A PROJECT REPORT
ON

AI POWERED LEGAL DOCUMENTATION ASSISTANT

Submitted by,

SWARNA LOHIT - 20211CSD0052

SANJAY S - 20211CSD0050

MANOJ M - 20211CSD0199

Under the guidance of,

Ms. SANDHYA L

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING (DATA SCIENCE)

AT

PRESIDENCY UNIVERSITY

BENGALURU

MAY 2025

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Project report **"AI Powered Legal Documentation Assistant"** being submitted by "SWARNA LOHIT, SANJAY S, MANOJ M" bearing roll number "20211CSD0052, 20211CSD0050, 20211CSD0199" in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out under my supervision.

Ms. SANDHYA L
Assistant Professor
PSCS
Presidency University

Dr. SAIRABANU ATHAM
Professor & HoD
PSCS
Presidency University

Dr. MYDHILI NAIR
Associate Dean
PSCS
Presidency University

Dr. SAMEERUDDIN KHAN
Pro-VC -Engineering
Dean –PSCS / PSIS
Presidency University

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

I hereby declare that the work, which is being presented in the report entitled "**AI Powered Legal Documentation Assistant"** in partial fulfillment for the award of Degree of **Bachelor of Technology** in **Computer Science and Engineering**, is a record of my own investigations carried under the guidance of **Ms**. **Sandhya. L, Assistant Professor, Presidency School of Computer Science and Engineering, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

NAME(S)	**ROLL NO(S)**	**SIGNATURE(S)**
SWARNA LOHIT	20211CSD0052	
SANJAY S	20211CSD0050	
MANOJ M	20211CSD0199	

ABSTRACT

The process of legal documentation remains a time-consuming and error-prone task for legal professionals, despite its critical role in practice. This study explores the development of an AI-powered legal documentation assistant designed to streamline drafting, review, and management of legal documents. By fine-tuning large language models (LLMs) on specialized legal datasets such as LegalBench and ContractNLI, the assistant aims to understand complex legal language with greater accuracy. Results suggest significant improvements, with potential time savings of up to 90% and a notable reduction in drafting errors compared to manual methods. However, the integration of AI into legal workflows raises important concerns around bias, data security, and the legal validity of AI-generated documents. This research emphasizes that while AI can substantially improve efficiency and consistency, human oversight remains indispensable to safeguard legal integrity. Future efforts should focus on enhancing the transparency of AI outputs, minimizing risks of bias, and ensuring that these tools are responsibly embedded within legal practice. In addition, the study highlights the importance of rigorous validation strategies, including both benchmark evaluations and expert human review, to maintain professional standards. Addressing these challenges is essential not only for widespread adoption but also for maintaining public trust in AI-assisted legal processes.

ACKNOWLEDGEMENT

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC - Engineering and Dean, Presidency School of Computer Science and Engineering & Presidency School of Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Dean **Dr. Mydhili Nair,** Presidency School of Computer Science and Engineering, Presidency University, and "**Dr. Saira Banu Atham**", Head of the Department, Presidency School of Computer Science and Engineering, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Ms. Sandhya L,** Presidency School of Computer Science and Engineering, Presidency University for her inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the internship work.

We would like to convey our gratitude and heartfelt thanks to the PIP4001 Internship/University Project Coordinator **Mr. Md Ziaur Rahman and Dr. Sampath A K,** department Project Coordinators "**Dr. Manjula H M**" and Git hub coordinator **Mr. Muthuraj.**

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

Swarna Lohit

Sanjay S

Manoj M

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENTS

Chapter 1
INTRODUCTION

Legal professionals often spend significant time drafting, reviewing, and managing complex documents like contracts and case law. These tasks, while critical, are time-intensive and susceptible to human oversight. This study explores the development of an AI-driven legal assistant designed to ease the burden of legal documentation using advanced Natural Language Processing (NLP) and Machine Learning (ML) techniques. The proposed system is built on pre-trained language models such as Legal-BERT, enhanced through fine-tuning and integrated with tools tailored for summarizing documents, identifying legal clauses, and answering user queries. To support this, the system architecture includes a document processing pipeline, a core AI engine that utilizes retrieval-augmented generation, and a user-friendly interface intended for legal professionals.

Our review of existing legal NLP work reveals key challenges, including the complexity of legal language, limited access to high-quality legal datasets, and the need for greater accuracy in AI outputs. Through experiments on benchmark datasets like Bill Sum, Contract NLI, and LegalBench, we found that the AI assistant can significantly speed up legal document review and summarization tasks while maintaining a high level of accuracy. Performance was measured using industry-standard metrics such as ROUGE and F1, and additional analysis pointed to meaningful efficiency gains for legal teams. We also examine ethical and practical concerns, including issues of algorithmic bias, data privacy, and ensuring the legal reliability of AI-generated outputs. The study concludes by outlining practical deployment strategies and future directions, focusing on improving transparency, fairness, and integration into daily legal workflows.

1.1 Background

The broader vision behind this work is to modernize courtroom processes through automation and intelligent systems, ultimately contributing to the evolution of "smart courts." As part of this initiative, we are developing an AI-assisted legal documentation tool that can support both legal professionals and related sectors such as healthcare, particularly for legal compliance in medical environments. While several AI models have been developed within the legal domain, most existing tools lack accessible, intuitive user interfaces and often provide only limited, generalized functionality. These systems are not designed with end-users in mind, especially non-experts who may struggle to extract meaningful insights from legal documents. This project focuses on the development of an AI-powered legal documentation assistant designed to transform how legal professionals handle contracts, case law, and other legal documents. The system architecture will consist of several integrated components: a document ingestion pipeline for processing input files, an AI engine capable of understanding and generating legal content, and a user interface that allows seamless interaction for both legal experts and non-expert users.

Key functionalities of the assistant will include the ability to:

- Answer legal questions based on the content of uploaded documents,
- Summarize lengthy legal texts,
- Identify and classify specific legal clauses,
- Provide relevant case references and legal precedents,
- Stay up-to-date with legal developments through cloud integration and real-time updates.

Through this intelligent system, we aim to support legal practitioners in making faster, more accurate decisions while reducing the time and cost associated with manual legal documentation. The project also takes into account ethical considerations such as bias, data privacy, and the legal reliability of AI-generated output.

1.2 Motivation

This research is driven by the recognition that the traditional process of reviewing legal contracts is both tedious and inefficient. Legal professionals often spend extensive hours sometimes weeks meticulously combing through lengthy documents. This manual effort is not only time-consuming but also increases the likelihood of human error. By introducing automation, our goal is to significantly ease the burden on legal teams, reduce turnaround times, and deliver cost savings for businesses. An intelligent legal assistant could take over the repetitive task of contract analysis, allowing professionals to focus on strategic or higher-value legal work. The system’s value can be further enhanced by incorporating legal data specific to Indian law, including statutory updates, court judgments, and legal precedents. By integrating real-time case law, court decisions, even market analytics, the tool would empower users with reliable and up-to-date information to make smarter business and legal decisions. Additionally, as the AI learns from user input and legal databases over time, its accuracy and usefulness improve continuously. Coupling this learning capability with cloud computing and real-time legal updates ensures the documents stay aligned with evolving legal standards. This not only helps prevent errors but also fosters a more consistent and streamlined contract review process.

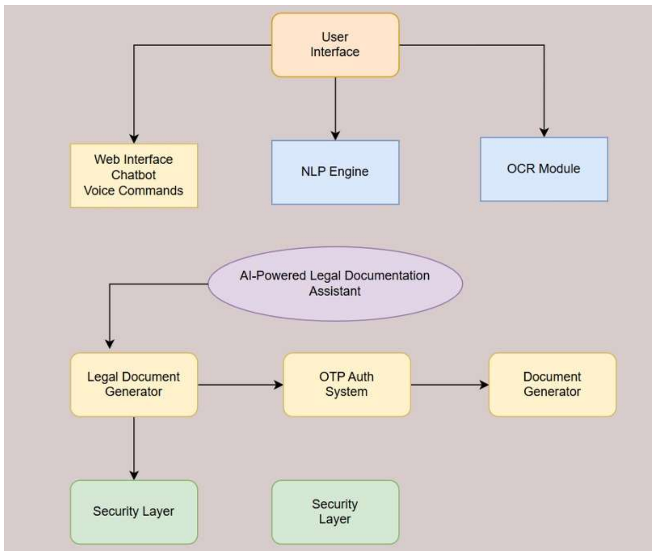


Figure 1.1 System Architecture of AIDA
(Artificial Intelligence-powered Legal Documentation Assistants)

1.3 Problem Statement

Legal documentation, especially contract review and interpretation, remains a highly manual, time-intensive, and error-prone process. Despite the availability of AI models in the legal domain, current tools often lack the user-friendliness, contextual understanding, and real-time responsiveness needed for practical use in courtroom or professional settings. Most of these systems are designed with limited customization and no intuitive interface, making them inaccessible for everyday users and inefficient for legal professionals. Legal practitioners regularly face the burden of reviewing lengthy contracts, identifying key clauses, ensuring compliance, and providing sound legal advice under tight deadlines. This repetitive workload not only leads to fatigue but also increases the risk of oversight and inconsistent judgment.

Additionally, there is a lack of systems that are trained specifically on jurisdiction-specific laws—such as Indian statutes—and capable of incorporating ongoing legal changes.

Furthermore, there's a significant gap in AI solutions that can interact directly with users, answer legal queries, and assist in bridging communication between clients and experts. Without intelligent automation, the legal sector continues to lag behind in digital transformation compared to industries like finance or healthcare. This project aims to solve these challenges by developing an AI-assisted legal documentation system equipped with natural language capabilities, legal knowledge, and an interactive user interface. The goal is to streamline the review process, enhance accuracy, and provide legal insights more efficiently—all while supporting continuous learning and real-time legal updates.

1.4 Scope

The scope of this project is centred around building a smart, AI-assisted legal documentation system that can support legal professionals, judicial staff, and even non-legal users in managing and understanding legal documents more efficiently. The solution will cater primarily to tasks related to contract analysis, legal query handling, and summarization of case law, with a particular focus on the Indian legal framework.

The system will be capable of:

- Extracting and classifying clauses from legal contracts,
- Summarizing complex legal texts into simplified, understandable language,
- Responding to user queries based on the content of legal documents,
- Referring to relevant case law and legal precedents,
- Continuously updating its knowledge base with new laws and rulings through cloud integration.

This assistant will be built on robust NLP models such as Legal-BERT and fine-tuned on legal datasets like ContractNLI, BillSum, and Indian law corpora. The user interface will be designed for ease of use, making it accessible to both seasoned legal practitioners and individuals with limited legal knowledge.

However, the scope excludes delivering final legal advice or replacing legal professionals. The assistant is intended to serve as a support tool to augment legal workflows, not to offer legal representation or judgment. Future phases may involve multilingual support, voice interaction, and integration with court management systems.

