ClearCase : Legal Text Classification with Explainable AI

Arun Karthik Sengottuvel sengottu@usc.edu
Ashwinkumar Venkatnarayanan venkatna@usc.edu
Indira Swaminathan iswamina@usc.edu
Bhavya Avuthu avuthu@usc.edu
Mona Teja Kurakula mkurakul@usc.edu
Mohan Sai Ganesh Kanna mkanna@usc.edu

Abstract

Classifying legal documents is essential for organizing case law and helping professionals manage legal information. While legal experts can navigate complex terminology, it's often challenging for non-experts. This study introduces a Transformer-based legal text classification system that not only supports professionals in categorizing cases, but also makes the reasoning behind these classifications clear and helpful to non-experts. Using pre-trained models like LegalBERT, Longformer, or RoBERTa, the system analyzes complex legal texts, accurately extracting citation classes and key phrases. By integrating Explainable AI (XAI) techniques such as SHAP, LIME, or attention visualization, it ensures both precise classifications and transparent, human-readable explanations of critical legal phrases influencing its decisions.

1 Introduction

With the exponential increase in legal texts, automating the classification process offers a scalable solution that saves time and reduces manual effort, allowing legal professionals to focus on case analysis and decision making. Given the high stakes and complex language of legal documents, AI-driven decisions must be both reliable and understandable.

1.1 Why is it useful?

Legal professionals often spend considerable time manually sorting and analyzing vast amounts of legal documents. Automating this process with transformer-based models increases efficiency, reduces human error, and speeds up access to relevant legal texts. What sets this approach apart is its emphasis on explainability, allowing legal professionals to trust and validate AI-generated classifications with confidence. This not only improves the effectiveness of the technology, but also makes it ideal for real-world legal settings, where transparency, trust, and accountability are of utmost importance.

1.2 How is it interesting?

The intersection of deep learning, legal technology, and explainability presents a unique challenge due to the complexity of legal language. Legal documents often contain specialized jargon, citations, and intricate structures that require domain-specific expertise. This project explores how fine-tuning transformers for the legal domain can enhance classification accuracy compared to general-purpose models, which are not domain specific. By integrating XAI techniques, we gain deeper insights into the decision-making process of these complex models.

2 Related Work

This section reviews prior works that have contributed to the fields of legal text classification, transformer-based models, and explainable AI (XAI), providing the foundation for the proposed approach.

Transformer-based architectures have revolutionized text classification tasks by leveraging self-attention mechanisms to capture contextual relationships. Chalkidis et al. (2020) introduced LEGAL-BERT, a domain-specific variant of BERT pre-trained on legal corpora, demonstrating its superior performance in multi-label classification and named entity recognition tasks compared to general-purpose models. Similarly, Limsopatham (2021) explored how BERT can be effectively adapted for legal NLP tasks using datasets such as the ECHR Violation and Overruling Task datasets. Their findings revealed that truncating long documents negatively impacts performance, underscoring the need for specialized architectures.

To address the limitations of standard transformers in processing long sequences, Beltagy et al. (2020) proposed Longformer, which employs a linear attention mechanism capable of handling sequences up to thousands of tokens. Building on this, Mamakas et al. (2022) modified Longformer and LegalBERT to accommodate sequences as long as 8,192 tokens, achieving state-of-the-art results in legal document classification. While these advancements focus on improving accuracy and scalability for lengthy texts, they often overlook interpretability and explainability that are critical requirements for real-world legal applications.

Explainability in AI systems is essential for fostering trust and transparency, particularly in sensitive domains like law. Several XAI techniques have emerged to address this need. Lundberg and Lee (2017) introduced SHAP (SHapley Additive exPlanations), a game-theoretic approach to quantifying feature contributions to predictions, ensuring local accuracy and consistency. Ribeiro et al. (2016) proposed LIME (Local Interpretable Model-Agnostic Explanations), which approximates black-box models with interpretable surrogates to explain individual predictions. Attention visualization tools, such as BertViz by Vig (2019), provide insights into token-level attention weights within transformer models. However, Jain and Wallace (2019) cautioned that attention weights do not always correlate with causal influence, underscoring the need for more robust interpretability methods.

Despite their success in domains like healthcare and finance, XAI techniques remain underutilized in legal NLP. Legal professionals require explanations aligned with legal reasoning, such as identifying key citations or phrases driving predictions. Current general-purpose XAI methods often fail to capture the nuanced structure of legal argumentation, highlighting the need for domain-specific adaptations.

3 Data Preparation

3.1 Dataset Description

The proposed dataset[1] by Mohankumar (2023) consists of 24,985 records, where each record is assigned a unique `case_id` in the format `CaseX`, with `X` indicating the case number. This `case_id` serves as a unique identifier for each case and does not contain duplicate or missing values. The dataset includes the following key columns:

[1]https://www.kaggle.com/datasets/amohankumar/legal-text-classification-dataset

- case_id: A unique identifier for each case.
- case_outcome: The verdict or outcome of the case, which serves as the class label for classification tasks.
- case_title: The headline or name of the case.
- case_text: The textual content of the case, which provides detailed information regarding the case.

