ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1714

LegalMind: A Multi-Agent Legal Reasoning Framework Leveraging Fine-Tuning of Large Language Models and Retrieval-Augmented Generation to Reduce Hallucinations

Soham Sachin Shelar[1]; Dr. Manisha Bharati[2]

[1]M. Tech (Data Science), Department of Technology Savitribai Phule Pune University, Pune, India
[2]Department of Technology Savitribai Phule Pune University, Pune, India

Publication Date: 2026/06/06

Abstract: The Indian legal system generates an enormous volume of judgments every year across thousands of courts, yet access to structured legal research remains limited for a large portion of the population. Existing large language models (LLMs), while capable of impressive natural language generation, suffer from hallucination—fabricating statutory provisions, inventing case citations, and producing reasoning that lacks grounding in actual evidence. These failures are especially dangerous in the legal domain, where an incorrect citation can invalidate an entire argument. This paper presents LegalMind AI, a multi-agent legal reasoning system designed specifically for Indian law. The system combines Retrieval-Augmented Generation (RAG) with a QLoRA fine-tuned Mistral-7B model trained on 42,465 cases from the IL-TUR Indian legal corpus, an NLI-based Verifier Agent powered by DeBERTa-v3, and a deterministic Judge Agent that synthesizes verified outputs into a structured final legal opinion. Evaluation across three real-world use cases demonstrates that the RAG-augmented pipeline eliminates fabricated statutory references, achieves 93.3% average retrieval relevance, and maintains 100% structural compliance in the Lawyer Agent's output. All three demonstrated queries passed verification with High confidence. The system is deployed as a Streamlit application backed by a FastAPI REST endpoint, making it accessible to legal practitioners without specialized machine learning expertise.

Keywords: *Multi-Agent AI; Legal Reasoning; Retrieval-Augmented Generation; QLoRA Fine-Tuning; Indian Law; Natural Language Inference; Hallucination Detection; Mistral-7B; FAISS; IL-TUR Dataset; DeBERTa; Streamlit.*

How to Cite: Soham Sachin Shelar; Dr. Manisha Bharati (2026) LegalMind: A Multi-Agent Legal Reasoning Framework Leveraging Fine-Tuning of Large Language Models and Retrieval-Augmented Generation to Reduce Hallucinations. *International Journal of Innovative Science and Research Technology*, 11(5), 3431-3439. https://doi.org/10.38124/ijisrt/26may1714

I. INTRODUCTION

India has one of the most expansive judicial systems in the world. The Supreme Court, twenty-five High Courts, and thousands of district courts collectively produce millions of judgments annually. Despite this scale, access to structured legal research remains deeply unequal: those without resources to retain experienced counsel often navigate complex statutory and case-law landscapes with little guidance.

Artificial intelligence offers a meaningful path toward bridging this gap. Large language models (LLMs) trained on broad corpora exhibit strong natural language understanding and can engage fluently with legal questions. However, they also demonstrate a well-documented failure mode known as hallucination—generating responses that are fluent and confident but factually wrong. In legal contexts, hallucination takes forms that carry serious realworld consequences: a non-existent section number cited as authority, an overturned precedent treated as binding, or a contractual interpretation derived from a foreign jurisdiction's doctrine rather than Indian law.

These failures motivate the need for a system that does not simply generate legal text, but grounds that text in retrieved evidence, verifies it against that evidence independently, and synthesizes a final opinion only after passing structured quality gates.

LegalMind AI is the system described in this paper. It addresses three interconnected challenges in Indian legal AI:

- *Hallucination*: Mitigated through RAG grounding, prompt constraints, and NLI-based post-hoc verification.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1714

- *Domain specificity*: Addressed through QLoRA finetuning of Mistral-7B on 25,000 instruction samples derived from 42,465 IL-TUR Indian legal cases.
- *Transparency*: Achieved through a three-tab UI that exposes retrieved context, the Lawyer Agent's raw analysis, and the Verifier's report to the end user.

The remainder of this paper is organized as follows. Section II surveys related work. Section III describes the dataset and preprocessing. Section IV presents the system architecture. Section V details the multi-agent pipeline. Section VI covers fine-tuning. Section VII presents results and evaluation. Section VIII concludes.

II. LITERATURE REVIEW

➢ *Legal NLP and Indian Law Corpora*

Early work on legal NLP focused on document classification and judgment outcome prediction. Aletras et al. [1] predicted European Court of Human Rights outcomes at 79% accuracy using textual features. The ILDC dataset [2] introduced the first large-scale Indian legal corpus with 35,000 annotated Supreme Court cases. Its extension,

CJPE [3], added explanation generation tasks. The ILTUR benchmark [4] unified these with multi-task annotation covering six legal reasoning tasks across 42,465 cases, forming the foundation of this work.

