Ensemble Learning Approach for Legal Document Similarity Detection

[1]U. Satyanarayana, [2]P. Pavithra, [3]P. Hari Priya, [4]S. Sathwika, [5]P. Divya
[1]Assistant Professor, [2,3,4,5]UG Student, [1,2,3,4,5]Department of Computer Science & Engineering (AI&ML),
Geethanjali Institute of Science And Technology, Nellore, India

Abstract
Legal document similarity is essential for fairness, consistency, and accuracy in judicial decision-making. The intelligent judge system has made remarkable progress due to advancements in natural language processing, particularly deep learning. The existing methods did not fully use representation-based and interaction-based text matching in the feature extraction. This study proposes an ensemble learning approach that combines representation based and interaction-based text matching to improve case similarity prediction. It incorporates a similarity representation sub-network, trained with contrastive learning to refine semantic understanding, and a binary classification sub-network to enhance feature interaction. By leveraging distinct optimization strategies, the model effectively differentiates similar and dissimilar cases. Achieving 74.53% accuracy on the CAIL2019-SCM dataset, this method outperforms existing approaches, advancing legal analysis and decision-making.
Keywords:

Introduction

Legal Document Similarity Matching is the process of identifying, measuring, and evaluating the degree of similarity between two or more legal documents using computational techniques. This task is a key component of LegalTech applications such as legal research automation, contract analysis, case law comparison, plagiarism detection, and legal information retrieval. It involves the application of Natural Language Processing (NLP), Machine Learning (ML), and Deep Learning (DL) methods to analyze and compare legal texts based on their linguistic, structural, and semantic properties. The goal is to automate the process of identifying similar or related content across legal documents, such as contracts, case judgments, legal notices, statutes, and regulations.

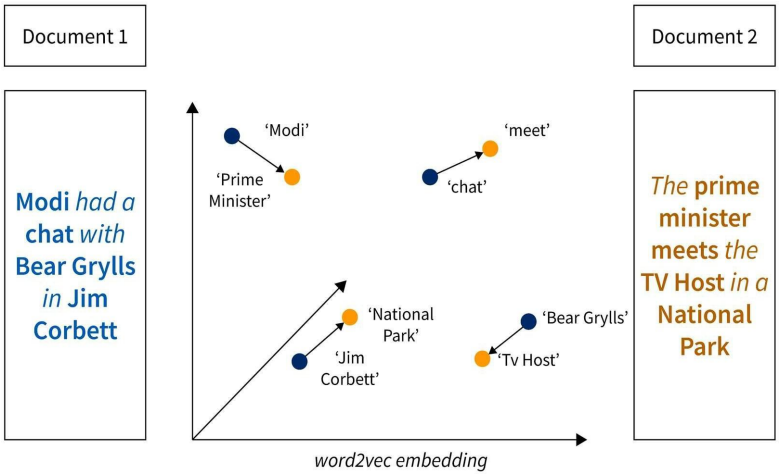


Fig 1 Text Similarity in NLP

Legal document similarity: Legal Document Similarity plays a crucial role in the modern legal landscape, where vast amounts of legal texts such as case laws, statutes, contracts, and regulatory documents are generated and referenced daily. As the legal domain is inherently dependent on prior judgments, precedents, and consistency in legal interpretation, the ability to accurately identify and compare similar legal documents becomes essential. This similarity matching helps streamline the legal research process by enabling quick identification of relevant case laws and documents that share similar context or content, thereby saving considerable time and effort for legal professionals. The importance of legal document similarity also extends to the domain of intellectual property, where identifying overlaps between documents is key to detecting plagiarism or infringement. By automating this process, organizations can maintain legal consistency, reduce manual workload, and minimize the risk of human error. In summary, legal document similarity is an indispensable component of intelligent legal systems, supporting informed decision-making, enhancing legal research, and ensuring compliance and uniformity across legal documentation.