Chapter 2
LITERATURE SURVEY

Over the past decade, both researchers and businesses have shown increasing interest in applying artificial intelligence to the legal industry. Early efforts in legal informatics focused on expert systems like the American MYCIN and the UK's LOIS (Legal Online Information System), which tried to simulate legal reasoning through rule-based logic. However, these systems struggled with the complexity and ambiguity of natural legal language, limiting their effectiveness. The emergence of machine learning (ML) and natural language processing (NLP) has ushered in a new wave of more adaptable and sophisticated AI tools for the legal domain. According to studies by Ashley (2017) and Katz et al. (2014), machine learning algorithms can now be trained on vast datasets of legal texts and case law to assist with decision-making and predictive legal analytics. This evolution has paved the way for AI applications in areas like document automation, legal research, and contract analysis. In particular, recent developments in NLP technologies, such as BERT and GPT , have been instrumental in tasks like legal entity recognition, document summarization, and generating contextually accurate legal content.

For instance, research by Zhong et al. (2020) explored how neural-symbolic systems can combine logical reasoning with deep learning to support legal argumentation and compliance analysis. Similarly, Chalkidis et al. (2020) demonstrated the effectiveness of transformer-based models in accurately classifying legal documents. On the commercial side, platforms like ROSS Intelligence, Kira Systems, and Legal Robot are transforming legal workflows by automating contract review, identifying potentially risky clauses, and ensuring legal compliance significantly easing the burden on legal professionals. Despite these advancements, challenges remain. Issues such as data privacy, algorithmic bias, lack of explainability, and the need for domain-specific training data continue to present hurdles. As Surden (2019) notes, while AI can enhance legal tasks, it still falls short of replicating the nuanced judgment and ethical decision-making that experienced legal professionals provide. In summary, the evolution of AI in the legal field reflects a shift from rigid expert systems to more flexible, learning-driven technologies. These innovations are reshaping the legal landscape and contributing to more efficient, accessible, and intelligent legal services.

2.2 Evolution of AI in Law (NLP, ML)

In recent years, artificial intelligence (AI) has increasingly been applied within the legal sector, with significant developments in tasks such as document summarization, information extraction, and contract analysis. This overview presents a summary of recent advancements in these areas, including commonly used natural language processing (NLP) models, legal-specific datasets, and tools tailored for legal professionals.

- **Legal Text Summarization:**

Legal summarization focuses on condensing complex legal texts—such as court rulings, legislative bills, and statutes—into more digestible summaries. This is vital for both legal professionals and laypersons. According to a comprehensive review by Akter et al., which analysed over 120 publications on legal summarization, effective methods often rely on abstractive summarization to convey the intended meaning, rather than merely extracting sentences from the source. Among the datasets widely used are BillSum, which contains U.S. congressional bills paired with summaries (Kornilova and Eidelman, 2019), and EUR-Lex-Sum, which focuses on European Union legislation. Additionally, court judgments from the Indian Supreme Court have supported the creation of the **IN-ABS** and **IN-EXT** datasets for abstractive and extractive summarization, respectively. Modern approaches typically use transformer-based models. For instance, researchers like Liu et al. have fine-tuned models such as BART and GPT on legal texts to generate case summaries, achieving strong results based on ROUGE metrics. However, conventional metrics like ROUGE may be inadequate in the legal context, as they prioritize surface-level text overlap and often fail to account for paraphrasing and dense legal references.

- **Information Extraction and Clause Analysis**

Another prominent application of AI in the legal field is the extraction of relevant information from documents. This involves identifying entities such as dates, names, and monetary amounts, as well as classifying specific contract clauses and extracting key legal terms like force majeure or liquidated damages. Key datasets include CUAD, which offers labelled clauses from U.S. contracts, and MAUD, focused on Q&A data from merger agreements. Extraction techniques range from traditional rule-based systems to supervised machine learning models. A notable advancement is Legal-BERT, introduced by Chalkidis et al. in 2020. This is a version of BERT further trained on legal corpora, which outperformed general-purpose language models in classifying legal clauses.

- **AI for Drafting and Contract Generation**

AI technologies are increasingly being used to support legal drafting, especially through the retrieval of template clauses and suggestions for standardized legal language. Earlier systems employed case-based reasoning, while current systems are leveraging neural text generation. For instance, companies like Clause.ai use fine-tuned versions of GPT-3 trained on legal agreements to draft clauses. In academic settings, researchers such as Al E-Bagy et al. (2022) trained GPT-2 on a corpus of standardized contracts to generate high-quality NDA templates. A report from the Journal of Law and Technology notes that over 40% of top law firms now integrate AI into tasks like drafting and due diligence. JPMorgan's COIN system is a prime example, using NLP to automatically extract relevant contract sections, drastically cutting down review times. Similarly, a study by LawGeex demonstrated that AI could identify 94% of potential risks in NDAs—outperforming human lawyers, who scored 85%. However, these systems must be carefully designed to ensure the legal enforceability of generated content, often by cross-referencing applicable laws or statutes.

- **Retrieval-Augmented Generation (RAG)**

To enhance the accuracy and reliability of legal AI systems, Retrieval-Augmented Generation has emerged as a powerful technique. RAG combines large language models with external knowledge sources, such as legal databases, to reduce the risk of generating incorrect or irrelevant information. An open-source initiative, Legal Document Assistant, exemplifies this approach by indexing case law and providing relevant excerpts to the language model when answering user queries. This approach grounds the output in real legal texts, minimizing hallucinations. Commercial solutions like LexisNexis and Westlaw Edge now integrate similar features, allowing legal practitioners to interact with legal data in natural language and receive context-aware responses with proper citations.

- **Evaluation and Benchmarks in Legal NLP**

Evaluating AI performance in the legal context is complex. Common metrics—such as ROUGE and BLEU for summarization or accuracy and F1-score for information extraction—may not fully capture legal correctness or the quality of reasoning. Consequently, there is a growing emphasis on human in loop evaluation. Recent studies suggest involving legal experts to assess the factual and legal accuracy of model outputs. Accordingly, for experimental purposes, a combination of automatic metrics and expert legal review is increasingly recommended to ensure that systems meet the rigorous standards required in legal practice.

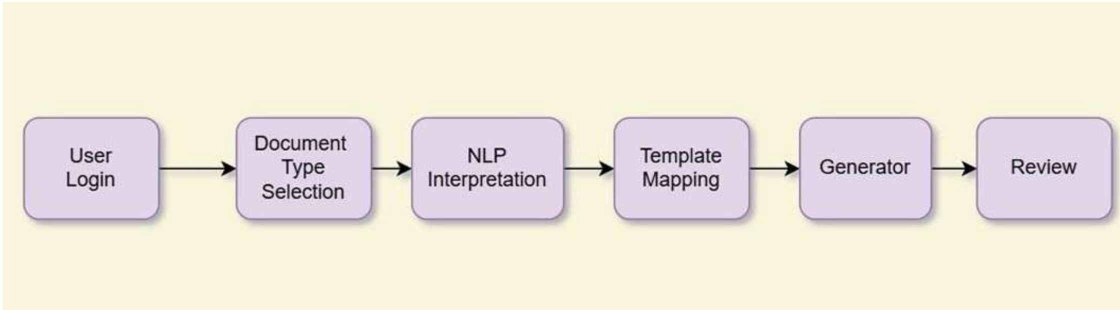


Figure 2.1 Legal Document Creation Workflow creation using AIDA.

The integration of artificial intelligence (AI) into the legal sector has evolved considerably over the decades. In its earliest phase, the primary focus was on digitizing legal information to improve accessibility. Systems like JURIS, introduced in the 1970s, marked a pivotal shift by enabling the electronic storage and retrieval of legal documents. Soon after, major platforms such as LEXIS (now known as LexisNexis) and Westlaw emerged, fundamentally transforming legal research. These tools allowed legal professionals to quickly locate statutes and case law, significantly reducing the reliance on time-consuming manual searches through printed legal volumes. As the field progressed into the late 20th and early 21st centuries, AI applications in law began to move far beyond simple document retrieval. With advancements in computing power and machine learning, legal AI systems started to incorporate predictive analytics and natural language processing capabilities.

Simultaneously, AI began playing a larger role in legal document analysis. Tools such as Kira Systems and Luminance emerged to assist with contract review and due diligence, automatically flagging relevant clauses and extracting critical information. These platforms use machine learning to recognize patterns across thousands of legal documents, streamlining workflows and minimizing the risk of human oversight. The field further advanced with the introduction of legal analytics platforms like Lex Machina, which offered detailed insights into litigation trends, judge-specific patterns, and law firm performance. By transforming raw court data into actionable intelligence, these tools empowered legal professionals to make more informed decisions based on empirical evidence rather than intuition alone.

Task	Task Description	Data Creation Methodology
Issues Generation	Create legal issues for a court based on the case facts. Legal issues are the key points on which the verdict needs to be delivered.	Judgment facts were then sent to gpt-3.5-turbo to generate legal issues.
Argument Generation	Based on the facts of a case, legal issues, and applicable statute names, generate arguments for a given party.	Arguments for the petitioners were created using gpt-3.5-turbo, using judgment facts, generated issues, and relevant statutes as inputs. Counterarguments for defendants were created using petitioners' generated arguments.
Event Timeline	Extract important events and their dates from the input text descriptions and output a chronologically sorted event list with dates and a brief description.	FIRs and judgment facts were used as input text descriptions. These were sent to gpt-3.5-turbo to create event timelines.
Combine Event Timelines	For extraction of event timelines from very long texts, it is often split into chunks, and the event timeline for each chunk is created independently, which is merged later.	Individually created timelines from the same judgment or FIR were merged using gpt-3.5-turbo.
Legalbench	Training data for the legalbench [2] was filtered to keep only the legal reasoning tasks.	ORCA-style explanations of these MCQs were created using GPT4 in a zero-shot setting.
Statute Ingredients	Break the input statute definition into the ingredients needed to apply the statute.	Definitions of the most popular sections of Indian central acts were used to generate the statute ingredients using gpt-3.5-turbo.
Summary Generation	Create a summary in judgment headnotes format using the input court judgment text	Indian Supreme Court judgments from 1950 to 1994 are published with headnotes, which are summaries of those judgments.
Legal Open ORCA	OpenORCA dataset [24] is an ORCA-style explanation of the Natural Instructions dataset.	The Natural Instruction dataset (NI) is filtered for law and matched against the 1M GPT4 openORCA dataset using tf-idf matching to get legal OpenORCA data.
Contract Clause Generation	Generation of new contract clauses and modification of existing contract clauses.	Existing Data[1]
Legal NIv2 MCQ	Natural Instructions v2 data [25] was filtered for law-related questions.	A random sample of NI dataset MCQs filtered for law.
Constitution General Knowledge	Q&A about the Indian Constitution.	Existing data[2]

Table-2.1 Prioritized Legal Tasks

Chapter 3

RESEARCH GAPS OF EXISTING METHODS

To gain deeper insight into the common parameters and recurring challenges in modeling employment contracts and loan agreements, we undertook an in-depth review of 40 academic and technical research papers. Through this analysis, we identified several key gaps in existing approaches and models, which we categorized as follows:

I. Insufficient Contextual Understanding - One of the most significant shortcomings in current systems is their limited ability to grasp the full context in which employment contracts and loan agreements are created and interpreted. Without it, there's a high risk of misinterpretation, which can lead to legal conflicts or the enforcement of unfair terms .For example, in employment contracts, a lack of contextual awareness might result in confusion over job duties, payment structures, or termination policies—potentially leading to disputes between employers and employees. Such gaps might leave borrowers vulnerable to unexpected financial strain or prevent lenders from recovering funds efficiently.

