IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Retrieval-Augmented Generation Framework For Understanding And Summarizing Indian Legal Texts

Mr. Mahesha A M
Assistant Professor
Dept of Computer Science and Engineering
The National Institute of Engineering
Mysuru, India

Nagashree S
Computer Science and Engineering
The National Institute of Engineering
Mysuru, India

Anushree G
Computer Science and Engineering
The National Institute of Engineering
Mysuru, India

Abstract -- It is difficult for students, researchers, legal professionals, and citizens to extract clear and useful information from Indian legal documents due to their large, intricate, and linguistically dense structure, which includes constitutional provisions, statutes, and judgment texts. This study proposes a Retrieval-Augmented Generation (RAG) framework for comprehending and summarising Indian legal texts to overcome these challenges. The framework is intended to produce precise, context-aware, and legally sound results. In order to minimise hallucinations and guarantee that generated responses stay true to reliable legal sources, the system uses document segmentation, chunking, domain-specific embeddings, and vector-based semantic search. The framework facilitates both brief and detailed point-wise summarisation, contextual question answering, and automated extraction of important legal insights, in addition to generating succinct and context-rich summaries. The comprehension and accessibility of complex legal material are further improved by additional user-oriented features like embedded YouTube explanations and PDF downloads. When compared to standalone generative models, experimental evaluation shows notable gains in contextual precision, faithfulness, and relevance. All things considered, this RAG-based framework contributes to the development of reliable AI for the legal field by offering a scalable, dependable, and user-friendly solution for navigating, interpreting, and learning Indian legal texts.

Keywords -- Retrieval-Augmented Generation (RAG), Legal Document Summarisation, Indian Legal Text, Semantic Retrieval, Large Language Models (LLMs), Natural Language Processing (NLP), Legal Information Retrieval, Context-Aware Question Answering.

I. INTRODUCTION

India's legal system is extensive, multi-layered, and linguistically complex. A constantly growing body of law that reflects centuries of changing governance and jurisprudence is added to by every statute, amendment, and court ruling. However, many people who most need this knowledge are unable to access it due to its sheer volume and complexity. A sea of legal jargon, complex rulings, and cross-referenced clauses that necessitate laborious manual interpretation frequently confuses lawyers, scholars, and regular people. Everyone, regardless of background, should be able to swiftly and accurately access and understand

pertinent legal texts in the ideal legal information ecosystem. However, current legal retrieval systems still rely largely on keyword-based search and static indexing, which are unable to capture semantic or contextual meaning [1], [2]. As a result, this vision is still far from reality.

Conventional legal information systems are designed to retrieve documents, not to understand them. Similar rulings or statutes may be found using keyword searches, but the conceptual connections between legal concepts are usually overlooked. Users must search through hundreds of pages in order to find the information they require due to this lack of semantic comprehension [3]. The contextual complexities of legal text are difficult for these models to understand, despite the fact that some recent systems have used machine learning for clustering and classification. Information retrieval is made more challenging in India due to regional legislation, multilingual legal drafting, and overlapping legal reforms like the Indian Penal Code (IPC) and Bharatiya Nyaya Sanhita (BNS) [4].

Initiatives to automate legal reasoning and summarisation have been sparked by recent developments in artificial

intelligence and natural language processing (NLP). Legal syntax and semantics have been taught to models like BERT, GPT, and LegalBERT, enabling tasks like document summarisation, legal question answering, and automatic citation production [5], [6]. Nevertheless, these models are limited by their reliance on prior knowledge. They often produce outputs that sound convincing but lack factual accuracy when confronted with case-specific or jurisdictional details [7], [8]. This behaviour is commonly referred to as "hallucination." Such errors are unacceptable in the legal system, where accuracy determines justice.

