UNIVERSITY OF COPENHAGEN
FACULTY OF SOCIAL SCIENCES

Master's Thesis — Social Data Science

Espen Rostrup

Exploring Danish court rulings using quantitative methods

The evolution of the Danish legal language and predicting who is to pay the cost of a trial

Char. count (incl. whitespace): 95,944
Page count: 39.98
Date of submission: May 31, 2022
Supervisors: Johanna Einsiedler & Nikolaj Arpe Harmon

Abstract

In this thesis, I use the most extensive historical Danish legal corpus, the journal Ugeskrift for Retvæsen, to 1. analyse the evolution of the Danish language through the use of word embeddings and to 2. evaluate whether it is possible to predict the Danish courts' decision on which party that is to pay the cost of a trial. I find that the estimated dynamic word embeddings reflect actual changes in the Danish courts' use of language over time. Furthermore, I can train a classifier using only the text in the court rulings that can predict who is to pay the cost of a trial with an average accuracy of 69 pct. Both analyses serve as proofs-of-concept and could be improved in various ways. The overall goal of this thesis is to incentivise further research of Danish legal documents using quantitative methods.

Table of contents

1 Introduction **6**

1.1 The Danish legal language and dynamic word embeddings 6

1.2 Predicting who pays the cost of a trial in a court case 8

1.3 A court ruling: decisions and judgments 9

2 The corpus: Ugeskrift for Retvæsen **9**

2.1 Collecting the data 10

2.2 Describing the data 11

2.2.1 Associated laws 11

2.2.2 The length of a sentence and the length of a word 13

3 Dynamic Danish word embeddings **16**

3.1 What is a word embedding? 16

3.1.1 Estimation of word embeddings 16

3.1.2 The Word2Vec estimation framework 16

3.1.3 Alternatives to Word2Vec 17

3.1.4 Danish Word2Vec embedding applications 17

3.1.5 Evaluation of the quality of static word embeddings 18

3.1.6 Dynamic word embeddings 18

3.2 The models I use to estimate word embeddings 19

3.2.1 The Skip-Gram model with Negative Sampling (SGNS) 19

3.2.2 Parameterisation and estimation of the SGNS 20

3.2.3 Aligned Word2Vec 21

3.2.4 Dynamic Word2Vec 22

3.3 Results 23

3.3.1 Static embeddings estimated using the SGNS model 23

3.3.2 Training the Dynamic Embeddings 24

3.3.3 The dynamic embeddings for the words *stoffer* and *fod* 25

3.3.4 Which are the words that changed the most? 28

3.4 Discussion 28

4 Predicting who pays the cost of a trial **29**

4.1 Legal outcome prediction 29

4.2 Identifying the legal outcome: The cost of a trial 31

4.2.1 Pattern matching to find who pays the cost of a trial 31

4.2.2 Creating the labels step-by-step 32

4.2.3 Remark on the difference in labelling between criminal and non-criminal cases 34

4.3 Text as predictors in a linear support vector classifier 35

4.3.1 Document embeddings 35

4.3.2 Preprocessing the text 36

4.4 The linear support vector classifier in a multiclass setup 36

4.4.1 Tuning and evaluating the classifier 37

4.5 Results 38

4.5.1 The resulting labels 38

4.5.2 Prediction results 39

4.6 Discussion 41

5 Closing remarks **42**

References **43**

A Count of words **49**

B Laws associated to court documents split in decisions and judgements **50**

C Sentence length through time by court **51**

D The objective function in the DW2V model **52**

E Investigations into the dynamic embeddings of *sex*, *imødegå*, *mand* and *overfald* **53**

F Evaluating the ten most changed words as found using DW2V embeddings **55**

G Evaluating the regex patterns used for entity matching **59**

H Stopwords **61**

I Distribution of labels by court instance **62**

J 5-fold cross-validation results **63**

List of Figures

1 An example of a collected document: U.1953.1088 . . . 11
2 Count of collected rulings by court and distribution of rulings by their associated law 12
3 Sentence- and word length in the UfR through time . . . 15
4 A screenshot of the interactive embedding projection . . . 24
5 Dynamic word embeddings estimated with the DW2V and AW2V model for the words *fod* and *stoffer* . . . 26
6 Dynamic word embeddings for the word *stoffer* cast to a one-dimensional space . . 27
7 Process of identifying who pays the cost of a trial . . . 33
A.1 Count of words in court rulings . . . 49
B.1 Dynamics of the 15 most associated laws of court documents divided into decisions and judgements . . . 50
C.1 Sentence length by court . . . 51
E.1 Dynamic word embeddings mapped in a two-dimensional space for the words *sex*, *imødegå*, *mand* and *overfald* using the DW2V embeddings . . . 53

List of Tables

1 Corpus distribution by association to regulation and document type . . . 14
2 Count of words and documents in custom time periods used to estimate the AW2V-model . . . 25
3 Distribution of legal outcome by case type . . . 39
4 Prediction results: Who pays the cost of the trial? Comparing masked and not masked data. . . . 39
5 Features with a large positive estimated decision function coefficient for the class "plaintiff/prosecutor pays the cost of the trial" . . . 40
6 Prediction results: Who pays the cost of trial? Comparing criminal and non-criminal cases with masked data. . . . 41
F.1 The ten words that have changed the most as evaluated by cosine distance . . . 55
F.2 Selected sentences using the top-10 most changed words as estimated with cosine distance . . . 57
G.1 All regex patterns used for entity matching . . . 59
I.1 Distribution of labels by court instance . . . 62
J.1 The preprocessing steps and hyperparameter found using 5-fold cross-validation . . 63

1 Introduction

The language of law and practice of law has always been intertwined. Legal professionals study the wording of legal documents in intricate detail to condense the exact meaning of a given legal subject. There is an abundance of legal studies examining the current practice of different law areas investigating court rulings qualitatively to understand the practice better; it is one of the core abilities a law graduate has obtained through their studies. However, understanding the Danish legal language through quantitative methods is mostly unexplored. Processing Danish court rulings quantitatively can give us an insight into the workings and culture of the Danish courts, especially when combined with a qualitative assessment of the results. For instance, in her PhD-thesis Kjærgaard (2010) studies the (lack of) impact of a language policy on the complexity of the language of the court using 95 court rulings (211,138 words). She finds a degree of language policy resistance in the Danish courts, which might be due to the different language ideologies of the employees at the Danish courts. Hence, using a quantitative approach, she lays the groundwork for understanding the society's reaction to a policy, which she then assesses qualitatively. International studies of culture using quantitative methods on text are plentiful. As an example Fuhse, Stuhler, Riebling, and Martin (2020) draws on quantitative methods to analyse the discourse of the Weimar Republic, gaining an understanding of, e.g. how political parties in Germany positioned themselves in an ideological space by using the word *wolk* [trans. people] in different ways.

This thesis aims to show some of the possibilities of using the most important collection of Danish court rulings, the journal Ugeskrift for Retvæsen (UfR), for research using quantitative methods. I will show some features of the corpus (the collection of rulings) in-depth, focusing on the temporal aspect of the corpus; the first court rulings in the corpus stem from 1867 and the most recent rulings are from 2021. Then I will try to answer two separate research questions. The first one is related to the *dynamics of the Danish legal language*, and the second one is related to *legal outcome prediction*:

> Is it possible to investigate how the Danish legal language has evolved through time using dynamic word embeddings?
>
> Is it possible to predict which party in a court case has to pay the cost of the trial using the text from the court ruling?

I will answer these questions in separate sections of the thesis since they are easier to think of as two independent studies even though they share the same underlying data.

1.1 The Danish legal language and dynamic word embeddings

By answering the first question, I provide a foundation for investigations into the dynamics of the legal language by estimating time-dependent numeric representations of words, i.e. dynamic word embeddings. The Danish legal language is the subject of interest for many linguistic scholars in part due to its separation from the "normal" danish language. The legal language is attributed to a particular style, *kancellistil* (similar to officialese in English), associated with overly complicated sentence structures and very formal words not used in typical danish. As P. Andersen (2015:34)

defines text written in kancellistil: a single thought is contained in a single sentence to include as much information as possible. Since the 1970s, the issue of the legal language being hard to grasp for a common danish citizen has been addressed in research to promote "klarsprog" [trans. Plain Legal Language] in legal texts (P. Andersen, 2015). In 2003 the Courts of Denmark released a language policy trying to incite a more accessible language (updated in 2014, see Danmarks Domstole (2014)). However, this policy did not change the written language all that much (Kjærgaard, 2010, 2012).

Word embeddings can, for instance, be used to analyse cultural trends. For example, Kozlowski, Taddy, and Evans (2019) explore the evolution of culture using word embeddings. They create six different social class dimensions by combining word pairs such as (*poor-rich*) and (*inexpensive-expensive*) to estimate the class dimension "affluence". Then they map other words onto the resulting class dimension. For instance, by mapping the word *tennis* to a combination of class dimensions, they find that tennis is *gender-neutral* (orthogonal to the gender dimension), but it scores high on the affluence axis, i.e. tennis is associated with wealth. They explore the dynamics of the estimated class dimension by comparing the similarity between each of them. They find that the education class dimension is not very similar to the affluence dimension at the beginning of the 20th century, but it is the class dimension most similar to the affluence class dimension in the 1990s, i.e. high education is associated with affluence (Kozlowski et al., 2019:922).

This is the first study that estimates Danish *dynamic* word embeddings. Dynamic word embeddings have been used to, e.g. investigate general statistical laws of semantic change (Hamilton, Leskovec, & Jurafsky, 2016b), evaluate whether word changes are due to cultural or linguistic processes (Hamilton, Leskovec, & Jurafsky, 2016a), predict the state of conflict in countries over time (Kutuzov, Velldal, & Øvrelid, 2017) and understand how harm-related concepts in psychology have changed (Vylomova, Murphy, & Haslam, 2019). The lack of research into the Danish language using dynamic word embeddings might be a consequence of few if any, good open-source digitalised texts spanning a large enough period for such embeddings to be estimated. The UfR corpus provides a unique opportunity of gaining insights into semantic changes of words through time, being a homogeneous text source containing continually comparable documents since 1867.

I estimate the dynamic word embeddings using two frameworks, aligning static embeddings as trained with the Skip Gram negative sampling model from the Word2Vec framework *post hoc* (Hamilton et al., 2016b) and the Dynamic Word2Vec model (Yao, Sun, Ding, Rao, & Xiong, 2018). To preview the results, I find that for select words, the dynamic embeddings reflect explainable changes in the language use of the Danish courts.

I will in section 3 explain the concept of word embeddings, how current state-of-the-art literature tackles the problem of tracking semantic change in language through time and the models I use to estimate the embeddings. Finally, I evaluate the quality of the embeddings exploring the semantic change for a select amount of words. I also present an online tool so the reader can investigate how words are related in a word embedding space using *static* word embeddings. The static word embeddings are estimated on the entire corpus. Since I am not able to disclose the data[1], the

[1]The publisher of the UfR, Karnov Group, terms of service states that one is not allowed to replicate their material to a third party, see: www.karnovgroup.dk/support/vidensbase/licensvilkaar-for-onlineprodukter

online visualisation tool provides the interested reader with a way of engaging with the data.[2].

1.2 Predicting who pays the cost of a trial in a court case

Addressing the second research question can contribute to a rapidly growing body of literature predicting legal outcomes using text as data. Legal outcome prediction is an exciting subfield due to the possible applications. Legal outcome prediction can improve our understanding of what drives a specific court ruling, e.g. which words are associated with a person being convicted or which arguments are most important for evicting a tenant in a rental dispute. In a legal methods book, Holtermann and Madsen (2021:62) even argues that legal outcome prediction can influence how judges approach new problems and help the arguments of legal professionals by relying on *big data analysis* such as legal outcome prediction.

There are no other studies I am aware of that try to predict the assignment of trial costs. Multiple studies are concerned with predicting the court's verdict, e.g. if an article of the European Human Rights Convention is breached (Aletras, Tsarapatsanis, Preoţiuc-Pietro, & Lampos, 2016) or whether a case is affirmed or reversed by the Supreme Court of the United States (Katz, Bommarito, & Blackman, 2017). Assignment of the cost of a trial is a non-perfect proxy of the court verdict. The Danish courts will assign the cost of the trial to the trial's losing party with a few notable exceptions, which I will elaborate upon below.

This study cannot be seen as an attempt to predict future court verdicts since it is not possible to separate information known prior to the final verdict from the rulings. The primary motivation for this study is that successful implementation can motivate future research in the area of legal outcome prediction in Denmark.

To answer the question, I use pattern matching to identify the party who pays the trial cost. Obtaining this information has value in and of itself, and the pattern matching approach can easily be used on court rulings not included in the UfR. The cost of a trial and the assignment hereof is of societal importance. For instance, it is argued by Olesen et al. (2020) that the cost of a trial can be a disproportional extra punishment to a defendant in a criminal case. The defendant both has to adhere to the court's sentence and has to pay the cost of the trial if found guilty. Providing a way of categorising court documents by who pays the trial cost might inform policymakers and provide a relevant keyword for legal information retrieval, e.g. if a legal professional is searching for court cases where the defendant pays the cost of trial.

In section 4, I present the current legal outcome prediction literature. Then I explain how I identify who pays the cost of a trial, followed by describing the pre-processing step; how I extract document level information from each court case. I outline the linear support vector classifiers I use to predict the outcome of a court case and finally display and discuss the classifiers' performance.

To preview the findings: I can predict who pays the cost of a trial with a 69 pct. accuracy using a classifier trained on the subset of court rulings where I can identify who is assigned to pay the trial cost. In comparison, a naive classifier predicting the most observed class in the data set independent of any other features has an accuracy of 39 pct. I explore the sensitivity of my

[2]Most of the code used in the thesis along with estimated word embeddings and other data is published at the GitHub repository www.github.com/EspenRostrup/ufr-analysis/

result by training classifiers on criminal cases and non-criminal cases separately and find that I can predict the cost of a trial of a criminal case with a 75 pct. (naive classifier 52 pct.) accuracy and non-criminal cases with an accuracy of 71 pct. (naive classifier 39 pct.) These classifiers performs slightly worse compared to similar studies using the same approach in different languages. The inferior results are probably due to the indirect outcome: the actual outcome of a court case is the verdict, and the assignment of the cost of a trial is not discussed in the ruling. By investigating the predictors of the classifier I find that this is plausible i.e. the classifier's performance relies heavily on the correlation between the court verdict and the trial cost assignment.

1.3 A court ruling: decisions and judgments

Before commencing my analysis, I introduce some terminology. I define a court *ruling* as either a court *decision* or a court *judgment*. A court decision (danish, kendelse or beslutning)[3] often regards the formalities of the case and does not necessarily finalise it (Walbom, 2021). I define a court *judgment* as all the documents that are not decisions. For both court judgments and decisions, the entire document might include some of the following sections: a statement of claim, the party's arguments, witness testimonies, the arguments of the court and a verdict. The verdict is the document section that explains the final sentence in a few sentences (it is the last section of the document). Furthermore, court rulings might include previous court instances' judgments or decisions in the text. I will, at some points, use *document* instead of *ruling*, mostly when discussing subjects of a more textual nature, but the terms are used interchangeably.

2 The corpus: Ugeskrift for Retvæsen

Ugeskrift for Retvæsen (UfR) is one of the most important sources for danish legal practice both in recent times and historically (M. B. Andersen, 2017:11). To exemplify the significance, the UfR indexation system[4] is the de facto standard for referring to the rulings presented in the journal.[5] There is no public database containing any large amount of court rulings. In 1988 Retsinformationsrådet (the council of legal information in Denmark) recommended the establishment of public databases containing court judgments (Retsinformationsrådet, 1988). It was first in January 2022 that a public database containing court judgments from *multiple* danish courts was released (domsdatabasen.dk).[6] There are separate databases containing supreme court judgments, selected high court judgments and judgments from the Maritime and Commercial High Court. None of these databases contains the number of documents required for large scale quantitative analysis. Christensen, Esmark, and Olsen (2021:180) presents a review of current digitalised legal documents.

The journal UfR has been published weekly since 1867 (M. B. Andersen, 2017). The journal is

[3] There is a difference in the implications of the requirements of the court whether it is a *kendelse* or a *beslutning*. However, it is not relevant for the present study.

[4] UfR indexation system follows this pattern: U.YEAR.PAGE

[5] Danske Advokater and DJØF note that identifying specific judgements in legislative work and other judgements are done using UfR indexation as a non-party intervenor in a trial where Karnov sued Schultz (another legal database company) for using the UfR indexation system as metadata in their database (Handelsretten, 2018:42).

[6] It is not fully operational yet, containing only around 1,000 judgments at the time of writing (May 2022).

split into two sections: Section A contains the "most relevant" court rulings from the High Courts, the Supreme Court, and the Maritime and Commercial High Court. Section B contains academic articles concerning legal matters. I have only collected the documents from section A. Note that the Danish court system is based on the two-tier principle, i.e. it is possible for the party in a court case to appeal the court's ruling to a higher court instance (at least once), why some cases might appear more than once.[7]. The data will not be disclosed due to the publisher's, Karnov Group's, terms of service.

An editorial board consisting of (mainly) judges select the court rulings to be included in the journal. They choose rulings that hold a certain precedent value [trans. præjudikatsværdi] (Holtermann & Madsen, 2021:63). Approximately 600 to 700 court rulings are published in UfR each year. These numbers are low compared to the total amount of rulings made by Danish courts: In 2021 577,868 cases were closed in the District Courts, 11,649 cases were closed in the Eastern and Western High Courts, 10,085 cases were closed in the Maritime and Commercial High Court, and 291 cases were closed in the Supreme Court (Domstolsstyrelsen, 2022). So when analysing documents from the UfR, it is important to keep in mind that the rulings in UfR are not representative of a typical ruling due to the selection criteria imposed by the publisher. Furthermore, the rulings might have been shortened, leaving out insignificant parts of the original form as published by the court (Christensen et al., 2021:181). I am not able to distinguish which rulings have been shortened and which rulings that have not.

2.1 Collecting the data

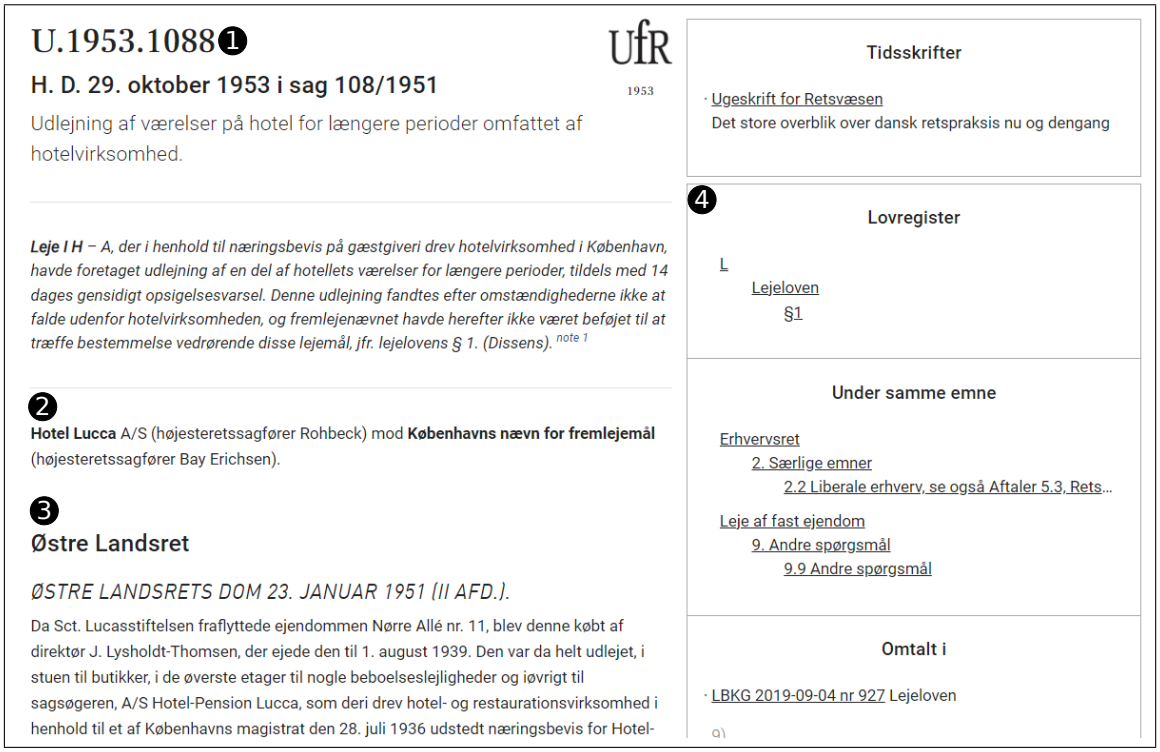
The court rulings are collected from UfR Online at pro.karnovgroup.com. To access the site requires a subscription.[8] According to (M. B. Andersen, 2017:9) all rulings from UfR since 1867 are published in UfR Online. In figure 1 an example of a court document that I have collected is shown.