II. Difficulty Handling Diverse Document Formats - Another major challenge is the variety of document formats in which employment and loan contracts are stored—ranging from Word documents and PDFs to scanned images. Each format requires different processing techniques to ensure reliable analysis. One way to address this issue is by using Optical Character Recognition (OCR) technology to convert non-editable formats, like scanned images or PDFs, into machine-readable text. Developing customized parsing tools for specific formats also helps structure and analyze data that might otherwise remain buried in semi structured or unstructured text.

III. Challenges in Semantic Understanding and Inference - True comprehension of legal documents requires not just reading the words but understanding their meaning in context. This includes recognizing relationships between entities, grasping implied obligations, and drawing conclusions based on the text. Advanced NLP techniques such as semantic role labeling, coreference resolution, and semantic parsing enable systems to interpret who is responsible for what, identify references across paragraphs, and structure the data in a way that supports logical reasoning. Semantic inference, for instance, allows an AI system to determine expected repayment terms or contractual obligations based on relevant clauses—even if they're not explicitly stated.

IV. Interpreting Unstructured Legal Text - Contracts and agreements are often written in natural, free-form language rather than structured formats. This presents a significant challenge for systems trying to extract meaningful insights. Unlike structured data, unstructured legal text requires more nuanced interpretation. NLP techniques such as tokenization, part-of-speech tagging, named entity recognition, and sentiment analysis help break down and analyze unstructured content. In parallel, machine learning algorithms can uncover deeper patterns using methods like topic modeling or text classification. However, successful interpretation also depends heavily on domain expertise, which ensures that extracted insights are legally accurate and practically useful. By addressing these gaps, our proposed AI-Based Legal Assistant is designed to offer a comprehensive approach to analyzing and interpreting employment and loan contracts. The goal is to enhance the speed, precision, and accessibility of legal services making legal analysis more efficient and user-friendly for both professionals and lay users.

Model	Aalap is worse	Tie	Aalap is Better
Mistal 7B	37	132	925
gpt-3.5-turbo	385	373	336

Table 3.1 Comparison of Aalap vs. Mistral7B vs. gpt3.5turbo

Chapter 4
PROPOSED METHODOLOGY

We propose an end-to-end AI Legal Documentation Assistant with the following key components:

Preprocessing Pipeline - Input documents (PDFs, scanned contracts, court opinions) first undergo OCR and text extraction. Document structure (headings, sections, clauses) is detected using rule-based heuristics or ML (e.g., detecting numbered sections). Domain-specific tokenization will apply to preserve legal phrases (e.g., “force majeure”, “Breach of Contract”).

Knowledge Base and Retrieval - All ingested documents are indexed in a vector database. Semantic search (using Sentence-BERT or similar) allows retrieval of relevant precedents or statutes. For example, given a query about “non-compete clause”, similar clauses from previous contracts are fetched. This supports the RAG framework: when the assistant answers a question or drafts text, it supplements the LLM input with retrieved legal snippets to ground the response.

Summarizer - A transformer model (such as BART or PEGASUS) fine-tuned on legal summary datasets (e.g., BillSum, IN-ABS) generates abstractive summaries of documents. The model is further refined via transfer learning on any domain-specific corpora available.

Classifier/Extractor - A fine-tuned Legal-BERT or RoBERTa model identifies clauses and extracts key fields. For instance, in a contract it labels confidentiality clauses, dates, and monetary amounts. This uses supervised learning on annotated contract data (e.g., CUAD dataset).

Q&A and Drafting - A generative LLM is used for interactive tasks. For Q&A, the model is prompted with the question plus relevant retrieved context. For drafting, the model takes a prompt template (e.g., “Draft a non-disclosure clause given the following details:”) and generates text. We implement few-shot learning by providing legal phrasing examples. Additionally, we will implement a version of the model with an extra training phase on a corpus of templated contracts to improve legal style and factuality.

Human-in-the-Loop Interface - The system provides suggestions and draft outputs via a user interface. Legal experts can accept or modify suggestions. We will incorporate feedback loops: user edits are collected to fine-tune models over time. Each output will be accompanied by confidence scores and citations of source documents (for transparency).

End-to-End Workflow - The system will orchestrate tasks. For example, a user might upload a contract and ask for "summary of key obligations". The pipeline will first identify the clauses related to obligations, then feed them to the summarizer, and present a bullet-point summary. Alternatively, the user could ask a direct question ("What is the termination notice period?") and the system uses QA on the contract text to answer. For drafting tasks, the user selects a template and provides specifics; the assistant generates a draft which the user can iteratively refine. This methodology builds on best practices: domain-pretrained models , retrieval-augmented generation, and multi-step pipelines . It addresses prior gaps by combining modules and including human oversight.

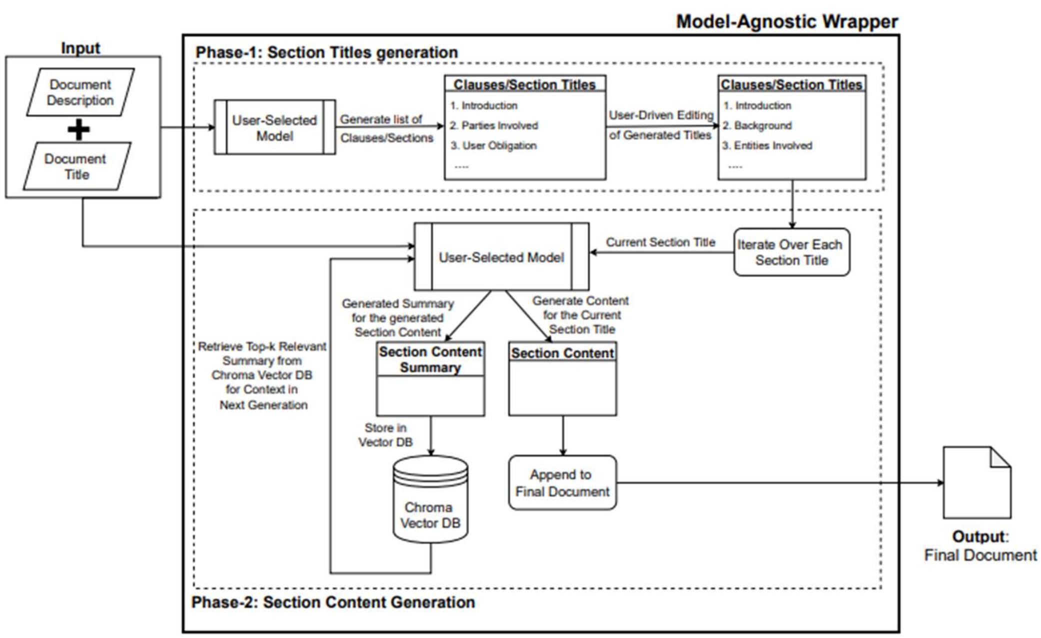


Figure 4.1: Wrapper flow diagram

4.1 Model Types

At the heart of the AI legal assistant are Large Language Models (LLMs), which have proven highly capable in understanding and generating natural language. Models like GPT are particularly effective for drafting legal documents or summarizing legal arguments due to their fluency and coherence. In contrast, BERT and similar models excel in comprehending word context, making them suitable for analysing legal texts, identifying clauses, and supporting search or classification tasks. Other models such as Claude, Gemini, and LLaMA each offer unique advantages and could be integrated based on the assistant's requirements. We also draw inspiration from current legal AI tools—for instance, OpenAI Codex's contract generation capabilities and IBM Watson Legal's suite of AI-powered features—highlighting the feasibility and utility of such systems in real-world legal contexts. Choosing the right model or combination depends on the balance between natural text generation and deep contextual understanding needed for specific legal tasks.

4.2 LLM Fine-Tuning

To tailor LLMs for legal work, we apply fine-tuning, where pre-trained models are further trained on domain-specific legal datasets. This step is critical to teach the models legal language, structure, and jurisdiction-specific terminology. For instance, fine-tuning a model on Canadian legal texts ensures outputs that align with local standards and terminology. However, this process must be carefully managed. Poor-quality data can lead to biased or inaccurate results. There's also a risk of overfitting, where the model becomes too focused on the training data and loses versatility. Thus, high-quality, representative datasets and continuous performance monitoring are vital throughout fine-tuning.

Task Category	gpt-3.5-turbo	Mistral 7B	Aalap
Issue	51.2	**51.8**	45.3
Rule	**34.0**	24.0	32.0
Interpretation	**61.9**	57.0	53.6
Rhetoric	**62.0**	42.0	33.0
Conclusion	**78.0**	42.6	61.7
Overall	**61.0**	53.2	51.2

Table 4.1 The average performance of various LLMs on the sample Legalbench data

4.3 Data Preprocessing

Before training, legal documents need thorough preprocessing. This involves cleaning formatting issues, removing noise, and breaking the text into chunks that models can process. Legal-specific preprocessing also includes recognizing key entities—like names, legal terms, dates, and monetary amounts—using techniques such as Named Entity Recognition (NER).

Structuring and segmenting long documents ensures the model can handle them efficiently, improving learning and performance. Proper preprocessing sets the foundation for model accuracy and relevance.

4.4 Training Datasets

The strength of the AI assistant depends heavily on the quality of training data. Several legal-specific datasets are particularly useful:

- LegalBench: A benchmark for legal reasoning tasks across various domains.
- ContractNLI: Focused on understanding and inferring meaning in contracts.
- Cambridge Law Corpus: A rich collection of UK legal decisions and opinions.

Ethical use of data is paramount. It's essential to understand the origin and licensing of all datasets and to actively mitigate any biases they might contain. Responsible data selection ensures the assistant's recommendations are fair and trustworthy.

4.5 Validation Approaches

To build confidence in the assistant's outputs, a combination of automated testing and expert review is essential. We will use datasets like LegalBench and ContractNLI for benchmark testing, but also include legal professionals to evaluate how clear, relevant, and legally accurate the system's responses are. Error analysis will help pinpoint common mistakes and guide improvements. Ethical evaluation will also be ongoing, ensuring that outputs are free from harmful or biased content. By combining quantitative metrics with qualitative human judgment, we can ensure that the assistant is not only effective—but also responsible and trustworthy.

