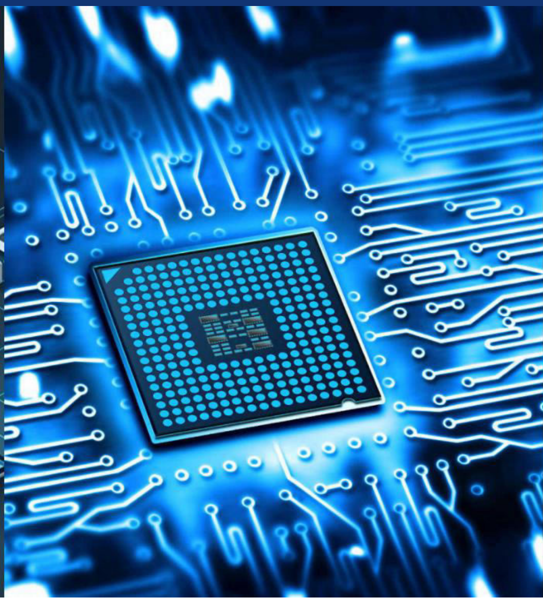
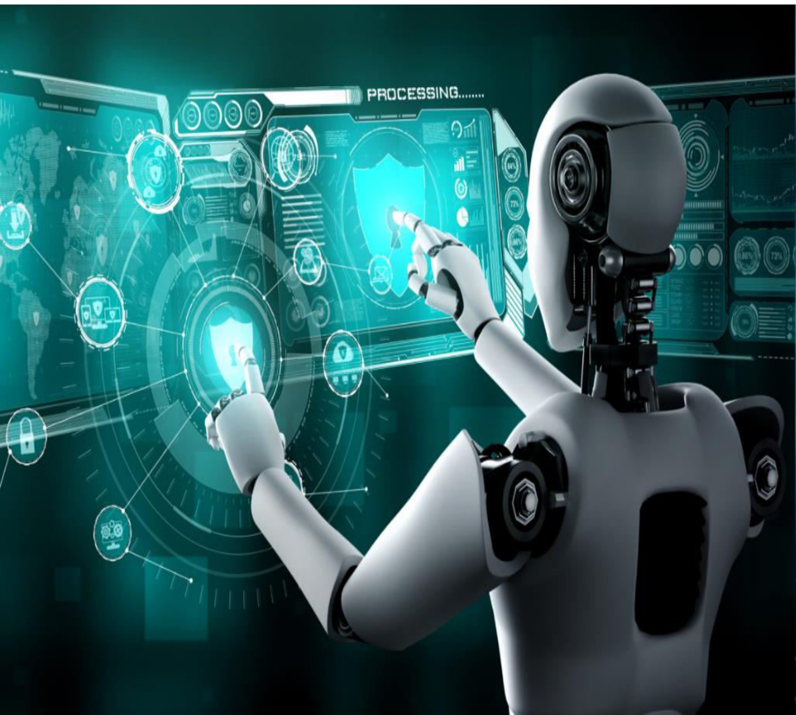
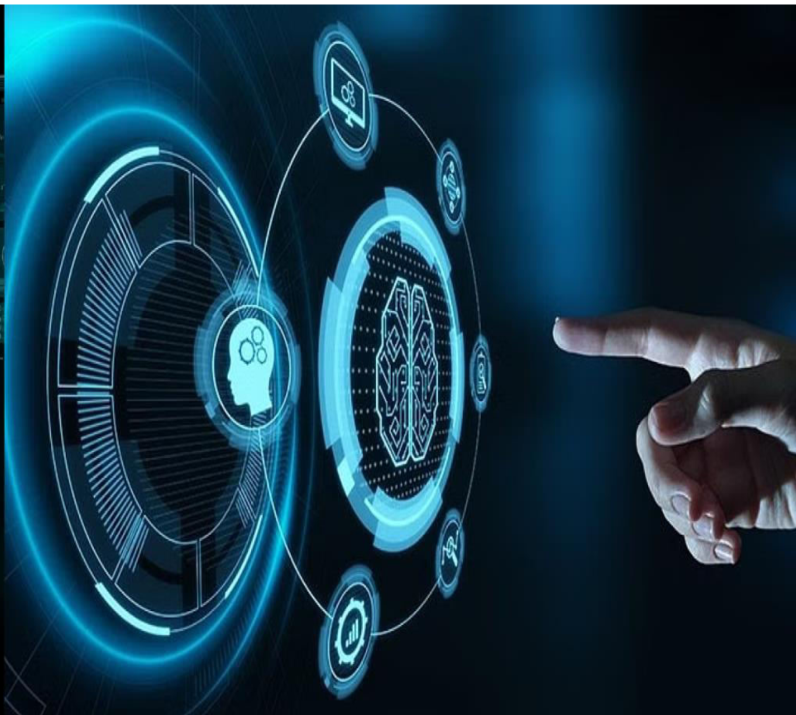
ISSN(O): 2320-9801
ISSN(P): 2320-9798