For each document, I obtain 1.[9] the UfR index ID and the title of the ruling, 2. the parties, 3. the text of the ruling and 4. metadata including e.g. the laws that the ruling is related to. I do not use the summary of the rulings since it is written by the journal's editors and not the court.

With regards to point 2. I note that for non-criminal cases, the plaintiff is the first party mentioned and the defendant is the second party. For criminal cases, the state prosecutor's office is always the first party. The parties are found with pattern matching using the word "mod" as a separator between parties. Pattern matching is described in section 4.2.1. In some cases, there are more than two opposing parties in one trial.

[7] For more information on the formalities of the danish court system (in English) I refer to (Domstolsstyrelsen, 2021).
[8] The cost of a subscription to Karnov was 18,000 DKK per user per year in 2018 (Handelsretten, 2018:10).
[9] The numbers refer to the numbers in figure 1.

Figure 1
An example of a collected document: U.1953.1088



2.2 Describing the data

The entire corpus consists of 63,915 court rulings collected from UfR Online.[10] In figure 2a I depict the count of rulings by court instance included in the UfR.

The figure shows an upward trend in the count of cases included in the data set from 1867 to 2021. Through time, the most notable change occurred in 1921, when more judgements from the high courts were included in the journal. The upward trend is also reflected in the count of words in the documents collected (shown in appendix A). I note that the increase in count and length of rulings starting in the 1990s is in line with M. B. Andersen (2017:7) who states that the increase in rulings in the end of the 1990s (beginning of 2000s) was due to a general popular demand. The increase in the length of the rulings is larger than the increase in the number of rulings, which can suggest one of two things: Either the editorial board keeps more of the original court ruling in the journal or the original court rulings are longer. In total there are 123 million words in the collected documents. In comparison the largest open source text corpus in Danish includes 1,045 million words (Strømberg-Derczynski et al., 2021).

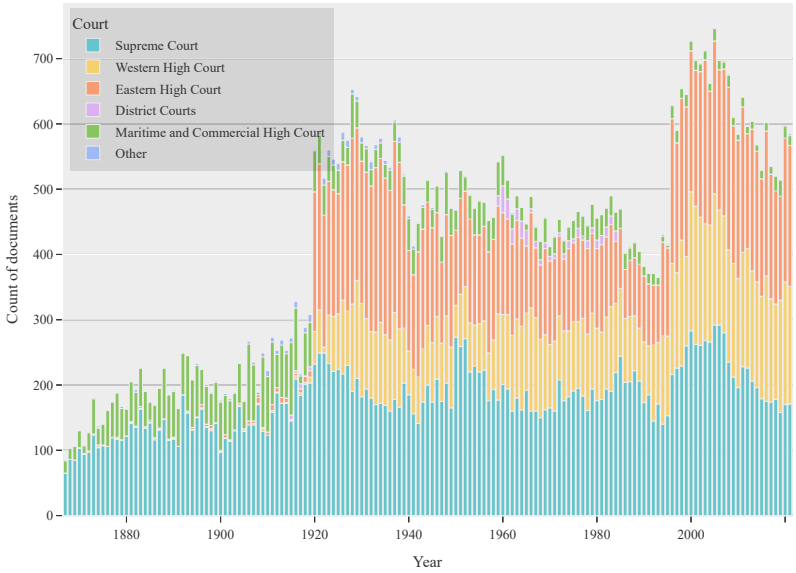
2.2.1 Associated laws

In figure 2b the distribution of relevant laws used in the court rulings through time is shown. A ruling can be associated with multiple laws. This information is not a part of the original law but is made by the publisher, Karnov Group, to ease information retrieval for users of their platform.

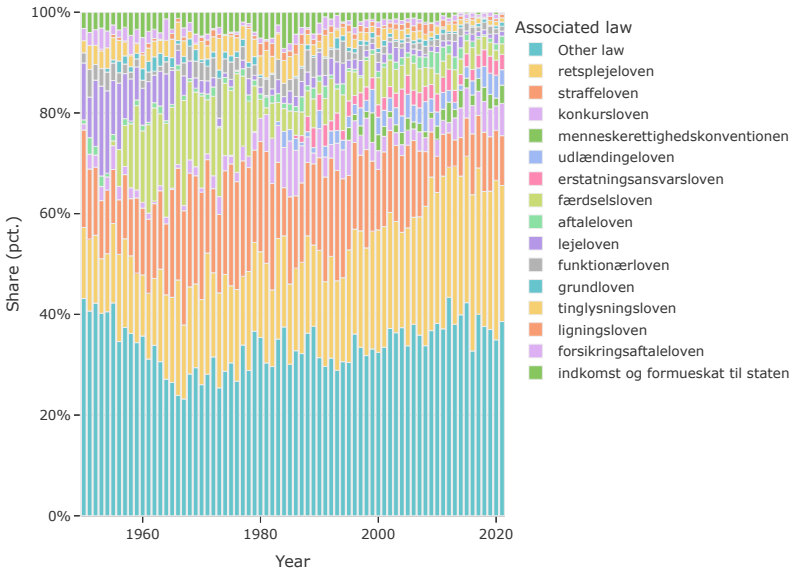
[10] The UfR IDs of all included court judgements and decisions are uploaded to the GitHub repository.

Figure 2
Count of collected rulings by court and distribution of rulings by their associated law

(a) Rulings by court



(b) Distribution of the most often associated regulations to a court ruling through time



Note: Figures are available in interactive formats. For figure 2a see www.rostrup.nu/distribution_of_court_documents_by_associated_law and for figure 2b see rostrup.nu/distribution_of_court_documents_by_associated_law.

The information is only provided for documents produced after 1950 and only for 77.4 pct. of these documents (28,768 out of 37,192 rulings). For illustrational purposes, laws not in the top 15 of the most referenced laws are grouped into "other laws".

From the figure, it can be seen that the composition of the laws documents is related to changes over time. As mentioned UfR publish judgements or decisions with precedent value: The change of a law leading to new type of court rulings is at interest for the journal at first but after setting a precedence subsequent rulings using the same part of a law might be less interesting. Examples of this can be observed directly from the figure. For instance, observe "færdselsloven" [trans. the Road Traffic Act] that was introduced in 1955 (Waaben, Munck, Eiriksson, & Aagard, 2017:83). In the figure, it is seen that a large proportion of documents are related to the road traffic act just after 1955, but the share of documents related to the law declines over time (note that the law was replaced with a new principal act in 1976). This pattern is even clearer when only looking at judgements disregarding the court decisions.[11] Investigating this further one could look at the relevant paragraphs of the Road Traffic Act, to get a picture of the evolution of the debate surrounding the replacement of the act in 1976. However, such analysis is beyond the scope of this thesis.

Documents related to associated laws are also different in length. In table 1 the count of documents and the average word count for judgements and verdicts are shown.

Cases that are related to grundloven [trans. the Danish constitution] are on average the longest (6,479 words on average) where cases related to retsplejeloven [trans. the Danish Administration of Justice Act] are on average the shortest (1,297 words on average). It is apparent that court decision documents are on average less than half the length of a court judgement – the proportion varies across associated regulations but court decisions are consistently shorter for all types of regulations as shown in the table. The observed difference in features between associated regulations and court decisions and judgements might be a product of the editorial strategy rather than reflecting the features of the court judgements and decisions as the court produced them.

2.2.2 The length of a sentence and the length of a word

In figure 3 the average sentence length through time is shown along with the average word length. The rulings are divided into sentences using the pre-trained NLTK sentence tokeniser. The NLTK tokeniser is trained on roughly 550,000 sentences published in the newspapers Berlingske Tidende and Weekend Avisen in 1995 using the punkt tokeniser model (Kiss & Strunk, 2006)[12]. The sentence length (left axis) is measured in words where a word is defined as any sequence of non-numeric characters separated by a white space character. The word length (right axis) is the count of non-numeric characters in a sequence separated by white space. The sentence tokeniser is by no means perfect. For instance the sentence "I sagsomkostninger for Højesteret skal appellanten, Alm. Brand af 1792, betale 50.000 kr. til statskassen."(Supreme Court, 1996:876) is incorrectly classified

[11] An equivalent of figure 2b only using judgements 2b is shown in appendix B figure B.1a. It is also shown in figure B.1b that the majority of the court decisions are related to retsplejeloven (the Administration of Justice Act), which is to be expected since, as noted above, decisions are of a more procedural nature. It might be beneficial for the reader to use the online version of the figures removing law categories interactively to see how it impacts the total share of documents.

[12] This information can be found in the readme-file, when downloading the zip-file for the Punkt Tokenizer Models at www.nltk.org/nltk_data/ (item 70 at time of writing)

Table 1
Corpus distribution by association to regulation and document type

	Count of documents			Word count (avg)		
Name of associated regulation:	Decisions	Judgements	Total	Decisions	Judgements	Total
aftaleloven	23	443	466	2,846	5,285	5,165
boligreguleringsloven	37	167	204	1,225	2,523	2,287
erstatningsansvarsloven	10	550	560	2,046	5,119	5,064
forsikringsaftaleloven	10	351	361	2,305	3,261	3,235
forældelsesloven	15	174	189	1,210	2,447	2,349
funktionærloven	8	631	639	2,538	3,092	3,085
færdselsloven	171	2,152	2,323	1,287	1,237	1,241
grundloven	19	233	252	3,567	6,717	6,479
indkomst og formueskat til staten	12	659	671	1,084	2,891	2,858
konkurrenceloven	4	247	251	2,297	4,313	4,281
konkursloven	598	578	1,176	1,395	3,575	2,466
købeloven	3	235	238	704	3,106	3,076
lejeloven	181	802	983	1,103	1,425	1,366
markedsføringsloven	20	213	233	2,506	4,712	4,522
menneskerettighedskonventionen	111	193	304	2,518	6,718	5,184
myndighedslov	21	169	190	560	1,392	1,300
retsplejeloven	4,541	1,889	6,430	1,069	1,845	1,297
straffeloven	209	3,385	3,594	1,679	1,507	1,517
tinglysningsloven	449	242	691	758	2,421	1,340
udlændingeloven	153	128	281	1,991	3,753	2,794
Other associated regulations	1,520	7,212	8,732	1,789	4,223	3,799
Total, document has association*	8,025	20,275	28,300	1,172	2,412	2,062
Total 1950-2021	9,375	27,817	37,192	1,232	2,909	2,486
Total Entire period	10,706	53,209	63,915	1,146	1,939	1,806

Note: *Total does not sum for document count of associated regulations since a document can be associated with multiple regulations. *Other associated regulations* refers to regulations that are not in the 20 regulations that are associated to most documents across time. I leave it to the reader to translate the different associated regulations.

into two sentences, namely "I sagsomkostninger for Højesteret skal appellanten, Alm." and "Brand af 1792, betale 50.000 kr. til statskassen." This seems to be the issue for many named entities that contain an abbreviation. Standard abbreviations such as "kr." are known by the tokeniser, so it does not split the sentences using the dot here, as shown in the example.

Before 1950 the sentence length seems somewhat unstable, which might be due to the scope of the sentence tokeniser's training data and the quality of the digitalisation of the documents in UfR Online. The NLTK-tokeniser was trained on data from 1995. Hence data far away from that point in time might yield worse sentence boundaries. The data from before 1950 was added in 2017 at once, including less metadata (M. B. Andersen, 2017:9). The documents might have been digitalised using Optical Character Recognition (OCR), since there seem to be many typographical errors and redundant space characters. After 1950 the sentence length shows a more stable trend decreasing in length until 2000 and increasing in length after that. Until the mid-1990s the length of a word increases whereafter it decreases, i.e. longer sentences with shorter words have been the trend for the last 20 years. The change in the length of a word is small though. The change in sentence length is similar across courts (see appendix C for a figure analogous to figure 3 where the average sentence length is shown for each court).

As noted in the introduction, the Courts of Denmark introduced a language policy in 2001 and a revised version in 2014. The language policy states that a "good" sentence is 15 to 18 words long

Figure 3
Sentence- and word length in the UfR through time

Note: An interactive version can be found at www.rostrup.nu/average_length_of_sentences_and_words. Sentence length by time and court type can be found in appendix C.

(Danmarks Domstole, 2014:11). The lack of adherence to this policy is described by Kjærgaard (2010) as language policy resistance. She also investigates the sentence length and other measures of language complexity, finding that the language policy has not been implemented, showing a slight increase in the length of sentences comparing the text produced by the same judges in the year 2002 and the year 2008 (Kjærgaard, 2010:64,197).

There is a correlation between the years with the shortest sentence length and the year where the data for training the sentence tokeniser has been obtained. The sentence tokeniser might be better at setting sentence boundaries in texts produced in the same period as the training data than texts produced long after or before the training data. However, it is not obvious that this creates a "shorter sentence bias": The tokeniser essentially learns common abbreviations used in the language and failure to recognise an abbreviation should yield a shorter sentence, all other things equal. However, conclusions about the language complexity of Danish courts solely using this figure should be made with care.

3 Dynamic Danish word embeddings

This section addresses the first research question outlined in the introduction. I start by explaining what a word embedding is and how the quality of a word embedding is evaluated. Then I describe the dynamic word embeddings and the most relevant literature. After that, I show the models I use to estimate word embeddings. Finally, I evaluate the quality of the static embeddings and the two sets of dynamic embeddings and discuss the results and my approach to estimating word embeddings.

3.1 What is a word embedding?

A word embedding is a numeric representation of a word, typically a fixed-size vector. This numerical representation only carries meaning in the context of other words embedded in the same vector *space*. A word embedding is constructed using the word's context as a reference. It relies on the distributional assumption (Firth, 1957): a word's meaning is defined by the way it co-occur with other words. In word embedding literature, e.g. (Hamilton et al., 2016b; Kozlowski et al., 2019; Church & Hanks, 1989; Chalkidis & Kampas, 2019), an often used citation when describing this assumption is: "You shall know a word by the company it keeps" (Firth, 1957).

Word embeddings allow for computational operations on a word. The computational operations, that be the comparison of words, classification tasks or similar, depend on the quality of the word embeddings. What "the quality" of a word embedding is depends on what the embeddings are being used for. For instance, studies might revolve around the semantic properties of the word embeddings, such as comparing a word's meaning through time. Other applications might use the embeddings for a downstream classification task and be concerned with the classification performance rather than the semantic properties of the embeddings. One would expect that the word embeddings that captures the real semantic relationship between words is better for classification tasks than word embeddings that do not; however, that is not always the case (Bakarov, 2018).

3.1.1 Estimation of word embeddings

There are numerous ways of estimating word embeddings. Early approaches (the 1970s) used a singular value decomposition (SVD) of the term-document matrix (documents, columns in the matrix, containing term-frequencies, rows in matrix) to construct word embeddings (Dumais, 2004; Kozlowski et al., 2019:Appendix A). A drawback of this approach is that all words in a document are considered equally "close" if they appear in the same document, i.e. the context that defines the word is (too) extensive, which leaves the resulting embeddings more imprecise.

3.1.2 The Word2Vec estimation framework

I use the skip-gram model from the Word2Vec framework to estimate word embeddings (Mikolov, Chen, Corrado, & Dean, 2013). The Word2Vec framework consists of two models that use a local context window to operationalise the distributional assumption. The window is the number of

words on each side of a selected word in a sentence that is considered the context. The window can be fixed or dynamic. If it is dynamic, it implies that the range of words included as context is bounded by a constant drawn from a discrete uniform distribution in a range between 1 and a positive integer. The dynamic window provides an implicit weighting of context words so that context words further away from the word matter less for the estimation of the word embeddings. The difference between the two models in the Word2Vec framework, the skip-gram model and the continuous bag-of-words (CBOW) model, lies in their classification task: The CBOW model classifies what the most likely word is conditional on the observed context and the Skip-gram model, on the other hand, classifies what the most likely context is conditional on a word.

3.1.3 Alternatives to Word2Vec

Recently, there has been a surge in using deep learning models to produce word embeddings, especially using *transformers* to create context-dependent word embeddings. Transformer based models (Vaswani et al., 2017) have achieved state-of-the-art results in nearly all text processing tasks, including legal-text classification tasks, e.g. (Chalkidis & Kampas, 2019) and (Medvedeva, Üstun, Xu, Vols, & Wieling, 2021). That the model creates context-dependent word embeddings means that the embedding for a given word change given the context that the word appears in. Hence, the models do not produce single word embeddings as the Word2Vec models. For instance, the first transformer-based language model, BERT (Devlin, Chang, Lee, & Toutanova, 2019), consists of multiple layers of transformer heads producing multiple different word embeddings for a word. Choosing or combining these embeddings does not necessarily yield more semantically meaningful word embeddings than the embeddings produced by a skip-gram or CBOW-model. I will not explore transformer-based models when creating the dynamic embeddings. Still, I note that the BERT model has achieved state-of-the-art results using some standard semantic tests (Wang, Cui, & Zhang, 2020).

3.1.4 Danish Word2Vec embedding applications

Nielsen and Hansen (2017) was some of the first to use the Word2Vec framework to estimate embeddings in Danish. They use three danish corpora to train word embeddings using models from the Word2Vec and GLoVe (Pennington, Socher, & Manning, 2014) frameworks and propose three evaluation data sets to assess the embeddings' semantic similarity. They found results comparable to state-of-the-art semantic tasks on English text using the same evaluation metrics.[13] Another relevant set of word embeddings is publicised by Egense (2018). He uses 30 million danish newspaper articles published in 1880 to 2005 to estimate static embeddings based on the mediestream.dk project. The data used to create the embeddings is not publicly available.

Certain features of the Danish language might make the quality of the estimated embeddings different from English word embeddings. As an example, Pedersen et al. (2012:43) highlights the "flexibility regarding dynamic generation of compounds such as *skiinstruktørsammenslutningssekretæraspirant* [lit. ski-instructor-association-secretary-aspirant]" in a status of the Danish language technology. They also underline that research in automated semantic analysis of the Danish lan-

[13]The word embeddings are published at www.github.com/fnielsen/dasem

guage was lacklustre as of 2012 (Pedersen et al., 2012:61). Since 2012 there has been development in automated semantic analysis. Key contributions in the form of open-source data sets, word embeddings and other natural language processing tools have been collected in the DaNLP project (Pauli, Barrett, Lacroix, & Hvingelby, 2021).

3.1.5 Evaluation of the quality of static word embeddings

As noted above, the quality of a word embedding model depends on what the model is being used for. Bakarov (2018) presents a survey on the word embedding evaluation methods highlighting that there are a lot of potential pitfalls when assessing the quality of a word embedding. Barakov uses two groups of evaluation metrics: An *extrinsic* evaluation of the embeddings – the performance of the embedding on a downstream classification task – and an *intrinsic* evaluation – the embeddings' ability to reflect word relations. I am only concerned with intrinsic evaluation, the embeddings' ability to reflect genuine semantic relationships between the words.

The DaNLP project includes two data sets used for intrinsic evaluation of the word embeddings' quality in Danish. The translated version of the wordsim353 (Finkelstein et al., 2001), the wordsim353-da (Nielsen & Hansen, 2017), and the Danish Similarity Dataset (DSD) (Schneidermann, Hvingelby, & Pedersen, 2020) which is comparable to the English similarity data set Simlex999 (Hill, Reichart, & Korhonen, 2015). The general quality of the embeddings is evaluated using Pearson's correlation coefficient between the human similarity score and the estimated cosine similarity of the word pairs as the relevant metric.[14]

The evaluation data sets rely on human annotators ranking the semantic similarity of a pair of words. Hill et al. (2015) and Schneidermann et al. (2020) argue that "semantic similarity" and "relatedness" are not the same concepts. They further state that the similarities in WordSim353 does not have this distinction. Hill et al. (2015); Bakarov (2018); Schneidermann et al. (2020) all use the same example to illustrate this point: The words "cup" and "coffee" might be closely associated, often appearing together. Still, they are two fundamentally different concepts and therefore, they should not be encoded as semantically similar. In WordSim353 the pair is, however, assigned a higher similarity score than, e.g. the pair "car" and "train", which arguably seems off.

3.1.6 Dynamic word embeddings

I use the term "dynamic embeddings" to refer to time-varying embeddings. Some word embedding literature, e.g. (Pauli et al., 2021), use the term dynamic to refer to whether a word's embedding varies with its context, i.e. context-dependent word embeddings, which is briefly explained in section 3.1.3 above.

According to Szymanski (2017:448) the first study using word embeddings to explore the semantic evolution of words was by Sagi, Kaufmann, and Clark (2011). They tracked semantic changes in language using latent semantic analysis (LSA), which uses a singular value decomposition of the term-frequency matrix to generate word embeddings. With the introduction of the Word2Vec-

[14] The cosine similarity is the dot-product of two word embeddings divided with the two word embeddings' lengths factorised.

framework, (Mikolov, Chen, et al., 2013) several new methods of constructing diachronic word embeddings emerged.