➢ *LLM Hallucination and Mitigation*

Ji et al. [6] conducted a comprehensive survey establishing that hallucination is pervasive across NLG systems. Bang et al. [7] evaluated ChatGPT on citation-heavy legal queries and reported significant failure rates. The Stanford HAI audit (2023) found hallucination rates of 17–88% in legal citation tasks using general-purpose LLMs. Lewis et al. [8] proposed Retrieval-Augmented Generation (RAG) as a grounding mechanism, substantially reducing extrinsic hallucination for knowledge-intensive tasks.

➢ *Efficient Fine-Tuning for Legal Domains*

Chalkidis et al. [13] introduced LEGAL-BERT, demonstrating that domain-specific pre-training yields significant gains on legal classification tasks. Hu et al. [15] introduced LoRA, enabling parameter-efficient fine-tuning by injecting trainable rank-decomposition matrices into frozen transformer layers. Dettmers et al. [16] extended this with QLoRA, combining 4-bit NF4 quantization with LoRA to enable fine-tuning of 7B+ models on single consumer GPUs. Jiang et al. [17] released Mistral-7B, which outperforms LLaMA-2-13B on multiple benchmarks with half the parameters, making it the preferred base model.

➢ *Multi-Agent Architectures*

Wu et al. [22] introduced AutoGen for multi-agent LLM collaboration. Park et al. [21] demonstrated generative agents with persistent memory and planning. In the legal domain, Chan et al. [23] proposed multi-agent debate between LLMs to improve reasoning quality. The LegalMind architecture follows a sequential verify-thensynthesize pattern that operationalizes the adversarial structure of legal proceedings within an automated pipeline.

➢ *Research Gap*

Table 1 summarizes the research gap. No existing system combines RAG grounding, domain-specific fine-tuning, NLI-based verification, and a deterministic synthesis layer for Indian law within a single end-to-end deployed application.

Table 1 Comparison of LegalMind AI with Related Systems

System	RAG	Fine-tuned	NLI	Indian	Deployed
LEGAL-BERT	✗	✓	✗	✗	✗
GPT-4	✗	✗	✗	Partial	✓
LawGPT	✗	✓	✗	✗	✗
InstructLaw	✓	✓	✗	✗	✗
LegalMind AI	✓	✓	✓	✓	✓

III. DATASET CONSTRUCTION

➢ *Source Dataset: IL-TUR*

The IL-TUR (Indian Legal Text Understanding and Reasoning) benchmark combines the ILDC and CJPE corpora with additional annotations, providing structured fields for each case: *facts*, *issues*, *reasoning*, *decision*, *outcome label*, and *arguments*. The dataset was loaded as a JSONL file containing 42,465 cases.

➢ *Field Completeness Analysis*

Before constructing the instruction dataset, an analysis of field completeness was conducted (Table 2).

Table 2 IL-TUR Dataset Field Completeness

Field	Non-empty	Coverage (%)
facts	41,890	98.6
outcome_label	42,120	99.2
decision	40,500	95.4
issues	38,240	90.1
reasoning	36,100	85.0
arguments	22,800	53.7

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1714

➢ *Instruction Dataset Design*

Four instruction task types were designed to span the spectrum of legal reasoning (Table 3). Quality filters excluded cases with fewer than 80 words in the facts field and fewer than 80 words in the reasoning field for reasoningdependent tasks. The resulting 25,000-sample dataset is split 90/10 into training (22,500) and validation (2,500) sets, stratified by task type.

Table 3 Instruction Dataset Task Types and Counts

Task Type	Count	Output Source
Issue Identification	8,000	issues field
Judgment Reasoning	10,000	structured template
Outcome Explanation	6,000	reasoning field
Doctrine Explanation	3,000	reasoning / issues
Total	25,000	

➢ *FAISS Index Construction*

IL-TUR cases were chunked into 700-character segments with 10% overlap between consecutive chunks to prevent relevant information from spanning chunk boundaries. Approximately 150,000 chunks were embedded using all-mpnet-base-v2 (768-dimensional dense vectors) and indexed in a FAISS flat inner-product index. A metadata JSONL file maps each embedding to its source case identifier and field label.

IV. SYSTEM ARCHITECTURE

LegalMind AI is organized into four functional layers as illustrated in Figure 1.

