

MSC APPLIED AI & DATA SCIENCE

2022

Shereese Georgette Graham

Natural Language Processing for Legal Document Review:

Categorising Deontic Modalities in Contracts

SOLENT UNIVERSITY
FACULTY OF BUSINESS, LAW & DIGITAL TECHNOLOGIES



MSC APPLIED AI & DATA SCIENCE
Academic Year 2021/22

Shereese Georgette Graham

**Natural Language Processing for Legal Document Review:
Categorising Deontic Modalities in Contracts**

Supervisor: Dr Hamidreza Soltani

September 9, 2022

This report is submitted in partial fulfilment of the requirements of
Solent University for the degree of MSc Applied AI & Data Science

ACKNOWLEDGEMENTS

I wish to thank the following individuals who took time from their busy schedules to provide guidance as I considered the feasibility of my research topic: Tom Bennett from Regulatory Genome, Joe Appleton and Dr Femi Isiaq from Solent University, David Deputy from Vertex, and my mentor, Ben Mills from Meta. Many thanks to The Atticus Project for creating CUAD, the dataset at the heart of this thesis. And last but certainly not least, I thank my volunteer annotators without which, there would be no gold standard: Andrene Hutchinson, K. Teddison Maye-Jackson, Odane C. Lennon, Ryan Gordon, and Tishanna Maxwell.

ABSTRACT

Natural language processing in the legal domain is in its infancy. Still, it is a much-needed solution to repetitive and time-consuming tasks such as contract review. There are several ways in which a contract can be reviewed using natural language processing including the classification of norm sentences into the categories of permission, obligation and/or prohibition. Such a process requires a thorough annotation scheme comprised of clear guidelines and adequate resources (data, time, expertise) followed by an appropriate word embedding method. This project outlines the methodology for an annotation scheme, albeit on a small dataset, highlights the significance of domain-specific word embedding, and further demonstrates the efficiency of convolutional neural network classifiers on multilabel classification tasks. A best result of 0.02 loss and 98% precision was achieved, a remarkable performance for a developing field.

CONTENTS

<i>ACKNOWLEDGEMENTS</i>	<i>ii</i>
<i>ABSTRACT</i>	<i>iii</i>
<i>FIGURES</i>	<i>v</i>
<i>TABLES</i>	<i>vi</i>
<i>ACRONYMS</i>	<i>vii</i>
1. INTRODUCTION	1
1.1 Background.....	1
1.2 Justification & Societal Impact.....	2
1.3 Aims & Objectives.....	3
2. RELATED WORK	5
2.1 Linguistic.....	5
2.2 General NLP.....	7
2.3 Legal Text Classification.....	12
2.4 Deontic Modality Classification.....	14
2.5 Discussion.....	17
3. RESEARCH PROCESS	18
3.1 Project Management.....	18
3.2 Data Source.....	19
3.3 Annotation Development Cycle.....	21
3.4 Annotation Scheme.....	22
3.5 Creating the Gold Standard Corpus.....	26
3.6 Exploratory Data Analysis.....	28
3.7 Data Pre-processing.....	32
3.8 Model Training & Testing.....	35
3.9 Model Deployment.....	37
4. DISCUSSION	40
5. CONCLUSION	42
4.1 Limitations.....	42
4.2 Recommendations.....	43
4.3 Conclusion.....	44
6. REFERENCES	45
<i>APPENDIX A - Gantt Chart</i>	<i>49</i>
<i>APPENDIX B - Ethics Checklist & Declaration</i>	<i>50</i>
<i>APPENDIX C - Annotation Brief</i>	<i>52</i>
<i>APPENDIX D - Annotators' Profiles</i>	<i>57</i>

FIGURES

- 1 Components of a norm sentence
- 2 Structure of a neural network
- 3 Structure of a convolutional neural network
- 4 Research steps for classifying norm sentences
- 5 Annotation workspace on Google Drive
- 6 Modal verbs used to assign deontic tags
- 7 Example of an annotation
- 8 Confusion matrix of annotator's ratings
- 9 Sample clause in contract
- 10 Value counts for tags in gold standard dataset
- 11 Bar graph showing class distribution after resampling
- 12 Stripplot showing spread of wordcount for sentences
- 13 Word cloud of popular verbs in the dataset
- 14 Code showing resampling of majority ('obligation') class
- 15 Function to clean text with NLTK
- 16 Code showing word embedding with law2vec
- 17 Flowchart showing operation of Contract Wiz application
- 18 Screenshot of Contract Wiz showing non-norm message
- 19 Screenshot of Contract Wiz showing success page

TABLES

- 1 Summary of Relevant Work
- 2 Tools used to complete project
- 3 Interpreting Cohen's kappa
- 4 Performance of models trained on norm dataset
- 5 Performance of models trained on gold standard dataset
- 6 Additional verbs that express norms in contracts

ACRONYMS

AI	Artificial Intelligence
AP	Average Precision
BERT	Bidirectional Encoder Representations from Transformers
BR	Binary Relevance
CBOW	Continuous Bag-of-Word
CNN	Convolutional Neural Network
CSV	Comma-Separated Values
CUAD	Contract Understanding Atticus Dataset
DT	Decision Tree
EDGAR	Electronic Data Gathering, Analysis and Retrieval
EU	European Union
GATE	General Architecture for Text Engineering
GloVe	Global Vectors for Word Representations
IAA	Inter-Annotator Agreement
IBM	International Business Machines Corporation
JSON	JavaScript Object Notation
LR	Logistic Regression
LSTM	Long Short-Term Memory
MATTER	Model, Annotate, Train, Test, Evaluate, Revise
ML	Machine Learning
MLPP	Multilabel Pairwise Perceptron
MMP	Multilabel Multiclass Perceptron
NB	Naïve Bayes
NER	Named Entity Recognition
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
NN	Neural Network
Non-NN	Non-Neural Network
PDF	Portable Document Format
RF	Random Forest
RNN	Recurrent Neural Network

SEC	Securities Exchange Commission
SGD	Stochastic Gradient Descent
SVM	Support Vector Machine
TF-IDF	Term Frequency-Inverse Document Frequency
US	United States
USA	United States of America
UK	United Kingdom

1. INTRODUCTION

Legal professionals are routinely required to review bulk of documents whether for an organisation's regulatory and compliance unit (Boella et al, 2018) or a client in private practice. Regarding the latter, tasks such as contract review take up 50% of a lawyer's time to identify and assess problematic norms, which are expressed by deontic modalities (permissions, obligations and prohibitions). This time factor results in high legal fees, which could dissuade lower-income clients (such as a small business owner) from seeking legal advice before signing contracts (Hendrycks et al, 2021). The use of natural language processing (NLP) techniques can reduce the time and cost involved in contract review thus improving access to legal services for lower-income clients and allowing legal professionals to use their time and skills more efficiently.

NLP is an area of artificial intelligence (AI) that converts unstructured text into numeric form so it can be understood by computers (Nay, 2018) thereby allowing computers to perform tasks such as text summarisation, information and relationship extraction, and text categorisation (Mitchell, 2020). Popular and effective NLP tools used on ordinary language texts (such as the Stanford Parser used for news corpora) are not effective on legal texts as legal language is more complex than ordinary language owing to jargons, semantics, style, interconnection between bodies of text, and structure (Nazarenko & Wyner, 2018). Ergo, this project aims to contribute to the state-of-art by researching and developing an artefact trained on an English-based dataset to review and categorise contract sentences based on deontic reasoning.

1.1 Background

In linguistics, *modality* refers to the way norms and attitudes are expressed whether in the form of (a) possibilities and necessities; (b) abilities; or (c) permissions, obligations, and prohibitions (O'Neill et al, 2017). The expression of permissions, obligations, and prohibitions is known as *deontic modality* and is concerned with the use of modal verbs such as "shall", "will", and "may". In

contract drafting, lawyers rely on deontic modality to express, without syntactic ambiguity, the duties and obligations of each party to the contract. It follows therefore that deontic modality is also at the core of the contract review process. Lawyers spend 50% of their time reviewing contracts for clauses [containing deontic modalities] that could be problematic for their clients (Hendrycks et al, 2021). A lower-income client could review business contracts without the help of an expensive lawyer; however, the complexity of the legal language - jargons, style, structure, interconnection with other texts, semantics including use of deontic modalities - would make this an arduous task.

But what if the complexity that is legal language could be understood quickly by a computer? Then perhaps, lawyers could review contracts in much shorter times and at more affordable costs. While NLP makes this possible, the legal sphere is yet to be inundated with NLP solutions (Tuggener et al, 2020).

1.2 Justification & Societal Impact

There are 2 main impacts of this project that justify it being carried out, namely: (a) improving access to legal services; (b) contributing to the state-of-art in a moderately developing field.

Firstly, the use of AI solutions for legal document review results in cost reduction as the technology is more efficient and accurate than a human lawyer (Chartis, 2019; Tromans, 2017). It is unsurprising, therefore, that organisations with a lot of resources have already adopted AI solutions to assist with their regulatory compliance functions. For instance, investment giant JP Morgan Chase has an in-house NLP software called COIN that assists the legal team with reviewing documents (Shroff, 2019) and 70% of respondents in a Chartis-IBM study (2019) stated that they use AI for their financial compliance functions.

Simultaneously, there are some organisations that lack the resources to develop in-house AI solutions (Chartis, 2019) or perceive legal fees as being expensive (YouGov, 2018). Retaining a lawyer is not cheap as hourly rates for even newly

qualified lawyers can cost hundreds of pounds (Diamond, 2016). The disparity and lack of transparency in legal fees is a dire situation which, according to senior judges in the United Kingdom (UK), could result in lack of access to justice for lower-income clients (Bowcott, 2016). However, with the use of AI, lawyers can at the very least semi-automate the contract review process and modify their legal fees to reflect the time saved. If legal services can be accessed by lower-income clients in a timely and affordable manner, there is improved access to justice thus upholding the rule of law (The Law Society, 2019).

Secondly, this project is novel as the author, at the time of writing, is not aware of any study that has addressed the issue of functional classification of clauses in English-based contracts. In related works, problems solved include:

- (a) Identification of deontic modalities in legislative documents (O'Neill et al, 2017; Boella et al, 2018)
- (b) Topical categorisation of clauses in contracts (Hendryck et al, 2021; Tuggener et al, 2020)
- (c) Identification of relationships between various parts of a legal document (Sulis et al, 2020)

What this study proposes is a hybrid of the works of O'Neill et al (2017) and Hendrycks et al (2021). The project will make the following contributions:

- i. Provision of a freely available corpus of deontic labels that can be used for functional classification of English-based contract sentences.
- ii. Development of an interface that can be used as part of an application to review contracts sourced from outside the corpus.

1.3 Aims & Objectives

This project aims to use machine learning (ML) to review contracts by classifying sentences based on deontic labels. To achieve this, the following objectives have been devised:

- i. Explore the extent to which NLP solutions are utilised in the legal industry
- ii. Examine different ML models that perform well on text classification tasks

- iii. Compare domain-specific and generic pre-trained word embedding techniques
- iv. Develop a gold standard corpus by manually annotating contract sentences
- v. Undertake exploratory data analysis on the gold standard corpus to uncover any insight about the data
- vi. Utilise NLTK and Keras to pre-process the gold standard corpus
- vii. Use the pre-processed data to train traditional ML models (Naive Bayes, Logistic Regression, Support Vector Machine) and evaluate their performance using accuracy, precision, and ranking loss
- viii. Use the pre-processed data to train neural network models (Convolutional Neural Network and Long Short-Term Memory Recurrent Neural Network) and evaluate their performance using accuracy, precision, and ranking loss
- ix. Assess the performance of the neural network models against the traditional ML models to determine the best model for scaling the project
- x. Design an interactive web-based interface using PyWebio that accepts a sentence, performs data cleaning and pre-processing, and determines what tag/s (if any) would be assigned if the review were performed by a human lawyer

This paper is organised as follows: [Chapter 1](#) introduces the research problem; [Chapter 2](#) summarises and discusses the related literature; [Chapter 3](#) details the research process including sections on data collection, pre-processing, model training and testing, and model deployment; [Chapter 4](#) discusses the performance of the models as well as the societal impact of the artefact; [Chapter 5](#) concludes. References to ‘this thesis’, ‘this project’, ‘the current project’, ‘the present study’, and ‘the current study’ should be interpreted synonymously and refers to this body of work.

2. RELATED WORK

A wide range of linguistic and technical sources were consulted. They can be grouped as:

- (a) **Linguistic** - sources that expound on linguistic concepts such as norms, modality, and deontic logic.
- (b) **General NLP** - sources that address NLP concepts and techniques such as corpus creation, word embeddings, neural networks (NN), non-neural networks (non-NN), and success metrics.
- (c) **Legal text classification** - works that explore various forms of text classification in legal documents.
- (d) **Deontic modality classification** - works that are similar to the current study in that they explore text classification based solely or in part on deontic reasoning.