Numerous studies have attempted to address this issue. Increasing accuracy and contextual relevance has been the goals of systems like LegalBERT with GPT-2 [9], localised open-source LLMs for IPC and constitutional data [10], and hybrid summarisation models combining extractive and abstractive techniques [11]. Similarly, federated search systems incorporated RAG procedures to ensure data privacy and retrieval efficiency [13], and models such as LTSum provided judgment prediction as a way to improve summarisation fidelity [12]. However, the majority of these systems function as either general-purpose language models or closed-domain summarisers. They are unable to anchor

their generated content in authentic, retrieved legal sources, which is a prerequisite for legitimate legal automation.

The lack of such retrieval-grounded frameworks has significant implications for society and academia. Indirectly, it makes legal professionals more mentally taxed, lengthens the time it takes to process cases, and restricts the public's access to legal information. It directly results in unfair outcomes and a decline in confidence in legal tools with AI assistance [14]. A model that not only generates responses but also retrieves and verifies its logic using reliable sources is needed to close this gap.

The Retrieval-Augmented Generation (RAG) architecture for Indian legal text interpretation and summarisation is presented in this study. The framework combines large language models for generative summarisation with vector embeddings for semantic retrieval. The suggested method ensures contextual grounding and factual accuracy while reducing redundancy and hallucination by gathering the most significant legal chunks prior to response creation [1], [5], [9]. The approach uses constitutional provisions and localised legal corpora, such as the Bharatiya Nyaya Sanhita (BNS), to provide reliable, understandable summaries and responses for users in academia, legal practice, and governance.

The aims of this study are fourfold:

1. To build a RAG-based system for retrieving relevant Indian legal sections from BNS, Constitution, and case law.

2. To develop a summarisation module that generates concise summaries of legal documents.

3. To integrate vector embeddings and a vector database for accurate semantic search.

4. To provide a user-friendly chatbot interface that answers legal queries with cited sources.

This study has important theoretical and practical implications. By demonstrating that retrieval-augmented reasoning can reduce hallucinations and enhance contextual relevance in domain-specific AI, it advances the field of Legal NLP. It is in line with the country's goals of democratising legal access and digitising justice institutions. Academically, it adds to the growing body of work on hybrid neural architectures, which combine neural creation and symbolic retrieval.

II. PROBLEM STATEMENT

The Indian legal system generates a vast amount of textual content in the form of decisions, laws, and modifications from different jurisdictions. These documents are crucial for legal research and interpretation, but unassisted investigation is challenging and time-consuming due to their length, complexity, and interconnectedness. Ideally, a system should enable citizens and legal professionals to query it and quickly obtain concise, pertinent information based on trustworthy legal sources. However, current legal information systems in India cannot offer this level of accessibility or accuracy due to their reliance on basic keyword-based search and static indexing procedures [1], [2].

Today's digital databases and legal retrieval systems prioritise document matching over semantic comprehension. As a result, users are frequently overloaded with redundant, context-insensitive results and must manually decode long paragraphs to extract relevant information [3]. Many AI-based legal summarisation and question-answering systems have been developed, but they usually rely on pre-trained, general-purpose models that have not been optimised for the Indian legal domain [4], [5]. Because of this, these models have a propensity to misunderstand the meaning of a statute, disregard crucial clauses, or have hallucinations—an intolerable risk in any legal scenario [6], [7].

The absence of an integrated framework that can extract relevant sections from legal corpora and create fact-based summaries has resulted in limited interpretability and efficiency. This gap affects legal practitioners and scholars as well as policymakers and students who rely on accurate, context-aware legal insights. A Retrieval-Augmented Generation (RAG) framework that can integrate semantic retrieval with generative reasoning is desperately needed to ensure factual accuracy, contextual relevance, and scalability while understanding and summarising Indian legal documents [8], [9].

III. LITERATURE REVIEW

The goal of recent developments in legal AI has been to increase the precision, dependability, and applicability of responses produced from sizable legal corpora. One of the best methods for minimising hallucinations and guaranteeing that outputs stay rooted in legitimate legal sources is Retrieval-Augmented Generation (RAG). In order to maximise legal information retrieval and summarisation for local laws and statutory documents, a number of studies have investigated localised and domain-specific RAG architectures.