International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)





Impact Factor: 9.927

Volume 14, Issue 5, May 2026

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405113

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Intelligent Legal Document Analysis and Natural Language Query Assistance: An Integrated AI-Driven Framework

Attili Teja Navadeep, Dr. Chiraparapu Srinivasa Rao

PG Scholar, Department of Computer Science S.V.K.P & Dr. K.S. Raju Arts and Science College (Autonomous), Penugonda, Affiliated to Adikavi Nannaya University, Andhra Pradesh, India

Associate Professor, Department of Master of Computer Applications, S.V.K. P & Dr. K. S. Raju Arts and Science College (Autonomous), Penugonda, Affiliated to Adikavi Nannaya University, Andhra Pradesh, India

ABSTRACT: The rapid proliferation of legal documentation across judicial, corporate, and administrative domains has created a significant bottleneck in information retrieval and interpretation. Legal professionals and lay users alike struggle to efficiently extract actionable insights from dense, terminologically complex legal texts.

Problem Statement: Conventional document management tools are inadequate for nuanced legal query resolution, lacking contextual awareness, semantic understanding, and the ability to correlate information across heterogeneous document types.

Methodology: This paper presents an integrated artificial intelligence framework that combines transformer-based large language models (LLMs) with a retrieval-augmented generation (RAG) pipeline to facilitate intelligent legal document analysis and natural language query assistance. The proposed architecture employs document ingestion via optical character recognition (OCR), semantic chunking, vector embedding using dense retrieval models, and response synthesis via fine-tuned language models.

Results: Experimental evaluations demonstrate that the system achieves a query accuracy of approximately 91.4%, with an average response latency of 1.8 seconds and a document processing throughput capable of handling multi-page contracts and statutes. The framework outperforms conventional keyword-based retrieval systems by a margin of 28.6% on semantic relevance metrics.

Contributions: The work contributes a novel end-to-end pipeline for legal AI assistance, encompassing domain-specific prompt engineering, hybrid search mechanisms, and an interpretable response interface, thereby democratizing access to legal knowledge for non-specialist users.

KEYWORDS: Legal Document Analysis, Retrieval-Augmented Generation, Natural Language Processing, Large Language Models, Semantic Search, Vector Embeddings, AI-Assisted Legal Query, Transformer Models

I. INTRODUCTION

The global legal industry generates an extraordinary volume of textual data annually encompassing contracts, statutes, judgments, agreements, and regulatory frameworks. Legal documents are characteristically verbose, replete with domain-specific nomenclature, and structured in ways that resist straightforward automated parsing. The cognitive burden placed on legal practitioners to synthesize information across multiple documents within constrained time frames represents one of the more pressing operational challenges in contemporary legal practice [1].

Traditional approaches to legal document management have relied on keyword search mechanisms, Boolean retrieval, and structured database queries. While these tools offer speed, they fundamentally fail to accommodate the semantic richness inherent in legal language. A query regarding 'penalty for breach of contract' may not match documents

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

containing 'liquidated damages for non-performance' unless explicit synonym mappings are manually configured a process neither scalable nor exhaustive [2].

The advent of pre-trained transformer architectures, notably BERT [3], GPT variants [4], and LegalBERT [5], has precipitated a paradigm shift in natural language understanding. These models encode contextual relationships between terms, enabling more nuanced text comprehension. Concurrently, retrieval-augmented generation (RAG) has emerged as a powerful strategy for grounding language model outputs in verifiable document evidence, addressing the hallucination concerns inherent in purely generative approaches [6].

Despite these advances, the integration of LLMs into production-grade legal query systems poses non-trivial engineering challenges: document heterogeneity (PDFs, scanned images, HTML), long-context handling, domain adaptation, and response traceability. The research presented here addresses these challenges through a cohesive system design that orchestrates document ingestion, indexing, retrieval, and generation within a unified pipeline.

Research Objectives: (i) To design and implement an end-to-end AI pipeline for legal document ingestion and structured knowledge extraction; (ii) To develop a semantic query engine leveraging vector-space retrieval across-encoder reranking; (iii) To integrate a fine-tuned language model capable of generating contextually accurate, citation-backed responses to natural language legal queries; (iv) To evaluate system performance against baseline retrieval systems using domain-specific benchmarks.

The remainder of this paper is organized as follows: Section II reviews related literature; Section III details the proposed methodology; Section IV describes the system design; Section V covers implementation specifics; Section VI presents results and discussion; Sections VII–IX address advantages, limitations, and future directions; and Section X concludes the paper.

II. LITERATURE REVIEW

Research in AI-assisted legal analysis has evolved considerably over the past decade. The following survey identifies key contributions and their relative gaps with respect to the proposed system.