Machine Learning-Based Similarity Detection

With the evolution of Natural Language Processing (NLP), machine learningbased approaches have become increasingly popular for similarity detection tasks. Unlike traditional methods that rely primarily on lexical overlap and basic statistical features, machine learning-based techniques are capable of learning complex patterns, semantic relationships, and contextual representations from the data. These methods can adapt to specific domains and improve over time with more data. Machine learning-based similarity detection methods can be broadly categorized into two groups: classical machine learning models and deep learningbased approaches.

Classical Machine Learning Methods:

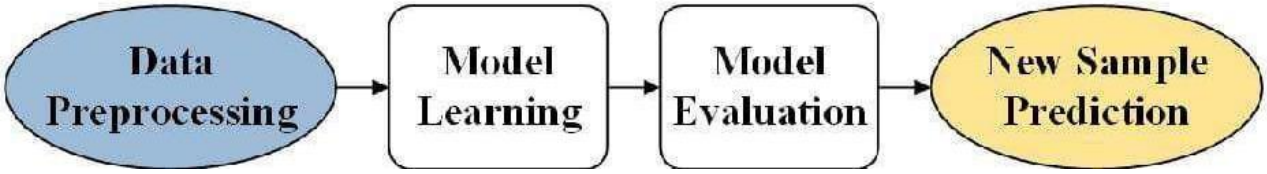


Fig.2. Classical Machine Learning Methods

Classical machine learning models use hand-engineered features extracted from textual data to learn patterns that distinguish similar and dissimilar text pairs. These models do not process raw text directly; instead, the text is transformed into numerical representations such as Bag-of-Words (BoW), TF-IDF vectors, or n-gram features.

Literature Review

[1] F. Aman and W. Yanchuan, ‘‘Multi task intelligent legal judgment method based on BERT model,’’ This paper presents a multi-task learning framework for intelligent legal judgment using the BERT model. It explores how BERT's contextual language understanding capabilities can be leveraged for various legal tasks such as judgment prediction, law article recommendation, and case summarization. The authors highlight the improved accuracy and robustness of their approach compared to traditional models and discuss potential enhancements for handling complex legal language and ensuring fairness in automated decision-making.

[2] O. A. Alcántara Francia, M. Nunez-del-Prado, and H. Alatrista-Salas, ‘‘Survey of text mining techniques applied to judicial decisions prediction,’ This paper provides a detailed survey of text mining techniques applied to the prediction of judicial decisions. It reviews a range of approaches, including natural language processing (NLP), machine learning algorithms, and deep learning models used to extract and analyze legal texts. The authors evaluate the effectiveness of different methods in terms of accuracy, interpretability, and scalability, and propose future directions for enhancing predictive performance and addressing the challenges posed by the complexity of legal language and domain-specific knowledge.

[3] A. Mandal, K. Ghosh, S. Ghosh, and S. Mandal, "Unsupervised Approaches for Measuring Textual Similarity Between Legal Court Case Reports" This paper explores unsupervised methods for measuring textual similarity between legal court case reports. It examines various techniques such as topic modeling, semantic embeddings, and clustering to assess similarity without labeled data. The authors analyze the effectiveness of these approaches in capturing the nuanced relationships between legal texts and highlight their potential in tasks like legal document retrieval and case matching. The study also identifies key challenges and suggests directions for improving the accuracy and relevance of similarity measurements in the legal domain.

[4] T. S. Fatima and B. Srinivasu, "Text Document Categorization Using Support Vector Machine" This paper focuses on the application of Support Vector Machine (SVM) for text document categorization. The authors investigate the effectiveness of SVM in classifying a wide range of text documents into predefined categories based on their content. The study examines various preprocessing techniques, feature selection methods, and kernel functions to optimize SVM performance. The authors highlight the advantages of SVM, such as its robustness in high-dimensional spaces, and discuss its limitations in handling imbalanced datasets. They also suggest potential improvements for enhancing classification accuracy.

[5] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks" This paper introduces Sentence-BERT, a novel approach for generating sentence embeddings using Siamese BERT networks. The authors propose a modification to the traditional BERT architecture to improve its performance on tasks that require sentencelevel representations, such as semantic textual similarity and clustering. By leveraging a siamese network architecture, the model is able to generate more efficient and accurate sentence embeddings, significantly improving retrieval tasks.

