A

MAJOR PROJECT-II REPORT

on

AI Powered Legal Assistant

Submitted by:

Anshul Yadav(220643)

Shruti Bajpayee(220576)

Saransh Bhargava(220379)

under mentorship of

Dr. Satyendr Singh (Professor)

Department of Computer Science Engineering

School of Engineering and Technology

BML MUNJAL UNIVERSITY, GURUGRAM (INDIA)

May, 2025

CANDIDATE’S DECLARATION

We hereby certify that the work on the project entitled, “AI Powered Legal Assistant”, in partial fulfillment of requirements for the award of Degree of Bachelor of Technology in School of Engineering and Technology at BML Munjal University, Is an authentic record of our own work carried out during a period from Feburary 2025 to May 2025 under the supervision of

Dr. Satyendr Singh.

Anshul Yadav (220643)

Shruti Bajpayee (220576)

Saransh Bhargava (220379)

SUPERVISOR’S DECLARATION

This is to certify that the above statement made by the candidate is correct to the best of

my knowledge.

Faculty Supervisor Name: Dr. Satyendr Singh

Signature:

ACKNOWLEDGEMENT

We am highly grateful to **Dr. Satyendr Singh, Professor**, BML Munjal University, Gurugram, for providing supervision to carry out the seminar/case study from Feb-May 2025.

He has provided great help in carrying out our work and is acknowledged with reverential thanks. Without wise counsel and able guidance, it would have been impossible to complete the training in this manner.

We would like to express thanks profusely to thank **Dr. Satyendr Singh**, for stimulating me from time to time. We would also like to thank the entire team at BML Munjal University. We would also thank my friends who devoted their valuable time and helped me in all possible ways toward successful completion.

Anshul Yadav (220643)

Shruti Bajpayee (220576)

Saransh Bhargava (220379)

Table of Contents

Chapter -1: Introduction

Legal awareness in India is often hindered by the overwhelming complexity and volume of legal content available to the public. Ordinary citizens struggle to comprehend legal documents due to jargon-heavy language and the lack of simplified explanations. This creates a divide between the law and the people it is intended to serve, further exacerbated by limited access to timely legal counsel or authentic online resources.

Legal professionals, including advocates, students, and scholars, face additional challenges. Although they may be trained in law, the lack of centralized digital tools to streamline reading, summarizing, and referencing relevant case laws or statutes consumes considerable time. Manual search, comprehension, and drafting add to their workload, increasing the demand for intelligent solutions.

Artificial Intelligence (AI), particularly advancements in Natural Language Processing (NLP), has emerged as a transformative force in several domains, including law. With the rise of transformer-based models like BERT, T5, and GPT, there is immense potential to process and interpret legal texts effectively. These models can decode the linguistic intricacies of law and convert them into human-readable summaries or precise answers.

This project presents an AI-powered legal assistant that integrates NLP technologies with retrieval and summarization models to build an interactive support system. It is capable of converting lengthy legal documents into concise summaries, responding to user legal questions using structured data, and delivering information through a conversational chatbot interface hosted on a web platform.

The assistant is trained on Indian legal datasets that include bare acts, statutes, and publicly available legal cases. These datasets are preprocessed, cleaned, and converted into formats suitable for training models and building a QA knowledge base. The platform not only serves the general public but also supports legal education and professional practice by acting as a reference and productivity tool.

The outcome is a scalable, modular, and cloud-accessible system that demonstrates how legal technology can democratize legal understanding. By merging AI and legal data, the assistant aspires to make legal information more transparent, interactive, and accessible across varying user backgrounds and digital literacy levels.

Chapter -2: Introduction To The Project

2.1 Overview

The AI-powered legal assistant developed in this project comprises three core modules that work in harmony to provide an efficient and user-friendly legal support system. Each module contributes a vital function—summarizing complex legal documents, retrieving accurate answers to statutory questions, and interacting with users through a chatbot hosted on a modern web interface.

The summarization module leverages a transformer-based T5 model, trained on the Indian Legal Case (ILC) dataset. This component processes lengthy case descriptions or legal judgments and generates concise, readable summaries. These summaries help users quickly grasp the essence of complex legal content without needing to read through verbose text.

For the Q&A system, a custom legal question-answer dataset was created by extracting information from major Indian laws. The extracted content was structured into JSON format to facilitate fast retrieval. A dense vector retrieval system using FAISS is employed to fetch relevant legal answers based on user queries, ensuring responses are contextually accurate and aligned with Indian statutes.

The chatbot and web application interface is built using React for the frontend and Firebase for real-time database communication. The backend, written in Python, handles all processing tasks such as querying FAISS, invoking the T5 summarizer, and returning results to the frontend. This triad of technologies ensures the system is interactive, modular, and ready for future deployment or scale.

2.2 Existing System

Currently, there is a noticeable gap in the availability of comprehensive digital tools specifically tailored for Indian law. While legal technology has progressed globally, most tools in the public domain either specialize in a single function—such as legal search, basic chatbots, or judgment summarization—or they are designed with foreign legal systems in mind, particularly those of the United States or United Kingdom.

Many of these tools do not address the complexities and nuances of the Indian legal framework, which is structurally and procedurally distinct from Western legal systems. Additionally, their scope is often limited to commercial use, subscription-based access, or academic prototypes that are not user-friendly for the general public. This restricts their reach and usability for everyday citizens or Indian legal practitioners.

2.3 User Requirement Analysis

The target audience for the AI-powered legal assistant includes a broad spectrum of users, primarily comprising law students, practicing lawyers, and everyday citizens. These users often face difficulty navigating lengthy legal texts and require tools that can provide swift and simplified insights.

Law students benefit from such tools as they can quickly comprehend complex case laws and statutory provisions, aiding in both academic study and legal research. Practicing lawyers can leverage the assistant to summarize legal documents efficiently and find relevant statutory references, improving their productivity during client consultations or court preparations.

Citizens who lack formal legal training can use this tool to better understand their rights and obligations as per Indian law. The system provides direct and comprehensible answers to legal questions, helping bridge the gap between legal professionals and the general public.

By catering to these three categories of users, the project aims to make legal content more approachable, transparent, and accessible, fostering greater legal literacy and promoting informed decision-making across different layers of society.

