SMU Data Science Review

Volume 7 | Number 1 Article 4

Using NLP to Model U.S. Supreme Court Cases

Katherine Lockard
Southern Methodist University, klockard12@gmail.com

Robert Slater
Southern Methodist University, rslater@smu.edu

Brandon Sucrese
Southern Methodist University, bsucrese@smu.edu

Using NLP to Model U.S. Supreme Court Cases

Katherine Lockard, Brandon Sucrese, Robert Slater

Master of Science in Data Science, Southern Methodist University,
Dallas, TX 75275 USA

klockard@smu.edu
bsucrese@smu.edu
rslater@smu.edu

Abstract. The advantages of employing text analysis to uncover policy positions, generate legal predictions, and inform or evaluate reform practices are multifold. Given the far-reaching effects of legislation at all levels of society these insights and their continued improvement are impactful. This research explores the use of natural language processing (NLP) and machine learning to predictively model U.S. Supreme Court case outcomes based on textual case facts. The final model achieved an F1-score of .324 and an AUC of .68. This suggests that the model can distinguish between the two target classes; however, further research is needed before machine learning models are used in the Supreme Court.

1 Introduction

The United States judicial system has many levels; however, the highest-level court is the Supreme Court. The Supreme Court hears cases that have been appealed by lower courts to provide a final ruling. Every year 7,500 cases are sent to the Supreme Court; however, less than 150 of these are heard each year (The White House, para. 11). The number of cases sent to the Supreme Court has increased substantially over time. In 1950, the Supreme Court received 1,195 cases and in 1975, it received 3,940 cases (Supreme Court of the United States, section 4). Due to the dramatically increased volume of cases, the Supreme Court receives “100 or more cases” that are removed from consideration “without plenary review” (Supreme Court of the United States, section 4).

The Supreme Court is different than lower courts because it does not usually hold trials. Rather, the Supreme Court hears cases that are related to the constitutionality of a law. Ultimately, the Supreme Court “interprets the meaning of a law” or “rule[s] on how a law should be applied” based upon the Supreme Court’s interpretation of the Constitution. Once a decision is made by the Supreme Court, all lower courts are required to abide by this decision (The White House, para. 10). As the ultimate decision maker in the American judicial system, the Supreme Court is responsible for providing Americans “the promise of equal justice under law” and acting as the “protector and interpreter of the Constitution.” (Bahn, n.d.).

The most common way a case is sent to the Supreme Court is by filing a petition called a writ of certiorari (The United States Courts, section 2). This petition is filed against the lower court that previously ruled on the case, requesting that the case be sent to the Supreme Court for further review. Four of the nine Supreme Court Justices must vote to accept the case for the case to be heard in the Supreme Court (The United States Courts, section 2). If the case is denied by the Supreme Court, this makes the decision in the lower court final. The average length of a Supreme Court case varies. Usually, cases that are decided unanimously are released sooner, some as soon as three months. However, more controversial cases may not be decided until the end of the Supreme Court's term (The United States Courts, section 8). This means that controversial cases can last up to nine months in the Supreme Court.

The U.S. judicial system is overwhelmed with cases. One way to avoid going to trial is to accept a plea bargain from the prosecutor of a case. While time is saved by not going to trial, there is concern that innocent defendants are being wrongfully coerced to accept these bargains (Finkelstein & Levin, 2020). The average number of guilty pleas, across 29 federal district courts, has increased from 63% in 1975 to 89% in 2017 (Finkelstein & Levin, 2020). While this does not directly affect the Supreme Court, some of these cases, if heard, might have escalated to the Supreme Court. As stated earlier, the Supreme Court is responsible for providing Americans "the promise of equal justice under the law" (Bahn, n.d.). This can only happen if the U.S. judicial system has the appropriate time and resources to hear cases.

The Supreme Court has a strong influence in shaping the laws of the United States and has a responsibility to serve as the final interpreter of the Constitution. Therefore, creating a predictive model for Supreme Court decisions could be a useful tool for those considering escalating their case to the Supreme Court, legal teams preparing for the Supreme Court, or law clerks deciding which cases to send to the Supreme Court Justices.

By employing machine learning and natural language processing to automate extracting variables of interest from existing court documents, this research can provide insight into the major factors that lead to verdicts. These tools will support legal case experts and researchers in modeling underlying relationships between case facts and case outcomes.

Building insight into case outcomes can allow for legal teams to focus time and energy on the areas that are most important, while better understanding the potential outcomes of the cases they are overseeing. Developing models to predict U.S. Supreme Court rulings will allow for a better understanding of inner workings of judicial branch, getting an insight on several factors that might play a role in a certain ruling that may not be obvious.

This research aims to develop further knowledge around the research into the U.S. legal system. Taking a critical look into the factors that are used to predict the outcomes of U.S. Supreme Court cases.