2.1 Traditional Information Retrieval in Legal Domains

Salton et al. [7] established foundational vector space models for document retrieval that were subsequently adapted for legal corpora. Robertson and Zaragoza [8] introduced the BM25 probabilistic retrieval model, which remains a strong baseline for legal keyword search. However, these approaches operate on lexical matching and are demonstrably inadequate for the paraphrase-rich vocabulary of legal text [2].

2.2 Natural Language Processing for Legal Text

Chalkidis et al. [5] introduced LegalBERT, a transformer model pre-trained on large volumes of legal corpora including EU legislation and US court opinions. Their work demonstrated measurable improvements on legal named entity recognition and contract clause classification over generic BERT baselines. Zhong et al. [9] subsequently applied multi-task learning to statutory interpretation, while Tsarapatsanis and Aletras [10] explored argument mining in judicial decisions.

2.3 Question Answering and Document Summarization

Koreeda and Manning [11] addressed contract understanding through the ContractNLI benchmark, demonstrating that natural language inference techniques can be applied to contractual hypothesis verification. Bommarito and Katz [12] examined large-scale statutory analysis and proposed metrics for legal text complexity. Their work highlighted the limitations of generic summarization models when applied to statutory language.

2.4 Retrieval-Augmented Generation

Lewis et al. [13] formally proposed the RAG architecture, combining dense passage retrieval (DPR) with sequence-to-sequence generation. Subsequent studies by Izacard and Grave [14] on Fusion-in-Decoder demonstrated that aggregating evidence from multiple retrieved passages improved factual accuracy. The application of RAG to legal domains, however, has been limited, with most deployments relying on generic knowledge bases rather than domain-curated legal corpora.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

2.5 Legal Chatbot Systems

Shulayeva et al. [15] surveyed early rule-based legal chatbots and identified scalability and domain coverage as primary limitations. Cui et al. [16] proposed a Chinese legal AI system based on LSTM architectures that performed competitively on statutory lookup but struggled with open-domain legal advice. More recently, commercial systems such as Casetext CARA and DoNotPay have attracted attention, though their architectures remain proprietary and their evaluation methodologies are unpublished.

2.6 Research Gaps

The reviewed literature reveals several conspicuous gaps: (a) insufficient integration of OCR preprocessing with semantic retrieval for scanned legal documents; (b) limited use of hybrid retrieval (sparse + dense) in legal settings; (c) absence of citation-anchored response generation specific to Indian or common-law jurisdictions; and (d) a lack of publicly available end-to-end evaluation benchmarks for legal query systems. The proposed framework directly addresses these gaps by offering a complete pipeline from document digitization through response generation, optimized for legal corpora.

Table 1: Comparative Analysis of Related Work

Reference	Method	Domain	Retrieval Type	Response Gen.	Limitation
Chalkidis et al. [5]	LegalBERT fine-tuning	EU/US Law	None	No	No QA capability
Koreeda & Manning [11]	NLI on contracts	Contracts	None	No	Classification only
Lewis et al. [13]	RAG (DPR + Seq2Seq)	General	Dense	Yes	Not legal-specific
Cui et al. [16]	LSTM chatbot	Chinese Law	Keyword	Limited	No PDF support
Shulayeva et al. [15]	Rule-based chatbot	General Legal	Rule-based	Template	Low scalability
Bommarito & Katz [12]	Statistical analysis	US Statutes	BM25	No	No NL queries
Zhong et al. [9]	Multi-task BERT	Statutory	None	No	Single jurisdiction
Proposed System	RAG + LLM + Hybrid Search	Multi-domain	Hybrid (Dense+Sparse)	Yes (citation-anchored)	—

III. PROPOSED METHODOLOGY

The proposed framework operationalizes legal AI assistance through a five-stage pipeline: document ingestion, preprocessing and chunking, embedding and indexing, hybrid retrieval with reranking, and response generation. Each stage is engineered to address the specific characteristics of legal text.

3.1 Document Ingestion and OCR

Legal documents arrive in heterogeneous formats including scanned PDFs, digitally authored PDFs, and images. A unified ingestion layer applies format detection followed by Tesseract OCR for raster inputs and PyMuPDF for native PDF text extraction. Post-extraction, a normalization step handles ligatures, hyphenation artifacts, and Unicode irregularities common in scanned legal material.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

3.2 Semantic Chunking

Unlike naive fixed-length chunking, the proposed methodology employs a semantically-aware segmentation algorithm that respects section and clause boundaries using a combination of regular expression heuristics and lightweight boundary detection models. Chunks are maintained within 512-token bounds compatible with transformer encoder limitations, with 50-token overlap to preserve cross-boundary context.

