Artificial Intelligence and Law
https://doi.org/10.1007/s10506-023-09379-2

ORIGINAL RESEARCH

Natural language processing for legal document review: categorising deontic modalities in contracts

S. Georgette Graham[1] · **Hamidreza Soltani**[1] · **Olufemi Isiaq**[2]

Accepted: 11 October 2023

Abstract

The contract review process can be a costly and time-consuming task for lawyers and clients alike, requiring significant effort to identify and evaluate the legal implications of individual clauses. To address this challenge, we propose the use of natural language processing techniques, specifically text classification based on deontic tags, to streamline the process. Our research question is whether natural language processing techniques, specifically dense vector embeddings, can help semi-automate the contract review process and reduce time and costs for legal professionals reviewing deontic modalities in contracts. In this study, we create a domain-specific dataset and train both baseline and neural network models for contract sentence classification. This approach offers a more efficient and cost-effective solution for contract review, mimicking the work of a lawyer. Our approach achieves an accuracy of 0.90, showcasing its effectiveness in identifying and evaluating individual contract sentences.

Keywords Natural language processing · Legal text classification · Annotation · Deep learning · Deontic reasoning

✉ S. Georgette Graham
igeorgette@hotmail.ca

Hamidreza Soltani
academic301@solent.ac.uk

Olufemi Isiaq
f.isiaq@arts.ac.uk

[1] Faculty of Business, Law and Digital Technologies, Solent University, Southampton, UK

[2] Programme Director, Computer & Data Science & AI, University of the Arts London, London, UK

Published online: 11 November 2023

1 Introduction

Lawyers often spend a significant amount of time reviewing contracts to identify and assess norms, such as permissions, obligations, and prohibitions. This process can be time-consuming and costly for clients, who are often billed based on the time spent by lawyers. To address this issue, this work proposes the use of natural language processing (NLP) techniques, specifically text classification, to streamline the contract review process with the goal of making the legal service more efficient and affordable for clients. The research question is whether NLP techniques, specifically dense vector embeddings, can help semi-automate the contract review process and reduce time and costs for legal professionals reviewing deontic modalities in contracts.

The complex language used in legal texts, including jargons, semantics, structure, and style, has made it challenging to use NLP tools effectively in the legal domain. The aim of this work is to provide a system that can analyse legal language by using machine learning techniques to categorise contract sentences based on deontic reasoning. This study contributes to the state-of-the-art by providing a corpus of deontic labels that can be used for functional classification of English-based contract sentences.

The methodology of this work includes manual annotation of contracts to create domain-specific word embeddings, which were used to train both traditional machine learning (ML) models (Naive Bayes, Logistic Regression, Support Vector Machine) and neural network (NN) models (Convolutional Neural Network and Long Short-Term Memory Recurrent Neural Network). The results of the evaluation are discussed, along with the limitations and recommendations for future work.

This paper is organised as follows: Chapter 1 introduces the problem; Chapter 2 provides background information on the relevant legal and linguistic concepts; Chapter 3 summarises and discusses the related work; Chapters 4–6 detail our methodology including data collection, pre-processing, and model training and testing; Chapter 7 examines the results, limitations, and recommendations; and Chapter 8 concludes.

2 Background

Legal professionals often face the challenge of reviewing a vast number of documents, such as contracts, to advise their clients. An important aspect of this process is identifying and assessing problematic norms, which are expressed through *deontic modalities*. Deontic modality refers to the expression of permissions, obligations, and prohibitions in legal texts (O'Neill et al. 2017). For example, a clause in a contract that states "Each party must fulfil their obligations" is a deontic sentence that prescribes the expected behaviour of the parties involved. The identification of deontic sentences, therefore, is a critical part of the contract

review process that allows legal professionals to understand the duties and obligations of each party to the contract. However, the manual process of identifying deontic sentences is time-consuming, taking up to 50% of a lawyer's time (Hendrycks et al. 2021), and retaining a lawyer can be expensive with hourly rates reaching hundreds of pounds (DIAMOND 2016). This disparity and lack of transparency in legal fees can result in a lack of access to justice for lower-income clients (BOWCOTT 2016). While it is possible for a lower-income client to review contracts without the help of an expensive lawyer, the complex language used in legal texts can make this a difficult task.

NLP techniques, such as text classification, have the potential to streamline the contract review process by automating the identification of norms in contracts. Text classification involves assigning a label to a text based on its content, making it well-suited for identifying and categorising norms in contracts. The use of NLP, including models such as Support Vector Machine (Joachims 1998), Logistic Regression (Aseervatham et al 2011), and Convolutional Neural Network (Kim 2014) can help to reduce the time and cost associated with manual contract review, increasing access to justice for all clients.

However, the complex language used in legal texts also presents challenges for NLP tools. Legal language often includes jargons, specific semantics, and specific styles that differ from ordinary language. Additionally, legal texts have a unique structure and interconnection between bodies of text, which can make it difficult to use NLP tools effectively. Despite these challenges, there has been growing interest in using NLP for legal language analysis. Maintaining the intended meaning of legal text during its transformation into numeric representation is crucial and there are different techniques used to achieve this. Sparse vectors, such as Term Frequency-Inverse Document Frequency (TF-IDF), are relatively straightforward to implement but have limited information or many zeroes in their numeric representation of words. Dense vectors, on the other hand, are better at preserving semantic meaning making them a desirable option for legal text analysis. However, popular dense vectors such as *word2vec* and *GloVe* are trained over generic corpora such as Google News and Wikipedia articles, which can result in a failure to capture the specifics of legal language. To overcome this issue, a dense vector model specifically designed for legal language called *law2vec* was created (Chalkidis and Kampas 2019). With over 169,000 words derived from over 123,000 legislative documents, law2vec serves as a public use legal word embedding model. By combining the use of a dense vector like law2vec with domain-specific embeddings derived from annotated datasets, it is possible to design an NLP tool that is well-suited for legal language analysis.

Previous research has used both sparse and dense vectors for various aspects of legal text analysis, including the identification of deontic modalities in legislative documents, topical classification of clauses in contracts, and identification of relationships between different parts of legal documents. The present study builds on this work by focusing on the functional classification of sentences in English-based contracts, with the creation of a dataset of deontic labels and the implementation of a system to classify contract sentences based on deontic reasoning.

3 Related work

The related work falls into two categories, namely:

(a) Deontic modality classification – works that are similar to the present study in that they explore text classification based solely or in part on deontic reasoning; and
(b) Text classification in contracts – studies that explore text classification in contracts.

