 6, half of which (458 cases) had a violation verdict, and half of which had a non-violation verdict. The table also shows that, out of 458 cases with a non-violation, 397 were identified correctly, and 61 were identified as cases with a violation. Additionally, 346 cases with a violation were classified correctly, and 112 cases were identified as a non-violation. Given that the amount of non-violation and violation cases is equal, it is clear from this matrix that the system for Article 6 is better at categorising non-violation cases than violation cases, as we can also see in Table 6.6.

Cases themselves may also influence the results. If there are many similar cases with similar decisions, it is easier to categorise the judgement of another similar case. Whenever there are several very diverse issues grouped under a single article of the ECHR, the perfor-

		Predicted outcome	
		Non-violation	Violation
Actual outcome	Non-violation	397	61
	Violation	112	346

Table 6.7 | Confusion matrix for 10-fold cross-validation for Article 6

mance is expected to be lower. This is the likely cause of the relatively low performance of Article 8 (*right to respect for private and family life*), which covers a large range of cases. The same can be said for Article 10 (*right to freedom of expression*), as the platforms for expression are growing in variety, especially online.

To investigate the categorisation errors made by each system, we focus on Article 13 (having the highest accuracy score). Our approach was to first list the n-grams with top-100 tf-idf scores for the incorrectly classified documents (for a violation classified as a non-violation, and vice versa). We then included only the n-grams occurring in at least three incorrectly classified documents, for each of the two types. We did the same for the correctly identified documents (also two types: violation correctly classified, and non-violation correctly classified), then looked at overlapping n-grams in the four lists.

While the lists contained very different words and phrases, we were able to observe some general tendencies. For instance, phrases related to prison (e.g., 'the prison', 'prisoner', etc.) generally appeared in cases with no violation. Consequently, violation cases which do contain these words are likely to be incorrectly classified (as non-violation). Similarly, words related to prosecutors (e.g., 'public prosecutor', 'military prosecutor', 'the prosecutor') are more often found within cases with a violation. Therefore, non-violation cases containing such phrases may be mislabelled. Table 6.8 shows a subjective selection of words which behave similarly.

	predicted: non-violation	predicted: violation
actual: non-violation	applicant **police** **security** commission imprisonment	**prosecutor** **criminal** **ukraine**
actual: violation	**police** **security** prison	**prosecutor** **criminal** military **ukraine** russian

Table 6.8 | Comparison of selected n-grams within top-100 tf-idf scores for correctly predicted and mislabelled documents for Article 13 (bold face: overlapping across true categories)

It should be noted that the error analysis remains rather speculative. It is impossible to pinpoint exactly what makes the largest impact on the categorisation, because the decision for one document is based on all the n-grams in that document.

In the future, more sophisticated methods (including semantic analysis) should be used, in order not to simply categorise decisions and potential basic determinants, but to identify the factors behind the choices made by the machine learning algorithm.

6.3.2 Experiment 2: Categorising the Future

6.3.2.1 Set-up

The test set for the first experiment was randomly sampled, without considering the year of the cases. In this section, we will assess how

well we are able to categorise future cases by dividing the cases used for training and testing on the basis of the year of each case. Such an approach has two advantages. The first is that it results in a more realistic setting, as categorising the outcome of a case for which the actual outcome is already known has no practical application (see Chapter 4). The second advantage is that times are changing, and this affects the law as well. For example, consider Article 8 of the ECHR. Its role is to protect citizens' private lives, including (for example) their correspondence. However, correspondence from 40 years ago looks very different to today's correspondence. This also suggests that using cases from the past to predict the outcome of cases in the future might reflect a lower, but more realistic, performance than the results reported in Table 6.5. For this reason, we set up an additional experiment to check whether this is indeed the case, and how sensitive our system is to this change. Due to the more specific requirements for data for this experiment, we only considered the datasets with the largest amounts of cases (i.e. Articles 3, 6, and 8), and we divided them into smaller groups on the basis of the year of the cases. Specifically, we evaluated the performance on cases from either 2014 to 2015 or 2016 to 2017, while using cases up to 2013 for training. Because violation and non-violation cases were not evenly distributed between the periods, we had to balance them again. Where necessary, we used additional cases from the violations test set (used in the previous experiment) to add more violation cases to particular periods. The final distribution of the cases over these periods can be found in Table 6.9.

	Art. 3	**Art. 6**	**Art. 8**
Training dataset up to and including 2013	356	746	350
Test set: 2014 - 2015	72	80	52
Test set: 2016 - 2017	140	90	56

Table 6.9 | Number of cases

We performed the same grid search (using three-fold cross validation) to determine the parameters for tf-idf and the SVM on the basis of the new training data as was performed during the first experiment. We did not opt to use the same parameters, as these were tailored to categorising mixed-year cases. Consequently, we performed the parameter tuning only using the data up to and including 2013.

The two periods are set up in such a way that we may evaluate their performance in categorising judgements which immediately follow those we train on, versus those which follow later on. In the latter case, there is a gap in time between the training period and testing period. Additionally, we conducted an experiment with a 10-year gap between training and testing. In this case, we trained the model on cases up to 2005 and evaluated its performance using the 2016 to 2017 test set.

In order to be able to interpret the results better, we conducted one additional experiment. For Experiment 1* we reduced the training data from Experiment 1 to a random sample of a size equal to the amount of cases available for training in Experiment 2 (i.e. 356 cases for Article 3, 746 cases for Article 6, etc.), but with all the time periods mixed together. We compared cross-validation results on this dataset (Experiment 1*) to the results from the 2014 to 2015 and 2016 to 2017 periods in this experiment. The reason we conducted this additional experiment is that it allows us to control for the size of the training dataset.

6.3.2.2 Results

As Table 6.10 shows, training on one period and categorising another is harder than when categorising a random selection of cases (as in Experiment 1). We can also observe that the amount of training data does not influence the results substantially. Experiment 1 resulted in an average accuracy of 0.77 for the chosen articles, while Experiment 1* had an accuracy which was almost as high. However, testing

on separate periods resulted in a much lower accuracy. This suggests that categorising future judgements is indeed a harder task, and it gets harder if the gap between the training and testing data increases.

Period	Art. 3	Art. 6	Art. 8	Avg.
2014-2015	0.72	0.64	0.69	0.68
2016-2017	0.70	0.63	0.64	0.66
2016-2017 (10 year gap)	0.69	0.59	0.46	0.58
Experiment 1*	0.78	0.78	0.72	0.76
Experiment 1	0.80	0.80	0.72	0.77

Table 6.10 | Results for Experiment 2

6.3.2.3 Discussion

The results of Experiment 2 suggest that we must take the changing times into account if we want to categorise future cases. Therefore, while we can categorise the judgements of the past year relatively well, performance drops when there is a larger gap between categorisation and the period on which the model was trained and tested. This shows that a continuous integration of published judgements in the system is necessary, in order to keep up with the changing legal world, and to maintain an adequate performance.

While there is a substantial drop in performance on the basis of the 10-year gap, this is also likely to be caused by a large reduction in training data. Due to the limit on the period, the number of cases used as training data was reduced to 112 (instead of 356) for Article 3, 354 (instead of 754) for Article 6, and 144 (instead of 350) for Article 8. Nevertheless, the large drop in performance for Article 8 suggests that the issues covered in that article have evolved more over the past decade than those of the other two articles.

Importantly, while these results show that categorising judgements for future cases is possible, the performance is lower than it is when simply categorising decisions for random cases (as in Experiment 1, and in the approach employed by Aletras et al., 2016 and Şulea et al., 2017).

6.3.3 Experiment 3: Judges

6.3.3.1 Set-up

We also wanted to experiment with a very simple model. Consequently, we used only the names of the judges that constitute a Chamber, including the president, but not including the Section Registrar and the Vice-Section Registrar (when present), as they do not decide cases. The surnames were extracted from the list provided by the ECtHR on its website.[9] However, ad hoc judges were not extracted, unless they were on the same list (e.g., from a different section), due to the unavailability of a full list of ad hoc judges for the whole period of the court's existence. In our extraction efforts, we did not account for any misspellings in the case documents. Therefore, only correctly spelled surnames were extracted.

We set-up our categorisation model in line with the previous two experiments. We used the surnames of the judges as input for the model. In total, there were 185 judges representing 47 states, at different times. The number of judges for a state largely depends on when that particular state ratified the ECHR. Given that nine-year terms for judges were established recently, some judges might have been part of the court for a very long time. Some states, such as Serbia, Andorra, and Azerbaijan, have only had two judges, while Luxembourg has had seven, and the United Kingdom has had eight. Only one judge represents each state, at any one time.

[9] `https://www.echr.coe.int/Pages/home.aspx?p=court/judges`, , accessed on 04/04/2022

We retained the same set of documents in the dataset as in Experiment 1, but provided the model with only the surnames of the judges. However, for this experiment we did not use tf-idf weighing. Instead, we represented features as the judge being either present at the bench, or not present at the bench.

6.3.3.2 Results

	Art. 2	**Art. 3**	**Art. 5**	**Art. 6**	**Art. 8**	**Art. 10**	**Art. 11**	**Art. 13**	**Art. 14**	**Avg.**
cross-val	0.61	0.67	0.67	0.68	0.59	0.56	0.67	0.73	0.66	0.65
test	0.62	0.64	0.66	0.67	0.55	0.65	0.60	0.79	0.73	0.66

Table 6.11|10-fold cross-validation and test set results for Experiment 3

Using the same approach as illustrated in Section 6.3.1, we obtained the results shown in Table 6.11. In addition, Figures 6.2 and 6.3 show the weights, determined by the machine learning program, for the top-20 predictors (i.e. the names of the judges) for categorising the violation outcome versus the non-violation outcome.

6.3.3.3 Discussion

While one may not know which judges will be assessing a particular case, the results show that the decision is influenced, to a large extent, by the judges in the Chamber.

In this experiment we did not consider how each judge voted, but what the final decision was in each case. Consequently, it is important to note that, while some judges may be strongly associated with cases which were judged to be violations (or non-violations), this does not mean that they will always rule in favour of a violation, when it comes to a particular article of the ECHR. It simply means that the judge sits more often in a Chamber which votes for a violation, irrespective of the judge's own opinion.

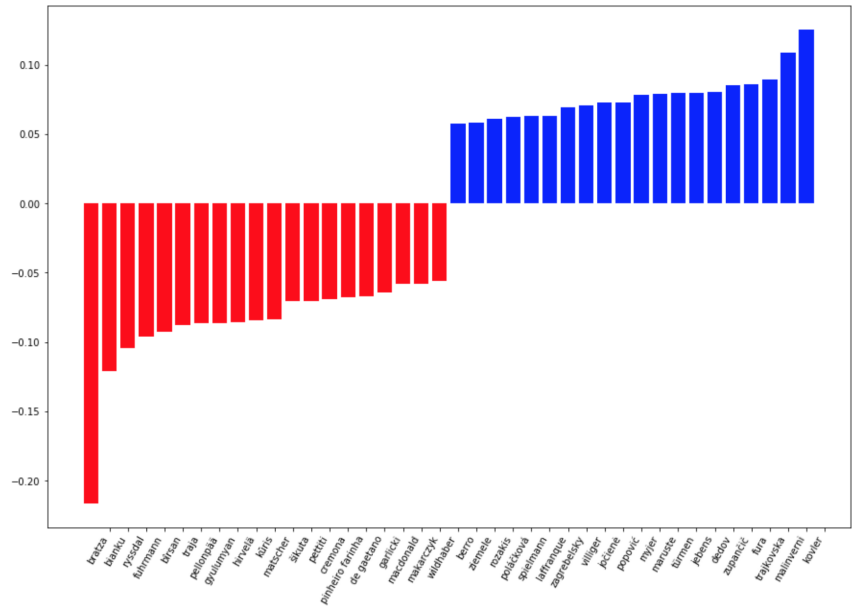


Figure 6.2 | Coefficients (weights) assigned to the names of the judges, to categorise judgements according to whether or not there was a violation of Article 13 of the ECHR. The top-20 violation predictors (blue, on the right) and the top 20 non-violation predictors (red, on the left) are shown.

Importantly, judges have different weights, depending on the article that we are considering. For example, Polish judge Lech Garlicki is frequently associated with a non-violation of Article 13, but for Article 14 he is more often often associated with a violation. This is consistent with the numbers we have in our training data. Garlicki was in a Chamber that voted for a non-violation of Article 14 36 times, and for a violation 34 times. On the other hand, Garlicki was in a Chamber that voted for a violation of Article 13 six times, and for a non-violation 38 times.

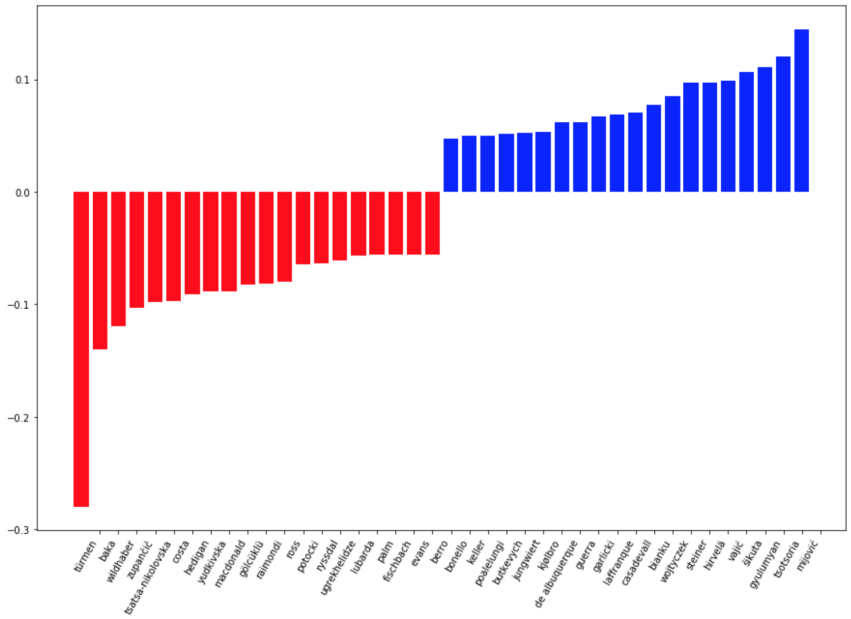


Figure 6.3 | Coefficients (weights) assigned the names of the judges to categorise judgements according to whether or not there was a violation of Article 14 of the ECHR. The top-20 violation predictors (blue, on the right) and the top-20 non-violation predictors (red, on the left) are shown.

It is interesting to see the results for the test set in this experiment. While the average results are very similar to the cross-validation results, the scores are very high for particular articles. For instance, when categorising judgements according to a potential violation of Article 13 (*right to an effective remedy*), the names of the judges are enough to obtain a correct outcome for 79% of the violation cases. Similarly, the number is also very high for Article 14 (prohibition of discrimination): 73%.

While the average results are lower than using the n-grams, it is clear that the identity of the judges is still a useful predictor, given that the performance is higher than the (random guess) performance of 50%.

6.4 Discussion

In this chapter, we have shown that there is potential in treating case law as quantitative data for the categorisation of judgements according to the outcomes of cases. With respect to Aletras et al. (2016), we increased the amount of articles, as well as the amount of cases we used for each article. We also made different decisions regarding which parts of the case should be used for by the machine learning algorithm. By excluding the *Law* part of the cases (which Aletras et al. (2016) did not), we reduced the potential bias of the model when it was given access to court discussions.

We achieved slightly lower scores (0.77 vs. 0.79) for the three articles analysed in Aletras et al. (2016). However, we believe that our approach is more representative, as we make use of all available data. After balancing the dataset, we have 1,942 cases for the three articles, while Aletras et al. (2016) included only 584. Furthermore, as they use the *Law* part of the cases, which sometimes also explicitly mentions the verdict, their results are likely biased. Thus, we have created a new, reproducible baseline that we (and others) may improve upon in future.

In this study, we have chosen to build separate models for different articles of the ECHR. When performing the parameter search, it was clear that different parameters work better for different articles. In all three experiments (using n-grams, categorising future cases, or using only the judges' names) we also observed varying performance for the different articles.

We used only balanced datasets to categorise the judgements, but

it is still important to remember that the court rules in favour of a violation much more often than it rules against it. This can be partly explained by the filtering out of non-violation cases during the admissibility stage of a ruling. Many cases with non-violations never make it to the merit stage. Therefore, if we were to teach the model to categorise violation cases better (e.g., when in doubt, categorise the case as a violation, or give violation features more weight), the performance would increase. The models we introduced here do not take this distribution into account, hence it is not fully representative to categorisation accuracy for all available cases. However, our approach does allow us to more clearly identify which features are most important for the system, and it therefore lets us make more informed decisions about adapting the model in future. Moreover, it would be interesting to experiment with various oversampling techniques (i.e. artificially generating more cases with a non-violation verdict), as well as targeted undersampling (i.e. removing only specific cases with violation, instead of random sampling) to create a better, more representative, training set.

It is important to note that, while we are trying to develop a system that can categorise judicial decisions automatically, we have no intention of creating a system that would replace judges. Rather, in this work we assess to what extent judges' decisions are predictable (i.e. transparent).

In this work, we assessed how well a very simple model is able to determine court judgements. Our method therefore may function as a baseline for future improvements. In future work we are hoping to be able to categorise court judgements more effectively, by including the use of more advanced machine learning techniques, as well as introducing more detailed linguistic information (such as semantics).

In addition to increasing the amount of information the model is provided with, we would like to take into account the context in which the words occur. For instance, an approach using so-called

word embeddings (see Mikolov et al., 2013) would allow us to have more abstract representations of words and sentences, instead of the words themselves. Due to our desire to create models which are intuitive and can be explained, the use of neural network approaches is less suitable, as these are often considered to be black boxes. However, further experiments must evaluate if it is possible to use neural network approaches for some parts of the data processing, while retaining the ability to analyse the results of the system.

Since the first version of the work described in this chapter (Medvedeva et al., 2018) was published, many have tried improving the scores for categorising decisions of the ECtHR. As we discussed in Chapter 4, many have also used systems that are not appropriate for a categorisation task, due to the use of unexplainable neural models. However, there are some notable exceptions to this. Specifically, Chalkidis et al. (2019) suggests using hierarchical attention networks that allow for the analysis of what the model paid attention to during the categorisation process, whereas Chalkidis et al. (2021) attempt to extract the rationale behind judicial decisions. It is important to note that even these steps are likely not sufficient by themselves, see, for instance, Branting et al. (2021), where the authors experimented with providing attention-based highlights in World Intellectual Property Organization cases to MITRE employees (with and without legal experience) as decision support, and found no benefit.