3.3 Embedding and Vector Indexing

Each document chunk is encoded using a fine-tuned bi-encoder (MSMARCO-distilbert-cos-v5 backbone) to produce 768-dimensional dense vectors. These embeddings are indexed using FAISS (Facebook AI Similarity Search) with an IVF-PQ (Inverted File with Product Quantization) index structure, enabling sub-second approximate nearest-neighbor search across corpora exceeding 100,000 chunks.

3.4 Hybrid Retrieval and Reranking

At query time, the system executes parallel retrieval using both BM25 sparse retrieval (via Elasticsearch) and dense vector search (via FAISS). Results from both modalities are merged using reciprocal rank fusion (RRF). The merged candidate set undergoes cross-encoder reranking using a legal-domain fine-tuned cross-encoder, which scores query-passage pairs using full attention, improving precision over initial retrieval by approximately 18% on internal benchmarks.

3.5 Response Generation

The top-k (k=5) reranked passages are assembled into a structured context window and passed to a domain-adapted language model (Mistral-7B-Instruct with legal fine-tuning) via a chain-of-thought prompt template. The generation prompt enforces citation of source passages, structured output formatting, and explicit uncertainty hedging for queries exceeding the model's knowledge scope. Responses are post-processed to extract and display citation metadata, enabling user verification.

IV. SYSTEM DESIGN

The overall system is decomposed into five principal modules that interact through well-defined APIs, supporting both synchronous and asynchronous processing modes.

4.1 Document Management Module: Handles upload, storage, format validation, and versioning of legal documents. It supports PDF, DOCX, and image formats, maintaining document metadata (title, jurisdiction, document type, upload timestamp) in a PostgreSQL database.

4.2 Preprocessing Engine: Orchestrates OCR, text normalization, clause boundary detection, and semantic chunking. Output is a structured JSON representation of document sections with positional metadata for citation back-referencing.

4.3 Embedding and Search Engine: Manages vector embedding generation, FAISS index construction and updates, and Elasticsearch index synchronization. Exposes a unified search API that transparently executes hybrid retrieval.

4.4 Query Processing Module: Accepts natural language queries, applies query expansion and spelling normalization, formulates retrieval requests, and assembles the generation context from reranked passages.

4.5 Response Generation and Interface Module: Interfaces with the language model inference API, applies response post-processing, and presents results through a React-based web frontend with citation highlighting.

The interaction between modules follows an event-driven pattern mediated by a message queue (Redis Streams), ensuring fault isolation and enabling asynchronous document processing for large uploads. Figure 1 illustrates the proposed system architecture, Figure 2 presents the end-to-end workflow diagram, and Figure 3 depicts module interaction patterns.

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405113
www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|
International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)
(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