3.1 Deontic modality classification

This section examines the literature relating to deontic modality classification in general, outside of the context of contracts.

O'Neill et al (2017) developed a system for classifying deontic modalities in financial legislative texts achieving 82.33% accuracy and F1 score of 0.79. They compared NN models such as CNN, Long Short-Term Memory (LSTM) and CNN-LSTM with non-NN models such as LR, SVM, and Decision Tree (DT). They found the LSTM model to be the most effective due to its ability to handle long-term dependency problem. The authors used a hybrid approach to word embedding by manually annotating 1297 sentences (607 permissions, 596 obligations, 94 prohibitions) from EU and United Kingdom (UK) legislation and training the word vectors on word2vec. The inter-agreement score was 0.74 indicating substantial agreement between the annotators.

Waltl et al. (2019) focused on non-NN models testing 5 classifiers, including Naïve Bayes (NB), LR and SVM, on predicting norms in German legislation achieving up to 83% accuracy. The methodology involved the annotation of 601 sentences based on 9 semantic types including duty, permission, and prohibition. The best performing model was the SVM which had 85% precision and 84% recall. The main difference between their work and ours is the formulation of the categories. The authors classified 'duty' (an action that must be done) and 'prohibition' as 'obligations'; classified 'permission' and 'indemnity' (a required action that does not have to be done) as 'rights'; and deemed indemnity and prohibition as the negative variation of permission and duty. We take a different approach as we consider a duty to be an obligation, and a prohibition to be the negation of either an obligation or permission.

Wyner and Peters (2011) examined the identification and extraction of deontic rules and conditions from regulations. Similar to the approach proposed by Aires et al. (2017), which will be discussed in the following section, Wyner and Peters identified norm sentences by considering the presence of a named party (referred to as an agent), modal verbs, and descriptions of behaviours (main verbs). They also considered exception clauses, sentence themes, and conditional sentences. However, the authors but did not address negations in their analysis. They applied

General Architecture for Text Engineering, a Java NLP toolkit, on a dataset of 1777 words achieving 100% precision and recall.

Boella et al. (2019) developed a system to assist legal professionals in understanding the meaning of legislative texts and legal concepts. Their approach system utilised Liblinear, a ML model that implements SVM and LR, to classify norms in Italian legislative documents, resulting in 70.64% precision and 79.70% recall. They used EuroVoc, a multilingual thesaurus that categorises European Union legislative documents, to categorise the text; however, most of the annotation was done manually. The key difference between their work and ours is that their focus was on identifying various roles (active or passive) and their relationship with a named entity as opposed to prescribing the modality of a sentence.

Baker et al. (2014) offered valuable insights into how modality can alter sentence meanings, even in non-legal contexts. Their semi-automated annotation scheme produced an Urdu-English modality/negation lexicon for machine translation, with an 86% precision rate for tagging. While their work covers various modalities beyond negation, the authors consider negations, such as the word "not", to be crucial for accurate event representation and translation. We similarly recognise the significance of negation, as the absence of "not" in a legal sentence can transform a prohibition into an obligation or permission. Therefore, their research influenced our approach to stop words.

3.2 Text classification in contracts

In this section, we examine the literature on text classification primarily in contracts, exploring different approaches that encompass both topical classification and functional classification based on deontic reasoning.

Lippi et al (2019) created CLAUDETTE, a system that uses machine learning to detect potentially unfair clauses in online terms of service. Their work involved the manual annotation of over 12,000 sentences from 50 online consumer contracts. Similar to our work, their approach consisted of a two-stage task, starting with binary classification where sentences were labelled as positive or negative, with positive sentences indicating the presence of potentially unfair clauses. Positive sentences were furthered classified into five unfairness categories using multilabel classification. They experimented with variations of SVM, CNN, and LSTM models achieving impressive results with over 80% detection of unfair clauses and 80% precision. The inter-agreement score was 0.871 indicating substantial agreement between the annotators. While our research differs in the specific classification task, we share the similarity of implementing binary and multilabel classification techniques to analyse contract sentences.

Hendrycks et al (2021) explored the topical classification of clauses in corporate commercial contracts to categorise them based on subject matter. For instance, a clause that stipulates the renewal of a contract was classified as 'Renewal Term' while one addressing the law that governs the contract was classified as 'Governing Law'. The authors' primary focus was the creation of a dataset for topical classification tasks, but they also used the data to train several transformer

models—Bidirectional Encoder Representations from Transformers—achieving maximum precision rate of 44% and recall rate of 80%. The authors regarded recall as more critical than precision since the task is like "finding needles in haystacks". However, we contend that precision is more important than recall in deontic modality classification, where a false negative can result in classifying an obligation as a prohibition and vice versa, leading to potentially damaging legal advice for a client.

Similar to Hendrycks et al (2021), Tuggener et al (2020) performed a topical classification of contract provisions, creating a multilabel dataset of over 60,000 corporate commercial contracts. Their approached achieved up to 95% accuracy, recall and precision. Their Logistic Regression (LR) model had a higher recall but lower precision than their NN model. The authors explained the variance in the scores by noting that the LR model, which had direct access to tokens through TF-IDF vectors, learned word associations more rapidly and thus achieved higher scores.

On the other hand, a number of works have addressed the functional classification of contract clauses, which involves categorising clauses based on their deontic modality. Firstly, Matulewska (2017) conducted an analysis of 45 contracts in British-English, American-English, and Polish to identify the various ways in which permission, obligation, and prohibition are expressed in contracts. The author also distinguished each modality as unlimited, conditional, or external, providing a detailed exploration of deontic reasoning. However, the subcategorization did not differentiate much among the modal verbs used. For example, 'shall' and 'will' are modal verbs for all three categories of obligation. Nevertheless, the author presented a practical outline of modal verbs, which we utilised in creating our Table of Modal Verbs.

Aires et al. (2017) examined the problem of norm conflicts in contracts and began by classifying modalities, an approach that aligns with the initial phase of our own research. The researchers manually annotated 92 contracts, identifying 9864 norms and 10,554 non-norm/common sentences to create a corpus. The annotated dataset was then used to train an algorithm that achieved a precision rate of 79% and a recall rate of 98%. Although no information was provided on the models or architecture used to pre-process or train the data, we still found their method of identifying norm sentences to be helpful, and we have incorporated it into our own methodology.