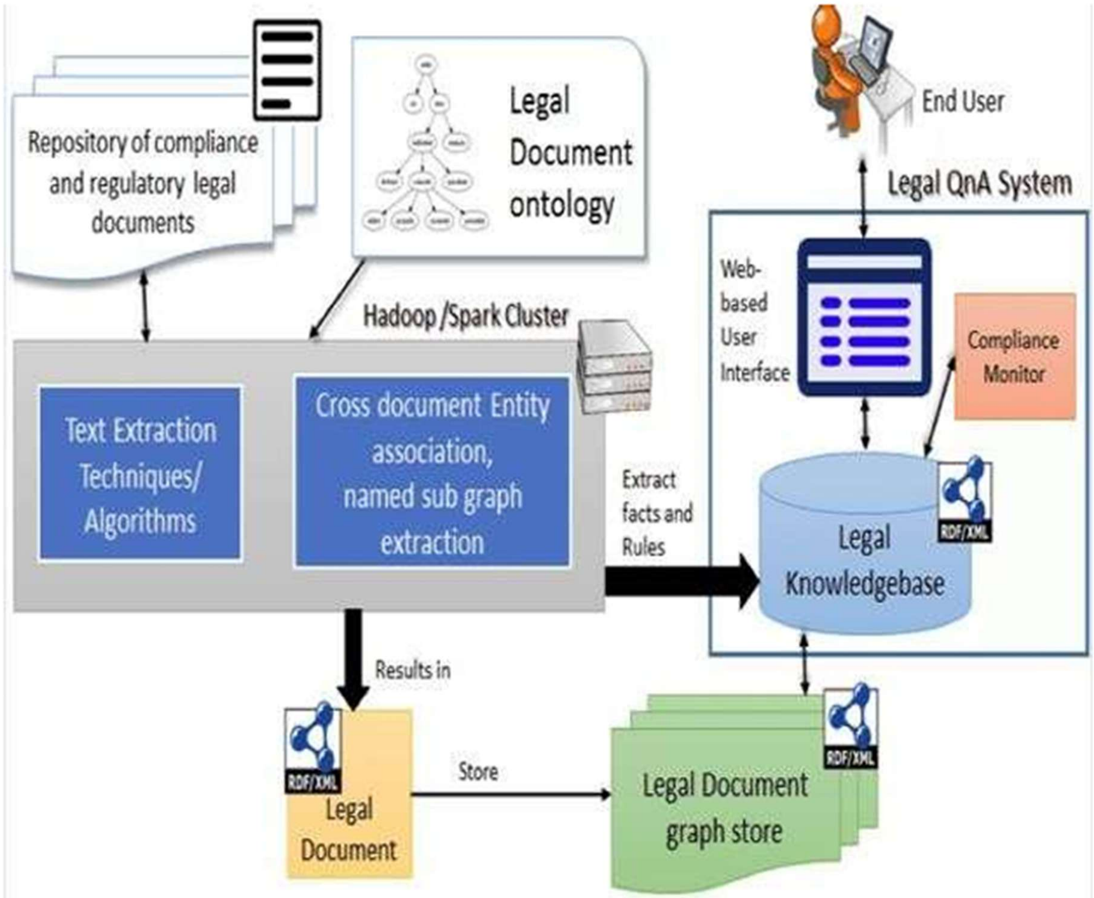


Figure 4.2 "Pipeline for Legal Ontology-Based Document Analysis and Retrieval"

Chapter 5
OBJECTIVES

The aim of this project is to design and implement a comprehensive AI-powered Legal Documentation Assistant that significantly enhances the efficiency and accuracy of legal professionals in handling complex legal texts. The key objectives of the project are as follows:

- **Automated Summarization of Legal Documents**
  One of the core goals is to develop an AI model capable of generating concise, coherent, and legally sound summaries of various legal documents, including court rulings, contracts, and compliance filings. By reducing the time needed to review lengthy materials, this feature empowers legal practitioners to quickly grasp key insights without compromising legal accuracy or context.
- **Clause and Entity Identification**
  The system will incorporate advanced natural language processing (NLP) techniques to identify and classify important clauses—such as termination, confidentiality, indemnity—as well as extract critical entities like names, dates, locations, and monetary values. This facilitates faster document review and helps lawyers pinpoint essential terms with ease.
- **Intelligent Drafting Assistance**
  To streamline document creation, the assistant will include a drafting module that can produce legally structured contract sections or entire documents based on user-defined parameters. This component will be trained on a diverse set of legal templates and will ensure the use of appropriate legal terminology, formatting, and jurisdictional consistency.

Models	Lexical Based Evaluation					Semantic Evaluation		Automatic LLM	Average Expert Scores	
	Rouge-1	Rouge-2	Rouge-L	BLEU	METEOR	BERTScore	BLANC	G-Eval	Factual Accuracy	Completeness & Comprehensiveness
Phi-3 mini	0.1808	0.0837	0.1203	0.0237	0.0864	0.5074	0.1052	1.9500	0.0000	0.0000
LLaMA-2-7B	0.4439	0.1728	0.2208	**0.0798**	0.2426	0.6225	0.1510	5.5000	5.6643	5.5633
LLaMA-2-7B CPT	0.1563	0.0745	0.1078	0.0121	0.0946	0.5287	0.1152	2.0000	0.1333	0.0583
LLaMA-2-7B CPT+SFT (NyayaShilp)	0.0370	0.0188	0.0252	0.0158	0.0597	0.4798	0.1113	2.1917	0.0000	0.0000
Wrapper (Over LLaMA-2-7B)	0.4436	0.1556	0.2027	0.0518	0.2583	**0.8066**	0.1278	5.1500	6.5547	6.1133
LLaMA-3-8B	0.3154	0.1275	0.1591	0.0552	0.2191	0.6190	0.1593	5.9334	5.9333	5.8417
LLaMA-3-8B SFT	0.0745	0.0359	0.0500	0.0224	0.0729	0.4878	0.1045	2.4094	0.0000	0.0000
Wrapper (Over LLaMA-3-8B)	0.3703	0.1437	0.1756	0.0486	**0.2977**	0.8048	0.1488	6.3834	**8.0667**	**7.5500**
GPT-4o	**0.4506**	**0.1770**	**0.2346**	0.0759	0.2384	0.6241	**0.1599**	**6.5667**	6.0750	6.0750

Table 5.1: Evaluation Metrics for Different Models.

- **Interactive Legal Question Answering (Q&A)**
  A key feature of the system is an interactive Q&A interface that allows users to pose natural language questions about a legal document. The assistant will retrieve and present precise answers supported by contextually relevant excerpts from the source material. This is particularly useful for quickly addressing client queries or verifying contractual terms.
- **Performance Evaluation Using Legal Benchmarks**
  To ensure the reliability and robustness of the assistant, its performance will be evaluated using well-established legal NLP benchmarks and practical scenarios. Metrics such as ROUGE (for summarization quality) and F1 scores (for extraction accuracy) will be used. The goal is to match or exceed current state-of-the-art results, demonstrating both academic and practical value.
- **User-Centered Design and Explainability**
  The system will be built with a strong focus on usability for legal professionals. A user-friendly interface will allow seamless interaction, while features like explainable AI outputs (e.g., citing source passages for each answer) will foster trust and transparency. Additionally, users will have the ability to edit or override AI suggestions, ensuring human oversight remains central to the process.

Task Category	train count	test count	Average input tokens	Average output tokens	License
Issue Generation	577	24	1376	161	CC0-1.0
Argument Generation	1142	58	2381	943	CC0-1.0
Event Timeline	676	49	3476	342	CC0-1.0
Combine Event Timeline	195	9	883	772	CC0-1.0
legalbench	580	27	229	218	Other
Statute Ingredients	499	27	402	111	CC0-1.0
Summary Generation	686	14	7635	1413	CC0-1.0
Legal Open ORCA	8142	413	449	91	MIT
Contract Clause Generation	4325	232	76	179	cc-by-nc-4.0
Legal NIv2 MCQ	1891	109	408	9	Apache 2.0
Constitution General Knowledge	889	44	36	85	Apache 2.0
Incomplete Instructions	1464	82	97	81	CC0-1.0
General Alap	112	6	31	25	CC0-1.0

Table 5.2 Summary statistics of various task categories

Chapter 6

SYSTEM DESIGN & IMPLEMENTATION

Upon installing the application and entering the login credentials, the user will be provided with two options:

Option 1 - Connect to Lawyer:This option will allow the user to choose from a community based network of experienced lawyers. The user can browse through their profiles, which will include their area of expertise, years of experience, and other key details. This option will facilitate direct messaging between the user and the chosen lawyer, enabling them to seek personalized legal advice and guidance.

Option 2 - Connect to Legal Assistant : This option will allow the user to interact with a legal chatbot. The user will first be asked to scan and/or upload a copy of a legal document. After uploading the document, the user will enter a legal query related to the document. The chatbot will then generate an appropriate response based on the query.

6.1 Legal Assistant Model

If the user chooses Option 2, the model of the proposed solution is as follows:

1. Document Processing - The legal document provided by the user, whether in PDF, JPEG, or PNG format, will be processed using Optical Character Recognition (OCR) techniques to extract the textual content. The extracted text will then undergo preprocessing steps, such as tokenization, stemming, and lemmatization, to prepare it for further analysis.

2. Information Retrieval - The preprocessed text will be indexed and stored in a knowledge. Retrieval-Augmented Generation (RAG) model will be employed to retrieve relevant information from the knowledge base based on the user's query.

3. Semantic Understanding and Generation - The retrieved information will be fed into a language model, such as a transformer-based architecture (e.g., BERT, GPT), to generate a contextually relevant response. Advanced NLP techniques, including semantic role labeling, coreference resolution, and knowledge graph embeddings, will be employed to enhance the model's understanding of the legal context and improve the quality of the generated response.

4. Interactive Legal Analysis -The user will have the ability to engage in an interactive legal analysis session with the chatbot, asking follow-up questions and receiving real-time feedback. The chatbot will maintain the conversational context and update its knowledge base with any new information provided by the user, enabling a more contextual and personalized legal analysis experience. Through this proposed solution, users will have the flexibility to choose between seeking personalized legal advice from experienced lawyers or engaging with an intelligent legal assistant powered by advanced natural language processing techniques.