The first proposed methods relied on two-step approaches estimating embeddings in each considered period and providing some way of aligning these embeddings. Kim, Chiu, Hanaki, Hegde, and Petrov (2014) train word embeddings using a skip-gram model estimated in each year for the period 1850-2009 using the Google Books NGram corpus (Michel et al., 2011). Instead of initialising the weights randomly in each year, they initialise the weights using the previous period's estimated word embeddings to ensure that the embeddings are comparable in the same vector space, i.e. they estimate the word embeddings continuously. They compare the cosine similarities between word pairs over time, finding that this approach identifies real semantic changes such as the word "gay" changing its meaning from "happy" to "homosexual". Kulkarni, Al-Rfou, Perozzi, and Skiena (2015) use a linear mapping of the embeddings trained in two time periods that minimises the distance between the embeddings. This ensures that the embeddings are comparable over time. In a similar vein Hamilton et al. (2016b) suggest minimising the distance between the trained embeddings seeing it as an "Orthogonal Procrustes" problem that can be solved using linear algebra. Shoemark, Liza, Nguyen, Hale, and McGillivray (2019) present a systematic comparison of the two-step estimation approaches varying the method for initialisation of weights and whether the embeddings are aligned or not. Among other things, they find that continually initialising embeddings without post hoc alignment of the embeddings performs poorly.

Some newer approaches to estimating dynamic embeddings involve training the dynamic embeddings in one step, i.e. these methods use all the available data without dividing the data into time slices first. Among such approaches can be mentioned Yao et al. (2018) that uses matrix factorisation to smooth the embeddings over time, Bamler and Mandt (2017) that uses an Uhlenbeck-Ornstein process to estimate the time-dependent embeddings and Rudolph and Blei (2018) that use a set of probability distributions known as exponential families to model the evolution of the embeddings.

3.2 The models I use to estimate word embeddings

In this study, I model the dynamic word embeddings using the method for aligning static word embeddings post hoc as proposed by Hamilton et al. (2016b) and the Dynamic Word2Vec (DW2V) model (Yao et al., 2018). I chose these two approaches to estimate word embeddings since they are vastly different (the two-step post hoc alignment approach and a smoothing approach), and since it was somewhat straightforward to estimate the embeddings, i.e. boilerplate code was accessible for alignment of embeddings and for training the DW2V model. Note that the goal of this study is merely to showcase that the Danish language can be explored with dynamic word embeddings and not to advance the research field of dynamic word embeddings methodologically.

3.2.1 The Skip-Gram model with Negative Sampling (SGNS)

I estimate static embeddings with the Skip-gram model with negative sampling (SGNS) in each period for the post hoc alignment approach. Furthermore, the DW2V model builds on some of the properties derived from the SGNS model. Hence, understanding the workings of the SGNS model

is important for understanding how the dynamic word embeddings are estimated.

I estimate static word embeddings using the skip-gram model with *negative sampling*. I use a Dynamic context window, *sub-sample* more frequent words and remove rare words before estimating the model. The pruning of rare words and the subsampling of frequent words increase the size of the context window since these prepossessing operations are done prior to defining the context (Goldberg & Levy, 2014). Subsampling is the process of sampling the most frequent words less frequently. The most frequent words are discarded with probability $P(w_i) = \max\left(1 - \sqrt{\frac{t}{f(w_i)}}, 0\right)$, where $f(w_i)$ is the frequency of word w_i and t is a threshold regulating the number of times a word needs to appear in the corpus for it to be considered frequent. Mikolov, Sutskever, Chen, Corrado, and Dean (2013) argues that frequent words provide less information value than infrequent words. They mention the word "the" as an example of a word that co-occurs with just about every word but contains little informational value. However, increased performance from introducing subsampling might be due to the implicit increase in the context window size as argued by Goldberg and Levy (2014:5) since the subsampling is done prior to estimating the context window. Rare words, as defined by a threshold, are removed due to the potential inaccuracy of the resulting word embeddings for these words.

3.2.2 Parameterisation and estimation of the SGNS

The main goal of the skip-gram model is to find the context words that are most likely to be observed together with a word. A context word is a word observed in proximity to a word, i.e. it is within the context window. More formally, I optimise the probability of observing context word c given the word w by changing the parameter θ i.e.

$$\arg\max_{\theta} \prod_{(w,c)\in D} P(c|w;\theta) \tag{1}$$

where D contains all word-context pairs, (w, c), in the corpus (Goldberg & Levy, 2014).

Mikolov, Sutskever, et al. (2013) propose two ways of estimating the embeddings that are more computationally efficient than the original approach of parameterising equation (1) outlined in (Mikolov, Chen, et al., 2013): using a hierarchical soft-max function and using negative sampling. In this thesis, I will only consider the negative sampling approach to parameterising the skip-gram model.

Negative sampling is based on Noise Contrastive Estimation (NCE) (Gutmann & Hyvärinen, 2010), where the idea is to estimate parameters by training a model to separate artificially generated noise from observed data. Hence, negative sampling implies adding word-context pairs *not* observed in the data to the set of word-context pairs and associating these pairs with a *negative* context value. The goal is then to find the parameters that maximise the probability of a word-pair being observed in the data, given that it actually is observed, and equivalently find the parameters that maximise the likelihood that a word-pair *is not* observed in the data, provided that the word-context pair is artificially generated. The parameters are the fixed-size word and context vectors,

where the fixed-size word vectors are what I refer to as the word embeddings.

Negative sampling reduces the optimisation problem to a binary classification problem for each word-context pair: whether the word-context pair is observed or not. For a single word-context pair the SGNS objective is

$$\log\left(\sigma(v_w \cdot v_c)\right) + k \cdot \mathbb{E}_{c_N \sim P_{D(c)}}\left[\log \sigma(-v_w \cdot v_{c_N})\right] \tag{2}$$

where v_w, v_c are the fixed-size word and context vectors, k is the amount of negative samples that are drawn and c_N is the artificially generated context drawn from the empirical unigram distribution $P_D(c) = \frac{\#(c)^\alpha}{|D|}$. $\#(c)$ is the count of context word c and $|D|$ is the count of word-context pairs in D . The unigram distribution is raised to the power of α.[15] $\sigma(\cdot)$ denotes the sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$ (Mikolov, Sutskever, et al., 2013).

The objective function can be inserted in the log-transformed version of equation (1) to obtain the objective function for all word-context pairs. The model is then optimised using stochastic gradient descent. For the exact parameter update rules I refer to Rong (2016:13–14).

3.2.3 Aligned Word2Vec

Word embeddings trained with the SGNS are only relevant compared to other word embeddings in the same *vector space*. Hence, multiple SGNS-models trained in different periods do not yield comparable word embeddings since each estimation's embedding space is different. Hamilton et al. (2016b) handles this by aligning the word embeddings estimated in each period. They do so by finding the orthonormal matrix, Q, that can rotate the embeddings to find the minimum euclidian distance between two following periods' embeddings. This approach relies on the assumption that words generally do not change much from period to period. The problem boils down to a linear algebra problem known as the Orthogonal Procrustes. The problem is

$$R_t = \arg\min_{Q^T Q = \mathbf{I}} ||QW_{(t)} - W_{(t+1)}||_F \tag{3}$$

$W_{(t)} \in \mathbb{R}^{|\mathcal{V}| \times d}$ are the embeddings trained in period t using the SGNS model, where d is the dimension of the fixed-size embeddings and $|\mathcal{V}|$ is the dimension of the vocabulary (the count of unique words). $||\cdot||_F$ is the Frobenius norm and $R_t \in \mathbb{R}^{d \times d}$. Finding Q involves SVD as shown by Schönemann (1966). Since there are more than two periods the alignment is done iteratively.

There are three main drawbacks of aligning the model this way, as noted by Yao et al. (2018): First, it requires a lot of data for each period. Secondly the method requires all unique words to be present in all time periods since $dim(W^t) = dim(W^{t-1})$ for equation (3) to be solvable reducing the available vocabulary drastically. Finally, this alignment approach does not use data close to

[15] Mikolov, Sutskever, et al. (2013) argues that $\alpha = 0.75$ improves the word embeddings substantially compared to a simple unigram distribution ($\alpha = 1$) which is corroborated by Levy, Goldberg, and Dagan (2015).

the period to produce a given period's embeddings, which underutilises the data. An advantage of this method is the simplicity of implementation: finding Q and rotating the SGNS-embeddings for that period requires few lines of code.[16] I will, from now on, refer to the combined estimations of the SGNS model in each period and the post hoc alignment as the AW2V-model.

3.2.4 Dynamic Word2Vec

Yao et al. (2018) propose an alternative to the two-step approach. They are inspired by the finding of Levy and Goldberg (2014) that the SGNS objective function, equation (2), can be optimized by setting the word-vector and context-vector product ($v_w \cdot v_c$) equal to the *shifted pointwise mutual information* (PMI) matrix (Yao et al., 2018:1). The PMI measures the association between a word and possible context words (Church & Hanks, 1989). Usually, it is represented in a matrix, where each row represents the unique words in the corpus (the vocabulary), and the columns are the possible context words. The degree of association is the joint probability of the context-word pair and the marginal probability of the word and the context. Empirically the PMI is estimated using

$$PMI_{w_i,c_j} = \log\left(\frac{\#(w_i, c_j) \cdot |D|}{\#(w_i) \cdot \#(c_j)}\right) \tag{4}$$

for word-context pair $(w_i, c_j) \in D$. The *shifted* PMI is the PMI with the constant term $-log(k)$ added, where k is the amount of negative samples. The Dynamic Word2Vec (DW2V) model does not shift the PMI, which makes the embeddings slightly different to the ones estimated in the SGNS. Levy et al. (2015) shows that not shifting the PMI yields superior embeddings, when estimating word embeddings using a singular value decomposition of the positive PMI (PPMI) as the word embeddings.

The DW2V model uses the Positive PMI (PPMI). The PMI is ill defined for a lot of word-context pairs due to the nominator of equation (4) being zero ($\log(0)$ is undefined) and it is dense. The PPMI is sparse and well-defined for all word-context pairs (Levy et al., 2015).

Yao et al. (2018) estimate the PPMI for each time period and try to set the embeddings so that the embeddings vectors are as close to the PPMI in that time period but with an alignment constraint, so that the embeddings are *approximately* equal from one time period to the next. More formally the goal is to set

$$\begin{aligned} &W(t)W(t)^T \approx PPMI(t) \\ &\quad \text{s.t. } W_i(t) \approx W_i(t-1) \text{ if word } i \text{ is semantically similar at time } t \text{ and } t-1 \\ &\qquad \text{where,} \\ &PPMI(t) = \max\{PMI(t), 0\}. \end{aligned} \tag{5}$$

$W(t)$ contains word embeddings of size d for each unique word i in the vocabulary across all

[16] My implementation use a code snippet by Ryan Hauser found at: bit.ly/3sZprGh

periods. $PMI(t)$ is the PMI estimated using all documents from period t. I outline the specific objective function I estimate in appendix D.

3.3 Results

In this section I present the estimated word embeddings. First I will present the static embeddings of fitting the SGNS on the entire data set and present a way for the reader to investigate these word embeddings by themselves. Then I will present the dynamic embeddings for selected words using different visualisation techniques.

3.3.1 Static embeddings estimated using the SGNS model

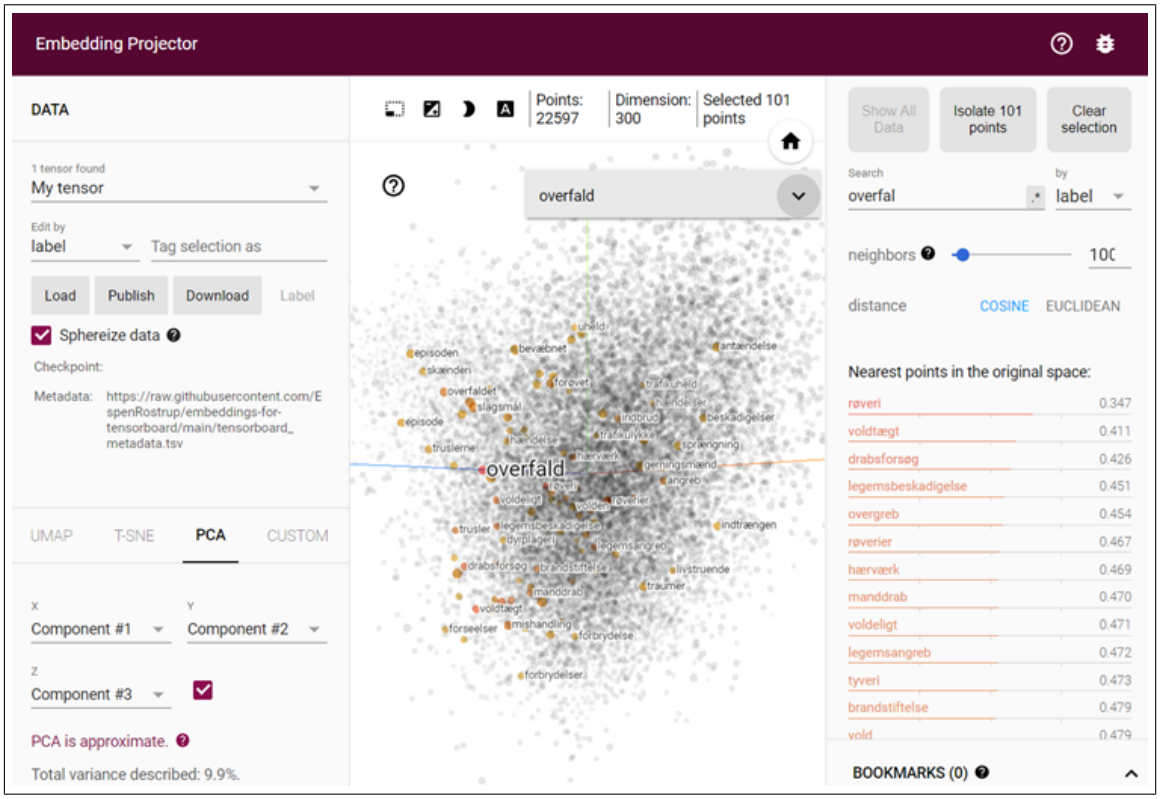
I fit the SGNS model using the Gensim library (Řehůřek & Sojka, 2010). I use a dynamic context window of 5, remove words that appear less than 200 times in the corpus, subsample words that appear more than 7,500 times, a fixed-size word embedding dimension of 300 and 8 negative samples drawn for each word-context pair from the unigram distribution raised to the power of $\alpha = 0.75$.[17] To show the reader and allow interested parties to investigate the embeddings I present an online interactive version of the trained static SGNS word embeddings made with the TensorFlow projector at bit.ly/3PGwSfl (Martín Abadi et al., 2015). A screenshot of this interactive figure is depicted in figure 4.

It is possible to investigate the words' similarity by assessing the cosine similarity of the embeddings using this tool. For instance, the screenshot in figure 4 show that the word *overfald* [trans. assault] most similar words are *røveri* [trans. robbery], *voldtægt* [trans. rape] and *drabsforsøg* [trans. attempted murder]. The tool has several features, including different dimensionality reduction techniques, different metrics for similarities between embeddings, search for words and isolating them in the feature space etc., that are left up to the reader to explore.

To conduct a more formal evaluation of the quality of the embeddings, I check the correlation between the similarity score as measured with cosine similarity with the annotated similarity score of the word pairs in WordSim353-da and DSD as described in section 3.1.5 above. Out of the 99 word pairs in the WordSim353-da data set, only 59 are present in the vocabulary. The Spearman correlation between the similarity score of the humanly annotated score and the cosine similarity is 0.47. This is comparable to the correlation coefficients estimated by Nielsen and Hansen (2017) that lie in the range of 0.42 to 0.52 using models trained on different data sets. Using the DSD dataset, only 59 words are present out of 99 words. The correlation coefficient is 0.22. In comparison Schneidermann et al. (2020) estimate correlations in the range 0.15 to 0.34. Note that these word sets have been made to test the semantic similarity of the general danish language and might not be perfectly suited for evaluating semantic similarity in the legal language, which the lack of words present in the vocabulary is an indicator of.

[17] All trained embeddings are published at github.com/EspenRostrup/ufr-analysis.

Figure 4
A screenshot of the interactive embedding projection



Note: Interact with the visualisation at bit.ly/3PGwSfl.

3.3.2 Training the Dynamic Embeddings

I use the static embeddings from above to initialise the embeddings when training the DW2V. Several other hyperparameters are used to train the Dynamic Word2Vec model, as can be seen in appendix D. I use the hyperparameters that Yao et al. (2018) have found to be optimal for their application except for the embedding space, which I set to 300, and do not fine-tune them. I calculate the PPMIs, which are used as the input data in the DW2V, grouping documents together in 5-year intervals yielding a total of 31 time periods.

For the AW2V-model, I divide the data into six time periods and train the SGNS model using the same hyperparameters as in section 3.3.1 above, except that I only remove words that appear less than ten times *each* period and subsample words that exceed a count of 3,000 in that period. The documents are grouped in the periods: 1867-1919, 1920-1949, 1950-1979, 1980-1994, 1995-2009 and 2010-2021. I have chosen these periods to balance the amount of words across the periods without loosing to much variance in form of the count of documents. In table 2 the document and word count in each time period is shown.

Note that the number of words is relatively low compared to what is usually recommended when using the SGNS model to estimate word embeddings. Having too few observations in each period might yield volatile, low-quality embeddings.

Table 2
Count of words and documents in custom time periods used to estimate the AW2V-model

Time period	Count of documents	Count of words
1867-1919	10,617	10,118,311
1920-1949	16,106	14,189,946
1950-1979	14,097	21,265,434
1980-1994	6,374	10,875,548
1995-2009	9,846	29,138,319
2010-2021	6,875	37,171,902
Total	63,915	122,759,460

The final vocabulary used in the AW2V-model consists of 11,767 unique words. The DW2V model consists of 22,597 unique words. As noted, words need to be present in all periods for them to be included in the AW2V model (section 3.2.3).

3.3.3 The dynamic embeddings for the words *stoffer* and *fod*

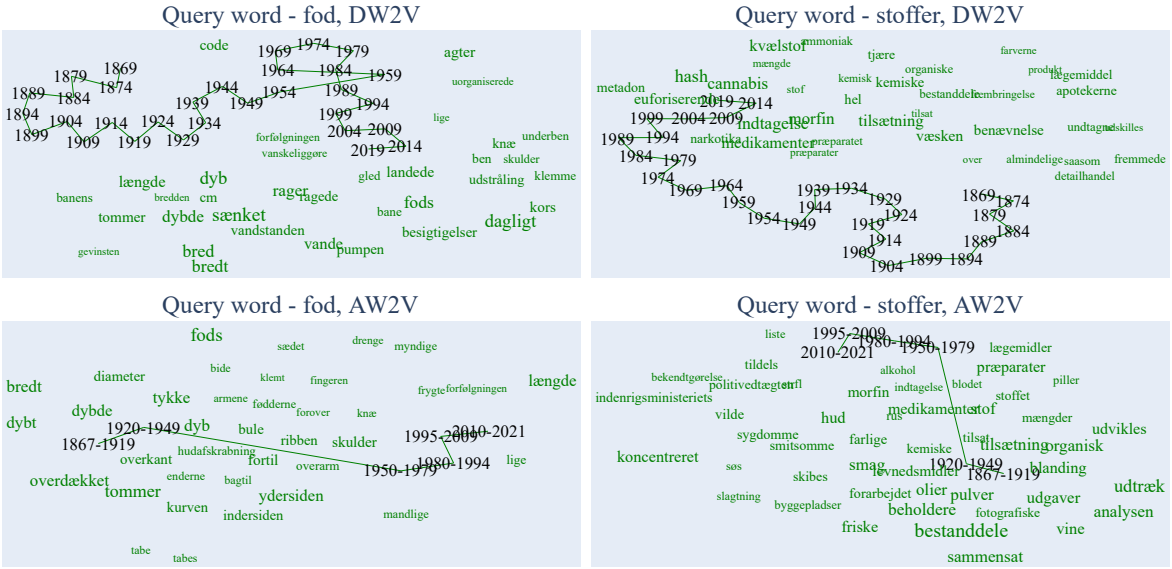
I visualise the semantic drift of a set of Danish words using an algorithm outlined in (Hamilton et al., 2016b:Appendix B). Variations of this visualisation technique is used in several dynamic word embedding papers (Kulkarni et al., 2015; Hamilton et al., 2016a; Yao et al., 2018). The visualisation shows how the word has changed through time, using the most similar words for each period.

For each period, I sample the N closest words to a word, where N can be varied (I have experimented with N = 3,5,10,20). The unique similar words' most recent embeddings are then collected in a list with the selected word's embeddings for each period. Intuitively each similar word is depicted using the most recent embedding since that is the current understanding of the said word. In reality, all word embeddings change over time, which would clutter the analysis if depicted, but should be kept in mind when analysing the results. The collection of embeddings is assigned a data-point in a two-dimensional space using the dimensional reduction technique made for visualising high dimensional data, t-SNE (Maaten & Hinton, 2008).

