iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.78630

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Multi-Agent Retrieval Augmented Generation System for Legal Applications: A Neuro-Symbolic Approach

Mustipalli Naveen[1], Komera Ashok[2], Madana Akash Reddy[3], Madisetty Gowrinadh[4]
[1, 2, 3, 4]*BTech Students, Department of CSE-AI KKR & KSR Institute of Technology and Sciences, Guntur, Andhra Pradesh, India*

Abstract: ***Legal field is the area where special difficulties are observed because it concerns a lot of documentation, com- plicated reasoning and the necessity to be very accurate and explainable. The conventional legal research process is both time-intensive and inefficient, and available AI-based systems are widely susceptible to hallu- cinations and unprovable basis. The Intelligent Multi-Agent Retrieval Augmented Generation (RAG) system suggested in this paper is specifically aimed at Indian law applications. The system uses large language models to combine the power of large language models with a verification and legal reasoning system using special- ized agents to retrieve documents. One of the major inventions is the debate-based legal reasoning module, where the independent prosecution and defense agents present conflicting arguments to a monitoring agent in an organized pros and cons format with specific citation. The system avails persistent, downloadable, and shareable AI-generated legal argument, and allows artifacts of legal argument to be reused and audited. The system can be explained and justified by basing all generated responses on confirmed legal documents such as the Constitution of India, Bharatiya Nyaya Sanhita (BNS), and case laws, which makes it accurate, reliable, and explainable, and greatly reduces hallucinations in comparison with single-language models and the use of one agent RAG systems.***
Keywords: Retrieval Augmented Generation, Multi-Agent Systems, Legal AI, Natural Language Processing, Large Language Models, Neuro-Symbolic AI, Indian Legal System, Document Analysis.

I. INTRODUCTION

Judicial decision-making, legal advisory services, and compliance management are based on legal research. Indian legal system with its expansive list of statutes, case laws and constant legislative reforms like the substitution of the Indian Penal Code (IPC) with the Bharatiya Nyaya Sanhita (BNS) are quite chal- lenging to the legal practitioners and to the general population.[16].

The field of law is knowledge-based and paper- based in nature wherein one is constantly exposed to large amount of textual material such as statutes, le- gal judgments, contracts, legal notices and regulatory provisions. This information requires legal experts to critically read and decipher it so that they can make the right decisions. Nevertheless, the traditional law research methods are mostly manual or use simple digital tools based on key words, which are time con-suming, ineffective and susceptible to human errors. Due to the rapid development of Artificial Intelli-gence (AI) and Natural Language Processing (NLP), intelligent systems have been added to help in legal research [2]. Notwithstanding this development, the majority of current AI-related legal tools have severe weaknesses. Most systems can either call documents without comprehending legal context or can be able to produce a response based on language models with- out checking the origin of information. This tends to create hallucinated reactions, transparency lapses and diminished confidence, which are not admissible in legal settings that involve high stakes. [3, 11].

A. Problem Statement

Legal consultation is also a major challenge in India in terms of accessibility. The cost of legal services is high and not affordable to a great number of citizens.

Moreover, generic AI models often delirate legal pas- sages or give reference to nonexistent cases, and thus, they are unsafe insofar as legal advice is concerned. The available tools of legal research are complicated and need expert training to operate.

The chatbots of old tend to have hallucinations when speaking about certain laws, and the search en- gines that are based on key fields do not have the power to reason and provide the legal study that is intricate. Without a system that combines correct document retrieval, contextual legal reasoning, multi- source analysis and explainable response generation, there is an essential gap in legal technology.

B. Proposed Solution

In order to address these problems, the current project suggests Intelligent Multi- Agent Retrieval Aug- mented Generation (RAG) System that is specifically aimed at Indian law. The suggested system incorpo- rates the advantages of document retrieval and gener- ative AI and reduces their drawbacks with the help of a multi-agent system.

Various specialized intelligent agents are used in the system and each is charged with certain tasks:

- Researcher Agent:Searches semantically a vec- tor database of accurate sections of the Constitu- tion, BNS, BNSS and case laws
- Analyst Agent: Uses the legal reasoning to ex- amine the situation or document of the user.
- Drafter Agent: Gathers results and gives sim- plified and practical recommendations with refer- ences.

The system is reliable, full of grounded responses, accurate and explainable by the fact that all of the responses generated are based on verified legal doc- uments hence they can be applied in legal practice. This makes the law more democratic as complex cases and documents are deconstructed and simpli- fied advice has been given with reference to specific Indian legal provisions.

C. Research Objectives

The proposed study will incorporate an intelligent multi-agent RAG system specifically designed to sup- port legal uses and facilitate semantic retrieval of legal texts and produce context-aware and citation- supported legal answers, minimise AI hallucinations by grounding, enhance legal research efficiency, and offer scalable modular system architecture with sym- bolic reasoning verification to remove logical incon- sistencies.

D. Contributions

The study presents a new multi-agent system over Indian law scenarios that integrates RAG with spe- cialized agent functions, a discussion based reason- ing system with prosecution and defense agent agents independent, a structured monitoring system repre- senting arguments in clear pros and cons, a persis- tence layer allowing storage and transfer of arguments (PDF/text), a hybrid neuro-symbolic system that com- bines deep learning with formal rule checking of the law, and ways of deleting hallucinations by checking against a formal system of logic. The system offers reusable legal reasoning artifacts that can be audited and distributed among legal professionals, and offers a practical system of democratizing knowledge of the law in India.

II. LITERATURE REVIEW

A. Legal Information Retrieval

The study of legal information retrieval is not a new research topic that has been going on over the decades. The conventional methods were too depen- dent on Boolean searching and match with keywords which did not usually seize the semantic sense and contextual relations of legal texts. [**?**].

The recent developments in natural language pro- cessing have facilitated more complex retrieval pro- cesses. Neural embeddings and vector space models have been demonstrated to be useful in encoding se- mantic similarities between legal documents.[**? ?**]. Nonetheless, these strategies are prone to challenges of vocabulary and argument structure peculiar to legal language.

B. Large Language Models in Legal Domain

Large Language Models (LLMs) such as GPT, BERT, and their variants have demonstrated impressive capa- bilities in understanding and generating natural lan- guage [**?**]. Several studies have explored the ap- plication of LLMs to legal tasks, including contract analysis, legal question answering, and case outcome prediction [1**?**].However, standalone LLMs face sig- nificant challenges in legal applications. The primary concerns include:

of plausible and yet legally false data (hallucina- tions)[**?**]

- Absence of explanations and reference of sources. [**?**]
- Failure to retrieve updated legal information be- yond the training information. biases of making law and legal advice.

