International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 04(April),2026
Received: 18-02-2026 Accepted: 23-03-2026 Published: 01-04-2026

LegalGuide - A MultiAgent Legal Reasoning System

M. Srividya[1], Amulya Dugyala[2], M.Vishnuvardhan[3], A.Sai Rutvij Reddy[4]

[1]*Assistant professor, Department of Information Technology , Matrusri Engineering College.*
m.srividyareddy09@gmail.com
[2345]*Students of Department of Information Technology , Matrusri Engineering College*
amulyadugyala1105@gmail.com
mvishnu01234@gmail.com
rutvijreddyadem@gmail.com

Abstract:
The rapid growth of legal data and the increasing complexity of legal systems make legal research and case analysis time-consuming and difficult for non-experts. This paper proposes a Multi-Agent Legal Reasoning System. It is designed to help users analyze legal problems by automatically identifying relevant laws, retrieving precedents, and generating structured legal reasoning.The proposed system has a multi-agent structure. Specialized AI agents carry out different tasks, including case intake, IPC section retrieval, precedent analysis, argument generation, and explainable legal output generation. These agents work together through a coordinated workflow to mimic the reasoning process of legal professionals.The system combines Large Language Models (LLMs), Retrieval-Augmented Generation (RAG), and structured legal datasets to improve clarity and transparency. Experimental results indicate that this multi-agent structure enhances reasoning clarity and modularity compared to traditional single-model legal AI systems. The system shows promise as an educational tool, a legal support assistant, and a research platform for legal AI applications.

Keywords:
Multi-Agent Systems, Legal AI, Legal Reasoning, Indian Penal Code, Large Language Models, Explainable AI, Legal Decision Support.

I.Introduction

Legal research and reasoning involve examining large amounts of statutes, case law, and precedents. Traditional legal research is manual and takes a lot of time. It also requires considerable expertise. Recently, Artificial Intelligence has been used to automate several legal tasks like document analysis, case classification, and judgment prediction.

However, single-model AI systems often face challenges with complex legal reasoning. They also deal with issues like hallucination and lack transparency. Multi-agent systems offer a promising option by breaking down complex reasoning tasks into smaller, specialized parts[1][3][13].

In a multi-agent legal reasoning setup, different agents take on roles found in legal practice, such as legal researchers, lawyers, and judges. These agents work together to analyze case facts, find legal provisions, create arguments, and give a final explanation[2][5][14].

This research suggests a Multi-Agent Legal Reasoning System for the Indian legal field. It brings together several specialized agents to automate legal analysis and provide structured reasoning outputs.

The main goals of this work are:

- To design a modular multi-agent structure for legal reasoning
- To integrate IPC law retrieval and legal precedent analysis
- To simulate legal debates between opposing views
- To generate clear legal reasoning outputs

II. Related Work

The development of Artificial Intelligence in the legal domain has progressed through multiple stages, beginning with rule-based systems and advancing toward modern large language model and multi-agent architectures. Early legal AI systems such as TAXMAN were designed as rule-based expert systems that encoded legal knowledge into predefined logical rules. These systems relied on symbolic reasoning and used "if–then" conditions to derive conclusions from structured inputs. While they demonstrated the feasibility of automating legal reasoning, they were limited by their rigidity and inability to handle ambiguity in legal language. The underlying technology in these systems was primarily knowledge representation and rule-based inference engines, which required extensive manual encoding of legal rules[6].

To overcome the limitations of rule-based systems, researchers introduced case-based reasoning (CBR) approaches. One of the most influential systems in this category is HYPO, which focused on analyzing trade secret law by comparing current cases with previously decided cases. HYPO utilized structured representations of legal cases and applied similarity-based reasoning to generate legal arguments. Building upon this, CATO introduced a more advanced argumentation framework that organized legal reasoning into factors and dimensions, allowing for better structuring of legal debates. These systems relied on case representation models, similarity metrics, and argumentation frameworks as their core technologies. Although they improved flexibility and reasoning capability, they required highly structured datasets and struggled with large-scale unstructured legal corpora[7][8].

With the rise of machine learning, legal AI systems began shifting toward data-driven approaches. Platforms such as LawGeex and Kira Systems apply machine learning techniques, particularly natural language processing (NLP) and supervised learning models, to automate contract review and clause extraction. These systems analyze large volumes of legal documents and identify risks, obligations, and inconsistencies. Similarly, ROSS Intelligence leveraged IBM Watson's NLP capabilities to provide legal question answering by interpreting user queries and retrieving relevant legal information. Traditional legal research platforms such as Westlaw and LexisNexis have also incorporated AI-driven search and recommendation systems using information retrieval techniques, machine learning ranking algorithms, and NLP-based semantic search. Despite their effectiveness, these systems often function as black-box models and lack transparency in reasoning[9][10].