3.2 Exploratory Data Analysis

Upon initial inspection, it is found that the case_id, case_title, and case_outcome columns have no missing values. However, the case_text i.e., our target label column contains 176 missing values. As a result, we decided to drop these 176 records, leaving us with 24,809 complete records for further analysis and model training.

The distribution of the case_outcome column is highly imbalanced, with some categories more prevalent than others.The distribution is summarized in Table 1.

Case Outcome	Count (%)	
Affirmed	106	(0.43%)
Applied	2438	(9.83%)
Approved	108	(0.44%)
Cited	**12110**	**(48.81%)**
Considered	1699	(6.85%)
Discussed	1018	(4.10%)
Distinguished	603	(2.43%)
Followed	2252	(9.08%)
Referred To	4363	(17.59%)
Related	112	(0.45%)

Table 1: Case Outcome Distribution in the Dataset

From the table, it is clear that the cited outcome dominates the dataset, comprising 48.8% of the total records. The referred to outcome follows, making up approximately 17.6% of the data. Several other outcomes, such as applied, considered, and followed, are less frequent but still significant. The dataset is imbalanced, which will need to be carefully handled during model training through techniques like stratified sampling, resampling, or class-weight adjustments.

3.3 Data Sampling

To start, the target variable, case_outcome, was encoded into numeric labels using the LabelEncoder. This encoding process transformed the case_outcome into numerical values (0-9 as there are 10 classes), allowing for seamless integration.

Following this, a stratified sampling approach was employed to ensure balanced splits across the target variable. The dataset was divided into three subsets: Train (80%), Validation (10%) and Test (10%) sets, preserving the distribution of case_outcome labels within each subset. This stratified split ensures that all splits maintain the class proportions, addressing potential class imbalances in the dataset. These final splits of training, validation, and testing were saved in separate CSV files, ensuring the dataset is properly prepared for the next steps in model development. This marks an end to the data preparation phase, ensuring that the dataset is optimized for training or fine-tuning transformer models with well-distributed outcomes.

4 Methodology

The workflow includes a systematic approach to legal text classification, leveraging transformer-based models and addressing challenges posed

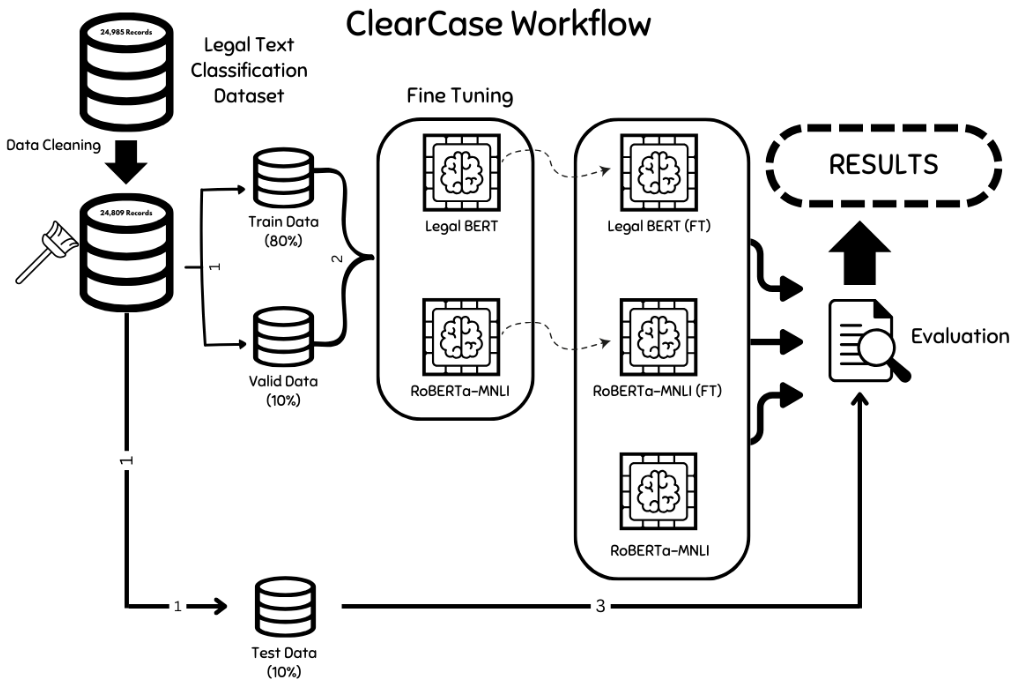

Figure 1: Curernt Workflow.

by lengthy legal documents. The methodology consists of two key phases: zero-shot classification and supervised fine-tuning, with strategies to handle token length limitations effectively.

4.1 Zero-Shot Classification with RoBERTa

To establish a baseline, we utilized the `roberta-large-mnli` model from Hugging Face's Transformers library for zero-shot classification. This model, trained on the Multi-Genre Natural Language Inference (MNLI) task, supports textual entailment-based inference. Each legal case's text (`case_text`) was treated as the "premise", while candidate labels such as `affirmed`, `cited`, and `applied` were hypothesized in the form: "`This case was [label].`" The model evaluated the likelihood of each hypothesis being entailed by the premise and selected the label with the highest entailment probability as its prediction.