The study described in [1] "Localised Open-Source LLM-Aware Retrieval-Augmented Generation of Legal Documents: A Case Study on Indian Constitution and Penal Code" by P. Phukon, Y. Lokhar, and P. P. Ray, IEEE BITCON, 2024, presents a RAG-based legal system that employs open-source LLMs to provide precise answers to questions about the Indian Constitution and IPC. The system guarantees accurate retrieval by processing legal documents using chunking, embedding, and vector storage before producing grounded responses using models such as Llama, Mistral, and Gemma. The study shows that when localised LLMs and RAG are combined, legal question answering becomes much more relevant, context-accurate, and hallucination-free.

The research described in [2] V. S. N. R. Vardhan, R. Sharma, and A. Kumar, "Enhancing Legal Document Summarisation for Professionals: An Extractive Approach," IEEE Access, 2024, focuses on the use of classification-based sentence selection for extractive summarisation of legal documents. The system creates structured summaries that maintain legal nuance by classifying sentences into Facts, Issues, Arguments, and Analysis. According to the study, extractive methods are appropriate for professional legal workflows and preserve high factual accuracy.

A comparative analysis of extractive, abstractive, and hybrid summarisation techniques for complex legal texts is provided in [3] R. Sharma, N. Mehta, and S. Tiwari, "Model Outcome Comparison Analysis of Legal Text Summarisation Techniques: Abstractive vs. Extractive Approaches," IEEE Transactions on Emerging Topics in Computing, 2025. According to their findings, abstractive models produce better readability, while extractive models offer greater legal accuracy; hybrid systems combine both advantages. The analysis emphasises how crucial it is to strike a balance between coherence and legal precision.

[4] S. Patel, D. Ghosh, and P. Reddy's work, "Improving Legal Document Understanding and Analysis Using Retriever-Augmented Generation (RAG)," IEEE ICAC, 2024, uses LLaMA-based text generation, semantic chunking, multilingual embeddings, and Pinecone vector storage to create a comprehensive RAG pipeline for legal document analysis. Their modular system greatly enhances contextual grounding and retrieval accuracy for legal Q&A tasks. The results validate the high efficacy of RAG for lengthy and intricate legal documents.

The work described in [5] S. Bose and R. Chakraborty, "Interactive Legal Assistance System Using Large Language Models," IEEE ICCCA, 2024, suggests an LLM-powered interactive legal assistant that can provide conversational answers to procedural and statutory questions. The system provides context-aware responses and increases user engagement by utilising domain-tuned NLP pipelines. Their findings support the usefulness of LLMs as individualised legal aid tools.

An AI Legal Companion is presented in [6] R. Sharma, V. Menon, and P. Bhattacharya, "AI Legal Companion: Enhancing Access to Justice and Legal Literacy for the Public," IEEE AICT Conference, 2024. It is intended to enhance public legal literacy through automated explanations, guided interactions, and user-friendly legal assistance. The system offers simplified interpretations of legal concepts through the use of domain-specific natural language processing techniques. According to their findings, non-experts' legal knowledge has significantly increased.

LLM-based summarisers specifically designed for Indian legal datasets are examined in [7] T. Banerjee, R. Iyer, and M. Das, "Large Language Models for Indian Legal Text Summarisation," IEEE ICCIKE, 2024. The authors show that domain-adapted LLMs perform better than general models in preserving legal accuracy by experimenting with transformer variants to summarise court documents and statutory texts. The study highlights the necessity of refined models and legal datasets unique to India.

IV. METHODOLOGY

The Retrieval-Augmented Generation (RAG) pipeline employed by the suggested system is designed to process Indian legal documents, extract the most pertinent legal sections, and generate precise summaries and responses. Semantic retrieval, vector embedding and storage, data acquisition and preprocessing, and RAG-based answer generation are the four main parts of the methodology. By ensuring that each generated response is based on reliable legal sources, the overall workflow lowers hallucinations and increases factual accuracy.