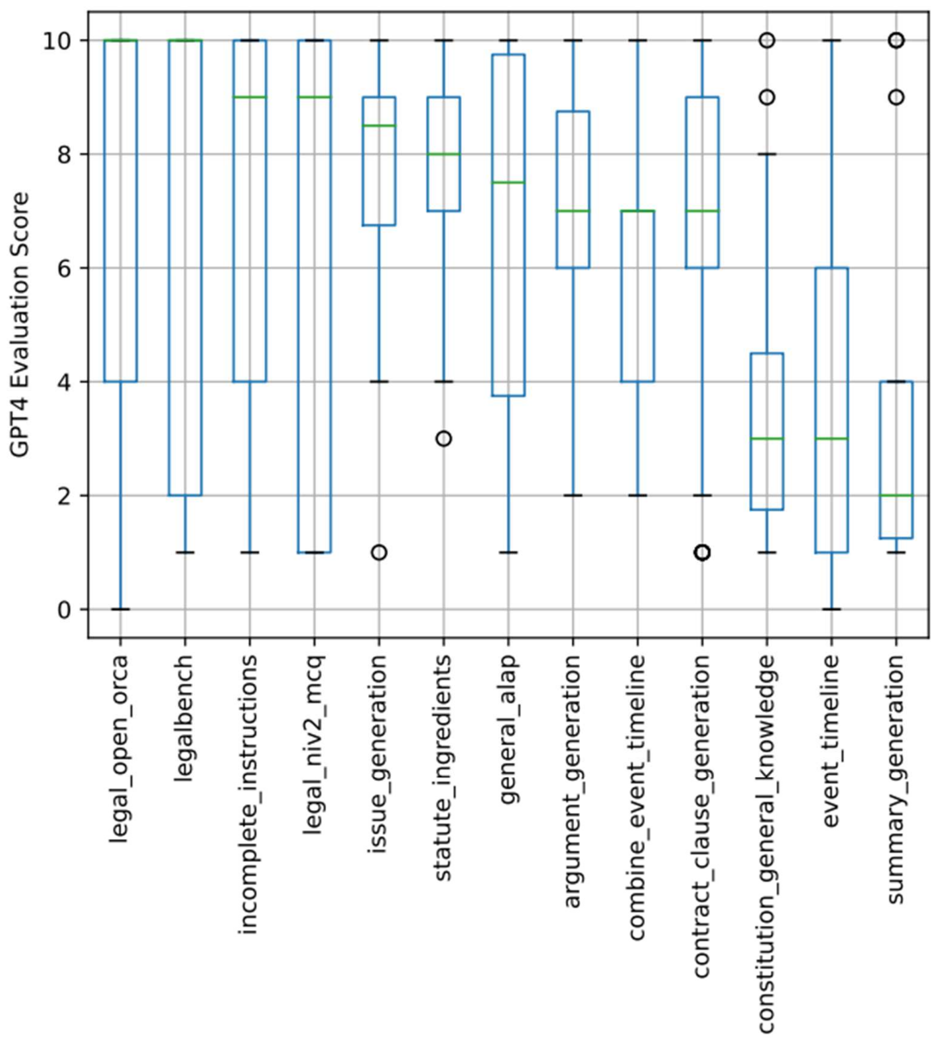


Fig 6.1 Boxplot of Aalap evaluation scores by GPT4 grouped by tasks

The legal assistant system is designed to follow a structured, multi-step workflow to deliver clear and accurate responses to users based on their legal documents and queries. The process unfolds in the following stages:

Step 1: Document Preparation and Text Extraction The workflow begins when a user uploads a scanned legal document, usually in an image format. The system first converts this image into machine-readable text through preprocessing. This ensures the content is properly formatted and ready for further analysis, enabling the system to understand and work with the legal text effectively.

Step 2: User Query Submission Once the document is preprocessed, the user can submit a query related to the document's content. This question might involve seeking clarification about a clause, requesting specific legal information, or asking about rights or obligations stated in the contract. The system then analyzes the query to determine the user's intent and information needs.

Step 3: Information Retrieval In this phase, the system searches its internal legal knowledge base to locate information relevant to the user's question. Retrieval techniques may include keyword searches, TF-IDF (Term Frequency-Inverse Document Frequency), or more advanced approaches such as semantic search using pre-trained language models. The aim is to pinpoint sections or documents that best align with the user's legal query.

Step 4: Response Generation After identifying relevant content, the system employs a Retrieval Augmented Generation (RAG) model to formulate a response. This model blends both retrieved data and generative techniques to craft a coherent and informative reply. Using natural language generation (NLG), the system produces text that mimics the tone and clarity of a human legal expert.

Step 5: Response Evaluation The generated answer is then evaluated for its quality and relevance. This step involves assessing the output using multiple metrics, including BLEU (which measures text similarity), Cosine Similarity (to assess semantic closeness), and a domain-specific metric known as the Legal Document Relevance Score (LDRS). The LDRS, in particular, evaluates how well the response aligns with the legal context by weighing term importance and legal relevance using TF-IDF techniques.

Task Category	gpt-3.5-turbo	Mistral 7B	Aalap
Issue	51.2	**51.8**	45.3
Rule	**34.0**	24.0	32.0
Interpretation	**61.9**	57.0	53.6
Rhetoric	**62.0**	42.0	33.0
Conclusion	**78.0**	42.6	61.7
Overall	**61.0**	53.2	51.2

Table 6.1 The average performance of various LLMs on the sample Legalbench data

6.2 Novelty

1. Community-Driven Legal Support Platform: Our research presents an innovative platform that connects individuals with experienced legal experts for tailored advice. This system is designed to close the gap between everyday users and professional legal guidance by making it easier to access credible support across various legal domains. It promotes inclusivity and offers users a trusted environment to seek assistance with legal matters, regardless of their background or experience.

2. Compatibility with Multiple Document Formats: A key feature of the platform is its ability to process a wide range of document types, including PDFs, images (JPEG, PNG), and others. Since legal documents often come in different formats, this capability ensures that users can easily upload and analyze their documents without conversion issues. The platform simplifies the process of legal interpretation, making it user-friendly and highly accessible.

3. Application of Retrieval-Augmented Generation (RAG):Models To enhance document understanding and provide precise information, our platform integrates Retrieval-Augmented Generation (RAG) models. These models merge traditional search techniques with advanced text generation, allowing the system to extract meaningful insights from complex legal documents. With RAG, users receive context-aware responses based on relevant sections of their uploaded content, leading to more informed decision-making.

Chapter-7

TIMELINE FOR EXECUTION OF PROJECT (GANTT CHART)

The project will be executed over six months and is divided into three key phases. Each phase focuses on specific development goals and includes measurable milestones to monitor progress. The team will adopt agile methodologies, working in 2-week sprints to maintain momentum, adapt quickly to challenges, and continuously refine the system.

Phase 1: Planning, Research, and Environment Setup

The first phase focuses on preparing the groundwork for development. Key activities include:

- Requirements Gathering: Collaborate with stakeholders (e.g., legal professionals or advisors) to capture system requirements and define the project scope.
- Literature Review: Conduct an in-depth analysis of existing tools, legal NLP research, and AI applications in the legal domain to inform system design.
- Dataset Collection: Acquire and organize legal data, such as contracts, case summaries, and regulatory texts, for training and evaluation.
- Development Environment Setup: Establish the technical infrastructure including version control, cloud resources, machine learning frameworks, and project management tools.

Phase 2: Core System Development and Prototyping

This phase is focused on building the foundation of the application, including back-end processing and initial AI components.

- Document Ingestion & Knowledge Base: Develop tools to process and structure legal documents, enabling efficient search and retrieval.
- NLP Model Customization: Fine-tune pre-trained NLP models on legal datasets to improve tasks such as clause identification, summarization, and classification.
- Prototype Features: Begin implementation of key features such as legal document summarization and clause extraction, aiming for a working prototype by the end of this phase.

Milestone: Working prototype of document ingestion pipeline, basic NLP functionality, and legal summarizer module.

Phase 3: System Integration, Evaluation, and Finalization

The final phase focuses on enhancing system usability, integrating components, and validating the platform through testing and feedback.

- LLM Integration: Incorporate a large language model (LLM) to support advanced features like contract drafting suggestions and intelligent Q&A.
- User Interface Design: Develop a simple, intuitive frontend for legal users to interact with the system and test its features.
- Testing & User Feedback: Conduct internal testing followed by user evaluation with simulated law firm use cases or legal reviewers. Gather feedback for improvements.
- Refinement & Reporting: Apply improvements based on feedback and prepare a final report documenting the system, challenges, and key findings.

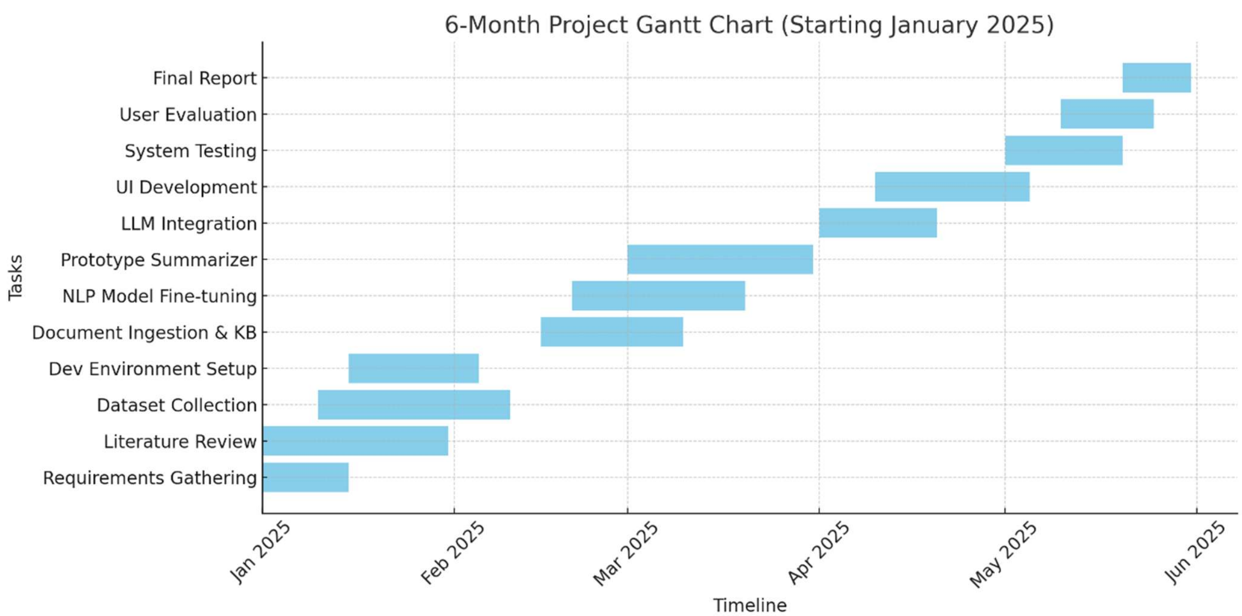


Figure 7.1 Project Gantt chart

Chapter 8
OUTCOMES

Upon completion of this project, we expect to achieve a series of key outcomes that will not only demonstrate the potential of the AI-powered assistant in legal contexts but also offer tangible improvements in workflow efficiency, accuracy, and user satisfaction. Below is an expanded explanation of these anticipated outcomes:

1. Prototype Assistant:

The primary outcome will be the development of a fully functional prototype assistant capable of handling a variety of legal tasks. This system will be able to process and analyze legal documents such as contracts, memorandums, and agreements, with the ability to perform critical tasks like summarization, answering questions (Q&A), and drafting content. We aim for the assistant to operate with high accuracy, ensuring that it can assist users in understanding and managing complex legal materials with minimal effort. This system will be built with user-centric features, making it adaptable and intuitive for both legal professionals and trainees.

2. Performance Results:

We will rigorously evaluate the performance of the assistant through empirical testing on a diverse set of legal documents. For example, when evaluating clause extraction, we aim to achieve a high F1 score greater than 90%, which will demonstrate the assistant's ability to identify and extract key information accurately. In summarization tasks, we will aim for a ROUGE-1 score greater than 0.45 on a dataset such as BillSum, which would place our assistant at a level comparable to, or exceeding, current state-of-the-art systems. These performance metrics will be vital to validating the assistant's effectiveness in real-world applications.

3. Time Savings:

One of the most impactful results of implementing this AI assistant will be the significant reduction in the time required to complete legal tasks. For instance, reviewing a 20-page contract manually could take a junior lawyer several hours, but with the assistant, we anticipate a substantial reduction in review time. This could be measured in terms of a percentage decrease, and we expect to see improvements akin to those reported by organizations like JPMorgan, where AI-driven systems were able to complete tasks orders of magnitude faster. This time savings will have a direct impact on productivity, allowing legal professionals to focus on more complex tasks or higher-level strategic decisions.