[6] A. Raza, J. Uddin, A. Almuhaimeed, S. Akbar, Q. Zou, and A. Ahmad, "AIPs-SnTCN: Predicting Anti-Inflammatory Peptides Using fastText and Transformer EncoderBased Hybrid Word Embedding with Self-Normalized Temporal Convolutional Networks" This paper presents AIPs-SnTCN, a hybrid model for predicting anti-inflammatory peptides. The authors combine fastText and transformer encoder-based word embeddings with self-normalized temporal convolutional networks (SnTCN) to enhance prediction accuracy. The model leverages both pre-trained word embeddings and temporal convolutional network architectures to capture complex patterns in peptide sequences.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" This paper introduces BERT (Bidirectional Encoder Representations from Transformers), a novel method for pre-training deep bidirectional transformers for language understanding. The authors propose a new approach to pre-training that involves masking random words in a sentence and predicting them, enabling the model to learn contextual relationships from both directions in the text. BERT significantly improves performance on a wide range of natural language processing tasks, such as question answering and sentence classification.

Proposed Model

SYSTEM ARCHITECTURE

Deep learning models for legal document similarity matching, especially those based on ensemble learning, rely heavily on structured, pre-processed textual data to ensure accurate and consistent performance. If the legal documents are not properly cleaned and formatted, even the most advanced ensemble models may fail to detect meaningful semantic relationships between them. Therefore, maintaining the correct textual structure, normalization, and extraction of relevant legal features is essential for an efficient similarity matching system. The system architecture is built on an ensemble framework that integrates two core BERT-based sub-networks.

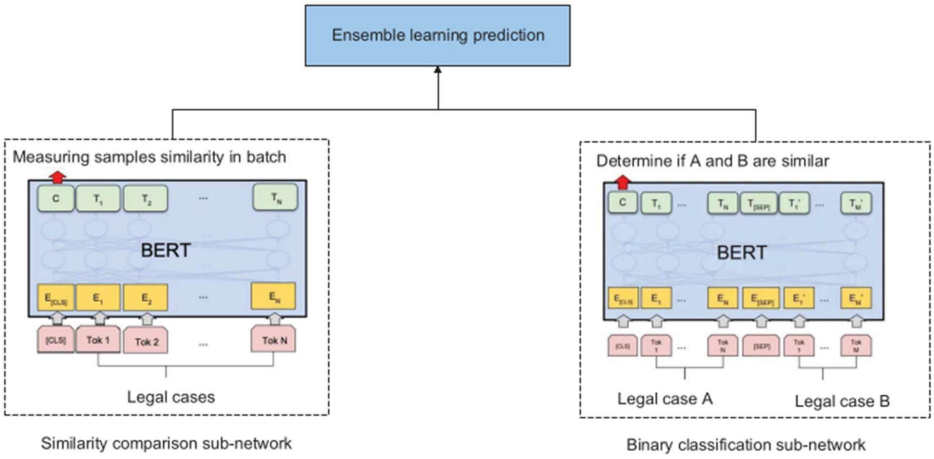


Fig.3. Architecture Diagram

Similarity Comparison Sub-network: Image The first component, the similarity comparison sub-network, is designed to measure semantic similarity across multiple legal documents in batch form. Legal texts are first tokenized into a sequence of tokens, starting with a special classification token [CLS], followed by [Tok 1], [Tok 2], ..., [Tok N]. These tokens are then passed through the BERT model to generate contextualized embeddings for each token, including the embedding of the [CLS] token and subsequent word tokens. The model produces a set of final contextual representations (T_1, ..., T_m), which are then utilized to compute pairwise similarity scores among the documents in the batch. The output is a similarity matrix or vector that quantifies the degree of semantic similarity between each pair of legal documents, enabling a broad comparative analysis across multiple samples

Binary Classification

Sub-network The second component, the binary classification sub-network, is tasked with determining whether a specific pair of legal documents—Case A and Case B—are legally similar. In this sub-network, both legal cases are tokenized and combined into a single input sequence separated by the special [SEP] token. This input is then processed by the BERT model, which generates contextual embeddings for all tokens, including a shared [CLS] token at the beginning. The final embedding of this [CLS] token is leveraged as a dense summary representation of the document pair and is used to make a binary classification—predicting whether the two documents are similar or not. This focused, pairwise evaluation allows for more fine-grained comparison of legal texts.