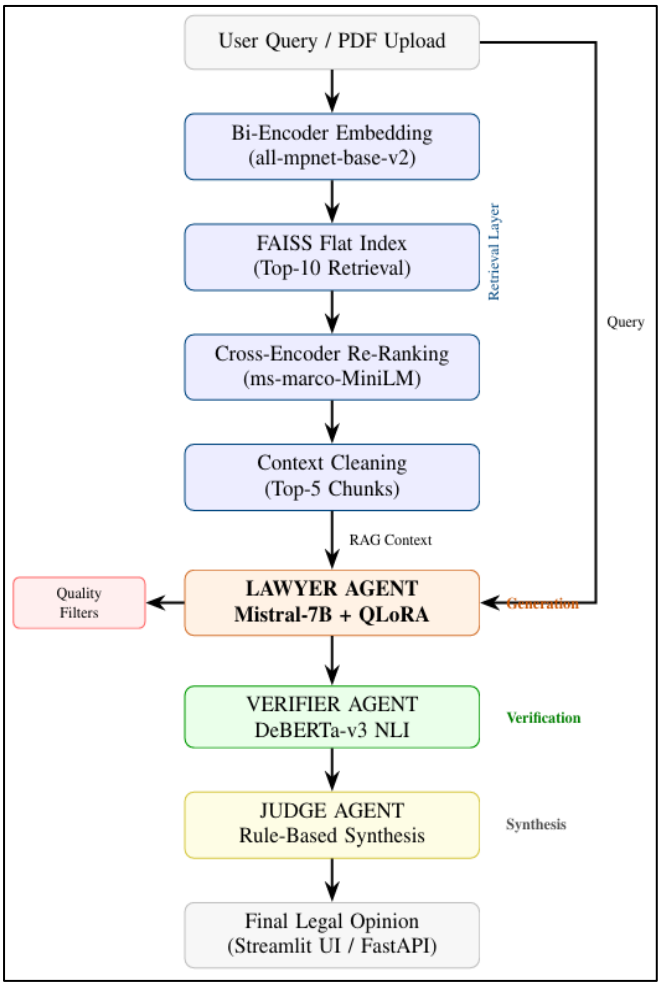

Fig 1 High-Level Architecture of LegalMind AI

➢ *Design Principles*

The architecture adheres to four core principles derived from the constraints of the legal domain:

- Evidence Grounding: Every claim in the output must be traceable to a specific retrieved passage from the ILTUR corpus.
- Epistemic Humility: When retrieved materials do not support a claim, the system explicitly states this rather than generating a plausible-sounding but ungrounded response.
- Transparency: The three-tab UI makes the retrieved context, raw lawyer analysis, and verification report all visible, enabling the user to audit the full reasoning chain.
- Jurisdictional Fidelity: A foreign-law rejection filter ensures that US, UK, or EU legal principles are never applied to Indian law queries.

➢ *Retrieval Layer*

Documents are encoded using sentence-transformers/allmpnet-base-v2, a 768-dimensional bi-encoder trained on over one billion sentence pairs. Query and document embeddings are normalized before retrieval:

$$\mathbf{e}_d = \frac{f_\theta(d)}{\|f_\theta(d)\|_2} \tag{1}$$

Top-10 FAISS candidates are re-ranked using a crossencoder ms-marco-MiniLM-L-6-v2 that scores querypassage pairs jointly:

$$s_{\text{rerank}}(q, d) = g_\phi([q;\ d]) \tag{2}$$

The top-5 re-ranked passages form the RAG context C passed to the Lawyer Agent. A deterministic cleaning step corrects recurring OCR artefacts from the IL-TUR corpus before truncating each chunk to 700 characters.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1714

V. MULTI-AGENT PIPELINE

Figure 2 shows the detailed flow through the three-agent pipeline.