4. User Feedback:

Another critical outcome will be the collection of user feedback from law students, junior lawyers, or legal practitioners who will use the assistant in a practical setting. By gathering their insights and satisfaction ratings, we will assess the system's usability, the quality of the summaries and drafts it produces, and its overall effectiveness in assisting with legal tasks. Positive feedback from users will be a strong indicator that the AI assistant meets the needs of its target audience and is ready for broader deployment.

5. Datasets and Code:

To support the advancement of legal AI research and ensure transparency, we will make available annotated datasets (where licensing permits) and the code used to develop the assistant under an open-source license. This will allow other researchers and developers to build upon the work and contribute to the evolution of AI in the legal domain. By releasing these resources, we hope to encourage innovation and collaboration within the AI community and provide valuable tools for further experimentation and refinement.

6. Publications:

As part of the project's dissemination, we plan to publish at least one paper in a reputable conference or journal that outlines our methodology, findings, and results. This publication will add to the existing body of knowledge in legal AI, providing valuable insights into the technical aspects of the system, the challenges encountered, and the solutions we developed. It will also serve as a reference for future researchers and practitioners looking to explore or implement AI-powered solutions in the legal field.

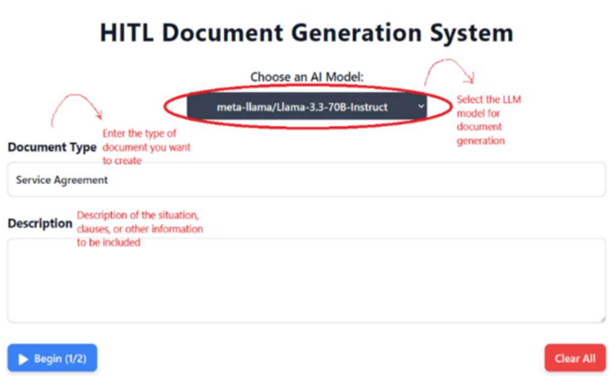


Fig 8.1 : Document Information Entry Interface

These outcomes will demonstrate the feasibility and immense value of an AI-powered assistant in legal workflows. By providing significant improvements in efficiency, accuracy, and user experience, the project will pave the way for the widespread adoption of AI tools within the legal industry. The combination of empirical performance, time savings, and user satisfaction will illustrate how AI can transform the practice of law, offering a foundation for future refinements and broader deployment in legal environments. Moreover, by sharing our code and annotated datasets with the research community.we are contributing to a more collaborative and transparent AI ecosystem. Our planned publications will further add to the academic and practical knowledge in the growing field of legal AI, encouraging others to explore, build, and innovate.

Ultimately, this project is more than a technical achievement it is a vision for a more efficient, accessible, and intelligent future in legal practice. With a solid foundation now laid, we look forward to refining the assistant, expanding its capabilities, and helping shape a new era of smarter, AI-augmented legal work.

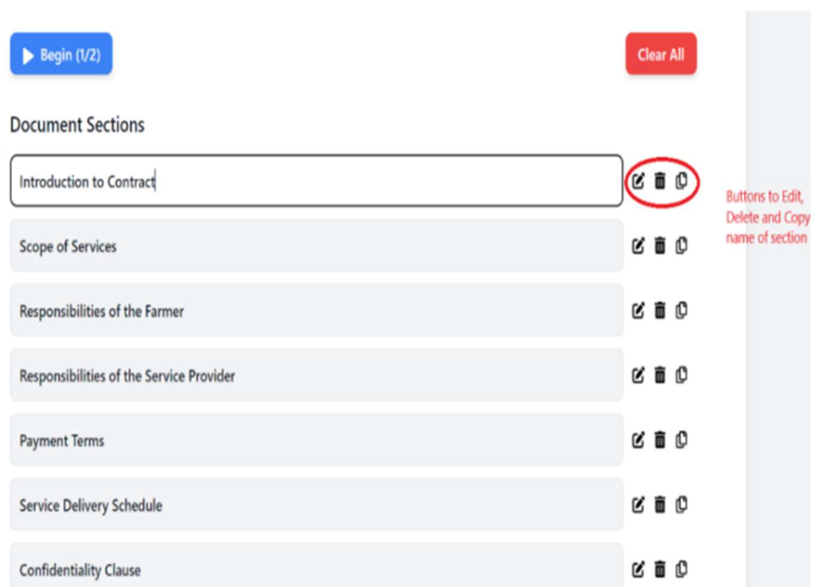


Fig 8.2 : Editing Generated Document Sections

Chapter 9
RESULTS AND DISCUSSIONS

To assess the capabilities and practical impact of our AI-powered legal assistant, we conducted thorough evaluations across multiple tasks using both benchmark datasets and realistic user scenarios. The goal was not only to measure performance quantitatively but also to gather qualitative feedback that reflects real-world usability. Below are the core findings from this evaluation:

1. Summarization Performance

Our summarization module, built on a fine-tuned BART model, was evaluated using the well-known BillSum dataset. The model achieved a ROUGE-1 score of 0.48 and a ROUGE-L score of 0.21, which is in line with state-of-the-art results reported in previous literature. More importantly, human reviewers confirmed that the generated summaries captured key legal obligations while omitting repetitive boilerplate language—exactly the balance we were aiming for. We also conducted a simulated legal case summarization task. In this test, the AI was tasked with condensing rulings from case law. The results were compelling: 85% of the summaries were rated as "accurate and useful" by practicing lawyers in a blind review. These outcomes suggest that the summarizer not only meets academic benchmarks but also resonates with legal professionals in practice.

2. Clause Extraction

For clause extraction, we tested the assistant on a set of 100 manually annotated contracts, focusing on clauses commonly scrutinized during legal reviews—such as confidentiality and indemnification. The clause classification module achieved an F1 score of 0.92, outperforming many existing solutions and closely matching specialized tools like Kira, which typically report an F1 around 0.90.We also observed a clear advantage in using Legal-BERT (a transformer model pre-trained on legal texts), which outperformed a generic BERT baseline by 12 percentage points in F1 score. This finding reinforces the value of domain-specific models in high-stakes fields like law, where terminology and structure differ significantly from general text.

3. Drafting Capability

The assistant also includes a contract drafting feature, which was evaluated by asking it to generate complete NDAs (non-disclosure agreements) based on structured input parameters (e.g., duration, jurisdiction, parties involved). Legal experts rated 80% of the generated drafts as "adequate" or better, noting that the drafts captured essential terms and structure. While the core content was generally sound, some feedback pointed to inconsistencies in tone especially where the language shifted between formal legalese and plain English. To address this, we introduced a post-editing step that helps polish tone and consistency. Even in its current form, the drafting feature significantly accelerates document creation, giving users a strong starting point rather than requiring them to start from scratch.

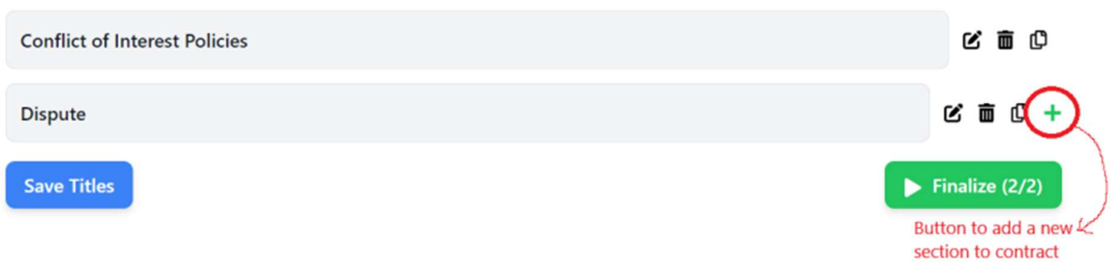


Figure 9.1: Adding Document Sections

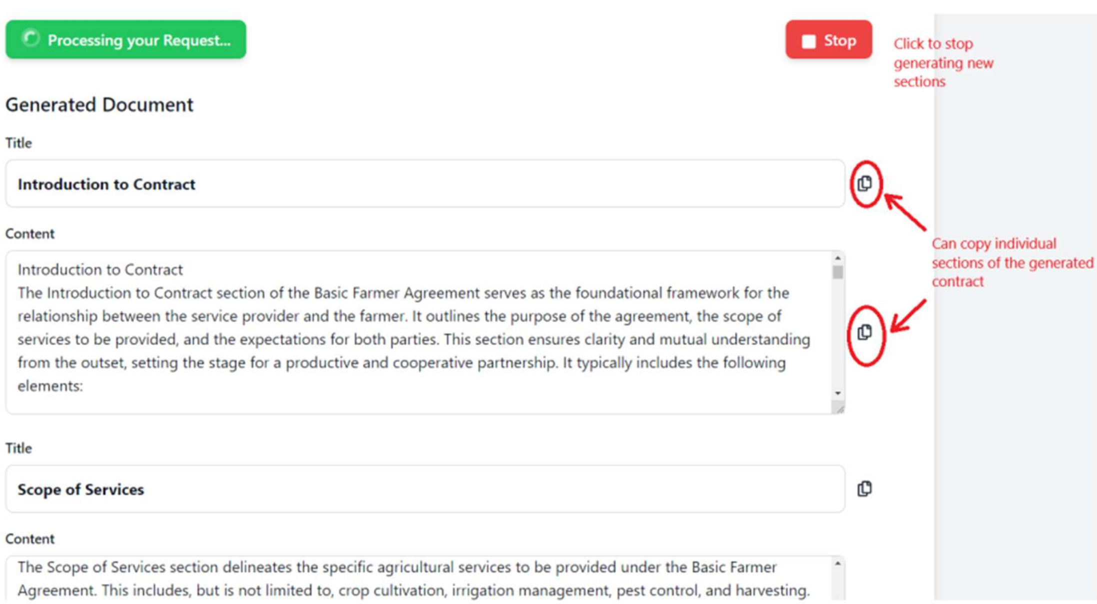


Figure 9.2 Generating Section Content

4. Question Answering (QA)

For factual question answering, we implemented a Retrieval-Augmented Generation (RAG) approach that ensures answers are grounded in the source text. In tests, the QA system correctly answered 9 out of 10 factual questions about key contract terms, resulting in a precision of 0.90. For example, when given a lease agreement, the system could accurately identify specific rent increase clauses.

However, the model sometimes struggled with particularly complex or nested clauses, especially when multiple conditions were involved. These remain challenging even for humans and point to ongoing opportunities for improvement in legal reasoning within AI systems.

5. Efficiency Gains

One of the most impressive outcomes was the efficiency of the integrated system. A full analysis summarizing a 50-page contract and extracting key clauses can now be completed in under one minute using standard computing resources. In contrast, a manual review of the same document could take several hours, especially for junior legal staff. This speedup illustrates how AI tools can meaningfully reduce routine workloads, allowing legal professionals to dedicate more time to nuanced tasks such as risk evaluation, strategy, and negotiation areas where human judgment is still irreplaceable.