2.1 Linguistic

The majority of the technical sources explained the linguistic terms that are relevant to their work. However, it is apt to extract this information from each such source and summarise it here for ease of reference.

Deontic logic - based on the Greek work *deon*- which means ‘as it should be’ or ‘duly’ - refers to the study of sentences that are comprised of logical words or normative expressions. A deontic sentence is not true or false; instead, it prescribes behaviours that are regarded as permitted, obligatory or prohibitory (Hilpinen, 1971). Applying this principle to legal texts, a norm or legal sentence (as opposed to a common/non-norm sentence) is one that prescribes the expected behaviour of a legal person (such as a party to a contract). Consequently, one cannot ask whether a legal sentence is true or false (Walzl et al, 2019). For example, a clause in a legislation that stipulates “Each citizen must pay their taxes” is not a statement of fact; instead, it expresses what is expected of a citizen in the ‘ideal world’, a deviation from which amounts to a violation (Aires et al, 2017).

Aires et al (2017) provided an adequate overview of deontic logic as it relates to contracts. A contract, according to the authors, consists of clauses that describe the expected behaviour (or norms) agreed by the parties. These norms are expressed using deontic modality identified by modal verbs (such as ‘must’, ‘may’, ‘will’). Of course, not all sentences in a contract are norms as not all sentences prescribe behaviours expected of the parties. To this end, the authors proposed a system to identify norm sentences. As shown in **Figure 1**, a norm sentence has 4 elements namely an index number/letter, named party/parties, a modal verb and a description of the behaviour expected of the party/parties. This system was adopted by the current project 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.

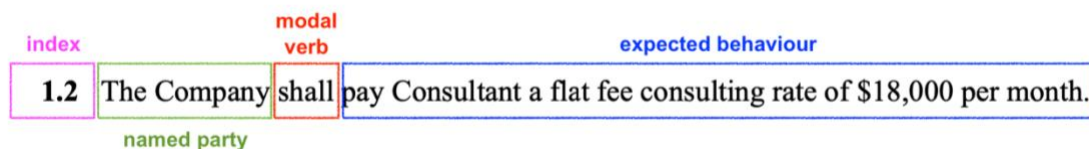


Figure 1: Components of a norm sentence

Matulewska (2017) further expounded on the concept of norms by highlighting how various deontic modalities used to express norms are identified in legal documents. *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”. It should be noted that these modal verbs are not exclusive to the deontic modalities they commonly express hence the interpretation of a sentence also relies on context. For instance, even though ‘shall’ expresses obligation, there are cases of misuse where it expresses permission (O’Neill et al, 2017).

2.2 General NLP

This section will discuss the concepts of word embedding, neural and non-neural network architectures, and evaluation metrics as detailed in the related works.

A. Word Embedding

An important part of NLP systems is word embedding or vectorization, that is, the process of embedding each word into a numerical representation or vector (Nay, 2018). This process is based on the *distributional hypothesis*, which states that words with similar meaning occur within a similar context (Harris, 1954). Word embeddings can take the form of dense or sparse vectors.

Mikolov et al (2013) and O’Neill et al (2017) focused on dense vectors. Mikolov et al (2013) introduced *word2vec*, a model that uses a shallow NN to create word embeddings from huge datasets. The model was trained over the Google News corpus (containing over 6B tokens) and takes 2 forms, namely, a continuous bag-of-words (CBOW) (predicting a word based on context) and a skip-gram architecture (predicting surrounding words/context based on a given word).

O’Neill et al (2017) compared *word2vec* with another popular model, Global Vectors for Word Representations (GloVe) trained on Wikipedia articles (Pennington et al (2014) but favoured *word2vec* which consistently performed better for their experiment. With the use of *word2vec*, the authors observed an improvement in the performance of non-NN classifiers such as Logistic Regression (LR), Support Vector Machine (SVM), and Decision Tree (DT).

On the other side of word embeddings, Boella et al (2018), Nanda et al (2018), Tuggener et al (2020), and Walzl et al (2019) utilised sparse vectors such as Term Frequency-Inverse Document Frequency (TF-IDF). *Term frequency* refers to how often a word appears in a document while *inverse document frequency* refers to the frequency of an uncommon word across a set of documents. The TF-IDF is calculated as follows:

$$tf-idf_{t,d} = (tf_{t,d}) \cdot \log \frac{N}{df_t}$$

where $tf_{t,d}$ is the frequency of a term t in a single document d , N is the number of documents in the corpus, and df_t is the frequency of the term t in the entire corpus. The rarer terms in a document set have higher TF-IDF scores thus indicating their importance to the document.

As language is the tool of the law, it is important to understand the various techniques that can be used to preserve the intended meaning of legal text when it is transformed to numeric form. While sparse vectors such as TF-IDF are easy to implement, dense vectors such as word2vec do a better job at preserving the semantic meaning of the text, which is an attractive quality for legal text analysis.

B. NLP Architectures

Sources were also consulted regarding the architecture for popular NLP models. NLP tasks such as text classification can be solved using both NN and non-NN classifiers. Non-NN models such as LR, SVM and DT were used by O'Neill et al (2017) as a baseline against which the NN models were evaluated. This work will also examine a number of non-NN models, namely NB, SVM, and LR:

(1) Naïve Bayes (NB) - This a simple algorithm often used as a baseline model as was the case in Mencia & Furnkranz (2010), Walzl et al (2019) and Nanda et al (2018). The algorithm is based on the *Bayes' Theorem*, which computes the probability of an event based on prior knowledge of conditions relating to that event (Grus, 2015). In NLP, a NB classifier calculates the probability of a class for a given input and outputs the class with the highest probability. NB, requiring a small amount of training data (Zhang, 2004), performs well as a classifier and is used in real-world applications such as spam filtering systems (Grus, 2015). However, NB's probability estimation is often poor even if the classifications are correct (Zhang, 2004). Consequently, this project also uses NB as baseline rather than a determinative model.

(2) Logistic Regression (LR) - Logistic regression is a linear algorithm for classification tasks, which uses a sigmoid function to calculate the

probability that an input belongs to a certain class. The algorithm is best suited for linearly separable classes (O'Neill et al, 2017).

- (3) Support Vector Machine (SVM) - Unlike LR models, a SVM can be applied in cases where classes are not linearly separable and operates by finding a hyperplane that maximises the nearest point in each class (Grus, 2015). The algorithm is known for performing highly on text databases (Cortes & Vapnik, 1995) and can generalize well on unseen data (O'Neill et al, 2017). However, SVMs fall short in that they do not generally perform well on massive datasets (Boella et al, 2018) though this should not be of concern in the current project.

While models such as NB and SVM have good performances on text data, NN architectures, capable of capturing the nuances of language, have been rapidly replacing these approaches (Chalkidis & Kampas, 2018). An artificial neural network (or NN) is an algorithm that simulates the way the human brain works, where a biological neuron uses the output of the neurons that feed into it to perform a calculation and decide whether or not to fire. In a NN classifier, artificial neurons perform similar calculations to biological neurons in order to produce an output (Grus, 2015). The architecture (depicted in **Figure 2**) consists of:

- (a) **an input layer**, where inputs are received and forwarded to the next layer unchanged;
- (b) **at least one hidden layer**, which is comprised of artificial neurons that accept the output of the previous layer, perform calculations, and forward the results to the next layer; and
- (c) **an output layer**, which returns the output value/s.

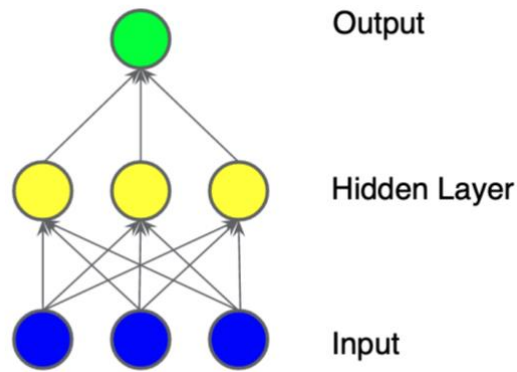


Figure 2: Structure of a neural network (Google, 2022)

While NNs perform well on linear and nonlinear data, they have several limitations including being prone to overfitting when the dataset is small (Gour, 2019) and suffering from the ‘black box’ phenomenon - it is still unclear *how* NN models solve problems (Grus, 2015).

Two common NNs are the convolutional neural network (CNN) and the recurrent neural network (RNN). CNN models are capable of capturing the sequential nature of language (Chalkidis & Kampas, 2018) thus perform well in sentiment analysis and text classification (Yao et al, 2016). The model, comprised of 3 layers (see **Figure 3**), uses locally connected layers to learn semantic representations of word vectors (O’Neill et al, 2017).

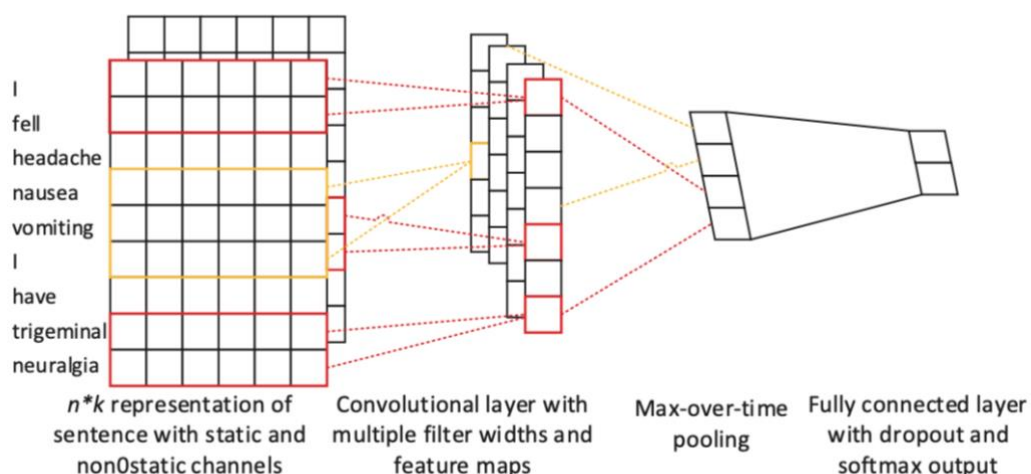


Figure 3: Structure of a convolutional neural network [input displayed is medical data] (Yao et al, 2016)

On the other hand, a RNN uses sequential-based features where information from prior inputs influences the current input and output. A Long Short-Term Memory (LSTM) network overcomes the limitation of the RNN due to the vanishing gradient problem, where it is hard for the model to learn dependencies (O'Neill et al, 2017). Consider a legal statement which begins with an obligation (“X shall do Y”) and ends with a prohibition (“but X shall not do Z”) - a LSTM model utilises a gate mechanism that stores past information (that is, the ‘obligation’ at the start of the sentence) so that the output is not constrained to a narrower window (that is, the ‘prohibition’ at the end of the sentence).

C. Evaluation Metrics

Several metrics have been used in related works to evaluate model performance. Works such as Boella et al (2018), O'Neill et al (2017), Sulis et al (2020) and Tuggener et al (2020) relied on *Precision*, *Recall* and *F-Measure*. *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 (Mencia & Furnkranz, 2010) while *Accuracy* is calculated by averaging the values for Precision, Recall, and F-Measure (Boella et al, 2018).

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, a loss metric such as *Hamming Loss* or *Ranking Loss* provides a fairer assessment of model performance. *Hamming Loss* measures the percentage of labels that are incorrectly classified while *Ranking Loss* returns the average number of label pairs that are not correctly ordered (Mencia & Furnkranz, 2010). A perfect value for both losses is 0.

2.3 Legal Text Classification

Hendrycks et al (2021) investigated the topical classification of clauses in corporate commercial contracts, that is, the categorisation of a clause based on its subject matter. For instance, a clause that stipulates the renewal of a contract would be classified as 'Renewal Term' while another addressing the law that governs the contract would be classified as 'Governing Law'. While the authors' focus was the creation of a dataset for use in topical classification tasks, they also used the data to train several transformer models - Bidirectional Encoder Representations from Transformers (BERT) - achieving best results of 44% precision and 80% recall. Regarding the low performance results, the authors submitted that recall is more important than precision since the issue is about "finding needles in haystacks". To the contrary, this study proposes that precision is more important than recall in the case of deontic modality classification where a false negative means, for instance, that a sentence has been classified as an obligation when it is in fact a prohibition - such a mistake could be damaging when advising a client of their legal risks.