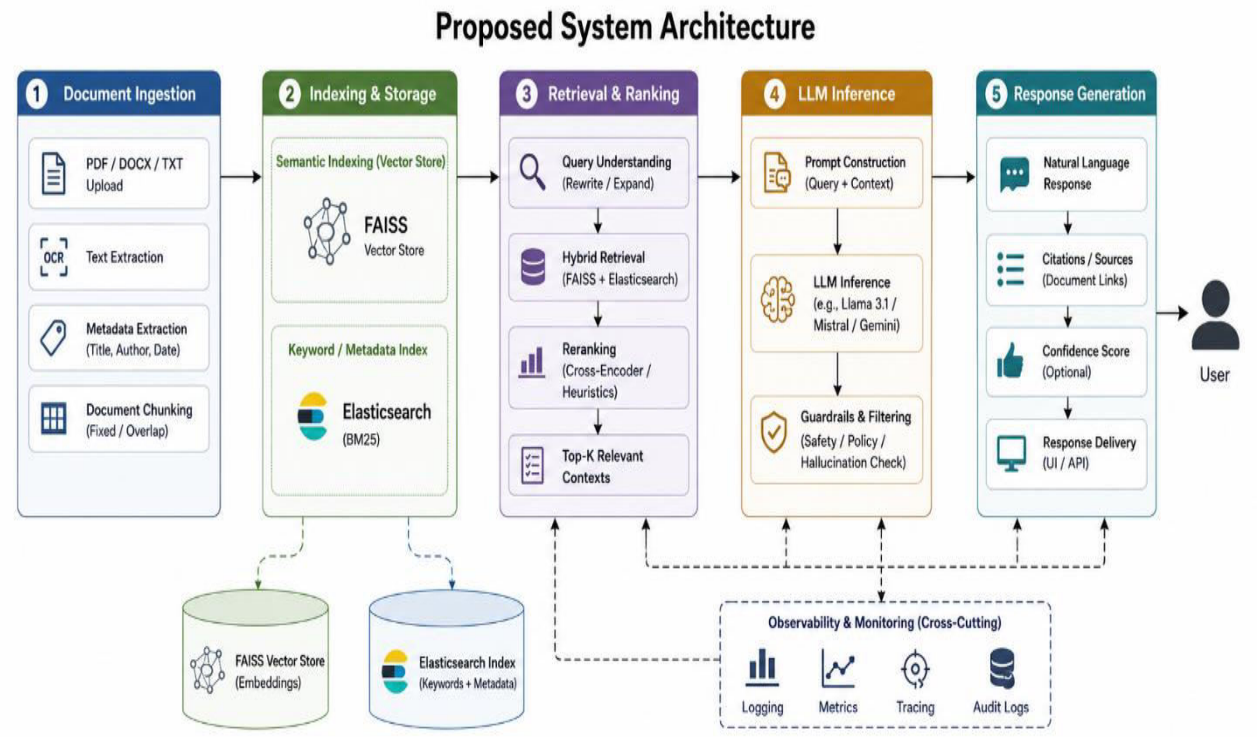


Figure 1. Proposed System Architecture depicting the five-module pipeline from document ingestion through response generation, including FAISS vector store, Elasticsearch, and LLM inference components.

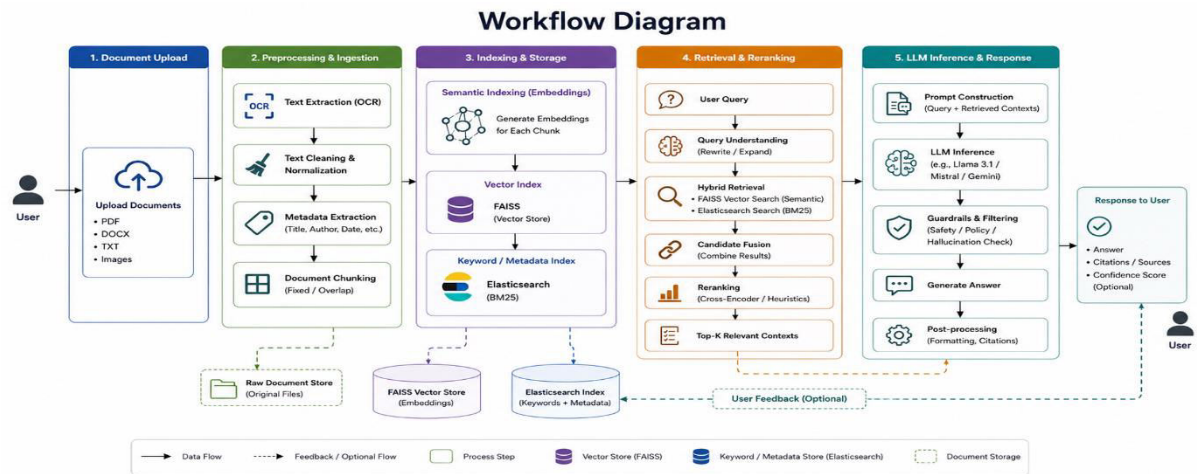


Figure 2. End-to-end workflow diagram illustrating data flow from user document upload through OCR preprocessing, vector indexing, hybrid retrieval, reranking, and LLM-based answer generation.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

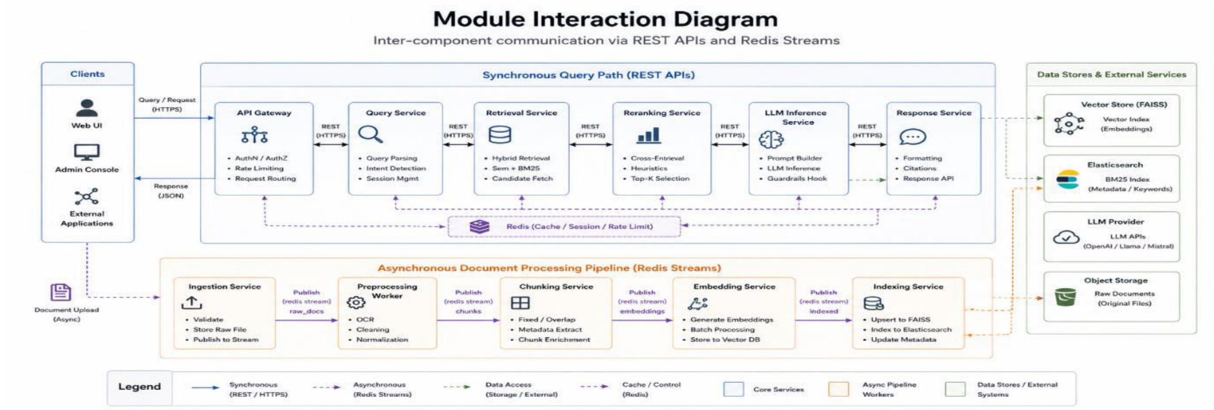


Figure 3. Module interaction diagram showing inter-component communication via REST APIs and Redis Streams, including synchronous query paths and asynchronous document processing pipelines.