4 Methodology

The bulk of our work was concerned with the creation of a dataset, which we then used to train and test various models—in Fig. 1, we show our methodology specifically designed for this work. The first stage is Annotation, where sentences in contracts are manually reviewed and tagged followed by the computation of a Kappa coefficient for each pair of annotation. Next, annotation pairs are merged to create a Gold Standard corpus, which is then pre-processed using tokenisation, lemmatisation, resampling and vectorisation. The final stage is Training and Testing where several traditional and neural network models are trained over the pre-processed data.

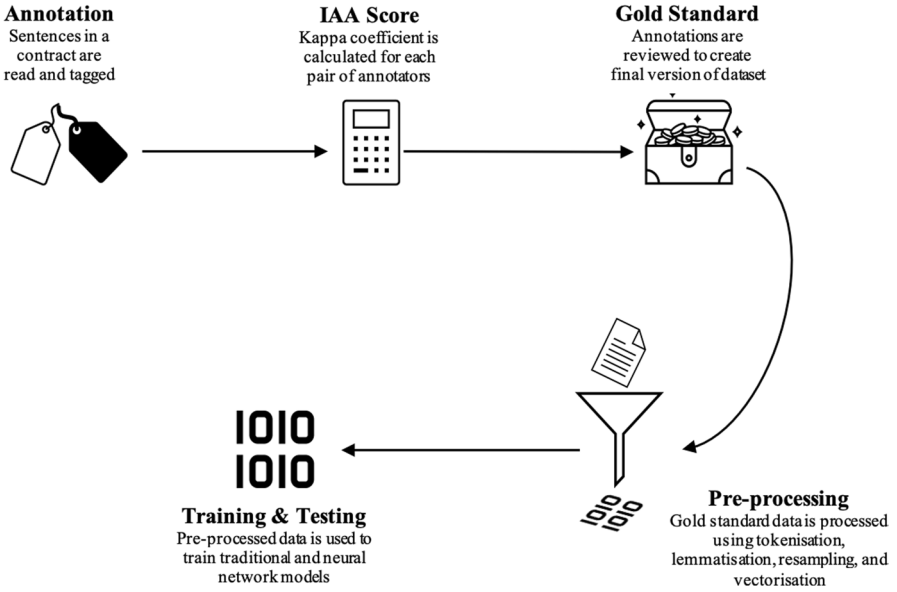

Fig. 1 Outline of our methodology. (Color figure online)

4.1 Defining deontic sentences

Deontic sentences in legal documents are often identified by the use of modal verbs. *Permission*, commonly expressed by the modal verb 'may', is a right to which a party is entitled; *obligation*, mainly identified by 'must' and 'shall', refers to a duty to perform something; and *prohibition* is a duty not to act typically written as a negation of an obligation for example "shall not". However, the interpretation of a sentence relies on context and not all sentences in a contract are deontic. Deontic sentences, also known as norms, prescribe the expected behaviour of the parties involved. In contrast, non-deontic sentences are referred to as non-norms.

Aires et al. (2017) identified norm sentences by 4 elements namely an index number/letter, named party/parties, a modal verb and a description of the behaviour expected of the party/parties—see Fig. 2. We adopted this method but with a slight modification, that is, the index number/letter was not given equal weight as the other elements as it is common to see contracts without indexing. Thus, a sentence that had no index but contained a named party, modal verb and expected behaviour was considered a norm.

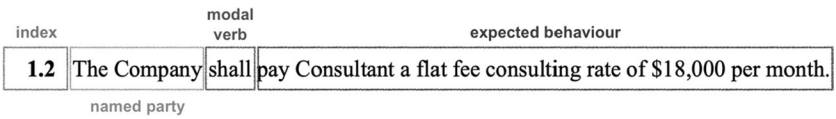

Fig. 2 Components of a norm sentence. (Color figure online)

4.2 Source dataset

The Contract Understanding Atticus Dataset (CUAD) was created using guidelines developed by The Atticus Project (Hendrycks et al 2021). It consists of 9283 pages from 510 commercial contracts retrieved from the Electronic Data Gathering, Analysis and Retrieval (EDGAR) System, a database maintained by the US Securities and Exchange Commission (SEC). In the original annotation process, a team of law students and expert attorneys in the USA manually annotated clauses in the contracts according to 41 topical categories that are considered important in contract review such as Governing Law, Document Name, Parties, Expiration Date, Agreement Date, and Renewal Term. These categories were topical, reflecting the subject matter of the reflective clauses.

We selected CUAD for our study due to its readily available data collected from the SEC website, which saved us time that would have been spent on web scraping. However, given the different classification tasks between Hendrycks et al (2021) and the current approach, we conducted a new annotation of CUAD. While the original annotation categorised clauses according to 41 topical labels, our focus is on categorising sentences into three functional labels: permission, obligation, and or prohibition.

4.3 Annotation

The task undertaken was a two-tier classification task: (a) a binary classification, where sentences are categorised as norm/non-norm, and (b) a multilabel classification where norm sentences are classified as permission/obligation/prohibition. We engaged six volunteer annotators, who are qualified lawyers practising in various areas of law including corporate, commercial, offshore, and taxation. To facilitate the annotation, a virtual workspace was created on Google Drive where annotators reviewed PDF versions of contracts and recorded their ratings in a spreadsheet. The annotation guidelines required that an annotator first checks if a sentence is a norm and if yes, assigns a tag based on the *Table of Modal Verbs*—Table 1. An example of an annotation is shown in Fig. 3.

At the end of a two-week period, 14 contracts were annotated by at least two annotators and the rest by at least one annotator. The size of the contracts ranged from 2 to 63 pages with an average review time of 36 minutes per contract.

4.4 Creating the gold standard

After the annotation process, we calculated an Inter-Annotator Agreement (IAA) score for each annotator-pair to measure how well different annotators made the same annotation decision. This score also served as a measure of whether the annotation guidelines were sufficiently clear. A popular IAA score is the Cohen's kappa, which is computed as:

Table 1 Modal verbs used to assign deontic tags

Tag		
Obligation	Permission	Prohibition
Must	Can	Can not
Ought	May	May not
Shall	Shall be able to	Must not
Will	Shall be allowed to	Ought not
To agree	Shall be entitled to	Shall not
To be bound by	Shall be permitted to	Will not
To be required to	Shall have first right to/of	Agrees not to
To represent	Will + be allowed to	Have no right to/of
To supersede	Will + be entitled to	Neither…will be liable
To undertake	Will + be permitted to	Neither + noun + shall + verb
To warrant		No + noun + shall + verb
Shall + verb + pursuant to		Shall not be entitled to
		Shall not + verb
		Will not + verb