The advancement of Natural Language Processing (NLP) further transformed legal AI by enabling systems to process unstructured legal texts such as court judgments, statutes, and legal queries. Platforms like Indian Kanoon and Manupatra provide access to extensive legal databases and utilize keyword-based search, semantic indexing, and document retrieval techniques to assist legal research. More advanced NLP applications include legal text summarization, named entity recognition for extracting legal entities, and question answering systems. These systems are typically powered by transformer-based models such as BERT and GPT variants, which enable contextual understanding of legal language. However, they still face challenges such as hallucination, lack of structured reasoning, and difficulty in providing explainable outputs.

To address issues related to factual accuracy and hallucination, recent research has introduced Retrieval-Augmented Generation (RAG) systems[11]. These systems combine information retrieval mechanisms with large language models (LLMs) to generate responses grounded in real legal documents. In such

architectures, relevant legal texts are first retrieved from a database and then used as context for response generation. This approach significantly improves the reliability of AI-generated legal outputs. RAG-based systems are widely used in modern legal chatbots and research assistants, leveraging technologies such as vector databases, embedding models, and transformer-based LLMs. While they improve factual grounding, they still do not fully replicate the structured reasoning process followed by legal professionals[12][15].

Recent advancements show that multi-agent systems improve reasoning by decomposing tasks into specialized agents, enabling better modularity and structured workflows [1][3][13]. Additionally, Retrieval-Augmented Generation (RAG) improves factual accuracy by grounding outputs in legal documents [2][4]. Multi-agent adversarial systems further enhance decision reliability through validation and critique among agents [5].

Despite these advancements, existing systems still lack a fully integrated approach that combines legal knowledge retrieval, argument generation, debate simulation, and explainable reasoning within a single framework, particularly in the context of Indian law. Most current systems either focus on retrieval, prediction, or isolated NLP tasks, without capturing the complete legal reasoning pipeline. This research addresses these gaps by proposing a multi-agent legal reasoning system that integrates IPC section identification, precedent analysis, argument generation from opposing perspectives, judicial evaluation, and explainable output generation into a unified architecture.

III. System Overview and Architecture

The proposed system is a Multi-Agent Legal Reasoning System. It aims to automate and simulate the legal reasoning process using several specialized AI agents. The system takes a natural language case description as input. It then processes this through a sequence of agents that work together to perform tasks like fact extraction, identifying legal provisions, retrieving precedents, generating arguments, and analyzing decisions.

Unlike traditional single-model systems, this approach uses a multi-agent setup[1][3][13]. Each agent has a specific responsibility. The system employs Large Language Models (LLMs) for reasoning and Retrieval-Augmented Generation (RAG) to ground outputs in legal data[2][4][11]. An orchestration framework, such as CrewAI, coordinates the workflow between agents. This ensures structured and sequential execution, similar to modern multi-agent framework[1][3].

The system follows a layered multi-agent architecture that separates user interaction, coordination, reasoning, and data handling. This design ensures modularity, scalability, and efficient execution of legal reasoning tasks.

User Interface Layer:
This layer lets users input legal queries in natural language and view the structured output. It offers a simple and accessible way to interact with the system.

Orchestration Layer:
This layer manages the workflow by coordinating communication between agents and ensuring tasks are done in sequence[14]. It uses frameworks like CrewAI to control how agents interact.

Agent Layer:
This core layer has multiple specialized agents that perform tasks such as fact extraction, law identification, argument generation, and decision-making. Each agent plays a role in a specific stage of the legal reasoning process.

Data & Knowledge Layer:
This layer stores legal resources like IPC sections and case laws for retrieval. It supports RAG by providing relevant context through vector databases and embedding models.

6.6 Workflow Summary
The overall workflow of the system begins with user input, followed by fact extraction, legal retrieval, argument generation, judicial

International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 04(April),2026
Received: 18-02-2026 Accepted: 23-03-2026 Published: 01-04-2026

evaluation, and explanation generation. Each agent contributes incrementally to the reasoning process, resulting in a comprehensive and explainable legal output.

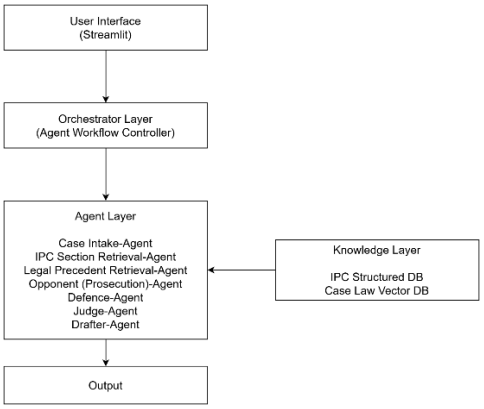

Figure 1: System Architecture

IV. Methodology

The proposed system uses a multi-agent, sequential approach to simulate real-world legal reasoning by combining Natural Language Processing (NLP), Retrieval-Augmented Generation (RAG), and Large Language Models (LLMs). The methodology consists of several stages, each handled by a specialized agent.