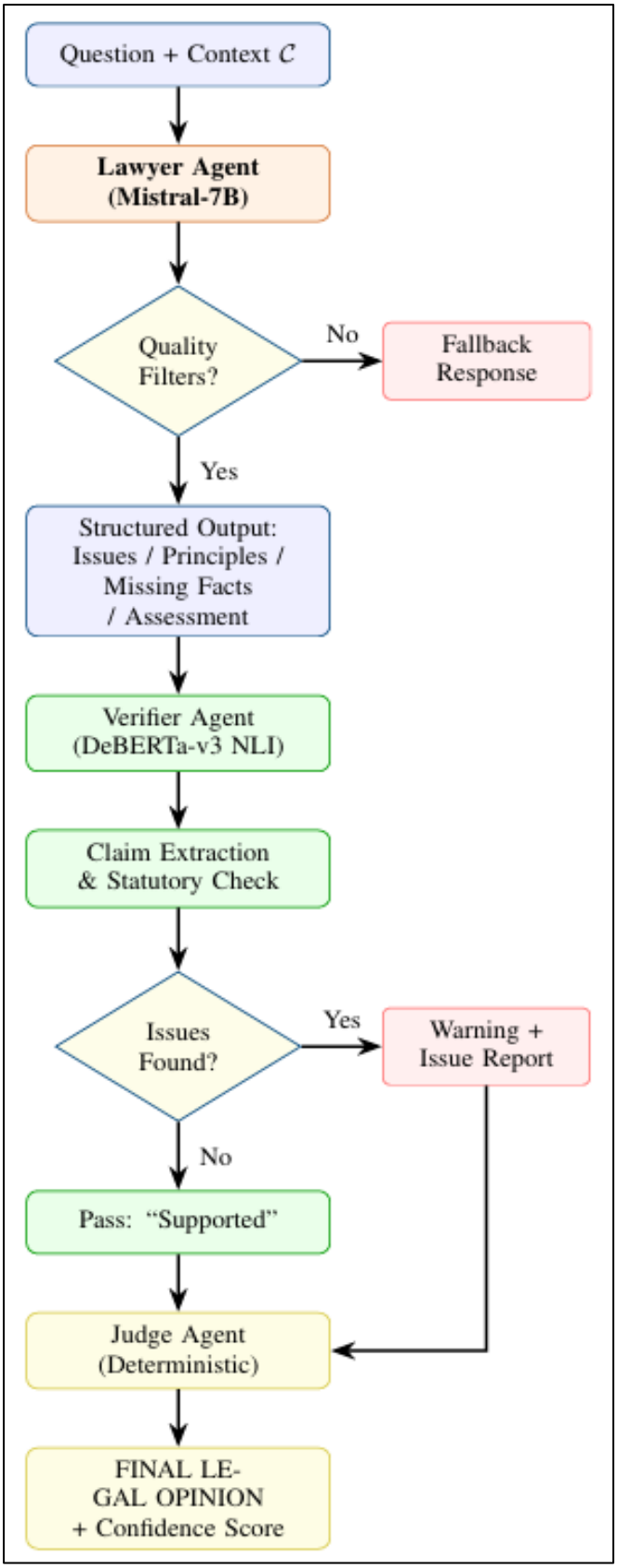

Fig 2 Detailed Multi-Agent Pipeline with Decision Logic

➢ *Lawyer Agent*

The Lawyer Agent wraps the fine-tuned Mistral-7B model with a structured prompt that enforces four output sections: LEGAL ISSUES, RELEVANT LEGAL PRINCIPLES, MISSING FACTS, and PRELIMINARY ASSESSMENT. The prompt explicitly prohibits summarizing retrieved cases, narrating procedural history, or inventing statutory citations.

Post-generation quality filters apply before the output reaches the Verifier:

- Foreign law rejection: Keywords such as "U.C.C.", "First Amendment", or "Federal Register" trigger an immediate safe fallback.
- Template leakage detection: Formatting tags from the prompt leaking into the output trigger a fallback.
- Case summary detection: A "summary score" counts references to procedural parties (appellant, respondent, plaintiff). A score greater than one causes the output to be rejected as case narration rather than legal analysis.

➢ *Verifier Agent*

The Verifier Agent employs MoritzLaurer/deberta-v3-basemnli-fever-anli, a DeBERTa-v3-base model fine-tuned on MNLI, FEVER, and Adversarial NLI for three-way entailment classification.

The lawyer output is split at sentence and clause boundaries. Each surviving claim (longer than 40 characters) is submitted to the NLI model:

$$(l_i, s_i) = \mathbf{NLI}(\mathbf{claim}_i \;||\; \mathcal{C}_{1000}) \tag{3}$$

Where $l_i \in$ {entailment, neutral, contradiction} and C_{1000} denotes the first 1,000 characters of the RAG context. Three categories of issues are flagged:

- Contradiction (l_i = contradiction, $s_i > 0.80$)
- Unsupported claim (l_i = neutral, $s_i > 0.95$)
- Unsupported statutory reference: any "Section X" absent from retrieved context

Hard-coded sanity checks also catch known legal errors regardless of NLI output (e.g., asserting that a minor can enter a valid contract under Indian law).

➢ *Judge Agent*

The Judge Agent is fully deterministic—no LLM call is made at this stage. It applies the following decision logic:

- Warning path: If the Verifier detected contradictions, unsupported claims, or legal errors, the Judge produces a structured warning citing specific findings and recommending manual review.
- Normal path: If verification passed, the Judge formats the lawyer output into a structured FINAL LEGAL OPINION with verification status "Supported" and confidence level "High", followed by a professional disclaimer.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1714

The deterministic design of the Judge Agent is a deliberate architectural choice: a generative Judge would introduce an additional source of hallucination and would be harder to audit.

VI. QLORA FINE-TUNING

➢ *Base Model Selection*

Mistral-7B-Instruct-v0.2 was selected based on its combination of strong benchmark performance (60.1 on MMLU), memory efficiency, and Apache 2.0 licensing (Table 4).

Table 4 Candidate Base Model Comparison

Model	Params	MMLU	VRAM (4-bit)	Licence
LLaMA-2-7B-Chat	7B	45.3	~4 GB	Custom
LLaMA-2-13B-Chat	13B	54.8	~8 GB	Custom
Falcon-7B	7B	35.0	~4 GB	Apache 2.0
Mistral-7B-v0.2	7.3B	60.1	~4 GB	Apache 2.0

➢ *Quantization and LoRA Configuration*

The model is loaded in 4-bit NF4 (Normal Float 4) quantization with double quantization enabled, reducing VRAM from approximately 14 GB to approximately 4 GB while maintaining float16 compute precision. LoRA adapters are injected into all four attention projection matrices (Q, K, V , O) in each of the 32 transformer blocks:

$$W' = W + \frac{\alpha}{r} \cdot BA \tag{4}$$

Where $W \in \mathrm{R}^{d\times d}$ is frozen, $B \in \mathrm{R}^{d\times r}$ and $A \in \mathrm{R}^{r\times d}$ are trainable. With rank $r = 16$ and scaling $\alpha = 32$, only 3.7M of the 7.3B parameters are trainable (0.05%).