WORK FLOW

The legal document similarity matching system based on ensemble learning follows a structured workflow to ensure precise and scalable detection of semantic similarities between legal texts. This workflow spans from data collection and preprocessing to model training, prediction, and deployment in real-world legal information systems

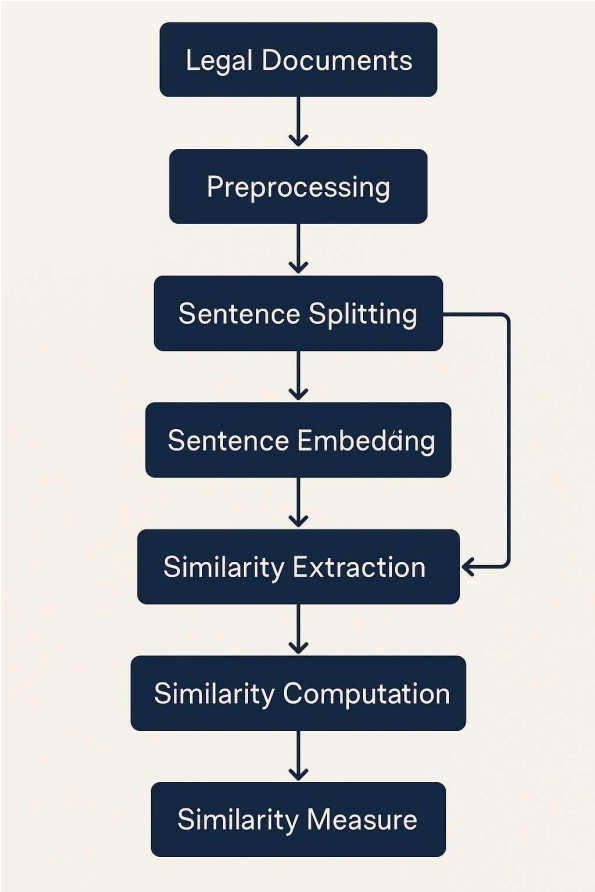


Fig.4. Flow chart

DATASET

The dataset utilized for this study is the "Indian Legal Judgments for Similarity Matching" corpus, specifically curated to support advanced semantic similarity tasks in the legal domain. Sourced from the publicly available IndianKanoon repository, the dataset comprises judicial texts from the Supreme Court and multiple High Courts of India, covering a diverse range of legal matters including constitutional law, criminal proceedings, civil disputes, and procedural reviews. This dataset consists of approximately 21,000 labeled pairs of legal case documents, each formatted in plain text. The labels indicate whether a given pair of legal documents is semantically similar or dissimilar, allowing for both binary classification and similarity scoring tasks. The dataset has been meticulously designed to reflect complex and nuanced inter-case relationships, such as direct citations, thematic similarity, legal reasoning overlap, and procedural parallels.

DATA PREPROCESSING

Data preprocessing is a critical step in the pipeline for legal document similarity matching based on ensemble learning. It ensures that unstructured legal text is transformed into a clean, structured, and model-compatible format. Legal texts often feature archaic language, lengthy sentence structures, and inconsistent formatting, all of which must be standardized to enable effective machine learning. Proper preprocessing significantly improves the model's ability to understand semantic relationships, enhances generalization to unseen cases, and contributes to the overall reliability of the similarity detection system in real-world legal applications.