4.1 Case Input and Fact Extraction

The process starts with the user providing a description of a legal case in natural language. The Case Intake Agent processes this input using NLP techniques like semantic parsing and entity recognition to extract key facts, entities, and legal issues. This step turns unstructured input into a structured format for further analysis.

4.2 Legal Provision Identification

The structured case information is sent to the IPC Section Agent, which identifies relevant sections of the Indian Penal Code. It uses a mix of keyword matching, semantic similarity, and reasoning based on LLMs. This step ensures the system connects its analysis to appropriate legal provisions.

4.3 Legal Precedent Retrieval

The Legal Precedent Agent retrieves similar past cases to support the reasoning process. It uses a Retrieval-Augmented Generation (RAG) approach, where legal documents are stored in a vector database and fetched using embedding-based semantic search. This allows the system to find contextually relevant precedents even when exact terms don't match .similar to modern retrieval-based systems[2][4][11].

4.4 Argument Generation (Adversarial Reasoning)

To mimic real-world legal proceedings, the system creates arguments from both sides. The Opponent Agent builds prosecution arguments that focus on establishing liability, while the Defense Agent produces counterarguments by questioning evidence and offering alternative interpretations. This setup promotes balanced reasoning and aligns with multi-agent adversarial reasoning framework[5][13].

4.5 Judicial Evaluation

The Judge Agent evaluates the arguments generated by both sides along with the identified laws and precedents. It uses logical reasoning to assess the evidence and make a fair and consistent decision, simulating a judge's role in legal proceedings similar to structured multiagent reasoning approaches[1][3].

4.6 Explainability

To promote transparency, the Explainability Agent offers a detailed breakdown of the reasoning process. It clarifies how facts, laws, and arguments contributed to the final decision, addressing the shortcomings of black-box AI systems.

4.7 Structured Output Generation

Finally, the Legal Drafter Agent compiles all the information into a structured legal format. The output usually includes a case summary, relevant IPC sections, arguments from both sides, judicial reasoning, and a final conclusion, making it suitable for academic or practical use.

International Journal of Engineering Science and Advanced Technology (IJESAT)
Vol 26 Issue 04(April),2026
Received: 18-02-2026 Accepted: 23-03-2026 Published: 01-04-2026

This structured methodology ensures modularity, balanced reasoning, improved accuracy, and explainability, effectively replicating the workflow of real-world legal analysis.

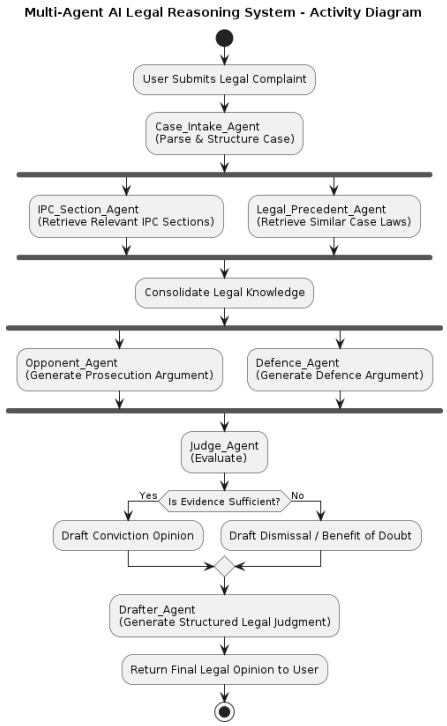

Figure 2 : Activity diagram

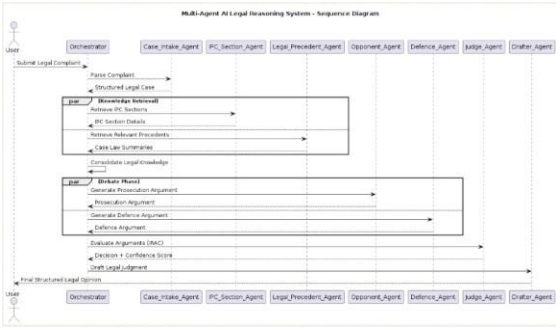

Figure 3: Sequence diagram

V. Result and discussion

The proposed Multi-Agent Legal Reasoning System was tested on sample legal scenarios to see how well it could extract facts, identify relevant IPC sections, and generate structured legal analysis. The system mapped case details to the right laws and retrieved relevant precedents using RAG, consistand with retrieval based approaches[2][4][11].