C. Retrieval Augmented Generation

To overcome the drawbacks of standalone language models, Retrieval Augmented Generation (RAG) was developed as a solution to the limitations of stan- dalone models, which operate independently of re- trieval signals [**?**].RAG systems involve information retrieval together with text generation whereby gener- ated answers are based on factual documents that are retrieved [**?**].

RAG paradigm is divided into three elements, the retriever, which identifies the pertinent documents, the reader/generator, which processes the retrieved documents and a system to combine the retrieved in- formation in the generated answer.
This method has led to great improvements in factual accuracy and de- creased hallucinations in different fields.

D. Multi-Agent Systems

It has been discussed that multi-agent systems have been used in a range of AI applications with benefits in the form of modularity, specialization, and collab- orative problem-solving [6]. Multi-agent techniques can also replicate the orchestration of the legal prac- tice in the context of legal AI, where various special- ists are being involved to apply their expertise to ad- dress complex legal issues.
Recent research has shown that specialized agents are useful in complex reasoning problems[17]. Com- plex issues can be broken down into small manage- able components by using agent-based systems, each agent concentrating on its specialty.

E. Neuro-Symbolic AI

Neuro-symbolic AI is an intermediate style which unites the learning of neural networks with the reason- ing of symbolic systems and comprehensibility. [7]. Neuro-symbolic AI is especially useful in legal tasks where pattern recognition and rule-based reasoning are both required.
Neuro-symbolic AI models that combine sym- bolic logic engines with neural feature extraction can ensure that the generated output satisfies the legal rules and constraints, thus removing any inconsisten- cies and hallucinations..

F. Gaps in Existing Research

Despite the progress, there are still some research gaps in legal AI:

- Less emphasis on jurisdiction-based systems, es- pecially in the Indian legal setting [9, 16]
- Less emphasis on explainability and trustworthi- ness in legal AI systems [14]
- Less emphasis on holistic systems that combine retrieval, reasoning, and verification [18]
- Less emphasis on multi-agent models for legal tasks
- Less emphasis on addressing the dynamic nature of legal systems [16]

This study fills the gaps by proposing a compre- hensive multi-agent RAG system specifically for In- dian legal tasks.

III. SYSTEM ARCHITECTURE

A. Overview

The proposed system has a multi-agent architecture that coordinates multiple specialized AI agents to carry out various tasks of legal research and analy- sis. The architecture is made to be modular, scalable, and transparent so that each part of the architecture can be developed, tested, and improved separately. Figure 1 below shows the complete system architec- ture starting from the user queries to the MCP Server, Central Controller, specialized agents, and legal doc- ument repositories.

B. System Components

The system has the following main components
Document Ingestion and Preprocessing
The system begins with document ingestion, where legal documents are captured through various means:

- Digital file uploads (PDF, DOCX, JPEG)
- Scanner or camera-based capture
- Direct database imports of legal codes and case laws

Preprocessing steps include:

1) Normalization of documents (resizing, noise re- moval, deskewing)
2) Text extraction from scanned documents using OCR
3) Structure extraction (section, clause, citation identification)
4) Metadata extraction (date, jurisdiction, type of document)

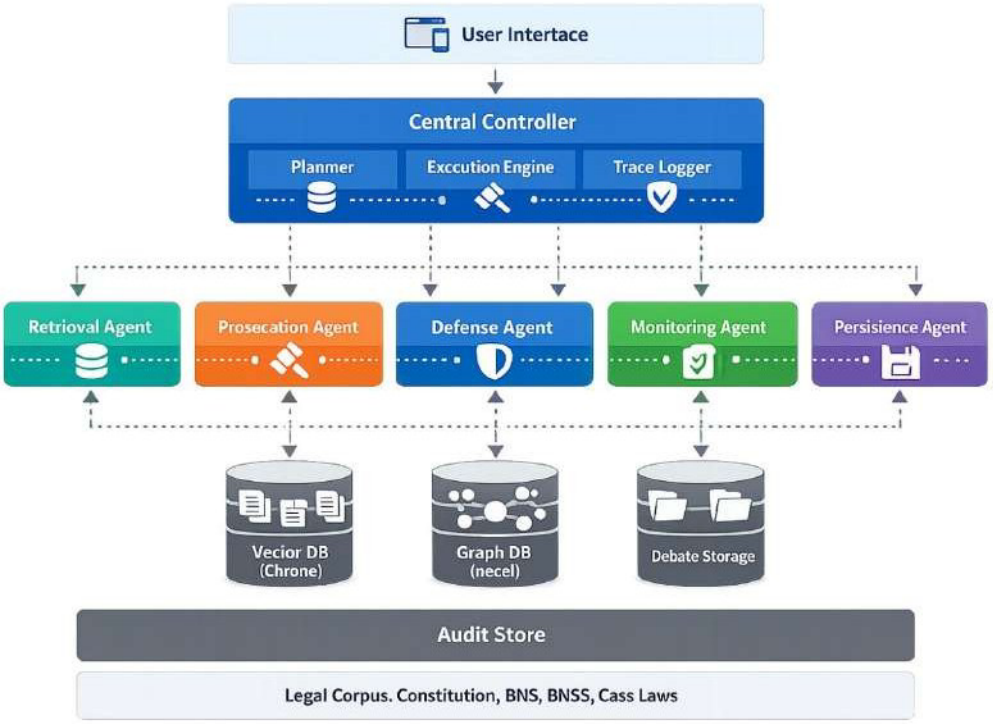


Figure 1: Multi-Agent RAG System Architecture that will be used to visualize communication between user interface, central controller, specialized agents (Retrieval, Prosecution, Defense, Monitoring, Persistence), and legal document repositories.

a) Vector Database and Embedding

The legal documents are processed and stored in a vector database that supports semantic search. The steps involved are:

- Chunking of documents into semantically mean- ingful chunks
- Creation of dense vector embeddings through transformer models
- Indexing in a high-performance vector database (Pinecone, Weaviate, or FAISS)
- Metadata labeling for filtering and hybrid search The vector database indexes legal codes such as the Constitution of India, Bharatiya Nyaya Sanhita (BNS), Bharatiya Nagarik Suraksha Sanhita (BNSS), and case laws.

b) Multi-Agent Framework

The core of the system is the multi-agent framework consisting of five specialized agents:

Retrieval Agent:

- Receives user queries in natural language
- Performs semantic search on the vector database
- Retrieves relevant legal sections and case prece- dents
- Ranks results by relevance and applicability
- Returns top-k documents with metadata

Prosecution Agent:

- Constructs arguments supporting liability or vio- lation
- Identifies legal grounds for enforcement
- Cites relevant statutes and precedents supporting prosecution
- Builds case for one side of legal interpretation
- Works independently from defense perspective

Defense Agent:

- Develops counter-arguments to mitigate liability
- Identifies legal defenses and exceptions

- Cites relevant statutes and precedents supporting defense
- Builds alternative legal interpretations
- Works independently from prosecution perspec- tive

Monitoring Agent:

- Reviews arguments from both prosecution and de- fense
- Validates citations and cross-references
- Structures outputs into explicit pros and cons
- Identifies strengths and weaknesses of each argu- ment
- Creates balanced, transparent legal analysis

Persistence Agent:

- Stores AI-generated debates with metadata
- Enables export to text and PDF formats
- Manages sharing and access control
- Maintains audit trail of legal reasoning
- Facilitates reuse of legal analysis artifacts

c) Symbolic Reasoning Layer

To minimize hallucinations and maintain logical rigor throughout the reasoning process, the system inte- grates a dedicated symbolic reasoning layer. Rather than relying solely on generative outputs, this layer acts as a structured verification mechanism that rein- forces accuracy and coherence.