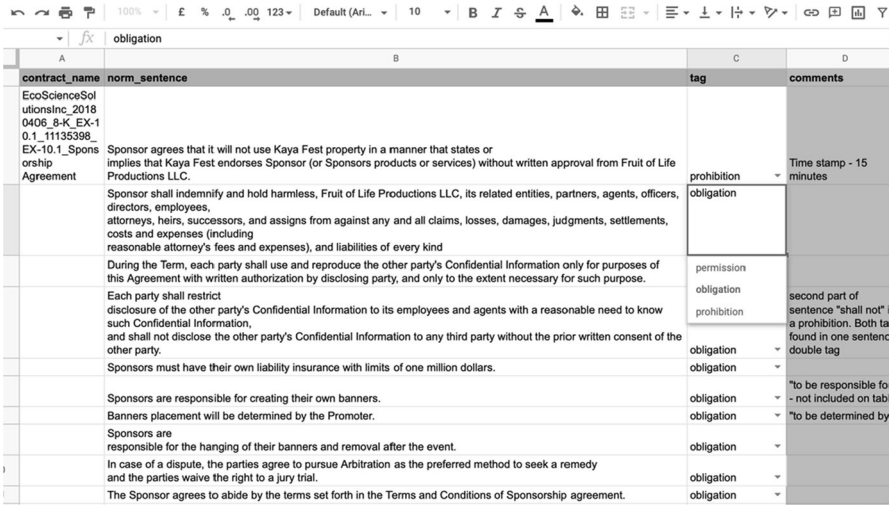

Fig. 3 An example of an annotation. (Color figure online)

$$\kappa = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)}$$

where $\Pr(a)$ is the actual observed agreement of the annotators and $\Pr(e)$ is the chance agreement (that is, what the agreement would be if the annotators randomly tagged the documents). The resulting score ranges from -1 to $+1$ where the level of agreement ranges from poor to almost perfect—see Table 2 for detail.

In computing Cohen's kappa, any sentence in a contract that was not tagged by either annotator was counted as 'untagged' and considered a non-norm sentence.

Table 2 Scale for interpreting Cohen's kappa scores

Score (κ)	Level of agreement
< 1	Poor
0.1 – 0.20	Slight
0.21 – 0.40	Fair
0.41 – 0.60	Moderate
0.61 – 0.80	Substantial
0.81 – 1.00	Almost perfect

Score refers to Cohen's kappa score and Level of agreement is the corresponding interpretation for each score

Where annotators had a different tag, we conducted a third review in line with Table 1 and decided which annotation to accept or reject, documenting our decision and reasons.

Table 3 shows the ratings of two annotators for a total of ten contracts. We computed Cohen's kappa by first finding the value for Pr(a), the percentage of observed agreement between the annotators. Out of 977 sentences reviewed, both annotators agreed on 401 untagged sentences, 97 permissions, 278 obligations, 70 prohibitions, 23 double tags, and 0 triple tags. The observed agreement thus was:

$$\Pr(a) = \frac{(401 + 97 + 278 + 70 + 23)}{977} = 0.889(88.9\%)$$

Next, Pr(e) was calculated for each tag by determining the percentage of the time that each annotator used the tag and multiplying both percentages. For instance, Annotator A tagged obligation 337 times (0.345 or 35.5% of the time) while Annotator B tagged obligation 304 times (0.311 or 31.1% of the time). The product of both, 0.345×0.311, is 0.107 so both annotators have a 0.107 chance of randomly tagging a sentence as an obligation. Performing the same calculations for the other 5 tags, the total Pr(e) is calculated by adding all 6 scores:

Table 3 Confusion matrix of annotators' ratings (total contracts = 10)

Annotator B	Annotator A						
	Permission	Obligation	Prohibition	Double Tag	Triple Tag	Untagged	**Total**
Permission	97	15	0	1	0	0	**113**
Obligation	2	278	1	3	0	20	**304**
Prohibition	1	6	70	0	0	3	**80**
Double tag	2	19	3	23	0	0	**47**
Triple tag	0	0	0	0	0	0	**0**
Untagged	5	19	6	1	1	401	**433**
Total	**107**	**337**	**80**	**28**	**1**	**424**	**977**

Annotator B and Annotator A refers to each person in the annotator-pair that reviewed the 10 contracts being discussed. The figures in bold are the total per tag per annotator

$$\Pr(e) = 0.107 + 0.007 + 0.013 + 0.001 + 0 + 0.192 = 0.32$$

Inserting the values of $\Pr(a)$ and $\Pr(e)$ into the equation for Cohen's kappa results in:

$$\kappa = \frac{0.889 - 0.32}{1 - 0.32} = 0.837$$

Based on Table 2, this score indicates an almost perfect agreement between both annotators.

Of course, Cohen's kappa is no indication of the correctness of the annotation and the tags on which the annotators could not agree proves the difficulty of the task. For instance, annotators often disagreed on whether a sentence is an obligation or a non-norm (untagged). This could be as a result of sentences having both a norm and non-norm element with one annotator deciding the entire sentence should be untagged and the other deciding the obligation aspect should be tagged. Representation and warranty clauses are an example—a representation is a statement of fact made within a contract and is not a norm whereas a warranty describes an undertaking by a party and can thus be classified as a norm. It is common to see both representations and warranties in the same sentence or clause in a contract. As shown in Fig. 4, part 2 of the sentence is a statement of fact that there is no existing conflict in relation to the agreement while part 3 creates an obligation whereby the party undertakes (or warrants) that they will not enter into any conflicting agreement; part 1 presents both sentences as one.

After reviewing each pair of annotations, a master dataset consisting of 1664 sentences was created. A second dataset of norm and non-norm sentences was also created –183 sentences initially tagged as permission/obligation/prohibition were labelled as norm (1) and 183 non-controversial sentences, that is, sentences that were untagged by both annotators, were labelled as non-norm (0). From here on, we will refer to the master dataset as the *gold standard dataset* and the second dataset as the *norm dataset*. Where the text does not specify, it should be assumed that reference is being made to the gold standard dataset. In the next section, we discuss how both datasets were analysed using Python.