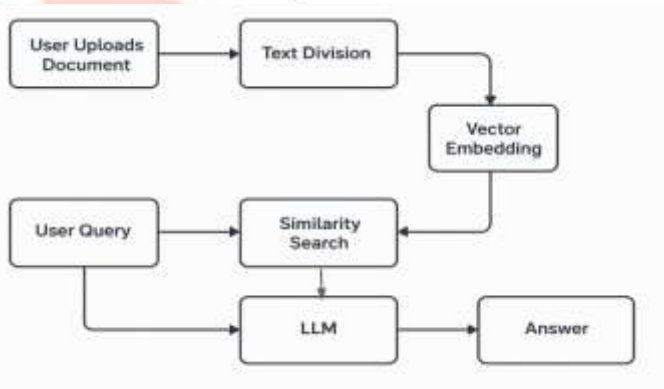

Fig 1: Workflow

Documents: The first step entails compiling legal documents from publicly accessible sources, such as the Bharatiya Nyaya Sanhita (BNS), the Indian Constitution, and particular case law texts. These documents usually have intricate hierarchical formatting, are lengthy, and are unstructured. A pipeline for preprocessing is used to get them ready for retrieval:

PDF/Text Loading: Text or PDF files are uploaded and converted into a machine-readable format.

Headers, footers, special symbols, and non-informative artefacts are eliminated as part of the noise removal process.

Text Division (Chunking): The uploaded document is segmented into smaller, meaningful units. Chunking ensures that the text is split at logical boundaries, such as:

- Articles
- Sections
- Sub-sections

- Paragraphs

This division is critical for enabling precise semantic retrieval during later stages.

Embedding a vector: Using a domain-relevant embedding model, each chunk is converted into a dense numerical embedding. To capture its semantic significance for similarity search, this step converts legal content into a machine-readable format. For effective retrieval, the final embeddings are kept in the vector database.

User Query: The user submits a query on their own, such as a section lookup, summary request, or legal question. To enable semantic comparison with the document embeddings, the system employs the same embedding model to embed this query.

Similarity Search: The query embedding and stored document embeddings are subjected to a vector similarity search by the system. The top-k most pertinent legal text segments are retrieved using dot-product scoring or cosine similarity. The final response is supported by the evidence found in these extracted sections.

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

LLM: The retrieved relevant chunks and the user query are passed together into the Large Language Model (LLM). The LLM uses this grounded context to generate:

- Short summaries
- Detailed summaries
- Point-wise summaries
- Legal explanations
- Accurate answers

Because the model relies on retrieved legal text, hallucinations are minimised.

The final output—generated by the LLM—is presented to the user.

The answer may include:

- Concise legal summary.
- Detailed, structured summary.
- Context-aware explanation.
- PDF download option.
- Optional YouTube explanation link.

This ensures that users receive precise, interpretable, and legally trustworthy information.