6.5 Conclusion

In this chapter, we conducted several experiments that involved analysing the language of European Court of Human Rights judgements, in order to categorise them according to whether or not there had been a violation of a person's rights. Our results showed that, using relatively simple and automatically obtainable information, our models are able to categorise decisions correctly in about 75% of

cases, which is much higher than chance (50%). We also discussed the possibility of analysing weights, assigned to different phrases by the SVM machine learning algorithm, and how the weights can be used to identify patterns within the text of the proceedings. Further research needs to assess how these systems can be improved by using more sophisticated legal and linguistic analysis.

CHAPTER 7

Automatic Judgement Forecasting for Pending Applications to the European Court of Human Rights

This chapter is dedicated to forecasting future decisions on pending applications to the European Court of Human Rights. To address this task, we released an initial benchmark dataset, consisting of documents from the European Court of Human Rights. The dataset included raw data, as well as pre-processed text from final judgements, admissibility decisions, and communicated cases. The latter are published by the court for pending applications (generally) many years before the case is judged, allowing judgements for pending cases to be forecasted. Here, we establish a baseline for this task, and show that it is much harder than simply categorising judgements.

Chapter adapted from:

Medvedeva, M., Üstun, A., Xu, X., Vols, M., and Wieling, M. (2021b). Automatic judgement forecasting for pending applications of the European Court of Human Rights. In *Proceedings of the Fifth Workshop on Automated Semantic Analysis of Information in Legal Text (ASAIL 2021)*.

7.1 Introduction

Digital access to case law (i.e. court judgements) provides us with a unique opportunity to process legal data automatically, and on a large scale, using natural language processing techniques. It is therefore not surprising that the use of machine learning for judgement categorisation has seen a substantial increase in recent years (see Chapter 4). If we rely on the presumption that legal systems and legal decision-making are consistent and predictable, we should be able to ultimately create a system that can automatically predict judicial decisions correctly. Consequently, such a system could also be used to identify patterns which might be less consistent, and could perhaps reveal biases in the legal system and judicial decision-making.

Much work has already been done on categorising the outcomes of final judgements (Aletras et al., 2016; Şulea et al., 2017; Kaufman et al., 2017; O'Sullivan and Beel, 2019; Chalkidis et al., 2019; Kaur and Bozic, 2019; Condevaux, 2020; Salaün et al., 2020; Medvedeva et al., 2020a; Shaikh et al., 2020). Categorisation of final judgements is (in principle) a useful task, as it may be used to identify important factors and court arguments, and thereby may provide insight into the process of decision-making. Some previous research even suggests that, one day, such categorisation systems will be able to provide legal assistance (Şulea et al., 2017) and promote accessibility to justice (Chalkidis et al., 2019), while others suggest that courts, such as the European Court of Human Rights (ECtHR), may eventually use it to prioritise violation cases (Aletras et al., 2016; O'Sullivan and Beel, 2019). Additionally, it has been argued that these systems will eventually be able reduce the human error of judges (Kaur and Bozic, 2019). While each of these suggestions can be scrutinised from a legal perspective, it is also clear that there are a large number of potential applications for a successful categorisation system.

While many of the currently proposed systems show promising re-

sults, with a categorisation performance of about 80% accuracy, this is an overly optimistic view of their performance. One reason for this is that categorisation performance is generally evaluated by predicting the outcome for a random subset of cases which were already known, but not considered, when creating the model. While this may seem fair, an arguably more interesting task is to predict future judgements.[1]

Importantly, however, all of the aforementioned studies claim to 'predict judicial decisions', which suggests that these systems are able to predict (future) rulings on the basis of available information. Unfortunately, classifying future judgements causes performance to suffer (see Chapter 6). This lower performance may be caused by, for example, changes in the interpretation of the law, or new social phenomena and developments due to changing societies. In addition, almost all categorisation systems rely on data about the case, which is made available when the outcome of the case is known. Having knowledge about the outcome of a case might influence how the facts of the case are described (e.g., facts irrelevant to the outcome may be removed, or facts identified after an investigation and relevant to the outcome may be highlighted), compared to a situation in which the outcome would not have been known. This would mean that systems using information composed when the outcome was not known may be disadvantaged, compared to systems using information composed when the outcome was known. One goal of this chapter is to evaluate whether this is indeed the case.

A further goal of this chapter is to demonstrate how large the distinction is between the two tasks of *forecasting* judgements and *categorising* judgements proposed in Chapter 4. The former requires tex-

[1] It is important to note that predicting court judgements is a very different task from actual decision-making. The machine learning systems that are the focus of this study make pattern-based guesses, on the basis of (sequences of) words in the text of a case. We discuss the ethical considerations for making this distinction in Chapter 9.

tual data describing the (facts of the) case, which were created *before* the decision was reached, so that the input of the forecasting system is not influenced by the outcome. For the latter, textual data about the (facts of the) case need to have been created *after* the decision was reached.

Forecasting thus requires data related to a judgement that were published before the actual judgement was delivered. While the courts publish more and more case law every day (Marković and Gostojić, 2018), very little access is provided to documents that were available before the judgements were made. Forecasting future judgements is therefore an impossible task for many datasets available online. For this reason, the large majority of machine learning systems for legal data were built to provide categorisation of court judgements, as opposed to forecasting judgements.

In this study, however, we concentrate on the ECtHR, as it publishes all of its final judgements online, together with many supplementary documents, including admissibility cases, press releases, summaries of cases, etc. Several of these documents were created before the decision was reached, and, therefore, this specific dataset enables both categorisation and forecasting of the judgements.[2]

In addition to evaluating whether forecasting is indeed a harder task than categorisation (by using the same cases to evaluate both sets of algorithms), we aim to compare the relative performance of classification algorithms previously used in this field for both tasks. For the forecasting task, specifically, we use the information published in the *communicated cases* (see Section 7.2.2). We do not introduce any new algorithms, as the purpose of this study is to determine the performance difference between the two types of tasks.

[2]To enable reproducibility, we provide our dataset containing pending ECtHR applications, as well as the admissibility decisions and final judgements of the court, which can be used for a variety of tasks: `https://drive.google.com/drive/folders/1ltIpHlcqcRlT_JDebHsyLgvgoa4Vbxo8?usp=sharing`

To our knowledge, only one study has tried to show that using documents from the early stages of the legal process may not always be as useful and *predictive* as final judgements. Specifically, Branting et al. (2020) conducted experiments using statements from attorney misconduct complaints submitted to the Bar Association in the USA. The researchers set up a task of predicting whether the case would be investigated or closed. Using six different machine learning systems, the authors showed that the text of the complaints themselves had very low predictive accuracy (maximum weighted f1-score: 0.52), and that adding more metadata (i.e. extra information filled in during the complaint, attorney history, sentiment score, etc.) was not very beneficial (maximum weighted f1-score: 0.55). Only data from later stages in the process (in particular, allegation codes assigned by the intake staff) improved the results (maximum weighted f1-score: 0.70). Nevertheless, these scores are still substantially lower than those reported by many studies classifying final decisions (see above). While Branting et al. (2020) also deal with legal documents, the documents are not judicial decisions, but rather disciplinary proceedings conducted by the Bar Association. They are therefore not directly comparable to the experiments conducted on court judgements.

As Chapter 4 showed, there are currently only very few studies that focus on forecasting judgements, and most show a lower performance level than studies on judgement categorisation. In this study, we aim to directly compare the performance of both tasks explicitly.

The following section is dedicated to describing the data we have used for our experiments, and the larger dataset we are releasing alongside this paper. In Section 7.3, we discuss various methods that can be used to forecast decisions, including their strengths and limitations. In Section 7.4, we report the results of the experiments we have conducted for this study. In Section 7.5, we discuss the results and make suggestions regarding future work. Finally, in Sections 7.6 and 7.7, respectively, we note the ethical issues to take into

consideration, and we draw conclusions.

7.2 Data

7.2.1 The Court

Similarly to the categorisation task we presented in Chapter 6, we used the case law of the European Court of Human Rights to conduct our experiments. In 2020 the court has made decisions on 39,190 applications. A total of 37,289 applications were dismissed, based on admissibility criteria, while the rest were decided by a Chamber or Grand Chamber (1,901 applications, resulting in fewer judgements as similar applications are often merged). The majority of the documents produced by the court during the process were published online by the court.[3] From 871 published judgements 762 found to have at least one violation of human rights.

7.2.2 Communicated Cases

In order to describe the data we use for our forecasting experiment system, it is important to clarify what the court application process entails.

A resident of any State that has ratified the ECHR can claim a potential violation within a certain time frame. The application is submitted via mail. On arrival, the application is registered by the court, and sent to the legal division dealing with cases for that particular State, as it will be familiar with the State's national legislation. Subsequently, the case is allocated to one of the court's judicial formations.

Most of the cases are found to be inadmissible without meriting an investigation, because they do not meet the formal admissibility criteria. For example, often the application is dismissed because the applicant did not file the complaint within the required time frame. A

[3] `https://hudoc.echr.coe.int`

decision regarding these cases is normally rendered by a single judge. If the application is not dismissed directly, the decision on admissibility is taken either by a Committee of three judges (if the court has dealt with a number of similar cases before) or by the Chamber of seven judges. In some cases, admissibility decisions may even be made by the Grand Chamber (consisting of seventeen judges). The latter usually concern either interpretation of the ECHR itself, or the risk of inconsistency with the court's previous judgements.

When an application is judged to be admissible based on formal parameters, the Chamber will examine its merits. Before doing so, the court will *communicate* the application to the government potentially violating the rights of the applicant (Rule 60 of the Court: Claims for Just Satisfaction). This is only carried out for some applications (approximately 15-20%). Such *communicated cases* contain a summary of the facts of the case, as well as questions to the government pertaining to the applicant's complaint. This document allows the government concerned to submit its observations on the disputed matter. These documents are often communicated years before the case is judged, which provides a unique opportunity to use them to predict future case judgements. Moreover, the questions posed to the State often reflect the court's legal characterisation of the complaint. See, for instance, a question from the case of Arki against Hungary (application no. 10755/14, communicated on June 6, 2014):

> 1. Have the applicants been subjected to inhuman or degrading treatment on account of their cramped prison conditions, in breach of Article 3 of the Convention?

As a consequence, these documents can potentially be used to identify the facts, or even (parts of) arguments, related to certain judgements, before those judgements are made.

Cases concerning repetitive issues do not merit a communicated case, and not every communicated case corresponds directly to a spe-

cific judgement. Multiple applications concerning the same events can be merged into a single case during the communication stage, but may be separated during final decision-making. Similarly, multiple applications can be communicated separately, but eventually judged together. Each year, thousands of applications are communicated (i.e. 6,442 in 2019, and 7,681 in 2020). Only communicated cases from the year 2000 and later are available online. The court decides on the order in which the cases are dealt with, based on the importance and urgency of the issues raised (Rule 41 of the Court: Order of Dealing with Cases).[4] Therefore, the cases being judged may be mixed, and not always be judged in the same chronological order as they were submitted.

For the machine learning systems created in our study, we will only use communicated documents that have judgements, or have been found inadmissible based on their merit for training and testing.

7.2.3 Data Collection

We collected the data for this study in the following way. We scraped the ECtHR's 'HUDOC' website[5] and downloaded all the communicated cases. We did the same for the judgements and admissibility decision documents, such as the admissibility cases from the Chamber and Committee. We filtered the cases on the website to only download English versions of the documents. As the filter did not always work adequately, we also filtered using Google's language detection (`langdetect`) library.[6] In addition, we extracted all the available metadata, such as the application number, state, importance level, etc. We used the application number of each communicated case to link the associated documents to corresponding admissibility decisions and

[4] `https://www.echr.coe.int/Documents/Rules_Court_ENG.pdf`, accessed on 04/04/2022

[5] `https://hudoc.echr.coe.int/`

[6] `https://pypi.org/project/langdetect/`

judgements. We then extracted the conclusions of court proceedings ('violation' or 'no violation'), as well as the facts of cases, from the judgement text. We used these facts in a categorisation model, so we could compare its performance to that of a forecasting model using data from the communicated cases.

While the facts in communicated cases are a summary of events as described by the applicant, the facts that end up in the final judgement are compiled after the investigation, and therefore also include the State's arguments. We only use the facts of a case from the final judgement, since these are most comparable to the communicated cases. Specifically, these have also been argued to be potentially available before an outcome was reached (Medvedeva et al., 2020a), and they do not contain references to the outcome (Medvedeva et al., 2020a; Chalkidis et al., 2019). Only extracting the facts also mirrors the set up in Chalkidis et al. (2019), which we follow.

To enable a fair comparison, the cases (but not the extracted information about them) used for training and testing are identical for both models. We assume that cases that were found to be *inadmissible based on merit* are similar to cases that were judged as having no violation. From a legal point of view, these cases can be characterised simply as clearer 'non-violation' cases. The court has made judgements on similar applications many times before, hence, these do not merit a full judgement. For cases that went though to the final judgement stage, we assigned the 'violation' label to all those judged to show a violation of at least one article of the ECHR.

As we mentioned before, individual communicated cases do not always correspond directly to unique cases which received a judgement or admissibility decision, as communicated cases can be either split or merged during the process. For the split cases, the assigned label depended on whether any of the corresponding judgements had a violation of at least one article ('violation' label), or not ('non-violation' label, i.e. none of the split cases exhibits a violation of any

article). To ensure that the set of cases considered for the categorisation task and the forecasting task was identical, we randomly selected a single judged case (from the associated split cases), where the assigned label matched that assigned to the communicated case. For judgements associated with multiple and merged communicated cases, we randomly chose one of the communicated cases and removed the rest. Finally, duplicate cases and judged cases which did not have (correctly formatted) facts were excluded from the dataset used for both tasks. In this way, we ensured that the set of cases considered for the categorisation task and the forecasting task was identical.

Subsequently, we split the data into training and test sets according to the years when the judgements were made, resulting in on average a 77% - 23% split. We trained each system three times, with different setups (with a decreasing amount of training data), to assess the robustness of the results. Setup 1 concerns model training with cases that received a judgement in the years 2000 to 2019, whereas model testing was conducted with cases that received a judgement in the year 2020. Setup 2 uses data from 2000 to 2018 for training, and 2019 data for testing. Setup 3 uses data from 2000 to 2017 for training, and 2018 data for testing. Each setup is used once to forecast judgements using data from the communicated cases, and once to classify judgements using data from facts extracted from the final judgement. As the number of violation cases exceeded the number of non-violation cases in every setup, we balanced the training set in each setup by removing older violation cases, until the same number of documents was present for each label. Table 7.1 shows the number of documents available for training and testing in each setup.

The data used for the two different tasks differs somewhat. For the communicated cases, we used all the available data (i.e. the facts and questions as they were presented in the text), whereas for the judgements we only used data from the facts section. In general,

the average number of words associated with the facts extracted from each judgement are only slightly higher (i.e. 2000 words) than the number of words in the associated communicated case (i.e. 1800 words).

	Setup 1: 2020	**Setup 2: 2019**	**Setup 3: 2018**
Training data (balanced)	2264	1806	1386
Testing data (no violation)	167	229	210
Testing data (violation)	342	311	309

Table 7.1 | Distribution of training and testing data for different setups

7.2.4 Published Dataset

In addition to the data used in this study, we extracted data for a large set of additional cases, which were not taken into account in our analysis in this chapter. This dataset is released together with this dissertation.[7] Specifically, this dataset contains all of the communicated cases, admissibility cases, and final judgements of the court which were published between 1960 and 2020. We provide the raw text, the metadata (e.g., date, court-assigned importance, parties, and section), the pre-processed text of communicated cases (split into facts and questions), the admissibility decisions (extracted facts) and the final judgements (split into sections: Procedure, Facts, Relevant domestic law, Law - including the arguments of the court, Outcome, and Dissenting opinions), in order to facilitate further research in ECtHR judgement forecasting and categorisation. In addition, the case numbers are linked throughout each stage of the court proceedings (where applicable). This dataset is suitable for a number of categorisation tasks in legal analysis, including judgement categorisation

[7] `https://drive.google.com/drive/folders/1ltIpHlcqcRlT_JDebHsyLgvgoa4Vbxo8?usp=sharing`

based on facts (using the Facts, and possibly the Procedure, section) and/or arguments (using the Law section).

7.3 Methodology

The approach most relevant to our study is that of Chalkidis et al. (2019). Specifically, in one of their tasks they focused on classifying court judgements depending on whether or not at least one article of the ECHR had been violated.[8] In addition, they experimented with using anonymised vs. non-anonymised data. While we perform the same task as Chalkidis et al. (2019), enabling us to benefit from more data than if we predict (non-)violation per article separately, we use non-anonymised data only. For an anonymised setup, Chalkidis et al. (2019) removed *named entities* (such as names or locations) from the text, to make sure that the model was not biased towards demographic information. While removing this potential bias is understandable when building a decision-making system, forecasting or classifying judgements is different. Specifically, given that locations may offer relevant information about the case (i.e. some countries are notorious violators of specific rights), models used for forecasting or categorisation benefit from including this information (which is also known to judges).

In our study, we implemented three systems used by Chalkidis et al. (2019), and compared their performance in categorisation and forecasting tasks. Specifically, we included the SVM model, the Hierarchical-BERT (H-BERT) model, and the LEGAL-BERT model (see below for more details). All the models were re-created on the basis of

[8]The purpose of Chalkidis et al. (2019) second task was to identify all of the violated articles for a single court document (i.e. multi-label classification). However, as the articles involved are known as soon as the application is submitted, the practical use of predicting the list of articles potentially violated is not clear. A realistic scenario for the ECtHR would only involve deciding whether or not a *given* article was violated.

the description provided by Chalkidis et al. (2019) and Chalkidis et al. (2020). As not all of the settings and (hyper)parameters were specified in their paper, our reproduction of their models may be slightly different. However, we believe these differences to be minor. Our goal is to see how some of the state-of-the-art models, which have been shown to perform very well when applied to final ECtHR judgements, perform when they are only provided with (communicated) data from applicants to the ECtHR (i.e. victims of an alleged human rights violation).

Our SVM classifier is a linear support vector classification model, including 1 to 5 n-grams. For a detailed explanation of text classification using machine learning (including linear SVM), see Chapter 3.

BERT (or, Bidirectional Encoder Representations from Transformers; Devlin et al., 2019) is a popular pre-trained transformer-based (Vaswani et al., 2017) machine-learning technique, which results in a so-called language model. The method also allows the language model to be fine-tuned for a specific task (i.e. adapting the pre-trained model to the target task), which in our case is categorising or forecasting ECtHR judgements.

To use BERT (which has a limit of 512 tokens) on long(er) case documents, without introducing a maximum text length restriction, H-BERT (Chalkidis et al., 2019) was introduced to process each fact separately, after combining them by using a self-attention layer to generate an embedding for a case. The embedding is then used for categorisation and forecasting.[9] Instead of the standard BERT model (which Chalkidis et al. (2019) reported to have sub-par performance), we used LEGAL-BERT (Chalkidis et al., 2020) in our experiments. LEGAL-BERT is a BERT model which was pre-trained on legal texts from different sources.

