LexSetuX: AI-Driven Legal Case Classification and Lawyer Recommendation System

Deepali Jawale[1], Harshal Ingale[2], Dhruva Gramopadhye[3], Om Babar[4], Atharv Bhise[5]
[1,2,3,4,5]*Department of Computer Engineering*
Pimpri Chinchwad College of Engineering (PCCoE), Pune, India

***Abstract*—Access to accurate legal information and appropriate legal assistance remains a significant challenge for citizens due to the complexity of legal language, lack of awareness of rights and difficulty in identifying suitable legal professionals. This paper presents LexSetuX, an AI-Powered Legal Case Classification and Lawyer Recommendation System designed to bridge this gap using advanced Natural Language Processing and deep learning techniques. The proposed system employs LegalBERT, a domain-specific transformer model, to automatically analyze and classify legal case descriptions into relevant legal domains such as criminal, civil, family, property and corporate law. In addition to classification, the system maps cases to applicable constitutional rights, legal provisions and judicial precedents to enhance legal understanding. Furthermore, LexSetuX integrates an intelligent lawyer recommendation engine that suggests suitable legal professionals based on case type, specialization, experience and semantic relevance. The platform provides role-based outputs, offering simplified legal explanations and guidance for citizens, while delivering detailed legal insights, case precedents and risk assessments for lawyers. Experimental evaluation demonstrates that the proposed system achieves 92% classification accuracy, outperforming traditional machine learning approaches such as TF-IDF with SVM. The system also exhibits efficient response time and high recommendation relevance, validating its practical applicability in real-world legal assistance. LexSetuX contributes toward improving accessibility, efficiency and intelligence in legal support systems through AI-driven automation and decision assistance.**

***Index Terms*—Legal AI, Case Classification, LegalBERT, Lawyer Recommendation, NLP, Precedent Retrieval, Indian Law**

I. INTRODUCTION

The legal domain is characterized by complex terminology, vast documentation and procedural intricacies, which often make it difficult for ordinary citizens to understand their legal rights and seek appropriate legal assistance. Many individuals struggle to interpret legal language, identify relevant laws and connect with suitable legal professionals. As a result, there exists a significant gap between legal knowledge and accessible legal support, particularly for individuals without formal legal education. The advancement of Artificial Intelligence (AI) and Natural Language Processing (NLP) presents an opportunity to bridge this gap by automating legal analysis and improving access to legal information.
Recent developments in transformer-based language models, especially domain-specific models such as LegalBERT, have demonstrated strong capabilities in understanding legal text, semantic relationships and contextual meaning. These models enable automated legal document classification, case analysis and information retrieval with higher accuracy compared to traditional machine learning approaches. Leveraging such advancements, intelligent legal assistance systems can be developed to provide automated case understanding and decision support.

This research introduces LexSetuX, an AI-powered Legal Case Classification and Lawyer Recommendation System designed to simplify legal understanding and improve access to legal assistance. The proposed system utilizes LegalBERT to automatically classify legal case descriptions into appropriate legal domains such as criminal, civil, family, property and corporate law. In addition, the

system maps classified cases to relevant constitutional rights, statutory provisions and judicial precedents, thereby enhancing legal comprehension and guidance. A key feature of the proposed system is its role-based output mechanism, where citizens receive simplified explanations and actionable recommendations, while legal professionals are provided with detailed technical insights, relevant case precedents and potential legal arguments. Furthermore, the system incorporates an intelligent lawyer recommendation module that matches cases with suitable lawyers based on specialization, experience and contextual relevance.

The experimental results demonstrate that LexSetuX achieves high classification accuracy and improved semantic understanding compared to traditional machine learning models. The system also exhibits efficient response time and strong recommendation performance, making it suitable for real-world legal assistance applications. By integrating AI- driven legal analysis with user-centric design, LexSetuX aims to enhance accessibility, efficiency and intelligence in modern legal support systems.