2 Literature Review

2.1 Legal Diction

When building natural language processing (NLP) models, the training materials used are important to garner reliable results. Whether NLP is being used in a predictive modeling or text generation field, it is important to have a clear understanding of the documents used and the expected results.

Many of the legal case documents that are accessible tend to be missing valuable information needed to perform the modeling processing. One example is the results of the case, this information is rarely given easily and must be extracted from reading the case conclusions manually (Petrova et al., 2020). This can often be a tedious task and can limit the number of cases used in the dataset, which can permutate throughout the training processing. This can limit which training methods can be used and can potentially damage the overall results received.

This is further exacerbated when looking to build models for domain specific industries. Processing legal case documents is different than processing financial contract documents. Models can struggle to recognize the nuisances of the domain specific terms and sentence structures that are important to understanding the document (Zhang et al., 2022). This means for the model to be effective, there needs to be a wide range of documents from that domain specific sector. The diversity in the available documents can drastically change the results depending on the model's application.

While text processing is important for model building, text generation continues to leave large areas of improvement. Plain text generation is not satisfactory when building various legal documents. Legal documents must follow a standardized structure and important pieces of information need to be added to follow protocol (Zhang, Y., 2021). It is not enough to just build generalized text; perfection is a requirement for the legal domain. The utilization of NLP in text generation of official documents can have large ramifications if done poorly. While NLP has made great strides in improving text generation, it has not yet met the required standards to be used in widespread applications of legal document generation (Zhang, Y., 2021).

The textual information given to a model is extremely important in receiving reliable results. Training data should be chosen carefully and reflect the application of the model.

2.2 Uncovering Policy Positions, Legal Predictions, & Informing Reform

Legal text analysis allows legal professionals and citizens to better understand the practices and language that lead to an outcome of interest in a time-intensive field. A field in which heavily regulated practices and language require extensive training. Further, legal corpora exist in abundance as first-hand historic registries and continue to be produced in large volume at present. The static nature of textual data lends itself to impactful, replicable research that may be iteratively improved alongside technological algorithmic progress and discoveries in the data science field.

In addition to analyzing American policy positions, advancements in NLP have introduced the possibility of unearthing policy positions through historical documents written in languages other than English. By developing and applying a text analysis method independent of semantics analysis, researchers were able to determine the policy stance of various British, Irish, and German parties (Laver et al., 2003). While the British and Irish policies analyzed were English text, the German documents were not. The models were effective in accurately modeling parties' economic and social political sentiments. These findings support the authors' contentions that their process is deployable and replicable without regard for language semantics, reducing the prohibitive time and labor costs of manually building training sets (Laver et al., 2003).

In contrast Nay's 2017 research utilizes an ensemble model capitalizing on the semantic structure to predict the likelihood of a Congressional bill passage. Notably, approximately 4% of congressional bills presented are passed (Nay, 2017). When making predictions against infrequently seen outcomes, the need to model both the meaning extracted from the text and contextualizing metadata about bill sponsors and their legal influence arises (Nay 2017). Similarly, Park and Hassairi (2021) model the importance of both legislator influence and legislature content as it impacts the successful enactment of early care education (ECE) policies. In contrast to Laver et al. (2003), these researchers employ a posteriori method to discover the six topic areas that govern the ECE field through Latent Dirichlet Allocation (LDA) applied in an NLP context.

By analyzing both failed and successful legislations the researchers discovered two larger groupings, ECE services and finance, each with 3 subcategories as follows. ECE policies and practices on society at large affect a pervasive impact on societies directly through the socialization of the youngest generations and indirectly through their impact on caregiver's day to day (Park & Hassairi, 2021). The discovery of these meta-policy priorities educates legal actors on the ECE policy topics and content yielding success. These comprehensive insights drawn from a body of legal works can then be leveraged to hone legal writing and legislators' focus to enact reform more effectively.

Not only have researchers leveraged legal text analytics to inform the focus of reform but also to evaluate the effectiveness of enacted reform (Sun et al., 2019). Adjacent to legal documents and enacted laws, there exist official documents recording the impact of school reform strategies. These documents include written documents denoting visions, performance, reform strategies, and planning and implementation reports. By employing LDA for topical analysis, the researchers were able to discover 15 substantive reform strategies aligned with leadership visions as determined through interviews (Sun et al., 2019). Furthermore, these researchers were able to analyze the relationship between the various strategies, student achievement and absenteeism outcome; thereby modeling both policy positions and effectiveness of reform within the educational field (Sun et al., 2019). As stated, researchers can repeatedly analyze these texts to uncover historically or presently impactful relationships between society and the legal sector.