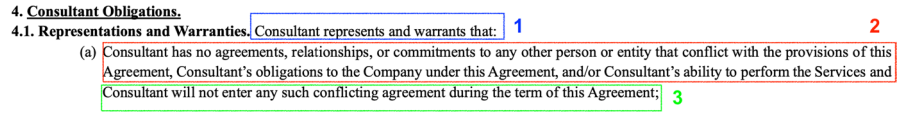

Fig. 4 Sample clause in a contract showing presence of both norm and non-norm sentences. (Color figure online)

5 Data pre-processing

As the creation of the norm dataset was more deliberate than the gold standard dataset, the resulting data consisted of 360 records split evenly between norm (1) and non-norm (0). However, the gold standard dataset, consisting of 1664 records, was highly imbalanced with over 50% of sentences tagged as 'obligation'. We downsampled the 'obligation' class resulting in a more balanced class distribution as shown in Fig. 5.

Additional data pre-processing was undertaken on both datasets to format the text such as removing line breaks, nulls, and duplicate records. We then performed tokenisation and lemmatisation using NLTK. In the former, the text is broken into smaller segments or individual words known as tokens; and in the latter, the words are transformed to their lexical roots or lemmas. It is also practice removing 'common' words (also known as stop words) from the text and libraries such as NLTK and SpaCy provide a list of stop words that can be used to do this. The list of NLTK stop words contains words that are critical to the meaning of the sentence and the tag it receives. For instance, words such as 'may', 'will', 'not' and 'neither' as seen in Table 1 would be removed. On this basis, we decided to pre-process the text without the removal of stop words.

The next stage of pre-processing is word embedding. Both sparse and dense vectors were applied. For the non-NN models, the TF-IDF vectorizer (with max feature of 2000 words) was applied to both datasets transforming the text into trigram representations. We applied TF-IDF to the training data only to prevent information entering the training phase from the test data. For the NN models, we applied Keras pre-processing to tokenise sentences, transform them to a sequence of integers and padded (for example, with 0s) to be of the same length of 200 words. For two of the NN models trained, we further applied the law2vec 100-dimensional model to pre-train the domain-specific word embeddings.

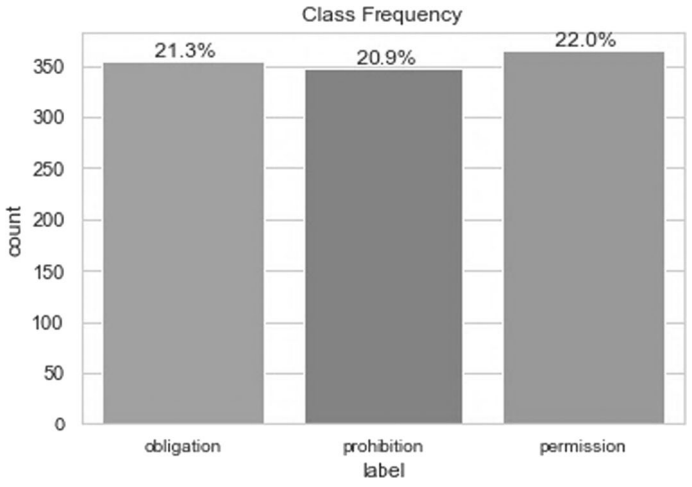

Fig. 5 Bar graph showing class distribution after resampling. (Color figure online)

6 Training and testing

We trained both NN and non-NN models in order to compare performances and select the best model for deployment. The models were trained on 80% of the dataset with the remaining 20% used for testing. For the selected best model, we performed cross-validation on the training set using the scikit-learn library's KFold function.

6.1 Binary classification results

As a baseline, 3 non-NN models were trained by iterating through a list of classifiers: SVM, LR, and a linear SVM optimised with Stochastic Gradient Descent (SGD). SVM outperformed LR achieving 88% accuracy to LR's 82%. The SVM with SGD training performed slightly better when predicting norms (obtaining F1 score of 0.88) but performed worse when predicting non-norms. The data was also trained on a CNN model whose input vectors for the embedding layer were pre-trained using law2vec. The model achieved the same accuracy as the SVM but had slightly higher F1 score. Table 4 summarises the performance of each model.

6.2 Multilabel classification results

The above non-NN models, as well as a multinomial NB classifier, were trained on the gold standard dataset. Unlike the norm dataset, which was a binary classification task, the models trained on the gold standard dataset utilised a OneVsRest strategy to fit one classifier per class thus reducing the multilabel task into independent binary tasks. Again, SVM was the best performing non-NN model with accuracy of 75% and ranking loss of 0.19. Though the NB model had the lowest accuracy, it was outperformed by the LR model, which had lower loss and a higher precision score.

On the NN side, 3 models were trained: a standard CNN, CNN with law2vec, and LSTM with law2vec. The standard CNN outperformed the other models (including the baseline models) obtaining 90% accuracy, 98% precision and ranking loss of 0.02. Overall, the worst performers were the NB and LSTM models with the NB boasting a higher precision score. Table 5 provides a summary of the performance of each model trained on the gold standard dataset.

Table 4 Performance of models trained on norm dataset

Model	Accuracy	Precision	Recall	F1 score
SVM	0.88	0.89	0.89	0.88
LR	0.82	0.84	0.86	0.83
SVM + SGD training	0.88	0.91	0.91	0.88
CNN + law2vec	0.88	0.89	0.89	0.89

Words in bold are the column headings

Table 5 Performance of models trained on gold standard dataset

Model	Accuracy	Precision score	Ranking loss
SVM	0.75	0.80	0.19
SVM + SGD training	0.73	0.78	0.19
LR	0.68	0.77	0.28
NB	0.65	0.73	0.31
CNN	0.90	0.98	0.02
CNN + law2vec	0.86	0.96	0.06
LSTM + law2vec	0.66	0.63	0.31

Words in bold are the column headings

6.3 Cross-validation

We conducted cross-validation on the best models, namely the CNN + law2vec for binary classification and CNN for multilabel classification, to ensure the reliability and consistency of the obtained results and mitigate the risk of overfitting. This involved dividing the training set into five mini-sets and training and evaluating the models on each fold.

For the binary classification task, which was trained on the norm dataset, we observed relatively low training and validation losses, indicating reasonable performance. However, the training and validation losses did not consistently decrease with each fold. This fluctuation could be attributed to several factors, including model instability, insufficient data, data quality, and model complexity. Given that the norm dataset consisted of less than 400 sentences, we believe that the inconsistency in cross-validation scores may be due to the limited size of the dataset.

On the other hand, for the multilabel classification task, trained on the gold standard dataset containing 1664 sentences, the CNN model demonstrated strong performance in terms of both training and validation losses across all folds. The training loss decreased with each fold indicating effective learning from the data, while the validation loss remained consistently low on the validation set suggesting good generalisation capabilities. These results, as presented in Table 6, provide evidence of the model's ability to learn and generalise well to unseen data.