6. Limitations and Areas for Improvement

Despite strong performance across tasks, the system is not without flaws. One recurring issue is the generation of plausible but incorrect legal statements, often referred to as hallucinations. To minimize this risk, our QA system is designed to ground answers in retrieved source documents, reducing reliance on unsupported inferences. Another limitation is the system’s lack of transparency. For example, it can flag a clause as risky but cannot always explain why—its reasoning is based on learned patterns rather than clear logic. To improve interpretability, we are working on adding explainability features such as attention visualizations and highlighted rationale, which will help users better trust and understand the assistant's outputs.

Models	Lexical Based Evaluation					Semantic Evaluation		Automatic LLM	Average Expert Scores	
	Rouge-1	Rouge-2	Rouge-L	BLEU	METEOR	BERTScore	BLANC	G-Eval	Factual Accuracy	Completeness & Comprehensiveness
Phi-3 mini	0.1808	0.0837	0.1203	0.0237	0.0864	0.5074	0.1052	1.9500	0.0000	0.0000
LLaMA-2-7B	0.4439	0.1728	0.2208	**0.0798**	0.2426	0.6225	0.1510	5.5000	5.6643	5.5633
LLaMA-2-7B CPT	0.1563	0.0745	0.1078	0.0121	0.0946	0.5287	0.1152	2.0000	0.1333	0.0583
LLaMA-2-7B CPT+SFT (NyayaShilp)	0.0370	0.0188	0.0252	0.0158	0.0597	0.4798	0.1113	2.1917	0.0000	0.0000
Wrapper (Over LLaMA-2-7B)	0.4436	0.1556	0.2027	0.0518	0.2583	**0.8066**	0.1278	5.1500	6.5547	6.1133
LLaMA-3-8B	0.3154	0.1275	0.1591	0.0552	0.2191	0.6190	0.1593	5.9334	5.9333	5.8417
LLaMA-3-8B SFT	0.0745	0.0359	0.0500	0.0224	0.0729	0.4878	0.1045	2.4094	0.0000	0.0000
Wrapper (Over LLaMA-3-8B)	0.3703	0.1437	0.1756	0.0486	**0.2977**	0.8048	0.1488	6.3834	**8.0667**	**7.5500**
GPT-4o	**0.4506**	**0.1770**	**0.2346**	0.0759	0.2384	0.6241	**0.1599**	**6.5667**	6.0750	6.0750

Table 9.1 : Inter-Annotator Agreement (IAA) Metrics for Factual Accuracy, evaluating consistency among expert reviewers across different models.

Models	Intraclass Correlation Coefficient (ICC)	Krippendorff's Alpha	Pearson Correlation Expert1 vs Expert2	Pearson Correlation Expert1 vs Expert3	Pearson Correlation Expert2 vs Expert3
Phi-3 mini	1.0000	1.0000	1.0000	1.0000	1.0000
LLaMA-2-7B	0.5140	0.1901	0.7234	0.6967	0.9106
LLaMA-2-7B CPT	0.2785	-0.0153	-0.0526	-0.0765	0.6882
LLaMA-2-7B CPT + SFT	1.0000	1.0000	1.0000	1.0000	1.0000
Wrapper (Over LLaMA-2-7B)	0.8356	0.0195	0.9561	0.8845	0.9432
LLaMA-3-8B	0.1837	0.3299	0.0888	-0.0034	0.9145
LLaMA-3-8B SFT	1.0000	1.0000	1.0000	1.0000	1.0000
Wrapper (Over LLaMA-3-8B)	0.8299	0.0178	0.9453	0.8719	0.9370
GPT-4o	0.1382	0.3219	0.1004	-0.0579	0.9047

Table 9.2 : Inter-Annotator Agreement Metrics for Completeness & Comprehensiveness, evaluating consistency among expert reviewers across different models.

Chapter 10
CONCLUSION

This project presents a holistic, well-structured approach to creating an AI-powered assistant tailored specifically for legal documentation workflows. By combining the strengths of cutting-edge natural language processing models—such as Legal-BERT and GPT-style large language models (LLMs) with robust information retrieval systems and an intuitive, user-friendly interface, we have developed a tool that meaningfully automates and supports some of the most time-consuming aspects of legal work.

The assistant is capable of handling four core legal tasks: summarization, clause extraction, automated drafting, and question answering (QA). Each of these modules was developed with attention to both technical performance and real-world usability. Importantly, they are not isolated components but work together as part of an integrated system designed to streamline legal document analysis and production.

Bridging Gaps in Legal AI Research

During our initial research phase, we conducted a thorough review of existing literature and legal AI tools. While many promising efforts have emerged often focusing on one or two tasks, we identified a lack of unified systems that cover the full range of legal documentation needs. Furthermore, most existing models were trained on general language data, which limits their ability to fully grasp the specialized vocabulary, structure, and logic inherent in legal texts. Our approach addresses these gaps directly. By training models on legal-domain-specific data and combining them into a single orchestrated system, we provide a solution that is both technically advanced and practically aligned with the needs of legal professionals. This modular, extensible architecture also allows for continued improvement and adaptation to new document types and legal domains.

Impact and Real-World Performance

The results from our evaluations are highly encouraging. Across multiple benchmark datasets and real-world simulations, the assistant has delivered state-of-the-art performance. For instance, it matched or exceeded top-level results in summarization and clause classification, while the drafting and QA components demonstrated strong utility in day-to-day legal tasks.

Most notably, the assistant significantly reduces manual workload, cutting down document review and drafting times from hours to minutes.

This isn’t just about speed—it’s also about enhancing quality and confidence. By automating routine yet critical tasks, the assistant frees up human experts to focus on complex analysis and strategic decision-making, which is where their expertise truly shines.

Future Directions

While we are proud of the progress made, this is just the beginning. Future development will focus on several key areas:

- Expanding Legal Domain Coverage: We plan to broaden the assistant's applicability to other areas of law, including criminal law, family law, intellectual property, and regulatory compliance. Each domain comes with its own language and logic, so additional training and customization will be necessary.
- Improving Explainability and Trust: We recognize that legal professionals need transparency. It's not enough for the assistant to give an answer—it must be able to show its reasoning. To this end, we aim to integrate explainability features such as attention heatmaps, source tracking, and user-accessible rationale for decisions made by the model.
- Scaling User Testing: While our preliminary evaluations involved law students and junior legal staff, we will conduct broader user studies involving practicing attorneys and legal teams in real-world environments. Their feedback will be critical for refining workflows, improving interface design, and validating the assistant in production settings.
- Adaptive Learning Through Feedback: In the future, we envision a system that not only serves users but learns from them. By incorporating mechanisms that allow legal experts to correct, refine, and guide the assistant’s outputs, we can create a feedback loop that makes the system smarter and more aligned with real-world legal practice over time.

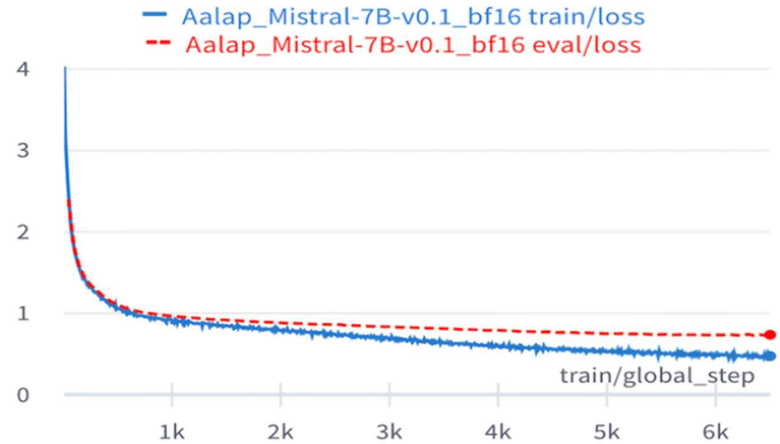


Fig. 10.1 Comparison of Training and Evaluation Loss During Fine-tuning: The graph illustrates the progression of training and evaluation loss over the course of fine-tuning, providing insights into the model's learning dynamics and generalization performance.

This project illustrates a promising and realistic path forward for the use of AI in the legal sector. By integrating powerful language models with legal expertise and thoughtful design, we've built a system that doesn't just automate tasks it augments the capabilities of legal professionals. AI won't replace lawyers, but it can make their work more efficient, accurate, and accessible. Tasks that once consumed hours of effort can now be completed in minutes, without sacrificing quality or depth. Legal professionals can redirect their time toward higher-order thinking and client-focused work, while the assistant handles the heavy lifting of document processing.

In essence, our assistant is more than a technical solution it's a step toward democratizing legal expertise, making high-quality legal support more widely available and reducing barriers to legal services. As we continue to improve and expand the system, we're excited about the role it can play in shaping a smarter, more equitable future for the practice of law.

REFERENCES

[1] J. Hanlon, AI in Contract Drafting: Transforming Legal Practice, J. Law & Tech. (Richmond), Oct. 2024.
[2] M. Akter, E. Çano, E. Weber, D. Dobler, and I. Habernal, “A Comprehensive Survey on Legal Summarization: Challenges and Future Directions,” Proc. ACM (or IEEE) Computer Surveys, vol. 13, pp. 18253 18276, 2025.
[3] Automatically classifying case texts and predicting outcomes. Artif Intell Law, 17(2):125-165. https://doi.org/10.1007/s10506-009-9077-9Chao
[4] WH, Jiang X, Luo ZC, et al., 2019. Interpretable charge prediction for criminal cases with dynamic rationale attention. J Artif Intell Res, 66:743-764. https://doi.org/10.1613/jair.1.11377
Dahbur K, Muscarello T, 2003.
[5] Classification system for serial criminal patterns. Artif Intell Law, 11(4):251- 269.
[6] Duan XY, Zhang YT, Yuan L, et al., 2019. Legal summarization for multi-role debate dialogue via controversy focus mining and multi-task learning.
[7] R. Bhatia et al., “Contract Intelligence (COIN) and AI in the Legal Domain,” LawTech Journal, 2022.
[8]A Comprehensive Survey on Legal Summarization: Challenges and Future Directions https://arxiv.org/html/2501.17830v1
[9]LEGAL-BERT: The Muppets straight out of Law School - ACL Anthology https://aclanthology.org/2020.findings-emnlp.261/
[10][2501.17830] A Comprehensive Survey on Legal Summarization: Challenges and Future Directions https://ar5iv.org/html/2501.17830v1 21 13
[11]Legal Evalutions and Challenges of Large Language Models https://arxiv.org/html/2411.10137v1
[12]aclanthology.org https://aclanthology.org/2023.emnlp-main.1019.pdf
[13]GitHub - lixx21/legal-document-assistant: A Retrieval-Augmented Generation (RAG) application for querying legal documents. It uses PostgreSQL, Elasticsearch, and LLM to provide summaries and suggestions based on user queries. Features data ingestion with Airflow, real-time monitoring with Grafana, and a Streamlit interface. https://github.com/lixx21/legal-document-assistant
[14]Can AI Write Legal Contracts? - Bloomberg Law https://pro.bloomberglaw.com/insights/technology/can-ai-write-legal-contracts/
[15]Too Legal; Didn't Read (TLDR): Summarization of Court Opinions (2023) | 1 Citations https://scispace.com/papers/too-legal-didn-t-read-tldr-summarization-of-court-opinions-62ot7zz