Like Hendrycks et al (2021), Tuggener et al (2020) undertook a topical classification of contract provisions creating a multilabel dataset of over 60,000 corporate commercial contracts achieving up to 95% accuracy, recall and precision. The LR model had a higher recall but lower precision than the NN model. The authors attributed this variance in numbers to the fact that the LR model - which had direct access to tokens through TF-IDF vectors - learned word associations quickly thus boasting higher scores.

Chalkidis & Kampas (2018) developed a public use legal word embedding model based on word2vec. The model, referred to as law2vec, was created from over 123,000 documents and has a final vocabulary of over 169,000 words. The authors, noting the efficiency of NN models, examined 3 approaches to word representation in the legal domain:

- (i) **Generic** - publicly available models such as word2vec and GloVe that are trained over generic corpora. They typically fail to capture the semantics of legal language.

- (ii) **Domain-specific** - word embeddings trained by researchers based on annotated datasets. While this approach improves performance results, it falls short in that the embeddings are not trained over a large dataset since annotation is a time-consuming process.
- (iii) **Hybrid** - an embedding that uses both generic and domain-specific embeddings thus supplementing the domain-specific embeddings.

Mencia & Furnkranz (2010) developed a system for classifying EUR-Lex, an online repository of European Union law texts (treaties, case law, legislation). No manual annotation was involved as each document is already categorised using EuroVoc, a multilingual thesaurus that organises EU legislative documents based on thousands of categories. The problem, a multilabel classification task, was approached using three methods:

- (i) **Binary relevance (BR)**, which reduces the multi-label task to several independent binary tasks based on the number of labels, i.e., a one-vs-rest approach.
- (ii) **Multilabel multiclass perceptron (MMP)**, an extension of BR, which trains one binary classifier per label without treating the classifier independently.
- (iii) **Multilabel Pairwise Perceptron (MLPP)**, which trains a classifier for each pair of classes.

The models were evaluated using a range of metrics including Average Precision (AP), Ranking Loss, and Hamming Loss. The MLPP algorithm outperformed the other approaches obtaining just over 50% AP, considered an 'encouraging result' given the enormity of the task. This work is similar to the current project in that they are both concerned with multilabel classification. However, the current project is not as complex as there is no dual set of classes. In their work, a document was classified according to a descriptor, director code, and subject matter whereas a norm sentence, in the current project, can only be classified according to deontic modality. As such, the BR approach would be more relevant.

2.4 Deontic Modality Classification

Matulewska (2017) analysed 45 contracts in British-English, American-English and Polish to identify how permission, obligation, and prohibition are expressed in contracts. The author further distinguished each modality as unlimited, conditional, external. While this subcategorization highlights the breadth of deontic reasoning, it does not make much distinction among the modal verbs used. For instance, 'shall' and 'will' are modal verbs for all 3 categories of obligation. Nonetheless, the author provided a practical outline of modal verbs, which were adopted by the current project.

O'Neill et al (2017) designed a system for classifying deontic modalities in financial legislative texts achieving 82.33% accuracy and F1 score of 0.79. The work compared NN models such as CNN, LSTM and CNN-LSTM with non-NN models such as LR, SVM, and DT. Like Chalkidis & Kampas (2018), the NN architecture, specifically LSTM, was preferred owing to its treatment of the long-term dependency problem. The authors took a hybrid approach to word embedding by manually annotating 1297 sentences (607 permissions, 596 obligations, 94 prohibitions) from EU and UK legislation then training the word vectors on word2vec. The inter-agreement (Cohen's kappa) score was 0.74 suggesting there was substantial agreement between the annotators, who mainly disagreed on clauses that involved prohibition and obligation.

Waltl et al (2019) focused on non-NN models testing 5 classifiers, including 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. This work mainly differs from the current project in 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. The current study takes a different approach where a duty

is an obligation, and a prohibition is the negation of either an obligation or permission.

Baker et al (2014), though focusing on ordinary rather than legal language, provided useful insight on how modality changes the meaning of sentences. Using a semi-automated annotation scheme, they developed a modality/negation lexicon for Urdu-English machine translation achieving 86% precision for tagging. According to the authors, negations (such as a single instance of the word “not”) are vital for correct representation of events and translation. Though their work is broader than the current project (extending to other forms of modality), the main principle of negation can be adopted. For instance, the absence of “not” in a legal sentence could change a prohibition into an obligation or permission.

Aires et al (2017) investigated the issue of norm conflicts in contracts, the first stage of which involved the classification of modalities. The corpus was created by manually annotating 92 contracts labelling 9864 norms and 10,554 non-norm/common sentences. The dataset was then trained on an algorithm that achieved 79% precision and 98% recall. While the first stage of this work is similar and directly relevant to the present study, no information was provided on the models or architecture used to pre-process and or train the data hence no further comment can be made regarding the technical elements of the work.

Wyner & Peters (2011) examined the identification and extraction of deontic rules and conditions from regulations. In order to identify qualifying (norm) sentences, they proposed a similar approach as Aires et al (2017), that is, a norm sentence contains a named party (referred to as an agent), modal verbs, and describes a behaviour (main verb). Additionally, the authors accounted for exception clauses, sentence themes and conditional sentences but did not address negations. They applied General Architecture for Text Engineering (GATE), a Java NLP toolkit, on a dataset of 1777 words achieving 100% precision and recall.

Boella et al (2018) designed a system to aid legal professionals in understanding the meaning of legislative texts and legal concepts. The system uses Liblinear (a

ML model that implements SVM and LR) to classify norms in Italian legislative documents achieving 70.64% precision and 79.70% recall. Like Mencia & Furnkranz (2010), the EuroVoc thesaurus was used to categorise the text; however, most of the annotation was done manually. This work differs from the current project in that the overarching task is concerned with identifying various roles (active or passive) and their relationship with a named entity as opposed to merely prescribing the modality of a sentence. In light of this, the authors considered nouns to be an important feature (for named entity tagging in order to establish relationships) whereas the present study considers only verbs to be the most informative feature of the text. **Table 1** summarises the directly relevant work against which the current study will be compared.

Table 1: Summary of Relevant Work

Work	Problem	Dataset Size	Evaluation Metrics		
			Accuracy	Precision	Recall
O'Neill et al (2017)*	Classification of deontic modality in financial legislative texts - ANN, LR, SVM, DT	1297 sentences	0.82	-	-
Hendrycks et al (2021)*	Topical classification of clauses in corporate commercial contracts - BERT, DeBERTa	9283 pages from 510 contracts	-	0.44 0.90	0.80 0.18
Mencia & Furnkranz (2010)	Multilabel classification of EUR-Lex database - BR, MMP, MLPP	19,596 documents	-	0.50	-
Aires et al (2017)*	Identification of potential conflicts between contractual norms	9862 norms from 92 contracts	0.78	0.79	0.98
Wyner & Peters (2011)	Identification of deontic rules and conditions - GATE	1777 words from 4 pages	-	1.00	1.00
Waltl et al (2019)*	Classification of legal norms in German tenancy law - NB, LR, SVM, RF, MLP	601 sentences	0.83	0.85	0.84
Boella et al (2018)*	Identification of semantic concepts in legislative text	20,000 documents	0.75	0.71	0.80

* Manually annotated

2.5 Discussion

The foregoing works reveal the numerous architectures that can be used in designing a successful text classification system. While some works (Hendrycks et al (2021), Tuggener et al (2020), Chalkidis & Kampas (2018)) are advocates for NN models, it has also been proven (by Mencia & Furnkranz (2010), Walzl et al (2019), Boella et al (2018)) that non-NN models also produce excellent results and might be better fits on small datasets. This division in approach implies there is no standard NLP algorithm and that the best model depends on external factors such as dataset size, quality of annotation (if any), and pre-processing techniques. The best approach is undoubtedly that of O'Neill et al (2017), who trained both NN and non-NN models in order to determine the best fit for that specific dataset and problem.

Another observation from the literature relating to legal text classification is that annotations are problem specific. To the best of this author's knowledge, there is no standard annotated dataset for legal texts that can be used in a range of problems without more. For context, both Walzl et al (2019) and O'Neill et al (2017) designed systems for identifying norms in legislative texts and undertook separate annotation procedures. A publicly available dataset of legislative texts annotated based on deontic modality could have been used in both works. Likewise, Tuggener et al (2020) and Hendrycks et al (2021) both created publicly available datasets of annotated contracts (based on topical categories) but neither could be used in the current project without undertaking a new annotation. While this limitation could be as a result of the nature and complexity of legal language - a dynamism that changes not only between sources but across jurisdictions and languages - it could also be a reflection of the infancy of NLP in the legal domain. On the bright side, there is at least one pre-trained word embedding model, law2vec, that is the legal language counterpart for ordinary language models such as word2vec and GloVe. Though law2vec is limited in that it was trained solely on legislative texts and court decisions (excluding legal sources such as contracts), it is a much-needed development in legal domain NLP that is expected to increase the quality of word embeddings.

3. RESEARCH PROCESS

The previous [chapter](#) summarised and discussed the related work. This chapter will detail the experimental setup including the creation of the dataset, the pre-processing techniques utilised, and model training and deployment.

3.1 Project Management

The project was conducted in accordance with a 3-month timeline as detailed in the Gantt chart in [Appendix A](#). **Figure 4** outlines the research steps and **Table 2** summarises the relevant tools and software (a requirements.txt is included in the artefact files).



Figure 4: Research steps for classifying norm sentences

Table 2: Tools used to complete project

Description	Tool/s
Programming Language	Python 3.8
Integrated Development Environment (IDE)	Visual Studio Code 1.70.2
Version Control	Git & GitHub
Python Libraries & Modules	<ul style="list-style-type: none"> • matplotlib 3.5.2 • nltk 3.7 • NumPy 1.22.4 • pandas 1.4.2 • pandas-profiling 3.2.0 • seaborn 0.11.2 • wordcloud 1.8.2.2 • keras 2.9.0 • scikit-learn 1.1.1 • tensorflow 2.9.1 • joblib 1.1.0 • Flask 2.1.2 • PywebIO 1.6.1
Operating System	MacOS Monterey 12.4
Annotation	Microsoft Excel

3.2 Data Source

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). To create the dataset, a team of law students and expert attorneys in the USA manually annotated clauses in the contracts according to 41 categories that are considered important in contract review such as Governing Law, Document Name, Parties, Expiration Date, Agreement Date, and Renewal Term. The categories are topical, that is, they are based on the subject matter of the clause. For instance, a clause that stipulates the date on which the agreement is effective was tagged as ‘Effective Date’ while another that allows a party to end the contract without cause was tagged as ‘Termination for Convenience’.

CUAD was selected for the current project because the data has already been collected from the SEC website thus re-allocating time that would be spent on web scraping to other critical stages of the research process such as annotation. Another advantage of using an existing dataset is that the size is already decided; however, it is important to decide from the outset what fraction of the dataset will be annotated (Pustejovsky & Stubbs, 2013). NLP typically requires large corpora (Chalkidis & Kampas, 2018) but time, money and resources limit how much annotation can be completed (Pustejovsky & Stubbs, 2013). While CUAD is small compared to popular NLP datasets (for instance, Sentiment140 has over 1,600,000 tweets!), even smaller datasets have been used in related works: 601 sentences by Walzl et al (2019) and just one document containing 1777 words by Wyner & Peters (2011). See **Table 1** for more details.

Renowned linguist, John Sinclair (2004), proposed 10 guidelines for developing a corpus including:

- (i) Make the corpus representative as possible of the language from which it is chosen; and

- (ii) Design and composition of the corpus should be fully documented including arguments and justification of the decisions made.

Following these guidelines as well as the precedent of related works, I decided to focus on creating a quality dataset rather than trying to annotate all the contracts used in CUAD.

The different classification tasks envisioned by Hendrycks et al (2021) and the current study necessitated a new annotation of CUAD. Whereas CUAD was annotated to categorise clauses according to 41 topical labels, the current study aims to categorise sentences according to 3 functional labels, namely, whether a sentence is a permission, obligation, and or prohibition. The CUAD file (consisting of 1 CSV file, 1 JSON file, 28 Excel files, 510 PDF files and 510 TXT files) was downloaded from The Atticus Project's official website¹ in accordance with the website's disclaimer and privacy policy. For the current project, only the PDF files (unannotated contracts as downloaded from EDGAR) were needed, 29 of which were annotated in accordance with [section 3.4](#).

The following features (inspired by the CUAD Datasheet²) are important in understanding the collection of the data and creation of the dataset:

- (a) Some contracts contain redacted clauses - depicted by asterisks (***) or underscores (___) or blank spaces - to protect the parties' confidentiality. These redactions were made before the contracts were filed with the SEC.
- (b) The dataset does not contain any obscene, insulting, discriminatory, threatening or otherwise harmful data.
- (c) While the dataset contains names, addresses and other identifying information for individuals and companies, this information is not treated as confidential since the documents are publicly available on the SEC website.
- (d) The data collection process did not involve the collection of data from individuals thus no consent was required and data protection laws on data

¹ <https://www.atticusprojectai.org>

² <https://drive.google.com/drive/u/1/folders/1Yu-JnZj1LbVBfTdPiHfMDnaKZj4eqks8>

usage, retention and destruction are therefore not applicable. Though some contracts refer to parties who are individuals, these contracts are publicly available hence the information is not treated as confidential.