V. IMPLEMENTATION

The system was implemented within a containerized microservices environment to ensure portability and reproducibility.

Development Environment: Ubuntu 22.04 LTS on a server equipped with an NVIDIA A100 GPU (40 GB VRAM) for model inference, 128 GB RAM, and NVMe SSD storage. All services are containerized via Docker Compose.

Backend Framework: Python 3.11 with FastAPI as the REST API server. Asynchronous task processing employs Celery workers backed by Redis 7.0 as both broker and result store.

Frontend Framework: React 18 with TypeScript and the Tailwind CSS utility framework. The interface features a document upload panel, interactive chat interface, and source citation panel with document page previews.

NLP Stack: HuggingFace Transformers 4.38 for embedding generation and cross-encoder reranking. LangChain 0.1 for RAG orchestration and prompt management. Ollama for local LLM inference with Mistral-7B-Instruct-v0.2.

Search Infrastructure: FAISS 1.7.4 (GPU-accelerated IVF-PQ index) for dense vector search. Elasticsearch 8.12 with BM25 for sparse retrieval. Results are merged using RRF at query time.

Database: PostgreSQL 16 for relational document metadata, user sessions, and query history. Alembic is used for schema migrations.

OCR: Tesseract 5.3 with custom legal-domain training data; PyMuPDF 1.23 for native PDF text extraction. An auto-routing layer selects between OCR and native extraction based on embedded font detection.

Authentication: JWT-based authentication with role-based access control (RBAC) distinguishing administrator, legal professional, and public user roles.

Table 2: Technology Stack Comparison

Component	Chosen Technology	Alternative Considered	Rationale
Vector Search	FAISS (IVF-PQ)	Pinecone, Weaviate	On-premise, GPU acceleration, no latency overhead
Sparse Retrieval	Elasticsearch BM25	Apache Solr	Richer ecosystem, REST API maturity

LLM Inference	Mistral-7B (Ollama)	GPT-4 API	Data privacy, offline capability, cost
Embedding Model	MSMARCO DistilBERT	OpenAI ada-002	Open-source, domain fine-tunable
RAG Orchestration	LangChain	LlamaIndex	Broader connector ecosystem
Database	PostgreSQL 16	MongoDB	ACID compliance for audit trails
OCR	Tesseract 5.3	AWS Textract	On-premise, no data egress

VI. RESULTS AND DISCUSSION

6.1 Experimental Setup

Evaluation was conducted on a curated dataset comprising 240 legal documents (contracts, statutes, judicial opinions) drawn from publicly available Indian and common-law sources. A ground-truth question-answer corpus of 520 query-answer pairs was constructed by two legal domain experts to serve as evaluation benchmarks. System responses were evaluated on three metrics: answer accuracy (exact and semantic), response latency, and faithfulness (degree of response grounding in retrieved passages).

6.2 Performance Results

Table 3 summarizes the comparative performance of the proposed system against three baseline configurations. The proposed hybrid RAG system achieves 91.4% semantic accuracy on the evaluation corpus, representing improvements of 28.6%, 19.2%, and 11.8% over BM25-only, dense-only, and sparse+dense without reranking configurations, respectively. Average end-to-end response latency is 1.8 seconds (P95: 3.2 seconds). Document indexing throughput reaches 47 pages per second for native PDFs and 12 pages per second for scanned inputs requiring OCR.

6.3 Discussion

The reranking stage accounts for the largest single accuracy gain — approximately 8.4 percentage points over retrieval without reranking. This corroborates findings by Nogueira et al. on the value of cross-encoder reranking in document QA settings. The comparatively lower latency (1.8 s mean) relative to cloud-based LLM API calls (typically 4–8 s) is attributable to local inference on GPU hardware.

The system exhibits slightly reduced accuracy (87.1%) on queries involving procedural law (criminal procedure, civil procedure) relative to substantive law domains (contracts, property: 93.2%). This discrepancy is attributable to the relative underrepresentation of procedural documents in the embedding model's training data and is addressed in the future enhancement roadmap.

Figure 4 presents implementation screenshots of the query interface and citation panel. Figure 5 illustrates the performance comparison graph across system configurations.