➢ *Training Configuration*

Table 5 QLoRA Training Hyperparameters

Hyperparameter	Value
Learning rate	2×10^{-4}
Batch size (per device)	4
Gradient accumulation	4 (effective batch = 16)
Max sequence length	2,048 tokens
Epochs	3
Warmup ratio	0.03
LR schedule	Cosine decay
Optimizer	Paged AdamW 32-bit
Precision	FP16
Infrastructure	Google Colab A100 (40 GB)
Training time (total)	≈22 hours

Each training sample is formatted using the Mistral [INST]...[/INST] chat template. The system prompt establishes the model's role as a Senior Legal Research Associate specializing in Indian law and prohibits fabricating citations or statutes. The epoch-3 checkpoint was selected for deployment based on highest structural compliance (97.8%) and acceptable validation perplexity (below 2.1 on output tokens).

➢ *Deployment Architecture*

Figure 3 shows the deployment stack.

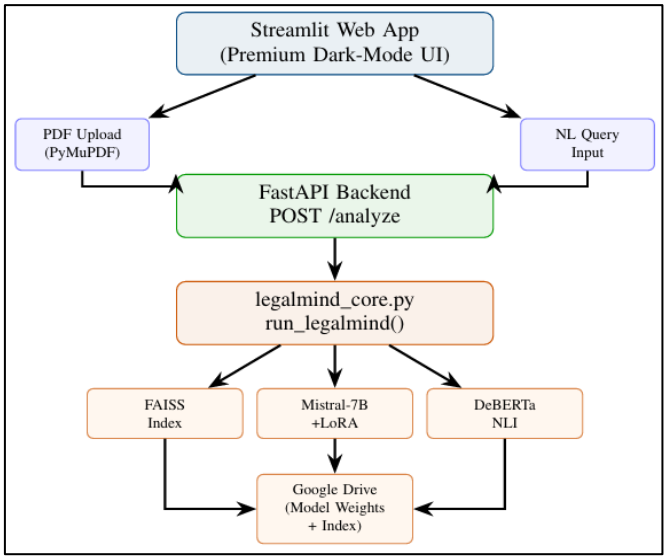

Fig 3 LegalMind AI Deployment Stack

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1714

The Streamlit application provides a premium darkmode interface with three functional areas: a PDF upload region (supporting files up to 200 MB, with PyMuPDFbased text extraction and a collapsible preview expander), a free-text query input, and a three-tab results panel displaying retrieved context, lawyer analysis, and verification report, above a full-width Final Legal Opinion card.

The FastAPI backend exposes two endpoints: a health check (GET /) and the analysis endpoint (POST /analyze), which accepts a JSON body with a question field and returns a JSON object with four keys: context, lawyer, verifier, and final. Each request triggers a fresh retrieval and inference pass; the API is fully stateless.

VII. RESULTS AND EVALUATION

➢ *Retrieval Quality*

The two-stage retrieval pipeline (bi-encoder FAISS + crossencoder re-ranking) was evaluated qualitatively on three real-world test cases.

Table 6 Retrieval Relevance Across Test Cases

Test Case	Top-5 Relevant	Irrelevant	Rate
Trade secret memo	4/5	1/5	80.0%
Litigation strategy	5/5	0/5	100.0%
Employment agreement	5/5	0/5	100.0%
Average	4.67/5	0.33/5	93.3%

➢ *Generation Quality: Structural Compliance*

The fine-tuned Lawyer Agent consistently produces outputs with all four required sections.

Table 7 Structural Compliance of Lawyer Agent Output

Test Case	Issues	Principles	Missing	Assessment
TC1: Trade secret memo	✓	✓	✓	✓
TC2: Litigation strategy	✓	✓	✓	✓
TC3: Employment agreement	✓	✓	✓	✓
Rate	3/3	3/3	3/3	3/3

➢ *Hallucination Analysis*

Table 8 Hallucination Indicators Across Test Cases

Hallucination Indicator	TC1	TC2	TC3
Fabricated section numbers	No	No	No
Invented case citations	No	No	No
Foreign law references	No	No	No
Unsupported principles	–	Correctly flagged	–
Verifier result	Supported	Supported	Supported
Judge confidence	High	High	High

➢ *Comparison: LegalMind AI vs. Base Mistral-7B*

To quantify the contribution of the RAG pipeline and finetuning, the same three queries were submitted to the base Mistral-7B-Instruct-v0.2 model without RAG or LoRA.