2.3 Automation

With the increased workload being handled by the judicial system, so has the need to automate the more tedious. The automation process in the legal system comes with its own challenges, and machine learning poses a unique solution to this problem. With the use of Natural Language Process (NLP) and various other methods, automation in the legal system is closer now than ever.

The deep neural networks found in NLP models are more capable of handling textual information than traditional rule-based models. More traditional models struggle to deal with the diversity and complexity of human languages (Chalkidis & Kampas, 2018). With the advancements in NLP, models can now process and build several types of textual documents with more accuracy.

Legal documents still pose significant hurdles that prevent machine learning from widespread use in the legal system. Legal documents must be organized within strict guidelines and rules. While models are highly developed, they do not often meet the expectations of perfection. Every legal document varies from domain and jurisdiction making perfection a moving target (Medvedeva et al., 2022). Some domains struggle with this issue more than others. In the construction industry, there is a lack of available documents to train models. (Hassan et al., 2021). This makes perfection impossible, however, some domains may find faster development depending on the available data.

Even with the limitations of machine learning, the legal domain has seen advancements. The focus of automation has been in ambiguity detection, risk-prone clauses and required structure (Hassan et al., 2021). While machine learning is far from automating every aspect of the legal system, its continued development has been effective. With the reduction of tedious tasks, more skilled legal personnel are freer to use their time focusing on the more complex tasks legal proceedings, while semi-automated processes are overseen by more junior personnel.

2.4 NLP Advancements for the Complexity of Legal Writing

As discussed, analyzing legal text is both time and resource intensive in its requirement of highly trained legal professionals. The nature of this work creates a bottleneck in this field precipitating the need for automation. However, traditional legal analytics require intensive manual resources to train due to the extended nature of legal writing. Traditional NLP processes were designed to address the comparatively shallow nature of day-to-day language.

Chalkidis and Kampas (2018) survey the field's transition from established machine learning, rule-based, and dictionary-based legal analytics models to those employing deep learning. This early use of Deep Neural Networks yields better results in analyzing semantics, indirect contextual relationships, and complexities of language due to the multi-layered nature of these methods. de Oliveira and Nascimento (2022) address the complexities of protracted legal documents using transformers as part of their NLP analysis of Brazilian court documents for clustering as necessitated by the exponential growth in the density of documents accompanying legal proceedings. Laver et al. (2003) introduces untraditional methods rather than attempting to model text by mimicking labor intensive, human processes of extracting meaning, they model words in each document as a collection of data in a novel process. They generate policy dimensions through reference texts then use relative word frequencies of unclassified texts to determine their positions in a prior determined policy dimensions (Laver et al., 2003). This process possesses the additional strength of yielding error estimates and confidence intervals (Laver et al., 2003).

A similarly scalable and emergent deep learning approach is outlined by Chalkidis and Kampas (2018). The researcher's survey is extensive covering three areas of interest which include classifying texts, extracting information, and retrieving information. These areas were studied through the lens of semantic based feature representation practices, an integral technical component of deep learning as applied to NLP. In de Oliveira and Nascimento's 2022 research, the authors present six transformer centered techniques leveraging artificial intelligence (AI) and NLP to group legal documents according to degree of similarity and address convoluted legal semantics. These six techniques were based on three transformers, BERT, GPT-2, and RoBERTa each trained either generally through the Portuguese language or more narrowly through Brazilian judiciary documents. Each technique's classification effectiveness was measured according to the cosine of the distance between group elements and their centroid.

Across their respective works, Ji et al. (2020) and Nay (2017) both employ a multistep process in contributing to this process of progressing traditional methods to cater to the depth of court document language. Ji et al. (2020) addresses the more complex and distant structure by deconstructing the task of evidence information extraction from court record documents into two technical issues. The first being grouping paragraphs, and the second being sequence labeling within that. In accordance with their deconstruction, Ji and researchers introduce a joint extraction process, which first employs a classification procedure that is then followed by an information extraction NLP step. These steps utilize the same encoder but different decoders (Ji et al., 2020). The joint extraction process is found to outperform existing models on legal documents and is a promising technique. Similarly, Nay (2017) finds that a combined, ensemble model is more performant in predicting congressional bill passage. Predictions are rendered by scoring each sentence of a bill to a semantic-laden, high dimensional vector space using an embedding model based on legislative language (Nay, 2017). The authors' models also account for bills changed after their original presentation (Nay, 2017).

Unlike typical texts, documents existing in the legal domain are more likely to be comprised of multi-sentence structures representing diverse types of information. Classic texts typically capture an element or existing relationships in shorter spans of words or phrases. As outlined above, researchers have complimented traditional NLP methods through various techniques including transformers, deep learning, AI, ensemble methods, and novel embedding process to address this challenge.