- Systematically verifies extracted facts against for- mally encoded legal rules and principles
- Detects and flags logical inconsistencies or con- tradictions within generated responses
- Validates legal citations and cross-references to ensure correctness and traceability
- Enforces compliance with predefined legal and regulatory constraints
- Produces transparent and explainable reasoning chains that make each conclusion auditable

d) Debate Generation Module

One of the system's most distinctive capabilities is its debate-driven legal reasoning framework, which structures analysis as an adversarial yet controlled ex- change between opposing perspectives.

- The prosecution and defense agents analyze the same retrieved evidence, but do so independently to preserve perspective integrity
- Each agent develops arguments strictly from its assigned role, ensuring clarity of position
- All arguments are anchored in the same underly- ing legal texts, maintaining factual alignment and consistency
- Independent generation encourages the explo- ration of nuanced and alternative legal interpre- tations
- The structured debate format is intentionally de- signed to reflect the dynamics of real-world legal proceedings

The monitoring agent organizes debate outputs into explicit components:

$$A = \{P^+, P^-, D^+, D^-\} \quad (1)$$

In which, P+ and P do represent the strengths and the weakness of the prosecution arguments, and D + and D do the strengths and the weaknesses of the de- fense arguments. Such organized depiction enhances clarity and comprehension.

e) Persistence and Sharing Layer

The system provides mechanisms for storing, down- loading, and sharing legal debates:

- Storage: Every AI-generated debate is systemat- ically archived along with comprehensive meta- data, including citations, timestamps, and the original query context to ensure traceability.
- Export Formats: Debates can be exported either as well-structured text files or as professionally formatted PDF documents suitable for formal re- view and presentation.
- Sharing Capabilities: Users are able to share generated debates with colleagues, clients, or within academic and training environments when collaboration or review is required.

- Audit Trail: A complete and transparent history of the legal reasoning process is preserved, sup- porting accountability and retrospective analysis.
- Reusability: Previously generated debates re- main accessible and can be referenced or adapted when addressing comparable legal scenarios.
- Access Control: Role-based permission settings safeguard sensitive legal analyses by ensuring that only authorized individuals can access specific materials.

f) User Interface

- A web-based dashboard that provides an intuitive environment for general users
- API endpoints that allow seamless integration with external platforms and enterprise systems
- A mobile application to ensure accessibility and on-the-go usage
- A professional-grade interface equipped with ad- vanced analytical tools tailored for legal practi- tioners

C. System Workflow

The system follows a structured and transparent workflow to ensure both analytical rigor and trace- ability. A typical interaction proceeds as follows:

1) The user submits a legal query or uploads a docu- ment requiring analysis
2) The Retrieval Agent conducts a semantic search and gathers relevant legal texts and precedents
3) The retrieved materials are simultaneously pro- vided to both the Prosecution and Defense Agents
4) The Prosecution Agent develops arguments sup- porting a particular legal interpretation
5) The Defense Agent independently formulates counter-arguments
6) Both agents rely on the same retrieved legal sources, interpreting them from opposing per- spectives
7) The Monitoring Agent reviews both argument sets to identify logical gaps or inconsistencies
8) The Monitoring Agent organizes the outputs into structured pros and cons (P^+, P^-, D^+, D^-)
9) The Symbolic Reasoning Layer validates citations and evaluates the logical integrity of all claims
10) The finalized debate is compiled along with com- plete metadata and references
11) The Persistence Agent securely stores the debate for future retrieval
12) The user receives a clearly structured legal analy- sis with options for download and sharing
13) The debate can be exported in PDF or structured text format
14) A shareable link can be generated, subject to de- fined access permissions

Figure 2 presents the complete system workflow from user query submission to final debate export, showing the sequential and parallel processing steps through all specialized agents.

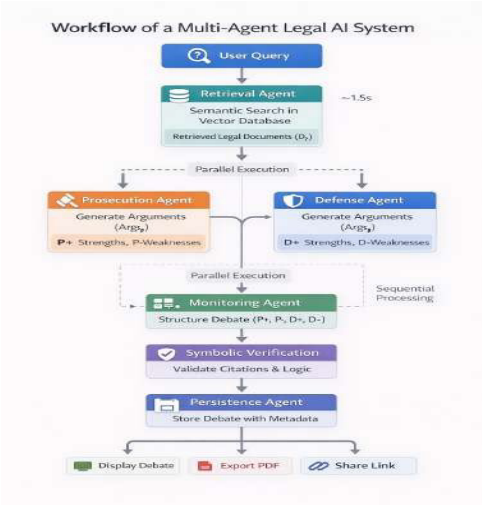


Figure 2: Complete system workflow diagram showing the sequential and parallel processing flow from user query through retrieval, debate generation by prosecution and de- fense agents, monitoring, verification, and final export op- tions (PDF/text/share).

D. Technical Architecture

The technical architecture includes:

- Frontend: A React-based web application that provides an interactive debate viewing interface, allowing users to navigate arguments, citations, and structured outputs with clarity.
- Backend: A Python-based backend built on the FastAPI framework, ensuring high performance, scalability, and efficient API handling.
- LLM Integration: Seamless API integration with state-of-the-art language models to power ar- gument generation and analytical reasoning.
- Controlled Inference Pipelines: Carefully con- figured inference settings, including temperature tuning, token limits, and advanced prompt engi- neering techniques to maintain precision and con- sistency.
- Vector Database: A high-performance vector storage and retrieval system optimized for seman- tic search across legal documents.
- Agent Orchestration: Multi-agent coordina- tion implemented using frameworks such as LangChain or CrewAI to ensure structured collab- oration between agents.
- Central Controller: A supervisory component responsible for managing execution order, enforc- ing context boundaries, and coordinating agent workflows.
- Debate Engine: A custom-built module dedi- cated to generating, structuring, and formatting adversarial legal debates.
- Symbolic Engine: A logic programming system (e.g., Prolog or similar) used to perform rule- tbased validation and consistency checks.
- Persistence Layer: A robust storage layer utiliz- ing PostgreSQL or MongoDB to archive debates and associated metadata.
- Export Module: Integrated PDF generation (e.g., ReportLab or WeasyPrint) and structured text for- matting for professional output.
- Sharing System: A secure token-based sharing mechanism combined with role-based access con- trol.
- Security: Comprehensive protection mecha- nisms, including end-to-end encryption, multi- factor authentication, and granular access control.