2.4 Feasibility Study

The system's feasibility is reinforced by its reliance on widely adopted open-source technologies and scalable cloud-based frameworks. Open-source transformer models such as T5 and libraries like Hugging Face Transformers enable the development of highly effective NLP components without the need for proprietary or expensive software licenses. These tools provide access to cutting-edge machine learning capabilities while significantly reducing development costs.

Firebase serves as an efficient backend infrastructure, offering real-time database operations, user authentication, and hosting services. Its integration simplifies data handling between the frontend and backend, making it ideal for rapid prototyping and deployment. The seamless communication between the React frontend and Firebase backend ensures real-time responses and a dynamic user experience.

To handle resource-intensive processes such as training and inference of deep learning models, Google Colab has been utilized. Colab provides access to GPU and TPU hardware accelerators, which are critical for reducing training time and computational overhead. For legal document retrieval, FAISS is employed as a high-performance similarity search engine that enables fast vector-based querying, even in large datasets.

Together, these tools not only demonstrate the technical viability of the system but also make it cost-effective and accessible for academic development and future scaling into production environments.

Chapter -3: Literature Review

3.1 Comparison

Several prior research efforts in legal AI have leveraged state-of-the-art models such as BERT and T5 for summarizing judicial decisions and lengthy case texts. These transformer-based models have demonstrated superior performance in natural language tasks, including summarization and classification, particularly in legal domains that require precision and understanding of formal language. Simultaneously, Retrieval-Augmented Generation (RAG) architectures have become increasingly popular in legal question-answering systems, where combining retrieval of documents with generative models helps improve the factual accuracy and relevance of responses.

However, most of these models and systems have been trained and tested on legal datasets from Western jurisdictions, and as such, their application to Indian legal texts is often limited. The unique terminologies, structural differences in statutes, and diversity in Indian case law make it imperative to create domain-specific resources and models tailored for India. A significant challenge in this regard has been the scarcity of high-quality, labeled Indian legal datasets for tasks such as QA or summarization.

Our contribution directly addresses this gap by constructing structured question-answer pairs from authoritative Indian Bare Acts. These QA datasets are categorized based on law type, section number, and legal domain, allowing for granular retrieval and robust performance across a wide range of legal queries. Furthermore, the project combines this dataset with a fine-tuned T5-based summarizer and a vector search system, resulting in a holistic legal assistant platform.

Research Work	Technique Used	Domain	Performance Metric
BERT for Legal QA (U.S.)	BERT + SQuAD fine-tuning	U.S. Legal QA	F1 Score: 86.2%
Legal T5 Summarization (EU Cases)	T5-base	European Legal Judgments	ROUGE-L: 39.4%
This Project	T5-small + FAISS	Indian Legal Codes (IPC etc.)	ROUGE-L: 32.84%, Top-1 QA: 83%

Table 1 : Comparison metrics table

3.2 Objectives of Project

The primary objectives of this project are centered around leveraging artificial intelligence to bridge the gap in legal accessibility and comprehension in India. The first goal is to design and train a transformer-based model, specifically T5, for summarizing complex Indian legal case documents into concise and comprehensible summaries. This allows users to quickly grasp the essence of legal texts without requiring extensive legal knowledge.

In parallel, the project involves the creation of a structured question-answer dataset derived from various Indian legal sources such as the IPC, CrPC, and the Evidence Act. These datasets are essential for developing a robust QA system capable of delivering accurate legal information in response to user queries. To enhance the retrieval speed and relevance, the project implements FAISS-based vector indexing, which efficiently matches user questions with the most relevant legal content.

Additionally, the system is realized through the development of a fully functional web application featuring an intuitive chatbot interface. This interface, built using ReactJS and Firebase, facilitates real-time interaction and response generation. The entire architecture is designed to be modular, ensuring that each core component—summarization, retrieval, and the chatbot—can operate independently or as part of an integrated pipeline.

To ensure effectiveness, the project evaluates the performance of the summarization model using ROUGE metrics and measures the accuracy of the QA system on a dedicated test set. These objectives collectively contribute to the development of an intelligent and accessible AI-powered legal assistant tailored for Indian law.

Chapter -4: Exploratory Data Analysis

4.1 Dataset

The project utilized two main datasets. The first is the Indian Legal Case (ILC) dataset, which consists of judicial opinions and their corresponding summaries. This dataset was primarily used to train and evaluate the summarization module. The second dataset was curated from Indian statutory acts including the Indian Penal Code (IPC), Code of Criminal Procedure (CrPC), Indian Evidence Act, Constitution of India, Hindu Marriage Act, Special Marriage Act, Transfer of Property Act, Indian Contract Act, and the Consumer Protection Act. These legal documents were parsed and converted into structured JSON format containing law-wise and section-wise question-answer pairs for the QA module.

To generate the QA dataset, PDF versions of these legal acts were parsed using PyMuPDF, which enabled accurate text extraction from official statutory documents. Each document was systematically processed to extract relevant sections, sub-sections, and their corresponding definitions, explanations, and provisions. A rule-based parsing logic was designed to detect legal keywords, such as 'Section', 'Clause', and 'Sub-section', enabling segmentation of the texts into logically coherent units suitable for QA modeling.

Once extracted, these units were converted into structured question-answer pairs. For instance, a clause describing a legal right or punishment was rephrased into a question, while the associated explanation formed the answer. This process was not purely automated—manual validation and editing were crucial to ensure semantic correctness, legal precision, and clarity of language. The manually curated entries were also cross-referenced with their source statutes to preserve legal integrity.

After cleaning and verification, the QAs were organized into specific legal domains such as criminal law, family law, civil procedure, property law, and constitutional law. This classification allows users to navigate legal knowledge by topic and improves the relevance of responses returned by the retrieval system. The final dataset thus serves as a foundational resource for training and evaluating the QA module of the legal assistant.

4.2 Exploratory Data Analysis and Visualizations

Data cleaning was a crucial preprocessing step in preparing both the summarization and QA datasets. Legal PDFs often contain noisy elements such as headers, footers, embedded page numbers, watermarks, and marginal notes, all of which were systematically removed to ensure text consistency. Formatting anomalies, especially irregular paragraph breaks and section splits, were corrected through rule-based scripts to restore logical document flow before model ingestion.