This phase provided insights into how general-purpose models perform on domain-specific tasks like legal text classification. While it offered a quick baseline, it underscored the need for supervised fine-tuning tailored to legal content for improved accuracy.

4.2 Supervised Training on Transformer Models

To enhance classification performance, we fine-tuned two transformer-based models: Legal-BERT[2] and RoBERTa[3]. Each model was trained to classify legal cases into one of ten predefined class labels (`case_outcomes`) provided in the dataset.

[2]https://huggingface.co/nlpaueb/legal-bert-base-uncased

[3]https://huggingface.co/roberta-large-mnli

Training Pipeline (Fine-Tuning):

- **Tokenization:** Inputs were tokenized using model-specific tokenizers with truncation strategies applied to fit within token limits:
  - RoBERTa and Legal-BERT: Maximum sequence length of 512 tokens.
- **Loss Function:** CrossEntropyLoss was employed for multi-class classification tasks.

Training was conducted over six epochs with a batch size of 32. Validation accuracy was calculated after each epoch, and the best-performing model was saved for evaluation on the test set.

4.3 Handling Token Length Limitations

Legal documents often exceed standard transformer token limits (512 tokens), requiring truncation strategies to manage lengthy inputs effectively. We explored two approaches:

End Truncation (Head-Only): This approach retained only the first 512 tokens of each document, discarding the rest. While this strategy assumes that introductory context is sufficient for classification, it risks losing critical information located near the middle or end of documents.

Head–Tail Truncation (Start+End): To mitigate the shortcomings of end truncation, this strategy preserved both the first 256 and last 256 tokens of each document. This balanced retention allowed the model to capture introductory context as well as concluding insights both crucial for legal case analysis.

5 Results

In this study, we fine-tuned two transformer-based models i.e, Legal-BERT and RoBERTa to classify legal case texts into one of the ten predefined case outcomes. Both models were trained under identical configurations, differing only in their underlying architectures and pretraining domains.

Model	**Accuracy (%)**
RoBERTa (Base)	27.12%
Legal-BERT	51.58%
RoBERTa	53.68%
Legal-BERT (Head-Tail)	54.79%
RoBERTa (Head-Tail)	**54.83%**

Table 2: Comparison of Model Performances

RoBERTa demonstrated slightly better performance compared to Legal-BERT across truncation strategies, achieving an accuracy of 54.83% using the head-tail truncation method, compared to Legal-BERT's 54.79%. While Legal-BERT benefits from pretraining on domain-specific legal corpora, RoBERTa's larger architecture and broader pretraining data likely contributed to its superior general language understanding. This highlights how high-capacity models can generalize effectively to specialized tasks.

The results also underscore the importance of input truncation strategies. The head-tail truncation method, which preserves both the first and last portions of a document, provided better performance for both models compared to the naive end truncation. However, it should be noted that even with improved truncation strategies, valuable contextual information may still be lost in the middle of the document. This limitation suggests that future work should explore methods to capture richer document context.

6 Challenges

Legal text classification presents unique challenges that impact both model training and performance. This section highlights two primary challenges encountered during the study.

6.1 Label Ambiguity in Repeated Case Texts

A minor challenge arises from the presence of 1,024 records where the `case_text` is repeated across different entries but assigned conflicting `case_outcome` labels. This label ambiguity introduces noise into the training process, as identical inputs correspond to different outputs, potentially confusing the model and affecting its ability to learn consistent patterns.

6.2 Token Length Constraints in Legal Documents

Approximately 13.5% of the dataset (3,347 records) contains `case_text` lengths between 512 and 1,024 tokens, requiring truncation or splitting during pre processing as standard transformer-based models have a 512 token limit. While truncation strategies like head-tail truncation preserve semantic signals from both ends of a document, they fail to capture critical information from the middle sections, which can degrade classification performance for legal texts where reasoning and citations span multiple sections.

7 Future Scope

Future work will focus on addressing challenges such as label ambiguity in repeated `case_text` entries and token length constraints in legal documents. Leverage advanced architectures like Longformer, capable of processing up to 4,096 tokens, offers promising solutions for retaining full document context without truncation.

Additionally, integrating Explainable AI (XAI) techniques into the proposed system takes at most importance in future efforts.

References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Ion Androutsopoulos, and Nikolaos Aletras. 2020. LEGAL-BERT: The muppets straight out of law school. *arXiv preprint arXiv:2010.02559*.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Nut Limsopatham. 2021. Effectively leveraging BERT for legal document classification. In *Proceedings of the Natural Legal Language Processing Workshop (NLLP)*.

Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Dimitris Mamakas, Ilias Chalkidis, Ion Androutsopoulos, and Nikolaos Aletras. 2022. Processing long legal documents with pre-trained transformers: Modding legalbert and longformer. In *Proceedings of the Natural Legal Language Processing Workshop (NLLP)*.

A. Mohankumar. 2023. Legal text classification dataset. Kaggle.

Marco T. Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Jesse Vig. 2019. BertViz: Visualize attention in NLP models. GitHub Repository.