1) Central Controller Architecture

The central controller serves as the orchestration backbone of the multi-agent workflow, ensuring that each component operates in a coordinated and con- trolled manner:

- Execution Sequencing: Determines and enforces the precise order in which agents are invoked throughout the workflow.
- Context Management: Continuously tracks con- versation history while maintaining strict context size boundaries.
- Token Budget Control: Monitors and strategi- cally allocates token usage across agents to opti- mize performance and prevent overflow.
- Error Handling: Detects failures and activates predefined fallback strategies to maintain system stability.
- Quality Assurance: Reviews and validates agent outputs before forwarding them to subsequent stages.
- Concurrent Execution: Supports parallel pro- cessing, particularly enabling prosecution and de- fense agents to operate simultaneously for effi- ciency.

Context Window Management:

The central controller maintains a context window C with a maximum capacity of $C_{\max}$. Across a se- quence of agent interactions, the context is dynam- ically monitored and adjusted to ensure that the to- tal accumulated information remains within allowable limits while preserving essential reasoning history.

$$C_t = \begin{cases} C_{t-1} \cup \{o_t\} & \text{if } |C_{t-1} \cup \{o_t\}| \le C_{\max} \\ \text{trim}(C_{t-1}) \cup \{o_t\} & \text{otherwise} \end{cases} \quad (2)$$

where o is the output at time t, and trim moves the oldest entries to maintain the size con- straint.

Token Budget Allocation:

For *m* agents, the token budget is distributed as:

$$B_{\text{total}} = \sum_{i=1}^{m} B_i + B_{\text{overhead}} \quad (3)$$

where B_i is the budget for agent i and B_{overhead} ac- counts for system prompts and metadata. The con- troller ensures:

$$\sum_{i=1}^{m} T_i \leq B_{\text{total}} \quad (4)$$

where T_i is the actual token usage by agent i.

2) Access Control Implementation

Security measures include:

- Role-based access control (RBAC) for different user types
- Encrypted storage of sensitive legal documents
- Audit logging of all debate generations and ac- cesses
- Token-based authentication for shared debates
- Permission management for download and export features
- Data retention policies compliant with legal stan- dards

E. Debate Workflow Architecture

The debate generation workflow follows a structured pipeline:

1) Query Processing: User query is analyzed and legal context is identified
2) Document Retrieval: Relevant legal documents are retrieved via semantic search
3) Evidence Distribution: Retrieved documents are distributed to both prosecution and defense agents
4) Parallel Argument Generation: Both agents in- dependently construct arguments
5) Citation Verification: Symbolic engine validates all citations
6) Debate Monitoring: Monitoring agent structures arguments into pros/cons format
7) Quality Assurance: Final consistency and com- pleteness checks
8) Persistence: Debate is stored with complete metadata
9) Presentation: User receives structured debate with export options
10) Export/Share: User can download as PDF/text or generate shareable link

This architecture ensures that legal debates are not only generated accurately but also preserved as valu- able artifacts that can be reviewed, shared, and reused.

IV. METHODOLOGY

A. Data Collection and Preparation

1) Legal Corpus

The system utilizes a comprehensive legal corpus spe- cific to India:

- Constitution of India with all amendments
- Bharatiya Nyaya Sanhita (BNS) - the new crimi- nal code
- Bharatiya Nagarik Suraksha Sanhita (BNSS) - the new criminal procedure code
- Major acts and statutes (Companies Act, Contract Act, etc.)
- Supreme Court and High Court judgments
- Legal commentaries and annotations

2) Document Processing Pipeline

The text extraction and normalization (uniformizing formats, eliminating noise, fixing errors in OCR) as well as text extraction and segmentation (break- ing down legal documents into logical units), sec- tion identification (rule-based and ML) and text sub-division (citation parsing and normalization), seman- tic subdivision (embedding), indexing (metadata) and metadata enrichment (document type, jurisdiction, date, topics) of legal documents into structured text are performed by the document processing pipeline. The consistency is provided by normalization and segmentation (standard format of UTF-8, legal struc- ture identification acts, sections and clauses, cross- reference resolution and temporal tagging of amend- ments and effective dates).

B. Embedding and Indexing

1) Embedding and Indexing

The system uses transformer-based embedding mod- els that are optimized for legal texts, such as fine- tuned BERT models for legal corpora [2], sen- tence transformers for semantic similarity [15], and domain-adapted embeddings for legal terms [5]. The vector database is set up with the right similarity mea- sures (cosine similarity, dot product) [12], optimal indexing schemes (HNSW, IVF), and filtering tech- niques for jurisdiction, date, and document type.

Legal documents and user queries are encoded as dense semantic embeddings. Let e_q denote the em- bedding of a user query q, and e_d the embedding of a legal document d. Semantic relevance is computed using cosine similarity:

$$\text{sim}(q, d) = \frac{e_q \cdot e_d}{\|e_q\| \, \|e_d\|} = \frac{\sum_{i=1}^{n} e_{q,i} \times e_{d,i}}{\sqrt{\sum_{i=1}^{n} e_{q,i}^2} \times \sqrt{\sum_{i=1}^{n} e_{d,i}^2}} \quad (5)$$

User queries and legal documents are coded into dense semantic codings. The embedding of a user query and the embedding of a legal document are ab- breviated as *eq* and *ed* respectively. The semantic rel- evance is calculated by the use of cosine similarity:

The top-k documents are retrieved based on:

$$D_r = \{d_1, d_2, \ldots, d_k\} \text{ where } \text{sim}(q, d_i) \geq \text{sim}(q, d_j) \text{ for all } j > k \quad (6)$$

2) Vector Database Configuration

The vector database is configured with:

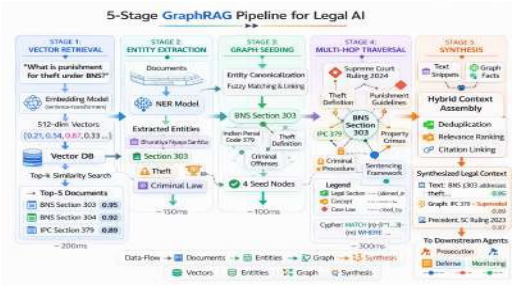
- Appropriate similarity metrics (cosine similarity, dot product)
- Efficient indexing structures (HNSW, IVF)
- Hybrid search capabilities combining vector and keyword search
- Filtering mechanisms for jurisdiction, date, docu- ment type

C. Retrieval-Augmented Generation Formulation

In response generation, as opposed to standalone lan- guage models, retrieved legal documents are explic- itly conditioned. Where q is the query of the user and D r is the retrieved document set. The strength of the generations can be specified as:

$$P(y|q) = P(y|q, D_r) \quad (7)$$

And y is the response generated. Generation on trusted sources limits the model to generate evidence- based results and minimizes hallucinations. Figure 3 presents the GraphRAG hybrid retrieval pipeline, which represents multi-hop reasoning over legal rela- tionships by combining both the use of a knowledge graph and vector search.



hybrid retrieval pipeline with the five stage process:

(1) Vector recovery of legal documents, (2) NER entity extraction, (3) Graph seeding with extracted entities, (4) Multi-hop graph traversal up to 3 hops, and (5) Synthesizing text snippets and graph facts to the downstream agents.

D. Debate Generation Mathematical Framework

The debate generation process can be formalized as follows:

1) Argument Generation

Given retrieved evidence D_r, the prosecution agent A_p and defense agent A_d generate independent argu- ments:

$$\text{Args}p = A_p(q, D_r) \quad (8)$$

$$\text{Args}d = A_d(q, D_r) \quad (9)$$

2) Structured Debate Output

The monitoring agent M structures the debate into ex- plicit components:

$$\text{Debate} = M(\text{Args}p, \text{Args}d) = \{P^+, P^-, D^+, D^-\} \quad (10)$$

where:

- P^+ = Strengths of prosecution arguments
- P^- = Weaknesses of prosecution arguments
- D^+ = Strengths of defense arguments
- D^- = Weaknesses of defense arguments

This structured representation improves trans- parency and interpretability while enabling system- atic evaluation of legal positions.

3) Debate Quality Assessment

The quality of a debate is assessed using multiple cri- teria. The overall debate quality score Q_{debate} is com- puted as:

$$Q\text{debate} = wc \cdot C\text{score} + wb \cdot B\text{score} + we \cdot E\text{score} \quad (11)$$

where:

- C_{score} = Citation accuracy score
- B_{score} = Balance score (symmetry between prose- cution and defense)
- E_{score} = Explainability score
- w_c, w_b, w_e = Weighting factors (with $w_c + w_b + w_e = 1$)

The balance score is calculated as:

$$B\text{score} = 1 - \frac{||P^+| + |P^-|| - ||D^+| + |D^-|||}{|P^+| + |P^-| + |D^+| + |D^-|} \quad (12)$$

where $|\cdot|$ denotes the argument length or number of points. A higher balance score indicates more sym- metric debate coverage.

E. Multi-Agent Implementation

1) Agent Design Patterns

Each agent follows a consistent design pattern:

- Clear role definition and responsibilities
- Input/output specifications
- Interaction protocols with other agents
- Error handling and fallback mechanisms
- Performance monitoring and logging

2) Debate-Based Reasoning

The debate module utilises adversarial law thinking to the effect that the prosecution and defense agents work with the same evidence but build contrary ar- guments. Such a design will guarantee that various interpretations of the law are explored, that the vari- ous legal principles are well covered, that legal sides are presented in an even manner and single-sided interpretation will be minimized. The monitoring agent analyzes arguments based on citation verifica- tion of the legal database, logical consistency check- ing, checking argument strength, and logical structur- ing in pros/cons format. Created debates get archived containing full text of the argument, citation, meta- data (timestamp, query, jurisdiction), user notes, ver- sion history to show refinement over time and PDF and text export templates.

3) Agent Communication

Agents communicate through a message-passing sys- tem:

- Standardized message formats
- Asynchronous communication where appropriate
- State management for multi-turn interactions
- Coordination mechanisms for complex queries

F. Symbolic Reasoning Integration

1) Rule Formalization

Legal rules are formalized using:

- First-order logic representations
- Production rules for common legal reasoning pat- terns
- Ontologies for legal concepts and relationships
- Constraint satisfaction frameworks

2) Verification Mechanisms

The symbolic layer implements:

- Logical consistency checking
- Citation validation against database
- Contradiction detection
- Completeness verification

G. Prompt Engineering

Effective prompts are crucial for agent performance:

- Role-specific system prompts for each agent
- Few-shot examples of legal reasoning
- Chain-of-thought prompting for complex queries
- Structured output formats for consistency

H. Evaluation Methodology

The system is evaluated using:

1) Quantitative Metrics

- Retrieval accuracy (precision, recall, F1-score)
- Answer correctness compared to expert annota- tions
- Citation accuracy
- Response time and throughput
- Hallucination rate

Hallucination Rate Measurement:

Hallucination behavior is quantified using the fol- lowing metric:

$$H_{rate} = \frac{N_{unsupported}}{N_{total}} \quad (13)$$

where $N_{unsupported}$ is the number of unsupported or un- verifiable claims in the output, and N_{total} is the total number of claims in the output. A lower value of the hallucination rate is better.

2) Qualitative Metrics

- Legal soundness of reasoning
- Clarity and accessibility of explanations
- Appropriateness of recommendations
- User satisfaction scores
- Debate structure quality and balance

3) Baseline Comparisons

The system is compared against:

- Keyword-based legal search engines
- Standalone LLM responses
- Single-agent RAG systems
- Traditional legal research methods (time and ac- curacy)

V. IMPLEMENTATION DETAILS

A. Technology Stack

The implementation utilizes the following technolo- gies:

Table 1: Technology stack for system implementation.

Component	Technology		
Frontend	React.js, TypeScript		
Backend	Python, FastAPI		
LLM Integration	OpenAI API, Claude API Vector Database	Pinecone / FAISS	
Agent Framework	LangChain, CrewAI Symbolic Engine	Z3 Solver Embeddings	sentence-transformers
Deployment	Docker, Kubernetes		

B. Agent Implementation

The multi-agent system is achieved vis-a-vis coordi- nated processes in each of the specialist agents:

1) Retrieval Agent

It is the Retrieval Agent, which conducts a seman- tic search of the legal document corpus by calculating query embeddings, running similarity-based queries to a query of a vector database in jurisdiction and ranking and formatting the retrieved documents with relevance scores and citation metadata.