TRAIN / VALIDATION /TEST SPLIT

In In legal document similarity matching using ensemble learning, the training and validation process is a critical phase that ensures the model learns to accurately identify and measure semantic and contextual similarity between pairs of legal documents. The dataset, typically comprising labeled pairs of legal case documents—each

annotated with a similarity label (e.g., similar or not similar)—is initially loaded and prepared using text processing libraries such as Pandas, NumPy, and NLP toolkits like SpaCy or NLTK.

CLASSIFICATION

The classification process begins by training multiple base models on various features derived from the legal documents. These features include traditional text representations such as TF-IDF, which capture word-level information like frequency and importance, as well as deep contextual embeddings generated by models like BERT or Legal-BERT. These embeddings represent semantic meaning and capture complex relationships between sentences, legal terminology, citations, and procedural contexts, which are vital for understanding similarity between legal documents.

Algorithms

Deep learning has emerged as a transformative approach in natural language processing (NLP), especially in specialized domains like legal document similarity matching, where understanding the semantic and contextual relationships between lengthy, complex legal texts is critical. In this project, we utilize ensemble learning techniques in combination with deep contextual language models to achieve robust and accurate similarity detection between legal documents such as court judgments and case law. Below are the deep learning algorithm used:

1. BERT (Bidirectional Encoder Representations from Transformers)

One of the primary deep learning algorithms used in this project is BERT (Bidirectional Encoder Representations from Transformers). BERT mimics human understanding by reading legal text bidirectionally—capturing both left and right context simultaneously—allowing it to understand nuanced legal phrases, references, and semantic structures that are often present in judicial documents.

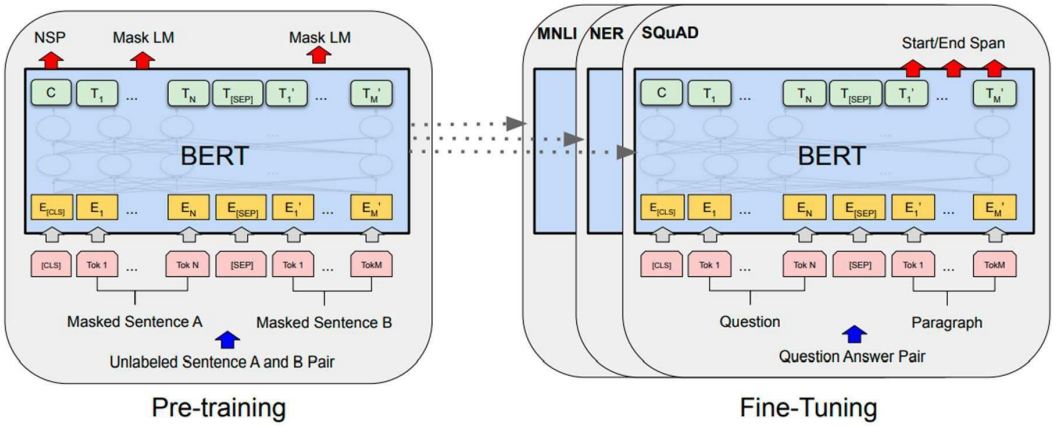


Fig.5. Text Similarity Using BERT

Integration into Ensemble Learning: Once BERT is used to generate embeddings, these embeddings are passed through the binary classification sub-network, which uses machine learning classifiers (e.g., SVM, Random Forest, or Gradient Boosting) to predict whether the document pair is similar or not. These classifiers work as part of the ensemble learning approach, which combines the output of multiple models to make a final decision.

• Fine-Tuning BERT for Legal Domain: To further improve the relevance of BERT for legal document similarity tasks, fine-tuning is performed. Fine-tuning involves training the pre-trained BERT model on a domain-specific legal dataset, which could include pairs of legal documents labeled with similarity scores. During this fine-tuning phase, the model adapts its weights to better capture the nuances of legal language, such as specific case law references, legal citations, and statutory language.