Table 6 Cross-validation results for CNN and CNN + law2vec models

Fold	Binary classification – CNN + law2vec Dataset		Multilabel classification – CNN	
	Train loss	Validation loss	Train loss	Validation loss
1	0.0110	0.2737	0.0349	0.2009
2	0.0132	0.1912	0.0299	0.0230
3	0.0063	0.3238	0.0174	0.0139
4	0.0086	0.3611	0.0051	0.0107
5	0.0077	0.2189	0.0033	0.0021

Words in bold are the column headings

6.4 Graphical user interface

We designed a simple interactive web-based application called Contract Wiz to demonstrate how the proposed system could form part of a complete software package. At the backend of the application, various saved models operate to make a prediction according to the flowchart in Fig. 6.

The stages in the flowchart are summarised as follows:

1. **User Input**—The user is prompted to insert a sentence which is read by the application as a string.
2. **CleanText Function**—The string is cleaned by applying a function that removes punctuations and non-alphabetic characters then performs tokenisation and lemmatisation using NLTK. The result is also a string.
3. **Binary Classification**—The application determines whether the cleaned string is a norm sentence by converting it to numeric form and making a prediction using the saved CNN model trained on the norm dataset. The result is an array of the prediction in numeric form, for which the class probability is predicted. If the class probability is 0, the sentence is not a norm, and a message is displayed alerting the user of this. If the probability is 1, however, the sentence is a norm and the application proceeds to stage 4.
4. **Multilabel Classification**—The application determines the tag(s) to be assigned to the string. Firstly, the cleaned string is converted to a padded sequence of integers. The saved CNN model, trained on the gold standard dataset, then predicts the probabilities for each class, resulting in an array of 3 probabilities, representing prohibition, obligation, and permission. To determine the predicted tags

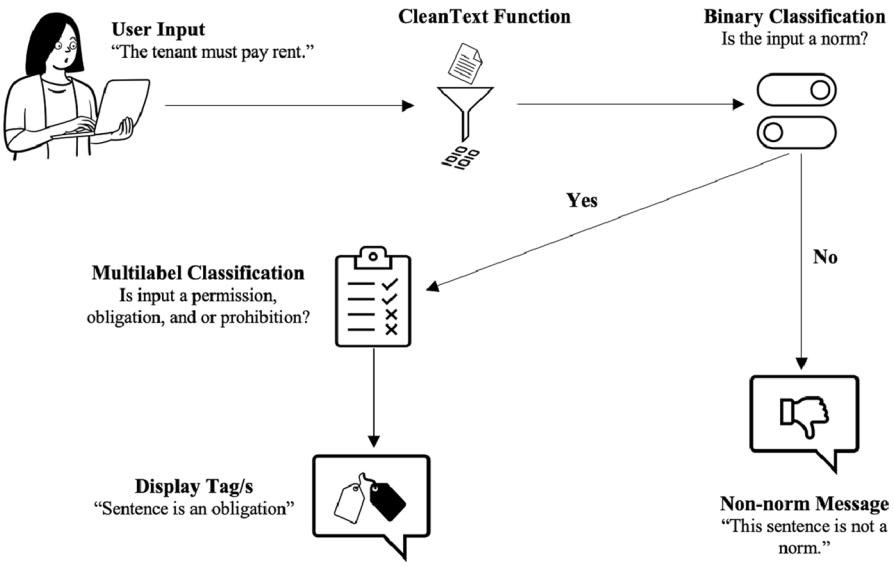

Fig. 6 Flowchart showing operation of Contract Wiz application. (Color figure online)

(classes) for the string, each class probability is compared against a threshold value of 0.5. Any probability greater than 0.5 is considered a positive prediction for that class. This approach allows for a flexible classification process, as a clause may exhibit characteristics of multiple deontic modalities. To retrieve the corresponding tag name, the class probabilities are inversely transformed using a Multilabel Binarizer. The predicted tags are displayed to the user in a summary table—See Fig. 7.

7 Discussion

In this section, we compare our results to related work and discuss the limitations of our work.

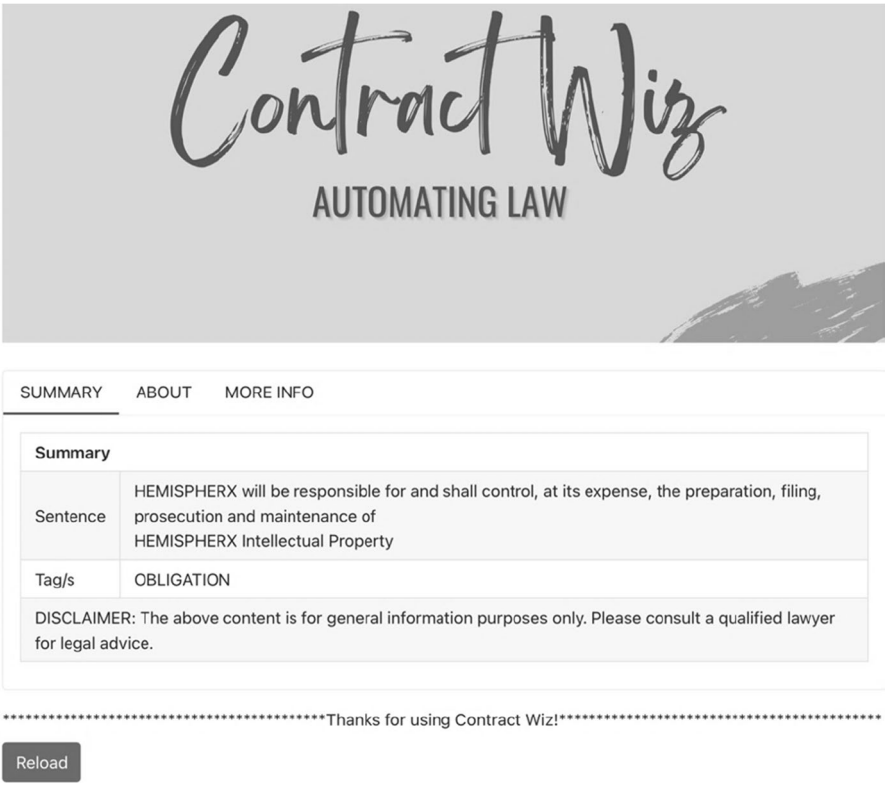

Fig. 7 Screenshot of Contract Wiz showing success page. (Color figure online)

7.1 Results

In evaluating the performance of our selected models, we used metrics such as Accuracy, Precision, Recall, F-Measure, and Ranking Loss. *Precision* refers to the fraction of predicted labels that are relevant while *Recall* refers to the fraction of relevant labels that are predicted. *F-Measure* is the harmonic mean of Precision and Recall while *Accuracy* is calculated by averaging the values for Precision, Recall, and F-Measure. In a binary classification task, these measures are often sufficient. However, for multilabel problems, evaluating performance can be challenging since a prediction is a list of classes rather than a single class thus misclassification is not as clearcut as binary problems. For instance, a prediction containing 1 or 2 labels (where it should be 3 labels) is neither completely wrong nor completely right. Consequently, we believe a loss metric such as *Ranking Loss* provides a fairer assessment of model performance. *Ranking Loss* returns the average number of label pairs that are not correctly ordered, a perfect value being 0.