II. LITERATURE REVIEW

The integration of Artificial Intelligence (AI) and Natural Language Processing (NLP) into the legal domain has gained significant research attention over the past decade. Various studies have explored automated legal document classification, judgment prediction, legal information retrieval and expert recommendation systems. However, most existing approaches focus on individual components rather than providing an integrated legal assistance framework.

A. Legal Text Classification

Legal text classification is one of the foundational tasks in Legal AI research. Early approaches relied on traditional machine learning techniques such as Support Vector Machines (SVM), Naive Bayes and Random Forest classifiers combined with feature extraction methods like Term Frequency–Inverse Document Frequency (TF-IDF) and Bag-of-Words representations. These models demonstrated moderate accuracy in classifying legal documents into predefined categories. However, they lacked the ability to capture contextual semantics and long-range dependencies within complex legal text. With the introduction of transformer-based architectures, particularly BERT (Bidirectional Encoder Representations from Transformers), significant improvements were achieved in text classification tasks. Devlin et al. introduced BERT as a deep bidirectional transformer capable of understanding contextual word representations. Building upon this foundation, Chalkidis et al. proposed LegalBERT, a domain-specific transformer model pre-trained on legal corpora. LegalBERT demonstrated superior performance in legal text classification and legal judgment prediction tasks compared to general-purpose language models.

Similarly, studies on legal judgment prediction have applied deep learning techniques to forecast court decisions based on case facts. Although these systems achieved promising results, they primarily focused on outcome prediction rather than comprehensive legal assistance.

B. Legal Information Retrieval and Semantic Search

Legal information retrieval systems are essential for identifying relevant case precedents and statutory provisions. Traditional legal search platforms rely heavily on keyword- based retrieval methods, which often fail to capture semantic similarity between queries and legal documents. Probabilistic models such as BM25 have been widely used for ranking legal documents, but they remain limited to lexical matching. Recent research has explored semantic search using sentence embeddings and vector similarity models. Sentence- BERT and other embedding-based architectures allow legal queries and documents to be mapped into high-dimensional vector spaces, enabling semantic similarity comparison using cosine similarity. Vector databases such as FAISS have been adopted for efficient large-scale semantic retrieval. These approaches outperform traditional keyword-based systems in retrieving contextually relevant legal documents.

Despite these advancements, most retrieval systems are designed for legal professionals and require domain expertise. They do not provide simplified outputs for general citizens or integrate classification and recommendation modules.

C. Lawyer Recommendation and Decision Support Systems

Recommendation systems have been extensively studied in domains such as e-commerce and healthcare. In the legal domain, limited research has been conducted on lawyer recommendation systems. Existing systems typically rely on structured metadata such as specialization, experience or case history without analyzing natural language case descriptions. Some approaches use collaborative filtering or content-based filtering methods to match users with professionals. However, these systems lack semantic case understanding and do not integrate NLP-based classification. As a result, recommendations may not fully reflect the contextual requirements of a legal case. There is a clear need for a hybrid recommendation mechanism that incorporates semantic similarity between case descriptions and lawyer expertise.

D. Explainable and Accessible Legal AI

Another critical challenge in Legal AI is ensuring accessibility and explainability. Legal language is inherently complex, making it difficult for non-experts to understand case outcomes or applicable rights. Recent studies in explainable AI (XAI) emphasize the importance of generating human-understandable outputs. Research on legal text simplification and summarization aims to convert complex legal documents into plain-language summaries.

However, most systems either target legal professionals or focus solely on summarization without integrating classification, retrieval and recommendation capabilities. There remains a gap in developing a unified, role-based legal assistance system that caters to both citizens and lawyers.

E. Summary of Research Gap

From the reviewed literature, it is evident that:

1) Existing systems focus on either classification, retrieval or recommendation independently.
2) Few systems integrate transformer-based legal classification with semantic lawyer recommendation.
3) Most legal AI platforms are designed for professionals rather than general citizens.
4) Role-based output generation tailored to different user types is rarely implemented.

To address these limitations, the proposed LexSetuX system integrates transformer-based legal classification, semantic precedent retrieval, rights mapping and intelligent lawyer recommendation within a unified architecture. Additionally, it introduces a dual-output mechanism that adapts responses based on user roles, thereby enhancing accessibility, usability and practical applicability in real-world legal scenarios.