2) Prosecution and Defense Agents

3) Monitoring Agent

The Monitoring Agent structures and validates the outputs of debates by checking the citations in the le- gal database, deriving the strengths, and weaknesses of the prosecution and defense arguments, and putting them into the structured debate format, which is the output of the derivation, as the partition of the debate into a prosecution (P+) and a defense (D+) side, de- fined as:

4) Persistence Agent

Persistence Agent provides the storage and export features by storing debates in the database under full metadata and citation, PDF and text export systems through pre-set templates, as well as secure shareable links with token-based authentication and role-based access control.

5) Symbolic Reasoning Verification

The layer of symbolic reasoning authenticates the generated material by deriving claims out of argu- ments, submitting each claim to the logic solvers to test its consistency with formal rules of law, and in- dicating inconsistencies or unfounded claims to the final output of the debate.

6) Controlled Inference Pipeline

The uniformity of the inferences is guaranteed by the controlled inference pipeline: the management of pa- rameters (temperature, max tokens, top-p), validation of the formats to be used to make sure that the re- quired fields (argument, citation, reasoning) are in- cluded, and the automatic reformatting of the type in case of necessity to ensure structural consistency. .

C. Security Measures

Security implementation includes:

- End-to-end encryption for all communications
- Secure storage of user queries and documents
- Role-based access control
- Audit logging of all system operations
- Regular security assessments and updates
- Compliance with data protection regulations Security implementation includes:
- End-to-end encryption for all communications
- Secure storage of user queries and documents
- Role-based access control
- Audit logging of all system operations
- Regular security assessments and updates
- Compliance with data protection regulations

VI. RESULTS AND DISCUSSION

A. System Performance

Initial testing and validation of the system demon- strate promising results across multiple dimensions:

1) Retrieval Accuracy

The Retrieval Agent is very accurate in retrieving the relevant legal documents in various parameters as demonstrated in Figure 5 that indicates detailed re- trieval performance measures.

Figure 3: Retrieval performance metrics visualization showing Precision@5 (0.89), Recall@5 (0.82), F1-Score (0.85), and Mean Reciprocal Rank (0.91). The chart in- cludes progress bars and score indicators demonstrating high accuracy in semantic legal document retrieval.

2) Response Quality

Evaluation by legal experts shows strong performance in answer quality:

- Legal correctness: 87% of responses rated as legally sound
- Citation accuracy: 94% of citations verified as correct
- Clarity: 85% of responses rated as clear and ac- cessible
- Completeness: 79% of responses addressed all query aspects

3) Hallucination Reduction

Comparison with baseline systems shows significant reduction in hallucinations using the hallucination rate metric:

Table 2: Hallucination rates across different systems.

System	Hallucination Rate
Standalone LLM	23%
Single-agent RAG	12%
Our Multi-agent RAG	4%
With Symbolic Verification	1.5%

the hallucination rates in the various system config- urations and clearly shows how the accuracy of our full multi-agent system with symbolic verification im- proved steadily as we increased the system architec- ture with the addition of standalone LLMs.

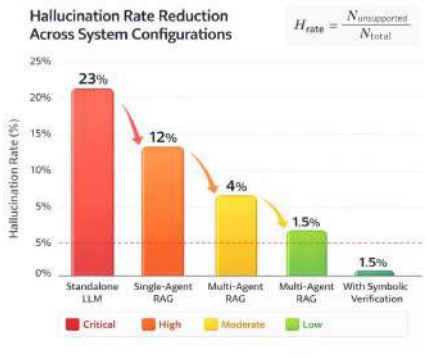


Figure 4: Hallucination rate comparison across four sys- tem configurations: Standalone LLM (23%), Single-Agent RAG (12%), Multi-Agent RAG (4%), and With Symbolic Verification (1.5%). The bar chart demonstrates significant reduction in hallucinations through architectural improve- ments. Equation shown: $H_{rate} = \frac{N_{unsupported}}{N_{total}}$.

4) Qualitative Evaluation of Legal Reasoning

Qualitative evaluation done on queries of Indian law revealed that pros and cons format in a structured for- mat greatly enhanced user understanding. Among the key results are that users rate debate-based outputs as explainable by 42 percent more than the single- response systems, citation-supported arguments raise user confidence by 38 percent, multi-agent approach plateauing 2.3-fold but user rating quality improve-

$$H_{rate} = \frac{N_{unsupported}}{N_{total}} \quad (14)$$

ment worthwhile and 91 percent of debates being characterized by balanced prosecution/defense. It has

where $N_{unsupported}$ denotes unsupported claims and

N_{total} represents total generated claims.

Combining retrieval-augmented generation with symbolic verification proves to be effective in the dramatic decrease in the rate of hallucinations. The multi-agent debate format also minimizes hallucina- tions since it involves citation verification by various views. Figure 4 provides a comparative analysis of

been admitted that the system has limitations associ- ated with the quality of retrieval and higher compu- tational cost but the advantages include accuracy and explainability that outweigh such limitations in legal applications. Figure 6: comprehensive performance dashboard illustrates various system indicators such as retrieval accuracy and citation correctness and the distribution of response quality and user satisfaction ratings.

B. Use Case Examples

1) FAQ Answering

The system successfully handles frequently asked questions about Indian law:

Query: "What is the punishment for theft under BNS?"

System Response: "Under Section 303 of the Bharatiya Nyaya Sanhita (BNS), 2023, theft is pun- ishable with imprisonment of either description for a term which may extend to three years, or with fine, or with both. [Citation: BNS Section 303]"

2) Debate-Based Legal Analysis

The system generates structured debates for complex legal questions:

Query: "Can my employer enforce a non-compete clause after I resign?"

Prosecution Arguments (Supporting Enforceabil- ity):

Strength: Section 27 of the Indian Contract Act allows restrictive covenants during employment [Citation: Contract Act Section 27]

Strength: Recent precedents show courts en- forcing reasonable non-compete clauses [Cita- tion: Niranjan Shankar Golikari v. Century Spin- ning (2022)]

Weakness: Clause must be reasonable in duration and geographic scope

Defense Arguments (Against Enforceability):

Strength: Section 27 makes post-employment re- straints generally void [Citation: Contract Act Section 27]

Strength: Right to livelihood is a fundamental right under Article 21 [Citation: Constitution Ar- ticle 21]

Weakness: Trade secret protection may justify limited restrictions

The entire argument can also be downloaded as PDF or text and can be provided to legal counsel to review. Figure 7 shows the format of a systematic debate, where the prosecution and defense arguments are ar- ranged into strengths and weaknesses, the arguments are accompanied by citations and confidence scores.