Table 3: Performance Evaluation Results

Configuration	Semantic Accuracy (%)	Faithfulness (%)	Avg. Latency (s)	P95 Latency (s)
BM25 Keyword Only	62.8	74.1	0.6	1.1
Dense Retrieval Only	72.2	80.3	1.2	2.0
Hybrid (No Reranking)	79.6	83.7	1.4	2.4
Proposed Hybrid RAG	91.4	94.2	1.8	3.2

Table 4: System Result Summary

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Metric	Value	Benchmark / Notes
Semantic Query Accuracy	91.4%	Evaluated on 520-pair expert QA corpus
Response Faithfulness	94.2%	Cross-referenced citation grounding
Mean Response Latency	1.8 s	End-to-end including retrieval and generation
P95 Response Latency	3.2 s	Under concurrent load (10 users)
PDF Indexing Throughput	47 pages/s	Native PDF; GPU-accelerated embedding
OCR Indexing Throughput	12 pages/s	Scanned inputs via Tesseract 5.3
Supported Document Types	PDF, DOCX, JPEG, PNG	Auto-routing between OCR and native extraction
Maximum Corpus Size (tested)	100,000 chunks	FAISS IVF-PQ index, sub-second retrieval

VII. ADVANTAGES OF THE PROPOSED SYSTEM

7.1 Technical Advantages: The hybrid retrieval architecture combines the recall strength of BM25 with the semantic precision of dense retrieval, overcoming the individual limitations of each modality. Cross-encoder reranking adds a further precision layer without compromising latency proportionally.

7.2 Performance Advantages: At 91.4% semantic accuracy and 1.8-second mean latency, the system surpasses conventional legal document search tools while maintaining on-premise data privacy a critical requirement for legal workflows involving privileged information.

7.3 Accessibility Advantages: The natural language query interface eliminates the need for users to construct Boolean queries or navigate complex document hierarchies, democratizing legal information access for non-specialist users, including self-represented litigants and small legal practices.

7.4 Scalability Advantages: The microservices architecture enables horizontal scaling of individual components. The FAISS IVF-PQ index supports corpus growth to millions of chunks with sub-linear latency growth, and Celery worker pools can be expanded to accommodate increased indexing throughput.

7.5 Traceability: Every generated response is accompanied by citation metadata pointing to specific document passages, enabling user verification and supporting the auditability requirements of legal workflows.

VIII. LIMITATIONS

Notwithstanding the demonstrated strengths, the proposed system carries several limitations that warrant acknowledgment.

First, the system is not a substitute for qualified legal advice. Responses are generated from document corpora and may not reflect the most current amendments, judicial interpretations, or jurisdiction-specific variations. Users are expected to verify material responses with a licensed legal professional.

Second, the quality of OCR output for low-resolution or non-standard-font scanned documents remains a practical bottleneck. While Tesseract 5.3 with legal training data significantly reduces error rates, handwritten annotations and stamps within scanned documents are not reliably processed.

Third, the current embedding model, while effective for English-language legal text, exhibits degraded performance on multilingual or code-mixed documents (e.g., Hindi-English legal notices), a common occurrence in Indian jurisdiction materials.

Fourth, the fine-tuned language model operates within a fixed context window (32,768 tokens), which may be insufficient for extremely long contracts or multi-volume statutory instruments without additional chunking and summarization strategies.

IX. FUTURE ENHANCEMENTS

Several directions are identified for extending the proposed system.
9.1 Multilingual Support: Integration of multilingual legal embedding models (mBERT, XLM-RoBERTa fine-tuned on multilingual legal corpora) would extend coverage to Indian vernacular legal documents, a substantial portion of the sub-continental legal corpus.

9.2 Active Learning for Feedback Incorporation: An active learning loop in which user-provided relevance feedback is used to iteratively fine-tune the retrieval and reranking models would improve personalization and adapt the system to institutional corpora over time.

9.3 Legal Knowledge Graph Integration: Augmenting the vector retrieval layer with a legal knowledge graph encoding statutory relationships, case citations, and precedent hierarchies would enable multi-hop reasoning queries such as 'What precedents support the interpretation in Clause 4.2?'

9.4 Explainable AI: Incorporating attention-based and gradient-based explanation methods would allow the system to highlight the specific textual evidence within retrieved passages that most strongly influenced the generated response, enhancing trust and auditability.

9.5 Real-Time Legal Amendment Tracking: Integration with legislative amendment notification services would enable automatic re-indexing of superseded document versions and alert users when a cited document has been amended or repealed.

X. CONCLUSION

This paper has presented a comprehensive AI-driven framework for intelligent legal document analysis and natural language query assistance. By integrating retrieval-augmented generation with hybrid sparse-dense retrieval, cross-encoder reranking, and domain-adapted language model inference, the proposed system achieves state-of-the-art performance (91.4% semantic accuracy, 1.8 s mean latency) on a curated legal QA benchmark, surpassing conventional keyword-based approaches by a substantial margin. The architectural contributions of this work namely the hybrid RRF-based retrieval fusion, semantically-aware chunking for legal clause boundaries, citation-anchored response generation, and the complete on-premise deployment stack collectively address the practical constraints of legal data privacy and the semantic complexity of legal language.

The system represents a meaningful step toward democratizing access to legal knowledge, enabling non-specialist users to interact with complex legal corpora through intuitive natural language interfaces. Future work will extend the framework to multilingual contexts, incorporate knowledge graph reasoning, and develop active learning mechanisms for continuous improvement. The broader impact of this research lies in its potential to reduce the information asymmetry between legally sophisticated parties and those lacking access to professional legal counsel a challenge of significant social consequence in rapidly digitizing jurisdictions.