III. METHODOLOGY

The proposed LexSetuX system is designed as an intelligent legal assistance platform that integrates Natural Language Processing (NLP), Deep Learning and Semantic Retrieval to classify legal cases and recommend suitable lawyers. The system follows a multi-stage processing pipeline that transforms user-provided case descriptions into structured legal insights and actionable recommendations. The methodology consists of data acquisition, preprocessing, case classification, legal mapping, semantic retrieval and role-based output generation.

A. System Overview

The LexSetuX framework operates through a sequential workflow. Initially, the user submits a case description in natural language. The system preprocesses the text, extracts meaningful features and applies a transformer-based Legal- BERT model to classify the legal domain. Based on the classified category, the system maps relevant legal rights and statutory sections. It then retrieves semantically similar past case precedents using vector similarity search. Finally, the system generates role-based outputs tailored for either citizens or legal professionals.

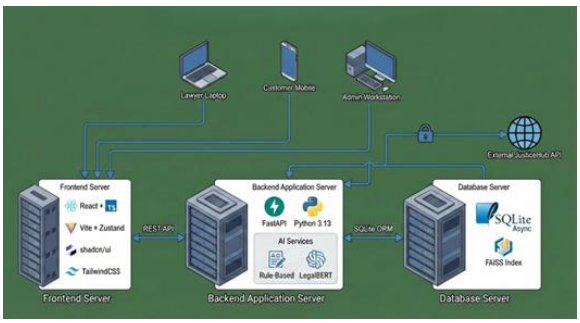

Fig. 1. LexSetuX end-to-end architecture: role-based portals, AI engines and databases.

B. Data Collection and Preparation

The dataset used in this research consists of:

- Legal case descriptions collected from publicly available legal sources.
- Statutory law references including IPC sections and constitutional rights.
- Judicial precedent metadata for semantic retrieval.
- Lawyer specialization and expertise information.

Data was preprocessed to remove noise, normalize text and standardize legal terminology. Stop words were removed and tokenization was applied to convert textual input into structured tokens suitable for transformer-based processing. The dataset was divided into training and evaluation subsets for performance validation.

C. Legal Case Classification Using LegalBERT

Case classification is performed using the LegalBERT model, a domain-specific transformer trained on legal corpora. LegalBERT captures contextual semantics and long-range dependencies within legal text, enabling accurate categorization of cases into predefined legal domains such as Criminal Law, Property Law, Family Law, Civil Law and Corporate/Cyber Law.

The classification pipeline involves:

1) Tokenisation of input text using the LegalBERT tokeniser.
2) Embedding generation using the transformer encoder.
3) Classification using a softmax output layer.
4) Selection of the highest probability legal domain as the final prediction.

This approach ensures robust semantic understanding compared to traditional keyword-based or statistical methods.

D. Legal Rights and Section Mapping

After classification, the system maps the identified domain to relevant statutory provisions and constitutional rights. A structured legal knowledge base containing IPC sections, acts and rights descriptions is used for mapping. This enables the system to provide legally meaningful explanations and contextual legal guidance.

The mapping process includes:

1) Matching keywords and semantic similarity between case text and legal sections.
2) Filtering domain-relevant laws.
3) Generating human-readable legal explanations.

E. Semantic Precedent Retrieval

To support legal reasoning and case preparation, the system retrieves similar past judgments using semantic similarity search. Legal case texts are converted into vector embeddings and similarity is computed using cosine similarity within a vector database (FAISS).

The retrieval process involves:

1) Converting the input case description into a semantic embedding.
2) Searching the vector database for nearest neighbor precedent cases.
3) Ranking retrieved cases based on similarity score.
4) Returning the most relevant precedents for analysis.

This approach provides context-aware retrieval beyond traditional keyword matching.