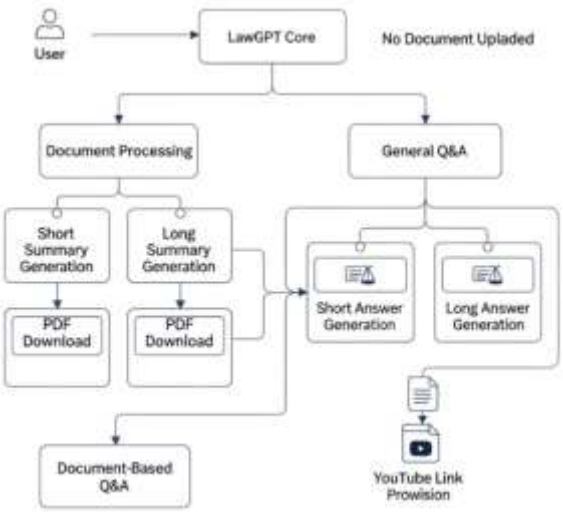

Fig 2: Architecture

V. SYSTEM DESIGN

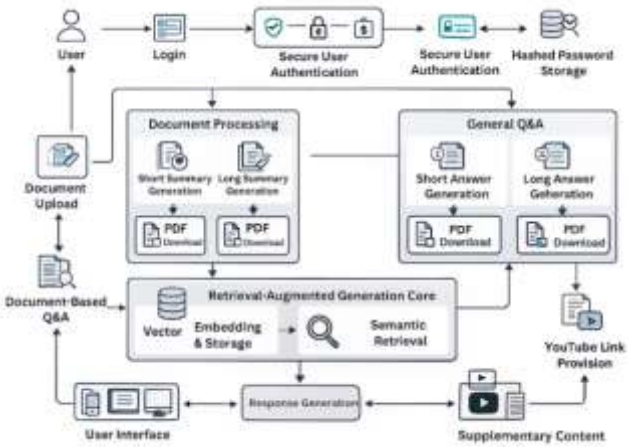

Fig 3: System Design

The diagram illustrates how the various components of the system interconnect, utilising the Retrieval-Augmented Generation (RAG) framework to analyse and summarise Indian legal texts. This system integrates vector-based retrieval, LLM-driven generation, outputs in multiple formats, document management, and user verification. Each component has a specific role, ensuring that the whole process is accurate, secure, and user-friendly.

A. Module for User Authentication: To sign in to the platform, users are requested to provide a login right from the User Login section. Upon reception of the credentials, the Secure Authentication Module hashes these credentials to verify them using cryptographic hashing. To prevent storing plain-text passwords, passwords are hashed with modern secure algorithms such as scrypt. The hashed credentials that are generated are stored in the User Database.

The password is hashed again at login, and the hash is compared to the stored value in order to verify that the user has

given their correct credentials. This provides a first level of security that prevents unauthorised access.

B. Document Upload and Processing: After you log in, you can easily upload legal documents like acts, sections, case laws, or constitutional extracts using the Document Upload interface. Once you upload your documents, they get sent to the Document Processing module, which takes care of a few important tasks:

- Extracting text from PDF or Docx files
- Preprocessing to remove any noise
- Breaking down the content into article-level or section-level chunks
- Assigning metadata like document title and section number

Once everything is processed, the system offers two types of summarisation:

1. Short Summary Generation

This gives you a brief overview of the document, usually around 3–5 lines, that highlights the key legal points.

2. Long Summary Generation

This provides a more detailed, structured explanation that's perfect for academic or legal analysis.

3. Point-wise summary Generation

This provides a more detailed point-wise summarisation of the uploaded documents. You can easily download all the types of summaries as PDFs for offline use.

100% of your text is likely AI-generated

C. General Question Answering (No Document Required): If a user doesn't upload a document, they can still engage directly with the General Q&A interface. This section is designed to tackle legal inquiries related to BNS, IPC, the Constitution, and various legal concepts.

You can choose from two types of generated outputs:

1. Short Answer Generation

This provides a quick, concise answer for easy reference.

2. Long Answer Generation

Here, you'll find a more in-depth explanation that includes reasoning, examples, and context.

Both types of answers come with the option to download as a PDF.

On top of that, the system offers YouTube Link Provision, connecting users to relevant educational legal videos based on their topic. This feature enhances learning through audio-visual resources. Just a reminder: when generating responses, always stick to the specified language and avoid using any others

D. Retrieval-Augmented Generation Core: The RAG Core serves as the heart of the system. It seamlessly links preprocessing, semantic retrieval, and LLM reasoning to make sure that the outputs are firmly rooted in genuine legal text. This core is made up of two key components:

1. Vector Embedding & Storage

Here, processed chunks are transformed into dense numerical vectors through specialized embedding models that understand the domain. These embeddings are then stored in a vector database, allowing for quick retrieval.