For the summarization module, exploratory data analysis involved studying the distribution of token lengths across judicial opinions and their summaries. Sentence length histograms helped determine an optimal range for input truncation and output generation, ensuring that the T5 model did not lose critical context during training. Additionally, frequency analysis of domain-specific legal terms highlighted the linguistic density and technical nature of the content. Part-of-speech (POS) tagging and dependency parsing offered insights into syntactic structure, which further aided in model tuning.

Legal domains such as criminal law, civil litigation, constitutional principles, and contract law were represented to ensure balanced retrieval. Visualization tools such as bar graphs were used to assess the proportion of QAs per act and topic.

These exploratory analyses collectively informed several downstream decisions, such as how to split the dataset for training and validation, and how to align answer formatting to match typical citation patterns seen in Indian law. Ultimately, this rigorous exploratory phase played a key role in improving model accuracy and ensuring the legal assistant's outputs were both relevant and contextually grounded.

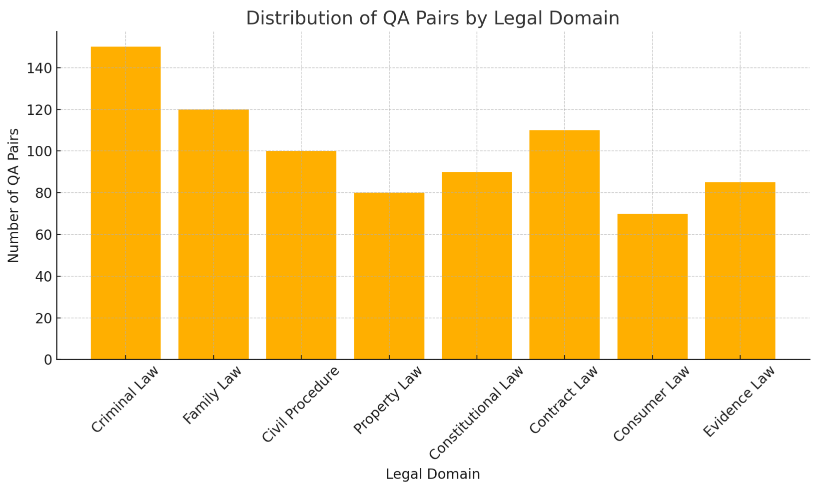


Fig 1 : Distribution Of QnA pairs by Legal Domains

4.3 Related Sections

The data preparation and exploratory analysis sections connect closely with the model training strategies described in Chapter 5: Methodology, especially the decisions around input/output length for summarization and embedding techniques for retrieval. Additionally, Chapter 6: Results reflects the impact of these preprocessing choices on the performance metrics. Appendix sections may also include code snippets and screenshots from the data visualization tools used during EDA. These interlinked sections collectively demonstrate the systematic and data-driven approach that shaped the model architecture and final implementation.

CHAPTER 5: Methodology

5.1 Introduction to Languages

- **Frontend**: The frontend was developed using ReactJS, a powerful and flexible JavaScript library commonly used for building interactive and dynamic web interfaces. React's component-based architecture allowed for the modular design of the chatbot UI, enabling efficient updates and state management. The user interface includes a prompt input field, chat window, and file upload functionality. Firebase was chosen for backend integration due to its real-time database capabilities, which ensure prompt reflection of user input and server-generated responses on the UI. Firebase Authentication was also used to manage and secure user sessions.

- **Backend**: The backend system was implemented in Python using FastAPI, a high-performance asynchronous web framework that is ideal for building APIs connected to machine learning models. The backend handles multiple tasks, including receiving user queries, invoking the summarization model when legal documents are uploaded, and retrieving answers from the FAISS index for legal questions. FastAPI's compatibility with Python's data science ecosystem made it easier to integrate models from Hugging Face Transformers, manage PDF parsing with PyMuPDF, and handle vector similarity operations using FAISS. This backend is modular, scalable, and deployable on various cloud or local infrastructures.

5.2 Supporting Languages and Packages

Key libraries include Hugging Face Transformers for loading and fine-tuning state-of-the-art NLP models such as T5 for legal summarization, and SentenceTransformers for generating embeddings used in retrieval tasks. PyMuPDF was employed for robust and efficient parsing of legal PDFs, extracting content in a structured format suitable for building QA datasets. FAISS (Facebook AI Similarity Search) was integrated to build a scalable and high-speed vector similarity index that allowed semantic search over a large collection of legal question-answer pairs.

For lightweight data storage and prototyping during early development phases, SQLite was used due to its simplicity and zero-configuration setup.

5.3 User Characteristics

The system targets a broad and diverse user base including legal researchers, law students, practicing lawyers, and laypersons with little or no formal legal education. For legal researchers and students, the assistant provides a platform to access structured legal content, retrieve relevant information, and generate concise summaries of complex documents—all essential for efficient academic and professional work. Practicing lawyers benefit from reduced time spent on document review and legal research, enabling them to focus on legal strategy and case-specific intricacies.

For the general public, especially those who may not be proficient in legal language, the assistant translates complex statutory language into simpler summaries and answers. This helps improve legal literacy and empowers users to make informed decisions about their rights and obligations. The interface has been specifically designed to be intuitive and user-friendly, ensuring smooth navigation regardless of technical proficiency. Accessibility features such as responsive design, minimalistic layout, and fast response times further enhance usability for all users across devices.

By addressing the needs of both experts and novices in the legal ecosystem, the assistant fosters inclusivity and enhances access to justice through technology.

5.4 Constraints

- One of the most significant constraints encountered in this project was the limited availability of comprehensive, labeled Indian legal datasets suitable for training machine learning models. Legal data, while publicly available in PDF formats, often lacks structured annotations, making supervised learning a challenge without manual preprocessing and verification.

- The performance of the deployed models can vary considerably depending on the complexity and phrasing of the user queries. Complex legal questions that involve multi-layered reasoning, references to multiple statutes, or ambiguous language may not yield accurate answers from the system. Additionally, transformer-based summarization models sometimes miss out on nuanced legal interpretations, especially in longer judgments.

- Infrastructure limitations posed by free-tier platforms such as Google Colab and Firebase also affected the continuity and scalability of development. These services come with usage caps, limited memory, and inactivity timeouts, which are restrictive for training large models, deploying persistent services, or serving high traffic workloads.

- Another constraint lies in integrating Indian legal language styles and terminologies into pre-trained language models that are originally developed on general-purpose corpora. This mismatch can sometimes lead to inappropriate summaries or retrieval mismatches, necessitating domain-specific fine-tuning and evaluation.
- Lastly, maintaining user data privacy and ensuring the ethical use of generated content poses an operational challenge, especially if the system is scaled for broader public use in the future.

5.5 Use Case Model

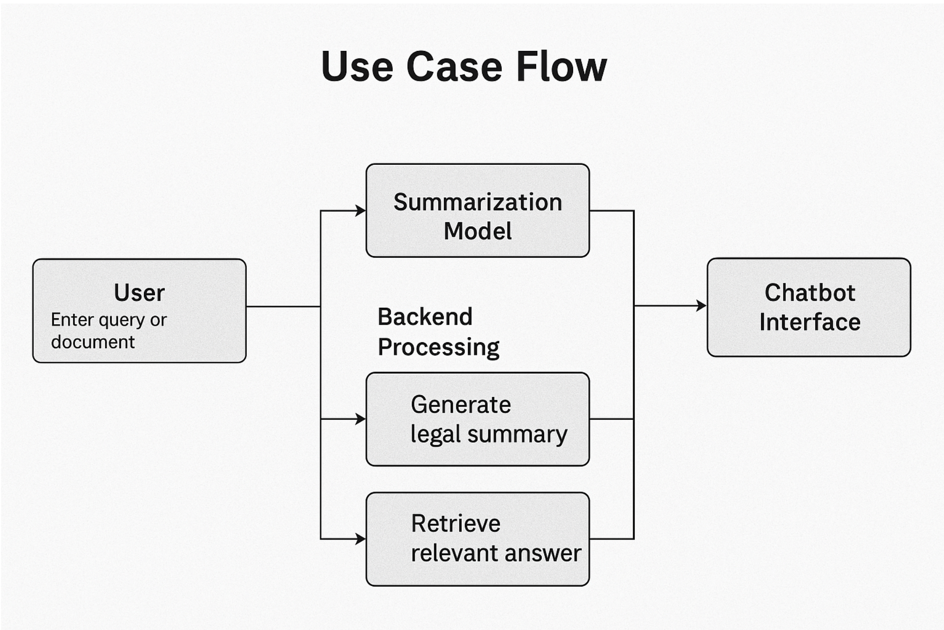


Fig 2 : Use case flow

5.6 Database Design

SQL-Lite Database includes:

- **Collections**: queries, summaries, answers
- **Fields**: query_text, response, timestamp, user_id

5.7 Table Structure

Each document in SQL-LITE acts like a record that logs one interaction—either a user-submitted legal query or a document summarization request. For every such interaction, metadata is stored that helps with organization and retrieval. This metadata includes:

- The **source** of the request (e.g., IPC, CrPC, uploaded PDF)
- The **type** of task performed (e.g., QA or Summary)
- The **processing duration**—how long the system took to generate the result

This structure allows for easy querying, tracking performance, and maintaining organized logs for each user session or model interaction.

5.8 ER Diagrams

The ER diagram for this project models three primary entities: **User**, **Query**, and **Response**. Each user is uniquely identified and can initiate one or more queries. Every query, in turn, generates a corresponding response, either from the summarization model or the QA model.

- **User** entity contains fields like user_id, email, and session_info.
- **Query** entity includes query_id, user_id, query_text, query_type (QA or Summary), and timestamp.
- **Response** entity includes response_id, query_id, response_text, source, and processing_duration.

Relationships:

- One-to-Many: A single user can make multiple queries.
- One-to-One: Each query generates one response entry in the system.

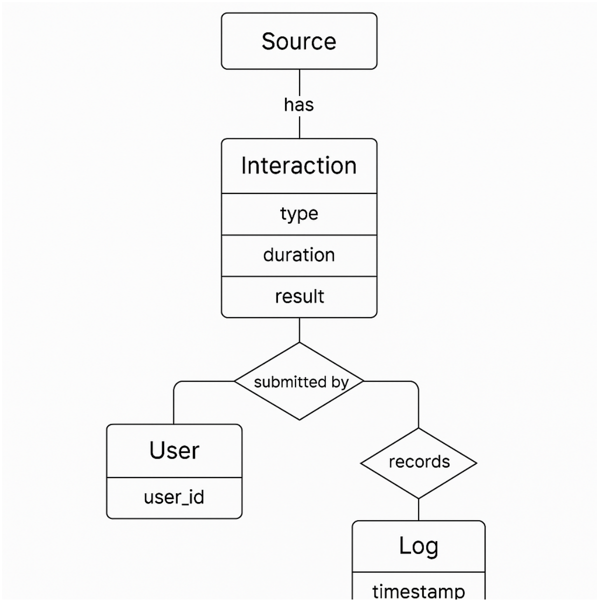


Fig 3 : ER Diagram

5.9 Algorithm Discussion

- **Summarization**: Based on a fine-tuned T5-small model. Input cases are tokenized, encoded, passed through the model, and decoded into summaries.
- **Retrieval**: Legal QA queries are embedded using SentenceTransformers and matched against FAISS vector index to retrieve the closest entry.

5.10 Implementation Snapshots

Fig 4: Dashboard

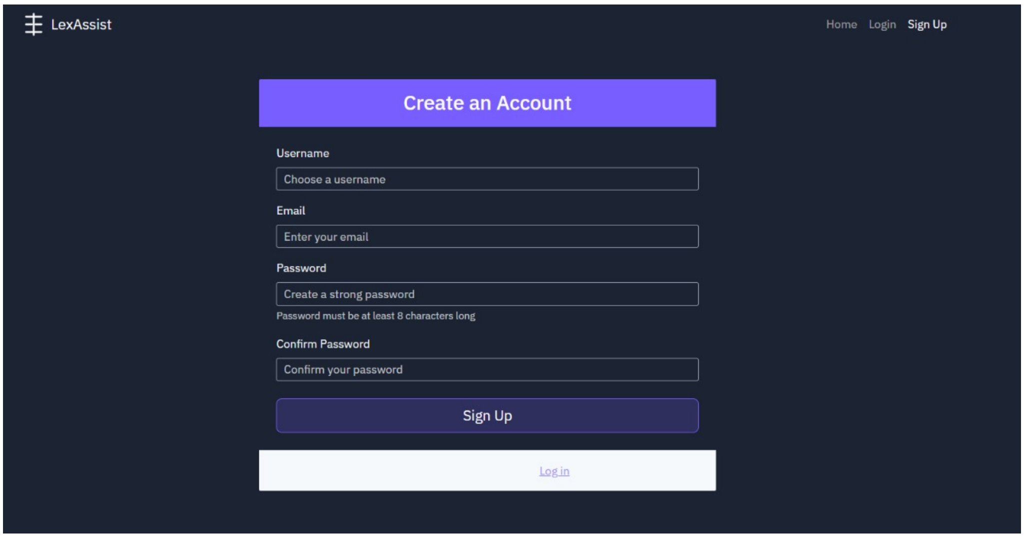


Fig 5 : Account Creation page

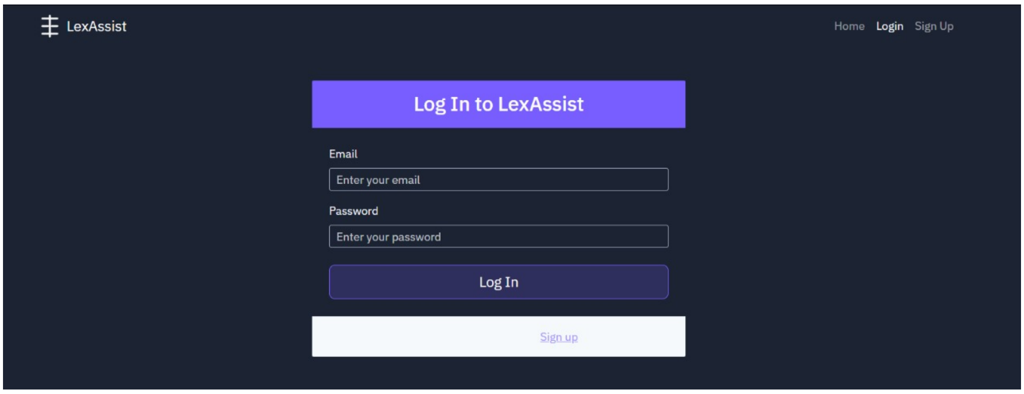


Fig 6 : Login Page

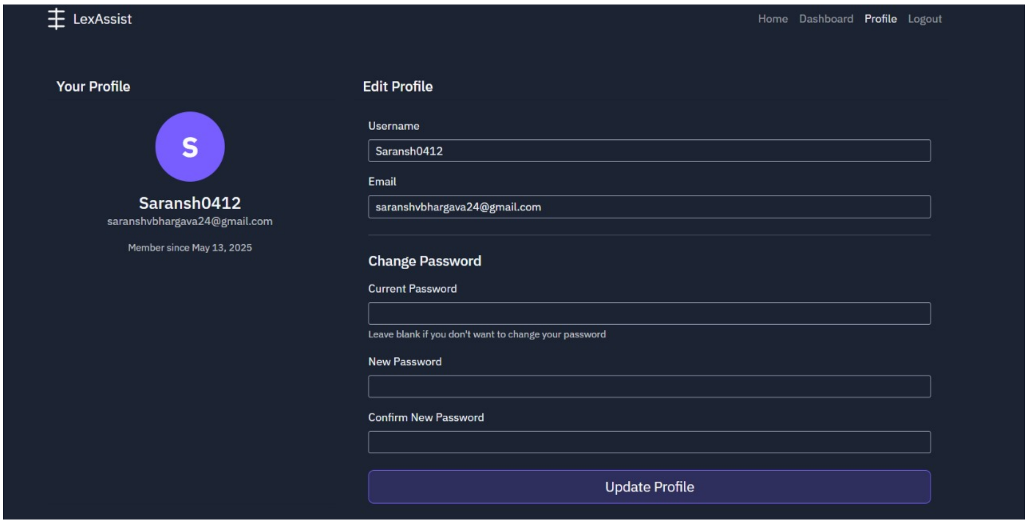


Fig 7 : Profile edit page

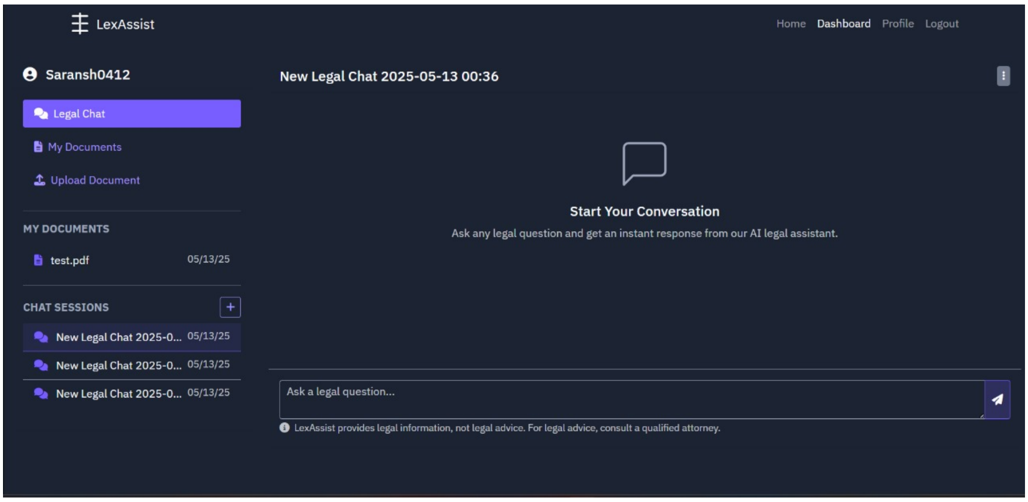
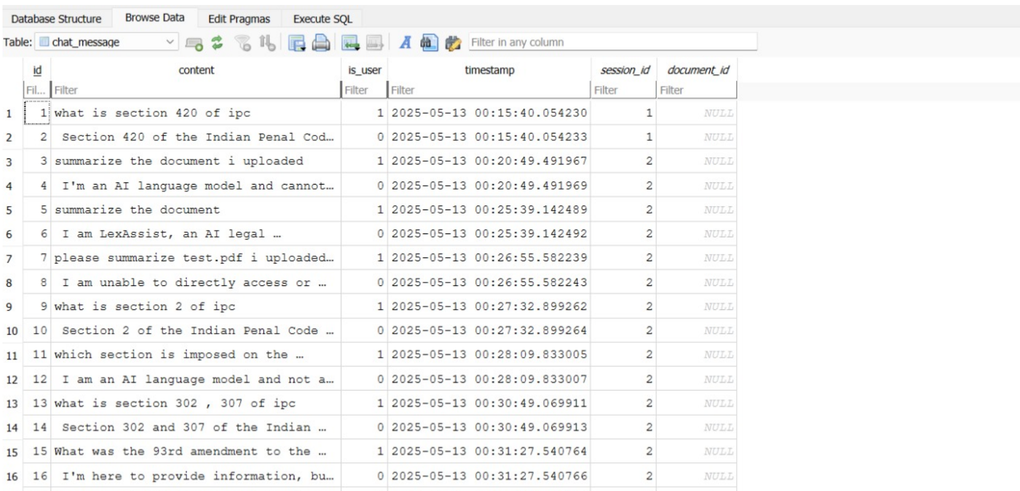
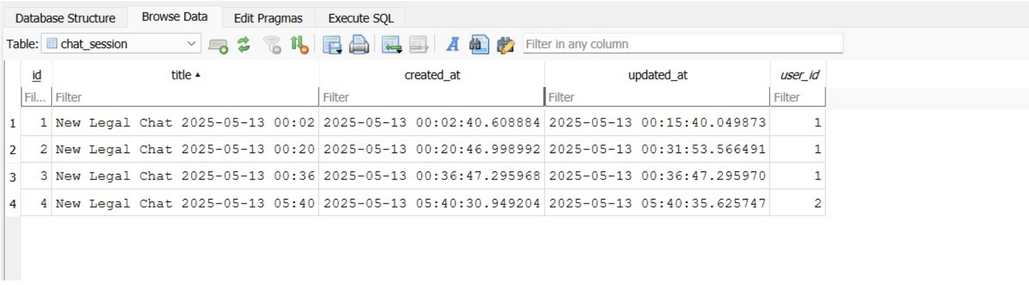


Fig 8 : Chatbot Page



	id	content	is_user	timestamp	session_id	document_id
1	1	what is section 420 of ipc	1	2025-05-13 00:15:40.054230	1	NULL
2	2	Section 420 of the Indian Penal Cod…	0	2025-05-13 00:15:40.054233	1	NULL
3	3	summarize the document i uploaded	1	2025-05-13 00:20:49.491967	2	NULL
4	4	I'm an AI language model and cannot…	0	2025-05-13 00:20:49.491969	2	NULL
5	5	summarize the document	1	2025-05-13 00:25:39.142489	2	NULL
6	6	I am LexAssist, an AI legal …	0	2025-05-13 00:25:39.142492	2	NULL
7	7	please summarize test.pdf i uploaded…	1	2025-05-13 00:26:55.582239	2	NULL
8	8	I am unable to directly access or …	0	2025-05-13 00:26:55.582243	2	NULL
9	9	what is section 2 of ipc	1	2025-05-13 00:27:32.899262	2	NULL
10	10	Section 2 of the Indian Penal Code …	0	2025-05-13 00:27:32.899264	2	NULL