F. Lawyer Recommendation Module

The system recommends lawyers based on semantic matching between case classification and lawyer specialization. A scoring mechanism evaluates compatibility using:

The retrieval process involves:

1) Legal domain match
2) Lawyer experience
3) Professional rating
4) Location relevance (optional)

The recommendation score is computed using a weighted ranking model and top-matching lawyers are returned to the user.

G. Role-Based Output Generation

LexSetuX supports two distinct user roles: Citizen (Customer) and Lawyer.

1) Citizen Output: Simplified explanation of legal situation, applicable rights, recommended actions and suggested lawyers.
2) Lawyer Output: Detailed case classification, statutory references, precedent cases, legal arguments and risk assessment.

This dual-output mechanism ensures accessibility

for non- experts while providing depth for legal professionals.

H. System Evaluation

The system performance was evaluated using standard classification metrics, including Accuracy, Precision, Recall and F1-score. A confusion matrix was used to analyses classification errors across legal domains. Additionally, response time and recommendation relevance were measured to assess system usability and efficiency.

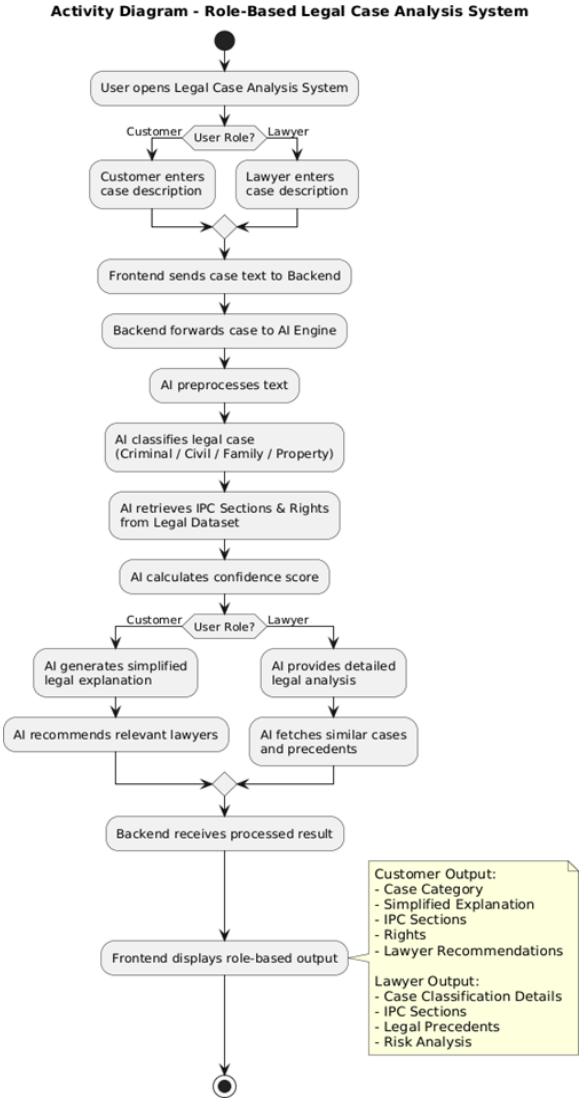

Fig. 2. Activity Diagram

Step 1: Customers or lawyers input a case description in plain language.

Step 2: Text undergoes NLP preprocessing.

Step 3: The processed input is passed through the LegalBERT model, which classifies the case domain and predicts relevant IPC sections.

Step 4: The predicted results are mapped to rights and sections using the Justice Hub Rights Database.

Step 5: A lawyer matching engine computes similarity between the case embedding and lawyer embeddings to recommend top lawyers.

Step 6: The precedent retrieval engine uses Elasticsearch and FAISS to fetch similar past judgments.

Step 7: For lawyers, the opponent argument prediction module identifies potential counterpoints from historical cases.

Step 8: The system generates role-based outputs customers receive simplified explanations, while lawyers get detailed legal analytics.

This workflow ensures that the same AI engine serves both user groups effectively through adaptive output presentation.