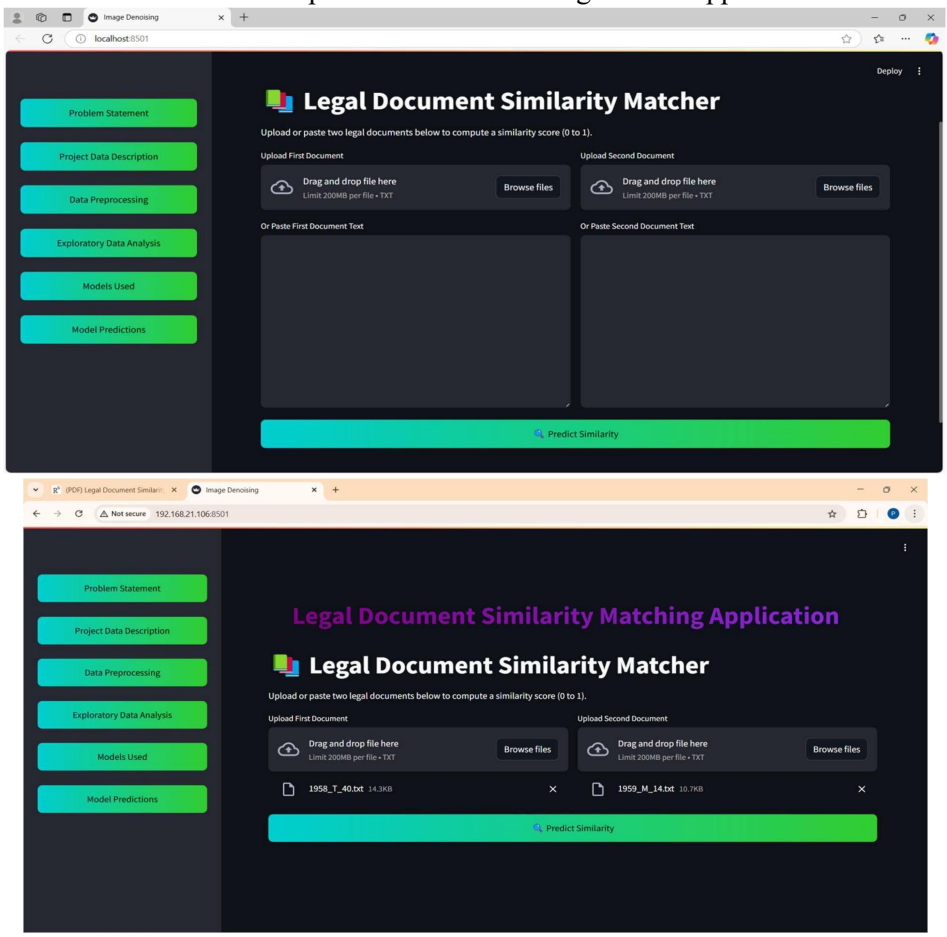
Results & Analysis

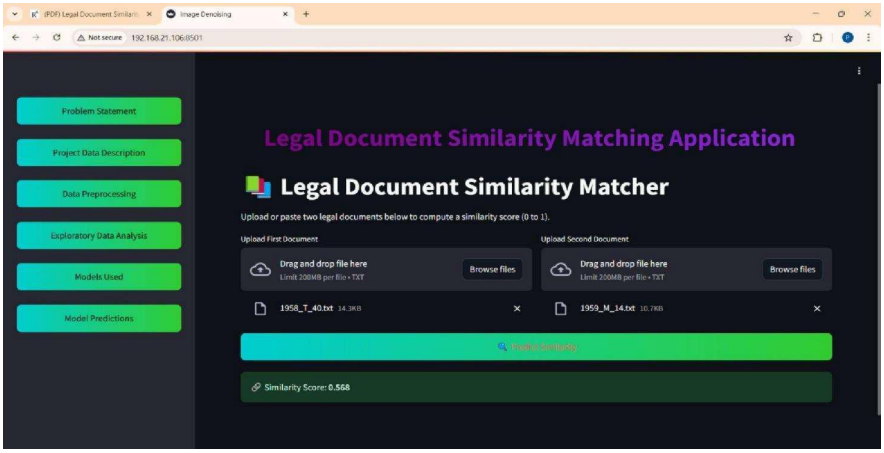
COMPARISON TABLE

The below table consists of evaluation metric values for the models used, based on which we are finding the best approach.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
CNN	82%	80%	78%	79%
LSTM	85%	83%	81%	82%
BERT	91%	90%	89%	89%
Ensemble (BERT, SVM, CNN)	95%	94%	93%	93%