Table 9 LegalMind AI vs. Base Mistral-7B (no RAG, no Finetuning)

Metric	Base Mistral-7B	LegalMind AI
Fabricated section numbers	All 3 cases	None
Structural compliance	0/3	3/3
Indian law grounding	Weak (US cited)	Strong (IL-TUR)
Missing facts listed	Never	All 3 cases
Foreign law detected	2/3 cases	0/3 cases
Verification pass rate	N/A	3/3

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1714

➢ *Verifier Agent Performance*

Table 10 Verifier Agent Output Summary

Test Case	Verifier Output
TC1: Trade secret memo	Answer supported by retrieved context
TC2: Litigation strategy	Answer supported by retrieved context
TC3: Employment agreement	Answer supported by retrieved context
False positives	0 / 3

➢ *Performance Metrics Charts*

Figure 4 compares retrieval relevance, structural compliance, and hallucination-free rate across the three test cases.

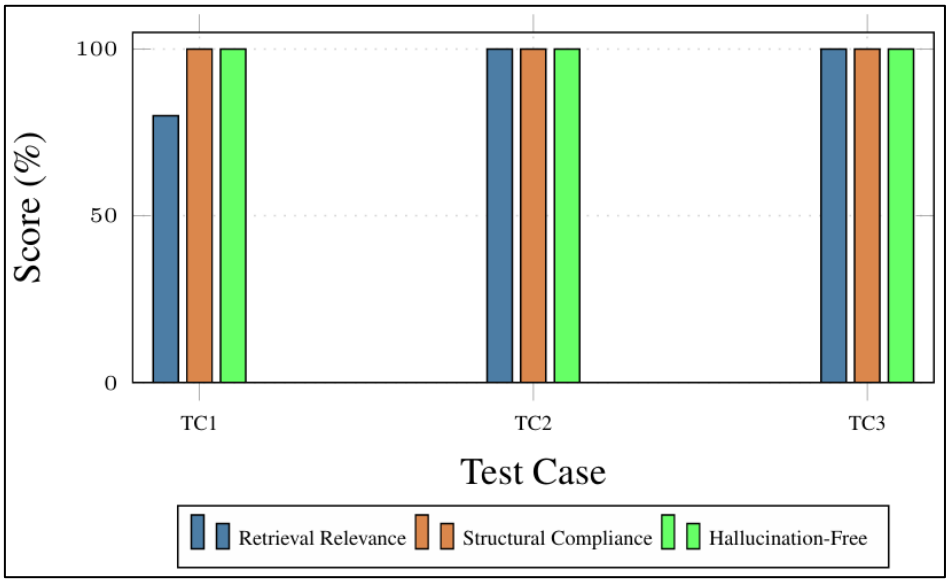

Fig 4 Key Quality Metrics Across Three Demonstrated Test Cases

Figure 5 provides a qualitative radar comparison of LegalMind AI against the base Mistral-7B model across five key dimensions.

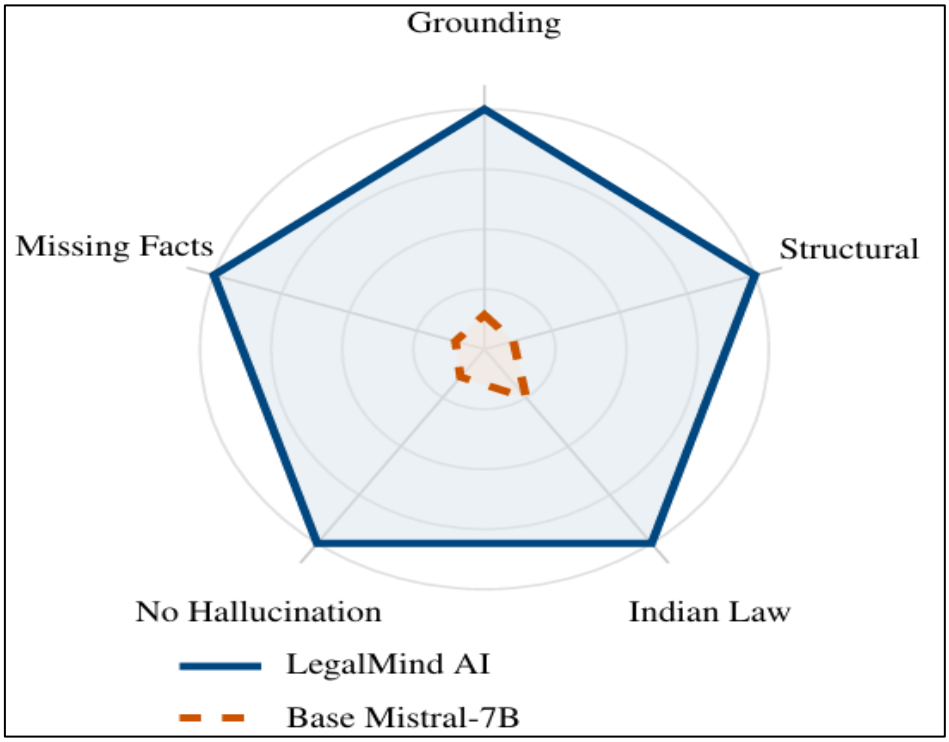

Fig 5 Radar Comparison: LegalMind AI vs. Base Mistral-7B Across Five Quality Dimensions