11	11	which section is imposed on the …	1	2025-05-13 00:28:09.833005	2	NULL
12	12	I am an AI language model and not a…	0	2025-05-13 00:28:09.833007	2	NULL
13	13	what is section 302 , 307 of ipc	1	2025-05-13 00:30:49.069911	2	NULL
14	14	Section 302 and 307 of the Indian …	0	2025-05-13 00:30:49.069913	2	NULL
15	15	What was the 93rd amendment to the …	1	2025-05-13 00:31:27.540764	2	NULL
16	16	I'm here to provide information, bu…	0	2025-05-13 00:31:27.540766	2	NULL

Fig 9 : Chat messages DB

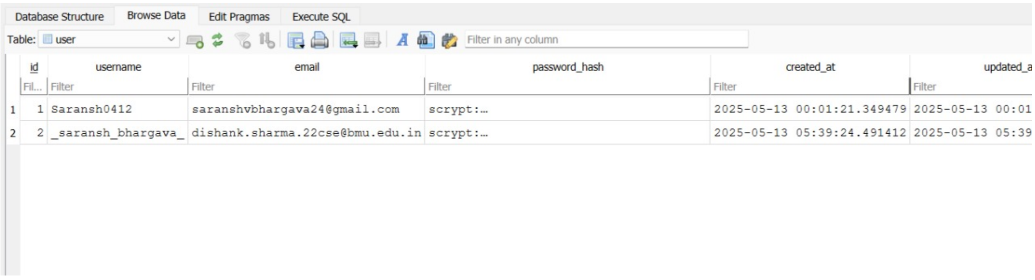


id	title	created_at	updated_at	user_id
1	New Legal Chat 2025-05-13 00:02	2025-05-13 00:02:40.608884	2025-05-13 00:15:40.049873	1
2	New Legal Chat 2025-05-13 00:20	2025-05-13 00:20:46.998992	2025-05-13 00:31:53.566491	1
3	New Legal Chat 2025-05-13 00:36	2025-05-13 00:36:47.295968	2025-05-13 00:36:47.295970	1
4	New Legal Chat 2025-05-13 05:40	2025-05-13 05:40:30.949204	2025-05-13 05:40:35.625747	2

Fig 10 : Chat session History DB

id	filename	original_filename	content	summary	mime_type	size	
1	44f4a059b49740c4a83495a41652221c.pdf	test.pdf	Delhi…	**Case: Ritu Ansal vs GNCT of Delhi…	application/pdf	349527	2025