With the exception of LSTM, the NN models generally performed better than the non-NN models when we compare their accuracy. This corroborates the approach taken by works such as O'Neill et al (2017) and Chalkidis and Kampas (2019). To complement the high accuracy, we examined the Ranking Loss for each model observing that the CNN model achieved negligible loss. Its notable performance on both datasets reinforces why its application to text databases is flourishing having been initially developed for use on image datasets (O'Neill et al 2017).

For comparison to earlier work, we rely on accuracy since ranking loss was not a prevalent metric in related works. As presented in Table 7, our CNN model also outperformed works that specifically addressed deontic modality classification. We view this achievement as noteworthy, particularly considering that our system outperformed works that utilised smaller datasets and more generic word embedding models. However, it is essential to note that various factors, such as dataset size, type of legal document, and type of classification including target labels, may have influenced the varying scores observed in related works. For instance, Boella et al. (2019) worked with a large dataset of legislative documents, which likely encompassed more intricate style and structural complexities compared to contracts. Additionally, it is crucial to recognise the variation in classification tasks and target labels among the studies compared. Regarding the kappa coefficient, we can only compare our results to O'Neill et al (2017) and Lippi et al (2019) in which case our annotation scheme had a slight edge with 0.889 over their 0.74 and 0.871 respectively.

At the lower end of model performance were NB, LR and LSTM. It was not surprising that the LR model (suited for linear datasets) and the NB (based on simple assumptions) were outperformed by the SVM and CNN models. However, the LSTM performing significantly worse than the CNN models was surprising since its gate mechanism is ideal for long, complex legal sentences. Its performance could possibly be improved by increasing the LSTM layers since the current architecture utilised only one layer consisting of 128 LSTM units.

Table 7 Comparison of related works based on accuracy

Work	Problem	Dataset size	Accuracy
This work	**Multilabel classification of contract sentences using 3 deontic labels – permission, obligation, prohibition**	**1664 sentences**	**0.90**
O'Neill et al (2017)	Multiclass classification of English financial legislation using 3 deontic labels – permission, obligation, prohibition	1297 sentences	0.82
Waltl et al. (2019)	Multiclass classification of legal norms in German tenancy legislation using 9 semantic labels including duty, permission, and prohibition	601 sentences	0.83
Aires et al. (2017)	Binary classification of contract sentences as norm or non-norm; multiclass classification of norm sentences using 3 deontic labels—permission, obligation, prohibition	9862 sentences	0.78
Boella et al. (2019)	Multilabel classification of semantic concepts in legislative text using, *inter alia*, semantic labels—active role, passive role, related notions	20,000 documents	0.75

7.2 Limitations

The main limitations identified concern the size of the corpus and the annotation scheme.

7.2.1 Size of corpus

Manual annotation is a time-consuming activity requiring adequate training and expertise. Given the timeframe within which the project had to be completed, only a fraction (5%) of an already small dataset was annotated. This vastly reduced the size of the corpus, which is not ideal for NLP tasks since word embeddings should be trained over large corpora (Chalkidis and Kampas 2019). While the results of our CNN model are impressive, it is worth noting that there is a possibility that the model may have learned the norm dataset too well due to its limited size of less than 400 sentences. Although we employed k-fold cross-validation to mitigate the risk of overfitting, the small size of the norm dataset remains a potential limitation to consider.

7.2.2 Annotation scheme

The success of an annotation scheme heavily depends on the clarity of the annotation guidelines. While the Cohen's kappa score indicates that the guidelines were sufficiently clear, three shortfalls have been identified. Firstly, the guidelines did not sufficiently address the situation where a sentence is both norm and non-norm. As the annotators were not instructed to tag sentences as norm, non-norm or both, the decision on which sentences to classify as non-norm in the norm dataset was largely made by one person (the reviewer).

Secondly, the definition of a norm sentence was narrowly defined as a sentence that describes a behaviour expected of a party to the contract. Yet, there are cases where norms also describe a behaviour expected from the agreement itself (Aires et al. 2017). For instance, the sentence "This Agreement shall enure to the benefit of and be binding upon the parties hereto" contains the modal verb 'shall'; however, the obligation is not directed at a party but at the agreement. As no parties are named in this sentence, the annotators, following the guidelines, would tag it as a non-norm.

Finally, norms can also be expressed by non-modal verbs. Table 8 shows examples of verbs and verb formations that were identified during the annotation process as expressing norms. For instance, the clause "each party waives" can be interpreted as an obligation (rewritten as "each party shall waive"); however, with 'to waive' absent from the *Table of Modal Verbs*, annotators would consider this sentence to be a non-norm.

7.3 Recommendations

The main recommendations relate to the foregoing limitations. With more time and resources, the annotation scheme can be perfected thus improving the accuracy of the annotations and increasing the size of the dataset. Secondly, more combination of classifiers could be explored for example the effect of stop word removal; stop

Table 8 Additional verbs that express norms in contracts

Tag	Verbs
Permission	to reserve the right to; to be free to; to agree but not be obliged to; shall have; will be free to; shall have authority to; shall not be prohibited from
Obligation	to be responsible for; to be determined by; to waive; to irrevocably submit; to irrevocably waive; to submit and agree to; to be obliged to
Prohibition	to be authorised + not; to be prohibited from; to have authority + not; shall refrain from

word removal combined with law2vec word embeddings; and the effect of other generic embeddings such as GloVe and word2vec. Additionally, a refinement of the project could involve named entity recognition to create relationships between a party and a norm. This would allow the system to perform a more thorough contract review such as identifying conflicts, missing norms, and unconscionable contracts.