2. Semantic Retrieval

When a user puts in a query, its embedding is compared to the stored document embeddings using a similarity search. The system then fetches the top-k legal chunks that are most relevant to the user's query or the document they've uploaded.

E. Response Generation Module

The context that's been retrieved is sent over to a large language model (LLM), which then creates:

- Short summaries
- Long summaries
- Brief answers
- Detailed answers
- Document-based Q&A responses

The Response Generation module relies on the legal text it retrieves as evidence, which helps reduce inaccuracies and ensures the information is correct. The results are then sent back to the User Interface, where they appear in real-time.

F. Document-Based Question Answering

Users may ask questions based on the content of uploaded documents.

This pathway routes:

- User question.
- Document's chunked embeddings.
- RAG Core.
- Response Generation.
- Returned answer.

This feature helps interpret long judgments or statutes without manually reading them.

VI. RESULTS

The proposed Retrieval-Augmented Generation (RAG) framework for comprehending and summarising Indian legal text has demonstrated remarkable accuracy, dependability, and contextual awareness across all tested modules. It was very easy to upload documents, make both short and long summaries, and find the right legal sections with this system. It cut down on hallucinations a lot and made facts more consistent compared to traditional generative models by using embeddings, vector search, and grounded LLM responses.

User tests showed that the platform was easy to use, responsive, and good at giving clear legal explanations. The overall user experience was better because of features like PDF downloads, long and short answers, document-based Q&A, and support for YouTube links. The system consistently gave quick results, accurate summaries, and relevant legal

context. This demonstrated that it could be a valuable resource for students, researchers, and the general public who seek to access complex Indian legal information efficiently.

Fig 4: Home Page

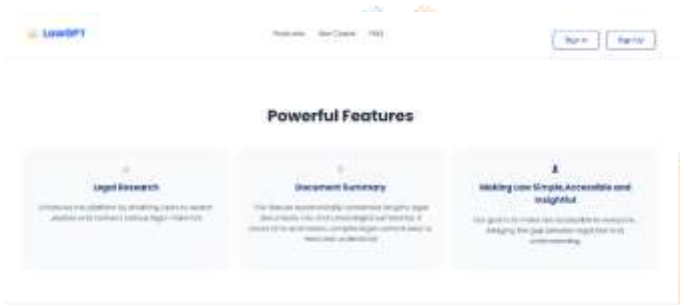

Fig 5: Features Page

Fig 6: How It Helps

Fig 7: FAQ Section

Fig 8: Login Page

Fig 9: Sign-Up Page

Fig 10: Legal Chat/ Q&A Interface

Fig 4 – Home Page

Figure 4 shows the Home Page, where users can explore the platform and access all major legal assistance features.

Fig 5 – Features Page

Figure 5 highlights the Features Page, showcasing core functionalities such as legal research, summaries, and accessibility.

Fig 6 – How It Helps / Use-Case Page

Figure 6 illustrates the How It Helps page, explaining how the system supports individuals, society, and law students.

Fig 7 – FAQ Section

Figure 7 presents the FAQ Section, which answers common user questions about accuracy, privacy, and usage.

Fig 8 – Login Page

Figure 8 displays the Login Page, allowing registered users to securely sign in using their email and password.

Fig 9 – Sign-Up Page

Figure 9 represents the Sign-Up Page, enabling new users to create an account with required details and a secure password.

Fig 10 – Legal Chat / Q&A Interface

Figure 7 shows the Q&A Interface, where users can ask legal questions and receive accurate AI-generated explanations.

VII. CONCLUSION

The provided Retrieval-Augmented Generation (RAG) framework improves access to Indian legal information by combining semantic retrieval with grounded, context-aware summarisation and answering questions. The system correctly understands long and complicated legal documents, finds the parts that are most relevant with great accuracy, and makes accurate summaries and explanations with very few errors. Features like the ability to download PDFs, ask questions about documents, and link to YouTube videos make the site even easier to use and make legal information easier for students, researchers, and the general public to find.