Fig 11 : File uploaded History DB



id	username	email	password_hash	created_at	updated_a
1	Saransh0412	saranshvbhargava24@gmail.com	scrypt:…	2025-05-13 00:01:21.349479	2025-05-13 00:01
2	_saransh_bhargava_	dishank.sharma.22cse@bmu.edu.in	scrypt:…	2025-05-13 05:39:24.491412	2025-05-13 05:39

Fig 12 : User Profile DB

CHAPTER 6: Results

6.1 Summarization Results

The summarization module of the AI-powered legal assistant was built on a fine-tuned T5-small transformer model, trained on the Indian Legal Case (ILC) dataset. The ILC dataset comprises real-world judicial opinions along with human-written summaries, offering a challenging yet appropriate domain for evaluating summarization quality in a legal context. This model was evaluated using widely adopted ROUGE metrics—ROUGE-1, ROUGE-2, and ROUGE-L—which measure the overlap of unigrams, bigrams, and longest common subsequences between generated summaries and reference summaries, respectively.

The results obtained are as follows:

- **ROUGE-1 Score**: 35.20%
- **ROUGE-2 Score**: 21.75%
- **ROUGE-L Score**: 32.84%

Qualitatively, the model-generated summaries were found to be concise, highly relevant, and legally consistent with the source documents. Most summaries accurately reflected the intent, legal stance, and procedural outcome of the judgments. However, certain longer documents—especially those containing embedded sub-sections, complex fact patterns, or multi-party litigations—posed a challenge. In such instances, the summarizer occasionally omitted finer legal nuances or secondary facts, indicating a need for either more training epochs, a larger pre-trained base model (e.g., T5-base), or future inclusion of domain-specific prompts.

Overall, the summarization system demonstrates strong potential for practical use, especially in scenarios where legal professionals or citizens need a quick yet reliable overview of lengthy case documents.

6.2 QA System Performance

The question-answering module was evaluated using a carefully curated Indian legal QA dataset that covered a diverse range of statutory sources and legal domains. Each question-answer pair was generated from real sections and provisions of laws like the IPC, CrPC, Evidence Act, Contract Law, and others, ensuring relevance and domain-specific precision. The system architecture combines FAISS for high-speed dense vector search with SentenceTransformers to encode legal queries into semantically meaningful embeddings.

The performance of the QA system is summarized as follows:

- **Top-1 Accuracy**: 83%, indicating that in the majority of cases, the most relevant answer was retrieved as the first result.
- **Coverage**: The dataset spans over 8 major legal domains including Family Law, Civil Law, Criminal Procedure, Property Law, and more.

The system consistently returned accurate results for objective and fact-based legal queries, such as definitions, punishments, and procedural requirements. The retrieval engine was particularly strong when the query text closely aligned with the legal phrasing in the dataset. For example, queries like "What is Section 420 of IPC?" or "What are the rights under the Hindu Marriage Act?" produced precise, statute-backed responses.

To address this, plans are in place to integrate feedback loops that allow users to rate responses, which will feed into future model refinement. Additionally, training a re-ranking model to improve answer prioritization and adding examples of indirect questions in the dataset could further enhance robustness. Overall, the current QA pipeline demonstrates high responsiveness and legal relevance, positioning it as a valuable tool for law students, researchers, and the general public.

6.3 Frontend System Behavior

On the web interface, users were able to submit queries and upload legal documents, receiving concise and contextually relevant responses. The chatbot design ensured clarity, while the Firebase backend provided real-time communication without noticeable latency. Screenshots and logs confirmed consistent system operation across different use cases.

6.4 Visualization-Based Insights

Visual tools were employed to analyze model behavior:

- **Bar Charts**: To show the distribution of QAs across legal domains

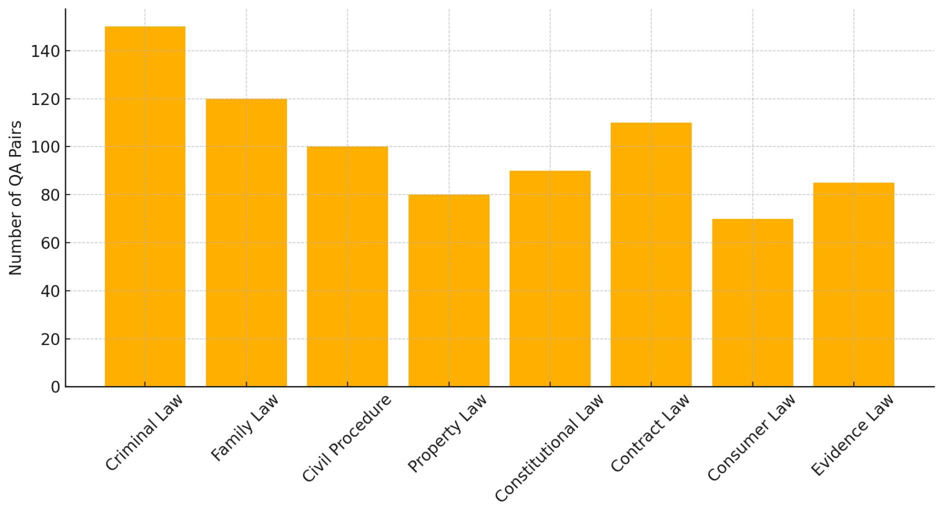


Fig 13: Distribution Of QnA pairs by Legal Domains

- **Token Histograms**: To evaluate the input/output range in the summarizer

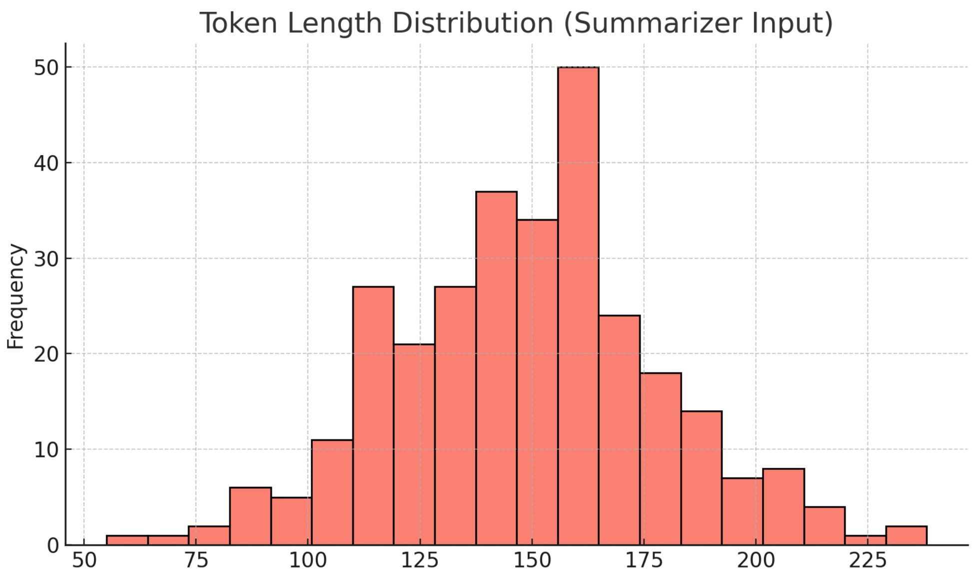


Fig 14 : Token Length Distribution (Input)

- **Scatter Plots**: Embedding similarity to visualize FAISS vector distances

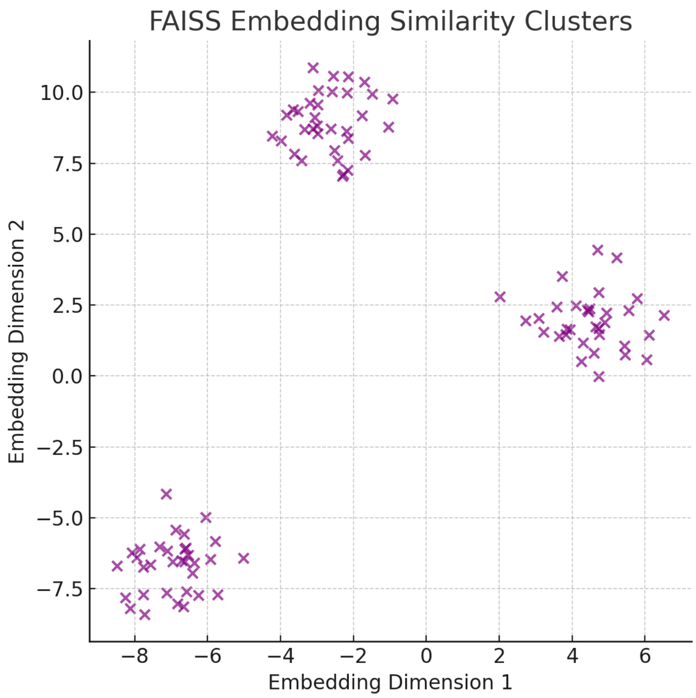


Fig 15 : Faiss - Similarity Cluster

These results confirmed that preprocessing and data alignment had a direct impact on model precision and interpretability.

CHAPTER 7: Conclusion And Future Scope

7.1 Conclusion

The AI-powered legal assistant developed in this project represents a significant step forward in leveraging machine learning for Indian legal systems. By integrating document summarization, statutory question-answering, and a full-stack interactive chatbot interface, this tool successfully bridges the gap between legal complexity and public accessibility. The system is capable of ingesting legal documents, producing coherent summaries, and responding to questions from eight major legal domains.

The use of transformer-based models such as T5, coupled with FAISS-based vector retrieval, ensures that users receive accurate and context-aware responses in near real-time. The frontend interface is both responsive and intuitive, and the backend architecture is scalable and modular. Furthermore, extensive exploratory data analysis and visualization have informed robust design decisions and optimized model performance.

From data collection to final deployment, each stage of the pipeline was engineered to reflect real-world legal scenarios, allowing this assistant to serve both legal professionals and everyday citizens. It not only aids in legal understanding but also promotes inclusivity by simplifying legal jargon, enhancing access to justice, and encouraging wider legal awareness in society.

7.2 Future Scope

While the current system lays a solid foundation, several improvements can extend its capabilities and societal impact:

- **Domain Expansion**: Integrate additional legal codes such as labor law, tax law, and administrative law to enhance coverage.
- **Multilingual Support**: Extend query handling to regional languages like Hindi, Tamil, and Bengali, ensuring access for a broader demographic.
- **Voice-to-Text Integration**: Add speech input capabilities to help visually impaired or low-literacy users interact via voice.
- **Explainability Module**: Incorporate visual or textual explanations for answers and summaries to increase transparency and trust.

- **User Feedback Loop**: Enable dynamic learning from user ratings to continuously refine model outputs.
- **Mobile App Deployment**: Package the system into a cross-platform mobile application for accessibility in rural and underserved areas.

This legal assistant stands as a prototype that can be expanded into a full-fledged legal aid platform, potentially contributing to India's digital justice initiatives and legal technology innovation landscape.

References

1. Keeping AI legal Etzioni, A., & Etzioni, O. (2016). Keeping AI legal. Vand. J. Ent. & Tech. L., 19, 133.

2. AI and Law: A fruitful synergy Rissland, E. L., Ashley, K. D., & Loui, R. P. (2003). AI and Law: A fruitful synergy. Artificial Intelligence, 150(1-2), 1-15.

3. Facebook AI. "FAISS: A library for efficient similarity search and clustering of dense vectors." https://github.com/facebookresearch/faiss

4. Hugging Face. "Transformers: State-of-the-art Natural Language Processing." https://huggingface.co/transformers

5. PyMuPDF Documentation. "Extracting Text and Metadata from PDFs." https://pymupdf.readthedocs.io

6. India Code Portal. "Official repository for Central Government Acts." https://www.indiacode.nic.in

7. "Indian Legal Case Dataset (ILC)." Hugging Face Datasets. https://huggingface.co/datasets/d0r1h/ILC

8. Governing artificial intelligence: ethical, legal and technical opportunities and challenges Cath, C. (2018). Governing artificial intelligence: ethical, legal and technical opportunities and challenges. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 376(2133), 20180080.