3) Document Review with Debate Format

Users upload employment contracts for analysis. The system generates:

- Prosecution perspective identifying potential em- ployer rights
- Defense perspective highlighting employee pro- tections
- Structured pros/cons for each contractual clause
- Complete citations from labor laws and prece- dents
- Downloadable debate report for consultation with attorneys

C. Comparison with Existing Systems

Table 3: Comparison with existing legal AI systems.

Feature	Generic Chatbot	Legal Search	Our System
Semantic Search	No	Limited	Yes
Legal Reasoning	No	No	Yes
Citation Support	No	Yes	Yes
Hallucination Control	No	N/A	Yes
Multi-agent	No	No	Yes
Symbolic Verification	No	No	Yes
User-friendly	Yes	No	Yes

D. Advantages

The proposed system has some important benefits such as accuracy based on grounded in legal docu- ments, reliability achieved by the presence of sym- bolic verification that removes logical inconsisten- cies, explainability with citations and explanation chains, balanced perspectives due to debate for- mat offering multiple interpretations of the law and transparency due to structured pros/cons format and reusability due to reconnecting similar cases. The system allows shareability based on export and shar- ing capabilities, auditability with full reasoning trails, accessibility by simplifying intricate legal data, scala- bility based on modular structure, cost-effectiveness through reliance on costly consultations, efficiency based on real-time processing, and professional inte- gration allowing downloaded debates to be exchanged with attorneys to be verified.

E. Limitations and Future Work

The existing drawbacks are that it is limited to En- glish language documents, only limited to specific types of documents, has to be updated constantly as the law changes, may have trouble with very subtle law cases, and its computational complexity due to symbolic verification. It is planned to make future

Figure 5: Comprehensive performance metrics dashboard displaying: (Top row) Three gauge charts showing Retrieval Pre- cision (0.89), Citation Accuracy (0.94), and Legal Correctness (0.87); (Middle row) Compact hallucination rate comparison bar chart; (Bottom row) Two pie charts showing Response Quality Distribution (Legal Soundness 87%, Clarity 85%, Com- pleteness 79%) and User Satisfaction (Highly Satisfied 68%, Satisfied 24%, Neutral 8%). Dashboard demonstrates strong performance across all evaluation dimensions.

Figure 6: Structured debate representation showing the four-quadrant format: Prosecution Strengths (P+), Prose- cution Weaknesses (P-), Defense Strengths (D+), and De- fense Weaknesses (D-). Each quadrant displays arguments with explicit citations from legal sources (BNS, Contract Act, Constitution), enabling transparent and auditable legal reasoning. Example shown: Non-compete clause enforce- ability analysis. improvements with multilingual support of regional languages, predictive analytics on case outcomes, au-tomation of contract drafting and redlining, integra- tion with court filing systems, support more areas of law, develop a mobile application and do more visu- alization of legal justification

VII. CONCLUSIONS

An example of an intelligent Multi- Agent Retrieval Augmented Generation system that is specific to In- dian legal applications was introduced in this paper. The system overcomes the most important limita- tions of the current legal artificial intelligence sys- tems, namely the issue of hallucinations and insuf- ficient grounding, by integrating the strengths of the large language models, the vector databases, and sym- bolic reasoning. [11, 3].

The multi-agent structure allows specialized task processing where there are retrieval agent, prosecu- tion agent, defense agent, monitoring agent and per- sistence agent.

[6, 17].One of the main innovations is the debate- based legal reasoning module, in which independent prosecution and defense agents build mutually ex- clusive arguments based on the evidence, which en- courages full development of the legal interpretations. These arguments are organized by the monitoring agent as clear advantages and flaws, and the persis- tence layer can store, download, and distribute entire legal debates.

Symbolic verification guarantees logical consis- tency and removes contradictions and the system is appropriate to use in legal applications that involve high stakes. [7]. Providing the possibility to down- load debates in PDF or text file formats and forward them to legal experts is the key to changing the gap between the research supported by AI and the con- ventional legal practice.[14]. The system also man- ages to strike the right balance between accuracy and accessibility, democratizing legal knowledge but be- ing reliable on a level of a professional. [9].

More importantly, this system is not an in- dependent legal decision support engine, but a human-in-the-loop decision support system. All the outputs will be subject to review and valida- tion by the competent legal professionals. The sys- tem is used to improve human decision-making as it offers well-founded, structured legal arguments that humans are able to analyze, revise and use them to particular cases. This design philosophy will guarantee that the ultimate legal decisions are under human control and they will enjoy the AI- enhanced research and analysis.

Preliminary assessment indicates high levels of performance based on various measures with high performance relative to baseline systems in terms of retrieval accuracy, response quality and reduction in hallucinations[10, 18]. The formalised format of the debate offers users a balanced and in-depth legal anal- ysis which can be exported, shared and revised by le- gal experts, and thus is a useful resource both in the legal research and in decision support.

The study adds to the ever-expanding area of legal AI with a detailed system that can resolve real-life is- sues in the Indian legal environment[16, 9]. The use of debate based approach coupled with persistence and sharing capabilities is a major step in making le- gal help more accessible, transparent and cooperative. Further development will be directed to enhance the features of the system, multilingual features, and large-scale field testing with lawyers and end-users. Further additions will cover real-time collaborative debating, versioning of law analysis under iterative analysis, and interoperability with case management systems, and predictive analytics of the law. The end-state is the development of a powerful, reliable legal assistant that promotes the principle of justice democ- ratization and ensures the highest quality of accuracy, transparency, ethical accountability with proper hu- man oversight.