In figure 5 I depict the dynamic embeddings using this visualisation technique for the word *fod* and the word *stoffer*. Both of these words are polysemous, i.e. they have more than one meaning. The figure depicts both sets of embeddings estimated using the DW2V model and the AW2V model.

The word *fod* was in the 19th century mostly associated with the words *tomme* [trans. inch], *længde* [trans. length] and *cm* highlighting that *fod* was commonly used as a measurement unit. During the 20th century, there was a shift in the word's use so that it became associated with words such as *gled* [trans. slipped], *forfølgningen* [trans. chase], *skulder* [trans. shoulder] and *knæ* [trans. knee]. Hence, the word is less commonly used as a measurement unit and more commonly used to describe the body part. In truth, the word still means both things, but the evolution in the use of the word embeddings is sensible: The metric system was introduced in Danish legislation in 1910 (Carneiro, 2013), why the relevance of foot as a metric unit plummets during the 20th century.

Figure 5
Dynamic word embeddings estimated with the DW2V and AW2V model for the words *fod* and *stoffer*



Note: The DW2V is plotted with the (N=)3 most similar words at a given time using the most recent (2019) embeddings for the list of most similar words. For the AW2V-embeddings, it is the (N=)10 most similar words for each period that is used. The year plotted in the DW2V is the centre year for the query word embedding in a given period, e.g. "2009" is the embedding for the query word in 2007-2011 mapped to the two dimensional space.

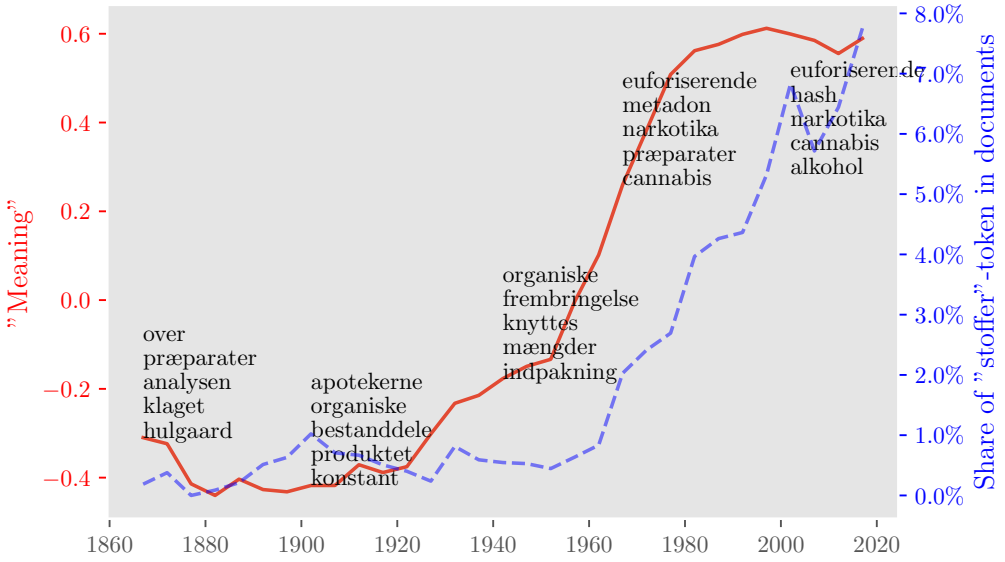
The evolution is clearer when inspecting the AW2V embeddings than the DW2V embeddings. This is most likely due to the wide timespans used to estimate the AW2V-embddings.

The word *stoffer* can mean drugs, cloth, subjects (as in e.g. the subject-matters of a text) and substances. If looking only at the DW2V-embeddings, the embeddings for *stoffer* are closer to words of a more medical nature, such as *lægemiddel* (medicament) and *apotek* (pharmacy) in the 19th century. During the 20th century the word moves toward words containing recreational drugs such as *cannabis*, *narkotika* and *metadon*, however medical terms are still closeby e.g. *medikamenter* implying that the word does not necessarily move away from the previous associations of the word. Looking at the AW2V embeddings these patterns are not entirely alike. Before 1950 the word is mainly associated with (chemical) substances, where the word post 1950 are closer related to drugs e.g. medicaments and alcohol. The difference in the story that can be told using one embedding type over the other highlights the volatility of using a tool such as dynamic word embeddings to understand the evolution of legal language.

Looking into the word *stoffer* more carefully I display a one-dimensional plot containing the evolution of the word's meaning together with the share of documents that contain the word *stoffer* in figure 6.

This figure is inspired by Rudolph and Blei (2018) evaluation of their trained word embeddings. It is a straightforward way of visualising the change in "meaning" over time of a chosen word. I collect all word embeddings for the word *stoffer* using the DW2V embeddings. I then change the basis of the embeddings using principal component analysis (PCA). The embeddings are cast to the first principal component (the component that explains the most variance in the changes in the

Figure 6
Dynamic word embeddings for the word *stoffer* cast to a one-dimensional space



word embeddings over time). This yields a single score for each period. The scores are generously interpreted as "meaning" by Rudolph and Blei (2018), but they do not necessarily reflect that – it is merely the component that explains the most variance in the word embeddings over time. To pinpoint whether changes in the score reflect changes in meaning the most correlated words at different time points are shown in the figure as well.

In the figure, the five most similar words to *stoffer* as measured with cosine similarity are shown in periods 1867-1871, 1902-1906, 1942-1946, 1967-1971 and 2002-2006 together with the evolution of the words meaning for all timespans. One thing to note is how frequent the word appears in a given court document. It is nearly non existent prior to 1950 only appearing in roughly 0.3 pct. of the documents. This might explain some of the noise in the embeddings reflected by e.g. the word *hulgaard* (a name) being the 5th most similar word to *stoffer* in 1867. A second thing to note is the drastic change in meaning starting in the 1950s. This change is probably triggered by the enactment of "Loven om Euforiserende stoffer" (the Euphoriants Act) in 1955 and the increase in recreational drug use that led to the introduction of the law (Houborg, 2011). Identifying when a word change meaning can be formally evaluated using *change-point detection* see e.g. (Kulkarni et al., 2015; Shoemark et al., 2019).

Both the words *fod* and *stoffer* are examples of polysemous words; hence it is not evident whether the word embeddings pick up on the changes in the use of another type of words. In appendix E I review the use of four different types of words using the same visualisation as in figure 5 above: *Sex*, a word that has changed meaning due to a change in orthography, *imødegå*, a contronym, *mand*, a gender/sex, and *overfald*, a type of crime. I find that the shift in embeddings of these words represents reasonable shifts in the use of language by the court.

3.3.4 Which are the words that changed the most?

Identifying interesting words to investigate and operationalise them in a research context is by no means trivial. To have a selection criteria for which words to study, one can assess the words that have changed the most. For instance, Hamilton et al. (2016b) identify the words that have changed the most by calculating the cosine distance between a word's embedding in the earliest period in the corpus and the word's most recent embedding for all words in the corpus. They then evaluate whether the word is identified to have changed due to a genuine semantic shift, a change in the discourse regarding e.g. gender or if it is a product of the type of corpus being evaluated.

I review the ten words that have changed the most as measured by the cosine distance between the estimated AW2V word embeddings in the period 1867 to 1919 and embeddings estimated in the period 2010 to 2021 in appendix F. To evaluate the change in the use of the word, I present a systematic overview of sentences incorporating the words in both periods (table F.2). The change in the word embeddings for these words can be attributed to various factors. They represent 1. a shift in the use of the polysemous words, *omgang*, *ansattes* and *stødende*, 2. the consequence of a more gender-neutral language used in the court, the abbreviation *fr.* and the word *mænd*, 3. a change in orthographic practices, *aa* and *ere*, 4. a change in legal practice, *tæring*, 5. a change in the jurisdiction of the Danish courts, *islandske*, and finally 6. a combination of several factors, *lo*.

Several other metrics could be used to measure the semantic change of words. For instance Shoemark et al. (2019) use several different approaches using e.g. a linear regression framework incorporating all the estimated embeddings for a word and interpreting the estimated slope as the change in the meaning of a word.

3.4 Discussion

One question that I have not tried to answer – but briefly touched upon in my review of static embedding evaluation methods – is what does *meaning* exactly constitute, and is it sensible to say that the drift of word embeddings described above constitutes semantic change? I would say that the word *fod* means the same today as it did in 1867; why the change is not a genuine semantic change. The evolution does instead reflect a change in language use by the court. This leads back to the distributional assumption that words can be known by their neighbours, and to some extent, it is true, but as argued in section 3.1.5, a *cup* and *coffee* are not semantically similar even though they co-occur in many sentences. Hence, for evaluating an actual drift of a word's meaning, the analysis of the change in word embeddings presented here is not sufficient.

I believe that the dynamic embeddings estimated are best used qualitatively, as showcased above, where the evolution of selected words are investigated in coherence with reading use cases for the word and critically reflecting upon why the change in "meaning" occurs. A framework for using dynamic embeddings like this could be the computational grounded theory (Nelson, 2020) as revisited by (Carlsen & Ralund, 2022). The theory highlights the possibilities of using an iterative approach to understand results from unsupervised learning models (which the two models are). Going back and forth between choosing parameters for analysis and having a human-in-the-loop evaluating the quality of the results can yield valuable insights into the text being analysed.

One of the key pitfalls of using the embeddings in research is the potential for researcher bias: Selecting the words for analysis and analysing their meaning through time might be influenced by the researcher's preconceived beliefs. Hence, having clear selection criteria when using dynamic embeddings for analysis and hypothesising about a word's change in meaning *before* examining the dynamic embeddings might increase the validity of the study.

In this thesis, I have, for instance, not motivated the choice of the words *stoffer* and *fod* other than that they are examples of polysemous words. Furthermore, I explain the semantic movement after examination of the embeddings. Hence, I violated my just stated best practice of using the embeddings for analysis: I selected the words with the expectation that the words' dynamic embedding would show meaningful change over time; however, I found the exact timestamps of the shift and the story to go with the change post examination. Other studies have a similar approach, e.g. Yao et al. (2018) show the evolution of the words "Obama", "Amazon" and "Apple" without motivating the selection of the words other than they are expected to have a semantic change. Similarly, Rudolph and Blei (2018) highlights the semantic shifts of "Iraq", "Bush" and "Computer" without much elaboration upon the choice of words. However, common for these studies is that they have other ways of validating their embeddings.

To increase the credibility of the dynamic word embeddings, some sort of objective criteria for what high-quality dynamic embedding constitutes, similar to the evaluation of the static embeddings in section 3.3.1, should be made. Having a similar data set to evaluate the quality of the dynamic embeddings would be ideal, but such data sets are sparse and nonexistent in most non-English languages. An approach to an evaluation strategy equivalent to the evaluation of the static embeddings could be to find words in the danish legal language that have had a semantic change and describe what meaning they are moving towards and (or) away from. This is in line with what Hamilton et al. (2016b) does. The lack of a more streamlined, quantitative approach to validate the embeddings' ability to reflect a change in meaning is a shortcoming of this study.

4 Predicting who pays the cost of a trial

This section addresses the second research question outlined in the introduction. Answering the question contributes to the research field of *legal outcome prediction*. A legal outcome can be a multitude of things. It could be whether a case is allowed to be appealed to a higher court instance (Lage-Freitas, Allende-Cid, Santana, & de Oliveira-Lage, 2019), whether a party is evicted from their property (Vols, 2019) or, for example, whether the U.S. Supreme Court affirms or reverse a lesser court's decision on a case (Alghazzawi et al., 2022). In this thesis, I am concerned with predicting which party is chosen to pay the cost of a trial in a court case.

4.1 Legal outcome prediction

There is a vast amount of literature motivating the problem of legal outcome prediction differently. Medvedeva, Wieling, and Vols (2022) presents the current state of the legal outcome prediction literature. They divide the research field into three categories that are convenient for assessing the applications of the studies within the field of legal outcome prediction: A category with studies

that *identify* the legal outcomes, a category of studies that *categorise* legal outcomes and finally, a category of studies that tries to forecast the legal outcomes of interest.

The division between studies that identify or categorise legal outcomes and studies that forecast the outcomes is straightforward: Studies that forecast legal outcomes are based on features produced *prior* to the observed legal outcome. For instance, Waltl, Bonczek, Scepankova, Landthaler, and Matthes (2017) predict the future appeal decisions of the German federal fiscal court based on features extracted from the rulings of the lesser court instance; hence, the information used to predict the outcome is known prior to the commencement of the appeal case.

Studies that *identify* legal outcomes find the outcome in the textual features of documents. In the first part of this study, this is what I strive to do: I find the party who pays the cost of a trial using the text in the ruling. Medvedeva et al. (2022) motivates the need for studies identifying legal outcomes as a need to generate more descriptive data for legal cases. The studies that *categorise* outcomes enable analyses of the processes leading to the relevant legal outcome. The studies predict the outcome in the same way as when forecasting an outcome but remove references to the outcome so that the classifier does not just identify a feature that is, in fact, the outcome. The features that the classifier finds significant for classification can then be evaluated. This method can give insights into which features are correlated with a particular outcome. It is, however, hard to imagine that any causal inference can be made credibly using this method – I have at least not been able to find such a study.

The difference between identifying and categorising legal outcomes is subtle and depends a lot on the researcher's ability to mask the outcome of interest. For instance, Sulea, Zampieri, Vela, and van Genabith (2017) predicts the verdicts of the French Court of Cassation. They find the outcome as a range of different verdicts in the text of the ruling, and afterwards, they use the text of the ruling to predict the outcome they found. When training the classifier, they try to mask the outcome by removing different phrases and words related to it. Medvedeva et al. (2022), however, argues that the outcomes are not sufficiently "masked" from the text used for prediction, which is why they classify this study as an "identification of legal outcomes"-study and not "categorisation of legal outcomes".

To predict what the outcome of a given court case is in this corpus of rulings from the UfR, I use the following three steps:

1. I Identify the legal outcome of interest.
2. I prepare the text data and train a linear support vector classifier on a subset of the available data.
3. I evaluate the classifier's performance using the subset of data on which the classifier was not trained.

I will outline each of the steps in the sections below.

4.2 Identifying the legal outcome: The cost of a trial

As a rule of thumb, the winning party's trial costs is reimbursed by the losing party. This has been the case, at least since the introduction of Retsplejeloven in 1916 (Justitsministeriet, 1916; Espersen, 2005). Hence, the assignment of trial costs can be seen as an incomplete proxy of who wins a court case. This distinction is important because most of the rulings contained in the corpus are not concerned with the assignment of the cost of a trial but the case itself. Therefore, I suspect that the performance of a classifier predicting who pays the trial costs will hinge on its ability to identify who won the court case. The quality of this assumption can be assessed by evaluating the predictors (words or combination hereof) importance for classification.

The assignment of cost of a trial in non-criminal cases is regulated in Retsplejeloven (RPL) §§ 312-316 (Justitsministeriet, 2021). There are several exceptions where the losing party should not pay the cost of the trial for the winning party.[18] One key exception is that the court can waive the obligations for the losing party to pay the trial costs if the case is of a "principiel" or "videregående" [Trans. fundamental or more extensive] character. The UfR editors select the rulings to include in the journal by their prejudicial value. Hence, the requirement for a party to pay the trial costs is expected to be waived more frequently for UfR rulings than what would be expected in a representative sample of trials.

The trial costs for a criminal case are regulated in RPL §§ 1007-1014 (Justitsministeriet, 2021). The public authorities pay the cost of a trial as a general rule and are reimbursed the trial costs if the defendant is found guilty of the charges. Therefore, for non-criminal cases, there are three possible outcomes, the plaintiff pays the cost of the trial, the defendant pays the cost of the trial, or no single party pays the cost of the trial. For a criminal case, there are only two possible outcomes, the public authorities pay the cost of the trial, or the defendant pays the cost of the trial. The distinction between criminal and non-criminal cases will be elaborated upon below in section 4.2.3.

4.2.1 Pattern matching to find who pays the cost of a trial

The assignment of trial costs is not recorded as metadata to a given ruling. Hence, I use *pattern matching* to find the party who is sentenced to pay the trial cost. Pattern matching allows checking whether a "textual pattern" is present in a text. This "textual pattern" could be a word, but it could also constitute a logical rule set, e.g. if the word "not" is present, do not match the word "violation"; otherwise, match the word "violation".

It is common in the legal outcome prediction literature to use pattern matching to identify the legal outcome of interest. However, the difficulty of obtaining the correct labels, i.e. the actual legal outcome, varies from corpus to corpus. Some studies only need to search for a single keyword, e.g. *violation* or *no-violation*, (Medvedeva et al., 2021) to find the outcome of interest or within a section of text looking for the type of verdict such as "cassation", "rejection", "cancellation" etc. (Sulea et al., 2017). Other studies use multiple keywords combining, e.g. "affirmed" and "guilty" as one label category and "reversed" or "innocent" as another label category in the prediction of the Phillipino Supreme Courts verdict (Virtucio et al., 2018) or the case of the German Fiscal Courts, where the authors use "several selected terms" (Waltl et al., 2017:6) to find the outcome of the

[18] For a detailed description, see the preparatory works for the change of RPL §§ 312 in 2005 (Espersen, 2005)

ruling using only the first part of the document. Some papers mention they use pattern matching to find the outcome without specifying the approach (Malik et al., 2021; Strickson & De La Iglesia, 2020).

The pattern matching done in this thesis is quite complicated since the court rulings do not produce a homogeneous pattern related to the assignment of cost of trial. Furthermore, just retrieving the entity name (the name of the party that pays) does not cut it since I want to map the entity's name to the defendant party or to the prosecutor/plaintiff (criminal cases/non-criminal cases). Hence, I outline my approach to identifying the trial's outcome in detail below. The quality of the labels is evaluated by comparing the automatically generated labels with a manually encoded random sample of 5 pct. of the rulings encoded by an external party.

4.2.2 Creating the labels step-by-step

To identify the labels, I use *regular expressions*. Regular expressions enables pattern matching in text using algebraic operations (Aho & Ullman, 1992:556). The operations of algebraic expressions can be quite convoluted. I use the RE module in PYTHON and refer to the documentation for a detailed review of regex notation (van Rossum, 2022). My proposed process of identifying label outcomes in non-criminal cases is sketched in a flow chart in figure 7. The process is slightly different for criminal cases due to the difference in possible outcomes. I will briefly explain the different steps and their motivation from top to bottom. The numbers in the list below follow the numbers in the figure.

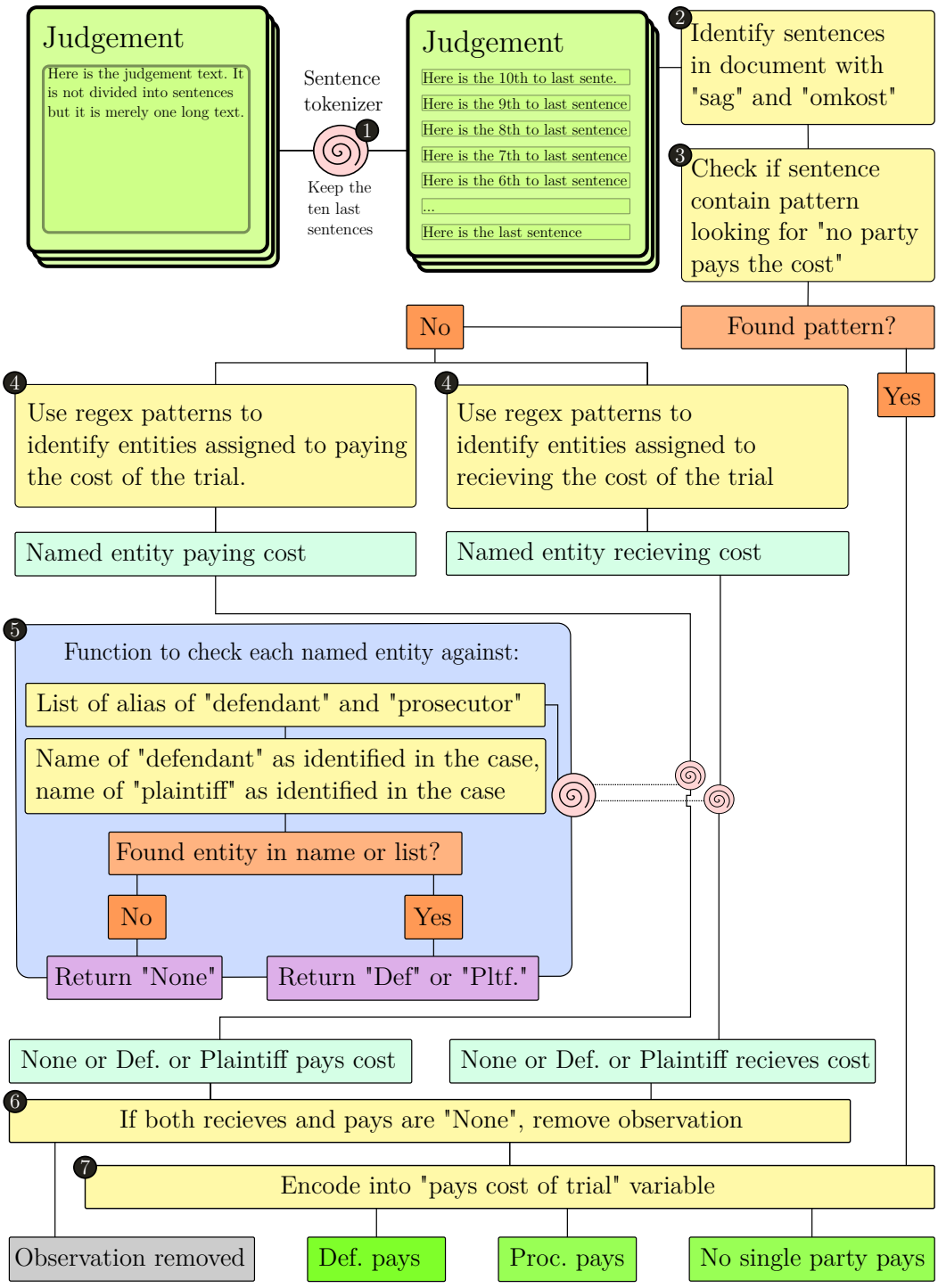
1. The rulings are divided into sentences using the pre-trained NLTK sentence tokeniser in the same way as in section 2.2.2 (the methodological reservations regarding the tokeniser discussed in section 2.2.2 also apply here). Some rulings contain the ruling of a lesser court instance in full. The lesser court ruling might include an assignment of trial costs as well. To avoid wrongly identifying the lesser court's trial cost assignment, I only consider the last ten sentences in the ruling. If the assignment of the trial costs is present in a ruling, it is nearly always at the end of it.

2. I only keep rulings where at least one of the last ten sentences include the pattern "sag" [trans. trial] or "omkost" (stemmed version of "omkostning" which translate to cost). The patterns are case insensitive and are allowed to be part of other words, implying that both the sentence forms "sagens omkostninger betales af..." and "sagsomkostninger udredes af..." are kept. This step reduces the corpus quite substantially since many of the rulings do not contain indications of who bears the cost of the trial.

3. I search for patterns associated with the label "no party pays the cost of trial". These patterns are listed in appendix G at the topmost rows of table G.1. If such a pattern is found, the ruling is encoded "no party pays the cost of trial".

4. I identify the entity who respectively paid and received the trial cost. This step requires the most sophisticated use of patterns since entities can be found in sentences in numerous ways. For an evaluation and justification of each of the regex patterns used to identify the entity, I refer to appendix G.

Figure 7
Process of identifying who pays the cost of a trial



5. Using this function, I identify whether the entity is the plaintiff or the defendant. The name of the entity might be the name of the plaintiff or the defendant but can also refer to the party´s role in a given type of court case, e.g. "tiltalte", "indstævnte", "sagsøgte" for the defendant in respectively a criminal case, an appeal court case and a civil court case and "appelanten" and "sagsøger" for the plaintiff in respectively an appeal court case and a civil court case. Since the name of the parties might include the name of their legal representatives or be slightly different to the entity named in the ruling, I also check whether the named entity found is just a part of the defendant's or plaintiff's name. If there is no match between the entity and the name or role of the parties, the function returns "None".

6. If both the paying and receiving party is "None", as identified above, the ruling is removed from the corpus.

7. The ruling can now be encoded into the three labels.

The sentences identified in step 2 are kept to "mask" the output when predicting who pays the trial cost, i.e. they are removed from the text used in the classifier below. The idea is that the classifier should not predict my pattern matching strategy, i.e. the classifier should rely on features that do not explicitly state the assignment of the cost of the trial.

Note that I also use which party recieves the cost of trial as to find who pays the cost of trial. In some non-criminal cases the defendant or prosecutor might be granted "fri proces" [trans. free legal aid] where the state will pay the cost of the trial if the party granted this loses. However, these cases will be removed in step 5, since the name of who pays the cost of trial will not coincide with any of the party names.

4.2.3 Remark on the difference in labelling between criminal and non-criminal cases

A notable exception to the labelling process outlined above is when the ruling regards a criminal case. "Anklagemyndigheden"/"Rigsadvokaten" (the state prosecutor) or lawyers acting on their behalf will, no matter if it is an appeal case or not, be denoted as the prosecutor in my data set. Furthermore, if the prosecutor loses the case, it will be "det offentlige" [trans. the state] (pre-1970) or "statskassen" [trans. the state treasury] (post-1970) who pays the cost of the trial. Hence, I will define all cases where the state pays the cost of the trial as the prosecutor paying the trial. This is problematic since I disregard a potential overlap of reimbursement of costs by the state unrelated to the prosecutor's office, some specific to only criminal cases defined in, e.g. RPL §1008 (Justitsministeriet, 2021). Furthermore, the state prosecutor will be labelled as prosecutor even if it is an appeal case, where the defendant in the lesser court instance appealed the case. Hence, to explore the impact of this crude generalisation of cost assignment to the state prosecutor, I train two classifiers predicting only criminal cases and non-criminal cases. Note that I define a criminal case as a case with a public prosecutor. In reality, it is not *all* criminal cases where there is a public prosecutor. In a few types of criminal offences, e.g. defamation, the aggrieved party will be able to act as the prosecutor (Lundum, 2021). Hence, my distinction between criminal and non-criminal cases is a division of cases in criminal cases pursued by public authorities and cases that are not.

4.3 Text as predictors in a linear support vector classifier

There are numerous ways of predicting who pays the cost of the trial, given the labels constructed above and the text from the ruling. Newer approaches involve deep learning models such as BERT (Chalkidis, Fergadiotis, Malakasiotis, Aletras, & Androutsopoulos, 2020; Medvedeva et al., 2022) (see section 3.1.3). However, the goal of this thesis is not to provide state-of-the-art classification results but merely to show that it *is* possible to predict who pays the cost of a trial, why I use a very simple classifier. I will first explain how the documents are represented in numeric form and then describe the linear support vector classifier I use to predict the assignment of the trial costs.

4.3.1 Document embeddings

The first hurdle when constructing a classifier based on textual data is the numeric representation of the documents analogue to the word embeddings discussed in section 3 above.

A simple document representation is the term (word) frequency, i.e. the number of times a word appears in a document. The document embedding matrix is then $TF \in \mathbb{R}^{d,v}$ where d is the document count, v is the number of unique words across documents and the values are the frequencies of the word in the document. This representation, however, does put arguably too much weight on uninformative words such as *og* [trans. and], den [trans. that], *er* [trans. am/is/are], etc., that appear multiple times in almost all documents. One can weigh the term frequency by the inverse document frequency of the term to handle this. The term-frequency inverse document frequency is commonly used in classification tasks using texts (Medvedeva, Vols, & Wieling, 2020), and it is also the document embedding I use for the prediction task at hand. I calculate the inverse document frequency for each term as

$$idf(t) = \log\left(\frac{1+n}{1+df(t)}\right) + 1 \tag{6}$$

where n is the count of documents and $df(t)$ is the count of documents that contains the given term, t. Hence a relatively rare term weighs more than a frequent term since $idf(t)$ decreases in $df(t)$. Since log(1)=0, 1 is added to give some weight to very frequent terms.[19] Each element in the tf-idf matrix (documents as rows, terms as columns) can be calculated as

$$tf\text{-}idf(d,t) = tf(d,t) \cdot idf(t) \tag{7}$$

where $tf\text{-}idf \in \mathbb{R}^{n,v}$.

[19] This is the standard implementation in SCIKIT-LEARN, which I use to construct the tf-idf. (Pedregosa et al., 2011).

4.3.2 Preprocessing the text

Before calculating the tf-idf matrix, selected preprocessing steps are taken. The prepossessing steps I use are inspired by Medvedeva et al. (2020) that use the same method to predict whether an article of the European Convention of Human Rights was breached using rulings from the European Court of Human Rights. The steps I consider specifically are:

- Different combinations of *n-grams*. The n is the number of tokens (words) that are considered a single feature in combination with one another. Hence, the "the fox jumps over dog" sentence presented in uni-grams is the list of words "the", "fox", "jumps", "over", "the","dog", and in bi-grams "the fox", "fox jumps", "jumps over", "over the", "the dog". I consider combinations of n-grams in the tf-idf up to 4-grams. Note that combining, e.g. 1-,2-,3- and 4-grams in a single tf-idf implies having many features per document.
- pruning the vocabulary to only include grams that appears at least a certain amount of times
- removing "stop words"; a list of words that are common to the danish language such as "og", "i", "jeg", "det", "at", "en", "den", "til". I use the 94 stopwords collected from the NLTK package (see appendix H for the complete list).

Each of these preprocessing steps has implications for the accuracy of the classifier. I will use 5-fold cross-validation to select which prepossessing steps yield the best performing classifier described in section 4.4.1 below.

4.4 The linear support vector classifier in a multiclass setup

The linear support vector classifier finds a linear function (or a hyperplane, to be more exact) that separates two classes given some input data. The dimension of the input data's feature space can be greater than the number of observations available (Vapnik, 1997). The *optimal* hyperplane is the one that maximises the distance of the *margin* between the two classes. In a perfect separable case, the margin is the distance from the hyperplane to the observation closest to it from each class.

However, in this case, and most other use cases, classes overlap in the feature space and are therefore not perfectly divisible. The soft margin support vector classifier allows for overlap introducing a slack variable denoting how much the classification lies on the wrong side of the margin. The slack is measured proportional to the margin. A slack of zero for a predicted observation implies that the observation lies on the right side of the hyperplane, a slack between 0 and 1 implies that the observation crosses the margin but is still on the right side of the hyperplane and a slack greater than one means that the observation is misclassified being on the wrong side of the hyperplane. The sum of slack is bounded by a constant, C, hence putting an upper limit on the total misclassification allowed (proportional to the distance to the margin) (Hastie, Tibshirani, & Friedman, 2009; James, Witten, Hastie, & Tibshirani, 2021:Ch. 9.2).

Finding the optimal margin allowing for slack can be stated as a quadratic problem with constraints solved using Lagrange multipliers. The solution shows that the margins, and hence the

hyperplane, are defined by the observations on or on the wrong side of the margin. These observations are called the *support* vectors. This is noteworthy since removing or altering observations that are correctly classified will not impact the estimation of the hyperplane and hence the classifier (as long as the alterations of its features do not leave them misclassified).

Since C controls the amount of misclassification tolerated, it represents a way of controlling the bias-variance trade-off. If C is small, the hyperplane is fitted locally using few observations implying a high *variance* (risk of overfitting) but a relatively small *bias* since it fits the training data well. On the other hand, using a large C implies that the model might generalise better (low variance) due to being more robust to the introduction or alteration of data points. However, it might have a higher bias since it is not fitted as well to the training data set (i.e. "allowing" for more misclassification) (Hastie et al., 2009; James et al., 2021:Ch. 9.2). Hence, C-parameter tuning is important for the classifier's performance.

Since the linear support vector classifier is a binary classifier, there are two main approaches to expanding it to a multiclass classification problem: *One-vs.-one* (OVO) and *one-vs.-the rest* (OVR). The OVO classifier takes each class and compares it with each of the remaining classes; hence for a 3 class classification problem, there are essentially estimated (3*2=)6 hyperplanes to separate the classes. The OVR classifier estimates a hyperplane for each class, evaluating whether they are in the given class or not. I use the OVR classifier since implementations of the linear support vector OVR classifier in the library SCIKIT-LEARN are much faster than the linear OVO classifier (disregarding the extra complexity of the OVO).

4.4.1 Tuning and evaluating the classifier

Altering the preprocessing steps and the C-parameter in the linear support vector classifier affects the classifier's performance. To select the hyperparameter C and the different preprocessing steps to perform, I use 5-fold cross-validation.[20] For each combination of hyperparameters and prepossessing steps, I split the data into five folds of equal size and fit the linear support vector classifier using the data from four out of five folds. The last fold is used to evaluate the classifier's accuracy. Using cross-validation for parameter selection is a way to reduce the risk of overfitting since the parameters are not evaluated using the data it is trained upon but on a validation set – the 5th fold. A greater amount of folds should imply that the error rate would be even lower since the validation fold would be smaller, and more data would be used to train the classifier. However, the classifier will probably also generalise worse to non-training data why the choice of folds can be seen as representing the bias-variance trade-off (James et al., 2021:Ch. 5).

Before tuning the classifier, the data will be separated in a training and test set (unrelated to the cross-validation above) using an 80-20 split, i.e. 20 pct. of the data is not used in training the classifier. Using a test set is standard practice when evaluating a classifier's performance since it is easy to train a model that perfectly fits the training data. I.e. using data that the model has not seen before provides a better picture of its performance. I will present the classifier's *accuracy* and *macro F1-score* for the test data set. The accuracy of a classifier is the ratio between the count of true classifications (assignment of trial costs is correct) and the number of observations in the data set. The F1 score can be interpreted as the mean of the classifier's *precision* and *recall*. Precision

[20] I consider the C-values 0.1, 0.5, 1 and 5.

is the fraction of correct classifications into a given class, and recall is the fraction between the count of observations classified into a class correctly and the count of observations in the class. The *macro* F1-score is the unweighted mean of the F1-score calculated for each class.

4.5 Results

First, I will describe the labels as found using the methodology outlined in section 4.2 above, and then I will show the classification results.

4.5.1 The resulting labels

Before applying the processing steps outlined in section 4.2 above I apply some restrictions on the corpus I classify. First, I only consider court judgements and not court decisions (see section 1.3 for distinction) and only court cases from 1950 until now, reducing relevant documents from 63,915 to 27,817. Secondly, I remove cases where more than two parties are identified or where the parties could not be found in the document. The classification outcome for multiple parties in a single case might be undefined if, e.g. the assignment of the trial costs differs between the defending parties.[21] This reduces the number of rulings from 27,817 to 22,478. Out of the 22,478 documents, the terms "sag" and "omkost" appears in 15,525. I classify 13,871 of these cases (89 pct.).[22]

The distribution of labels is shown in table 3. In 33 pct. (4,570) of the rulings, the defendant party pays the cost of the trial, in 50 pct. of the rulings, the prosecuting party pays the trial costs and in 17 pct. of the rulings, no single party pays the cost of the trial. In appendix I, I present the distribution of labels across the different court instances in the corpus. An interesting insight is that the plaintiff is more likely to pay the cost of a trial in the supreme court for non-criminal cases than in the high courts. This could be explained by the case having at least been processed once before in one of the high courts and possibly also in the district courts. However, the documents I label are not a representative sample of the rulings in the UfR and not a representative sample of an out-of-sample court case altogether, as argued in section 2 above. Hence, conclusions drawn using the composition of labels shown above should be made with care.

To evaluate the quality of the labels, I got a first-year law student to encode which party pays the cost of the trial for 694 of the rulings I was able to classify (5 pct. of the 13,789 classified rulings). The 694 rulings were selected using stratified random sampling, ensuring that the distribution of the labels shown in table 3 are the same in the random sample that the annotator encodes. The annotator was instructed to follow the same rules for annotating respectively criminal cases and non-criminal cases as I describe in the section 4.2.3. I find that 98.8 pct. of the rulings (686

[21] In practice, this is difficult since the parties are retrieved from the document (see section 2.1). More specifically, I remove cases where there is more than one "conflict" described in the party sentence (number 2 in figure 1). If the word "mod" [trans. versus] appears multiple times there are multiple conflicts in the ruling. Furthermore, I remove cases where there is more than one set of brackets ("(", ")") in the defendant's party name. The bracket-pair count indicates the count of legal representatives for the party since the legal representative(s) for a party (usually) is enclosed in brackets. Multiple sets of legal representatives indicate multiple parties where the assignment of cost for each defendant might be different. Note that the defending parties might have the same representative for some cases (especially criminal cases), but the verdict and the assignment of cost to the parties might differ. Alas, this is not a bulletproof way of removing multiple parties.

[22] The classification of the 13,871 judgements is published at the GitHub-repository.

Table 3
Distribution of legal outcome by case type

Who pays?	Type of case: Criminal	Non-criminal	Total
Defendant	1,513	3,421	4,570
Pros./plaintiff	1,295	5,224	6,883
No single party	0	2,418	2,418
Total	2,808	11,063	13,871

out of 694) are correctly classified when comparing the manually encoded rulings to the labels I construct.[23] In line with Medvedeva et al. (2022) I argue that the identifying of labels is only really useable if the resulting identification has an accuracy of close to 100 pct., since one could just read the ruling to obtain the correct information. 98.8 pct. is close but suggests a small room for improvement since it should be possible to find patterns that identify the party correctly for all rulings. I note that the annotator only evaluated rulings where I could automatically locate a sentence related to the trial costs (step 3 in figure 7). A more rigorous approach to evaluating the labels could include assessing my process of identifying rulings that contain the assignment of trial costs.

4.5.2 Prediction results

I fit two linear support vector classifiers for the training set using the original and masked version of the training data set using cross-validation. The accuracy and F1-score of the classifier evaluated on the test set are reported in table 4. Furthermore, I report the results of a naive classifier that predicts the most observed class at all times independent of the features (hence producing the same results for the masked and the not-masked data). The test set contains the same rulings for evaluating the not-masked and the masked data set.

Table 4
Prediction results: Who pays the cost of the trial? Comparing masked and not masked data.

Classifier	Not-masked		Masked	
	Accuracy	F1-score	Accuracy	F1-score
Linerar SVC	87.72	87.35	69.06	61.78
Most frequent classifier	38.80	33.74	38.80	33.74

The Linear SVC performs significantly better than the naive classifier; hence the model at least learns some patterns from the features included in the tf-idf. It is seen from the table that the accuracy is 87.72 using the tf-idf based on the unmasked data and conversely 69.06 using the tf-idf based on the masked data. The F1 score is slightly lower than the classifier's accuracy trained on the masked data set. This reflects that the classifier especially has trouble with predicting the class "no single party pays the cost of a trial" but is more accurate when it comes to predicting whether it is a defendant or prosecutor/plaintiff who pays the cost of trial. Since there is usually no direct

[23] The manually encoded labels and the ID of the ruling is published at the GitHub-repository.

Table 5
Features with a large positive estimated decision function coefficient for the class "plaintiff/prosecutor pays the cost of the trial"

Not masked	**Masked**
frifindes sagsomkostninger	appellantens påstand tage følge
stadfæstes tiltalte	appellantens påstand tage
højesteret udredes tiltalte	sagen stadfæstes
begge retter betaler indstævnte	dom sagen stadfæstes
afsigelse betale	findes passende
pålagdes sagsøgte	findes indstævnte
dage højesteretsdoms afsigelse betale	tiltale indstævnte
højesteretsdoms afsigelse betale	stemmeflertallet kendes ret indstævnte
indstævnte appellanten	tage følge
omkostninger højesteret betales tiltalte	sagsøgerens påstand

indication of the court's processes of assigning trial costs the classifier struggles with predicting exceptions to the general rule (the losing party pay the trial costs).

The difference between using the data as-is and the masked data is expected: The labels are generated using the sentences which are removed from the original rulings, hence removing the sentences worsen the performance of the prediction task. To illustrate this, I have depicted the features (n-grams) with the largest associated positive coefficients for the outcome "plaintiff/prosecutor pays the cost of trial" in table 5. The coefficients are used in the decision function (the hyperplane) for classification: If a coefficient is positive, the feature *increases* the probability of the ruling being classified as "plaintiff/prosecutor pays the cost of the trial" and conversely if negative. Using the tf-idf based on the raw data, the n-grams with the largest positive coefficient are related to the cost of a trial, e.g. "begge retter betaler indstævnte" [trans. both courts pay the defendant]. On the other hand, the model trained using tf-idf based on the masked data has the largest positive coefficient associated with terms related to the court verdict, such as "sagen stadfæstes" [trans. the case uphold] and "appellantens påstand tage følge" [trans. the appellant's claim upheld].

It can also be noted from the table that the features included in the tf-idf of both estimated models contain up to a 4-gram. The hyperparameters and preprocessing steps found by the 5-fold cross-validation are reported in appendix J. They are nearly identical across the classifiers estimated; the tf-idfs are mostly based on 2-, 3- and 4-grams that appear at least 3 times in the training data set where stopwords have been removed. The hyperparameter found to produce the best accuracy with cross-validation is $C = 5$ across the classifiers.

As motivated in section 4.2.3 above, I train separate models for criminal and non-criminal cases. I only train the classifiers using the tf-idf matrix using the masked data.

The classifier trained on criminal cases is more accurate than for the non-criminal cases, which is natural since it is a binary classification problem. The classifier trained on non-criminal cases' accuracy and F1-score increased slightly when removing the criminal cases. Hence, including information on the type of case generally improves the classifier's performance.