The involvement of Opponent and Defense agents helped create balanced arguments, while the Judge Agent offered logical conclusions. The Explainability Agent improved transparency by providing clear reasoning steps. Compared to single-model approaches, this system produced more structured and easier-to-understand results, aligning with multi-agent systems research[1][3][13].

The adversarial setup improves decision robustness, similar to validation frameworks [5].

However, performance relies on the quality of the retrieved data and the consistency of the LLM. This can sometimes lead to minor inaccuracies. Overall, the system shows strong potential as a reliable tool for legal decision support and education.

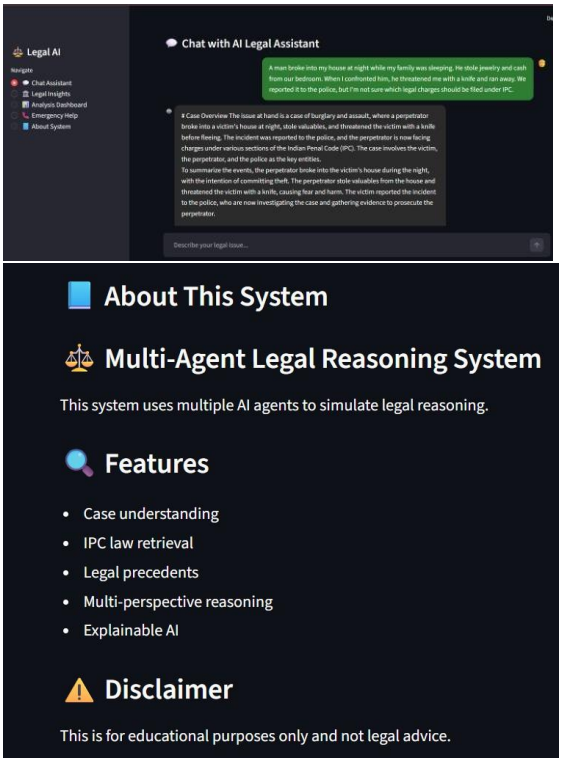

Figure 4: Output

VI. Conclusion and Future Scope

This paper presented a Multi-Agent Legal Reasoning System that simulates real-world legal analysis using specialized agents for fact extraction, law identification, argument generation, and decision-making. By integrating LLMs with Retrieval-Augmented Generation, the system produces structured, explainable, and balanced legal outputs. The

modular multi-agent architecture improves reasoning clarity and aligns with recent multi-agent research [1][3][11].
The integration of RAG enhances factual grounding [2][4], while adversarial reasoning improves robustness [5].

For future work, the system can be enhanced by integrating larger and more diverse legal datasets, including real-time court judgments. Additional improvements may include multilingual support, advanced knowledge graphs, and more robust evaluation mechanisms to improve accuracy and reliability. Expanding the system to handle multiple legal domains and real-world deployment scenarios can further increase its practical applicability.

VII. References

[1] Z. Wang et al., "L-MARS: Legal Multi-Agent Workflow with Orchestrated Reasoning and Agentic Search," arXiv, 2025.

[2] T. V. Nguyen et al., "MA-RAG: Multi-Agent Retrieval-Augmented Generation via Collaborative Reasoning," 2024.

[3] R. Petros et al., "PAKTON: A Multi-Agent Framework for Question Answering," EMNLP, 2025.

[4] M. Buffa et al., "Enhancing Legal Document Building with Retrieval-Augmented Generation," Computer Law & Security Review, 2025.

[5] "Multi-Agent Adversarial Verification for Regulated AI Decision Systems," SSRN, 2025.

[6] A. Branting, "Automated Legal Reasoning: The State of the Art," Artificial Intelligence and Law, 2017.

[7] K. D. Ashley, "Modeling Legal Argument: Reasoning with Cases and Hypotheticals," MIT Press, 1990.

[8] D. B. Skalak and E. Rissland, "Arguments and Cases: An Inevitable Intertwining," Artificial Intelligence and Law, 1992.

[9] J. Zhong et al., "Legal Judgment Prediction via Topological Learning," EMNLP, 2018.

[10] D. Chalkidis et al., "Legal-BERT: The Muppets Straight Out of Law School," Findings of EMNLP, 2020.

[11] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," NeurIPS, 2020.

[12] T. Brown et al., "Language Models are Few-Shot Learners," NeurIPS, 2020.

[13] S. Yao et al., "ReAct: Synergizing Reasoning and Acting in Language Models," 2023.

[14] Y. Schick et al., "Toolformer: Language Models Can Teach Themselves to Use Tools," NeurIPS, 2023.

[15] OpenAI, "GPT-4 Technical Report," 2023.