2.5 Predictive Modeling

Predictive modeling has become a growing target of interest in determining the outcome of cases. At the forefront of this exploration is the use the natural language processing or NLP. Because most case documents provide a large quantity of textual data, NLP is a great fit for handling this problem. However, the use of NLP can come with complications.

NLP provides a robust way to process substantial amounts of textual content without constant user input. The major issue, especially with the use of NLP on legal documents, is the vast variety in how legal documents are structured and worded from jurisdiction to judication. This can make the application of a modeling hard to apply in all different areas. For example, a model used on New Mexico documents trained with documents from a different state. The final model could only reach an F-measure of 60%. This model struggled to handle terms like affirm and reverse because it had not seen them in the training data (Petrova et al., 2020). This can limit the widespread use of a single model when trying to apply it to a variety of different applications. Instead, several different models would have to train for each individual application to ensure consistent results.

Targeting the correct information in such documents plays an influential role in having strong predictions. Processing all the information provided while at first may seem like the best choice, this can add unnecessary noise for the model to handle. Providing a simpler subset of the information can ensure only the most important textual content is utilized. With a strong correlation between the facts of the case and the decisions made by the judges (Aletras et al., 2016), this information can be used as a foundation for establishing a precise model. While other content may be of use, utilizing only the most essential information can provide better overall results.

NLP is not the only option for understanding what leads to predictive outcomes cases. An average of 65% accuracy was achieved in predicting the outcomes of violations of articles 9 of the European Convention on Human Rights by simply using the surname of the judges overseeing the case (Medvedeva et al., 2020). NLP provides a sophisticated option to a complex issue; however simpler options should not be ignored. Understanding who the judges are, lower courts and times can all provide valuable insight into the decision-making process. Even searching for key terms inside of the textual data can be a valuable alternative to the use of NLP (Medvedeva et al., 2020). NLP is a strong tool for processing data; however, it should not be the only factor in understanding the discussion making process of cases.

3 Methods

The workflow provided in Figure 1 provides a high-level overview of the methods used in this research.

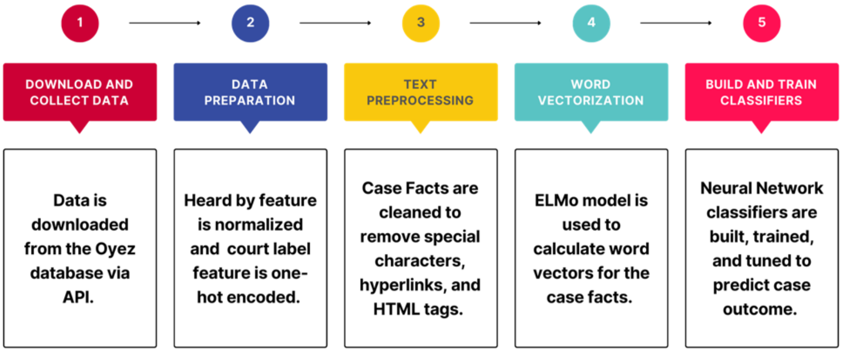
1
DOWNLOAD AND COLLECT DATA
Data is downloaded from the Oyez database via API.
2
DATA PREPARATION
Heard by feature is normalized and court label feature is one-hot encoded.
3
TEXT PREPROCESSING
Case Facts are cleaned to remove special characters, hyperlinks, and HTML tags.
4
WORD VECTORIZATION
ELMo model is used to calculate word vectors for the case facts.
5
BUILD AND TRAIN CLASSIFIERS
Neural Network classifiers are built, trained, and tuned to predict case outcome.

Figure 1. Using NLP to Model Supreme Court Case Outcomes Workflow

3.1 Data

The data used in this research is a collection of U.S. Supreme Court cases provided by the Oyez database. Oyez is an archive of Supreme Court resources put together by Cornell's Legal Information Institute (LII), Justia, and Chicago-Kent College of Law. This database has information on Supreme Court cases dating from 1789 to the present. Data provided includes the date each case was granted, argued, and decided, the parties involved, the lower court where the case originated, the case facts, as well as the outcome of the case. To access this data, an API is used to download the data in JSON format. Filters can be applied to the API call to retrieve specific cases as needed.

This research examined Supreme Court cases from the 2000 – 2019 Supreme Court terms. In some cases, multiple issues were argued, meaning multiple rulings were made in one case. For modeling, only cases in which one issue was argued and ruled upon were included in the dataset. In addition, cases in which the winner was the appellant or appellee were also removed from the dataset, as they represented a very small minority of the dataset. The winner labels were designated as the target variable in this study.