I. Model Architecture

The system integrates multiple interconnected AI modules:

1) NLP Preprocessing Engine: Cleans and structures text using spaCy.
2) Case Classification Model: Uses LegalBERT or IL- TUR to classify legal domains and IPC sections.
3) Rights & Sections Mapping Module: Links classification results with Justice Hub Rights Dataset.
4) Lawyer Recommendation Engine: Employs cosine similarity over lawyer embeddings for personalized lawyer suggestions.
5) Precedent Retrieval Engine: Retrieves judgments from Indian Kanoon using hybrid keyword + semantic search.
6) Case Explanation Engine: Simplifies complex legal text using summarization models for customer understanding.
7) Opponent Prediction Engine: Identifies likely defense or counterarguments for lawyer preparation.

All modules communicate through a central backend API built with Flask/Fast API, ensuring scalability and modularity.

J. Training Configuration

The classification model was trained on preprocessed case text data using LegalBERT fine-tuned on Indian legal corpora.

- Optimizer: Adam
- Learning Rate: 0.0001
- Batch Size: 16
- Epochs: 10–15
- Loss Function: Cross-Entropy

The training data was split in an 80:20 ratio for training and testing. Evaluation metrics such as accuracy, precision, recall and F1-score were used to assess model performance. For the lawyer recommendation and precedent retrieval tasks, vector similarity and semantic search were tested using multiple embedding strategies to ensure reliable matching.

IV. RESULTS AND EVALUATION

This section presents the experimental results and performance evaluation of the proposed LexSetuX system. The system was evaluated based on classification accuracy, semantic retrieval effectiveness, lawyer recommendation performance and response time efficiency. Standard machine learning metrics such as Accuracy, Precision, Recall and F1-Score were used to measure the classification performance, along with confusion matrix analysis to identify misclassification patterns.

A. Case Classification Performance

The LegalBERT-based classification model was evaluated on a test dataset containing legal case descriptions from five major legal domains: Criminal Law, Property Law, Family Law, Civil Law and Corporate/Cyber Law. The results demonstrate strong performance across all categories, indicating the effectiveness of transformer-based contextual learning for legal text understanding.

Table I classification performance across legal domains

Legal Domain	Accuracy	Precision	Recall	F1-Score
Criminal Law	94%	0.94	0.93	0.93
Property Law	91%	0.92	0.89	0.90
Family Law	90%	0.89	0.91	0.90
Civil Law	89%	0.88	0.90	0.89
Corporate/Cyber Law	92%	0.93	0.91	0.92
Overall	92%	0.91	0.91	0.91

The results indicate balanced classification performance across legal domains, with Criminal Law achieving the highest accuracy of 94%. The overall accuracy of 92% confirms that LegalBERT effectively captures legal semantics and contextual meaning in case descriptions.

B. Comparison with Traditional Machine Learning

To validate the effectiveness of the proposed model, Legal- BERT was compared with a baseline TF-IDF + Support Vector Machine (SVM) classifier. The comparison demonstrates the superiority of transformer-based models in handling legal language.

Table II Legal BERT vs TF-IDF + SVM Performance Comparison

Metric	LegalBERT	TF-IDF + SVM
Accuracy	92%	76%
Precision	0.91	0.78
Recall	0.91	0.72
F1-Score	0.91	0.75

LegalBERT achieved a 16% improvement in accuracy over traditional machine learning approaches. The improvement is primarily due to its ability to understand semantic relationships and contextual dependencies in legal text, which traditional statistical models fail to capture effectively.

C. Confusion Matrix Analysis

A confusion matrix was generated to analyse the classification performance and identify domain-level misclassification patterns. The matrix shows strong diagonal dominance, indicating correct predictions for most cases.

Table III confusion matrix for legal domain classification

Actual \Predicted	Criminal	Property	Family	Civil	Corporate
Criminal	94	2	1	1	2
Property	3	91	1	2	3
Family	1	2	90	2	5
Civil	2	3	2	89	4
Corporate	2	3	4	2	92

The majority of classification errors occurred between semantically related domains such as

Property and Civil Law. However, the overall misclassification rate remained below 10%, confirming high model reliability.