[9]While BERT can process each case including up to 512 tokens (i.e. meaningful word parts), our H-BERT implementation can use up to 1,024 tokens (i.e. 128 tokens for each of the first eight facts).

BERT and many of its variations, including H-BERT, have resulted in substantial improvements in a large variety of text classification tasks, compared to the previous state-of-the-art. Specifically, Chalkidis et al. (2019) have shown that using H-BERT resulted in very high performance (f1-score of 0.82) for the binary task (violation of at least one article of ECHR vs. no violation), and an even higher f1-score of 0.83 for LEGAL-BERT, on the same dataset (Chalkidis et al., 2020).

In the following, we report the results per class, for each model. Our main evaluation metric is the f1-score (see Chapter 3).

7.4 Results

We started our experiments with Setup 1, by testing on all the data from 2020. To our surprise, the results (see Tables 7.2 and 7.3 for the performance, per class) for classifying the final judgements were very low, compared to Chalkidis et al. (2019). In contrast with our expectations, forecasting final judgements on the basis of communicated cases, instead of on the basis of the facts of final judgements, yielded better results when using H-BERT. Compared to Chalkidis et al. (2019), however, our training set was much smaller (2,264 cases vs. 7,100 cases, respectively). The reason for this difference was that not all cases are communicated by the court (a requirement for inclusion in our dataset).

However, when trying Setup 2, where we trained using less data (i.e. until 2018) and tested on all the 2019 data, results were as expected. Specifically, f1-scores ranged between 0.79 and 0.92 for the categorisation task (see Table 7.4), and performance was much lower for the forecasting task, with f1-scores ranging from 0.60 to 0.65 (see Table 7.5).

In line with Setup 2, the results of Setup 3 (i.e. training with data up to and including 2017 and testing on all of the 2018 data)

2020 - Final judgements						
		Prec.	Recall	F1	Accuracy	#
SVM	no viol.	0.46	0.93	0.62		167
	violation	0.93	0.46	0.62		342
	avg.	**0.70**	**0.70**	**0.62**	0.62	509
H-BERT	no viol.	0.42	0.92	0.58		167
	violation	0.91	0.38	0.53		342
	avg.	**0.66**	**0.65**	**0.56**	0.56	509
LEGAL-BERT	no viol.	0.42	0.90	0.58		167
	violation	0.89	0.40	0.55		342
	avg.	**0.66**	**0.65**	**0.57**	0.57	509

Table 7.2 | Performance (precision, recall, f1-score, and accuracy) for SVM, H-BERT, and LEGAL-BERT models (per class) for final judgement categorisation, trained on cases between 2000 and 2019 and tested on cases decided in 2020 (setup 1)

show a similar (and expected) pattern. Tables 7.6 and 7.7 provide an overview of these results.

When running the same experiments using successively smaller datasets (i.e. testing on data from 2017 and 2016), the same pattern is visible as for Setups 2 and 3. That is, the performance when classifying final judgements is much higher than when forecasting final judgements. Table 7.8 shows the f1-scores, for both tasks, for all years (of the test set) ranging from 2016 to 2020, and for all three algorithms. Besides showing that categorisation performance is generally (except for 2020) higher than forecasting performance, these results also show that while H-BERT and LEGAL-BERT generally outperform SVM in categorisation (except for 2020), they do not perform better than SVM in forecasting.

2020 - Communicated cases						
		Prec.	Recall	F1	Accuracy	#
SVM	no viol.	0.47	0.51	0.49		167
	violation	0.75	0.72	0.73		342
	avg.	**0.61**	**0.61**	**0.61**	0.65	509
H-BERT	no viol.	0.45	0.61	0.52		167
	violation	0.77	0.63	0.69		342
	avg.	**0.61**	**0.62**	**0.60**	0.62	509
LEGAL-BERT	no viol.	0.42	0.54	0.47		167
	violation	0.74	0.63	0.68		342
	avg.	**0.58**	**0.58**	**0.57**	0.60	509

Table 7.3 | Performance (precision, recall, f1-score and accuracy) for SVM, H-BERT, and LEGAL-BERT models per class for forecasting judgements, trained on communicated cases between 2000 and 2019 and tested on communicated cases that received a judgement in 2020 (setup 1)

7.5 Discussion

Our results confirm our intuition regarding the increased difficulty of the task of forecasting judgements, as opposed to categorising judgements. However, the tasks are conceptually very different, and therefore comparing them in terms of accuracy may not be entirely fair. Nevertheless, both fall under 'predicting court decisions' in the existing literature. Our results illustrate that predicting court decisions which have not yet been made is a much harder task than current academic research may suggest.

One potential explanation for the higher performance of the categorisation approach, compared to the forecasting approach, may be the higher amount of data used (i.e. an average of 2,000 words for the facts part of the judgement, versus 1,800 words for the communicated case). Since LEGAL-BERT and H-BERT have a limited input

2019 - Final judgements						
		Prec.	Recall	F1	Accuracy	#
SVM	no viol.	0.69	0.95	0.80		229
	violation	0.95	0.68	0.79		311
	avg.	**0.82**	**0.81**	**0.79**	0.79	540
H-BERT	no viol.	0.90	0.92	0.91		229
	violation	0.94	0.93	0.93		311
	avg.	**0.92**	**0.92**	**0.92**	0.92	540
LEGAL-BERT	no viol.	0.87	0.90	0.88		229
	violation	0.92	0.90	0.91		311
	avg.	**0.90**	**0.91**	**0.90**	0.90	540

Table 7.4 | Performance (precision, recall, f1-score and accuracy) for SVM, H-BERT, and LEGAL-BERT models per class for final judgement categorisation, trained on cases between 2000 and 2018 and tested on cases decided in 2019 (setup 2)

length of up to 512 or 1,024 tokens (respectively), the difference will not play a role for these models. However, the SVM does not have such a limit. Consequently, we also evaluated the SVM on the 'shortened' facts of the final judgements. Specifically, we removed the facts from the middle of the text (under the presumption that the most important information is present at the beginning and the end), until it was approximately the same length as the text of the corresponding communicated case. This change, however, did not affect the performance, as the SVM on this trimmed data yielded f1-scores of 0.61, 0.83, and 0.77 for 2020, 2019 and 2018, respectively (compared to 0.62, 0.79, and 0.78). This suggests that the facts are formulated so as to be affected by the final ruling, rather than that the higher amount of data brings a tangible benefit.

The SVM model allows us to inspect the top coefficients (weights) of n-grams assigned by the system. We observe that the system often prioritises longer n-grams (the average length for the 100 top features

2019 - Communicated cases						
		Prec.	Recall	F1	Accuracy	#
SVM	no viol.	0.62	0.53	0.57		229
	violation	0.69	0.77	0.73		311
	avg.	**0.66**	**0.65**	**0.65**	0.67	540
H-BERT	no viol.	0.57	0.67	0.61		229
	violation	0.72	0.63	0.67		311
	avg.	**0.64**	**0.65**	**0.64**	0.65	540
LEGAL-BERT	no viol.	0.55	0.50	0.52		229
	violation	0.66	0.70	0.68		311
	avg.	**0.60**	**0.60**	**0.60**	0.61	540

Table 7.5 | Performance (precision, recall, f1-score and accuracy) for SVM, H-BERT, and LEGAL-BERT models per class for forecasting judgements, trained on communicated cases between 2000 and 2018 and tested on communicated cases that received a judgement in 2019 (setup 2)

2018 - Final judgements						
		Prec.	Recall	F1	Accuracy	#
SVM	no viol.	0.67	0.91	0.77		210
	violation	0.92	0.70	0.79		309
	avg.	**0.79**	**0.80**	**0.78**	0.78	519
H-BERT	no viol.	0.86	0.72	0.78		210
	violation	0.83	0.92	0.87		309
	avg.	**0.84**	**0.82**	**0.83**	0.84	519
LEGAL-BERT	no viol.	0.88	0.78	0.83		210
	violation	0.86	0.93	0.89		309
	avg.	**0.87**	**0.85**	**0.86**	0.87	519

Table 7.6 | Performance (precision, recall, f1-score and accuracy) for SVM, H-BERT, and LEGAL-BERT models per class for final judgement categorisation, trained on cases between 2000 and 2017 and tested on cases decided in 2018 (setup 3)

2018 - Communicated cases						
		Prec.	Recall	F1	Accuracy	#
SVM	no viol.	0.62	0.55	0.58		210
	violation	0.72	0.77	0.74		309
	avg.	**0.67**	**0.66**	**0.66**	0.68	519
H-BERT	no viol.	0.60	0.63	0.61		210
	violation	0.73	0.71	0.72		309
	avg.	**0.67**	**0.67**	**0.67**	0.68	519
LEGAL-BERT	no viol.	0.59	0.52	0.55		210
	violation	0.69	0.75	0.72		309
	avg.	**0.64**	**0.63**	**0.64**	0.66	519

Table 7.7 | Performance (precision, recall, f1-score and accuracy) for SVM, H-BERT, and LEGAL-BERT models per class for forecasting judgements, trained on communicated cases between 2000 and 2017 and tested on communicated cases that received a judgement in 2018 (setup 3)

F1-score					
	2020	2019	2018	2017	2016
Training set size	2264	1806	1386	976	640
Test set size	509	540	519	503	447
SVM (forecasting)	0.61	0.65	0.66	0.65	0.64
H-BERT (forecasting)	0.60	0.64	0.67	0.66	0.66
LEGAL-BERT (forecasting)	0.57	0.60	0.64	0.64	0.58
SVM (categorisation)	0.62	0.79	0.78	0.78	0.75
H-BERT (categorisation)	0.56	0.92	0.83	0.84	0.82
LEGAL-BERT (categorisation)	0.57	0.90	0.86	0.84	0.82

Table 7.8 | F1-scores for the SVM, H-BERT, and LEGAL-BERT models for both tasks, evaluated on test set data from 2016 - 2020, including the size of training and testing sets

is 2.475) for final judgements, while for communicated cases it prioritises unigrams and common collocations consisting of two words, such as 'public prosecutor' or 'minor offences' (the average length for the 100 top features is 1.405).

We should also take into account that a communicated case is a summary of one applicant's complaint. As a result, it only reflects one party's version of events, and it may therefore be subjective and incomplete. After sending the communicated case to the State involved, the court conducts an investigation and inspects the arguments of the State as well. Consequently, the final judgement contains a more thorough and objective description of the facts, which takes the arguments of both parties into account. This explains why the facts available in communicated cases can differ considerably from the set of facts presented in the final judgement.

This bias towards a violation of human rights can also be observed in the results. For the forecasting task, all the models show a higher performance when predicting the 'violation' label than when predicting the 'non-violation' label (see Tables 7.3, 7.5, and 7.7). In contrast, the gap in performance when predicting the two labels for the categorisation task is considerably smaller (see Tables 7.2, 7.4, and 7.6), which confirms our expectation that the description of the facts in the final judgements is a better representation of the events, and therefore a better predictor of the outcome. Nevertheless, for the 2018 and 2019 data, the performance when predicting the 'violation' label using the communicated cases data (i.e. the forecasting task) is still lower than the overall performance (or the 'violation' label performance) when using facts extracted from the final judgements (i.e. the categorisation task).

The only time that forecasting judgements showed a higher performance than categorising judgements was when the 2020 test data was used. However, this was caused by the much lower than usual categorisation performance. Unfortunately, we have no explanation

for this pattern, despite the effort we spent trying to investigate whether or not the 2020 data showed deviating patterns compared to the data from earlier years. For example, the average length of the 2020 cases and the overall vocabulary is consistent with previous years, as well as the distribution of cases between different States, and therefore between different Chambers. The court judged only slightly (4%) fewer cases in 2020 than in 2019, and it did not adopt any new policies compared to previous years. There is no indication that the court used a different selection approach for the cases it ruled on. Since the cases originated in the years before 2020, it is also unlikely that this pattern is due to pandemic-related human rights violations. Finally, the format of case law has also remained the same. For now, we are forced to consider the performance based on 2020 data to be an anomaly (as with so many other things in 2020). Whether this deviating pattern will continue in 2021 remains to be seen.

7.5.1 Future Work

We have discussed a range of approaches to forecasting the outcomes of pending applications. Each of these could be improved with more careful tuning, pre-processing, data selection, feature selection, etc. Additional categorisation or forecasting algorithms could also be used. However, this was not the goal of the present paper. By releasing our dataset, together with a number of baselines reported in this paper, we hope to have provided a new starting point for the task of *forecasting* ECtHR judgements.

Regarding future research, it would be interesting to assess whether or not selecting the last tokens, or tokens from specifically chosen facts, would be beneficial for BERT-like models. For example, these models might yield better results, as initial facts are generally about the procedure and the applicant themselves, while facts from the end of the document are often more closely related to events connected with the alleged violation of human rights. Due to the limited

data available, we only investigated whether or not a case violated *any* article of the ECHR. However, it would be interesting to assess how the difference in performance between forecasting and categorisation is affected when individual articles are investigated.

While we can forecast pending applications using data from communicated cases, this does not allow us to forecast the judgements for any given future case, as this data may not always be available (e.g., not all cases are communicated to the State). Forecasting decisions using other data that are available before the judgement is known (i.e. from other sources, such as submissions by the parties, etc.) may be even harder than using the uniform documents created by the court for communicated cases.

While predicting judgements is an interesting task in itself, it is beneficial to also gain insight into how the system reaches a certain outcome, and therefore to take a step toward explainable AI (Bench-Capon, 2020; Collenette et al., 2020) and large-scale automatic legal analysis. This requires (for example) an understanding of which facts lead to which judgement. Determining the basis of a categorisation is important for the categorisation task in particular, as there is no practical use in determining a an already known outcome (see Chapter 4).

Several methods that are often used in categorisation tasks allow for the categorisation basis to be determined (to some extent). An SVM (for example) allows the inspection of its coefficients, to evaluate which words and phrases are more characteristic for one class than another. In Chapter 8 we also suggest evaluating such a system at the sentence level, to identify and highlight sentences that have the highest probability of belonging to a specific class. Furthermore, the architecture of H-BERT (for example) allows one to assess which of the (eight) facts (or questions) included had the largest impact on categorisation, on the basis of so-called attention (Vaswani et al., 2017). Unfortunately, LEGAL-BERT cannot be used for this. While it often

produces very high scores, especially for final judgement categorisation, and it may function as a good reference point for high categorisation performance, it remains a black box.

7.6 Ethical Considerations

We believe it is important to emphasise that our goal with this work is only to (try to) forecast and classify court judgements. Our interest is scientific, and it is focused on assessing whether natural language processing systems are able to identify certain patterns in legal judgements. We do not think that any of the models described in this chapter can or should be used for *making decisions* in court, especially those where human rights are at stake (which concerns the majority of the courts around the world). Moreover, we are opposed to the use of such models in other high-stakes situations, due to the inability of these types of models to deal with new legal developments and interpretations, previously unobserved issues (Campbell, 2020; Berk et al., 2019), a lack of transparency (Završnik, 2020; Deeks et al., 2019), and cybersecurity concerns (Nichols, 2019). Chapter 9 provides a further discussion on the ethical considerations in this field.

7.7 Conclusion

In this chapter, we investigated the distinction between forecasting court judgements and classifying court judgements. Forecasting of court judgements is based on data which are available before the outcome is known (such as the ECtHR cases that have already been communicated), whereas classifying court judgements is based on (a subset of) data compiled when the outcome was already known (such as facts from an ECtHR ruling). As we suggested in Chapter 4, making this distinction is important. Both earlier research (Branting et al., 2020) and the experiments conducted in this chapter show that per-

formance seems to be substantially lower when forecasting future judgements than when decisions already made by the court are categorised. Categorisation performance should therefore not be used as an indication of how well these types of systems are able to forecast court judgements. Interestingly, while more sophisticated models appeared to be beneficial for the simpler categorisation task, this was not the case for the harder forecasting task.

CHAPTER 8

JURI SAYS

In this short chapter we present the web platform JURI SAYS, which automatically predicts European Court of Human Rights decisions based on communicated cases, similar to the system described in Chapter 7. Our system therefore forecasts *future* court judgements. The platform is available at `https://jurisays.com`, and it shows predictions of the court decisions, compared to its actual decisions. It is automatically updated every month, by including predictions for all new cases. Additionally, the system highlights the sentences and paragraphs that are most important for forecasting (i.e. violation vs. no violation of human rights).

Chapter adapted from:

Medvedeva, M., Xu, X., Wieling, M., and Vols, M. (2020b). Juri says: Prediction system for the European Court of Human Rights. In *Legal Knowledge and Information Systems: JURIX 2020: The Thirty-third Annual Conference, Brno, Czech Republic, December 9-11, 2020*, volume 334, page 277. IOS Press.

8.1 Introduction

In this chapter, we introduce an online platform, JURI SAYS, which automatically retrieves legal documents from the ECtHR database, and subsequently forecasts the outcomes of cases on the basis of information which was available *before* a judgement was made. In addition

to forecasting decisions, JURI SAYS identifies and highlights sentences that were key to its prediction process.

The JURI SAYS system can be divided into three parts: 1) a database, 2) a machine learning system, and 3) a web platform. Each part is independent from the others and offers an Application Programming Interface (API) to add flexibility for the future, allowing (for example) more documents to be added, new machine learning models to be included, or the interface to be adjusted. Before discussing the architecture of the system, we provide some necessary background on the legal data supporting our system.

8.2 JURI SAYS

8.2.1 Database

Our database only includes documents in English. Every month new documents are automatically downloaded, and a new machine learning model is automatically trained to forecast the ECtHR decisions of that month (see below). At the moment of the platform going online, the database contained 4,929 communicated cases, along with the decisions they are associated with. As opposed to the dataset in Chapter 7, only admissible cases were used for JURI SAYS. While the forecasting is based only on communicated cases, we also include information from cases over the last ten years that were not communicated in order to increase the amount of data available to train our model. For those cases, we only extract the *Facts* part from the final judgement document, as in Chapter 6.

Our system automatically extracts the raw text of communicated cases from the database of the ECtHR, in addition to some metadata, such as decisions (for admissibility cases and judgements), dates, parties, articles involved, etc. The decisions are then associated with the communicated cases, according to their application numbers.