Lastly, the current system (or an enhancement that merges topical classification of contract sentences) is likely to be beneficial to a lawyer by providing a quick summary of norms in a contract—the lawyer could then decide which norm to review in more detail. However, if the system is to also improve access to legal services for lower-income clients, a more holistic and client-focused approach is required. For instance, a client-user with no legal expertise will perhaps find it more useful if the system converts the legal jargon of a tagged sentence into ordinary language. This could be achieved with text summarisation techniques.

8 Conclusion

Our study has answered the question of whether NLP techniques, specifically dense vector embeddings, can help semi-automate the contract review process and reduce time and costs for legal professionals reviewing deontic modalities in contracts. We developed a system that reviews English-based contracts by classifying sentences based on their deontic modality, that is, permission, obligation, prohibition. To facilitate this research, we created a master dataset by annotating commercial contracts. This dataset, containing 1664 sentences, has been made publicly available on GitHub, enabling its use and augmentation for future work.

By employing both NN and non-NN classifiers, we evaluated the performance of our system. While non-NN models proved effective, our standard CNN (without pre-trained word embeddings) proved outstanding with minimal loss of 0.02, accuracy of 90% and up to 98% precision. These results highlight the potential of dense vector embeddings in the semi-automation of contract review, enabling legal professionals to streamline the identification and evaluation of deontic modalities in legal documents.

The involvement of legal experts as annotators in our study, who recognised the system's utility for contract review, highlights the value of their domain expertise in evaluating the effectiveness of our approach. Their positive feedback

further strengthens the justification for our work, as highlighted in our Background, which aims to address the time-consuming and expensive nature of manual contract review.

Lastly, while our approach has yielded promising results, we recognise the importance of incorporating explicability particularly when dealing with neural network models that are often considered black boxes. By incorporating deontic logic-based annotation, our model can provide transparent explanations for its predictions. This explicability not only justifies the model's classifications but also enhances its trust and credibility thus empowering legal professionals to understand the reasoning behind the automated decisions.

Acknowledgements We wish to thank the following individuals who took time out of their busy schedules to manually annotate contracts: Andrene Hutchinson, K. Teddison Maye-Jackson, Odane Lennon, Ryan Gordon and Tishanna Maxwell.

Funding No funding was received to assist with the preparation of this manuscript.

Declarations

Conflict of interest The authors declare that there are no financial or non-financial interests that are directly or indirectly related to the work submitted for publication.

Ethics approval No ethics approval is required for this study.

References

Aires JP, Pinheiro D, Strube de Lima V, Meneguzzi F (2017) Norm conflict identification in contracts. Artif Intell Law 2017(25):397–428. https://doi.org/10.1007/s10506-017-9205-x

Aseervatham S, Antoniadis A, Gaussier E et al (2011) A sparse version of the ridge logistic regression for large-scale text categorization. Pattern Recogn Lett 32(2):101–106. https://doi.org/10.1016/j.patrec.2010.09.023

Baker K, Bloodgood M, Dorr B et al (2014) A modality lexicon and its use in automatic tagging. https://doi.org/10.48550/arXiv.1410.4868

Boella G, Di Caro L, Leone V (2019) Semi-automatic knowledge population in a legal document management system. Artif Intell Law 27:227–251. https://doi.org/10.1007/s10506-018-9239-8

BOWCOTT O (2016) Legal fees investigation reveals huge disparities between law firms. The Guardian, April 5, 2016 [viewed on 06 July 2022]. Available from: https://www.theguardian.com/law/2016/apr/05/legal-fees-nvestigation-reveals-huge-disparities-between-law-firms

Chalkidis I, Kampas D (2019) Deep learning in law: early adaptation and legal word embeddings trained on large corpora. Artif Intell Law 27:171–198. https://doi.org/10.1007/s10506-018-9238-9

DIAMOND J (2016) The Price of Law. Centre for Policy Studies [viewed 06 July 2022]. Available from: https://cps.org.uk/research/the-price-of-law/

Hendrycks D, Burns C, Chen A, Ball S (2021) CUAD: An Expert-Annotated NLP Dataset for Legal Contract Review. 35th Conference on Neural Information Processing Systems (NeurIPS 2021) Track on Datasets and Benchmarks. https://doi.org/10.48550/arXiv.2103.06268

Hilpinen R (1971) Deontic Logic: Introductory and Systematic Readings. D. Reidel Publishing Company (pp 1–10)

Joachims T (1998). Text categorization with support vector machines: learning with many relevant features. In: Nédellec, C., Rouveirol, C. (eds) Machine Learning: ECML-98. ECML 1998. Lecture Notes in Computer Science, vol 1398. Springer, Berlin, Heidelberg. https://doi.org/10.1007/BFb0026683

Kim Y (2014) Convolutional neural networks for sentence classification. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) 1746–1751. https://doi.org/10.48550/arXiv.1408.5882

Lippi M, Palka P, Contissa G et al (2019) (2019) CLAUDETTE: an automated detector of potentially unfair clauses in online terms of service. Artif Intell Law 27:117–139. https://doi.org/10.1007/s10506-019-09243-2

Matulewska A (2017) Deontic modality and modals in the language of contracts. Comparat Legilinguistics 2:75–92. https://doi.org/10.14746/CL.2010.2.07

Nay J (2018) Natural language and machine learning for law and policy texts. In: Katz DM, Dolin R, Bommarito M (eds), Legal Informatics. Cambridge University Press. https://ssrn.com/abstract=3438276

Nazarenko A, Wyner A (2018) Legal NLP Introduction. TAL, 58(2):7–19

O'Neill J, Buitelaar P, Robin C, O'Brien L (2017) Classifying sentential modality in legal language: a use case in financial regulations, acts and directives. In: Proceedings of the 16th International Conference on Artificial Intelligence and Law, London, UK, June 12–15, 2017, pp 159–168

Tuggener D, von Daniken P, Peetz T, Cieliebak M (2020) LEDGAR: A large-scale multilabel corpus for text classification of legal provision in contracts. Proceedings of the 12th Conference on Language Resources and Evaluation, 11–16 May 2020, pp 1235–1241

Waltl B, Bonczek G, Scepankova E, Matthes F (2019) Semantic types of legal norms in German laws: classification and analysis using local linear explanations. Artif Intell Law 27:43–71. https://doi.org/10.1007/s10506-018-9228-y

Wyner A, Peters W (2011) On rule extraction from regulations. In: Atkinson KM (eds), Frontiers in Artificial Intelligence and Applications Volume 235: Legal Knowledge and Information Systems, pp 113–122. https://doi.org/10.3233/978-1-60750-981-3-113