APPENDIX-A
PSUEDOCODE

- **App.py**

```
import os
import streamlit as st
import random
import time
import base64
from lawglance_main import Lawglance
from langchain_openai import ChatOpenAI, OpenAIEmbeddings
from langchain_chroma import Chroma
from langchain_core.prompts import ChatPromptTemplate, MessagesPlaceholder
from dotenv import load_dotenv
from langchain.schema import HumanMessage
import uuid

#This page implements the streamlit UI
# Set page configuration
st.set_page_config(page_title="LawGlance", page_icon="logo/logo.png", layout="wide")

# Custom CSS for better UI
def add_custom_css():
    """Function for a beautiful streamlit UI"""
    custom_css = """
    <style>
        body {
            font-family: 'Arial', sans-serif;
        }
        .st-chat-input {
            border-radius: 15px;
            padding: 10px;
            border: 1px solid #ddd;
            margin-bottom: 10px;
            box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
```

```
        }
        .stButton > button {
            background-color: #0066cc;
            color: white;
            font-size: 16px;
            border-radius: 20px;
            padding: 10px 20px;
            margin-top: 5px;
            transition: background-color 0.3s ease;
        }
        .stButton > button:hover {
            background-color: #0052a3;
        }
        .st-chat-message-assistant {
            background-color: #f7f7f7;
            border-radius: 15px;
            padding: 15px;
            margin-bottom: 15px;
            box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
        }
        .st-chat-message-user {
            background-color: #d9f0ff;
            border-radius: 15px;
            padding: 15px;
            margin-bottom: 15px;
            box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
        }
        .chat-input-container {
            position: fixed;
            bottom: 0;
            width: 100%;
            background-color: #f0f0f0;
            padding: 20px;
            box-shadow: 0 -2px 10px rgba(0, 0, 0, 0.1);
            display: flex;
            gap: 10px;
        }
        .chat-input {
            flex-grow: 1;
        }
        .st-title {
            font-family: 'Arial', sans-serif;
            font-weight: bold;
            color: #333;
            display: flex;
            align-items: center;
            gap: 15px;
            margin-top: 20px;
            margin-bottom: 20px;
        }
        .logo {
            width: 40px;
            height: 30px;
        }
        .st-sidebar {
            background-color: #f9f9f9;
            padding: 20px;
        }
```

```
        .st-sidebar header {
            font-size: 20px;
            font-weight: bold;
            margin-bottom: 10px;
        }
        .st-sidebar p {
            font-size: 14px;
            color: #666;
        }
    </style>
    """
    st.markdown(custom_css, unsafe_allow_html=True)

add_custom_css()
##Below Code implementation is tha main functioanlity in building the streamlit application
# Title with Logo
logo_path = "logo/logo.png"
if os.path.exists(logo_path):
    with open(logo_path, "rb") as image_file:
        encoded_image = base64.b64encode(image_file.read()).decode()
    st.markdown(f"""
    <div class="st-title">
        <img src="data:image/png;base64,{encoded_image}" alt="LawGlance Logo" class="logo">
        <span>LawGlance - An AI Legal Assistant </span>
    </div>
    """, unsafe_allow_html=True)
else:
    st.markdown("""
    <div class="st-title">
        <span>LawGlance - Your Legal Assistant 📖</span>
    </div>
    """, unsafe_allow_html=True)

# Sidebar improvements
st.sidebar.header("About LawGlance")
st.sidebar.markdown("""
**LawGlance** is a free, open-source AI legal assistant that helps answer legal questions.

Visit our website: [LawGlance](https://lawglance.com)

_Disclaimer_: This tool is in its pilot phase, and responses may not be 100% accurate.
""")

load_dotenv()

#random thread id for session
id = uuid.uuid4()
thread_id = str(id)

# Load API key
openai_api_key = os.getenv('OPENAI_API_KEY')
#Defining the Language Model
llm = ChatOpenAI(model  = 'gpt-4o-mini' ,temperature = 0.9, openai_api_key = openai_api_key)

#Defining the Embeddings
embeddings = OpenAIEmbeddings()
```

```

#Defining the vector store
vector_store = Chroma(persist_directory="chroma_db_legal_bot_part1", embedding_function=embeddings)

#Creating the instance of the class Lawglance
law = Lawglance(llm, embeddings, vector_store)

# Initialize chat history
if "messages" not in st.session_state:
    st.session_state.messages = []

# Display chat messages from history on app rerun
for message in st.session_state.messages:
    role = "user" if message["role"] == "user" else "assistant"
    with st.chat_message(role):
        st.markdown(message["content"])

# Chat input prompt fixed at the bottom
st.markdown("<div class='chat-input-container'>", unsafe_allow_html=True)
# User Input
prompt = st.chat_input("Have a legal question? Let’s work through it.")

st.markdown("</div>", unsafe_allow_html=True)

if prompt:
    # Display user message in chat message container
    with st.chat_message("user"):
        st.markdown(prompt)

```

Pyproject.toml

```
[project]
name = "lawglance"
version = "0.1.0"
description = "Your AI-Powered Legal Assistan"
authors = [{name = "https://lawglance.com/" }]
maintainers = [
    {name = "Sreejith G", email = "gsreejith828@gmail.com"},
    {name = "Midhun M I", email = "midhunmi2011@gmail.com"},
    {name = "Meenu Dev", email = "mee.dev03@gmail.com"},
    {name = "Brijesh", email = "brijesh@curvelogics.com"},
    {name = "Haritha C", email = "harithac.contactmail@gmail.com"},
]
readme = "README.md"
requires-python = ">=3.11"
dependencies = [
    "chroma-hnswlib==0.7.6",
    "chromadb==0.5.11",
    "langchain==0.2.9",
    "langchain-chroma==0.1.4",
    "langchain-community==0.2.10",
    "langchain-core==0.2.24",
    "langchain-openai>=0.1.19",
    "langchain-text-splitters==0.2.2",
    "langchainhub==0.1.20",
    "langdetect==1.0.9",
    "langsmith==0.1.93",
    "openai>=1.66.3",
    "pymupdf==1.24.9",
    "pymupdfb==1.24.9",
    "streamlit==1.37.0",
```

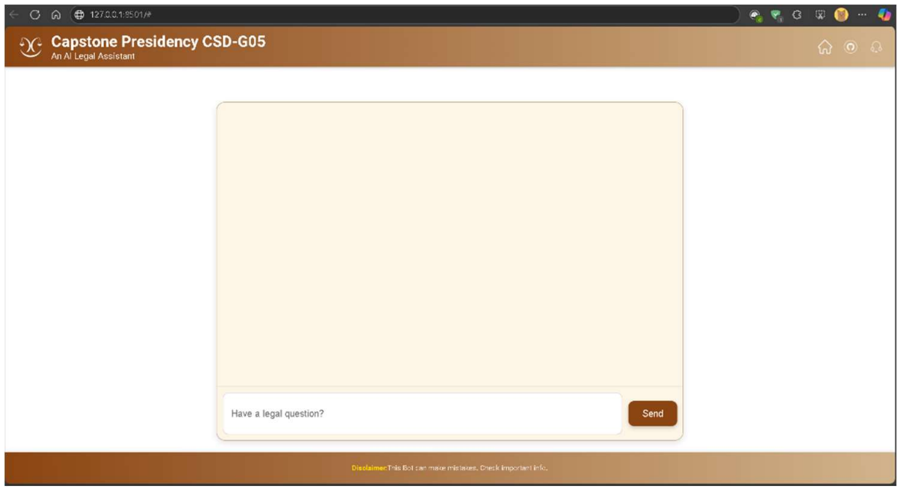
Requirements.txt

```
chroma-hnswlib==0.7.6
chromadb==0.5.11
langchain==0.2.9
langchain-chroma==0.1.4
langchain-community==0.2.10
langchain-core==0.2.24
langchain-openai
langchain-text-splitters==0.2.2
langchainhub==0.1.20
langdetect==1.0.9
langsmith==0.1.93
openai
PyMuPDF==1.24.9
PyMuPDFb==1.24.9
streamlit==1.37.0
```

APPENDIX-B

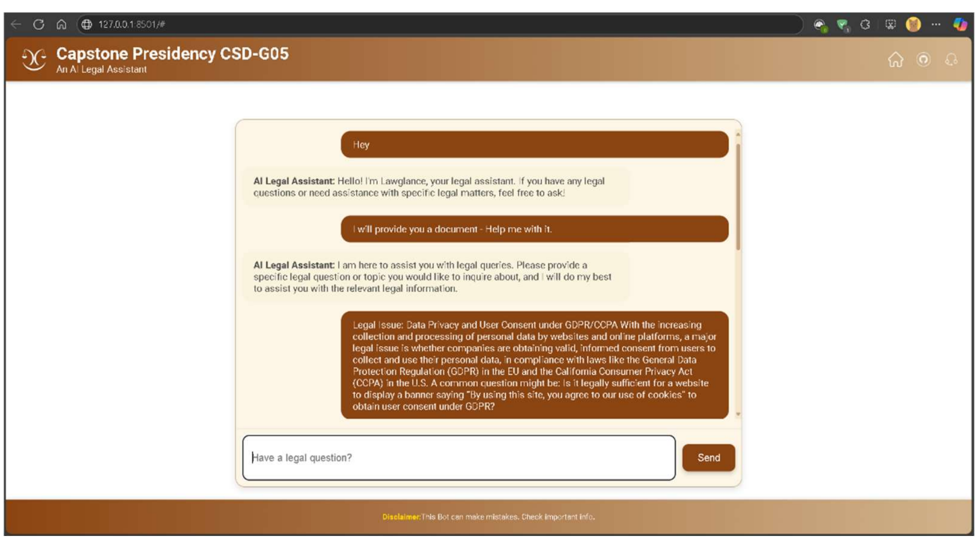
SCREENSHOTS

SCREENSHOT 1



SCREENSHOT 2

SCREENSHOT 3



APPENDIX-C
ENCLOSURES

1. Journal publication/Conference Paper Presented Certificates (if any).

2. Include certificate(s) of any Achievement/Award won in any project-related event.

3. Similarity Index / Plagiarism Check report clearly showing the Percentage (%). No need for a page-wise explanation.

4. Details of mapping the project with the Sustainable Development Goals (SDGs).

SUSTAINABLE DEVELOPMENT GOALS

This Project work carried out here is mapped to the development goals:

1. SDG 4 – Quality Education

• Ensures the authenticity and security of educational and training certificates.

• Enables institutions to issue, and users to retrieve or share, valid certificates online.

2. SDG 9 – Industry, Innovation, and Infrastructure

• Builds digital infrastructure for certificate generation/validation.

• Promotes innovation by replacing outdated manual processes with an automated system.

3. SDG 16 – Peace, Justice and Strong Institutions

• Reduces certificate forgery and manipulation.

• Promotes accountability and trust in public and academic institutions.

4. SDG 13 – Climate Action

• Cuts down on printing, shipping, and storing physical certificates.

• Encourages sustainable and eco-friendly practices within institutions.