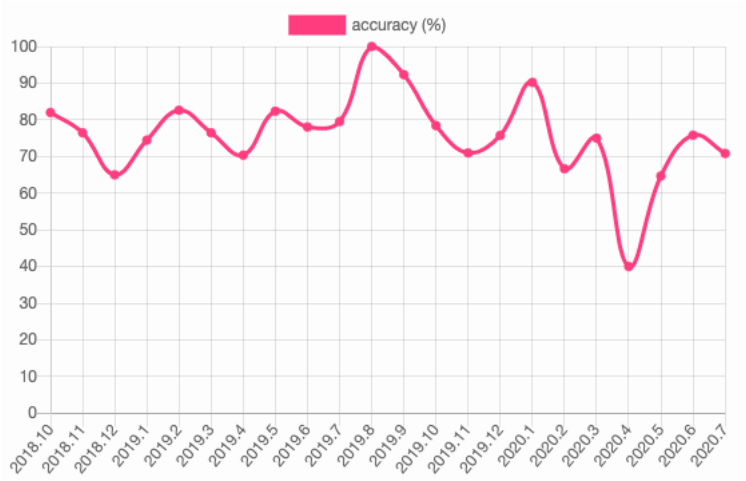


Figure 8.1 | Accuracy of JURI SAYS in 2018-2020

8.2.2 Machine Learning System

Every month, after downloading the new documents, the system powering our web platform JURI SAYS carries out three tasks automatically. It first trains a new SVM machine learning model using all data *except* for data from the most recent month. It then forecasts the outcomes of judicial decisions for cases from the most recent month on the basis of the newly-created model. The performance (accuracy) of JURI SAYS for each month up to July 2020 can be found in Figure 1. Finally, the system identifies how strongly each sentence in the text of the communicated case is related to the actual judgement of the court (by estimating the probability of the sentence belonging to a case with a violation of human rights, versus a case without a violation; see also Figure 2).

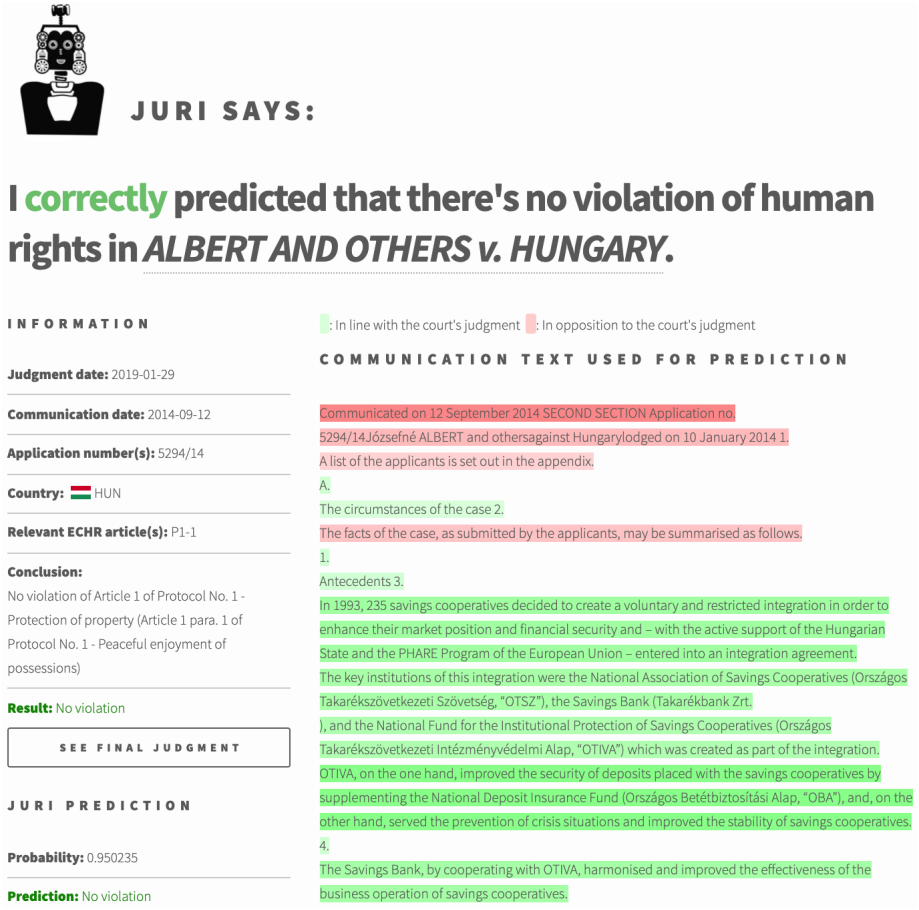


Figure 8.2|An example of a correctly forecasted case by JURI SAYS with highlighted sentences. Sentences highlighted in green are consistent with the actual outcome.

8.2.3 Web Platform

JURI SAYS is the web platform which presents the results of applying our machine learning system to the extracted ECtHR data. JURI SAYS is updated every month, publishing predictions for the most recent ECtHR cases. It also offers a list of all the historical cases that may be ordered or filtered, by date or by the relevant article. For every single case there is a separate page offering more detailed information, including the forecasted outcome of the case, together with an associated probability of that forecasted outcome, and the actual court judgement. For each sentence in the text of the communicated case, the forecasted label and associated probability is shown when the mouse pointer is hovered over a sentence. This is implemented by splitting the text into sentences, and forecasting the outcome using each sentence separately. Then, confidence scores [1] produced by the classifier are used to determine the probability. Sentences which are highlighted in green are consistent with the court's actual decision, and those in red are more likely to be associated with the opposite decision. See Figure 8.2 for an example. The intensity of the colour reflects how strongly associated the sentences are with the respective decisions.

8.3 Conclusion

In this short chapter, we presented JURI SAYS, an automatic outcome forecasting system for ECtHR judicial decisions. Our system uses automatically extracted textual information from documents which are available long before the court decision is made. In addition, our model predicts cases for the following month (i.e. the future), which is a hard task, as we have demonstrated in Chapter 6. Therefore, it is nice to see the relatively high performance of our system, which has

[1] The confidence scores are computed by the classifier using Platt scaling (Platt et al., 1999).

an accuracy of 75% (as of July 2020). By automatically highlighting important sentences, and automatically updating every month, our system aims to offer a user-friendly web platform for legal professionals.

PART III

Ethical consideration and conclusions

CHAPTER 9

Innocent until Predicted Guilty

In this chapter, we discuss the implications of using machine learning for judicial decision-making in situations where human rights may be infringed. We argue that the use of such tools in these situations should be limited, due to the inherent status quo bias and dangers of reverse-engineering. We discuss how these issues already exist in judicial systems, before any machine learning tools are used, and how introducing these tools might exacerbate such issues.

9.1 Introduction

Ever since legal technology began rapidly gaining popularity, law firms and government organisations have been using tools to automate, speed up, and simplify their work. For example, Estonia introduced a robo-judge to resolve small claims in its courts (Niiler, 2019), and courts in the USA are using risk assessment tools to determine amounts of bail granted (Stevenson, 2018).

Most of the technology and algorithms used in legal tech were developed in other domains, then applied to legal data later on. However, in the legal domain, as in (for instance) the medical domain, the consequences of using these algorithms may be much greater than for the domains for which the algorithms were originally created. Often, these consequences are not considered when the algorithms are applied to a new domain.

One of the subjects which fascinates people working in the field of AI and Law is how to create a fair and unbiased decision-making system that would be able to assist judges in their decision-making, or even replace them in court (Bex and Prakken, 2020). The idea of making judges' work easier, and courts faster and more transparent, is appealing. We agree that automatic systems may be suitable for use at some levels of judicial decision-making (see Stranieri and Zeleznikow, 2006 and Berk et al., 2019 for a variety of methods), particularly in low-stakes environments dealing with minor claims. However, we also argue that this might not be a good idea in situations where human rights may have been infringed.

Consider the following hypothetical situation, in which a criminal court uses a fully automated system to make decisions, based on the facts of a case provided by the parties. The model weighs the facts, and comes up with a decision based on precedents. The system is completely transparent and explainable, and it makes its decisions with very high accuracy. Now consider this from the perspective of a prosecutor's office that uses this system to deal with many cases appearing before the court. Having dealt with a very large number of cases, the prosecutor is able to understand (almost) exactly how the automatic system works, and reverse-engineer (i.e. determine the rules which determine the judgements) it, in order to evaluate the outcome when presenting the case to the court in a certain way. Let us assume that they were able to reverse-engineer the system, and that their system overlaps with the court's system 99% of the time. They play out every scenario before the trial, and make sure that they use the strategy which (almost) guarantees that they will win. If they are not able to identify a winning strategy, they decide not to prosecute, thereby providing the office with a 99% winning rate. Would such a scenario be desirable? What if, instead of the prosecutor's office, the system is available to a very large law firm which defends violent criminals or corrupt politicians, and is able to present the facts in

such a way that the system would be favourable to its clients?

We are, of course, quite a long way from building such a system yet. Predictive models that are developed today do not have a high enough accuracy to be used for decision-making in actual courts. We have demonstrated in previous chapters that the accuracy of forecasting future decisions remains much lower, reaching only up to 75% (see Chapter 8). Unfortunately, this does not stop the courts from using such models. Specifically, decision-predicting systems are now used for decision-making, for example, by courts in the United States. The most common system is a risk assessment tool called COMPAS,[1] which is used to decide on bail and probation, and is sometimes even used to decide sentencing (Angwin et al., 2016), even though it may show a racial bias. Unfortunately, the system predicts recidivism with the same accuracy as a random person with little or no criminal expertise (Dressel and Farid, 2018).

While the performance of decision-predicting models is not very high, the interest in using such systems is still growing. Aletras et al. (2016) suggested that using predictive models could be useful for prioritising certain cases over others in the European Court of Human Rights. Chen et al. (2019) suggested using their prison-term-prediction system (for the Chinese Supreme Court) to anonymously check a judge's decision, although their system predicts exactly the same term as the judge in only about 9% of cases. Zhong et al. (2018a) suggested that legal judgement predictions can be used to assist lawyers and judges. Their model, TopJudge, is designed to predict prison terms, but it occasionally suggests a death penalty in cases where the judge had decided on a prison term of less than one year.

In this chapter, we argue that in high-stakes court cases (e.g., cases in which people's lives or fundamental rights are at stake, or which

[1] `https://www.equivant.com/practitioners-guide-to-compas-core/`, accessed on 04/04/2022

may have a strong influence on public policy), the use of such systems might have implications that are detrimental to people's lives and accessibility to justice, which should be prevented.

While previous research considered several ethical and technical considerations regarding such systems, we focus specifically on the dangers of reverse-engineering. In addition, we discuss several reasons why decision-making tools should be developed with extreme caution. Such reasons include the risk of misinterpreting decision predicting as decision-making, and an unavoidable inherent status quo bias.

9.2 Arguments against Automatic Decision-making

The push for transparency in court settings is natural and intuitively clear. The courts should be able to judge systematically and predictably, and should be accountable in cases where that does not happen. A multitude of laws, including constitutions, ensure that this is the case in countries across the world.

With predictive legal models becoming better and better, there appears to be an increasing tendency to evaluate whether judges and their biases may be replaced by machines which can control biases (Chohlas-Wood et al., 2020; Khademi et al., 2019). We would like to note that the idea of having a completely unbiased judge is somewhat peculiar. When we think about laws, we think of very elaborate instructions that (try to) account for any situation. And while laws are written in the hope that they would cover most issues, they can often be interpreted in several different ways, whenever a new situation is encountered. Legal systems rely on judges to interpret general laws in individual cases (Završnik, 2020). By doing so, they introduce their personal bias. Two different judges might judge the same case differently, and neither of them would be absolutely wrong in their decision. Of course, this does not mean that all judicial biases are

justified. For example, discrimination based on gender, skin colour, or ethnicity is, of course, unacceptable. However, in the high-stakes situations we focus on in this chapter, some level of judge bias is inevitable.

At present, judicial assistance tools are being introduced and (unfortunately) they can be (incorrectly) used as judicial decision-making systems. Below, we discuss why attempts to build such systems may be misguided in high-stakes situations.

9.2.1 Decision-predicting vs. Decision-making Systems

The first issue in discussions on automatic judicial decision-making is that of mixing up definitions. In many tasks that involve machine learning, decision-predicting and decision-making may appear synonymous. However, this is not the case in a court setting.

Classification using machine learning functions by providing the model with some kind of representation of (the text of) old cases (i.e. data points for each case). The model then tries to identify which data points (for example, whether or not the word 'children' occurs in the description of facts available to the court) are most representative for each class (i.e. verdict). Therefore, if there are any biases that can be found within those data points, the system will exploit them to improve its prediction. However, it may not always be the case that these biases should remain present in the decision-making process (Edwards and Veale, 2017). In essence, the decision-predicting system is able to determine and *distinguish between past decisions*, whereas a decision-making system should be able to *generate new decisions*. It is clear that (past) decision-predicting systems should not be used for making new decisions.

9.2.2 Status Quo Bias

Given that one always has to train a machine learning system on older cases, for it to be able to predict decisions in future cases, the system will always reflect the way old cases were decided (Campbell, 2020; Berk et al., 2019). It is no surprise, therefore, that predicting the decisions of future judgements is consistently harder than predicting judgements from the same period (see Chapter 6).

Without explicit knowledge about gradually changing concepts (e.g., the introduction of electronic mail), even the most advanced NLP-based machine learning techniques cannot predict changes in how the law needs to be interpreted. Human intervention is necessary, to prevent an inherent status quo bias.

9.2.3 Dangers of Reverse-engineering

Another issue when using these types of algorithms lies within the area of cybersecurity. Nichols (2019) raised a concern about the possibility of hacking and manipulating algorithms, in order to benefit self–interested third parties. Nichols thus argued for transparency in the development of algorithms, as have many others (Završnik, 2020; Thomsen, ming; Deeks et al., 2019). This intuitive argument for transparency, however, may also be problematic. Specifically, a transparent predictive system may create opportunities to abuse the algorithm, using adversarial machine learning techniques (Kurakin et al., 2016).

Consider the following artificial example regarding a low-risk decision-making machine. In the hallway of the court, there is an automatic cookie-vending machine that decides whether or not you can have a cookie. It bases its decision on your personal history. Consequently, the vending machine asks a range of questions to determine either that you can have another cookie, or that you have already had enough cookies. To do so, it asks what kind of cookies you have

already eaten, if you ate any fruit for breakfast, etc., and it uses stored information from times you previously used the cookie machine (in a similar way as risk-assessment tools). You do not know how the machine works, you just answer the questions and, unfortunately, you are denied a cookie. You think this is unreasonable, as you really wanted a cookie. If you ask for an explanation, and it appears the system is (in legal terms) not explainable, you were denied justice. However, if the model is explainable, it should provide you with the details on how the answers were weighted and how a certain decision was reached. If you know what the system already knows about you and how the system determines its decision, you may be able to create a computer program which provides you with the answers you have to give to increase the number of cookies you are able to obtain. Importantly, if the system is consistent, it is also possible to recreate the algorithm, even without access to the specifics of how it works.

Of course, when the risks are higher, the consequences of being able to reverse-engineer a system may be a lot more dire. Similarly to deceiving the cookie machine, one could play out every scenario before appearing in court, and (for instance) subsequently decide either to go to trial or to settle.

9.3 Discussion and Conclusion

All of the aforementioned issues also exist in judicial systems where machine learning tools are not used. Judges and lawyers deal with precedents, to make their cases and come to a certain decision, and law firms and lawyers try to 'reverse-engineer' the judge, to predict their behaviour given the specific circumstances of the case (Bruijn and Vols, 2020). The presence of an automated system, however, amplifies the problem.

In this chapter, we pointed out the difference between legal decision-predicting and legal decision-making, and argued that the

latter has no place in a court where human rights may be at stake. We pointed out that, given how machine learning works, it is impossible to avoid a status quo bias in decision-making, and we stressed that explainable AI is vulnerable to reverse-engineering. If one knows that the machine will judge systematically, given enough data, one may be able to predict the outcome of all cases. This could allow law firms to play out various strategies "in-house" (evaluating what the results of a case will be), before going to court, which creates ample opportunities to abuse the system.

In this chapter, we unfortunately are not able to offer a solution to this issue. We merely caution the reader that, although using machine learning has substantial potential in legal analytics and decision support, we think it should not be introduced for making judicial decisions in situations where human rights are at stake. In addition, in cases where it has already been introduced, stricter regulations need to be enforced, to ensure that decisions are never made solely on the basis of a machine learning system's predictions.

CHAPTER 10

Discussion and Conclusion

Judicial systems across the world suffer from a backlog in terms of handling court cases, which limits accessibility to justice.[1] The technology available today has the potential to automate many legal processes and radically change the way that litigation is approached. In this dissertation, we therefore focused on automatic legal analysis, and specifically, on predicting court judgements.

After a general introduction (in Chapter 1), and a high-level overview of the early quantitative techniques used in legal analysis (in Chapter 2), we provided an introduction to the techniques which are presently used in the field of automatically predicting court judgements (in Chapter 3). In these chapters, we opted for a high-level discussion that should be comprehensible to researchers working in the legal domain.

We then provided definitions of different tasks that have historically fallen under the term of 'predicting court decisions' in Chapter 4. We specifically distinguished *outcome identification*, *outcome-based judgement categorisation*, and *outcome forecasting* and provided a review of the previous research within each of the three tasks.

[1] `https://www.ibanet.org/article/62C03066-B9C0-452F-950B-37718E5AD5B6`, accessed on 23/01/2022)

In the following chapters, we demonstrated our own research and experiments within each of the three tasks. In Chapter 5, we discussed an identification task we undertook, where eviction-related cases were extracted from a large Dutch case law database, after which we identified their outcomes. This approach was very useful for collecting datasets on a specific topic. In Chapter 6, we demonstrated a categorisation task which involved classifying final judgements from the ECtHR, using facts extracted from the judgements. Additionally, we experimented with using earlier data for training and newer data for testing, and demonstrated that doing so had a considerable negative effect on performance. In the same chapter, we also experimented with categorising judgements using only judges' surnames. In Chapter 7, we conducted experiments in forecasting ECtHR decisions, by using documents published by the court long before a certain judgement was made. We demonstrated that this is a much harder task, which produces an overall lower performance than the categorisation of final judgements. In Chapter 8, we presented an online platform, JURI SAYS, which automatically forecasts outcomes of ECtHR judgements. In addition, the system highlights the sentences and paragraphs which are the largest contributors to a particular prediction. Finally, Chapter 9 was dedicated to discussing ethical considerations regarding the use of predictive models in high-stakes situations. Specifically, we discussed the issues of the status quo bias, and the dangers of reverse-engineering these predictive models.

It is important to realise that the experiments described in this dissertation should be seen only as a first step in a more elaborate legal analysis. The identification results can be used to supplement datasets and benefit research on a specific topic. The categorisation experiments may point towards particular patterns within judgements, but they still need to be analysed and made sense of from a legal perspective. Similarly, if the forecasting results are used to make estimates about the future, the systems must become easier to

explain, so that the estimates are understandable.