Table 1. Distribution of the Target Variable in the Supreme Court Dataset

Case Winner	Value Counts
Petitioner	1022
Respondent	140
Appellant	15
Appellee	5
Total Observations	1182

In addition to the case facts, the “heard by” and “granted date” features were also considered. The “heard by” feature is a label that describes which justices were members of the Supreme Court at the time when the case was heard and decided. The “granted date” is the date in which the Supreme Court agrees to hear the case. To prepare these two features for modelling, the heard by feature was one-hot encoded and the granted date was normalized.

To create a training and validation set for model training, the data was split into a training and test set using a 75/25 train-test split.

3.2 Text Pre-Processing

First, the case facts underwent pre-processing to prepare the text for NLP. To clean the text, special characters were removed. Some case facts contained hyperlinks that referenced other cases in the database. These links and html formatting were removed using the *beautifulsoup* package. Finally, once the text was deemed clean, it was converted to lower case. These steps are important to ensure proper word tokenization, which is imperative for model performance. Other popular pre-processing techniques include stemming, lemmatization, and stop-word removal. These techniques do not appear to affect model performance significantly. Therefore, they were not included in the pre-processing pipeline. An example of a final, cleaned case facts can be seen below.

Table 2. Example of Case Facts After Text Pre-Processing

Cleaned Case Facts Example
in 1963, henry montgomery was found guilty and received the death penalty for the murder of charles hunt, which montgomery committed less than two weeks after he turned 17. he appealed to the louisiana supreme court, and his conviction was overturned because of community prejudice. at his new trial, montgomery was again convicted, but he was sentenced to life without parole. in 2012, the u.s. supreme court decided miller v. alabama, in which the court held that mandatory sentencing schemes requiring children convicted of homicide to be sentenced to life imprisonment without parole violate the eighth amendment. in light of that decision, montgomery filed a motion in state district court to correct what he argued was now an illegal sentence. the trial court denied montgomery’s motion, and the louisiana supreme court denied montgomery’s application by holding that the decision in miller does not apply retroactively.

3.3 Word Vectorization using ELMo

Embeddings from Language Models (ELMo) is a model for word vectorization created by AllenNLP. ELMo word vectors are generated using a bidirectional language model (biLM). Many words have multiple meanings, sometimes causing difficulty in capturing the true contextual meaning of a word within a word vector. However, due to the architecture of the model, the word vectors calculated by ELMo provide better representation of the meaning of words based on their context in a sentence. Therefore, the word vectors returned are not just a measure of the words used, but their contextual meaning within the text.

The cleaned case facts were sent to the ELMo model for vectorization. Using ELMo's default setting, the case facts were tokenized and split into individual words using whitespaces. To retrieve the word embeddings from the ELMo model, the "elmo" option was used. This output returns a 3D array of word vectors representing the case facts. The first dimension represents the number of observations, the second dimension represents the number of words, and the third dimension represents the word vectors. These word vectors generated from ELMo were then used as features in the final model.

The longest set of case facts in this dataset was 653 words. To make sure the text was the same length for every observation, padding was added to the front of each observation that was less than 653 words. This ensured that the array of ELMo word vectors was the same shape for every observation. This is important to note when building the neural network classifier, as all inputs need to be the same size.

3.4 Classification Model

To build the neural network classifier, an LSTM model was built using the Keras Model API to predict the winning party of a case using the using ELMo word vectors, the normalized granted date, and the one-hot encoded heard by feature. The model was trained using the training set and validated using the test set in each epoch.

To fit this model, the word embeddings were first sent to an LSTM layer with 256 neurons. Then, the word embeddings were sent through a dropout layer to prevent overfitting. Next, the output of the dropout layer was concatenated with the 'granted date' and 'heard by' features. This new input is then sent through a dense layer with 100 nodes and ReLU activation, followed by another dense layer with 10 nodes and ReLU activation, and a final dense layer with 1 node and sigmoid activation used to generate the predicted winner of the case.

The optimizer chosen for this model was the Adam optimizer. Additional hyperparameters were also tuned to improve model performance. The class weights were adjusted and tuned to address the imbalance in the target variable. The learning rate and dropout rate were also tuned. The tuned values for each of these hyperparameters are shown in Table 3.

Table 3. Summary of Tuned Hyperparameters in Neural Network

Hyperparameter	**Tuned Value**
Dropout Rate	.3
Learning Rate	.00001
Class Weights	Petitioner: 1 Respondent: 7

Finally, early stopping was used to optimize training time. To implement early stopping, the validation loss was monitored. If the validation loss did not improve after 10 epochs, the model stopped training and the weights were restored from the epoch with the lowest validation loss. Since this is a binary classification problem, the loss function chosen for this model was Binary Cross Entropy.

The final output of the model is the probability that the Respondent party is the winner of the case. When the output is .5 or greater, the case winner is classified as respondent. When the output is less than .5, the case is classified as petitioner.

4 Results

After the model was tuned using the training data, next it was fitted to the test data to assess its performance on new data. A confusion matrix showing the model's performance on the test set is provided below.