- (e) The annotation process was done by volunteer annotators who are, bar one, experienced lawyers. None of the volunteers were compensated. An ethics checklist and declaration, which was approved by the Ethics Administration at Solent University, is annexed as [Appendix B](#).
- (f) The Atticus Project welcomes contributions (extension, augmentation) to their dataset thus usage of CUAD in the current study is in accordance with privacy policy of The Atticus Project available at <https://www.atticusprojectai.org/privacy-policy> and their disclaimer available at <https://www.atticusprojectai.org/disclaimer>
- (g) CUAD includes contracts from 25 types of commercial contracts. The final dataset in the current project is a subset of CUAD, taking samples from the following 12 types of contracts: Affiliate, Co-Branding, Consulting, Development, Distributor, Endorsement, Franchise, Hosting, Intellectual Property, Joint Venture, License, Sponsorship.

3.3 Annotation Development Cycle

Annotation is the process by which elements of a dataset are marked up using tags, which then allow a computer to identify these tags more easily and accurately (Pustejovsky & Stubbs, 2013). Common annotation schemes include tagging words in a sentence based on their part of speech or tagging books based on genre. An Annotation Development Cycle as expounded by Pustejovsky & Stubbs (2013) has 6 stages stylised as MATTER: Model, Annotate, Train, Test, Evaluate, Revise.

- (i) **Model** - problem definition (including the identification of the type of ML task required) is an important first step as it allows the researcher to decide if an entire document or merely parts of it must be tagged. The task being undertaken by this project was initially identified as a two-tier classification task: (a) a binary classification, where sentences (rather than clauses) are categorised as norm/non-norm, and (b) a

multiclass classification where norm sentences are classified as permission/obligation/prohibition. As such, the entire document/contract required annotation. Despite the second classification task being changed to multilabel, this did not reduce the portions of the document requiring annotation.

- (ii) **Annotate** - this stage involves the training of human annotators to manually label the documents either in accordance with tags that are already a part of the document (consuming) or tags that are inserted but not associated with any part of the text (non-consuming). A topical classification task might include consuming tags since legal texts such as legislations often have a margin heading/paragraph title that describes the subject matter of the text. However, the current project uses non-consuming tags (norm, non-norm, permission, obligation, prohibition).
- (iii) **Train & Test** - the final version of the annotated text is a gold standard corpus, which can be used to train and test various appropriate algorithms.
- (iv) **Evaluate & Revise** - an evaluation of performance (including checking for errors and areas of improvement) will determine whether the method should be revised.

The [next section](#) will address stage 2 of the MATTER cycle, that is, annotation.

3.4 Annotation Scheme

It is suggested that a corpus be annotated by at least 2 people to allow a researcher to determine whether the annotation guidelines were sufficiently defined. If the guidelines are clear, the annotation can be reproduced by a larger or different group of annotators in future work. To determine if the guidelines were sufficiently clear, an Inter-Annotator Agreement (IAA) score is used to measure how well different annotators make the same annotation decision (Pustejovsky & Stubbs, 2013). A popular IAA score is the Cohen's kappa, which is computed as:

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

where $\text{Pr}(a)$ is the actual observed agreement of the annotators and $\text{Pr}(e)$ 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 3** for detail.

Table 3: Interpreting Cohen's kappa

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

Three measures were taken in order to prepare clear guidelines for annotation:

- (i) the annotation brief was written in simple language (including a list of Dos and Don'ts) to reduce the likelihood of confusion;
- (ii) visual aids such as a video demo and flowchart were prepared; and
- (iii) annotators were tested during training and asked to sign a Declaration Form if they had full understanding of the project and guidelines.

[Appendix C](#) is a copy of the Annotation Brief, which includes the guidelines, the Declaration Form, and content covered during the annotation training.

Unlike the CUAD annotation scheme, only one law school graduate was involved in the current project; all other annotators are qualified lawyers practising in Jamaica and the British Virgin Islands in various areas of law including corporate, commercial, offshore, and taxation. Given the short timeline for the project, it was decided that having experienced lawyers as annotators would reduce the training time and cost. The names of the annotators and their LinkedIn pages are provided in [Appendix D](#).

To facilitate the annotation, a virtual workspace - depicted in **Figure 5** - was created on Google Drive which includes a folder for each pair of annotators and a folder containing training resources (Annotation Brief and training presentation). Within the annotators' folders are a folder of 9 or 10 assigned contracts in PDF format and a spreadsheet on which to record their annotation. Each spreadsheet contains 2 sheets, one for each annotator (annotators were instructed not to look at their co-annotator's sheet during the process).

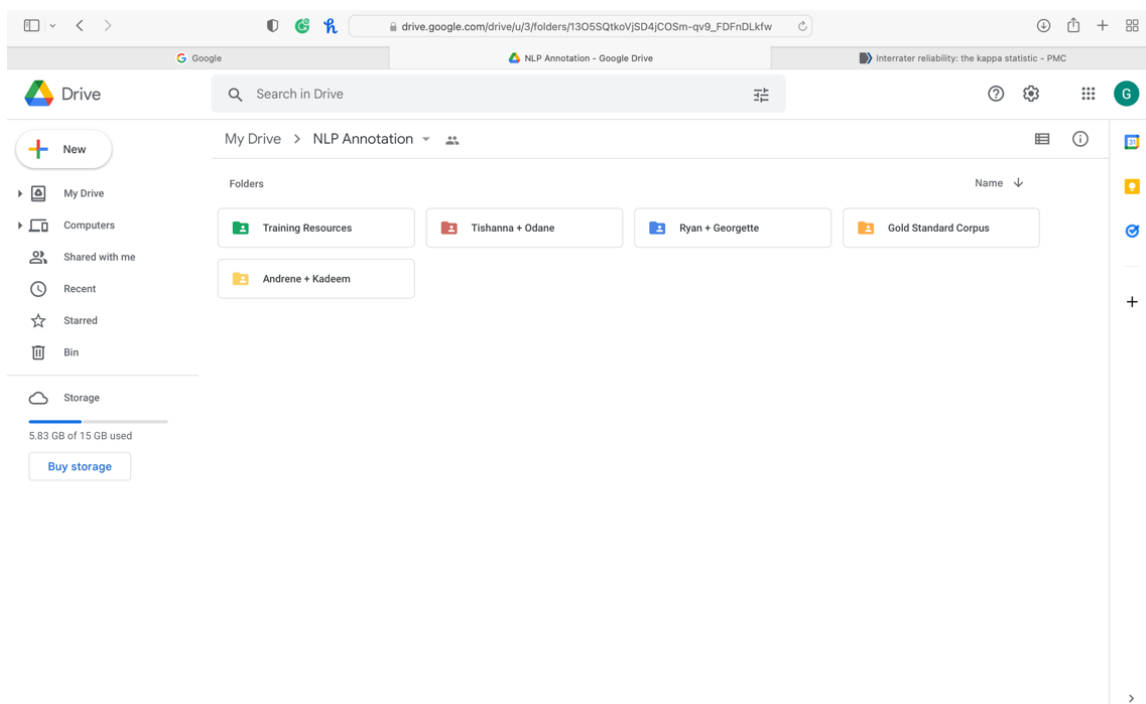


Figure 5: Annotation workspace on Google Drive

The essence of the annotation guidelines is that an annotator first checks if a sentence is a norm and if yes, assigns a tag based on the *Table of Modal Verbs* - **Figure 6**. The annotators were instructed to copy and paste all norm sentences in their assigned spreadsheets and to select the corresponding tag from the dropdown list - see **Figure 7**.

Tag:	Obligation	Permission	Prohibition
Modal Verbs	must ought shall will to agree to be bound by to be required to to represent to supersede to undertake to warrant shall + verb + pursuant to	can may shall be able to shall be allowed to shall be entitled to shall be permitted to shall have first right to/of will + be allowed to will + be entitled to will + be permitted to	can not may not must not ought not shall not will not agrees not to have no right to/of neither...will be liable neither + noun + shall + verb no + noun + shall + verb shall not be entitled to shall not + verb will not + verb

Figure 6: Modal verbs used to assign deontic tags

contract_name	norm_sentence	tag	comments
EcoScienceSolutionsInc_20180406_8-K_EX-10.1_11135398_EX-10.1_Sponsorship Agreement	Sponsor agrees that it will not use Kaya Fest property in a manner that states or implies that Kaya Fest endorses Sponsor (or Sponsors products or services) without written approval from Fruit of Life Productions LLC.	prohibition	Time stamp - 15 minutes
	Sponsor shall indemnify and hold harmless, Fruit of Life Productions LLC, its related entities, partners, agents, officers, directors, employees, attorneys, heirs, successors, and assigns from against any and all claims, losses, damages, judgments, settlements, costs and expenses (including reasonable attorney's fees and expenses), and liabilities of every kind	obligation	
	During the Term, each party shall use and reproduce the other party's Confidential Information only for purposes of this Agreement with written authorization by disclosing party, and only to the extent necessary for such purpose.	permission	
	Each party shall restrict disclosure of the other party's Confidential Information to its employees and agents with a reasonable need to know such Confidential Information, and shall not disclose the other party's Confidential Information to any third party without the prior written consent of the other party.	obligation	second part of sentence "shall not" is a prohibition. Both tags found in one sentence double tag
	Sponsors must have their own liability insurance with limits of one million dollars.	obligation	
	Sponsors are responsible for creating their own banners.	obligation	"to be responsible for" - not included on tab
	Banners placement will be determined by the Promoter.	obligation	"to be determined by"
	Sponsors are responsible for the hanging of their banners and removal after the event.	obligation	
	In case of a dispute, the parties agree to pursue Arbitration as the preferred method to seek a remedy and the parties waive the right to a jury trial.	obligation	
	The Sponsor agrees to abide by the terms set forth in the Terms and Conditions of Sponsorship agreement.	obligation	

Figure 7: Example of an annotation

During annotation, it was discovered (as it was not previously contemplated) that some norm sentences can have two or even three tags owing to compound sentences that prescribe multiple behaviours. To address this issue, the annotators were instructed to add a comment next to any such sentence - for instance, a sentence that includes both an obligation and permission was tagged as obligation with a comment stating, “double tag...permission”. This effectively changed the classification task from multiclass to multilabel.

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.

3.5 Creating the Gold Standard Corpus

After the annotation process, I reviewed each pair of annotation in order to calculate Cohen's kappa and create a gold standard, the final version of the dataset to be used for training and testing. The annotators' sheets were compared against each other and against the source contract with ratings recorded in a confusion matrix as seen in **Figure 8**. Any sentence in the contract that was not tagged by either annotator was counted as 'untagged' and considered a non-norm sentence. Where annotators had a different tag, I conducted a third review in line with the *Table of Modal Verbs* and decided which annotation to accept or reject, noting my decision and reason in the comments.

		Annotator 1						
		permission	obligation	prohibition	double tag	triple tag	untagged	Total
Annotator 2	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	<u>977</u>

Figure 8: Confusion matrix of annotators' ratings (total contracts = 10)

Using the ratings in **Figure 8**, Cohen's kappa was computed by first finding the values for $\Pr(a)$ and $\Pr(e)$. $\Pr(a)$, the percentage of observed agreement between the annotators, was calculated using the ratings where both annotators agreed on the tags. 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 is:

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

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 1 tagged obligation 337 times (0.345 or 35.5% of the time) while Annotator 2 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:

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

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

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

Based on **Table 3**, this score indicates an almost perfect agreement between both annotators. The scores calculated for the other annotators were:

- (a) 0.656 where 4 of 9 contracts were annotated by both annotators
- (b) 1.00 where only 1 of 10 contracts was annotated by both annotators

Even with partial annotation, the scores show there was substantial to almost perfect agreement between annotators thus proving that the annotation guidelines were sufficiently clear.

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 **Figure 9**, 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.

4. **Consultant Obligations.**
 4.1. **Representations and Warranties.** Consultant represents and warrants that: ¹
 (a) Consultant has no agreements, relationships, or commitments to any other person or entity that conflict with the provisions of this Agreement, Consultant's obligations to the Company under this Agreement, and/or Consultant's ability to perform the Services and Consultant will not enter into any such conflicting agreement during the term of this Agreement; ² ³

Figure 9: Sample clause in contract

After reviewing each pair of annotations, a master dataset consisting of 1664 sentences was created. A second dataset of norm and common 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, the master dataset will be referred to as the *gold standard dataset* and the second dataset referred to as the *norm dataset*. Where the text does not specify, it should be assumed that reference is being made to the gold standard dataset. The [next section](#) demonstrates how both datasets were analysed using Python.

3.6 Exploratory Data Analysis

The norm dataset consists of 360 records and 2 features, namely:

- (i) *Text*, a categorical variable; and
- (ii) *Norm*, the target label, which is a numerical variable taking one of two values (1/0).

The dataset is split evenly into norm (1) and non-norm (0) - no data imbalance was expected as the creation of the norm dataset was more deliberate than the gold standard dataset. The wordcount ranges from 7 to 555 words per sentence with an average wordcount of 43.62 words.

The gold standard dataset consists of 1664 records and 2 categorical features (*Sentence* and *Tag*). Tag consists of a string of tags: permission, obligation, prohibition, double tag (where 2 tags are present for example ['prohibition', 'permission']) and triple tag (all 3 tags are present). In its raw state, the tags are strings hence double and triple tags are returned as separate tags resulting in 10 unique values for Tag as seen in **Figure 10**.

```

# This also means that checking the value counts and data imbalance will display
# Will address this later but first, let's get an idea of how imbalanced the data is
df['tag'].value_counts()

```

[22] Python

...	['obligation']	966
	['permission']	311
	['prohibition']	269
	['obligation', 'prohibition']	43
	['prohibition', 'obligation']	19
	['permission', 'obligation']	19
	['obligation', 'permission']	19
	['prohibition', 'permission']	9
	['permission', 'prohibition']	4
	['prohibition', 'obligation', 'permission']	3

Figure 10: Value counts for tags in gold standard dataset

Even without consolidating the tags, it is clear that there is a data imbalance with 'obligation' being about 3 times the size of the other classes. This imbalance was anticipated as contracts tend to express labels disproportionately (O'Neill et al, 2017). **Figure 11** shows the class distribution after resampling the 'obligation' class, reducing it from approximately 58.1% to 21.3% of the dataset.

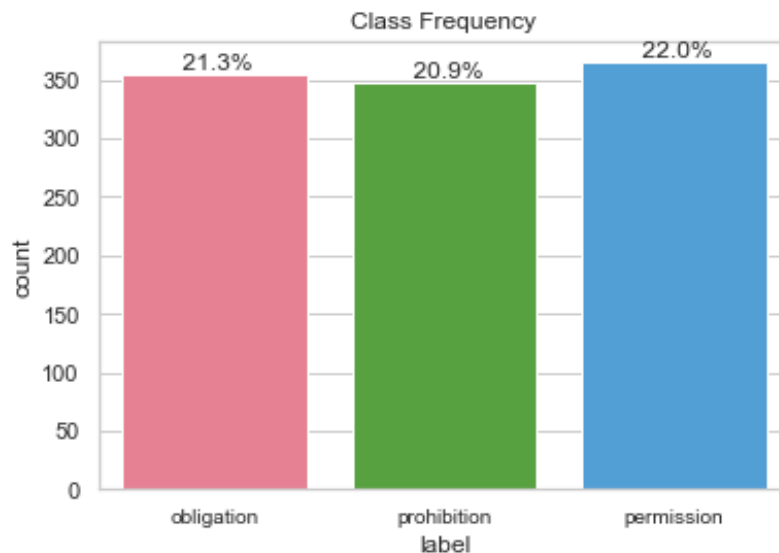


Figure 11: Bar graph showing class distribution after resampling

At the heart of the dataset are words with a strong focus on modal verbs. It was therefore important to explore the features of these words. The wordcount ranges from 7 to 691 words with a mean of 52.52 words per sentence - the graph in **Figure 12** visualises the spread of the wordcount with noticeable outliers above 300 words. The presence of outliers is not unexpected and can be explained by the nature of legal language and the divergence in contract drafting styles. Some contracts are drafted with simple, concise sentences for example the shortest sentence in the dataset is, “We may also offer optional training programs.”. Still, it is not uncommon to find verbose contracts filled with compound sentences (having independent clauses connected by words such as ‘however’ and ‘provided that’) and list structures. The sentences with the top 5 wordcounts are both compound sentences with lists. Where there is a list that includes a modal verb in the introductory clause, the entire sentence may have one tag since the modal verb applies to each item of the list. However, if the list is not preceded by a modal verb or the sentence is compound, the sentence is likely to contain a double or triple tag.

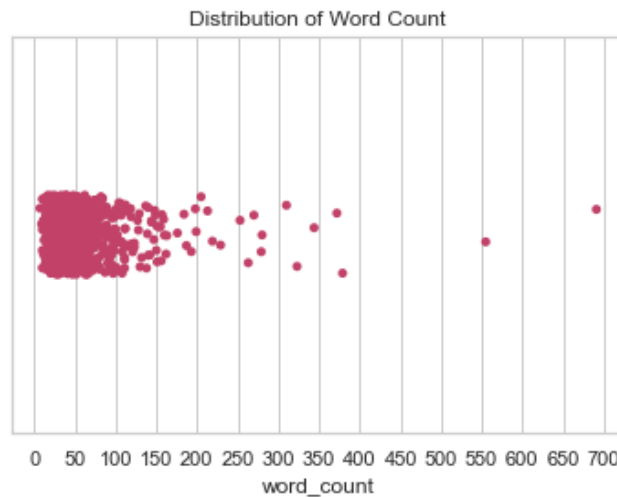


Figure 12: Stripplot showing spread of wordcount for sentences

A word cloud was used to visualise the popular words in the dataset based on frequency. As modal verbs are directly relevant to tags, the word cloud was created using only modal verbs and other verb types (by performing NLTK part of speech tagging). The resulting visual in Figure 13 shows that ‘may’ and ‘will’ are the most popular modal verbs followed by verb formations such as ‘agree’, ‘provide’ and ‘written’. While ‘may’ and ‘will’ were expected, the absence of ‘shall’ was surprising and might have been excluded due to the way the data was pre-processed for the word cloud. The word cloud also highlights many stop words, which are words often removed during NLP pre-processing. The data was thus pre-processed without removing stop words - the [next section](#) addresses this.

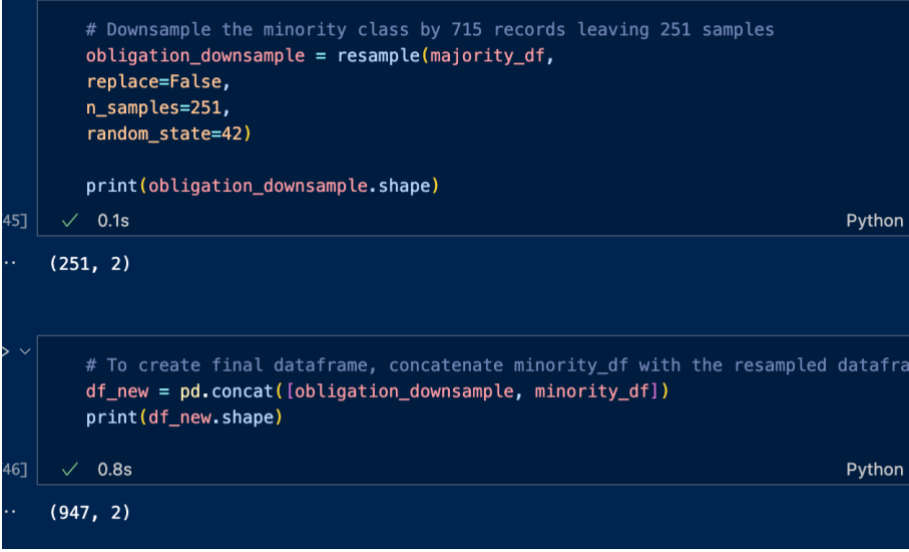


Figure 13: Word cloud of popular verbs in the dataset

3.7 Data Pre-processing

The data was pre-processed in two stages: a general text cleaning completed in Excel and Python followed by specialised pre-processing based on the ML model being trained.

After the annotation process, the gold standard corpus consisted of unformatted text - nonprinting characters, line breaks, and white spaces - resulting from several annotators copying and pasting texts. Excel's TRIM & SUBSTITUTE features were used to remove lines breaks and blank spaces. Next, there was a separate column identifying double and triple tags. As these tags were not considered pre-annotation, the spreadsheet for data collection did not allow annotators to select multiple tags. I therefore manually entered double and triple tags wherever indicated and used the CONCATENATE feature to convert the values to a list format for example ['prohibition'] and ['obligation', 'permission'].



```

# Downsample the minority class by 715 records leaving 251 samples
obligation_downsample = resample(majority_df,
replace=False,
n_samples=251,
random_state=42)

print(obligation_downsample.shape)
45] ✓ 0.1s Python
.. (251, 2)

>
# To create final dataframe, concatenate minority_df with the resampled datafra
df_new = pd.concat([obligation_downsample, minority_df])
print(df_new.shape)
46] ✓ 0.8s Python
.. (947, 2)

```

Figure 14: Code showing resampling of majority ('obligation') class

After cleaning in Excel, both norm and gold standard datasets were saved in CSV format to enable pre-processing in Python. Using Pandas, the basics were performed such as checking for nulls and duplicates. There were no missing values for either dataset; however, duplicate records were discovered and removed, and empty/unnamed columns were dropped from the gold standard dataset. The data

imbalance in the gold standard dataset was addressed by splitting the dataframe into majority ('obligation' class) and minority (all other classes) dataframes; downsampling the majority class by 715 records; and creating an updated dataframe by concatenating the minority and downsampled dataframes. **Figure 14** shows a section of the code used to resample the dataset.

The target variable (Tag), being a string, was read as having 10 unique values when there should only be 3 ('permission', 'obligation', and 'prohibition'). To address this, the string was converted to a list using AST's `literal_eval` method so that each element of the list could be read separately. With the target variable converted to usable format, the next step was pre-processing the Sentence variable.

The first step was tokenisation, that is, the separation of a body of text into smaller segments or individual words known as tokens (Nanda et al, 2018). Following tokenisation, the words were transformed to their lexical roots/lemmas in a process called lemmatisation (Boella et al, 2018). It is also common to remove 'common' words (also known as stop words) from the text and popular libraries such as NLTK and SpaCy provide a list of stop words that can be used during pre-processing. 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' and 'will' as displayed in the word cloud (**Figure 13**) and 'not' and 'neither' as seen in the *Table of Modal Verbs* (**Figure 6**) would be removed. On this basis, it was decided to pre-process the text without the removal of stop words. As seen in **Figure 15**, each sentence in the dataset was cleaned by tokenising using NLTK's `word_tokenize`, removing standalone punctuations, and lemmatising using `WordNetLemmatizer`. This function was also applied to the norm dataset.

```

# Pre-process text. Stopwords will not be excluded

# Create variable for nltk lemmatizer method
wnl = WordNetLemmatizer()

# Create function to clean text
def clean_text(sentence):
    a = []

    # Split sentence into tokens
    tokens = word_tokenize(sentence)

    # Remove tokens that are not alphabetic (e.g. standalone punctuation)
    tokens = [token.lower() for token in tokens if token.isalpha()]

    # Reduce each word to its lemma
    for token in tokens:
        lem_word = wnl.lemmatize(token)
        a.append(lem_word)

    sentence = " ".join(a)
    return sentence

# Apply function to dataframe
df_new['clean_sentence'] = df_new['sentence'].apply(clean_text)

```

Figure 15: Function to clean text with NLTK

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. As recommended by Mencia & Furnkranz (2010), TF-IDF was only applied to the training data to prevent information entering the training phase from the test data.

For the NN models, Keras pre-processing was used 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. Though the lengths of the longest sentences are 555 (norm) and 691 (gold standard), these figures are outliers being well above the averages of 43.62 and 52.52. As such, to reduce computational costs, a max length of 200 was used to minimise padding. In addition to pre-processing with Keras, the law2vec 100d model was used to pre-train the domain-specific word embeddings before fitting 2 of 3 NN models.

```

# law2vec 100 dimensional word embeddings
vocab_size = len(tokenizer.word_index) + 1

embeddings_dictionary = dict()

law2vec_file = open('./Law2Vec.100d.txt', encoding="utf8")

# Parse each line and store word-vector pairs in a dictionary
for line in law2vec_file:
    records = line.split()
    word = records[0]
    vector_dimensions = asarray(records[1:], dtype='float32')
    embeddings_dictionary[word] = vector_dimensions
law2vec_file.close()

# Each row corresponds to a word with its 100 dimensional word vector
embedding_matrix = zeros((vocab_size, 100))

# tokenizer.word_index is a list of (word, id) tuples
for word, index in tokenizer.word_index.items():
    embedding_vector = embeddings_dictionary.get(word)
    if embedding_vector is not None:
        embedding_matrix[index] = embedding_vector

```

✓ 6.1s

Figure 16: Code showing word embedding with law2vec

As shown in **Figure 16**, the law2vec file (containing 100-dimensional word embeddings) was loaded and word-vector pairs created by parsing each line of the file and storing the pairs in a dictionary. From this dictionary, an embedding matrix was created containing only those words in the dictionary that are present in the corpus.

3.8 Model Training & Testing

Like several works - Chalkidis & Kampas (2018); Boella et al (2018); Nanda et al (2018); O'Neill et al (2017) - the current project trained both NN and non-NN models in order to compare performances and select the best model for deployment. All models were trained on 80% of the datasets with the remaining 20% used for testing.

Binary Classification on Norm Dataset

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 suffered a noticeable loss of 0.26. **Table 4** summarises the performance of each model.

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	-	-	-

Multilabel Classification on Gold Standard Dataset

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 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

3.9 Model Deployment

A simple interactive web-based application called Contract Wiz was designed using PyWebio in order to demonstrate how the 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 **Figure 17**.

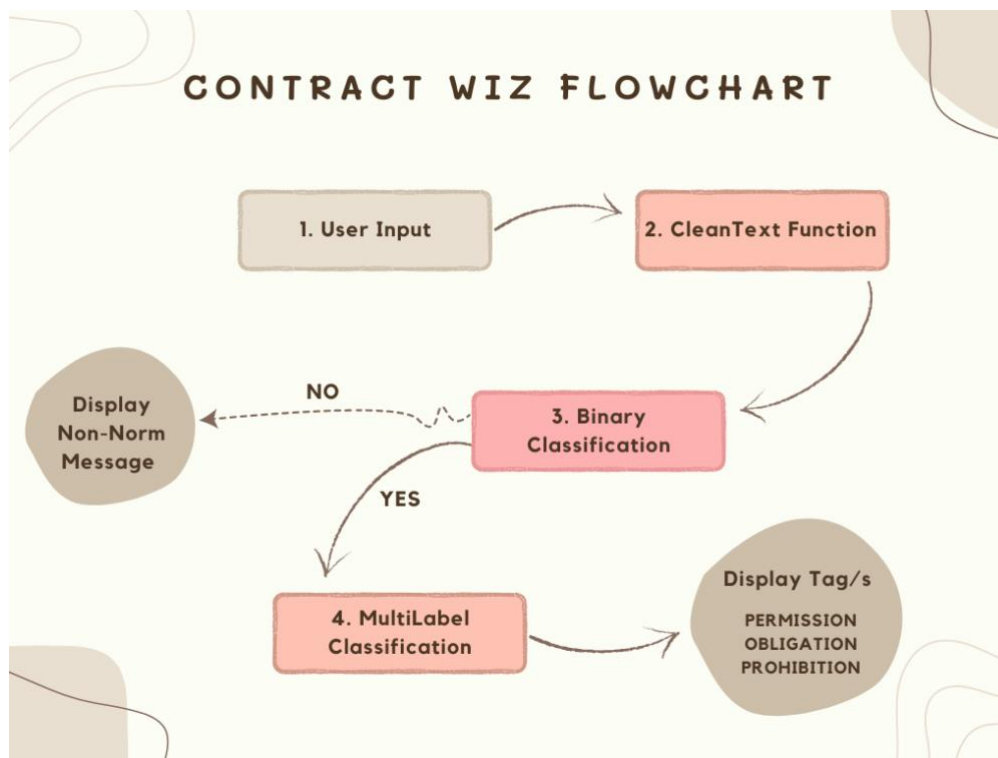


Figure 17: Flowchart showing operation of Contract Wiz application

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 - see **Figure 18**. If the probability is 1, however, the sentence is a norm and the application proceeds to stage 4.

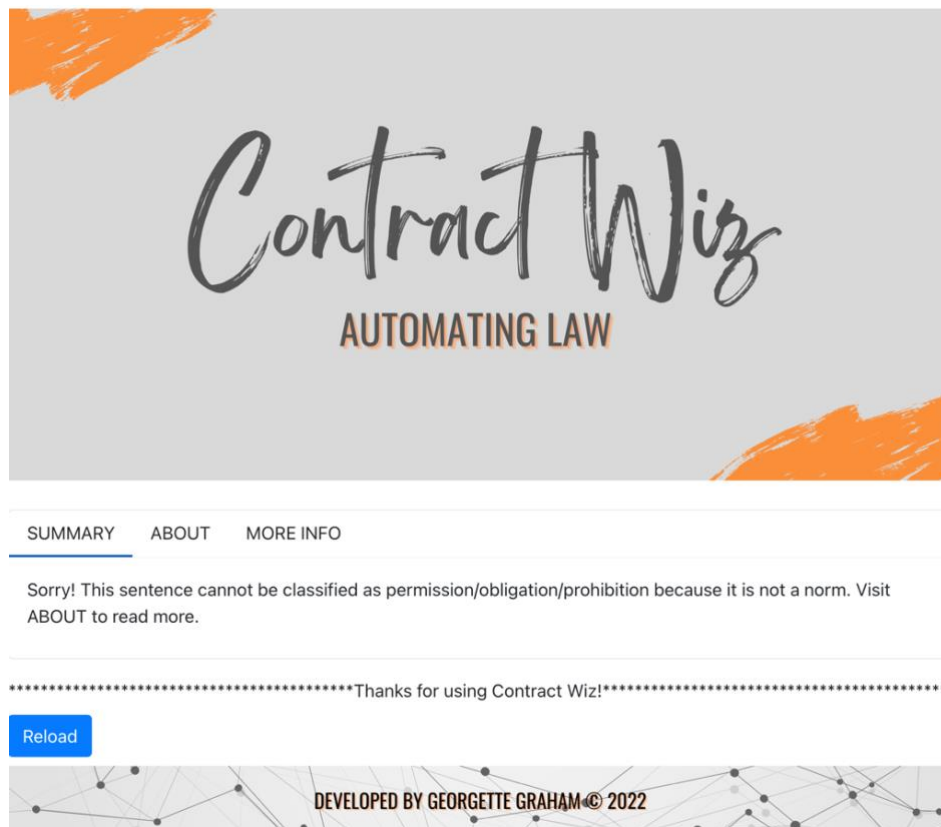


Figure 18: Screenshot of Contract Wiz showing non-norm message

4. **Multilabel Classification** - the application determines the tag/s to be assigned to the string. The cleaned string is converted to a padded sequence of integers and a prediction made using the saved CNN model trained on the gold standard dataset. The result is an array of 3 predictions. The class probabilities are calculated and inversed transformed to retrieve the class string using the Multilabel Binarizer. The predicted tags (classes with probability greater than 0.5) are displayed to the user in a summary table - see **Figure 19**.

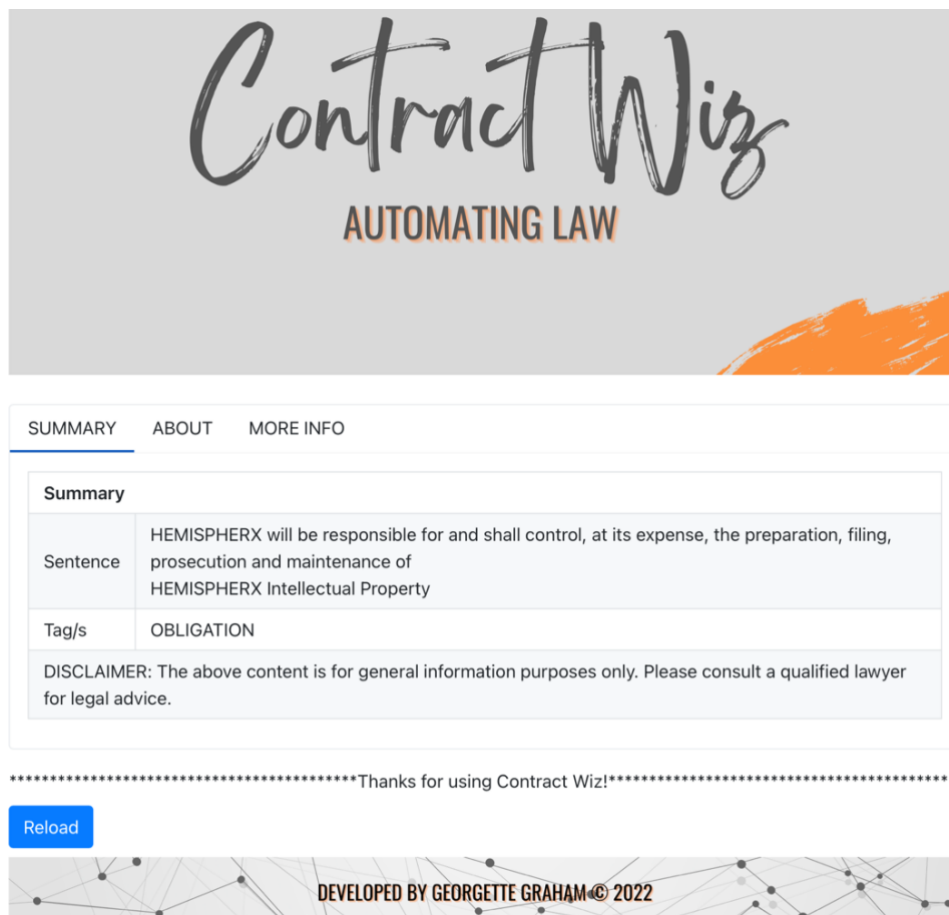


Figure 19: Screenshot of Contract Wiz showing success page

4. DISCUSSION

With the exception of LSTM, the NN models generally performed better than the non-NN models. This corroborates the approach taken by works such as O'Neill et al (2017) and Chalkidis & Kampas (2018). Of course, as earlier discussed, accuracy is not a fair measure of a system's performance particularly where multilabel classification is involved; therefore, the high accuracy received for both NN and non-NN models must be met with scrutiny. If evaluation is based on ranking loss, for instance, only the CNN model is worth mentioning as it 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).

In addition to outperforming the other models trained in the current project, the CNN results also outperformed related work (see **Table 1**). Nonetheless, the model's limitations should not be ignored. For instance, the small size of the dataset is a recipe for overfitting when dealing with NN models and it is possible the model learned the data too well, particularly the norm dataset which contained less than 400 sentences.

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.

The promising performance of the NN models means there is possibility for real-world impact. A system that semi-automates the contract review process could allow lawyers to utilise their time more efficiently, Ideally, this system would also reduce the cost of contract review thus improving access to legal services for lower-income clients. In the real world, however, there are 2 main drawbacks to

this ideal. Firstly, technological solutions, such as an application that semi-automates contract review, can result in less billable hours for lawyers. And secondly, the use of such technology, while improving value to clients, is likely to negatively impact the work of junior lawyers, to whom contract review and similar time-consuming administrative tasks are usually assigned (Croft, 2017). Nevertheless, I concur with Susskind & Susskind (2015) that these issues can be resolved by implementing alternative billing frameworks where it is the value received (from the technology) rather than the effort expended that dictates a client's fees.

5. CONCLUSION

The preceding chapters have discussed related works, outlined the data creation and pre-processing techniques, and analysed the performance of the models trained. This chapter will conclude by discussing the limitations of the project and making recommendations for future work.

4.1 Limitations

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

A. Size of corpus

Manual annotation is a time-consuming activity (Sulis et al, 2018) 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 & Kampas, 2018).

B. Annotation scheme

The success of an annotation scheme heavily depends on the clarity of the annotation guidelines. While the Cohen's kappa scores indicate 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 researcher).

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 6** 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.

Table 6: 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

4.2 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 word removal combined with law2vec word embeddings; and other generic embeddings such as GloVe and word2vec. Additionally, a refinement of the project could involve named entity recognition (NER) 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.

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 NER and text summarisation techniques.

4.3 Conclusion

This project developed a system (including an interactive web-based interface) that reviews English-based contracts by classifying sentences based on deontic modality, that is, permission, obligation, prohibition. The system involved the annotation of commercial contracts to create a master dataset which was pre-processed with sparse and dense word embeddings in order to train both NN and non-NN classifiers. The dataset, containing 1664 sentences, will be publicly available on GitHub for augmentation (or use as is) in future work. While non-NN models proved sufficient, the standard CNN (without pre-trained word embeddings) proved outstanding with minimal loss of 0.02 and up to 98% precision. With the use of a small dataset, this work has contributed to an emerging principle in legal domain NLP, that is, the classification of norms. By perfecting the annotation scheme and increasing the size of the dataset, future work can result in a more comprehensive contract review system.