The general performance of all three classifiers trained on the masked data set is lower than other comparable legal outcome prediction literature. Medvedeva et al. (2022) presents an overview

Table 6
Prediction results: Who pays the cost of trial? Comparing criminal and non-criminal cases with masked data.

Classifier	Criminal		Non-criminal	
	Accuracy	F1-score	Accuracy	F1-score
Linerar SVC	74.56	74.23	70.90	63.71
Most frequent classifier	51.25	51.56	39.63	34.19

of 27 papers' classifiers' accuracy or F1-score on different legal outcome prediction tasks, and most of the classifiers that are not concerned with forecasting outcomes have an accuracy or F1 score above 75. However, the prediction tasks are widely different across studies, with varying legal outcomes, size of corpus and features accessible. Most likely, the worse performance from my models reflects that the outcome I use is not the direct outcome of the court case but a derivative of it.

4.6 Discussion

The results suggest that it is possible to estimate who pays the trial costs given the text in a ruling, but not very accurately. Assessing the most important predictors, it is evident that the classifier relies on identifying the court verdict. This correlation is not a very remarkable insight into the workings of the law since the law dictates the assignment of the cost of trial given the outcome of the verdict. To improve the study without changing the research setup too much, one could try to mask the outcome differently so that the entire verdict was removed instead of just removing the sentence related to the assignment of cost.

The Ugeskrift for Retvæsen contains many types of cases where there might be different legal practices in the assignment of costs. I handle one of the most apparent distinctions by examining the classification of criminal cases and non-criminal cases separately. However, non-criminal cases are very heterogeneous as shown in section 2.2, so the practice and regulation in the assignment of trial costs might differ. For instance, in cases concerned with family law, no single party pays the cost of the trial no matter who "wins" or "loses" the case unless particular circumstances favour it (Justitsministeriet, 2021:RPL §312 pt. 7).

Many tweaks could be made to increase the classifiers' performance. One might consider oversampling more infrequent classes in training, different preprocessing steps and hyperparameter choices or using a different classifier altogether. Even though interesting from a Machine Learning perspective, increasing the performance might not have many other applications. As Medvedeva et al. (2022:12) reasons, legal outcome prediction should be based on the needs of the legal community; and forecasting a derivative outcome as the assignment of trial costs is might not have many use cases for legal professionals. Hence, I believe that the most significant contribution to this community is my identification of who pays the cost of a Danish trial.

5 Closing remarks

In this thesis, I collect and analyse rulings from the journal Ugeskrift for Retvæsen. I outline two research questions that are very different; hence the thesis has been divided into two parts: I estimate dynamic word embeddings to investigate the language use of the Danish courts, and I predict the assignment of the cost of a trial using only the text from the court rulings.

I find that it is possible to explore the evolution of the language used by the Danish courts using dynamic word embeddings. Albeit, I do not provide substantial evidence for the dynamic embeddings to be credibly used for generalising semantic movements in the legal language; for such statistical aggregations, further exploration into the quality of the embeddings is required. I do not operationalise the embeddings to answer any questions about culture or language, but I publish them so that future work can use them. Research applications using embeddings could involve tracing legal concepts of interest through time equivalently to how Vylomova et al. (2019) explore the use of harm-related concepts in psychology through time or investigation into the dynamics of legal culture equivalent to Kozlowski et al. (2019). As (Kutuzov, Øvrelid, Szymanski, & Velldal, 2018:9) state, the dynamic embeddings contain an immense amount of information buried "between the lines" and it is up to the researchers to dig it up.

Furthermore, I find it is possible to predict which party has to pay the cost of a trial in a court case using court rulings where sentences related to the assignment of trial costs have been removed. I find that the predictors that matter the most are related to the actual court verdict. Since the trial costs are regulated by law to be assigned to the losing party of a court case, this is not a very novel insight. I argue that the most significant contribution from this part of the thesis is the generation of new metadata that can be attached to a court ruling, namely, who pays the cost of the trial. A follow-up study using my approach to identifying who pays the trial costs could be to *forecast* the assignment of the cost of the trial. Such a study would be possible using the newly released domsdatabasen.dk, where the entire "case-string" is published for the judgements that are included in the database, i.e. all previous judgements of the same case are included in separate documents. As of May 2022, the database is not big enough to conduct such a study, but hopefully, it will contain a substantial amount of court rulings in the near future.

References

Aho, A. V., & Ullman, J. D. (1992). Chapter 10. Patterns, Automata, and Regular Expressions. In *Foundations of Computer Scienc.* Retrieved 2022-04-27, from http://infolab.stanford.edu/~ullman/focs.html

Aletras, N., Tsarapatsanis, D., Preoţiuc-Pietro, D., & Lampos, V. (2016, October). Predicting judicial decisions of the European Court of Human Rights: a Natural Language Processing perspective. *PeerJ Computer Science*, *2*, e93. Retrieved 2022-05-30, from https://peerj.com/articles/cs-93 (Publisher: PeerJ Inc.) doi: 10.7717/peerj-cs.93

Alghazzawi, D., Bamasag, O., Albeshri, A., Sana, I., Ullah, H., & Asghar, M. Z. (2022, January). Efficient Prediction of Court Judgments Using an LSTM+CNN Neural Network Model with an Optimal Feature Set. *Mathematics*, *10*(5), 683. Retrieved 2022-04-26, from https://www.mdpi.com/2227-7390/10/5/683 (Number: 5 Publisher: Multidisciplinary Digital Publishing Institute) doi: 10.3390/math10050683

Andersen, M. B. (2017). Ugeskrift for Retsvæsen gennem 150 år. *Ugeskrift for Retvæsen*. Retrieved 2022-03-21, from https://www.karnovgroup.dk/ufr150/artikler/ugeskrift-for-retsv%C3%A6sen-gennem-150-%C3%A5r

Andersen, M. H. (2016, September). Om bogstavet x. *Nyt fra Sprognævnet*, *3*. Retrieved 2022-05-26, from https://dsn.dk/wp-content/uploads/2021/01/september-2016-pdf.pdf

Andersen, P. (2015, July). Chapter 2. Kancellisprogets fødsel. In *Retten i sproget: Samspillet mellem ret og sprog i juridisk praksis* (pp. 33–44). DJØF Forlag.

Bakarov, A. (2018, January). *A Survey of Word Embeddings Evaluation Methods* (Tech. Rep. No. arXiv:1801.09536). arXiv. Retrieved 2022-05-17, from http://arxiv.org/abs/1801.09536 (arXiv:1801.09536 [cs] type: article)

Bamler, R., & Mandt, S. (2017, July). Dynamic Word Embeddings. *arXiv:1702.08359 [cs, stat]*. Retrieved 2022-04-25, from http://arxiv.org/abs/1702.08359 (arXiv: 1702.08359)

Carlsen, H. B., & Ralund, S. (2022, January). Computational grounded theory revisited: From computer-led to computer-assisted text analysis. *Big Data & Society*, *9*(1), 20539517221080146. Retrieved 2022-05-09, from https://doi.org/10.1177/20539517221080146 (Publisher: SAGE Publications Ltd) doi: 10.1177/20539517221080146

Carneiro, K. (2013). *metersystemet.* Retrieved 2022-05-09, from https://denstoredanske.lex.dk/metersystemet

Chalkidis, I., Fergadiotis, M., Malakasiotis, P., Aletras, N., & Androutsopoulos, I. (2020, November). LEGAL-BERT: The Muppets straight out of Law School. In *Findings of the Association for Computational Linguistics: EMNLP 2020* (pp. 2898–2904). Online: Association for Computational Linguistics. Retrieved 2022-03-25, from https://aclanthology.org/2020.findings-emnlp.261 doi: 10.18653/v1/2020.findings-emnlp.261

Chalkidis, I., & Kampas, D. (2019, June). Deep learning in law: early adaptation and legal word embeddings trained on large corpora. *Artificial Intelligence and Law*, *27*(2), 171–198. Retrieved 2022-03-25, from http://link.springer.com/10.1007/s10506-018-9238-9 doi: 10.1007/s10506-018-9238-9

Christensen, M., Esmark, M., & Olsen, H. P. (2021, August). Ch. 7: Uoverskuelige mængder af ret: Datalogisk retsvidenskab. In M. J. Christensen, J. R. Herrmann, J. v. H. Holtermann, & M. R. Madsen (Eds.), *De juridiske metoder: 10 bud* (pp. 163–192). Hans Reitzels Forlag. Retrieved from https://dejuridiskemetoder.digi.hansreitzel.dk/?id=p8&loopPrevention=1

Church, K. W., & Hanks, P. (1989, June). Word Association Norms, Mutual Information, and Lexicography. In *27th Annual Meeting of the Association for Computational Linguistics* (pp. 76–83). Vancouver, British Columbia, Canada: Association for Computational Linguistics. Retrieved 2022-04-13, from https://aclanthology.org/P89-1010 doi: 10.3115/981623.981633

Danmarks Domstole. (2014). Sprogpolitik for Danmarks Domstole. Retrieved from https://domstol.dk/media/qxsjet2k/sprogpolitik-for-danmarks-domstole.pdf

Den Danske Ordbog. (n.d.). *imødegå.* Det Danske Sprog- og Litteraturselskab. Retrieved 2022-05-09, from https://ordnet.dk/ddo/ordbog?query=im%C3%B8deg%C3%A5

Den Danske Ordbog. (1993, November). Nyhedsbrev. *Nyhedsbrev*, *2*. Retrieved 2022-05-26, from http://www.dansksproghistorie.dk/wp-content/uploads/2018/06/Nyhedsbrev-nr.-2-November-1993.pdf#page=3

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019, May). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*. Retrieved 2022-04-21, from http://arxiv.org/abs/1810.04805 (arXiv: 1810.04805)

Domstolsstyrelsen. (2021). *A Closer Look at the Courts of Denmark*. Kbh.: Author. (OCLC: 1263674115)

Domstolsstyrelsen. (2022). *Nøgletal om sagsflow og sagsbehandlingstider*. Retrieved 2022-05-30, from https://www.domstol.dk/om-os/tal-og-fakta/noegletal/

Dumais, S. T. (2004). Latent semantic analysis. *Annual Review of Information Science and Technology*, *38*(1), 188–230. Retrieved 2022-04-21, from https://onlinelibrary.wiley.com/doi/abs/10.1002/aris.1440380105 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/aris.1440380105) doi: 10.1002/aris.1440380105

Egense, T. (2018, July). Word2Vec dictionary for 30million Danish newspaper pages. Retrieved 2022-05-17, from https://loar.kb.dk/handle/1902/329 (Accepted: 2018-07-03T10:09:06Z) doi: 10.21994/loar159

Espersen, L. (2005, March). *L 2005-06-24 nr 554, Bemærkning til nr. 8 (kapitel 30)*.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., & Ruppin, E. (2001). Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web* (pp. 406–414).

Firth, J. R. (1957). *A synopsis of linguistic theory 1930–1955* ((Repr.). ed.). Oxford :: Basil Blackwell,. (Studies in linguistic analysis.)

Fuhse, J., Stuhler, O., Riebling, J., & Martin, J. L. (2020, February). Relating social and symbolic relations in quantitative text analysis. A study of parliamentary discourse in the Weimar Republic. *Poetics*, *78*, 101363. Retrieved 2022-04-26, from https://www.sciencedirect.com/science/article/pii/S0304422X18302754 doi: 10.1016/j.poetic.2019.04.004

Goldberg, Y., & Levy, O. (2014, February). word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method. *arXiv:1402.3722 [cs, stat]*. Retrieved 2022-04-13, from http://arxiv.org/abs/1402.3722 (arXiv: 1402.3722)

Gutmann, M., & Hyvärinen, A. (2010, March). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics* (pp. 297–304). JMLR Workshop and Conference Proceedings. Retrieved 2022-04-22, from https://proceedings.mlr.press/v9/gutmann10a.html (ISSN: 1938-7228)

Hamilton, W. L., Leskovec, J., & Jurafsky, D. (2016a, September). Cultural Shift or Linguistic Drift? Comparing Two Computational Measures of Semantic Change. *arXiv:1606.02821 [cs]*. Retrieved 2022-05-09, from http://arxiv.org/abs/1606.02821 (arXiv: 1606.02821)

Hamilton, W. L., Leskovec, J., & Jurafsky, D. (2016b, May). Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change. *arXiv:1605.09096 [cs]*. Retrieved 2022-04-25, from http://arxiv.org/abs/1605.09096 (arXiv: 1605.09096 version: 1)

Handelsretten, S.-. (2018, May). V-17-17. Retrieved 2022-03-18, from http://domstol.fe1.tangora.com/media/-300011/files/V0017001.pdf

Hastie, T., Tibshirani, R., & Friedman, J. (2009). Support Vector Machines and Flexible Discriminants. In T. Hastie, R. Tibshirani, & J. Friedman (Eds.), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (pp. 417–458). New York, NY: Springer. Retrieved 2022-04-29, from https://doi.org/10.1007/978-0-387-84858-7_12 doi: 10.1007/978-0-387-84858-7_12

Hill, F., Reichart, R., & Korhonen, A. (2015, December). SimLex-999: Evaluating Semantic Models With (Genuine) Similarity Estimation. *Computational Linguistics*, *41*(4), 665–695. Retrieved 2022-05-17, from https://direct.mit.edu/coli/article/41/4/665-695/1517 doi: 10.1162/COLI_a_00237

Holtermann, J. v. H., & Madsen, M. R. (2021, August). Ch. 2: Forudsigelser om ret. In M. J. Christensen, J. R. Herrmann, J. v. H. Holtermann, & M. R. Madsen (Eds.), *De ju-*

ridiske metoder: 10 bud (pp. 163–192). Hans Reitzels Forlag. Retrieved from https://dejuridiskemetoder.digi.hansreitzel.dk/?id=p8&loopPrevention=1

Houborg, E. (2011, January). Kriminalisering af narkotika - den politiske og samfundsmæssige baggrund for kriminaliseringsprocesser i periode 1950-2004. In H. V. Dahl & V. A. Frank (Eds.), *Kriminalitet og illegale rusmidler*.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An Introduction to Statistical Learning: with Applications in R*. New York, NY: Springer US. Retrieved 2022-04-30, from https://link.springer.com/10.1007/978-1-0716-1418-1 doi: 10.1007/978-1-0716-1418-1

Justitsministeriet. (1916, April). Lov om Rettens Pleje. *Lovtidende, for 1916*(Nr. 17), 417–675. Retrieved from https://retsplejelov.dab.dk/

Justitsministeriet. (2021, September). *Bekendtgørelse af lov om rettens pleje*. Retrieved 2022-04-26, from https://www.retsinformation.dk/eli/lta/2021/1835

Jørgensen, J. U. (2014, March). *kost og tæring*. Retrieved 2022-05-26, from https://denstoredanske.lex.dk/kost_og_t%C3%A6ring

Katz, D. M., Bommarito, M. J., & Blackman, J. (2017, April). A general approach for predicting the behavior of the Supreme Court of the United States. *PLOS ONE*, *12*(4), e0174698. Retrieved 2022-02-22, from https://dx.plos.org/10.1371/journal.pone.0174698 doi: 10.1371/journal.pone.0174698

Kim, Y., Chiu, Y.-I., Hanaki, K., Hegde, D., & Petrov, S. (2014, May). Temporal Analysis of Language through Neural Language Models. *arXiv:1405.3515 [cs]*. Retrieved 2022-04-25, from http://arxiv.org/abs/1405.3515 (arXiv: 1405.3515)

Kiss, T., & Strunk, J. (2006). Unsupervised Multilingual Sentence Boundary Detection. *Computational Linguistics*, *32*(4), 485–525. Retrieved 2022-04-27, from https://aclanthology.org/J06-4003 doi: 10.1162/coli.2006.32.4.485

Kjærgaard, A. (2010). *Sådan skriver vi – eller gør vi? – En undersøgelse af de tekstlige effekter af to sprogpolitiske projekter i Danmarks Domstole og Københavns Kommune og af årsagerne til projekternes gennemslagskraft*. University of Copenhagen.

Kjærgaard, A. (2012, October). Hvordan ser de på sprog i Danmarks Domstole? – En sprogideologisk analyse. *Klart språk i Norden*. Retrieved 2022-05-22, from https://tidsskrift.dk/ksn/article/view/18329 doi: 10.7146/ksn.v0i0.18329

Kozlowski, A. C., Taddy, M., & Evans, J. A. (2019, October). The Geometry of Culture: Analyzing the Meanings of Class through Word Embeddings. *American Sociological Review*, *84*(5), 905–949. Retrieved 2022-04-08, from https://doi.org/10.1177/0003122419877135 (Publisher: SAGE Publications Inc) doi: 10.1177/0003122419877135

Kulkarni, V., Al-Rfou, R., Perozzi, B., & Skiena, S. (2015, May). Statistically Significant Detection of Linguistic Change. In *Proceedings of the 24th International Conference on World Wide Web* (pp. 625–635). Florence Italy: International World Wide Web Conferences Steering Committee. Retrieved 2022-04-25, from https://dl.acm.org/doi/10.1145/2736277.2741627 doi: 10.1145/2736277.2741627

Kutuzov, A., Velldal, E., & Øvrelid, L. (2017). Tracing armed conflicts with diachronic word embedding models. In *Proceedings of the Events and Stories in the News Workshop* (pp. 31–36). Vancouver, Canada: Association for Computational Linguistics. Retrieved 2022-05-25, from http://aclweb.org/anthology/W17-2705 doi: 10.18653/v1/W17-2705

Kutuzov, A., Øvrelid, L., Szymanski, T., & Velldal, E. (2018, June). Diachronic word embeddings and semantic shifts: a survey. *arXiv:1806.03537 [cs]*. Retrieved 2022-04-25, from http://arxiv.org/abs/1806.03537 (arXiv: 1806.03537)

Lage-Freitas, A., Allende-Cid, H., Santana, O., & de Oliveira-Lage, L. (2019, April). Predicting Brazilian court decisions. *arXiv:1905.10348 [cs]*. Retrieved 2022-04-26, from http://arxiv.org/abs/1905.10348 (arXiv: 1905.10348)

Levy, O., & Goldberg, Y. (2014). Neural Word Embedding as Implicit Matrix Factorization. In *Advances in Neural Information Processing Systems* (Vol. 27). Curran Associates, Inc. Retrieved 2022-04-13, from https://proceedings.neurips.cc/paper/2014/hash/feab05aa91085b7a8012516bc3533958-Abstract.html

Levy, O., Goldberg, Y., & Dagan, I. (2015, December). Improving Distributional Similarity with Lessons Learned from Word Embeddings. *Transactions of the Association for Com-*

putational Linguistics, *3*, 211–225. Retrieved 2022-04-19, from https://direct.mit.edu/tacl/article/43264 doi: 10.1162/tacl_a_00134

Lundum, J. (2021, April). *Private straffesager*. lexdk. Retrieved 2022-05-19, from https://denstoredanske.lex.dk/private_straffesager

Maaten, L. v. d., & Hinton, G. (2008). Visualizing Data using t-SNE. *Journal of Machine Learning Research*, *9*(86), 2579–2605. Retrieved 2022-04-26, from http://jmlr.org/papers/v9/vandermaaten08a.html

Malik, V., Sanjay, R., Nigam, S. K., Ghosh, K., Guha, S. K., Bhattacharya, A., & Modi, A. (2021, May). ILDC for CJPE: Indian Legal Documents Corpus for Court Judgment Prediction and Explanation. *arXiv:2105.13562 [cs]*. Retrieved 2022-04-27, from http://arxiv.org/abs/2105.13562 (arXiv: 2105.13562)

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, ... Xiaoqiang Zheng (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Retrieved from https://www.tensorflow.org/

Medvedeva, M., Vols, M., & Wieling, M. (2020, June). Using machine learning to predict decisions of the European Court of Human Rights. *Artificial Intelligence and Law*, *28*(2), 237–266. Retrieved 2022-01-28, from https://doi.org/10.1007/s10506-019-09255-y doi: 10.1007/s10506-019-09255-y

Medvedeva, M., Wieling, M., & Vols, M. (2022, January). Rethinking the field of automatic prediction of court decisions. *Artificial Intelligence and Law*. Retrieved 2022-03-03, from https://link.springer.com/10.1007/s10506-021-09306-3 doi: 10.1007/s10506-021-09306-3

Medvedeva, M., Üstun, A., Xu, X., Vols, M., & Wieling, M. (2021). Automatic Judgement Forecasting for Pending Applications of the European Court of Human Rights. In *ASAIL/LegalAIIA@ICAIL*.