The results of the three tasks are dependent on the information that is available in (publicly accessible) judgements from different courts, and how this information is structured. Since dealing with text requires the use of natural language processing techniques, one needs to take into account that the majority of NLP systems are notoriously reliant on the specific language of the text. Whereas many different techniques are available for English, this is not the case for all the other languages used in courts around the world. More effort, therefore is needed, to make NLP techniques (in the legal domain) available for other languages than English.

Another potential issue for all of the automation tasks using case law as input is selection bias (Berk, 1983). Since most courts do not publish all their case law online, one must always take into account that systems developed with the available data might not be general enough. Whereas these systems may be good at classifying decisions in the test set, they might also show much worse performance when dealing with completely new cases. While such a situation is relatively easy to identify, more nuanced situations are also possible. For example, consider a system for forecasting supreme court decisions using (textual) information extracted from court of appeal cases. The first issue might be that the court of appeal does not publish all of its cases online. This creates a selection bias, generated by the court of appeal. In this situation, one needs to consider the publishing policy, and whether or not this might have an effect on prediction. Of course, this is not an issue if one has access to all case law of the court in question. The second issue, however, is harder to solve. When using the decisions of the court of appeal, one relies on the fact that someone decided to proceed to appeal in the supreme court, and that the supreme court has not dismissed the case. Cases that were not moved to the supreme court, but which would have been won, are not part of the dataset, as these are unknown. To alleviate this problem some-

what, the system should be made easier to explain. In this case, it is clear which parameters lead to a specific outcome, and can be compared against cases that have not (yet) been judged. One should also never rely on information which might have been biased at the outset. For example, if one tries to predict recidivism in committing a crime, that information only becomes available when someone is *caught* re-offending, which may result in a biased dataset. Consequently, it is essential to be aware and transparent about the (potential) selection bias present in the data one is analysing.

In conclusion, in this dissertation we have tried to show that the field of predicting court decisions shows potential with regard to the automation of legal processes and legal analysis. We have also demonstrated that there are many limitations to what today's systems are able to do. We have tried to introduce terminology that can be used to categorise the research on predicting court decisions, which should help to show the purpose of each task for the legal community. Finally, through this dissertation, which brought together legal analysis and natural language processing, we have shown the benefits of cross-disciplinary collaboration.

Bibliography

Aletras, N., Tsarapatsanis, D., Preoţiuc-Pietro, D., and Lampos, V. (2016). Predicting judicial decisions of the European Court of Human Rights: A natural language processing perspective. *PeerJ Computer Science*, 2.

Angwin, J., Larson, J., Mattu, S., and Kirchner, L. (2016). Machine bias: there's software used across the country to predict future criminals. and it's biased against blacks. ProPublica 2016. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing.

Arnold, C., Engst, B., and Gschwend, T. (2021). Scaling court decisions with citation networks. *Journal of Law and Courts*.

Ashley, K. D. (2017). *Artificial intelligence and legal analytics: new tools for law practice in the digital age*. Cambridge University Press.

Ashley, K. D. and Brüninghaus, S. (2009). Automatically classifying case texts and predicting outcomes. *Artificial Intelligence and Law*, 17(2):125–165.

Barco Ranera, L. T., Solano, G. A., and Oco, N. (2019). Retrieval of semantically similar philippine supreme court case decisions using doc2vec. In *2019 International Symposium on Multimedia and Communication Technology (ISMAC)*, pages 1–6.

Basile, A., Dwyer, G., Medvedeva, M., Rawee, J., Haagsma, H., and Nissim, M. (2017). N-gram: New groningen author-profiling model—notebook for PAN at CLEF 2017. In *CEUR Workshop Proceedings*, volume 1866.

Basile, A., Dwyer, G., Medvedeva, M., Rawee, J., Haagsma, H., and Nissim, M. (2018). Simply the best: Minimalist system trumps complex models in author

profiling. In Bellot, P., Trabelsi, C., Mothe, J., Murtagh, F., Nie, J. Y., Soulier, L., SanJuan, E., Cappellato, L., and Ferro, N., editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 143–156, Cham. Springer International Publishing.

Behn, D. and Langford, M. (2017). Trumping the environment? An empirical perspective on the legitimacy of investment treaty arbitration. *The Journal of World Investment & Trade*, 18(1):14–61.

Bench-Capon, T. (2020). The need for good old fashioned AI and law. *International Trends in Legal Informatics: A Festschrift for Erich Schweighofer. Editions Weblaw, Bern*, pages 23–36.

Berk, R., Berk, and Drougas (2019). *Machine learning risk assessments in criminal justice settings*. Springer.

Berk, R. A. (1983). An introduction to sample selection bias in sociological data. *American sociological review*, pages 386–398.

Bertalan, V. G. F. and Ruiz, E. E. S. (2020). Predicting judicial outcomes in the Brazilian legal system using textual features. In *DHandNLP@ PROPOR*, pages 22–32.

Bex, F. and Prakken, H. (2020). De juridische voorspelindustrie: onzinnige hype of nuttige ontwikkeling? *Ars aequi*, 69:255–259.

Bex, F. and Prakken, H. (2021). On the relevance of algorithmic decision predictors for judicial decision making. In *Proceedings of the 19th International Conference on Artificial Intelligence and Law (ICAIL 2021)*. ACM Press.

Bhilare, P., Parab, N., Soni, N., and Thakur, B. (2019). Predicting outcome of judicial cases and analysis using machine learning. *International Research Journal of Engineering and Technology*, 6(3):326–330.

Branting, K., Balhana, C., Pfeifer, C., Aberdeen, J., and Brown, B. (2020). Judges are from Mars, pro se litigants are from Venus: Predicting decisions from lay text. In *Legal Knowledge and Information Systems: JURIX 2020: The Thirty-third Annual Conference, Brno, Czech Republic, December 9-11, 2020*, volume 334, page 215. IOS Press.

Branting, L. K., Pfeifer, C., Brown, B., Ferro, L., Aberdeen, J., Weiss, B., Pfaff, M., and Liao, B. (2021). Scalable and explainable legal prediction. *Artificial Intelligence and Law*, 29(2):213–238.

Bricker, B. (2017). Breaking the principle of secrecy: An examination of judicial dissent in the european constitutional courts. *Law & Policy*, 39(2):170–191.

Bruijn, L. M. and Vols, M. (2020). Upperdogs versus underdogs. *Recht der Werkelijkheid*, 1:25–49.

Bruijn, L. M., Vols, M., and Brouwer, J. G. (2018). Home closure as a weapon in the Dutch war on drugs: Does judicial review function as a safety net? *International Journal of Drug Policy*, 51:137–147.

Bruijn, M. (2021). *The alternative war on drugs: drug evictions, cannabis regulation and the legal consequences of adapting to the limitations of criminal law in the field of drug control*. PhD thesis.

Bruinsma, F. J. (2007). The room at the top: Separate opinions in the grand chambers of the ECHR (1998-2006). *Recht der werkelijkheid*, 2007(2):7–24.

Bruinsma, F. J. and De Blois, M. (1997). Rules of law from westport to wladiwostok. separate opinions in the European Court of Human Rights. *Netherlands Quarterly of Human Rights*, 15(2):175–186.

Campbell, R. W. (2020). Artificial intelligence in the courtroom: The delivery of justice in the age of machine learning. *Colo. Tech. LJ*, 18:323.

Chalkidis, I., Androutsopoulos, I., and Aletras, N. (2019). Neural legal judgment prediction in English. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4317–4323, Florence, Italy. Association for Computational Linguistics.

Chalkidis, I., Fergadiotis, M., Malakasiotis, P., Aletras, N., and Androutsopoulos, I. (2020). LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Chalkidis, I., Fergadiotis, M., Tsarapatsanis, D., Aletras, N., Androutsopoulos, I., and Malakasiotis, P. (2021). Paragraph-level rationale extraction through regularization: A case study on European Court of Human Rights cases. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 226–241, Online. Association for Computational Linguistics.

Chao, W., Jiang, X., Luo, Z., Hu, Y., and Ma, W. (2019). Interpretable charge prediction for criminal cases with dynamic rationale attention. *Journal of Artificial Intelligence Research*, 66:743–764.

Chen, H., Cai, D., Dai, W., Dai, Z., and Ding, Y. (2019). Charge-based prison term prediction with deep gating network. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6361–6366, Hong Kong, China. Association for Computational Linguistics.

Cheng, X., Bi, S., Qi, G., and Wang, Y. (2020). Knowledge-aware method for confusing charge prediction. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 667–679. Springer.

Chien, C. V. (2011). Predicting patent litigation. *Tex. L. Rev.*, 90:283.

Chohlas-Wood, A., Nudell, J., Nyarko, J., and Goel, S. (2020). Blind justice: Algorithmically masking race in charging decisions. Technical report, Technical report.

Christensen, M. L., Olsen, H. P., and Tarissan, F. (2016). Identification of case content with quantitative network analysis: An example from the ECtHR. In *JURIX*, pages 53–62.

Collenette, J., Atkinson, K., and Bench-Capon, T. (2020). An explainable approach to deducing outcomes in European Court of Human Rights cases using adfs. *Frontiers in Artificial Intelligence and Applications*, 326:21–32.

Condevaux, C. (2020). Neural legal outcome prediction with partial least squares compression. *Stats*, 3(3):396–411.

Custers, B. and Leeuw, F. (2017). Quantitative approaches to empirical legal research. *Journal of Empirical Legal Studies*, 34:2449–2456.

De Jaeger, T. (2017). Gerechtelijke achterstand: de piñata van de wetgever. *NJW*, pages 290–307.

Deeks, A., Lubell, N., and Murray, D. (2019). Machine learning, artificial intelligence, and the use of force by states. *J. Nat'l Sec. L. & Pol'y*, 10:1.

Deng, L. and Liu, Y. (2018). *Deep learning in natural language processing*. Springer.

Derlén, M. and Lindholm, J. (2014). Goodbye van gend en loos, hello bosman? using network analysis to measure the importance of individual CJEU judgments. *European Law Journal*, 20(5):667–687.

Derlén, M. and Lindholm, J. (2017a). Is it good law? network analysis and the cjeu's internal market jurisprudence. *Journal of International Economic Law*, 20(2):257–277.

Derlén, M. and Lindholm, J. (2017b). Peek-a-boo, it's a case law system: Comparing the European Court of Justice and the United States Supreme Court from a network perspective. *German LJ*, 18:647.

Derlén, M. and Lindholm, J. (2017c). Serving two masters: CJEU case law in swedish first instance courts and national courts of precedence as gatekeepers. *Available at SSRN 2952783*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dhami, M. K. and Belton, I. (2016). Statistical analyses of court decisions: an example of multilevel models of sentencing. *Law and Method*, 10:247–266.

Doron, I. I., Totry-Jubran, M., Enosh, G., and Regev, T. (2015). An American friend in an Israeli court: An empirical perspective. *Israel Law Review*, 48(2):145–164.

Dressel, J. and Farid, H. (2018). The accuracy, fairness, and limits of predicting recidivism. *Science advances*, 4(1):55–80.

Dyevre, A. (2015). The promise and pitfalls of automated text-scaling techniques for the analysis of judicial opinions. *Available at SSRN*.

Dyevre, A. (2020). Text-mining for lawyers: How machine learning techniques can advance our understanding of legal discourse. *Available at SSRN 3734430*.

Edwards, L. and Veale, M. (2017). Slave to the algorithm: Why a right to an explanation is probably not the remedy you are looking for. *Duke L. & Tech. Rev.*, 16:18.

Epstein, L., Landes, W. M., and Posner, R. A. (2013). *The Behavior of Federal Judges: a Theoretical and Empirical Study of Rational Choice*. Harvard University Press.

Epstein, L. and Martin, A. D. (2010). Quantitative approaches to empirical legal research.

Evans, M., McIntosh, W., Lin, J., and Cates, C. (2007). Recounting the courts? applying automated content analysis to enhance empirical legal research. *Journal of Empirical Legal Studies*, 4(4):1007–1039.

Evans, M., McIntosh, W., Lin, J., and Cates, C. (2017). Kruisbestuiving tussen straf- en bestuursrecht: de ontwikkeling van de verwijtbaarheid in het bestuursrecht. *Nederlands Tijdschrift voor Bestuursrecht*, 10:351–357.

Fan, Y., Zhang, L., and Wang, P. (2020). Leveraging label semantics and correlations for judgment prediction. In *China Conference on Information Retrieval*, pages 70–82. Springer.

Frankenreiter, J. (2016). Are advocates general political? policy preferences of eu member state governments and the voting behavior of members of the european court of justice. *SSRN*, page 2778803.

Frankenreiter, J. (2017a). Network analysis and the use of precedent in the case law of the CJEU-a reply to derlen and lindholm. *German LJ*, 18:687.

Frankenreiter, J. (2017b). The politics of citations at the ECJ—policy preferences of EU member state governments and the citation behavior of judges at the European Court of Justice. *Journal of Empirical Legal Studies*, 14(4):813–857.

Garoupa, N., Gili, M., and Gómez-Pomar, F. (2012). Political influence and career judges: an empirical analysis of administrative review by the Spanish Supreme Court. *Journal of Empirical Legal Studies*, 9(4):795–826.

Goanta, C. (2017). Big law, big data. 7(3):1–20.

Golbeck, J., Robles, C., Edmondson, M., and Turner, K. (2011). Predicting personality from twitter. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, pages 149–156. IEEE.

Górski, Ł. (2021). Network science in law: A framework for Polish case-law citation network analysis. *IT Professional*, 23(5):62–66.

Hartung, D. (2021). Quantitative legal research in germany. In *Research Handbook on Big Data Law*. Edward Elgar Publishing.

Holá, B., Bijleveld, C., and Smeulers, A. (2012). Consistency of international sentencing: ICTY and ICTR case study. *European Journal of Criminology*, 9(5):539–552.

Huang, Y.-X., Dai, W.-Z., Yang, J., Cai, L.-W., Cheng, S., Huang, R., Li, Y.-F., and Zhou, Z.-H. (2020). Semi-supervised abductive learning and its application to theft judicial sentencing. In *2020 IEEE International Conference on Data Mining (ICDM)*, pages 1070–1075. IEEE.

Hunter, C., Nixon, J., and Blandy, S. (2008). Researching the judiciary: Exploring the invisible in judicial decision making. *Journal of Law and Society*, 35(s1):76–90.

Hutchinson, T. and Duncan, N. (2012). Defining and describing what we do: Doctrinal legal research. *Deakin L. Rev.*, 17:83.

Jiang, X., Ye, H., Luo, Z., Chao, W., and Ma, W. (2018). Interpretable rationale augmented charge prediction system. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 146–151.

Joachims, T. (2002). *Learning to classify text using support vector machines: Methods, theory and algorithms*, volume 186. Kluwer Academic Publishers Norwell.

Katz, D. M., Bommarito II, M. J., and Blackman, J. (2017). A general approach for predicting the behavior of the Supreme Court of the United States. *PloS one*, 12(4).

Kaufman, A., Kraft, P., and Sen, M. (2017). Machine learning, text data, and Supreme Court forecasting. *Project Report, Harvard University*.

Kaufman, A. R., Kraft, P., and Sen, M. (2019). Improving Supreme Court forecasting using boosted decision trees. *Political Analysis*, 27(3):381–387.

Kaur, A. and Bozic, B. (2019). Convolutional neural network-based automatic prediction of judgments of the European Court of Human Rights. In *AICS*.

Khademi, A., Lee, S., Foley, D., and Honavar, V. (2019). Fairness in algorithmic decision making: An excursion through the lens of causality. In *The World Wide Web Conference*, pages 2907–2914.

Kowsrihawat, K., Vateekul, P., and Boonkwan, P. (2018). Predicting judicial decisions of criminal cases from Thai Supreme Court using bi-directional GRU with attention mechanism. In *2018 5th Asian Conference on Defense Technology (ACDT)*, pages 50–55. IEEE.

Kufakwababa, C. Z. (2021). Artificial intelligence tools in legal work automation: The use and perception of tools for document discovery and privilege classification processes in southern african legal firms. Master's thesis.

Kurakin, A., Goodfellow, I., and Bengio, S. (2016). Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*.

Kyo, S. (2022). A quantitative analysis of legislation with harsher punishment in Japan. *Asian Journal of Law and Society*, 9(1):81–107.

Lage-Freitas, A., Allende-Cid, H., Santana, O., and de Oliveira-Lage, L. (2019). Predicting Brazilian court decisions. *arXiv preprint arXiv:1905.10348*.

Law, D. S. (2017). The global language of human rights: A computational linguistic analysis.

Le, H., Vial, L., Frej, J., Segonne, V., Coavoux, M., Lecouteux, B., Allauzen, A., Crabbé, B., Besacier, L., and Schwab, D. (2019). FlauBERT: Unsupervised language model pre-training for French. *arXiv preprint arXiv:1912.05372*.

Li, Y., He, T., Yan, G., Zhang, S., and Wang, H. (2019). Using case facts to predict penalty with deep learning. In *International Conference of Pioneering Computer Scientists, Engineers and Educators*, pages 610–617. Springer.

Lindholm, J. and Derlén, M. (2012). The Court of Justice and the Ankara agreement: Exploring the empirical approach. *Europarättslig tidskrift*, (3):462–481.

Lippi, M., Pałka, P., Contissa, G., Lagioia, F., Micklitz, H.-W., Sartor, G., and Torroni, P. (2019). Claudette: an automated detector of potentially unfair clauses in online terms of service. *Artificial Intelligence and Law*, 27(2):117–139.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Liu, Y.-H. and Chen, Y.-L. (2018). A two-phase sentiment analysis approach for judgement prediction. *Journal of Information Science*, 44(5):594–607.

Liu, Z. and Chen, H. (2017). A predictive performance comparison of machine learning models for judicial cases. In *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–6. IEEE.

Livermore, M. A., Riddell, A. B., and Rockmore, D. N. (2017). The Supreme Court and the judicial genre. *Ariz. L. Rev.*, 59:837.

Long, S., Tu, C., Liu, Z., and Sun, M. (2019). Automatic judgment prediction via legal reading comprehension. In *China National Conference on Chinese Computational Linguistics*, pages 558–572. Springer.

Luo, B., Feng, Y., Xu, J., Zhang, X., and Zhao, D. (2017). Learning to predict charges for criminal cases with legal basis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2727–2736, Copenhagen, Denmark. Association for Computational Linguistics.

Lupu, Y. and Voeten, E. (2012). Precedent in international courts: A network analysis of case citations by the European Court of Human Rights. *British Journal of Political Science*, 42(2):413–439.

Madsen, M. R. (2017). Rebalancing european human rights: Has the Brighton declaration engendered a new deal on human rights in europe? *Journal of International Dispute Settlement*.