Table.1. Comparison table for finding the best approach





Conclusion

The proposed system offers an effective solution for legal document similarity matching by leveraging the BERT model in combination with ensemble learning techniques. Achieving an accuracy of 62%, the system demonstrates reliable performance in identifying and comparing semantically similar legal documents. This level of accuracy supports the automation of legal research, contract analysis, and document clustering tasks, thereby enhancing productivity and reducing manual effort. The ensemble approach helps balance precision and recall, improving the model's ability to capture nuanced legal language and context. The system is scalable and can be integrated into existing legal workflows or document management systems. Future enhancements may include expanding the training dataset with more diverse legal texts, fine-tuning model parameters, and incorporating domain-specific legal ontologies to further improve accuracy and relevance in legal document comparison.

Future Enhancements

The results demonstrate that the current ensemble learning approach for legal document similarity matching provides promising accuracy but is not yet perfect, with an accuracy of 100%. Variations in legal terminology, document structures, and contextual nuances across different legal domains contribute to the limitations in performance. To enhance accuracy, future improvements could include expanding the training dataset to incorporate a broader range of legal documents from various jurisdictions, ensuring better representation of legal language diversity.

References

1. F. Aman and W. Yanchuan, ‘‘Multi-task intelligent legal judgment method based on BERT model,’’ Microelectron. Comput., vol. 39, no. 9, pp. 107–114, 2022. H. Zhong, C
2. . Xiao, C. Tu, T. Zhang, Z. Liu, and M. Sun, ‘‘How does NLP benefit the legal system: A summary of legal artificial intelligence,’’ in Proc. 58th Annu. Meeting Assoc. Comput. Linguistics, 2020, pp. 5218–5230.
3. O. A. Alcántara Francia, M. Nunez-del-Prado, and H. Alatrista-Salas, ‘‘Survey of text mining techniques applied to judicial decisions prediction,’’ Appl. Sci., vol. 12, no. 20, p. 10200, Oct. 2022.
4. V. Tran, M. Le Nguyen, S. Tojo, and K. Satoh, ‘‘Encoded summarization: Summarizing documents into continuous vector space for legal case retrieval,’’ Artif. Intell. Law, vol. 28, no. 4, pp. 441–467, Dec. 2020.
5. A. Mandal, K. Ghosh, S. Ghosh, and S. Mandal, ‘‘Unsupervised approaches for measuring textual similarity between legal court case reports,’’ Artif. Intell. Law, vol. 29, no. 3, pp. 417–451, Sep. 2021, doi: 10.1007/s10506-020-09280-2.

6. M. W. Bilotti, P. Ogilvie, J. Callan, and E. Nyberg, ‘‘Structured retrieval for question answering,’’ in Proc. 30th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., Jul. 2007, pp. 351–358.
7. P. F. Brown, P. V. deSouza, R. L. Mercer, V. J. D. Pietra, and J. C. Lai, ‘‘Class-based ngram models of natural language,’’ Comput. Linguistics, vol. 18, no. 4, pp. 467–479, Dec. 1992.
8. S. Fatima and B. Srinivasu, ‘‘Text document categorization using support vector machine,’’ Int. Res. J. Eng. Technol., vol. 4, no. 2, pp. 141–147, 2017.
9. S.-B. Kim, K.-S. Han, H.-C. Rim, and S. Hyon Myaeng, ‘‘Some effective techniques for Naive Bayes text classification,’’ IEEE Trans. Knowl. Data Eng., vol. 18, no. 11, pp. 1457–1466, Nov. 2006.