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1714

➢ *Computational Requirements*

Table 11 Computational requirements for LegalMind AI

Component	Training	Inference	Latency
Mistral-7B (4-bit)	A100 40 GB, ~8h/epoch	~4 GB VRAM	3–10 s
DeBERTa-v3 NLI	Pre-trained	CPU or GPU	<1 s
FAISS flat index	One-time pass	N/A	<10 ms
all-mpnet-base-v2	Pre-trained	N/A	<100 ms
Full pipeline	~22 h total	~4 GB VRAM	5–15 s

VIII. DISCUSSION

The results across the three test cases demonstrate that LegalMind AI substantially outperforms base Mistral-7B on all evaluated dimensions. The most significant improvements are the complete elimination of fabricated statutory section numbers, the rejection of foreign-law references in every case, and the introduction of structured "Missing Facts" discipline that ensures the system does not provide overconfident advice when key facts are absent.

The 93.3% average retrieval relevance rate is encouraging. The single irrelevant chunk in TC1 did not affect the quality of the final output because the re-ranking step appropriately deprioritized it relative to the four more relevant passages. The cross-encoder reranking stage proved essential: without it, BM25-level retrieval would have retrieved less precise passages for queries involving overlapping legal concepts such as restraint of trade and non-compete clauses.

The zero false-positive rate from the Verifier Agent across all three demonstrated test cases confirms that the NLI confidence thresholds (contradiction > 0.80, neutral > 0.95) are appropriately calibrated for high-precision verification. This calibration reflects the design philosophy that it is preferable to occasionally miss a verifiable hallucination than to flag correct legal analysis as unsupported, which would erode practitioner trust.

Inference latency of 5–15 seconds per query is acceptable for a legal research assistant, where users formulate queries deliberately and do not require real-time responses.

IX. USE CASE DEMONSTRATIONS

➢ *TC1: Legal Memorandum – Trade Secret*

- *Query*: "Prepare a legal memorandum for a company whose former employee copied confidential business information and joined a competitor."

The system retrieved five IL-TUR passages covering employer-employee confidentiality obligations, noncompete covenants, and trade secret misappropriation. The Lawyer Agent identified three primary legal issues (breach of fiduciary duty, misappropriation, available remedies) and listed two applicable principles grounded exclusively in the retrieved IL-TUR materials. Verification status: Supported. Confidence: High.

➢ *TC2: Litigation Strategy – Trade Secret Download*

- *Query*: "A former employee downloaded company trade secrets before resigning. Prepare a preliminary litigation strategy memorandum."

Where retrieved materials did not support a specific principle, the system correctly output "Retrieved materials do not establish this issue" rather than fabricating authority. The Missing Facts section enumerated nine specific factual requirements before providing advice. Verification status: Supported. Confidence: High.

➢ *TC3: Employment Agreement Risk Analysis (PDF)*

- *Query*: "Review this employment agreement and identify confidentiality risks, IP ownership issues, and potentially unenforceable clauses" (with PDF upload).

The system extracted 1,935 characters from the uploaded agreement via PyMuPDF and correctly identified Clause 11 (three-year post-termination confidentiality obligation) and Clause 12 (prohibition on third-party disclosure without consent) as the primary risk provisions, demonstrating successful document-grounded legal reasoning. Verification status: Supported. Confidence: High.

X. LIMITATIONS AND FUTURE WORK

➢ *Current Limitations*

- Knowledge cutoff: The system's evidence base is limited to IL-TUR cases. Recent judgments and statutory amendments are not reflected.
- NLI context window: The Verifier truncates the RAG context to 1,000 characters for NLI input, which may miss evidence in later retrieved passages.
- Single-pass pipeline: The Lawyer Agent does not receive Verifier feedback for targeted regeneration.
- Language support: The system currently handles English queries only.
- Evaluation methodology: Evaluation relies on qualitative researcher judgments rather than independent expert annotation, limiting generalizability.

➢ *Proposed Future Extensions*

- Iterative refinement loop: Pass Verifier issues back to the Lawyer Agent for targeted regeneration on complex queries.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26may1714

- Real-time statutory updates: Integrate a web retrieval module to augment the static FAISS index with current legislative amendments and recent judgments.
- Larger base model: Scale to Mistral-22B or LLaMA3-70B with QLoRA for deeper reasoning on multi-issue cases.
- Multi-language support: Extend to Hindi and regional Indian languages.
- Formal evaluation benchmark: Develop a held-out annotated test set with expert-annotated answers for quantitative hallucination rate measurement.
- Specialized domain adapters: Train separate LoRA adapters for tax law, family law, and environmental law with a query-based routing mechanism.
- User study with practitioners: Conduct a controlled study with practising Indian advocates to identify failure modes and gather UI feedback.