Malik, V., Sanjay, R., Nigam, S. K., Ghosh, K., Guha, S. K., Bhattacharya, A., and Modi, A. (2021). ILDC for CJPE: Indian legal documents corpus for court judgment prediction and explanation. *arXiv preprint arXiv:2105.13562*.

Marković, M. and Gostojić, S. (2018). Open judicial data: A comparative analysis. *Social Science Computer Review*.

Matthews, A. A. (2017). *Connected Courts: the Diffusion of Precedent Across State Supreme Courts*. PhD thesis, The University of Iowa.

Medvedeva, M., Dam, T., Wieling, M., and Vols, M. (2021a). Automatically identifying eviction cases and outcomes within case law of Dutch courts of first instance. In *Legal Knowledge and Information Systems*, pages 13–22. IOS Press.

Medvedeva, M., Kroon, M., and Plank, B. (2017). When sparse traditional models outperform dense neural networks: the curious case of discriminating between similar languages. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 156–163.

Medvedeva, M., Üstun, A., Xu, X., Vols, M., and Wieling, M. (2021b). Automatic judgement forecasting for pending applications of the European Court of Human Rights. In *Proceedings of the Fifth Workshop on Automated Semantic Analysis of Information in Legal Text (ASAIL 2021)*.

Medvedeva, M., Vols, M., and Wieling, M. (2018). Judicial decisions of the European Court of Human Rights: Looking into the crystal ball. In *Proceedings of the Conference on Empirical Legal Studies*.

Medvedeva, M., Vols, M., and Wieling, M. (2020a). Using machine learning to predict decisions of the European Court of Human Rights. *Artificial Intelligence and Law*, 28:237–266.

Medvedeva, M., Wieling, M., and Vols, M. (2022). Rethinking the field of automatic prediction of court decisions. *Artificial Intelligence and Law*, pages 1–18.

Medvedeva, M., Xu, X., Wieling, M., and Vols, M. (2020b). Juri says: Prediction system for the European Court of Human Rights. In *Legal Knowledge and Information Systems: JURIX 2020: The Thirty-third Annual Conference, Brno, Czech Republic, December 9-11, 2020*, volume 334, page 277. IOS Press.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Mochales, R. and Moens, M.-F. (2008). Study on the structure of argumentation in case law. In *Proceedings of the 2008 Conference on Legal Knowledge and Information Systems*, pages 11–20.

Nichols, P. M. (2019). Bribing the machine: Protecting the integrity of algorithms as the revolution begins. *American Business Law Journal*, 56(4):771–814.

Niiler, E. (2019). Can AI be a fair judge in court. Estonia thinks so. *WIRED*. www.wired.com/story/can-ai-be-fair-judge-court-estonia-thinks-so/.

Novotná, T. et al. (2020). Document similarity of Czech Supreme Court Decisions. *Masaryk University Journal of Law and Technology*, 14(1):105–122.

O'Hear, M. and Wheelock, D. (2021). Life "with" or "without"? an empirical study of homicide sentencing. *Journal of Empirical Legal Studies*, 18(2):377–420.

Olsen, H. P. and Esmark, M. (2020). Needles in a haystack: using network analysis to identify cases that are cited for general principles of law by the european court of human rights. In *Computational Legal Studies*. Edward Elgar Publishing.

Olsen, H. P. and Küçüksu, A. (2017). Finding hidden patterns in ECtHR's case law: On how citation network analysis can improve our knowledge of ecthr's article 14 practice. *International Journal of Discrimination and the Law*, 17(1):4–22.

op Vollenbroek, M. B., Carlotto, T., Kreutz, T., Medvedeva, M., Pool, C., Bjerva, J., Haagsma, H., and Nissim, M. (2016). Gronup: Groningen user profiling. *CLEF (Working Notes) 2016: 846-857*.

O'Sullivan, C. and Beel, J. (2019). Predicting the outcome of judicial decisions made by the European Court of Human Rights. In *AICS 2019 - 27th AIAI Irish Conference on Artificial Intelligence and Cognitive Science*.

Otter, D. W., Medina, J. R., and Kalita, J. K. (2020). A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2):604–624.

Panagis, Y., Christensen, M. L., and Sadl, U. (2016). On top of topics: Leveraging topic modeling to study the dynamic case-law of international courts. In *JURIX*, pages 161–166.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., VanderPlas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Petrova, A., Armour, J., and Lukasiewicz, T. (2020). Extracting outcomes from appellate decisions in US State Courts. In *Legal Knowledge and Information Systems: JURIX 2020: The Thirty-third Annual Conference, Brno, Czech Republic, December 9-11, 2020*, volume 334, page 133. IOS Press.

Platt, J. et al. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.

Quemy, A. and Wrembel, R. (2020). On integrating and classifying legal text documents. In *International Conference on Database and Expert Systems Applications*, pages 385–399. Springer.

Rachlinski, J. J. and Wistrich, A. J. (2017). Judging the judiciary by the numbers: Empirical research on judges. *Annual Review of Law and Social Science*, 13:203–229.

Rangel, F. and Rosso, P. (2013). Use of language and author profiling: Identification of gender and age. *Natural Language Processing and Cognitive Science*, 177.

Remmits, Y. (2017). Finding the topics of case law: Latent dirichlet allocation on supreme court decisions.

Renberg, K. M. and Tolley, M. C. (2021). Mapping europe's cosmopolitan legal order: a network analysis of the European Court of Human Rights, the Court of Justice of the European Union, and high national courts. *Eur. J. Legal Stud.*, 13:45.

Rodrigues, C. and Campina, A. (2021). Health and human rights, ECHR and the constitution of the Portuguese republic: an interpretation. *Adjuris-Perspectives of Law and Public Administration*, 10(3):180–191.

Ruggeri, F., Lagioia, F., Lippi, M., and Torroni, P. (2022). Detecting and explaining unfairness in consumer contracts through memory networks. *Artificial Intelligence and Law*, 30(1):59–92.

Šadl, U. and Olsen, H. P. (2017). Can quantitative methods complement doctrinal legal studies? using citation network and corpus linguistic analysis to understand international courts. *Leiden Journal of International Law*, 30(2):327–349.

Salaün, O., Langlais, P., Lou, A., Westermann, H., and Benyekhlef, K. (2020). Analysis and multilabel classification of Quebec court decisions in the domain of housing law. In *International Conference on Applications of Natural Language to Information Systems*, pages 135–143. Springer.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2019). DistilBERT, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Schwartz, H. A., Eichstaedt, J. C., Kern, M. L., Dziurzynski, L., Ramones, S. M., Agrawal, M., Shah, A., Kosinski, M., Stillwell, D., Seligman, M. E., et al. (2013). Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one*, 8(9):e73791.

Sert, M. F., Yıldırım, E., and İrfan Haşlak (2021). Using Artificial Intelligence to predict decisions of the Turkish Constitutional Court. *Social Science Computer Review*.

Shaikh, R. A., Sahu, T. P., and Anand, V. (2020). Predicting outcomes of legal cases based on legal factors using classifiers. *Procedia Computer Science*, 167:2393–2402.

Sharma, R. D., Mittal, S., Tripathi, S., and Acharya, S. (2015). Using modern neural networks to predict the decisions of Supreme Court of the United States with state-of-the-art accuracy. In *International Conference on Neural Information Processing*, pages 475–483. Springer.

Shulayeva, O., Siddharthan, A., and Wyner, A. (2017). Recognizing cited facts and principles in legal judgements. *Artificial Intelligence and Law*, 25(1):107–126.

Silveira, R., Fernandes, C. G., Neto, J. A. M., Furtado, V., and Pimentel Filho, J. E. (2021). Topic modelling of legal documents via LEGAL-BERT. *Proceedings http://ceur-ws. org ISSN*, 1613:0073.

Soh, J. (2019). A network analysis of the singapore court of appeal's citations to precedent. *Singapore Academy of Law Journal*, 31(1):246–284.

Spaeth, H., Epstein, L., Ruger, T., Whittington, K., Segal, J., and Martin, A. D. (2014). Supreme Court database code book.

Stevenson, M. (2018). Assessing risk assessment in action. *Minn. L. Rev.*, 103:303.

Stranieri, A. and Zeleznikow, J. (2006). *Knowledge discovery from legal databases*, volume 69. Springer Science & Business Media.

Strickson, B. and De La Iglesia, B. (2020). Legal judgement prediction for UK courts. In *Proceedings of the 2020 The 3rd International Conference on Information Science and System*, pages 204–209.

Sulea, O.-M., Zampieri, M., Malmasi, S., Vela, M., Dinu, L. P., and Van Genabith, J. (2017). Exploring the use of text classification in the legal domain. In *In Proceedings of the 2nd Workshop on Automated Semantic Analysis of Information in Legal Texts (ASAIL 2017)*.

Şulea, O.-M., Zampieri, M., Vela, M., and van Genabith, J. (2017). Predicting the law area and decisions of French Supreme Court cases. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 716–722, Varna, Bulgaria. INCOMA Ltd.

Sun, S. and Gu, X. (2017). Word embedding dropout and variable-length convolution window in convolutional neural network for sentiment classification. In *International Conference on Artificial Neural Networks*, pages 40–48. Springer.

Tagny-Ngompé, G., Mussard, S., Zambrano, G., Harispe, S., and Montmain, J. (2020). Identification of judicial outcomes in judgments: A generalized Gini-PLS approach. *Stats*, 3(4):427–443.

Tan, H., Zhang, B., Zhang, H., and Li, R. (2020). The sentencing-element-aware model for explainable term-of-penalty prediction. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 16–27. Springer.

Tarissan, F. and Nollez-Goldbach, R. (2014). The network of the International Criminal Court decisions as a complex system. In *Iscs 2013: Interdisciplinary symposium on complex systems*, pages 255–264. Springer.

Tarissan, F. and Nollez-Goldbach, R. (2015). Temporal properties of legal decision networks: a case study from the International Criminal Court. In *28th International Conference on Legal Knowledge and Information Systems (JURIX'2015)*.

Tarissan, F. and Nollez-Goldbach, R. (2016). Analysing the first case of the international criminal court from a network-science perspective. *Journal of Complex Networks*, 4(4):616–634.

ter Haar, B. (2020). Is the CJEU discriminating in age discrimination cases? *Erasmus L. Rev.*, 13:78.

Thomsen, F. (forthcoming). Iudicium ex machinae – the ethical challenges of automated decision-making in criminal sentencing. In Roberts, J. and Ryberg, J., editors, *Principled Sentencing and Artificial Intelligence*. Oxford: Oxford University Press.

Trompper, M. and Winkels, R. (2016). Automatic assignment of section structure to texts of Dutch court judgments. In *Legal knowledge and information systems*, pages 167–172. IOS Press.

Vacek, T. and Schilder, F. (2017). A sequence approach to case outcome detection. In *Proceedings of the 16th edition of the International Conference on Artificial Intelligence and Law*, pages 209–215.

van Dijck, G. (2018). Victim-oriented tort law in action: An empirical examination of catholic church sexual abuse cases. *Journal of Empirical Legal Studies*, 15(1):126–164.

Van Hoecke, M. (2011). Foreword in 'methodologies of legal research'. *European Academy of Legal Theory Series*, pages I–IX.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Vazirgiannis, M., Boniol, P., Panagopoulos, G., Xypolopoulos, C., Rajaa, E.-H., and Amariles, D. R. (2020). Performance in the courtroom: Automated processing and visualization of appeal court decisions in france. *Revista Eletrônica da Faculdade de Direito de Franca*, 15(2).

Verbruggen, P. (2021). *Methoden van systematische rechtspraakanalyse: Tussen juridische dogmatiek en data science*. Boom juridisch.

Virtucio, M. B. L., Aborot, J. A., Abonita, J. K. C., Avinante, R. S., Copino, R. J. B., Neverida, M. P., Osiana, V. O., Peramo, E. C., Syjuco, J. G., and Tan, G. B. A. (2018). Predicting decisions of the Philippine Supreme Court using natural language processing and machine learning. In *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, volume 2, pages 130–135. IEEE.

Visentin, A., Nardotto, A., and O'Sullivan, B. (2019). Predicting judicial decisions: A statistically rigorous approach and a new ensemble classifier. In *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 1820–1824. IEEE.

Vols, M. (2018). Evictions in the Netherlands. In *Loss of Homes and Evictions across Europe*, page 214–238. Edward Elgar Publishing.

Vols, M. (2019). European law and evictions: Property, proportionality and vulnerable people. *European Review of Private Law*, 27(4).

Vols, M. (2021). *Legal Research. One Hundred Questions and Answers*. Eleven.

Vols, M. and Jacobs, J. (2017). Juristen als rekenmeesters: Over de kwantitatieve analyse van jurisprudentie. *P. A. J. van den Berg, & G. Molier (Eds.)*, pages 89–104.

Vols, M., Tassenaar, P., and Jacobs, J. (2015). Anti-social behaviour and european protection against eviction. *International Journal of Law in the Built Environment*, 7(2):148–161.

Waltl, B., Bonczek, G., Scepankova, E., Landthaler, J., and Matthes, F. (2017). Predicting the outcome of appeal decisions in Germany's tax law. In *International Conference on Electronic Participation*, pages 89–99. Springer.

Whalen, R. (2016). Legal networks: the promises and challenges of legal network analysis. *Mich. St. L. Rev.*, page 539.

White, R. C. and Boussiakou, I. (2009). Separate opinions in the European Court of Human Rights. *Human Rights Law Review*, 9(1):37–60.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Wyner, A., Mochales-Palau, R., Moens, M.-F., and Milward, D. (2010). Approaches to text mining arguments from legal cases. In *Semantic processing of legal texts*, pages 60–79. Springer.

Xu, H., Šavelka, J., and Ashley, K. D. (2020). Using argument mining for legal text summarization. In *Legal Knowledge and Information Systems*, pages 184–193. IOS Press.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Ye, H., Jiang, X., Luo, Z., and Chao, W. (2018). Interpretable charge predictions for criminal cases: Learning to generate court views from fact descriptions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1854–1864, New Orleans, Louisiana. Association for Computational Linguistics.

Završnik, A. (2020). Criminal justice, artificial intelligence systems, and human rights. In *ERA Forum*, volume 20, pages 567–583. Springer.

Zhang, A. H., Liu, J., and Garoupa, N. (2018). Judging in europe: Do legal traditions matter? *Journal of Competition Law & Economics*, 14(1):144–178.

Zhong, H., Guo, Z., Tu, C., Xiao, C., Liu, Z., and Sun, M. (2018a). Legal judgment prediction via topological learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3540–3549.

Zhong, H., Xiao, C., Guo, Z., Tu, C., Liu, Z., Sun, M., Feng, Y., Han, X., Hu, Z., Wang, H., et al. (2018b). Overview of CAIL2018: Legal judgment prediction competition. *arXiv preprint arXiv:1810.05851*.

Summary

In this dissertation, we address the potential of using language analysis and automatic information extraction to facilitate statistical research in the legal domain. More specifically, we demonstrate and discuss the possibilities of natural language processing (NLP) techniques for the automatic prediction of judicial decisions, as well as their limitations.

Our experience shows that, in the majority of cases, true interdisciplinary collaborations are essential to producing technological systems that provide solutions to existing issues in the legal domain, therefore we have put extra effort into making sure that this dissertation is accessible for both legal professionals and NLP specialists. We chose to do so, because we believe that this interdisciplinary field should always be a collaboration between the two disciplines, where both sides play different roles but are familiar with each other's field.

Part I, *Background and definitions*, discusses previous work in natural legal language processing, and suggests new terminology for the field of predicting court decisions. Specifically, **Chapter 2** discusses a range of previous quantitative (non-machine learning) research conducted on legal data, this includes different methods, as well as available legal datasets.

Chapter 3 explains the machine learning terminology applicable to the field, to support the reader in understanding the methodology presented in previous work and our own experiments.

Chapter 4 provides a review of the main research in predicting court decisions. In this chapter, we discuss why existing terminology in the field is problematic, and suggest new terminology that better reflects the tasks that have until now been generalised under the term 'predicting court decisions'. Specifically, we define and discuss the differences between outcome identification, outcome-based judgement categorisation, and outcome forecasting, and review how various studies fall into these categories.

We present outcome identification as the task of identifying a verdict in the full text of a published judgement, judgement categorisation as the task of categorising documents based on the outcome, and outcome forecasting as the task of predicting the future decisions of a particular court. Through these definitions we discuss how important it is to understand the legal data one works with, in order to determine which task can be performed. After reviewing existing literature it becomes clear that the majority of papers that claim to predict court decisions are unable to make predictions about the cases that have not been judges yet. Finally, we reflect on the needs of the legal discipline regarding the analysis of court judgements.

Part II, *Experiments*, describes our experiments as three sub-tasks that we defined in Chapter 4. **Chapter 5** describes a method for identifying eviction-related cases and their outcomes, within all the Dutch judiciary case law available online. In this chapter we suggest a practical method for collecting legal datasets on a specific topic, using machine learning. To do so, we performed two experiments. The first focused on identifying judgements related to eviction, which were able to do with with 88% accuracy. The second experiment focused on identifying the outcome of the cases in the judgements (eviction vs. dismissal of the landlord's claim). We were able to identify eviction related cases, for which we achieved 92% accuracy. We have also found that a keyword-based approach is not straightforward when the information is provided as natural text, and thus, machine learning might often yield better results. In the process of conducting the experiments for this study, we have created a manually annotated dataset of eviction related judgements and their outcomes.

Chapter 6 focuses on NLP methodology for judgement categorisation, to identify factors that may result in finding patterns, as well as a better understanding of judicial decision-making, using the European Court of Human Rights as an example. With an average accuracy of 75% in categorising the judgements, according to whether or not there was a violation of 9 articles of the European Convention on Human Rights, our approach highlights the potential of machine learning within the legal domain. We also show, however, that categorising future cases based on past cases negatively impacts performance (with an average accuracy range from 58% to 68%). Furthermore, we demonstrate that a relatively high classification performance (average accuracy of 65%) can be achieved, when categorising judgements based only on the surnames of judges trying case, and discuss how this does not indicate judicial bias since the judges are not assigned the cases randomly.

Chapter 7 is focused on forecasting the (future) decisions of the European Court of Human Rights, using documents published by the court (sometimes) years before a decision is made. To address this task, we released an initial benchmark dataset, consisting of documents from the European Court of Human Rights. The dataset included raw data, as well as pre-processed text from final judgements, admissibility decisions, and communicated cases. The latter are published by the court for pending applications before the case is judged, allowing judgements for pending cases to be forecasted. In this chapter we established a baseline for this task (0.57-0.67 F1-score), and showed that it is much harder than simply categorising judgements (0.56-0.92 F1-score) We have also found that performance of machine learning systems in the categorisation task can sometimes fluctuate throughout the years, and therefore one should not expect that the results will always remain consistent.