Michel, J.-B., Shen, Y. K., Aiden, A. P., Veres, A., Gray, M. K., THE GOOGLE BOOKS TEAM, ... Aiden, E. L. (2011, January). Quantitative Analysis of Culture Using Millions of Digitized Books. *Science*, *331*(6014), 176–182. Retrieved 2022-05-25, from https://www.science.org/doi/full/10.1126/science.1199644 (Publisher: American Association for the Advancement of Science) doi: 10.1126/science.1199644

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013, September). Efficient Estimation of Word Representations in Vector Space. *arXiv:1301.3781 [cs]*. Retrieved 2022-04-13, from http://arxiv.org/abs/1301.3781 (arXiv: 1301.3781)

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013, October). Distributed Representations of Words and Phrases and their Compositionality. *arXiv:1310.4546 [cs, stat]*. Retrieved 2022-04-21, from http://arxiv.org/abs/1310.4546 (arXiv: 1310.4546)

Nelson, L. K. (2020, February). Computational Grounded Theory: A Methodological Framework. *Sociological Methods & Research*, *49*(1), 3–42. Retrieved 2022-05-09, from https://doi.org/10.1177/0049124117729703 (Publisher: SAGE Publications Inc) doi: 10.1177/0049124117729703

Nielsen, F., & Hansen, L. K. (2017). Open semantic analysis: The case of word level semantics in Danish. *undefined*. Retrieved 2022-04-15, from https://www.semanticscholar.org/paper/Open-semantic-analysis%3A-The-case-of-word-level-in-Nielsen-Hansen/45999db5873b468cf3102c5778b2194f627711e5

Olesen, A., Rosenholm, A. B., Jørgensen, K., Trabolt, T. H., Simonsen, S. B., Hansen, T., ... Hermansen, A. (2020). *Sagsomkostninger i straffesager: Ny viden om et gammelt problem*. Forlag1.dk. Retrieved from https://vbn.aau.dk/da/publications/sagsomkostninger-i-straffesager-ny-viden-om-et-gammelt-problem

Pauli, A. B., Barrett, M., Lacroix, O., & Hvingelby, R. (2021, May). DaNLP: An open-source toolkit for Danish Natural Language Processing. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)* (pp. 460–466). Reykjavik, Iceland (Online): Linköping University Electronic Press, Sweden. Retrieved 2022-05-09, from https://aclanthology.org/2021.nodalida-main.53

Pedersen, B. S., Wedekind, J., Bøhm-Andersen, S., Henrichsen, P. J., Hoffensetz-Andersen, S., Kirchmeier-Andersen, S., ... Thomsen, H. E. (2012). *The Danish Language in the Digital Age: Det danske sprog i den digitale tidsalder* (G. Rehm & H. Uszkoreit, Eds.). Springer

Science+Business Media. Retrieved 2022-05-17, from https://research.cbs.dk/en/publications/the-danish-language-in-the-digital-age-det-danske-sprog-i-den-dig doi: 10.1007/978-3-642-30627-3

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*(85), 2825–2830. Retrieved 2022-04-29, from http://jmlr.org/papers/v12/pedregosa11a.html

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1532–1543). Retrieved from http://www.aclweb.org/anthology/D14-1162

Retsinformationsrådet (Ed.). (1988). *Retsinformationsrådets betænkning om databaser med konkrete afgørelser*. Kbh. [København]: Statens informationstjeneste.

Rong, X. (2016, June). word2vec Parameter Learning Explained. *arXiv:1411.2738 [cs]*. Retrieved 2022-04-22, from http://arxiv.org/abs/1411.2738 (arXiv: 1411.2738)

Rudolph, M., & Blei, D. (2018). Dynamic Embeddings for Language Evolution. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web - WWW '18* (pp. 1003–1011). Lyon, France: ACM Press. Retrieved 2022-04-06, from http://dl.acm.org/citation.cfm?doid=3178876.3185999 doi: 10.1145/3178876.3185999

Sagi, E., Kaufmann, S., & Clark, B. (2011, December). Tracing semantic change with Latent Semantic Analysis. In *Tracing semantic change with Latent Semantic Analysis* (pp. 161–183). De Gruyter Mouton. Retrieved 2022-04-26, from https://www.degruyter.com/document/doi/10.1515/9783110252903.161/html doi: 10.1515/9783110252903.161

Schneidermann, N., Hvingelby, R., & Pedersen, B. (2020, May). Towards a Gold Standard for Evaluating Danish Word Embeddings. In *Proceedings of the 12th Language Resources and Evaluation Conference* (pp. 4754–4763). Marseille, France: European Language Resources Association. Retrieved 2022-05-09, from https://aclanthology.org/2020.lrec-1.585

Schönemann, P. H. (1966, March). A generalized solution of the orthogonal procrustes problem. *Psychometrika*, *31*(1), 1–10. Retrieved 2022-04-22, from https://doi.org/10.1007/BF02289451 doi: 10.1007/BF02289451

Shoemark, P., Liza, F. F., Nguyen, D., Hale, S., & McGillivray, B. (2019, November). Room to Glo: A Systematic Comparison of Semantic Change Detection Approaches with Word Embeddings. 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)* (pp. 66–76). Hong Kong, China: Association for Computational Linguistics. Retrieved 2022-05-25, from https://aclanthology.org/D19-1007 doi: 10.18653/v1/D19-1007

Strickson, B., & De La Iglesia, B. (2020, March). Legal Judgement Prediction for UK Courts. In *Proceedings of the 2020 The 3rd International Conference on Information Science and System* (pp. 204–209). Cambridge United Kingdom: ACM. Retrieved 2022-04-26, from https://dl.acm.org/doi/10.1145/3388176.3388183 doi: 10.1145/3388176.3388183

Strømberg-Derczynski, L., Ciosici, M. R., Baglini, R., Christiansen, M. H., Dalsgaard, J. A., Fusaroli, R., ... Varab, D. (2021, May). The Danish Gigaword Project. *arXiv:2005.03521 [cs]*. Retrieved 2022-05-04, from http://arxiv.org/abs/2005.03521 (arXiv: 2005.03521)

Sulea, O.-M., Zampieri, M., Vela, M., & van Genabith, J. (2017, August). Predicting the Law Area and Decisions of French Supreme Court Cases. *arXiv:1708.01681 [cs]*. Retrieved 2022-03-01, from http://arxiv.org/abs/1708.01681 (arXiv: 1708.01681)

Supreme Court. (1996). U.1996.872/2. *Ugeskrift for Retvæsen*.

Szymanski, T. (2017, July). Temporal Word Analogies: Identifying Lexical Replacement with Diachronic Word Embeddings. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (pp. 448–453). Vancouver, Canada: Association for Computational Linguistics. Retrieved 2022-04-26, from https://aclanthology.org/P17-2071 doi: 10.18653/v1/P17-2071

van Rossum, G. (2022, March). 6.2 re — Regular expression operations. In *The Python Library Reference* (Vol. Release 3.8.13, pp. 104–122). Python Software Foundation.

Vapnik, V. N. (1997). The Support Vector method. In W. Gerstner, A. Germond, M. Hasler, & J.-D. Nicoud (Eds.), *Artificial Neural Networks — ICANN'97* (pp. 261–271). Berlin,

Heidelberg: Springer. doi: 10.1007/BFb0020166

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017, December). Attention Is All You Need. *arXiv:1706.03762 [cs]*. Retrieved 2021-12-07, from http://arxiv.org/abs/1706.03762 (arXiv: 1706.03762)

Virtucio, M., Aborot, J., Abonita, J., Avinante, R., Co, R., Neverida, M., ... Tan, G. (2018, July). Predicting Decisions of the Philippine Supreme Court Using Natural Language Processing and Machine Learning. In (pp. 130–135). doi: 10.1109/COMPSAC.2018.10348

Vols, M. (2019, March). European Law and Evictions: Property, Proportionality and Vulnerable People. *European Review of Private Law*, *2019*. doi: 10.54648/ERPL2019040

Vylomova, E., Murphy, S., & Haslam, N. (2019). Evaluation of Semantic Change of Harm-Related Concepts in Psychology. In *Proceedings of the 1st International Workshop on Computational Approaches to Historical Language Change* (pp. 29–34). Florence, Italy: Association for Computational Linguistics. Retrieved 2022-05-25, from https://www.aclweb.org/anthology/W19-4704 doi: 10.18653/v1/W19-4704

Waaben, H., Munck, K. S., Eiriksson, B. A., & Aagard, H. (2017). *Færdselsloven: med kommentarer* (1. udgave, 1. oplag ed.). København: Jurist- og Økonomforbundets Forlag.

Walbom, A. (2021, April). *kendelse.* Retrieved 2022-03-18, from https://denstoredanske.lex.dk/kendelse

Waltl, B., Bonczek, G., Scepankova, E., Landthaler, J., & Matthes, F. (2017, September). Predicting the Outcome of Appeal Decisions in Germany's Tax Law. In (Vol. LNCS-10429, p. 89). Springer International Publishing. Retrieved 2022-04-27, from https://hal.inria.fr/hal-01703326 doi: 10.1007/978-3-319-64322-9_8

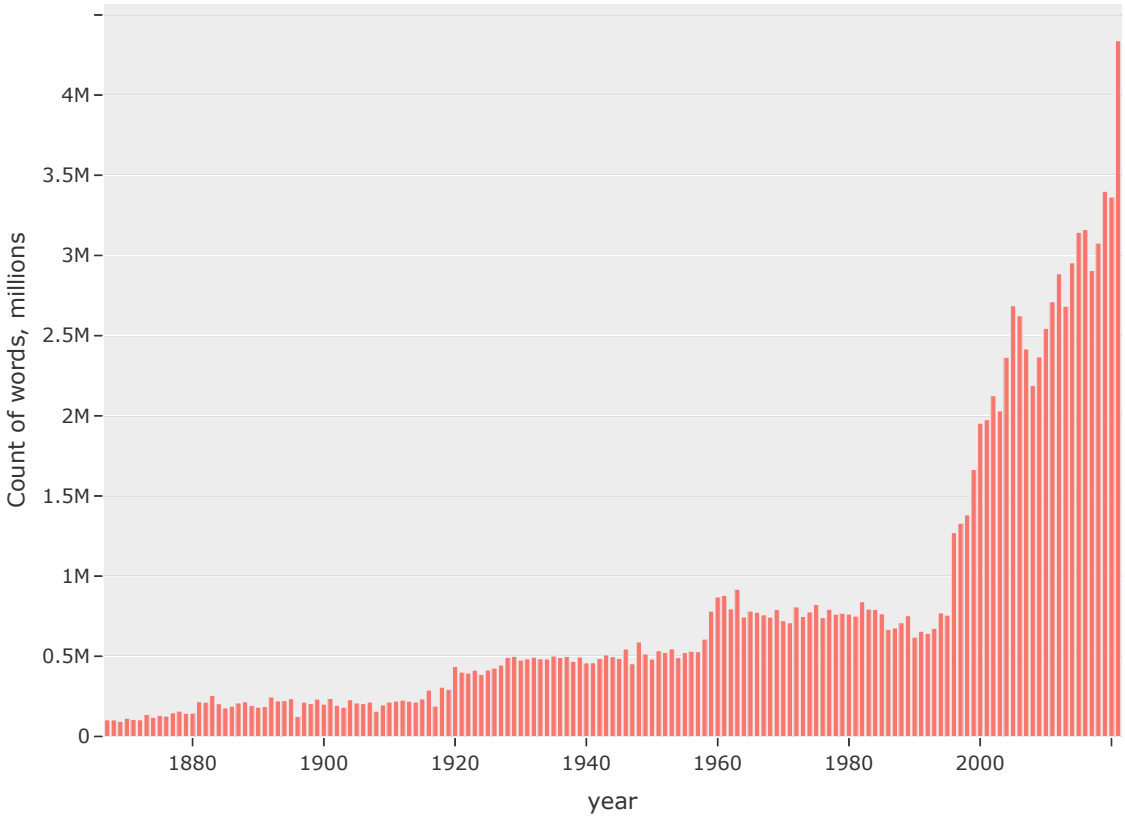
Wang, Y., Cui, L., & Zhang, Y. (2020, April). How Can BERT Help Lexical Semantics Tasks? *arXiv:1911.02929 [cs]*. Retrieved 2022-05-10, from http://arxiv.org/abs/1911.02929 (arXiv: 1911.02929)

Yao, Z., Sun, Y., Ding, W., Rao, N., & Xiong, H. (2018, February). Dynamic Word Embeddings for Evolving Semantic Discovery. *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 673–681. Retrieved 2022-03-31, from http://arxiv.org/abs/1703.00607 (arXiv: 1703.00607) doi: 10.1145/3159652.3159703

Řehůřek, R., & Sojka, P. (2010, May). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks* (pp. 45–50). Valletta, Malta: ELRA.

Appendix A Count of words

Figure A.1
Count of words in court rulings

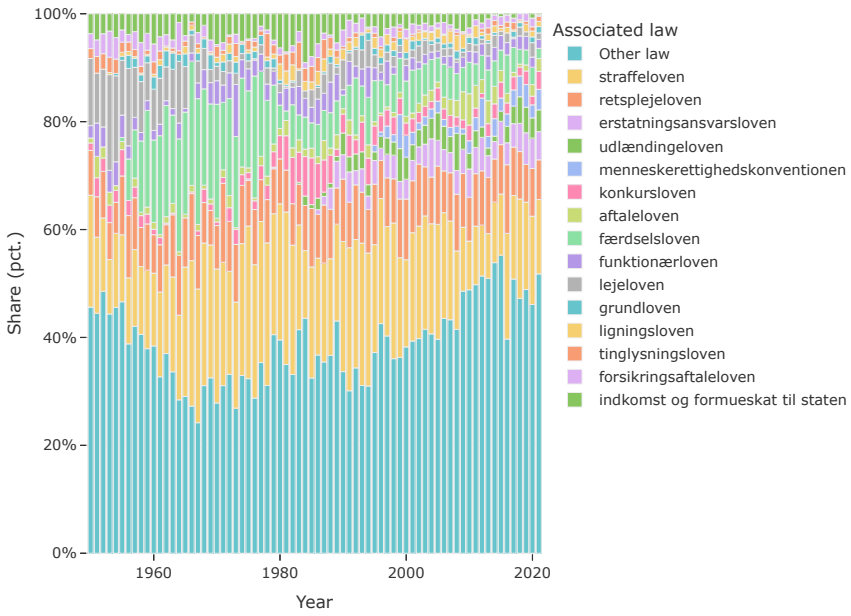


Note: See interactive figure at www.rostrup.nu/count_of_words.

Appendix B Laws associated to court documents split in decisions and judgements

Figure B.1
Dynamics of the 15 most associated laws of court documents divided into decisions and judgements

(a) Associated laws of court judgements

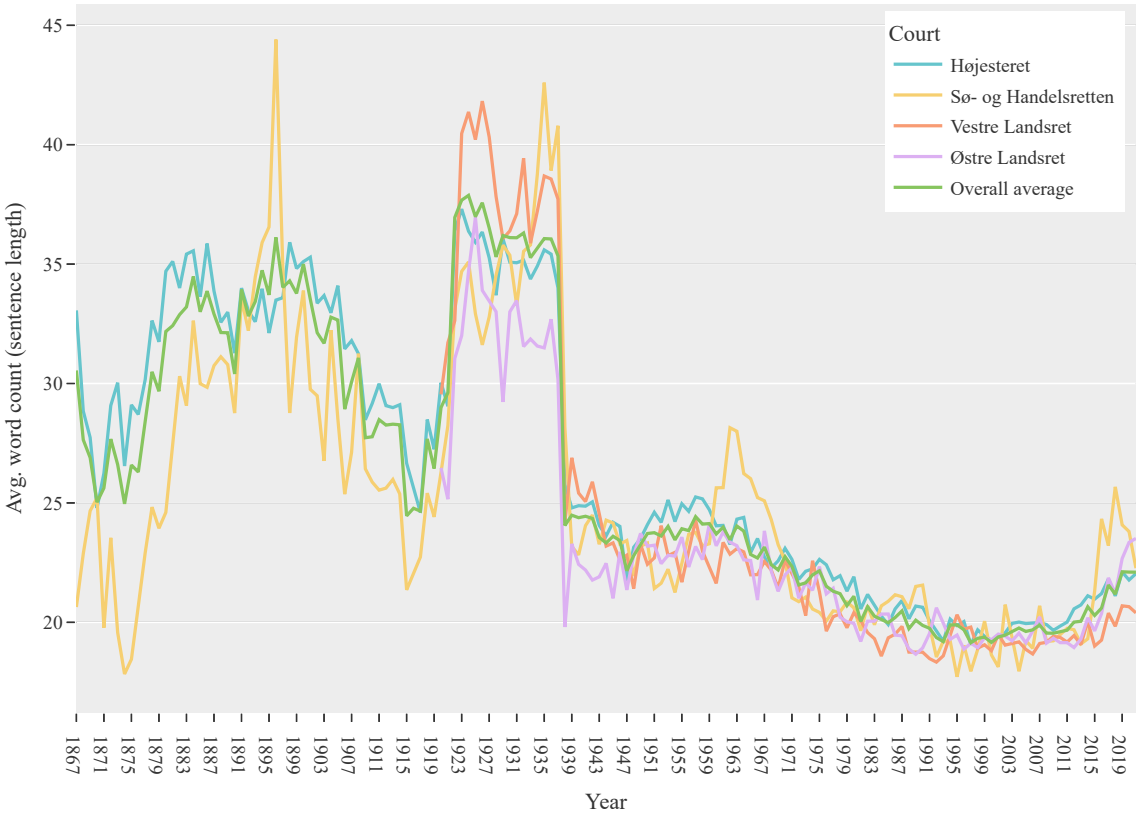


(b) Associated laws of court decisions

Note: Figures are available in interactive formats. For figure B.1a see www.rostrup.nu/distribution_of_judgements_by_associated_law and for figure B.1b see interactive format at www.rostrup.nu/distribution_of_court_decisions_by_associated_law

Appendix C Sentence length through time by court

Figure C.1
Sentence length by court



Note: An interactive version can be found at www.rostrup.nu/average_length_of_sentences_court_type. For simplicity other court types than the ones shown has been removed from the graph due to few samples for the given year leading to a large volatility in sentence length. They are still included in the overall average.

Appendix D The objective function in the DW2V model

I outline the objective function that is used to estimate the embeddings in the DW2V model. The following objective function is proposed by Yao et al. (2018:3) to estimate $W(t)$:

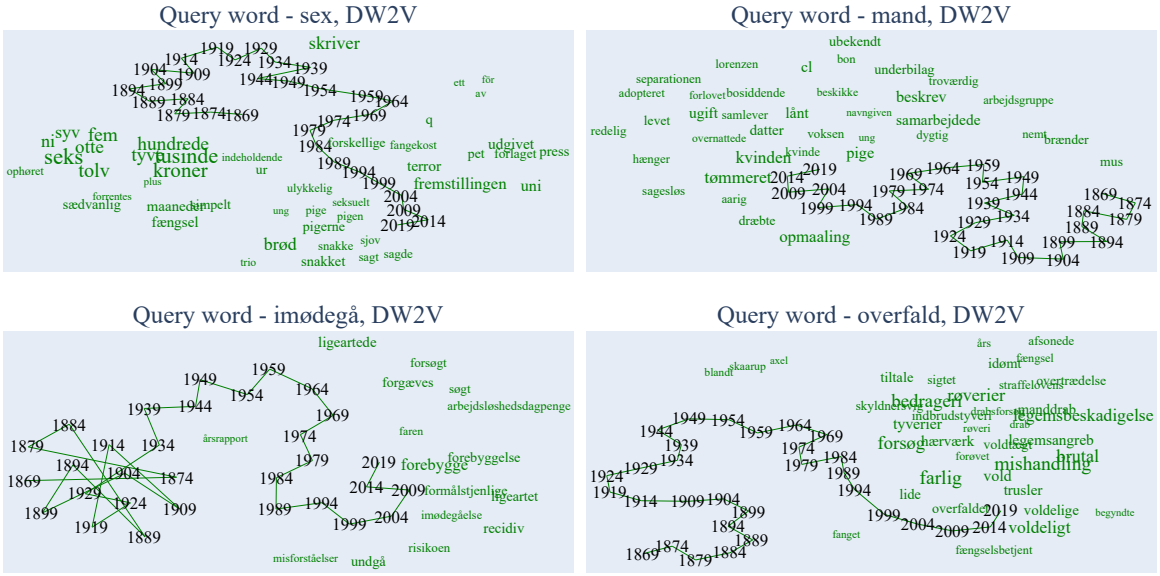
$$E = \min_{W(1),\ldots,W(T)} \underbrace{\frac{1}{2}\sum_{t=1}^{T} ||PPMI(t) - W(t)W(t)^T||}_{\text{Relevance of word embeddings}} + \underbrace{\frac{\lambda}{2}\sum_{t=1}^{T} ||W(t)||_F^2}_{\text{Penalty term: data fidelity}} + \underbrace{\frac{\tau}{2}\sum_{t=1}^{T} ||W(t-1) - W(t)||_F^2}_{\text{Penalty term: alignment over time}} \tag{8}$$

Hence, the optimisation includes two penalty terms, one ensuring data fidelity (a restriction on the size of the elements) and one ensuring alignment of the embeddings. The hyperparameters λ and τ control to which degree data fidelity and alignment are preferred. For instance, $\tau \to \infty$ will imply that the embeddings are static and $tau = 0$ will imply no alignment of embeddings. The optimisation problem in (8) cannot be solved analytically due to $U(t)U(t)^T$ being quartic. Yao et al. (2018) relaxes the optimisation problem by introducing a new variable to change $U(t)U(t)^T$ to $U(t)W(t)^T$ and adding penalty terms for the new variable. Holding either $W(t)$ or $U(t)$ fixed, the objective function can now be minimised using block coordinate descent (BCD). For more detail, I refer to the original implementation in (Yao et al., 2018:3–4).