4.1 Neural Network using ELMo Word Vectors

The model correctly classified 202 observations as petitioner and 12 observations as respondent. The model misclassified 28 observations labeled petitioner and 22 observations labeled as respondent.

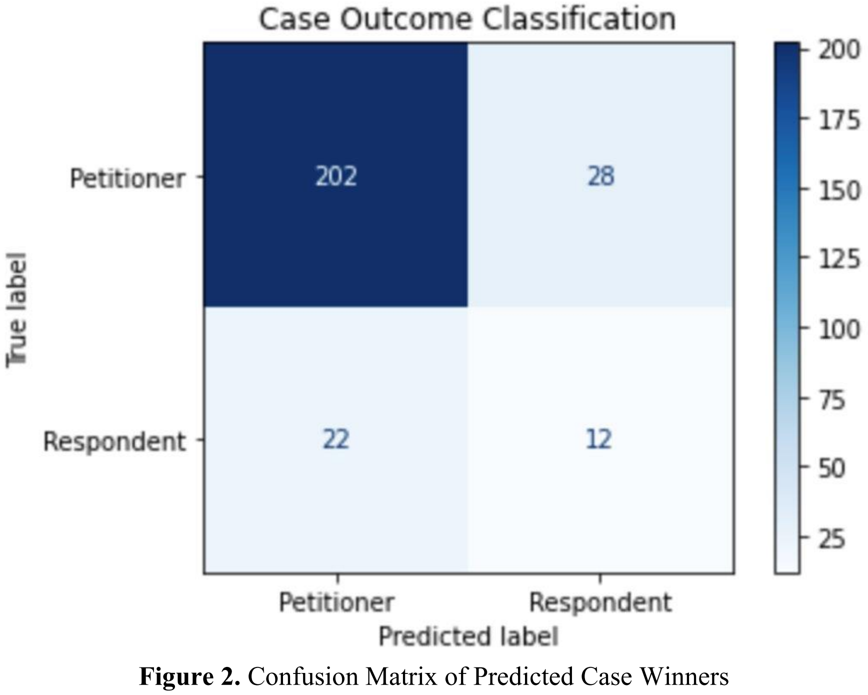

Figure 2. Confusion Matrix of Predicted Case Winners

4.2 Summary of Model Performance

In Table 4, the model's performance metrics on the test set are shown.

. **Table 4.** Summary of Performance Metrics

Model	Accuracy	F1 Score	Precision	Recall	AUC
Neural Network	81.1%	32.4%	30%	35.3%	68%

The formulas for F1 score, precision, and recall are shown below.

$$F_1\,Score = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

$$recall = \frac{true\,positives}{true\,positives + false\,negatives}$$

$$precision = \frac{true\,positives}{true\,positives + false\,positives}$$

5 Discussion

Due to the unregulated nature of text data, it can be difficult to process and use in predictive modelling. However, the use of word vectors has made it possible to summarize meaningful information from a body of text.

Creating an accurate classification model has proven to be challenging due to the imbalanced nature of this dataset. Of the cases used in this study, 88% of them are labeled as petitioner and 12% are labeled as respondent. As it pertains to Supreme Court cases, the petitioner is the party that filed the case, requesting the Supreme Court's review, and the respondent is the party being sued by the petitioner.

To best assess the performance of our model across both classes, the F1 score will be used as this takes both precision and recall into account. While accuracy of this model is 81.1%, F1 score was 32.4%. This is also evident in the confusion matrix provided. This suggests the model is having a difficult time identifying winning cases classified as respondent. However, the model achieved an AUC above .5, suggesting that the model can separate and distinguish between the two target class distributions. Further research exploring additional variables could be useful in modelling Supreme Court case outcomes.

This study only examined Supreme Court cases. It would also be valuable to apply these methods to cases in lower courts, such as State Supreme Courts or Federal District Courts. Another challenge that may be posed in this research is related to the nature of the Supreme Court's case selection and review process. As stated in the introduction of this paper, thousands of cases are sent to the Supreme Court every year but only a fraction are actually heard. It could also be lucrative to explore modelling whether a case sent to the Supreme Court will be heard or not.

Finally, the Supreme Court generates a single decision on a case, but these decisions are decided by the opinions of nine justices. Other opportunities for additional research could include modeling how individual Supreme Court justices would rule on a case and using their individual behaviors to predict a case's outcome.

5.1 Ethics

While the results are promising, there is a long way to go for machine learning to automate processing high level contextual information. These models can provide an important insight into certain behaviors and outcomes, making it a tool for the user. Even though machine learning is not used for making important decisions in this domain, it certainly can have an impact on the people who do. If lawyers knew the predicted results of a case, this would influence how they handle that case. If a client is predicted to lose, less effort might be dedicated to that case, further perpetuating the current issues the legal system has. These tools are certainly powerful, but it is important to understand their limitations.