Chapter 8 describes an online interface that incorporates the experiment of Chapter 7. The web platform JURI SAYS, automatically predicts European Court of Human Rights decisions based on communicated cases, similar to the system described in Chapter 7. The platform is available at `https://jurisays.com`, and it shows predictions of the court decisions, compared to its actual decisions. It is automatically updated every month, by including predictions for all new cases. Additionally, the system highlights the sentences and paragraphs that are most important for forecasting (i.e. violation vs. no violation of human rights).

In **Part III**, *Ethical consideration and conclusions*, we discuss ethical concerns associated with systems that deal with case law, as well as discuss the overall findings of our work and draw conclusions. In **Chapter 9** we point out that while the ethical concerns regarding predicting court decisions has already been widely discussed in the legal community, we hope to have introduced a perspective which acknowledges the technological limitations of such systems. We argue that the use of such tools in these situations should be limited, due to the inherent status quo bias and dangers of reverse-engineering. We discuss that these issues already exist in judicial systems, before any machine learning tools are used, but how introducing these tools might exacerbate such issues by creating a situation where the system can be tested an unlimited amount times until a desired result is achieved.

In **Chapter 10** we discuss how all the sub-tasks described in the dissertation are only first steps in a more elaborate legal analysis. The identification results can be used to supplement datasets and benefit research on a specific topic. The categorisation experiments may point towards particular patterns within judge-

ments, but they still need to be analysed and made sense of from a legal perspective. Similarly, if the forecasting results are used to make estimates about the future, the systems must become easier to explain, so that the estimates are understandable. We also discuss how the models built for one court's judgements might not work well on the other, due to specific structure of the text, and even more so due to dependency on language of the majority of NLP techniques. We also point out that the case law data is imperfect due to selection bias. Firstly, due to selective publishing of judgements by many courts, and secondly, due to some cases not going through the some of the steps of the judicial process that they could potentially go through. For instance, when using the decisions of the court of appeal, one relies on the fact that someone decided to proceed to appeal in the supreme court, and that the supreme court has not dismissed the case. Cases that were not moved to the supreme court, but which would have been won, are not part of the dataset, as these are unknown. We then discuss how an explainable system can help one understand which parameters it relies on, and therefore somewhat alleviate the issue.

In conclusion, in this dissertation we have tried to show that the field of predicting court decisions shows potential with regard to the automation of legal processes and legal analysis. We have also demonstrated that there are many limitations to what today's systems are able to do.

Samenvatting

In dit proefschrift gaan we in op het potentieel van het gebruik van taalanalyse en automatische informatie-extractie om statistisch onderzoek in het juridische domein te vergemakkelijken. Meer specifiek demonstreren en bespreken we de mogelijkheden van natural language processing (NLP) technieken voor de automatische voorspelling van rechterlijke beslissingen, evenals hun beperkingen.

Onze ervaring leert dat in de meeste gevallen echte interdisciplinaire samenwerking essentieel is voor het produceren van technologische systemen die oplossingen bieden voor bestaande problemen in het juridische domein. Om deze reden hebben we extra moeite gedaan om ervoor te zorgen dat dit proefschrift toegankelijk is voor zowel juristen als specialisten op het gebied van NLP. We hebben hiervoor gekozen omdat we van mening zijn dat dit interdisciplinaire vakgebied altijd een samenwerking tussen de twee disciplines zou moeten zijn, waarbij beide partijen verschillende rollen spelen maar wel bekend zijn met elkaars vakgebied.

Deel I, *Background and definitions*, bespreekt eerder werk in natuurlijke juridische taalverwerking, en stelt nieuwe terminologie voor op het gebied van het voorspellen van rechterlijke beslissingen. In het bijzonder bespreekt **hoofdstuk 2** een reeks van eerdere kwantitatieve (niet-machine learning) onderzoeken uitgevoerd op juridische data, die gebruik maken van verschillende methoden, evenals verschillende beschikbare datasets.

In **hoofdstuk 3** wordt de terminologie van machine learning uitgelegd, om de lezer te helpen de methodologie van eerder werk en onze eigen experimenten te begrijpen.

Hoofdstuk 4 geeft een overzicht van het belangrijkste onderzoek naar het voorspellen van rechterlijke beslissingen. In dit hoofdstuk bespreken we waarom de bestaande terminologie op dit gebied problematisch is, en stellen we

nieuwe terminologie voor die beter de taken beschrijft die tot nu toe werden veralgemeend onder de noemer 'predicting court decisions'. In het bijzonder definiëren en bespreken we de verschillen tussen uitkomstidentificatie, op uitkomsten gebaseerde categorisering van beslissingen, en voorspelling van uitkomsten, en bekijken we hoe verschillende studies in deze categorieën vallen.

Wij presenteren uitkomstidentificatie als de taak om een uitspraak te identificeren in de volledige tekst van een gepubliceerde beslissing, beslissingcategorisatie als de taak om documenten te categoriseren op basis van de uitkomst, en uitkomstvoorspelling als de taak om de toekomstige uitspraken van een bepaalde rechtbank te voorspellen. Aan de hand van deze definities bespreken we hoe belangrijk het is om de juridische gegevens waarmee men werkt te begrijpen, om te kunnen bepalen welke taak kan worden uitgevoerd. Na bestudering van de bestaande literatuur wordt duidelijk dat de meeste papers die beweren rechterlijke uitspraken te voorspellen, niet in staat zijn voorspellingen te doen over de zaken die nog niet door rechters zijn behandeld. Ten slotte denken we na over de behoeften van de juridische discipline met betrekking tot de analyse van rechterlijke uitspraken.

Deel II, *Experiments*, beschrijft onze experimenten in de vorm van drie subtaken die we in hoofdstuk 4 hebben gedefinieerd. **Hoofdstuk 5** beschrijft een methode voor het identificeren van aan huisuitzetting gerelateerde zaken en hun uitkomsten, binnen alle Nederlandse rechtspraak die online beschikbaar is. In dit hoofdstuk stellen we een praktische methode voor om juridische datasets over een specifiek onderwerp te verzamelen, met behulp van machine learning. Daartoe hebben wij twee experimenten uitgevoerd. Het eerste experiment was gericht op het identificeren van uitspraken met betrekking tot uitzetting, wat met 88% nauwkeurigheid lukte. Het tweede experiment was gericht op het identificeren van de uitkomst van de zaken in de vonnissen (ontruiming vs. verwerping van de vordering van de verhuurder). We waren in staat om ontruimingszaken te identificeren, waarbij we een nauwkeurigheid van 92% behaalden. Wij hebben ook vastgesteld dat een aanpak op basis van keywords niet eenvoudig is wanneer de informatie als natuurlijke tekst wordt verstrekt, en dat machine learning dus vaak betere resultaten kan opleveren. Tijdens het uitvoeren van de experimenten voor deze studie hebben we een manueel geannoteerde dataset van uitzettingsgerelateerde uitspraken en hun resultaten gecreëerd.

Hoofdstuk 6 richt zich op NLP methodologie voor het categoriseren van beslissingen, om factoren te identificeren die kunnen leiden tot het vinden van patronen, alsook tot een beter begrip van de rechterlijke besluitvorming, met het

Europees Hof voor de Rechten van de Mens als voorbeeld. Met een gemiddelde nauwkeurigheid van 75% bij het categoriseren van de beslissingen, op basis van het al dan niet schenden van 9 artikelen van het Europees Verdrag voor de Rechten van de Mens, benadrukt onze aanpak het potentieel van machine learning binnen het juridische domein. We tonen echter ook aan dat het categoriseren van toekomstige zaken op basis van zaken uit het verleden een negatieve invloed heeft op de prestaties (met een gemiddelde nauwkeurigheid die varieert van 58% tot 68%). Verder tonen we aan dat een relatief hoge classificatieprestatie (gemiddelde nauwkeurigheid van 65%) kan worden bereikt wanneer uitspraken worden gecategoriseerd op basis van slechts de achternaam van de rechters die de zaak behandelen, en bespreken we hoe dit niet duidt op rechterlijke vooringenomenheid omdat de rechters de zaken niet willekeurig toegewezen krijgen.

Hoofdstuk 7 richt zich op het voorspellen van de (toekomstige) uitspraken van het Europese Hof voor de Rechten van de Mens, gebruikmakend van documenten die het Hof (soms) jaren voordat een uitspraak wordt gedaan, publiceert. Om deze taak uit te voeren, hebben we een eerste benchmark dataset vrijgegeven, bestaande uit documenten van het Europees Hof voor de Rechten van de Mens. De dataset bevat ruwe gegevens, maar ook voorbewerkte tekst van definitieve vonnissen, ontvankelijkheidsbeslissingen, en *communicated cases*. De laatstgenoemde worden door het hof gepubliceerd voor aanhangige zaken voordat de zaak is beoordeeld. Wij gebruiken deze communicated cases om toekomstige uitspraken voor aanhangige zaken voorspellen. In dit hoofdstuk hebben we een baseline voor deze taak vastgesteld (0.57-0.67 F1-score), en aangetoond dat deze taak veel moeilijker is dan het categoriseren van beslissingen (0.56-0.92 F1-score). We hebben ook vastgesteld dat de prestaties van machine learning systemen in de categorisatietaak soms door de jaren heen kunnen schommelen, en dat men dus niet moet verwachten dat de resultaten altijd consistent zullen blijven.

Hoofdstuk 8 beschrijft een online interface waarin het experiment van hoofdstuk 7 is verwerkt. Het webplatform JURI SAYS voorspelt automatisch uitspraken van het Europees Hof voor de Rechten van de Mens op basis van communicated cases, vergelijkbaar met het systeem dat beschreven wordt in hoofdstuk 7. Het platform is beschikbaar op https://jurisays.com en toont voorspellingen van uitspraken van het Hof, vergeleken met de werkelijke uitspraken. Het wordt elke maand automatisch bijgewerkt met voorspellingen voor alle nieuwe zaken. Bovendien benadrukt het systeem de zinnen en paragrafen die het belangrijkst zijn voor de voorspellingen (d.w.z. schending vs. geen schending van

de mensenrechten).

In deel III, *Ethical consideration and conclusions*, bespreken we ethische overwegingen en de algemene bevindingen van ons werk, en trekken we conclusies. In **hoofdstuk 9** wijzen we erop dat, hoewel de ethische bezwaren met betrekking tot het voorspellen van rechterlijke uitspraken reeds uitgebreid besproken zijn in de juridische gemeenschap, wij hopen een perspectief te hebben geïntroduceerd dat de technologische beperkingen van dergelijke systemen erkent. Wij stellen dat het gebruik van dergelijke hulpmiddelen in deze situaties beperkt zou moeten zijn, wegens de inherente status quo bias en de gevaren van reverse-engineering. We bespreken dat deze problemen al bestaan in gerechtelijke systemen voordat er machine learning hulpmiddelen worden gebruikt, maar dat de invoering van deze hulpmiddelen deze problemen zou kunnen verergeren door een situatie te creëren waarin het systeem een onbeperkt aantal keren kan worden getest tot het gewenste resultaat is bereikt.

In **hoofdstuk 10** bespreken we hoe alle in het proefschrift beschreven deeltaken slechts de eerste stappen zijn in een meer uitgewerkte juridische analyse. De identificatieresultaten kunnen worden gebruikt om datasets aan te vullen en het onderzoek naar een specifiek onderwerp ten goede komen. De categorisatie-experimenten kunnen wijzen op bepaalde patronen binnen beslissingen, maar die moeten nog worden geanalyseerd en vanuit een juridisch perspectief worden geïnterpreteerd. Ook als de voorspellingsresultaten worden gebruikt om ramingen over de toekomst te maken, moeten de systemen gemakkelijker uit te leggen zijn, zodat de ramingen begrijpelijk zijn. Wij bespreken ook hoe de modellen die zijn gebouwd voor de uitspraken van de ene rechtbank misschien niet goed werken voor de andere, vanwege de specifieke structuur van de tekst, en nog meer vanwege de afhankelijkheid van taal van de meeste NLP-technieken. Wij wijzen er ook op dat de juridische datasets onvolmaakt zijn als gevolg van selectiebias. Ten eerste door de selectieve publicatie van beslissingen door veel rechtbanken, en ten tweede door het feit dat sommige zaken niet alle stappen van de gerechtelijke procedure doorlopen die ze potentieel zouden kunnen doorlopen. Wanneer men bijvoorbeeld gebruik maakt van de beslissingen van het gerechtshof, vertrouwt men op het feit dat iemand heeft besloten om in beroep te gaan bij het hogere gerechtshof, en dat het hogere gerechtshof de zaak niet heeft verworpen. Zaken die niet zijn doorverwezen naar het hogere gerechtshof, maar die wel zouden zijn gewonnen, maken geen deel uit van de dataset, omdat deze onbekend zijn. Vervolgens bespreken we hoe een verklaarbaar systeem kan helpen te begrijpen op welke parameters het zich baseert, en zo het probleem enigszins kan

verlichten.

Concluderend hebben we in dit proefschrift geprobeerd aan te tonen dat het gebied van het voorspellen van rechterlijke uitspraken potentieel vertoont met betrekking tot de automatisering van juridische processen en juridische analyse. We hebben ook aangetoond dat er veel beperkingen zijn aan waartoe de huidige systemen in staat zijn.

Groningen Dissertations in Linguistics (GRODIL)

---

1. Henriëtte de Swart (1991). *Adverbs of Quantification: A Generalized Quantifier Approach*.
2. Eric Hoekstra (1991). *Licensing Conditions on Phrase Structure*.
3. Dicky Gilbers (1992). *Phonological Networks. A Theory of Segment Representation*.
4. Helen de Hoop (1992). *Case Configuration and Noun Phrase Interpretation*.
5. Gosse Bouma (1993). *Nonmonotonicity and Categorial Unification Grammar*.
6. Peter I. Blok (1993). *The Interpretation of Focus*.
7. Roelien Bastiaanse (1993). *Studies in Aphasia*.
8. Bert Bos (1993). *Rapid User Interface Development with the Script Language Gist*.
9. Wim Kosmeijer (1993). *Barriers and Licensing*.
10. Jan-Wouter Zwart (1993). *Dutch Syntax: A Minimalist Approach*.
11. Mark Kas (1993). *Essays on Boolean Functions and Negative Polarity*.
12. Ton van der Wouden (1994). *Negative Contexts*.
13. Joop Houtman (1994). *Coordination and Constituency: A Study in Categorial Grammar*.
14. Petra Hendriks (1995). *Comparatives and Categorial Grammar*.
15. Maarten de Wind (1995). *Inversion in French*.
16. Jelly Julia de Jong (1996). *The Case of Bound Pronouns in Peripheral Romance*.
17. Sjoukje van der Wal (1996). *Negative Polarity Items and Negation: Tandem Acquisition*.
18. Anastasia Giannakidou (1997). *The Landscape of Polarity Items*.
19. Karen Lattewitz (1997). *Adjacency in Dutch and German*.
20. Edith Kaan (1997). *Processing Subject-Object Ambiguities in Dutch*.
21. Henny Klein (1997). *Adverbs of Degree in Dutch*.
22. Leonie Bosveld-de Smet (1998). *On Mass and Plural Quantification: The case of French 'des'/ 'du'-NPs*.
23. Rita Landeweerd (1998). *Discourse semantics of perspective and temporal structure*.
24. Mettina Veenstra (1998). *Formalizing the Minimalist Program*.
25. Roel Jonkers (1998). *Comprehension and Production of Verbs in aphasic Speakers*.
26. Erik F. Tjong Kim Sang (1998). *Machine Learning of Phonotactics*.
27. Paulien Rijkhoek (1998). *On Degree Phrases and Result Clauses*.
28. Jan de Jong (1999). *Specific Language Impairment in Dutch: Inflectional Morphology and Argument Structure.*
29. H. Wee (1999). *Definite Focus.*
30. Eun-Hee Lee (2000). *Dynamic and Stative Information in Temporal Reasoning: Korean tense and aspect in discourse.*
31. Ivilin P. Stoianov (2001). *Connectionist Lexical Processing.*

32. Klarien van der Linde (2001). *Sonority substitutions.*
33. Monique Lamers (2001). *Sentence processing: using syntactic, semantic, and thematic information*.
34. Shalom Zuckerman (2001). *The Acquisition of "Optional" Movement*.
35. Rob Koeling (2001). *Dialogue-Based Disambiguation: Using Dialogue Status to Improve Speech Understanding*.
36. Esther Ruigendijk (2002). *Case assignment in Agrammatism: a cross-linguistic study.*
37. Tony Mullen (2002). *An Investigation into Compositional Features and Feature Merging for Maximum Entropy-Based Parse Selection.*
38. Nanette Bienfait (2002). *Grammatica-onderwijs aan allochtone jongeren.*
39. Dirk-Bart den Ouden (2002). *Phonology in Aphasia: Syllables and segments in level-specific deficits.*
40. Rienk Withaar (2002). *The Role of the Phonological Loop in Sentence Comprehension.*
41. Kim Sauter (2002). *Transfer and Access to Universal Grammar in Adult Second Language Acquisition.*
42. Laura Sabourin (2003). *Grammatical Gender and Second Language Processing: An ERP Study.*
43. Hein van Schie (2003). *Visual Semantics.*
44. Lilia Schürcks-Grozeva (2003). *Binding and Bulgarian.*
45. Stasinos Konstantopoulos (2003). *Using ILP to Learn Local Linguistic* S*tructures*.
46. Wilbert Heeringa (2004). *Measuring Dialect Pronunciation Differences using Levenshtein Distance*.
47. Wouter Jansen (2004). *Laryngeal Contrast and Phonetic Voicing: A Laboratory Phonology*.
48. Judith Rispens (2004). *Syntactic and phonological processing in developmental dyslexia*.
49. Danielle Bougaïré (2004). *L'approche communicative des campagnes de sensibilisation en santé publique au Burkina Faso: Les cas de la planification familiale, du sida et de l'excision*.
50. Tanja Gaustad (2004). *Linguistic Knowledge and Word Sense Disambiguation.*
51. Susanne Schoof (2004). *An HPSG Account of Nonfinite Verbal Complements in Latin*.
52. M. Begoña Villada Moirón (2005). *Data-driven identification of fixed expressions and their modifiability.*
53. Robbert Prins (2005). *Finite-State Pre-Processing for Natural Language Analysis.*
54. Leonoor van der Beek (2005) *Topics in Corpus-Based Dutch Syntax.*
55. Keiko Yoshioka (2005). *Linguistic and gestural introduction and tracking of referents in L1 and L2 discourse.*
56. Sible Andringa (2005). *Form-focused instruction and the development of second language proficiency*.
57. Joanneke Prenger (2005). *Taal telt! Een onderzoek naar de rol van taalvaardigheid en tekstbegrip in het realistisch wiskundeonderwijs.*