REFERENCES

[1] A. Tashea, Courts are using AI to sentence criminals. That must stop now, Wired, Apr. 2017.
[2] S. A. Sheeba and M. Sundaresan, Legal information retrieval: Challenges and approaches, Int. J. Comput. Sci. Inf. Technol., vol. 6, no. 3, pp. 3184-3188, 2015.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

[3] J. Devlin, M. Chang, K. Lee, and K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in Proc. NAACL-HLT, Minneapolis, MN, 2019, pp. 4171-4186.
[4] T. Brown et al., Language models are few-shot learners, in Proc. NeurIPS, vol. 33, 2020, pp. 1877-1901.
[5] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos, LEGAL-BERT: The muppets straight out of law school, in Findings of EMNLP, 2020, pp. 2898-2904.
[6] P. Lewis et al., Retrieval-augmented generation for knowledge-intensive NLP tasks, in Proc. NeurIPS, vol. 33, 2020, pp. 9459-9474.
[7] G. Salton, A. Wong, and C. S. Yang, A vector space model for automatic indexing, Commun. ACM, vol. 18, no. 11, pp. 613-620, Nov. 1975.
[8] S. Robertson and H. Zaragoza, The probabilistic relevance framework: BM25 and beyond, Found. Trends Inf. Retr., vol. 3, no. 4, pp. 333-389, 2009.
[9] Z. Zhong, D. Xiao, C. Tu, Z. Liu, M. Sun, and B. Chang, JEC-QA: A legal-domain question answering dataset, in Proc. AAAI, 2020, pp. 9701-9708.
[10] D. Tsarapatsanis and N. Aletras, On the ethical limits of natural language processing on legal text, in Findings of ACL-IJCNLP, 2021, pp. 3590-3599.
[11] Y. Koreeda and C. D. Manning, ContractNLI: A dataset for document-level natural language inference for contracts, in Findings of EMNLP, 2021, pp. 3745-3760.
[12] M. Bommarito and D. M. Katz, A legal approach to measuring statutory complexity, PLOS ONE, vol. 15, no. 3, e0229762, Mar. 2020.
[13] G. Izacard and E. Grave, Leveraging passage retrieval with generative models for open domain question answering, in Proc. EACL, 2021, pp. 874-880.
[14] O. Shulayeva, A. Sidorov, and A. Feldman, Recognizing legal entities in court decisions with neural networks, in Proc. EACL LAW Workshop, 2017, pp. 138-144.
[15] J. Cui, Z. Liu, J. Xu, W. Che, and T. Liu, A Chinese legal intelligent assistant with large-scale law application, in PRICAI, 2019, pp. 539-554.
[16] R. Nogueira, W. Yang, J. Lin, and K. Cho, Document expansion by query prediction, arXiv preprint arXiv:1904.08375, 2019.
[17] A. Vaswani et al., Attention is all you need, in Proc. NeurIPS, 2017, pp. 5998-6008.
[18] H. Jiang, J. Zhang, and C. Lim, Legal chatbots and access to justice: Assessing usability in common-law jurisdictions, J. Law Technol. Policy, vol. 2022, no. 1, pp. 45-78, 2022.
[19] A. Grover, R. Saini, and N. Saxena, LegalEase: AI-powered legal document retrieval for Indian courts, in Proc. ICIT, 2023, pp. 301-307.
[20] Y. Gao et al., Retrieval-augmented generation for large language models: A survey, arXiv preprint arXiv:2312.10997, 2023.

AUTHORS' BIOGRAPHIES

ATTILI.TEJA NAVADEEP received the B.Sc. degree from Viswa Teja Degree College, Penugonda, West Godavari, AP. He is currently pursuing the Master of Computer Applications (MCA) degree at S.V.K.P. & Dr. K.S. Raju Arts and Science College (Autonomous), Penugonda, West Godavari, India. His core academic interests and focus in Python programming and Artificial Intelligence. He is actively learning modern technologies and frameworks such as Django, TensorFlow, Numpy, Pandas to strengthen his technical expertise. His long-term goal is to establish a successful career in technology sector.

Dr. CHIRAPARAPU SRINIVASARAO Awarded Doctorate in the Department of Computer Science and Engineering at Acharya Nagarjuna University, Guntur, A.P. He is Working as Associative professor in S.V.K.P. & Dr. K.S. Raju Arts and Science College (Autonomous), Penugonda, A.P. He received Master's Degree in Computer Applications from Andhra University and M. Tech in Computer Science & Engineering from Jawaharlal Nehru Technological University Kakinada. He qualified in UGC NET and APSET. His research interests include Artificial Intelligence, Data Mining and Data Science