The Supreme Court's goal is to serve as the final interpreter of the constitution and to rule on the constitutionality of laws. The Supreme Court's interpretation of the constitution has changed over time, as evidenced by cases that have been overturned as time has passed. After a ruling, case decisions are accompanied by support and dissenting opinions which explain why the Justices ruled the way they did. On the contrary, the use of ELMo vectorizations and neural networks to make predictions becomes abstract quickly. These abstract models make it difficult to understand and explain why certain outcomes end up the way they did. This abstraction also opens the door to tough questions around the morality and ethics of use of machine learning in interpreting the United States Constitution and predicting the outcome people's lives.

A primary ethical concern within this dataset is discrimination. In the case facts used, names of those involved in each case are provided. In creating the word vector for each case's facts, the names are included in these calculations. If a model like this is implemented for real-world use, it is imperative that there is a way to protect against accidental discrimination based upon the names or other sensitive information that may be revealed in the case facts.

There are several additional limitations when working with legal documents. Legal documents are highly confidential, making it a challenge to obtain the data initially. While there are publicly available databases, more specific data will require approval.

Even though publicly available court data exists, a wide range of information in that data may or may not be collected. Each court stores data in a wide variety of structures and means of access. While there are certain databases that try to collect this information in an easily accessible way, such as the Oyez database, there is often information missing that is vital to predicting the outcome of cases. Many databases did not provide a winner of the case, meaning those values would need to be generated manually. When there are several thousand observations that need processing, this can quickly become a daunting task.

Reproducing this experiment with this data can be done as the data is publicly available in the Oyez database. However, with the lack of data options, reproducing its structure on a different dataset would be difficult. Each data source comes with unique obstacles and hurdles that would need to be tackled individually.

When utilizing these tools in production, it is important to understand the limitations of the models. Used properly these tools can help improve how the legal system operates when trying to predict case outcomes. With the understanding that these tools are not perfect, they can be maximized to their fullest potential while limiting potential biases.

6 Conclusion

The United States judicial system is an incredibly complex system. The Supreme Court alone processes thousands of cases a year. (The White House, para. 11). The increase of cases year over year demands more of the resources available. This strain often leaves the most vulnerable at risk. Without proper representation many find themselves in dire situations without the means to protect themselves. While machine learning is not a permanent fix to this growing issue, it can find a role in assisting those working in the field.

When predicting the results of cases, the case facts were the primary focus in model building. ELMo was used to process the facts of the cases to produce usable word vectors. These vectors could then be used in building an LTSM model, which resulted in the accuracy of 81.1%. While these metrics seem promising, when looking at the F1 score, the model scored 32.4%. As the result of data imbalance, the model struggled with recall and low F1 scores. While there is room for improvement in classification performance, this research suggests that allocating time and resources to processing legal text data is valuable, as there is essential information found within this unstructured data.

With some promising results when using natural language processing in case predictions, there is more to explore. Adding additional variables may help improve the results of the neural network models, however the accessibility of large balanced data of legal documents is limited. With limited data and a lack of sufficient information, models struggle to achieve high performance in all metrics.

The legal field is filled with difficult and complex challenges when trying to introduce machine learning. This is an additional step, in a massive undertaking to move towards the adoption of machine learning in the legal domain. While it may not replace personnel, it will have an impact on their daily lives.

With high stakes, it is important to remember the limitations of these models. Understanding the interworking's will ensure they are used responsibly and ethically. With improper use, these tools can contribute to the issues found in the legal system. As machine learning expands, there are exciting opportunities to improve processes in the judicial system. Used as a tool, machine learning can be an important piece in reducing workloads and improving decision making.

Acknowledgments. We would like to thank Dr. Robert Slater for his guidance throughout the entirety of this project and Dr. Jacquelyn Cheun-Jensen for her feedback and review of this paper. Finally, we would like to thank Bethel Kumsa. While she was not able to complete this research with the team, she contributed to the literature review in this paper.

References

1. Aletras, N., Tsarapatsanis, D., Preotiuc-Pietro, D., & Lampos, V. (2016). Predicting judicial decisions of the European Court of Human Rights: a Natural Language Processing perspective. *PeerJ.Computer Science, 2*, e93. https://10.7717/peerj-cs.93

2. Bahn, J. *How Does the Supreme Court Work?* Retrieved November 5, 2022, from https://www.americanbar.org/groups/young_lawyers/publications/after-the-bar/essentials/how-does-the-supreme-court-work/

3. Chalkidis, I., & Kampas, D. (2018). Deep learning in law: early adaptation and legal word embeddings trained on large corpora. *Artificial Intelligence and Law, 27*(2), 171-198. https://10.1007/s10506-018-9238-9

4. de Oliveira, R. S., & Nascimento, E. G. S. (2022). Brazilian Court Documents Clustered by Similarity Together Using Natural Language Processing Approaches with Transformers.