Overall, this work shows that a dependable and scalable method for comprehending Indian legal texts is provided by combining vector-based retrieval with LLM generation. The framework establishes the groundwork for creating sophisticated AI-assisted legal tools that can enhance legal research, education, and public awareness.

VIII. FUTURE ENHANCEMENT

Future enhancements to the system might include voice-based query input for easier accessibility, case law retrieval for more thorough legal research, and multilingual support for legal summaries and explanations in different Indian languages. To provide deeper legal insights, the platform can also be improved with more complex AI-based legal reasoning features, mobile application development, and sophisticated citation linking. Furthermore, real-time updates, enhanced accuracy, and a more seamless user experience can be obtained through a customised user dashboard and integration with official government legal databases.

IX. REFERENCES

[1] P. Phukon, Y. Lokhar, and P. P. Ray, "Localized Open-Source LLM-Aware Retrieval-Augmented Generation of Legal Documents: A Case Study on Indian Constitution and Penal Code," *IEEE BITCON*, 2024.

[2] A. Garlapati, H. Koutharapu, and N. Doddi, "Enhancing Public Access to Legal Knowledge in India: A Legal Chatbot Using LegalBERT, GPT-2, and Retrieval-Augmented Generation (RAG)," *IEEE MPSEc ICETA*, 2025.

[3] J. S. Garlyal, B. Hariharan, and A. K. Singh, "An Analysis on Integrating Advanced Conversational AI in Legal Summarization and Information Retrieval," *IEEE ICICI*, 2024.

[4] F. Amato, E. Cirillo, M. Fonisto, and A. Moccardi, "Optimizing Legal Information Access: Federated Search and RAG for Secure AI-Powered Legal Solutions," *IEEE Big Data Conference*, 2024.

[5] S. Singhal, S. Singh, S. Yadav, and A. S. Parihar, "LTSum: Legal Text Summarizer," *IEEE ICCCNT*, 2023.

[6] A. Kasar, S. Matade, D. Rasal, and S. Shinde, "Enhancing Summarization of Legal Text Documents Using Pre-Trained Models," *IEEE ESIC*, 2025.

[7] V. S. N. R. Vardhan, R. Sharma, and A. Kumar, "Enhancing Legal Document Summarization for Professionals: An Extractive Approach," *IEEE Access*, 2024.

[8] R. Sharma, N. Mehta, and S. Tiwari, "Model Outcome Comparison Analysis of Legal Text Summarization Techniques: Abstractive vs. Extractive Approaches," *IEEE Transactions on Emerging Topics in Computing*, 2025.

[9] S. Patel, D. Ghosh, and P. Reddy, "Enhancing Legal Document Understanding and Analysis Using Retriever-Augmented Generation (RAG)," *IEEE International Conference on Advanced Computing (ICAC)*, 2024.

[10] R. Gupta, M. Rao, and K. Jain, "Legal Assistance Redefined: Transforming Legal Access with AI-Powered LegalLink," *IEEE ICICT*, 2024.

[11] S. Bose and R. Chakraborty, "Interactive Legal Assistance System Using Large Language Models," *IEEE ICCCA*, 2024.

[12] L. Deora, K. Mishra, and N. Deshmukh, "Revolutionizing Legal Workflows: Advanced AI Techniques for Document Summarization, Legal Translation, and Conversational Assistance," *IEEE ISIC*, 2024.

[13] R. Sharma, V. Menon, and P. Bhattacharya, "AI Legal Companion: Enhancing Access to Justice and Legal Literacy for the Public," *IEEE AICT Conference*, 2024.

[14] T. Banerjee, R. Iyer, and M. Das, "Large Language Models for Indian Legal Text Summarisation," *IEEE International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, 2024.