58. Neslihan Kansu-Yetkiner (2006). *Blood, Shame and Fear: Self-Presentation Strategies of Turkish Women's Talk about their Health and Sexuality*.
59. Mónika Z. Zempléni (2006). *Functional imaging of the hemispheric contribution to language processing.*
60. Maartje Schreuder (2006). *Prosodic Processes in Language and Music.*
61. Hidetoshi Shiraishi (2006). *Topics in Nivkh Phonology.*
62. Tamás Biró (2006). *Finding the Right Words: Implementing Optimality Theory with Simulated Annealing.*
63. Dieuwke de Goede (2006). *Verbs in Spoken Sentence Processing: Unraveling the Activation Pattern of the Matrix Verb.*
64. Eleonora Rossi (2007). *Clitic production in Italian agrammatism.*
65. Holger Hopp (2007). *Ultimate Attainment at the Interfaces in Second Language Acquisition: Grammar and Processing.*
66. Gerlof Bouma (2008). *Starting a Sentence in Dutch: A corpus study of subject- and object-fronting.*
67. Julia Klitsch (2008). *Open your eyes and listen carefully. Auditory and audiovisual speech perception and the McGurk effect in Dutch speakers with and without aphasia.*
68. Janneke ter Beek (2008). *Restructuring and Infinitival Complements in Dutch.*
69. Jori Mur (2008). *Off-line Answer Extraction for Question Answering.*
70. Lonneke van der Plas (2008). *Automatic Lexico-Semantic Acquisition for Question Answering.*
71. Arjen Versloot (2008). *Mechanisms of Language Change: Vowel reduction in 15th century West Frisian.*
72. Ismail Fahmi (2009). *Automatic term and Relation Extraction for Medical Question Answering System.*
73. Tuba Yarbay Duman (2009). *Turkish Agrammatic Aphasia: Word Order, Time Reference and Case.*
74. Maria Trofimova (2009). *Case Assignment by Prepositions in Russian Aphasia.*
75. Rasmus Steinkrauss (2009). *Frequency and Function in WH Question Acquisition. A Usage-Based Case Study of German L1 Acquisition.*
76. Marjolein Deunk (2009). *Discourse Practices in Preschool. Young Children's Participation in Everyday Classroom Activities.*
77. Sake Jager (2009). *Towards ICT-Integrated Language Learning: Developing an Implementation Framework in terms of Pedagogy, Technology and Environment.*
78. Francisco Dellatorre Borges (2010). *Parse Selection with Support Vector Machines.*
79. Geoffrey Andogah (2010). *Geographically Constrained Information Retrieval.*
80. Jacqueline van Kruiningen (2010). *Onderwijsontwerp als conversatie. Probleemoplossing in interprofessioneel overleg.*

81. Robert G. Shackleton (2010). *Quantitative Assessment of English-American Speech Relationships*.
82. Tim Van de Cruys (2010). *Mining for Meaning: The Extraction of Lexico-semantic Knowledge from Text.*
83. Therese Leinonen (2010). *An Acoustic Analysis of Vowel Pronunciation in Swedish Dialects.*
84. Erik-Jan Smits (2010). *Acquiring Quantification. How Children Use Semantics and Pragmatics to Constrain Meaning*.
85. Tal Caspi (2010). *A Dynamic Perspective on Second Language Development.*
86. Teodora Mehotcheva (2010). *After the fiesta is over. Foreign language attrition of Spanish in Dutch and German Erasmus Student.*
87. Xiaoyan Xu (2010). *English language attrition and retention in Chinese and Dutch university students.*
88. Jelena Prokić (2010). *Families and Resemblances*.
89. Radek Šimík (2011). *Modal existential wh-constructions*.
90. Katrien Colman (2011). *Behavioral and neuroimaging studies on language processing in Dutch speakers with Parkinson's disease.*
91. Siti Mina Tamah (2011). *A Study on Student Interaction in the Implementation of the Jigsaw Technique in Language Teaching.*
92. Aletta Kwant (2011).*Geraakt door prentenboeken*. *Effecten van het gebruik van prentenboeken op de sociaal-emotionele ontwikkeling van kleuters.*
93. Marlies Kluck (2011). *Sentence amalgamation.*
94. Anja Schüppert (2011). *Origin of asymmetry: Mutual intelligibility of spoken Danish and Swedish.*
95. Peter Nabende (2011). *Applying Dynamic Bayesian Networks in Transliteration Detection and Generation.*
96. Barbara Plank (2011). *Domain Adaptation for Parsing.*
97. Cagri Coltekin (2011).*Catching Words in a Stream of Speech: Computational simulations of segmenting transcribed child-directed speech.*
98. Dörte Hessler (2011). *Audiovisual Processing in Aphasic and Non-Brain-Damaged Listeners: The Whole is More than the Sum of its Parts.*
99. Herman Heringa (2012). *Appositional constructions.*
100. Diana Dimitrova (2012). *Neural Correlates of Prosody and Information Structure.*
101. Harwintha Anjarningsih (2012). *Time Reference in Standard Indonesian Agrammatic Aphasia*.
102. Myrte Gosen (2012). *Tracing learning in interaction. An analysis of shared reading of picture books at kindergarten.*

103. Martijn Wieling (2012). *A Quantitative Approach to Social and Geographical Dialect Variation*.
104. Gisi Cannizzaro (2012). *Early word order and animacy*.
105. Kostadin Cholakov (2012). *Lexical Acquisition for Computational Grammars. A Unified Model.*
106. Karin Beijering (2012). *Expressions of epistemic modality in Mainland Scandinavian. A study into the lexicalization-grammaticalization-pragmaticalization interface.*
107. Veerle Baaijen (2012). *The development of understanding through writing.*
108. Jacolien van Rij (2012). *Pronoun processing: Computational, behavioral, and psychophysiological studies in children and adults.*
109. Ankelien Schippers (2012). *Variation and change in Germanic long-distance dependencies.*
110. Hanneke Loerts (2012).*Uncommon gender: Eyes and brains, native and second language learners, & grammatical gender.*
111. Marjoleine Sloos (2013). *Frequency and phonological grammar: An integrated approach. Evidence from German, Indonesian, and Japanese*.
112. Aysa Arylova. (2013) *Possession in the Russian clause. Towards dynamicity in syntax.*
113. Daniël de Kok (2013). *Reversible Stochastic Attribute-Value Grammars.*
114. Gideon Kotzé (2013). *Complementary approaches to tree alignment: Combining statistical and rule-based methods.*
115. Fridah Katushemererwe (2013). *Computational Morphology and Bantu Language Learning: an Implementation for Runyakitara*.
116. Ryan C. Taylor (2013). *Tracking Referents: Markedness, World Knowledge and Pronoun Resolution.*
117. Hana Smiskova-Gustafsson (2013). *Chunks in L2 Development: A Usage-based Perspective.*
118. Milada Walková (2013). *The aspectual function of particles in phrasal verbs.*
119. Tom O. Abuom (2013). *Verb and Word Order Deficits in Swahili-English bilingual agrammatic speakers.*
120. Gülsen Yılmaz (2013). *Bilingual Language Development among the First Generation Turkish Immigrants in the Netherlands.*
121. Trevor Benjamin (2013). *Signaling Trouble: On the linguistic design of other-initiation of repair in English conversation.*
122. Nguyen Hong Thi Phuong (2013). *A Dynamic Usage-based Approach to Second Language Teaching.*
123. Harm Brouwer (2014). *The Electrophysiology of Language Comprehension: A Neurocomputational Model.*
124. Kendall Decker (2014). *Orthography Development for Creole Languages.*

125. Laura S. Bos (2015). *The Brain, Verbs, and the Past: Neurolinguistic Studies on Time Reference.*
126. Rimke Groenewold (2015). *Direct and indirect speech in aphasia: Studies of spoken discourse production and comprehension.*
127. Huiping Chan (2015). *A Dynamic Approach to the Development of Lexicon and Syntax in a Second Language.*
128. James Griffiths (2015). *On appositives*.
129. Pavel Rudnev (2015). *Dependency and discourse-configurationality: A study of Avar.*
130. Kirsten Kolstrup (2015). *Opportunities to speak. A qualitative study of a second language in use.*
131. Güliz Güneş (2015). *Deriving Prosodic structures.*
132. Cornelia Lahmann (2015). *Beyond barriers. Complexity, accuracy, and fluency in long-term L2 speakers' speech.*
133. Sri Wachyunni (2015). *Scaffolding and Cooperative Learning: Effects on Reading Comprehension and Vocabulary Knowledge in English as a Foreign Language.*
134. Albert Walsweer (2015). *Ruimte voor leren. Een etnogafisch onderzoek naar het verloop van een interventie gericht op versterking van het taalgebruik in een knowledge building environment op kleine Friese basisscholen.*
135. Aleyda Lizeth Linares Calix (2015). *Raising Metacognitive Genre Awareness in L2 Academic Readers and Writers.*
136. Fathima Mufeeda Irshad (2015). *Second Language Development through the Lens of a Dynamic Usage-Based Approach.*
137. Oscar Strik (2015). *Modelling analogical change. A history of Swedish and Frisian verb inflection.*
138. He Sun (2015). *Predictors and stages of very young child EFL learners' English development in China.*
139 Marieke Haan (2015). *Mode Matters*. *Effects of survey modes on participation and answering behavior*.
140. Nienke Houtzager (2015). *Bilingual advantages in middle-aged and elderly populations.*
141. Noortje Joost Venhuizen (2015). *Projection in Discourse: A data-driven formal semantic analysis.*
142. Valerio Basile (2015). *From Logic to Language: Natural Language Generation from Logical Forms*.
143. Jinxing Yue (2016). *Tone-word Recognition in Mandarin Chinese: Influences of lexical-level representations.*
144. Seçkin Arslan (2016). *Neurolinguistic and Psycholinguistic Investigations on Evidentiality in Turkish.*

145. Rui Qin (2016). *Neurophysiological Studies of Reading Fluency. Towards Visual and Auditory Markers of Developmental Dyslexia.*
146. Kashmiri Stec (2016). *Visible Quotation: The Multimodal Expression of Viewpoint*.
147. Yinxing Jin (2016). *Foreign language classroom anxiety: A study of Chinese university students of Japanese and English over time.*
148. Joost Hurkmans (2016). *The Treatment of Apraxia of Speech. Speech and Music Therapy, an Innovative Joint Effort.*
149. Franziska Köder (2016). *Between direct and indirect speech: The acquisition of pronouns in reported speech.*
150. Femke Swarte (2016). *Predicting the mutual intelligibility of Germanic languages from linguistic and extra-linguistic factors.*
151. Sanne Kuijper (2016). *Communication abilities of children with ASD and ADHD. Production, comprehension, and cognitive mechanisms.*
152. Jelena Golubović (2016). *Mutual intelligibility in the Slavic language area*.
153. Nynke van der Schaaf (2016). *"Kijk eens wat ik kan!" Sociale praktijken in de interactie tussen kinderen van 4 tot 8 jaar in de buitenschoolse opvang.*
154. Simon Šuster (2016). *Empirical studies on word representations.*
155. Kilian Evang (2016). *Cross-lingual Semantic Parsing with Categorial Grammars.*
156. Miren Arantzeta Pérez (2017). *Sentence comprehension in monolingual and bilingual aphasia: Evidence from behavioral and eye-tracking methods*.
157. Sana-e-Zehra Haidry (2017). *Assessment of Dyslexia in the Urdu Language.*
158. Srđan Popov (2017). *Auditory and Visual ERP Correlates of Gender Agreement Processing in Dutch and Italian.*
159. Molood Sadat Safavi (2017). *The Competition of Memory and Expectation in Resolving Long-Distance Dependencies: Psycholinguistic Evidence from Persian Complex Predicates.*
160. Christopher Bergmann (2017). *Facets of native-likeness: First-language attrition among German emigrants to Anglophone North America.*
161. Stefanie Keulen (2017). *Foreign Accent Syndrome: A Neurolinguistic Analysis.*
162. Franz Manni (2017). *Linguistic Probes into Human History.*
163. Margreet Vogelzang (2017). *Reference and cognition: Experimental and computational cognitive modeling studies on reference processing in Dutch and Italian.*
164. Johannes Bjerva (2017). *One Model to Rule them all. Multitask and Multilingual Modelling for Lexical Analysis: Multitask and Multilingual Modelling for Lexical Analysis.*
165. Dieke Oele (2018). *Automated translation with interlingual word representations*.
166. Lucas Seuren (2018). *The interactional accomplishment of action.*
167. Elisabeth Borleffs (2018). *Cracking the code - Towards understanding, diagnosing and remediating dyslexia in Standard Indonesian.*

168. Mirjam Günther-van der Meij (2018). *The impact of degree of bilingualism on L3 development English language development in early and later bilinguals in the Frisian context*.
169. Ruth Koops van 't Jagt (2018). *Show, don't just tell: Photo stories to support people with limited health literacy.*
170. Bernat Bardagil-Mas (2018).*Case and agreement in Panará*.
171. Jessica Overweg (2018). *Taking an alternative perspective on language in autism*.
172. Lennie Donné (2018). *Convincing through conversation*: *Unraveling the role of interpersonal health communication in health campaign effectiveness.*
173. Toivo Glatz (2018). *Serious games as a level playing field for early literacy: A behavioural and neurophysiological evaluation.*
174. Ellie van Setten (2019). *Neurolinguistic Profiles of Advanced Readers with Developmental Dyslexia*.
175. Anna Pot (2019). *Aging in multilingual Netherlands: Effects on cognition, wellbeing and health*.
176. Audrey Rousse-Malpat (2019). *Effectiveness of explicit vs. implicit L2 instruction: a longitudinal classroom study on oral and written skills*.
177. Rob van der Goot (2019). *Normalization and Parsing Algorithms for Uncertain Input*.
178. Azadeh Elmianvari (2019). *Multilingualism, Facebook and the Iranian diaspora*.
179. Joëlle Ooms (2019). *"Don't make my mistake": Narrative fear appeals in health communication.*
180. Annerose Willemsen (2019). *The floor is yours: A conversation analytic study of teachers' conduct facilitating whole-class discussions around texts*.
181. Frans Hiddink (2019). *Early childhood problem-solving interaction: Young children's discourse during small-group work in primary school*.
182. Hessel Haagsma (2020). *A Bigger Fish to Fry: Scaling up the Automatic Understanding of Idiomatic Expressions*.
183. Juliana Andrade Feiden (2020). *The Influence of Conceptual Number in Coreference Establishing: An ERP Study on Brazilian and European Portuguese.*
184. Sirkku Lesonen (2020). *Valuing variability: Dynamic usage-based principles in the L2 development of four Finnish language learners*.
185. Nathaniel Lartey (2020). *A neurolinguistic approach to the processing of resumption in Akan focus constructions*.
186. Bernard Amadeus Jaya Jap (2020). *Syntactic Frequency and Sentence Processing in Standard Indonesian.*
187. Ting Huang (2020). *Learning an L2 and L3 at the same time: help or hinder?*.

188. Anke Herder (2020). *Peer talk in collaborative writing of primary school students: A conversation analytic study of student interaction in the context of inquiry learning*.
189. Ellen Schep (2020). *Attachment in interaction: A conversation analytic study on dinner conversations with adolescents in family-style group care*.
190. Yulia Akinina (2020*). Individual behavioural patterns and neural underpinnings of verb processing in aphasia*.
191. Camila Martinez Rebolledo (2020). *Comprehending the development of reading difficulties in children with SLI*.
192. Jakolien den Hollander (2021). *Distinguishing a phonological encoding disorder from Apraxia of Speech in individuals with aphasia by using EEG*.
193. Rik van Noord (2021). *Character-based Neural Semantic Parsing*.
194. Anna de Koster (2021). *Acting Individually or Together? An Investigation of Children's Development of Distributivity*.
195. Frank Tsiwah (2021). *Time, tone and the brain: Behavioral and neurophysiological studies on time reference and grammatical tone in Akan*.
196. Amélie la Roi (2021). *Idioms in the Aging Brain*.
197. Nienke Wolthuis (2021). *Language impairments and resting-state EEG in brain tumour patients: Revealing connections*.
198. Nienke Smit (2021). *Get it together: Exploring the dynamics of teacher-student interaction in English as a foreign language lessons*.
199. Svetlana Averina (2021). *Bilateral neural correlates of treatment-induced changes in chronic aphasia*.
200. Wilasinee Siriboonpipattana (2021). *Neurolinguistic studies on the linguistic expression of time reference in Thai*.
201. Irene Graafsma (2021). *Computer programming skills: a cognitive perspective*.
202. Pouran Seifi (2021). *Processing and comprehension of L2 English relative clauses by Farsi speakers*.
203. Hongying Peng (2021). *A Holistic Person-Centred Approach to Mobile-Assisted Language Learning*.
204. Nermina Cordalija (2021). *Neurolinguistic and psycholinguistic approaches to studying tense, aspect, and unaccusativity*.
205. Aida Salčić (2021). *Agreement processing in Dutch adults with dyslexia*.
206. Eabele Tjepkema (2021). *Exploring content-based language teaching practices to stimulate language use in grades 7 and 8 of Frisian trilingual primary education*.
207. Liefke Reitsma (2021). *Bilingualism and contact-induced language change: Exploring variation in the Frisian verbal complex*.

208. Steven Gilbers (2021). *Ambitionz az a Ridah: 2Pac's changing accent and flow in light of regional variation in African-American English speech and hip-hop music*.
209. Leanne Nagels (2021). *From voice to speech: The perception of voice characteristics and speech in children with cochlear implants*.
210. Vasilisa Verkhodanova (2021). *More than words: Recognizing speech of people with Parkinson's disease*.
211. Liset Rouweler (2021). *The impact of dyslexia in higher education*.
212. Maaike Pulles (2021). *Dialogic reading practices: A conversation analytic study of peer talk in collaborative reading activities in primary school inquiry learning*.
213. Agnes M. Engbersen (2022). *Assisting independent seniors with morning care: How care workers and seniors negotiate physical cooperation through multimodal interaction*.
214. Ryssa Moffat (2022). *Recognition and cortical haemodynamics of vocal emotions—an fNIRS perspective*.
215. Diane Mézière (2022). *Using eye movements to develop an ecologically-valid AI measure of reading comprehension*.
216. Ann-Katrin Ohlerth (2022). *Improving preoperative nTMS with a dual-task protocol - the contribution of action naming to language mapping: The contribution of Action Naming to language mapping*.
217. Masha Medvedeva (2022). *Identification, Categorisation and Forecasting of Court Decisions*.

GRODIL
Center for Language and Cognition Groningen (CLCG)
P.O. Box 716
9700 AS Groningen
The Netherlands