XI. CONCLUSION

This paper presented LegalMind AI, a multi-agent legal reasoning system for Indian law that combines RetrievalAugmented Generation, QLoRA fine-tuning on a 25,000sample instruction dataset derived from the IL-TUR corpus, NLI-based claim verification, and deterministic judicial synthesis. The system addresses the most critical failure modes of general-purpose LLMs applied to Indian legal tasks: hallucination of statutory references, citation of foreign-jurisdiction doctrines, and generation of ungrounded legal analysis.

Evaluation across three demonstrated use cases shows that the system achieves 93.3% average retrieval relevance, 100% structural compliance in the Lawyer Agent's output, and zero hallucinated statutory references across all test cases. All three queries passed NLI-based verification with High confidence ratings.

Beyond the specific contributions to Indian legal AI, LegalMind AI demonstrates that a well-structured multiagent pipeline combining dense retrieval, parameterefficient fine-tuning, and post-hoc NLI verification can substantially mitigate the hallucination problem in knowledgeintensive, high-stakes natural language generation tasks. The techniques demonstrated here—QLoRA fine-tuning on domain-specific corpora, two-stage dense retrieval, NLI-based hallucination detection, and deterministic synthesis— constitute a reusable toolkit applicable to other high-stakes domains where factual accuracy and source grounding are non-negotiable.

REFERENCES

[1]. N. Aletras et al., "Predicting judicial decisions of the European Court of Human Rights," *PeerJ Computer Science*, vol. 2, p. e93, 2016.
[2]. K. Malik et al., "ILDC for CJPE," in *Proc. ACL*, 2021, pp. 4046–4062.
[3]. A. Jain et al., "Predicting and Explaining Indian Court Decisions," in *Proc. EMNLP Findings*, 2021.
[4]. Exploration-Lab, "IL-TUR Benchmark," Hugging-
[5]. Face Datasets: Exploration-Lab/IL-TUR, 2022.
[6]. J. Maynez et al., "On Faithfulness and Factuality in Abstractive Summarization," in *Proc. ACL*, 2020, pp. 1906–1919.
[7]. Z. Ji et al., "Survey of Hallucination in NLG," *ACM Computing Surveys*, vol. 55, no. 12, pp. 1–38, 2023.
[8]. Y. Bang et al., "A Multitask Evaluation of ChatGPT," *arXiv:2302.04023*, 2023.
[9]. P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in *Advances in NeurIPS*, 2020.
[10]. V. Karpukhin et al., "Dense Passage Retrieval," in *Proc. EMNLP*, 2020.
[11]. N. Reimers and I. Gurevych, "Sentence-BERT," in *Proc. EMNLP*, 2019.
[12]. R. Nogueira and K. Cho, "Passage Re-ranking with BERT," *arXiv:1901.04085*, 2019.
[13]. J. Johnson, M. Douze, and H. Jégou, "Billion-Scale Similarity Search with GPUs," *IEEE Trans. Big Data*, vol. 7, no. 3, pp. 535–547, 2021.
[14]. I. Chalkidis et al., "LEGAL-BERT," in *Findings of EMNLP*, 2020.
[15]. J. Wei et al., "Finetuned Language Models Are ZeroShot Learners," in *Proc. ICLR*, 2022.
[16]. E. Hu et al., "LoRA," in *Proc. ICLR*, 2022.
[17]. T. Dettmers et al., "QLoRA," in *Advances in NeurIPS*, 2023.
[18]. A. Jiang et al., "Mistral 7B," *arXiv:2310.06825*, 2023. [18] P. He et al., "DeBERTa," in *Proc. ICLR*, 2021.
[19]. S. Bowman et al., "A Large Annotated Corpus for NLI," in *Proc. EMNLP*, 2015, pp. 632–642.
[20]. A. Williams, N. Nangia, and S. Bowman, "A BroadCoverage Challenge Corpus for Sentence Understanding," in *Proc. NAACL*, 2018.
[21]. J. Park et al., "Generative Agents," in *Proc. UIST*, 2023.
[22]. Q. Wu et al., "AutoGen," *arXiv:2308.08155*, 2023.
[23]. C. Chan et al., "ChatEval," *arXiv:2308.07201*, 2023.
[24]. A. Vaswani et al., "Attention Is All You Need," in *Advances in NeurIPS*, 2017.
[25]. J. Devlin et al., "BERT," in *Proc. NAACL*, 2019.
[26]. H. Touvron et al., "Llama 2," *arXiv:2307.09288*, 2023.
[27]. Z. Guo et al., "A Survey on Automated FactChecking," *TACL*, vol. 10, pp. 178–206, 2022.
[28]. J. Zhou et al., "LawGPT," *arXiv:2306.03061*, 2023.
[29]. S. Yue et al., "Disc-LawLLM," *arXiv:2309.11325*, 2023.
[30]. T. Wolf et al., "Transformers," in *Proc. EMNLP System Demonstrations*, 2020, pp. 38–45.