6. REFERENCES

- AIRES, J.P. et al, 2017. Norm conflict identification in contracts. *Artificial Intelligence and Law* (2017) 25: 397-428
- BAKER, K. et al, 2014. A Modality Lexicon and its use in Automatic Tagging [viewed 10 July 2022]. Available from: <https://arxiv.org/abs/1410.4868>
- BOELLA, G., L. DI CARO and V. LEONE, 2018. Semi-automatic knowledge population in a legal document management system. *Artificial Intelligence and Law* (2019) 27: 227-251
- 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-investigation-reveals-huge-disparities-between-law-firms>
- CHALKIDIS, I. and D. KAMPAS, 2018. Deep learning in law: early adaptation and legal word embeddings trained on large corpora. *Artificial Intelligence and Law* (2019) 27: 171-198
- CHARTIS, 2019. *AI in RegTech: a quiet upheaval*. Co-Branded Report. 04 February 2019 [viewed 11 June 2022]. Available from: <https://www.chartis-research.com/technology/artificial-intelligence-ai/ai-regtech-quiet-upheaval-10726>
- CORTES, C. and V. VAPNIK, 1995. Support-vector networks. *Mach Learn* 20 (3): 273-297
- CROFT, J., 2017. Artificial intelligence closes in on the work of junior lawyers. *Financial Times*, May 4, 2017 [viewed on 07 September 2022]. Available from: <https://www.ft.com/content/f809870c-26a1-11e7-8691-d5f7e0cd0a16>
- 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/>
- GOOGLE, 2022. *Neural Networks: Structure* [viewed on 06 September 2022]. Available from: <https://developers.google.com/machine-learning/crash-course/introduction-to-neural-networks/anatomy>
- GOUR, R., 2019. Artificial Neural Network for Machine Learning - Structure & Layers. *Medium*. February 15, 2019 [viewed 06 September 2022]. Available from: <https://medium.com/@rinu.gour123/artificial-neural-network-for-machine-learning-structure-layers-2a275f73f473>
- GRUS, J., 2015. *Data Science from Scratch First Principles with Python*. O'Reilly Media, Inc.

- HARRIS, Z., 1954. Distributional structure. *Word* 10, 2-3 (1954), 146-162
- HENDRYCKS, D. et al, 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 [viewed 24 June 2022]. Available from: <https://doi.org/10.48550/arXiv.2103.06268>
- HILPINEN, R., 1971. *Deontic Logic: Introductory and Systematic Readings*. D. Reidel Publishing Company (pp 1-10).
- LAW SOCIETY, THE, 2019. Technology, Access to Justice and the Rule of Law. *The Law Society*, September 16, 2019 [viewed 06 July 2022]. Available from: <https://www.lawsociety.org.uk/en/topics/research/technology-access-to-justice-and-the-rule-of-law-report>
- MARASOVIC, A. et al, 2016. Modal Sense Classification at Large. *Linguistic Issues in Language Technology* 14 (2016). In: O'Neill, J. et al. 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
- MATULEWSKA, A., 2017. Deontic Modality and Morals in the Language of Contracts. *Computer Languages, Systems & Structures*, 2, 75-92.
- McHUGH, M.L., 2012. Interrater Reliability: the Kappa Statistic. *Biochemia Medica*, 2012 Oct; 22(3): 276-282 [viewed 31 August 2022]. Available from: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3900052/>
- MENCIA, E.L. and J. FURNKRANZ, 2010. Efficient Multilabel Classification Algorithms for Large-Scale Problems in the Legal Domain. In: Francesconi, E., Montemagni, S., Peters, W., Tiscornia, D. (eds), *Semantic Processing of Legal Texts. Lecture Notes in Computer Science()*, vol 6036. Springer, Berlin, Heidelberg
- MIKOLOV, T. et al, 2013. Efficient Estimation of Word Representations in Vector Space [viewed 25 June 2022]. Available from: <https://doi.org/10.48550/arXiv.1301.3781>
- MITCHELL, E., 2020. *Natural Language Processing for RegTech: Uncovering Hidden Patterns in Regulatory Documents*. Elder Research, 10 July 2020 [viewed 11 June 2022]. Available from: <https://www.elderresearch.com/blog/natural-language-processing-for-regtech-uncovering-hidden-patterns-in-regulatory-documents/>
- NANDA, R. et al., 2018. Unsupervised and supervised text similarity systems for automated identification of national implementing measures of European directives. *Artificial Intelligence and Law* (2019) 27: 199-225
- NAY, J., 2018. Natural Language and Machine Learning for Law and Policy Texts. In D.M. Katz, R. Dolin and M. Bommarito (eds), *Legal Informatics*. Cambridge University Press

NAZARENKO, A. and A. WYNER, 2018. Legal NLP Introduction. *Traitement Automatique des Langues*, Volume 58 - No. 2/2017, 7-19

O'NEILL, J. et al, 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

PENNINGTON, J., R. SOCHER and C.D. MANNING, 2014. *GloVe: Global Vectors for Word Representation* [viewed on 27 July 2022]. Available from: <https://nlp.stanford.edu/projects/glove/>

PUSTEJOVSKY, J. and A. STUBBS, 2013. *Natural Language Annotation for Machine Learning*. O'Reilly Media, Inc. (pp 1-137).

SHROFF, R., 2019. Natural Language Processing in Banking: Current Uses. *Towards Data Science*. December 27, 2019 [viewed 07 June 2022]. Available from: <https://towardsdatascience.com/natural-language-processing-in-banking-current-uses-7fbbae837de>

SINCLAIR, J., 2004. Developing Linguistic Corpora: a Guide to Good Practice. *AHDS Literature, Language and Linguistics*. [viewed 15 July 2022]. Available from: <https://users.ox.ac.uk/~martinw/dlc/chapter1.htm>

SULIS, E. et al, 2020. Exploring Network Analysis in a Corpus-Based Approach to Legal Texts: a Case Study. *CEUR Workshop Proceedings*, Vol 2690, 27-38 [viewed 23 June 2022]. Available from: <http://ceur-ws.org/Vol-2690/>

SUSSKIND, R. and D. SUSSKIND, 2015. *The Future of the Professions*. Oxford University Press. (pp 137)

TROMANS, R., 2017. Legal AI: A beginner's guide. *Legal Insights Europe*, February 20, 2017. [viewed 07 June 2022]. Available from: <https://legalsolutions.thomsonreuters.co.uk/blog/2017/02/20/legal-ai-beginners-guide/>

TUGGENER, D. et al, 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. et al, 2019. Semantic Types of Legal Norms in German Laws: Classification and Analysis Using Local Linear Explanations. *Artificial Intelligence and Law* 27: 43-71 [viewed on 15 July 2022]. Available from: <https://doi.org/10.1007/s10506-018-9228-y>

WYNER, A. and W. PETERS, 2011. On Rule Extraction from Regulations. *Frontiers in Artificial Intelligence and Applications* 235: 113-122 [viewed on 16 July 2022]. Available from: <https://doi.org/10.3233/978-1-60750-981-3-113>

YAO, C. et al, 2016. A Convolutional Neural Network Model for Online Medical Guidance. IEEE Access, vol 4, pp 4094-4103 [viewed on 06 September 2022]. Available from: <https://ieeexplore.ieee.org/document/7523892>

YOUGOV and EUROPE ECONOMICS, 2018. *Price transparency in the legal services market: a study of small businesses with legal issues*. YouGov Plc [viewed on 06 July 2022]. Available from: <https://www.sra.org.uk/sra/research-publications/price-transparency/>

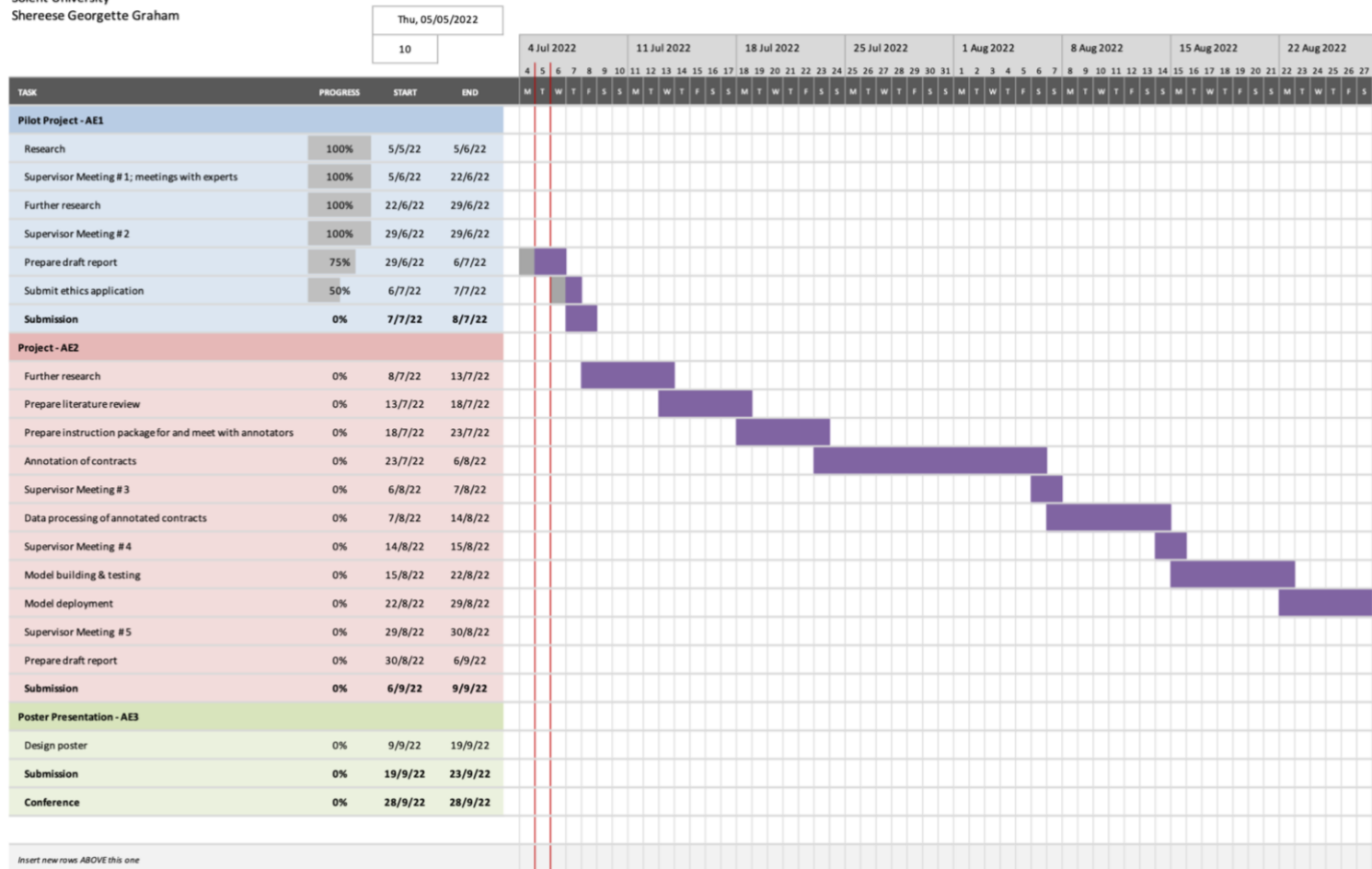
ZHANG, H., 2004. *The Optimality of Naïve Bayes*. [viewed on 06 September 2022]. Available from: https://scikit-learn.org/stable/modules/naive_bayes.html

APPENDIX A - Gantt Chart

MSc Applied AI & Data Science 2021/22 Dissertation

Solent University
Shereese Georgette Graham

SIMPLE GANTT CHART by Vertex42.com
<https://www.vertex42.com/ExcelTemplates/simple-gantt-chart.html>



APPENDIX B - Ethics Checklist & Declaration

Ethical clearance for research and innovation projects

Project status

Status

Approved

Actions

Date	Who	Action	Comments
12:18:00 08 July 2022	Femi Isiaq	Supervisor approved	
19:18:00 06 July 2022	Shereese Graham	Principal investigator submitted	
11:59:00 05 July 2022	Shereese Graham	Principal investigator saved	

Ethics release checklist (ERC)

Project details

Project name:

Principal investigator:

Faculty:

Level:

Course:

Unit code:

Supervisor name:

Supervisor search:

Other investigators:

Checklist

Question	Yes	No
Q1. Will the project involve human participants other than the investigator(s)?	<input checked="" type="radio"/>	<input type="radio"/>
Q1a. Will the project involve vulnerable participants such as children, young people, disabled people, the elderly, people with declared mental health issues, prisoners, people in health or social care settings, addicts, or those with learning difficulties or cognitive impairment either contacted directly or via a gatekeeper (for example a professional who runs an organisation through which participants are accessed; a service provider; a care-giver; a relative or a guardian)?	<input type="radio"/>	<input checked="" type="radio"/>
Q1b. Will the project involve the use of control groups or the use of deception?	<input type="radio"/>	<input checked="" type="radio"/>
Q1c. Will the project involve any risk to the participants' health (e.g. intrusive intervention such as the administration of drugs or other substances, or vigorous physical exercise), or involve psychological stress, anxiety, humiliation, physical pain or discomfort to the investigator(s) and/or the participants?	<input type="radio"/>	<input checked="" type="radio"/>
Q1d. Will the project involve financial inducement offered to participants other than reasonable expenses and compensation for time?	<input type="radio"/>	<input checked="" type="radio"/>
Q1e. Will the project be carried out by individuals unconnected with the University but who wish to use staff and/or students of the University as participants?	<input type="radio"/>	<input checked="" type="radio"/>

Q1e. Will the project be carried out by individuals unconnected with the University but who wish to use staff and/or students of the University as participants?