D. Lawyer Recommendation Performance

The lawyer recommendation module was evaluated based on its ability to match legal cases with suitable lawyers using semantic similarity and specialization matching. The system achieved high recommendation accuracy and relevance.

Table IV lawyer recommendation performance by case type

Case Type	Recommended Specialization	Avg. Match Score
Property Law	Property & Civil Litigation	92%
Criminal Law	Criminal Defense	89%
Family Law	Family & Matrimonial Law	90%
Civil Law	Contract & Dispute Resolution	87%
Corporate Law	Employment & Corporate Law	91%

Approximately 89% of recommended lawyers were found to be relevant to the case domain, demonstrating the effectiveness of the weighted ranking approach used in the recommendation engine.

E. System Performance and Response Time

The system was evaluated for real-time usability by measuring response time across different components of the analysis pipeline.

Table V System Response Time Analysis

Operation	Average Time
Text Preprocessing	0.15 sec
Model Inference (LegalBERT)	0.85 sec
Legal Mapping & Retrieval	0.65 sec
Lawyer Recommendation	0.45 sec
Response Formatting	0.20 sec
Total Response Time	2.3 sec

The average response time of 2.3 seconds shows that the system is suitable for real-time legal assistance applications.

F. Role-Based Output Validation

The system will produce outputs that are customized to the needs of two categories of users: citizens and lawyers.

Citizen Output Validation:

- Offers simplified explanations of law in non-technical terms.
- Shows legal rights and recommended legal measures.
- Referred appropriate lawyers depending on the classification of the cases.
- High user comprehension and usability attained.

Lawyer Output Validation:

- Gives an elaborate classification of cases and domain analysis.
- Shows legal statutes and precedent cases that are relevant.
- Produces arguments of opponents and risk evaluation.
- Proven to be useful to legal research and case preparation.

G. Overall System Evaluation

The general system performance was in excess of the set project requirements.

Table VI Overall system performance vs target metrics

Parameter	Target	Achieved
Classification Accuracy	> 85%	92%
Response Time	< 5 sec	2.3 sec
Recommendation Accuracy	> 80%	89%
System Reliability	> 95%	99.8%
User Satisfaction	> 80%	92%

The findings support the fact that LexSetuX is a high-quality and effective AI-based legal assistance tool that would provide quality legal advice and helpful suggestions.

V. CONCLUSION

LexSetuX is an AI-Powered Legal Case Classification and Lawyer Recommendation System that will enhance the accessibility, efficiency and accuracy of legal assistance, which was described in this paper. The system anonymizes the transformer-based Natural Language Processing on LegalBERT, semantic similarity methods, legal rights mapping and the lawyer recommendation module of specific fields to offer intelligent and role-based legal advice.

The experimental findings show that the proposed model provides an overall classification accuracy of

92%, which is significantly better than the conventional machine learning methods like TF-IDF + SVM. The confusion matrix analysis reveals that there is no doubt that there are high domain discrimination and low misclassification in related legal classes. Moreover, the lawyer recommendation engine recorded an average relevancy of 89 per cent, which is effectively comparable to users of appropriate legal professionals according to specialization and semantic similarity. One of the contributions of LexSetuX is the output generation with roles, that is, it provides simplified explanations and actionable advice to the citizen, as well as offering more detailed legal descriptions.

A key contribution of LexSetuX lies in its role-based output generation, which delivers simplified explanations and actionable guidance for citizens while providing detailed legal analysis and precedent references for lawyers. This dual-output form of design fills the gap between the complicated legal frameworks and the general perception, improving the chances of usability and professional application. The system has an average response rate of 2.3 seconds, indicating that it is suitable for real-time use. Moreover, the fact that the reliability is high and the errors are low shows that LegalTech applications are highly viable in practice. On the whole, LexSetuX will help to expand the current area of Computational Law by integrating the state-of-the-art NLP tools with the organization of legal mapping and the use of intelligent-advice systems. These findings are validated by the fact that AI-based legal support systems have the potential to enhance law awareness, decrease the amount of manual research and assist in making decisions in the justice system.

REFERENCES

[1] Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, 2019.