5. Finkelstein, M. O., & Levin, B. (2020). Why plea bargains are a bad deal for some. Significance (Oxford, England), 17(1), 20-25. https://10.1111/j.1740-9713.2020.01354.x

6. Hassan, F. u., Le, T., & Lv, X. (2021). Addressing Legal and Contractual Matters in Construction Using Natural Language Processing: A Critical Review. *Journal of Construction Engineering and Management, 147*(9)https://10.1061/(ASCE)CO.1943-7862.0002122

7. Iftikhar, A., Ul Qounain Jaffry, S. W., & Malik, M. K. (2019). *Information Mining From Criminal Judgments of Lahore High Court*. Institute of Electrical and Electronics Engineers (IEEE). https://10.1109/access.2019.2915352

8. Ji, D., Tao, P., Fei, H., & Ren, Y. (2020). An end-to-end joint model for evidence information extraction from court record document. *Information Processing & Management, 57*(6), 102305. https://10.1016/j.ipm.2020.102305

9. Laver, M., Benoit, K., & Garry, J. (2003). Extracting Policy Positions from Political Texts Using Words as Data. *The American Political Science Review; Am Polit Sci Rev, 97*(2), 311-331. https://10.1017/S0003055403000698

10. Mahendra, M., Luo, Y., Mills, H., Schenk, G., Butte, A. J., & Dudley, R. A. (2021). *Impact of Different Approaches to Preparing Notes for Analysis With Natural Language Processing on the Performance of Prediction Models in Intensive Care*. Ovid Technologies (Wolters Kluwer Health). https://10.1097/cce.0000000000000450

11. Martinez, A. R. (2010). Natural language processing. *Wiley Interdisciplinary Reviews. Computational Statistics, 2*(3), 352-357. https://10.1002/wics.76

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

13. Medvedeva, M., Wieling, M., & Vols, M. (2022). Rethinking the field of automatic prediction of court decisions. Artificial *Intelligence and Law,* https://10.1007/s10506-021-09306-3

14. Nay, J. J. (2017). Predicting and understanding law-making with word vectors and an ensemble model. *PloS One; PLoS One, 12*(5), e0176999. https://10.1371/journal.pone.0176999

15. Park, S. O., & Hassairi, N. (2021). What predicts legislative success of early care and education policies?: Applications of machine learning and Natural Language Processing in a cross-state early childhood policy analysis. *PloS One; PLoS One, 16*(2), e0246730. https://10.1371/journal.pone.0246730

16. Petrova, A., Armour, J., & Lukasiewicz, T. (2020). *Extracting Outcomes from Appellate Decisions in US State Courts*. IOS Press. https://10.3233/faia200857

17. Roitblat, H. L., Kershaw, A., & Oot, P. (2010). Document categorization in legal electronic discovery: computer classification vs. manual review. *Journal of the American Society for Information Science and Technology, 61*(1), 70-80. https://10.1002/asi.21233

18. Sun, M., Liu, J., Zhu, J., & LeClair, Z. (2019). Using a Text-as-Data Approach to Understand Reform Processes: A Deep Exploration of School Improvement Strategies. *Educational Evaluation and Policy Analysis, 41*(4), 510-536. https://10.3102/0162373719869318

19. Supreme Court of the United States. *The Supreme Court at Work.* Retrieved October 1, 2022, from https://www.supremecourt.gov/about/courtatwork.aspx

20. The United States Courts. *Supreme Court Procedures.* Retrieved November 22, 2022, from https://www.uscourts.gov/about-federal-courts/educational-resources/about-educational-outreach/activity-resources/supreme-1

21. The White House. *The Judicial Branch.* Retrieved October 1, 2022, from https://www.whitehouse.gov/about-the-white-house/our-government/the-judicial-branch/

22. Vu, S. T., Le Nguyen, M., & Satoh, K. (2021). Abstract meaning representation for legal documents: an empirical research on a human-annotated dataset. *Artificial Intelligence and Law, 30*(2), 221-243. https://10.1007/s10506-021-09292-6

23. Zhang, D., Zhang, H., Wang, L., Cui, J., & Zheng, W. (2022). Recognition of Chinese Legal Elements Based on Transfer Learning and Semantic Relevance. *Wireless Communications and Mobile Computing, 2022*, 1-11. https://10.1155/2022/1783260

24. Zhang, Y. (2021). Exploration of Cross-Modal Text Generation Methods in Smart Justice. *Scientific Programming, 2021*, 1-14. https://10.1155/2021/3225933