Q2. Will the project involve sensitive materials or topics that might be considered offensive, distressing, politically or socially sensitive, deeply personal or in breach of the law (for example criminal activities, sexual behaviour, ethnic status, personal appearance, experience of violence, addiction, religion, or financial circumstances)?

Q3. Will the project have detrimental impact on the environment, habitat or species?

Q4. Will the project involve living animal subjects?

Q5. Will the project involve the development for export of 'controlled' goods regulated by the Export Control Organisation (ECO)? (This specifically means military goods, so called dual-use goods (which are civilian goods but with a potential military use or application), products used for torture and repression, radioactive sources.) [Further information from the Export Control Organisation](#)

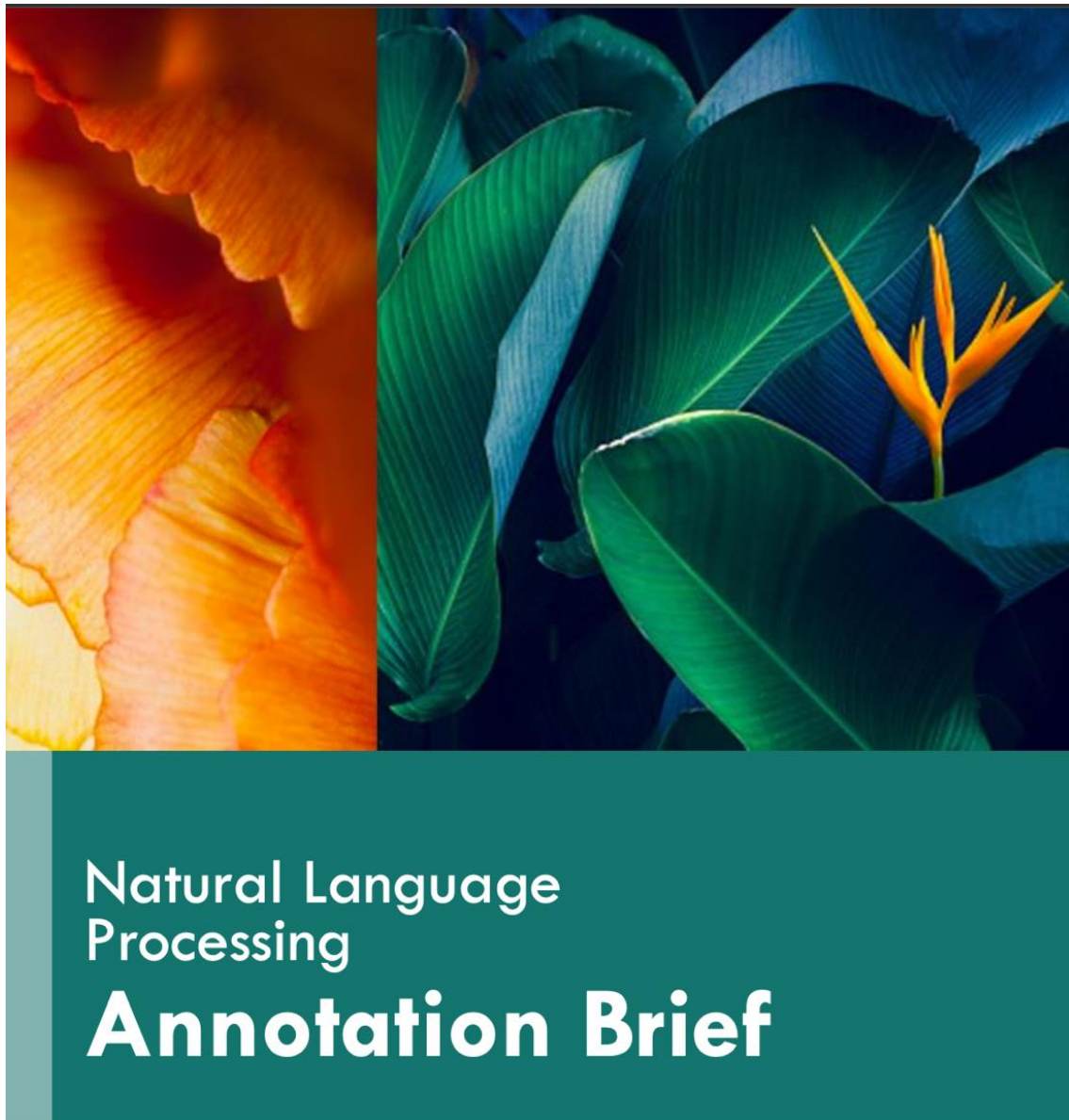
Q6. Does your research involve: the storage of records on a computer, electronic transmissions, or visits to websites, which are associated with terrorist or extreme groups or other security sensitive material? [Further information from the Information Commissioners Office](#)

Declarations

I/we, the investigator(s), confirm that:

- The information contained in this checklist is correct.
- I/we have assessed the ethical considerations in relation to the project in line with the University Ethics Policy.
- I/we understand that the ethical considerations of the project will need to be re-assessed if there are any changes to it.
- I/we will endeavour to preserve the reputation of the University and protect the health and safety of all those involved when conducting this research/enterprise project.
- If personal data is to be collected as part of my project, I confirm that my project and I, as Principal Investigator, will adhere to the General Data Protection Regulation (GDPR) and the Data Protection Act 2018. I also confirm that I will seek advice on the DPA, as necessary, by referring to the [Information Commissioner's Office further guidance on DPA](#) and/or by contacting information.rights@solent.ac.uk. By Personal data, I understand any data that I will collect as part of my project that can identify an individual, whether in personal or family life, business or profession.
- I/we have read the [prevent agenda](#).

APPENDIX C - Annotation Brief



Natural Language Processing for Legal Document Review:
Categorising Deontic Modalities in Contracts

By S. Georgette Graham, Solent University, 2021/22

Table of Contents

Welcome.....	3
Project Overview.....	4
The Technical Bits.....	4
Project Timeline.....	4
Dataset.....	5
Annotation Basics.....	5
Table of Modal Verbs.....	6
Annotation Steps.....	7
Declaration.....	8

Hi!

Thank you for volunteering to be a part of my Applied AI & Data Science dissertation. Natural language processing for functional classification in contracts is an emerging field, which requires domain knowledge to manually tag contracts. Annotators are essential to the first and most crucial step of the project – data creation – and I am delighted that you wish to contribute your time, energy, and expertise.

This document outlines your role as an annotator and provides key information about the research process. If you need clarification on any section, just send me a message.

With thanks,
Georgette



Project Overview

Natural Language Processing for Legal Document Review: Categorising Deontic Modalities in Contracts

Contract review is widely considered a boring, repetitive activity that takes up approximately 50% of a lawyer’s time. The review process is mainly carried out to identify and assess problematic clauses. Such clauses are likely to be in the form of norms, which use deontic statements of obligation, prohibition, and permission. These statements are usually expressed by use of modal verbs e.g. *shall, may, must*.

The automation of the identification of deontic statements in contract review can save time and cost thus allowing lawyers to use their time and skills more efficiently. The goal of this project, therefore, is to develop a machine learning model that reviews and categorises contract clauses based on deontic modality.

The Technical Bits

In order to develop a machine learning model that can categorise contract clauses, a computer must be trained with examples of target clauses. There are many pre-trained corpora (body of text) that are used to train computers. However, they were created from sources such as news articles and Wikipedia and are thus not very effective on legal documents, which are often complex owing to legal jargons, semantics, style, and structure. It is therefore useful to create a domain-specific corpus that has been annotated by experts.

During annotation, a document or sentence is manually reviewed and tagged as belonging to a particular category. For example, an annotator could read and categorise law reports as “criminal” or “civil”. The annotated document is then converted into numbers (literally 1s and 0s) so it can be learned by a computer (training). If the computer has been trained and tested to a satisfactory standard, it can then categorise documents never before seen.

Project Timeline

Task	Date
Annotation training	Jul 21 – 22
Annotation	Jul 22 – Aug 6
Model building (pre-processing of annotated contracts, model building, testing & deployment)	Aug 7 – 29
Project submission	Sept 9
Forum and project presentation	Sept 28

Dataset

The Contract Understanding Atticus Dataset (CUAD) is a dataset created from 510 commercial contracts scraped from the US Securities and Exchange Commission (SEC) website. A team of law students and expert attorneys in the USA manually annotated the contracts to tag clauses in accordance with 41 categories that are considered important in contract review. For instance, a clause speaking of the expiration date of the contract was tagged as “Expiry” while another stating which law should apply in interpreting the contract was tagged as “Governing Law”.

CUAD needs to be re-annotated for this project since the classification task is different – whereas CUAD was initially annotated according to 41 topical labels, the current project aims to classify clauses according to 3 functional labels, i.e., deontic statements of obligation, prohibition and permission.



Annotation Basics

A sentence can be a **norm** or **non-norm**. The key difference between the two is that a norm sentence contains a modal verb (any verb that indicates the presence of an obligation, permission or prohibition). The structure of a norm sentence is:

Indexing number → party’s name → modal verb → behaviour description

Each norm sentence in each contract should be annotated with the following tags:

1. **Obligation** – a behaviour/duty that must be executed; can equate to the negation of a permission or prohibition not to act
2. **Permission** – a behaviour that is allowed to be executed
3. **Prohibition** – a behaviour that must not be executed otherwise there is a violation

Table of Modal Verbs details the verbs that relate to each tag.

Table of Modal Verbs

Tag:	Obligation	Permission	Prohibition
Modal Verbs	must ought shall will to agree to be bound by to be required to to represent to supersede to undertake to warrant shall + verb + pursuant to	can may shall be able to shall be allowed to shall be entitled to shall be permitted to shall have first right to/of will + be allowed to will + be entitled to will + be permitted to	can not may not must not ought not shall not will not will not agrees not to have no right to/of neither...will be liable neither + noun + shall + verb no + noun + shall + verb shall not be entitled to shall not + verb will not + verb



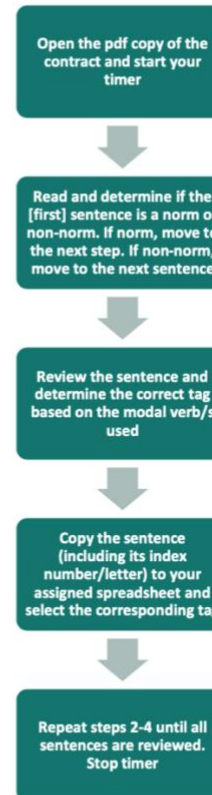
Annotation Steps

Do:

- ✓ Record the time taken to review each contract
- ✓ Highlight and bring to the attention of the researcher any challenges e.g. any ambiguous clauses
- ✓ Tag sentences in accordance with the **Table of Modal Verbs** even if your interpretation differs
- ✓ Use the designated spreadsheet, which has been formatted to allow ease of data processing
- ✓ Seek clarification, if needed

Don't:

- ✗ Annotate non-norm sentences
- ✗ View your co-annotator's spreadsheet
- ✗ Edit norm-sentences even where errors are identified




Declaration

By my signature below, I, _____
acknowledge that I have read and understand the Annotation
Brief and specifically that I:

- (a) understand the aim of the project;
- (b) have completed the Annotation Training;
- (c) understand my role as an annotator; and
- (d) am participating in this project as a volunteer
annotator and have not been provided any financial
inducement to do so.

Signature

Date



Can robots be lawyers? 😊

Natural Language Processing Annotation Brief

© Georgette Graham 2022

APPENDIX D - Annotators' Profiles

Annotator	LinkedIn URL
Andrene Hutchinson	https://www.linkedin.com/in/andrenehutchinson
K. Teddison Maye-Jackson	https://www.linkedin.com/in/k-teddison-maye-jackson-7954504b
Odane C. Lennon	https://www.linkedin.com/in/odane-c-lennon
Ryan Gordon	https://www.linkedin.com/in/ryan-gordon-ba7176180
S. Georgette Graham	https://www.linkedin.com/in/sgeorgettegraham
Tishanna Maxwell	https://www.linkedin.com/in/tishanna-maxwell-38167483