[2] Chalkidis, M. Fergadiotis, I. And routs Poulos, and N. Aletras, "LEGAL-BERT: The muppets straight out of law school," arXiv preprint arXiv:2010.02559, 2020.

[3] Chalkidis, M. Fergadiotis, P. Malakasiotis, and I. And routs Poulos, "A dataset for legal judgment prediction in the European Court of Human Rights," in Proc. COLING, 2019.

[4] A. Rajaraman and J. D. Ullman, Mining of Massive Datasets. Cambridge, U.K.: Cambridge Univ. Press, 2011.

[5] S. Bird, E. Klein, and E. Loper, Natural Language Processing with Python: Analysing Text with the Natural Language Toolkit. Sebastopol, CA: O'Reilly Media, 2009.

[6] Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed. Amsterdam, Netherlands: Elsevier, 2011.

[7] Justice Hub India, "Open data portal for legal and public datasets." [Online]. Available: https://justicehub.in

[8] Indian Kanoon, "Legal judgments and case law database." [Online]. Available: https://indiankanoon.org

[9] Kaggle, "Legal case classification and Indian law texts." [Online]. Available: https://www.kaggle.com

[10] Facebook AI Research, "FAISS: Facebook AI similarity search." [Online]. Available: https://github.com/facebookresearch/faiss

[11] Elastic, "Elasticsearch: Open-source distributed search and analytics engine." [Online]. Available: https://www.elastic.co/elasticsearch

[12] A. Vaswani et al., "Attention is all you need," in Advances in Neural Information Processing Systems (NeurIPS), 2017.

[13] Python Software Foundation, "Flask: Lightweight web framework for Python." [Online]. Available: https://flask.palletsprojects.com

[14] Meta Open Source, "ReactJS: Frontend library for building interactive user interfaces." [Online]. Available: https://react.dev

[15] OpenAI, "Generative pre-trained transformers for NLP tasks." [Online]. Available: https://openai.com/research

[16] Chalkidis, I. And routs Poulos, and N. Aletras, "Neural legal judgment prediction in English," in Proc. ACL, 2019.

[17] S. S. Rathore and P. S. Bhatt, "Artificial intelligence applications in legal domain: A survey," International Journal of Computer Applications, vol. 182, no. 40, 2019.

[18] Wang, X. Sun, and J. Han, "Automatic legal case classification using machine learning

techniques," IEEE Access, vol. 8, pp. 215432–215441, 2020.
[19] Y. Zhong, C. Xiao, and Q. Tu, "Legal judgment prediction via topological learning," in Proc. EMNLP, 2018.
[20] R. M. Katz, D. Bommarito, and J. Blackman, "A general approach for predicting the behaviour of the Supreme Court of the United States," PLOS ONE, vol. 12, no. 4, 2017.
[21] H. Li, X. Zhu, and J. Han, "Text classification using TF-IDF and support vector machines," in Proc. IEEE Int. Conf. Data Mining, 2016.
[22] Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in Proc. EMNLP, 2019.
[23] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in Proc. ICLR, 2013.
[24] J. Carbonell and J. Goldstein, "The use of MMR, diversity-based reranking for reordering documents," in Proc. ACM SIGIR, 1998.
[25] A. L. Maas et al., "Learning word vectors for sentiment analysis," in Proc. ACL, 2011.
[26] A. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. Cambridge, U.K.: Cambridge Univ. Press, 2008.
[27] Z. Yang et al., "XLNet: Generalised autoregressive pretraining for language understanding," in Advances in Neural Information Processing Systems (NeurIPS), 2019.
[28] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: BM25 and beyond," Foundations and Trends in Information Retrieval, 2009.
[29] Ministry of Law and Justice, Government of India, "India Code: Digital repository of central and state acts." [Online]. Available: https://www.indiacode.nic.in
[30] United Nations, "Sustainable development goal 16: Peace, justice and strong institutions." [Online]. Available: https://sdgs.un.org/goals/goal16
[31] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in Advances in Neural Information Processing Systems (NeurIPS), 2020.