Appendix E Investigations into the dynamic embeddings of *sex*, *imødegå*, *mand* and *overfald*

In figure E.1 I present an equivalent visualisation to figure 5 for the words *sex*, *imødegå*, *mand* and *overfald*. I only show the DW2V embeddings. I will briefly describe what I deduce from the patterns that emerge.

Figure E.1
Dynamic word embeddings mapped in a two-dimensional space for the words *sex*, *imødegå*, *mand* and *overfald* using the DW2V embeddings



In Danish orthography, the character *x* was for many words substituted with the characters *ks* in 1889 (M. H. Andersen, 2016). *Sex* meant the number six until this change, whereafter the word was not used much in the corpus. Lately, the word *sex* has been adopted from the English word *sex*. This is reflected in the patterns from the embeddings, where prior to 1950 the word was mostly associated with other numbers or units of measurement and after 1950 primarily related to social interactions.

The word *imødegå* is a contronym, i.e. it has two meanings that are each other's opposite. *Imødegå* means to oppose something, but at least since the 1990s, the word has also been used as *imødekomme*, meaning to comply with or meet each other's requests, demands etc. (Den Danske Ordbog, 1993:4). This pattern is not evident from the embeddings, which can be due to many reasons. For instance, the letter *å* was introduced instead of *aa* in a reform of danish orthography in 1948, implying that the word is not observed before 1948 in the corpus. Secondly, the courts aim to be precise, and hence they probably avoid words that can be misinterpreted. If the courts use the word, it will most likely be with its original meaning.[24]

I have included the word *mand* [trans. man] to give an indication of whether the court's gender

[24] According to a Danish dictionary, the most recent use of the words is considered incorrect by many (Den Danske Ordbog, n.d.).

discourse can be illustrated with the embeddings. Note that the words for man and woman in Danish are the same as for female and male. The word does not show any significant movement in the two-dimensional space, indicating that the courts' language does not contain gender-biased language. However, for such a conclusion to be made a more in-depth analysis of the embeddings and the corpus is necessary, e.g. by using more positive or negative loaded synonyms of the gender, reading samples of rulings where the genders appear, gaining insights into the correlation between arguments and gender and so forth. For instance, I argue in the appendix F that the plural of *mand*, *mænd* has changed substantially.

The last word *overfald* [trans. assault] is an example of a crime. All the words similar to crime are close to the most recent embedding but relatively far away from the first embeddings. This could suggest that the entire subset of words related to crime moves in the same direction; keep in mind that it is only the most recent embeddings of the similar words shown in the two-dimensional space. Hence, the word does not show a change in meaning using this approach.

Appendix F Evaluating the ten most changed words as found using DW2V embeddings

I will, in this appendix, evaluate the words that have changed the most as assessed by the largest cosine distance between the word embeddings estimated in the period 1867-1919 and the period 2010-2021 using the AW2V model. In table F.1 these words are shown together with the cosine distance.

Table F.1
The ten words that have changed the most as evaluated by cosine distance

Word	Cosine distance
omgang	0.93
fr	0.91
aa	0.91
lo	0.90
ansattes	0.90
tæring	0.89
mænd	0.89
islandske	0.89
stødende	0.89
ere	0.89

I will go through each word using examples of sentences from each period shown in the table F.2. Note that I have denoted the first period as 1867 to 1949 below; this is due to an unfortunate oversight, when I did the analysis.

omgang — *Omgang* is a polysemous word. From 1867 to 1949, the word is used in various contexts. It is used as, e.g. a violent action and a round at an auction. In the period 2010-2021 it is used mostly to describe the commencement of something: An English translation could be "initally" or "to begin with", however in some cases the word is used to describe other things, there is just a much larger proportion of uses of the word in the "initally"-form.

fr — *fr* is used as an abbreviation. From 1867 to 1949, it was used to abbreviate a woman's title irrespective of her marital status *frue* [trans. Mrs] or *frøken* [trans. Miss]. It also used an abbreviation of *forordning* [trans. statutory instrument]. Both of these abbreviation are not used in the most recent period. Its use seems a bit more random being a part of the name of a plane in a court case and used to abbreviate the standardisation mark registry process.

aa — After the orthography reform in 1948 *aa* was replaced by the character *å*. Hence, since *aa* (and *å*) means river, this is a common use of the word in the earliest period. Furthermore, the word was used to abbreviate first names starting with *aa*. Just as *fr* the use of *aa* is more sporadic, being included when denoting an appendix (*bilag aa.*) and when referring to a particular type of bonds *aa1-obligationer*.

lo — The change in this word is partly due to the word being used to describe a part of a farm, where the grain is threshed, and partly due to, what I believe is a flaw in the way the old rulings have been digitalised. The characters *lo* resembles the number 10 in writing. Hence, several sentences include *lo* in a context where a number was intended to be. The more recent rulings are written on a computer, so this flaw is not apparent. In the most recent period, the word is used as the past tense of laughing and an abbreviation for the Danish Confederation of Trade Unions and the Danish Tenants Organization (lejernes lo).

ansattes — *Ansattes* is a polysemous word. From 1867 to 1949 it was primarily used to estimate a value (e.g. estimated to a value of 10,000 kr.) In the most recent time period it is mainly used to describe an employee.

tæring — *Tæring* was used as a part of the phrase "kost og tæring", which describes compensation for a legal party's troubles in dealing with a court case (Jørgensen, 2014). With the introduction of the Administration of Justice Act in 1916, the phrase was discontinued. Today *tæring* is mainly used to describe the corrosion of an object.

mænd — The main difference in the use of *mænd* [trans. men] is that people appointed by the court to do something in rulings from 1867 to 1949 are denoted men. Today the word is used to describe the plural of man.

islandske — The use of *islandske* [trans. Icelandic] changed due to the severance of Iceland from the Kingdom of Denmark in 1918.

stødende — Stødende is a polysemous word. In rulings from 1876 to 1949, the word is mostly used as a preposition similar to adjacent always in combination with the word "til". In the most recent period the word is used primarily to categorise behaviour as offensive. The use as a preposition still exists, though.

ere — I am still unsure how exactly the word *ere* is used from 1876 to 1950. The word appears instead of the word *er* [trans. is/are] in documents prior to 1915, but not consistently. After 1915 the word only existed in the form that it does today: a suffix denoting the plural form of a noun, which is (wrongly) classified as a word. For instance, *ere* is classified as a word if it appears in the word *hjemme-pc'ere* [trans. personal computers].

Table F.2
Selected sentences using the top-10 most changed words as estimated with cosine distance

Word	UfR ID	Sentence in 1867-1949	UfR ID	Sentences in 2010-2022
Omgang	U.1921.27	"[...] der ved den paagældende lejlighed har fundet legemlig *omgang* sted mellem parterne [...]"	U.2021.600	"Han regnede i første *omgang* med, at det bare var en sene [...]"
	U.1867.672	"Spørgsmaal om hvorvidt en tiltalt havde gjort sig skyldig i barnefødsel i dølgsmaal eller ialtfald i uforsvarlig *omgang* med sit nyfødte barn"	U.2021.3330	"At afgive forklaring var en barsk *omgang* for ham"
	U.1876.97	"[...] hvilket panthaveren efter *omgang* ifølge konkurslovens § 155 havde realiseret ved auktion [...]"	U.2018.3108	"[...] og at indførslen af amfetamin og skunk er sket ad én *omgang*."
fr	U.1928.943	"[...] gaardejer *fr* kiærholm af hyllested skovgaard [...]"	U.2018.3442	"påstand 1 dansk fællesmærkeregistrering *fr* 1937 00008, sig det med blomster (ord) skal ophæves i kl. 14, 16, [...]"
	U.1880.980	"[...] ikke fundne at kunne henregnes til de i *fr.* 1 oktober 1802 § 27 [...]"	U.2016.3577	"[...] skyldes kompensation for flyforsinkelse den 7. november 2014 med fly *fr* 1229 [...]"
	U.1878.883	"§ 281, frifunden, da ikke noget efter *fr.* 8 septbr. [...]"		
aa	U.1948.987	"[...] fhv. direktør n. *aa.* rasmussen [...]"	U.2020.1036	"Den 25. august 2016 modtog z a/s en e-mail fra ... (v ag) (bilag *aa*)."
	U.1919.122	"[...] havde nedrammet ca. 70 pæle i gedsted- lerchenfeldt *aa*, idømt bøde [...]"	U.2013.182	"aa1 obligationer i *aa*-kategorierne er i alle henseender af god kvalitet [...]"
lo	U.1945.52	"[...] en skydedør fra hestestald til *lo*, som hestene var kommet ud af [...]"	U.2012.33	"u (selv) mod l (v/lejernes *lo*)"
	U.1922.519/2	"[...] der i ca *lo* aar havde været knyttet til kraks vejviser [...]"	U.2016.2334	"[...] den 21. december, hvor de havde en fin tid, de *lo* og var på biblioteket [...]"
	U.1922.940	"[...] installation til en 10 hk motor i stakhjelm og *lo* [...]"	U.2020.1615	"i da og *lo*-aftalen af 27. oktober 2006 om kontrolforanstaltninger"
ansattes	U.1919.204	"[...] og han *ansattes* i den derpaa følgende tid som vikarierende kandidat [...]"	U.2014.1828	"Efter emballageoverenskomsten er sygelønnen den *ansattes* almindelige dagtursløn [...]"
	U.1947.259	"[...] *ansattes* det beløb, som k skulde tilsvare [...]"		
tæring	U.1892.1128/2S	"[...] indstævnte har priccipaliter paastaaet sagen afvist og sig tillagt kost og *tæring* [...]"	U.2012.3048	"[...] der gennem mange år var sket en *tæring* af tanken [...]"

Table F.2
Examples of sentences using the top-10 most changed words as estimated with cosine distance (cont.)

Word	UfR ID	Sentence in 1867-1949	UfR ID	Sentences in 2010-2022
	U.1887.41S	"[...] indstævnte har paastaaet sig tillagt kost og *tæring* [...]"	U.2014.1823	"[...] der var også tydelig *tæring* af rørene [...]"
mænd	U.1945.542	"[...] mellem parterne i overværelse af to af dem valgte sagkyndige *mænd* [...]"	U.2013.2460	"det kan godt passe, at der var både *mænd* og kvinder [...]"
	U.1899.11S	"[...] de 2 udnævnte *mænd* valgte derefter en 3die mand, og disse 3 *mænd* [...]"	U.2021.794	"hos k. er utryg ved *mænd*, selv ved sin morfar!!"
	U.1908.29	"[...] et saadant beløb, som uvillige af retten udmeldte *mænd* maatte fastsætte [...]"		
islandske	U.1894.15	"Den *islandske* landsoverrets dom af 24 august 1891 er saalydende [...]"	U.2013.277	"[...] det *islandske* marked forsvandt helt [...]"
	U.1919.348	"Appellanten har til støtte for sin paastand [...] henvist til § 50 i den *islandske* forfatningslov [...]"	U.2015.2308S	"Al handel med *islandske* kroner var herefter suspenderet."
støndende	U.1939.252	"[...] han paatænker at anvende det til vejen *støndende* areal af matr. [...]"	U.2020.948	"Der blev klaget over l's opførsel, herunder *støndende* kommunikation [...]"
	U.1920.198	"[...] i den til karlekammeret *støndende* følbox [...]"	U.2010.2142	"[...] det vil virke *støndende* for den almindelige retsfølelse [...]"
			U.2015.676	"Denne gren forløber dels mellem gavlene på de til hver side *støndende* ejendommes huse [...]"
ere		"[...] alle retfærdige ville give ham medhold, og dem, som *ere* og have været imod ham, kalder han simple uslinger."	U.2010.1776	"[...] indkøb af hjemme-pc'*ere* til medarbejderne."
		"Sagens omstændigheder *ere* følgende [...]"	U.2010.3009:	"hvorfor de to mp'*ere* måtte tage fat i ham for at undgå"

Appendix G Evaluating the regex patterns used for entity matching

The method to identify the assignment of the trial costs depends on how consistently I can identify the entity that either pays or receives the cost of the trial. The patterns are shown in table G.1. The first patterns are associated with the "No party pays cost of trial" and does as such, not refer to an entity (step 3 in figure 7). The most matched pattern is simple and matches all sentences where "ingen af parterne" is present verbatim in the document. 1,337 sentences are matched this way.

The most used pattern for identifying the entity that pays the cost of trial is `(?<=(sagsomkostninger).*skal\s)(?!betale)((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=.*betale)`, identifying the entity that pays the cost of trial in 3,024 of the rulings. It finds the entity in senteces such as "I sagsomkostninger for begge retter skal Andelsboligforeningen U inden 14 dage betale 25.000 kr. til statskassen og 22.240 kr. til L " (Andelsboligforeningen) and "I sagsomkostninger for landsret og Højesteret skal appellanten betale 55.000 kr. til indstævnte" (appellanten).

When identifying the party that receives the cost of the trial, I mainly use one pattern. This pattern identifies roughly 10,000 entities. The pattern is very inclusive and essentially says that the word after the word "til" (with a few exceptions) is the entity that receives the cost of trial. A lot of these entities are not in fact entities, however it will *not* identify the party that *receives* the cost of trial but more likely noise words such as "3500", "skøn" and "advokat".[25] These noise words will, however, be filtered out in step 5 of figure 7 why, as long as the pattern does not match the *wrong* party, it is not of great concern.

Table G.1
All regex patterns used for entity matching

Pattern	Matches count
No party pays cost of trial patterns	
ingen\saf\sparterne	1337
hver\s(af\s)?(part(.*)?ære(r)?|ære(r)?part(.*)?)\s(sin|egn)(e)?\s(sags)?omkostninger	567
ophæve	436
hver\spart	40
ingen\spart\s	21
hver\saf\sparterne	13
ingen\saf\ssagens\sparter	9
Entity that pays cost of trial	
(?<=(sagsomkostninger).*skal\s)(?!betale)((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=.*betale)	3024
(?<=.*((sagens\somkostninger)|(omkostninger\sfor\sagen)|(sagsomkostninger)).*betaler\s)((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)	2838

[25] Matches corresponds to: "Erstatningen skønnes herefter at kunne fastsættes til *3500* kr., hvorhos sagsøgte bør betale sagens omkostninger med 350 kr." (3500), "Sagsøgte bør derhos betale udgifterne til *skøn* med 900 kr. og sagens omkostninger med 500 kr" (skøn) and "Sagens omkostninger for Højesteret, derunder i salær til *advokat* Poul Christiansen 500 kr., udredes af tiltalte Karl Jensen." (avokat)

Table G.1
All regex patterns used for entity matching (cont.)

Pattern	Matches count
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\sskal\sbetale\s)	1501
(?<=udredes\saf\s)((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)	862
(?<=\sbetales\saf\s)([a-zA-Z0-9ÆØÅæøå]+)	497
((sagens\somkostninger)|(omkostninger\sfor\sagen)|(sagsomkostninger))	267
.*findes\s\K.* (?=\sat\sburde)	
.*(?=((\sdømmes\stil\sat)|\sdømtes|(\sbør)).*(betale).*	238
((sagens\somkostninger)|(omkostninger\sfor\sagen)|(sagsomkostninger)))	
(?<!(.*)?til\s.*)((det\s|de\s)?[a-zA-Z0-	218
9ÆØÅæøåü]+)(?=\sbetaler.* ((sagens\somkostninger)|(omkostninger\sfor\sagen)	
|(sagsomkostninger)))	
(?<=det\s(pålagdes|pålægges)\s)((det\s|de\s)?[a-zA-Z0-	209
9ÆØÅæøåü]+)(?=.*til)	
(?<=\svil\s).*(?=\shave\sat\sbetale)	128
(?<=i\ssagsomkostninger\s.*bør\s).*(?=\sbetale)	127
(?<=efter\ssagens\sudfald\sskal\s).*(?=\s((betale\ssagens\somkostninger)	119
|(betale\somkostninger\sfor\sagen)|(i\ssagsomkostninger)))	
(?<=\svil\s).*(?=\shave\sat\sgodtgøre\s)	116
(?<=sagens\somkostninger.*\spålagdes\s)((det\s|de\s)?[a-zA-Z0-	114
9ÆØÅæøåü]+)	
(?<=Det\sblev\spålagt).*(at\sbetale)	85
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\sbør\si\ssagsomkostninger)	27
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\stilpligtedes\sat\sbetale)	26
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\sbør(.*betale.*til|.*til.*betale))	22
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\sbetaler\si\ssagsomkostninger)	20
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\sbør\s(til|derhos)\s.*(betale|godtgøre))	16
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\sfindes\sat\sburde\sbetale)	9
(?<=((sagens\somkostninger)|(omkostninger\sfor\sagen)|(sagsomkostninger))	8
\sbør\s).*(?=\sbetale)	
(som\snedenfor\snævnt)	6
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\svil\shave\sat\sbetale)	5
(?<=(sagens\somkostninger)|(omkostninger\sfor\sagen)|(sagsomkostninger)\s	5
udreder\s) ((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)	
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\svil\shave\sat\sgodtgøre\s)	3
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\sfindes\sat\sburde\stilsvare)	2
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\svil\shave\sat\stilsvare)	2
((det\s|de\s)?[a-zA-Z0-9ÆØÅæøåü]+)(?=\sbør.*erstatte)	2
Entity that recieves cost of trial	
(?<=(?!(ind))til\s(?!(dækning|betaling|følge|at)))((det\s|de\s)?[a-	10205
zA-Z0-9ÆØÅæøåü]+)	
(?<=\stillagdes\sder\s).*	327
(?<=sagsomkostninger\stilkendtes\sder\s)((det\s|de\s)?[a-zA-Z0-	33
9ÆØÅæøåü]+)	

Appendix H Stopwords

A list of stopwords from the NLTK-PACKAGE. The stopwords can be downloaded at www.nltk.org/nltk_-data/ (item 86 at time of writing). The stopwords are:

> og, i, jeg, det, at, en, den, til, er, som, på, de, med, han, af, for, ikke, der, var, mig, sig, men, et, har, om, vi, min, havde, ham, hun, nu, over, da, fra, du, ud, sin, dem, os, op, man, hans, hvor, eller, hvad, skal, selv, her, alle, vil, blev, kunne, ind, når, være, dog, noget, ville, jo, deres, efter, ned, skulle, denne, end, dette, mit, også, under, have, dig, anden, hende, mine, alt, meget, sit, sine, vor, mod, disse, hvis, din, nogle, hos, blive, mange, ad, bliver, hendes, været, thi, jer, sådan.

Appendix I Distribution of labels by court instance

This table presents the distribution of the estimated labels by court instance.

Table I.1
Distribution of labels by court instance

Court	Who pays	Not-criminal	Criminal	Total	Share
District Courts	Defendant	19	0	19	36.5%
	Proc./Plaintiff	9	0	9	17.3%
	No single party	24	0	24	46.2%
Subtotal		52	0	52	100%
Maritime and Commercial High Court	Defendant	320	5	325	44.7%
	Proc./Plaintiff	189	5	194	26.7%
	No single party	208	0	208	28.6%
Subtotal		717	10	727	100%
Western High Court	Defendant	715	44	759	37.5%
	Proc./Plaintiff	842	57	899	44.4%
	No single party	367	0	367	18.1%
Subtotal		1,924	101	2,025	100%
Eastern High Court	Defendant	596	446	1,042	38.1%
	Proc./Plaintiff	798	392	1,190	43.5%
	No single party	501	0	501	18.3%
Subtotal		1,895	838	2,733	100%
Supreme Court	Defendant	1,407	1,018	2,425	29.1%
	Proc./Plaintiff	3,750	841	4,591	55.1%
	No single party	1,318	0	1,318	15.8%
Subtotal		6,475	1,859	8,334	100%
All courts	Defendant	3,421	1,513	4,934	35.6%
	Proc./Plaintiff	5,224	1,295	6,519	47.0%
	No single party	2,418	0	2,418	17.4%
Total		11,063	2,808	13,871	100%

Appendix J 5-fold cross-validation results

Table J.1
The preprocessing steps and hyperparameter found using 5-fold cross-validation

	Not-masked	Masked	Masked, non-criminal	Masked, criminal
Preprocessing steps				
Stopwords removed	✓	✓	✓	✓
N-gram range	(2, 4)	(2, 4)	(2, 4)	(1,4)
Feature freq. min.	3	3	3	3
Hyperparameter				
C	5	5	5	5