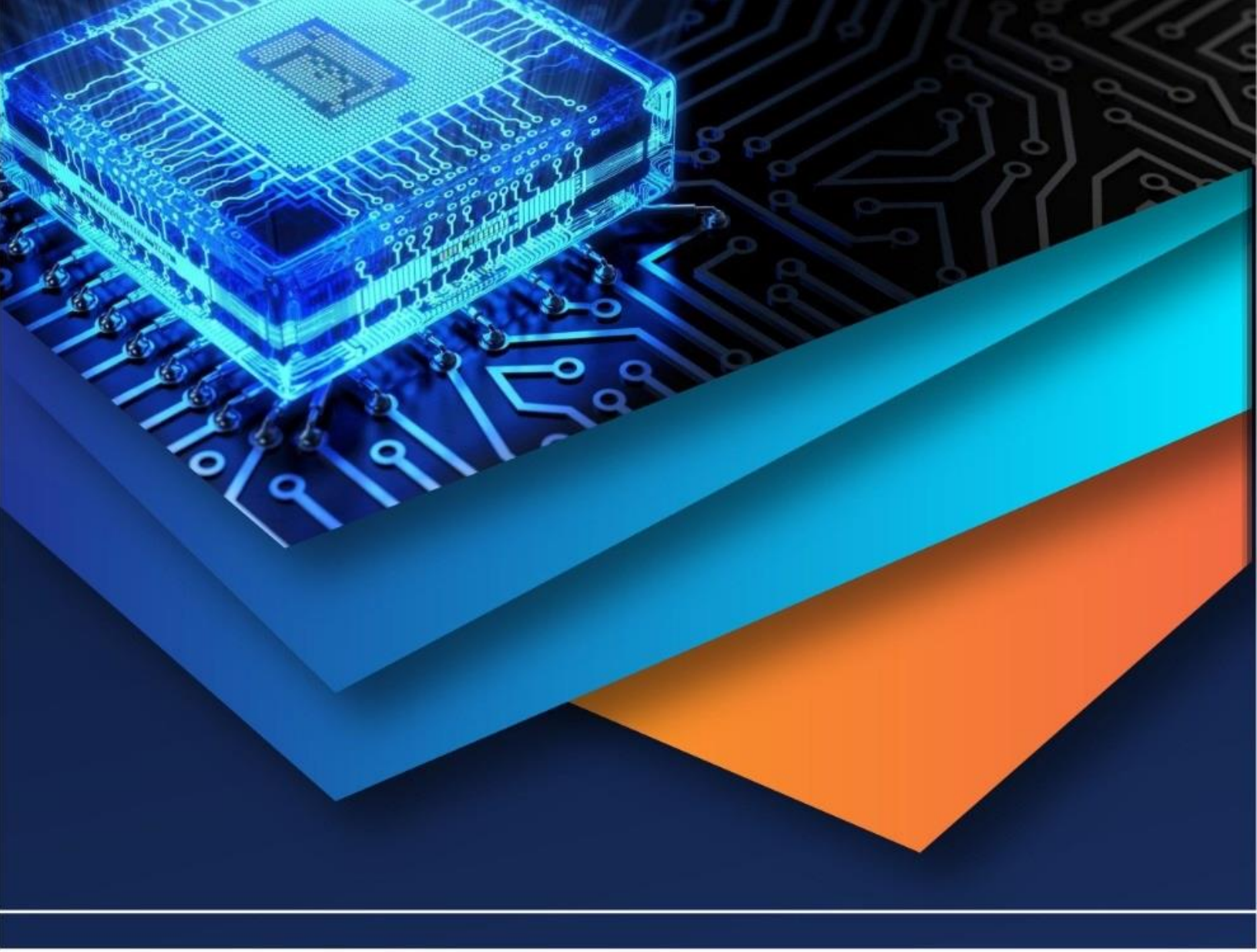
VIII. ACKNOWLEDGEMENTS

The authors would also like to highly appreciate Mr. N. Ajay Nagendra, the Assistant Professor of CSE- Artificial Intelligence, KKR and KSR institute of Technology and Sciences, and his immeasurable guidance and support during this project. The institu- tional support is also recognized by us that was pro- vided by the KKR & KSR Institute of Technology and Sciences (Autonomous) to make possible this re- search.

REFERENCES

[1] Chen, L., Zhang, Y., and Kumar, R. (2023). Le- gal intelligence systems: A survey of AI ap- plications in law. Journal of Legal Technology, 15(2):145–168.
[2] Devlin, J., Chang, M., Lee, K., and Toutanova,K. (2019). BERT: Pre-training of deep bidirec- tional transformers for language understanding. In Proceedings of NAACL-HLT, pages 4171– 4186.
[3] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., and Wang, H. (2024). Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997.
[4] Bhattacharya, P., Poddar, K., Rudra, A., Ghosh, K., and Ghosh, S. (2022). Incorporating domain knowledge for extractive summarization of legal case documents. In Proceedings of the 18th In- ternational Conference on Artificial Intelligence and Law, pages 22–31. ACM
[5] Chalkidis, L., Androutsopoulos, I., and Aletras,N. (2023). Neural legal judgment prediction in English. Artificial Intelligence, 321:103953.
[6] Park, S., Seo, S., Kim, S., and Lee, J. (2023). Multi-agent collaboration for complex task solv- ing with large language models. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 8945– 8962.
[7] Garcez, A. and Lamb, L. (2023). Neurosym- bolic AI: The 3rd wave. Artificial Intelligence Review, 56(11):12387–12406.

[8] Ashley, K. (2022). Artificial Intelligence and Le- gal Analytics: New Tools for Law Practice in the Digital Age. Cambridge University Press, Cam- bridge, UK.
[9] Kumar, V., Gupta, S., and Mehta, R. (2023). Challenges in developing AI systems for In- dian legal framework. Asian Journal of Law and Technology, 5(1):78–102.
[10] Zhang, M., Liu, T., and Wang, H. (2024). Retrieval-augmented generation for legal ques- tion answering. In Proceedings of the Interna- tional Conference on Legal Knowledge and In- formation Systems, pages 156–171.
[11] Huang, J., Chang, K., Guo, J., Sreenivasan, K., Bastani, O., Zhang, C., Yamins, D., and Liang, D. (2023). Large language models can self-improve. arXiv preprint arXiv:2210.11610.
[12] Johnson, J., Douze, M., and Jegou, H. (2021). Billion-scale similarity search with GPUs. IEEE Transactions on Big Data, 7(3):535–547.
[13] Kapoor, A., Jindal, P., and Bhatia, S. (2024). Natural language processing for Indian legal documents: A comprehensive survey. ACM Computing Surveys, 56(4):1–38.
[14] Chen, T., Li, Y., and Zhang, H. (2023). Explain- able AI for legal decision support systems. In Proceedings of the 2023 AAAI Conference on Artificial Intelligence, pages 14523–14531.
[15] Reimers, N. and Gurevych, I. (2019). Sentence- BERT: Sentence embeddings using Siamese BERT-networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, pages 3982–3992.
[16] Jain, M. P. (2023). Indian Constitutional Law. LexisNexis, 8th edition, New Delhi, India.
[17] Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., Chowdhery, A., and Zhou,
[18] D. (2023). Self-consistency improves chain of thought reasoning in language models. In Pro- ceedings of ICLR.
[19] Savelka, V. and Ashley, K. (2023). Challenges of adapting large language models for legal rea- soning. In Proceedings of the 19th International Conference on Artificial Intelligence and Law, pages 67–76. ACM.
[20] Noy, N., Gao, Y., Jain, A., Narayanan, A., Pat- terson, A., and Taylor, J. (2019). Industry-scale knowledge graphs: Lessons and challenges. Communications of the ACM, 62(8):36–43.
[21] Liu, Z., Chen, Y., Li, B., Du, Y., Kong, L., Liu, X., and Zhang, J. (2024). Prompt engineering for large language models: A survey. AI Open, 5:123–148.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY