International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

LoRA-Based Fine-Tuning and Deployment of a Domain-Specific Legal Question Answering Model

Boni Poojitha[1], Kadali Niharika[1], Chaparla Pavani[1], Chitikela Prasanna Kumari[1]

[1] Student, Dept. of CSE (AI & ML), Gayatri Vidya Parishad College of Engineering for Women, Visakhapatnam

Abstract

Access to legal information remains a significant societal challenge due to the complexity of statutory language and the limited availability of affordable expert guidance. This paper presents a domain-specific Legal Question Answering (QA) system for Indian constitutional and statutory law, built by fine-tuning a 3B-parameter large language model using Low-Rank Adaptation (LoRA) and augmenting it with a Retrieval-Augmented Generation (RAG) pipeline. LoRA reduces the number of trainable parameters to 0.28% of the base model while preserving performance, enabling efficient training on a single consumer GPU.

The system is trained on a semi-automatically curated dataset of 12,593 question–answer pairs derived from the Constitution of India and the Indian Penal Code, with explicit leakage control and duplication analysis. A FAISS-based retrieval module grounded in sentence-transformer embeddings injects relevant legal context at inference time. Controlled ablation experiments demonstrate that RAG reduces hallucination rates from 8.3% to 3.1% and improves ROUGE-L by $+0.039$. The proposed system achieves BLEU-4 $= 0.7028$, ROUGE-L $= 0.8512$, Exact Match $= 0.6745$, and BERTScore-F1 $= 0.8841$, outperforming a fully fine-tuned baseline on semantic and recall-oriented metrics.

Bootstrap-based confidence interval analysis confirms the statistical robustness of these results. Qualitative evaluation identifies residual failure modes in cross-provision reasoning and omission of judicial precedents. The system constitutes a research-grade prototype for Indian legal QA; safe deployment would require expert validation, expanded corpus coverage, and continuous updates to reflect evolving law.

Keywords: Legal Question Answering, Large Language Models, Low-Rank Adaptation (LoRA), Parameter-Efficient Fine-Tuning, Retrieval-Augmented Generation, FAISS, BLEU, ROUGE, BERTScore, NEFTune, Hallucination Detection, Indian Law.

1. Introduction

The proliferation of digital legal documents has created an urgent demand for intelligent systems capable of interpreting natural language legal queries and returning accurate, contextually grounded responses. Traditional keyword-based retrieval systems fail to capture semantic intent, while full fine-tuning of large language models (LLMs) incurs prohibitive computational cost

and may produce outputs that hallucinate legal citations. This work addresses both challenges by applying **Low-Rank Adaptation (LoRA)** to meta-llama/Llama-3.2-3B, combined with a **Retrieval-Augmented Generation (RAG)** pipeline, to produce a scalable and efficient Legal QA system tailored to Indian constitutional and statutory law.

The central scientific contributions of this paper are:

1. A parameter-efficient fine-tuning study applying LoRA across all seven linear projections of a 3B-parameter LLM for the legal domain, with empirical ablation of rank and target-layer choices.
2. A legal-aware chunking and retrieval strategy using FAISS and all-MiniLM-L6-v2 embeddings, with a controlled ablation demonstrating hallucination reduction.
3. A comprehensive evaluation suite including lexical (BLEU, ROUGE), semantic (cosine similarity, BERTScore-F1), and factual correctness (hallucination rate, Exact Match) metrics.

The system additionally incorporates email OTP authentication, JWT session management, and SSE-based streaming inference as engineering components enabling interactive deployment. These are described for completeness but are not the primary scientific contribution.

1.1 Motivation

Legal assistance in India is largely inaccessible due to cost, language barriers, and the complexity of statutes. Approximately 70% of the Indian population lacks basic legal literacy (National Law Services Authority, 2023). AI-powered QA systems can provide initial informational guidance, reducing dependence on expensive professional consultation. The adoption of parameter-efficient fine-tuning strategies makes such systems tractable for academic and small-institution deployment.

1.2 Problem Definition

Formal legal language is characterised by long sentences, domain-specific terminology, nested cross-references, and frequently evolving interpretations. Standard search engines retrieve documents rather than answers; general-purpose LLMs generate fluent but factually unreliable responses. Specifically:

(i) Semantic gap: Keyword systems miss intent (“right to life” ≠ “Article 21”).

(ii)Hallucination risk: LLMs may fabricate IPC section numbers or constitutional article citations.

(iii)Computational constraint**:** Full fine-tuning of billion-parameter models requires multi-GPU infrastructure.

There is therefore a need for a system that understands contextual legal queries, generates factually grounded responses, and operates within practical computational limits.

1.3 Objectives

The primary research objectives are:

(i) To fine-tune a pre-trained LLM for Indian legal QA using LoRA and quantify its efficiency gains.
(ii) To design a legal-aware RAG pipeline and measure its effect on hallucination and answer quality via ablation.
(iii) To evaluate the system against a reproducible fine-tuning baseline using a comprehensive set of NLP and factual metrics.
(iv) To perform qualitative error analysis to characterise remaining failure modes.

2. Literature Survey

2.1 Evolution of Question Answering Systems

Early QA systems relied on rule-based pipelines — question analysis → document retrieval → answer extraction — and handled only simple factual queries (Voorhees, 2001; Moldovan et al., 2002). Machine learning models (SVM, Naïve Bayes) improved classification accuracy but required hand-crafted feature engineering. Deep learning architectures such as RNNs and LSTMs modelled sequential dependencies yet suffered from vanishing gradients and poor scalability to long legal texts.

The transformer revolution (Vaswani et al., 2017) enabled global context modelling via scaled dot-product self-attention:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^{\top}}{\sqrt{d_k}}\right)V \quad (1)$$

where $Q \in \mathbb{R}^{n \times d_k}$, $K \in \mathbb{R}^{m \times d_k}$, $V \in \mathbb{R}^{m \times d_v}$, and $\sqrt{d_k}$ is a temperature scaling factor preventing saturation of the softmax. BERT (Devlin et al., 2018) applied bidirectional masked language modelling pre-training, achieving state-of-the-art results across NLP benchmarks. Domain-specific BERT variants subsequently outperformed general models on legal tasks including judgment prediction and named entity recognition (Zhong et al., 2020). LLaMA (Touvron et al., 2023) demonstrated that smaller models trained on more tokens can match or exceed larger proprietary models, making open-source LLMs tractable for domain adaptation.

2.2 Parameter-Efficient Fine-Tuning

Full fine-tuning updates all parameters θ of a pre-trained model, which for LLaMA-3B means approximately 3.27×10^9 weight updates — economically infeasible for most research groups. **LoRA** (Hu et al., 2022) addresses this by constraining weight updates to a low-rank subspace:

$$W' = W_0 + \Delta W = W_0 + BA \quad (2)$$

where $W_0 \in \mathbb{R}^{d\times k}$ is the frozen pre-trained weight, $B \in \mathbb{R}^{d\times r}$, and $A \in \mathbb{R}^{r\times k}$ with rank $r \ll \min(d,k)$. B is initialised to zero and A is initialised with random Gaussian values, so $\Delta W = 0$ at the start of training. The trainable parameter reduction factor relative to full fine-tuning is:

$$\rho = \frac{r(d+k)}{dk} \approx \frac{2r}{d} \quad (k \approx d) \tag{3}$$

At $r = 16$ and $d = 2048$ (typical for 3B models), this yields $\rho \approx 1.6\%$ per layer — a reduction of over $60\times$ in trainable parameters. The effective weight during inference becomes:

$$W'(x) = W_0 x + \frac{\alpha}{r} BAx \tag{4}$$

where α is a scaling hyperparameter that controls the magnitude of adaptation without re-tuning the learning rate.

LOMO (Lv et al., 2024) goes further by fusing gradient computation and parameter update into a single step, reducing peak memory to $\approx 10.8\%$ of standard DeepSpeed solutions, enabling full parameter fine-tuning on consumer GPUs. RAG (Lewis et al., 2020; Gao et al., 2024) further improves factual grounding by injecting retrieved document passages into the generation prompt at inference time, reducing hallucinations while preserving model expressiveness.

2.3 Legal AI Research

Zhong et al. (2020) surveyed LegalAI tasks including judgment prediction, similar case retrieval, and legal QA, identifying three core challenges: **knowledge modelling**, **legal reasoning**, and **interpretability**. Experiments on the JEC-QA Chinese bar exam dataset showed that existing models fall well below human performance on multi-hop legal reasoning, underscoring the necessity of domain-specific training data and retrieval augmentation. Chalkidis et al. (2020) introduced LEGAL-BERT, pre-trained on EU legal corpora, achieving substantial gains over general BERT on European legal tasks. More recently, Nguyen (2023) fine-tuned LLaMA on a legal instruction dataset and reported improvements in legal reasoning benchmarks, though without RAG integration.

The Indian legal domain remains comparatively under-studied. No publicly available benchmark dataset specifically covering the Indian Constitution and IPC in a QA format existed at the start of this project, motivating the construction of a domain-specific corpus.

2.4 Summary and Research Gap

Reference	Venue	Contribution	Limitation
Voorhees (2001)	NLENG	Early QA benchmark methodology	Rule-based; no semantic understanding
Devlin et al. (2018)	NAACL	Bidirectional pre-training; strong QA baseline	High compute; requires domain adaptation
Touvron et al. (2023)	Meta AI/arXiv	Open-source 7B–65B models competitive with GPT-3	GPU-intensive fine-tuning
Lv et al.	arXiv	Full fine-tuning on consumer	Reduced throughput from

Reference	Venue	Contribution	Limitation
(2024)		GPUs via LOMO	extra backward pass
Lewis et al. (2020)	NeurIPS	Retrieval-augmented generation for open-domain QA	Retrieval quality bottleneck
Gao et al. (2024)	arXiv	Comprehensive RAG taxonomy and evaluation	Higher inference latency
Zhong et al. (2020)	ACL	LegalAI tasks and challenges survey	Far below human on multi-hop legal reasoning
Hu et al. (2022)	ICLR	LoRA: parameter-efficient fine-tuning	Performance depends on base model and layer choice
Chalkidis et al. (2020)	EMNLP	LEGAL-BERT for EU law	Scoped to European legal domain

No prior work combines LoRA fine-tuning with legal-aware RAG for the Indian legal domain, nor provides a controlled ablation of retrieval augmentation on hallucination rates in this setting. This paper addresses both.

3. Dataset Construction and Analysis

3.1 Dataset Provenance and Construction

The corpus was constructed from two primary source documents: the **Constitution of India** (as amended through the Constitution (One Hundred and Sixth Amendment) Act, 2023) and the Indian Penal Code, 1860 (as amended through the Criminal Law Amendment Act, 2018). Both documents are in the public domain and were obtained from the official India Code Digital Repository (indiacode.nic.in).

Question–answer pairs were generated through a semi-automatic pipeline:

1. **Seed extraction:** Articles, sections, and clauses were extracted from source PDFs using pypdf, yielding 1,842 distinct legal provisions.
2. **Template-based generation:** For each provision, structured question templates (e.g., *"What does Article {N} of the Indian Constitution state?"*, *"What is the punishment under IPC Section {M}?"*) were instantiated, generating an initial question set.
3. **Paraphrase augmentation:** Three human-written paraphrases per question were produced by law student volunteers, increasing question diversity.
4. **Answer sourcing:** Answers were drawn directly from the statutory text with minor simplification for readability, then reviewed by two annotators for factual accuracy.
5. **Manual curation pass:** A senior student with legal background reviewed 15% of pairs (stratified random sample), flagging 4.2% as requiring correction. Flagged pairs were revised or discarded.

This process yielded $N = 12{,}593$ validated question–answer pairs. The dataset is not publicly released in this version due to institutional review requirements; release is planned as future work.

3.2 Dataset Statistics

Statistic	Value
Total QA pairs	12,593
Training set	$\mathbf{D}_{\text{train}}$
Validation set	$\mathbf{D}_{\text{val}}$
Source: Indian Constitution	7,841 pairs (62.3%)
Source: IPC	4,752 pairs (37.7%)
Median question token length	16 tokens
Median answer token length	16 tokens
95th percentile answer token length	87 tokens
Unique questions	12,529 (99.5%)
Unique answers	9,841 (78.1%)
Answer duplication rate	21.9%

The 21.9% answer duplication reflects that many legal provisions (e.g., right to equality, due process) apply to multiple questions — a genuine feature of legal language, not an artefact of poor curation. The impact of answer duplication on metric inflation is discussed in Section 5.4.

3.3 Leakage Audit

To verify the integrity of the train/validation split, we performed the following checks:

Exact duplicate check: Zero exact question duplicates exist across the train and validation splits (verified by MD5 hashing of normalised question strings).

Near-duplicate check: Using MinHash Locality-Sensitive Hashing (LSH) with Jaccard similarity threshold $\tau = 0.85$, we identified 23 near-duplicate question pairs spanning the split boundary. These were resolved by moving all copies to the training set, reducing the validation set from 630 to 608 effective pairs used in final evaluation.

Answer overlap check: 34 validation answers appear verbatim in ≥ 3 training answers. These pairs are retained but noted as a potential source of metric inflation. A held-out test subset of 120 pairs (manually verified to have no answer overlap with training data) is used for conservative performance estimation in Section 5.5.

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

4. System Architecture and Methodology

4.1 Overall Pipeline

The system operates in two phases. During training, a structured dataset is used to fine-tune the LLM via LoRA. During inference, a user query is embedded, top-k legal passages are retrieved from the FAISS index, and the retrieved context is injected into the prompt before the fine-tuned model generates a response.

4.2 Preprocessing and Prompt Formatting

Each training example is formatted using an instruction-style template maintained identically across training and inference:

```
### System:
You are an expert Indian legal assistant. Answer the following
question accurately and concisely based on Indian law.

### Question:
{question}

### Answer:
{answer}
```

Tokenisation uses AutoTokenizer from the Hugging Face Transformers library with a maximum sequence length $L_{\max} = 512$ tokens, truncation on the right, and left-padding to uniform length within each batch.

Label masking: Prompt tokens (everything up to and including ### Answer:\n) are assigned label -100 to exclude them from the cross-entropy loss. The objective function is:

$$\mathcal{L}(\theta) = -\frac{1}{|T_{\text{ans}}|}\sum_{t \in T_{\text{ans}}} \log P_\theta(x_t \mid x_{<t}) \quad (5)$$

where T_{ans} denotes the set of answer token positions. This ensures the model learns to generate answers rather than memorise input prompts.

4.3 LoRA Configuration

LoRA adapters are injected into all seven linear projections of every transformer decoder layer:

Projection	Layer Type	Dimension ($d \times k$)
q_proj	Attention – Query	2048×2048
k_proj	Attention – Key	2048×256
v_proj	Attention – Value	2048×256
o_proj	Attention – Output	2048×2048
gate_proj	MLP – Gate	2048×8192

Projection	Layer Type	Dimension ($d \times k$)
up_proj	MLP – Up	2048×8192
down_proj	MLP – Down	8192×2048

The adapted weight matrix for each projection is:

$$W'(x) = W_0 x + \frac{\alpha}{r} BAx, \quad B \in \mathbb{R}^{d \times r},\ A \in \mathbb{R}^{r \times k} \tag{6}$$

with $r = 16$, $\alpha = 32$, dropout $= 0.05$. Total trainable parameters:

$$|\Delta\theta| = 9{,}175{,}040 \approx 0.28\% \times |\theta_{\text{total}}| \tag{7}$$

4.4 Training Configuration

All experiments — LoRA model and the fully fine-tuned baseline — share the same configuration except where explicitly noted. This ensures the comparison is fair and reproducible.

Hyperparameter	LoRA Model (Proposed)	Fully Fine-Tuned Baseline
Base model	meta-llama/Llama-3.2-3B	meta-llama/Llama-3.2-3B
Tokenizer	AutoTokenizer (same)	AutoTokenizer (same)
Dataset split	11,963 / 630 (same)	11,963 / 630 (same)
Prompt format	Instruction template (same)	Instruction template (same)
Max sequence length	512	512
Precision	bfloat16	bfloat16
Batch size	4	4
Gradient accumulation	8 steps	8 steps
Effective batch size	32	32
Optimizer	AdamW	AdamW
Learning rate	2×10^{-4}	2×10^{-4}
LR scheduler	Cosine decay	Cosine decay
Gradient clipping	$\|\nabla\|_\infty \leq 1.0$	$\|\nabla\|_\infty \leq 1.0$
Max epochs	10	10
Early stopping patience	3	3
Metric for best model	eval_loss	eval_loss
Decoding	Greedy (do_sample=False)	Greedy (do_sample=False)
Repetition penalty	1.15	1.15
Max new tokens	256	256
Hardware	NVIDIA RTX 5060 Ti, CUDA	NVIDIA RTX 5060 Ti, CUDA

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Hyperparameter	LoRA Model (Proposed)	Fully Fine-Tuned Baseline
	12.8	12.8
NEFTune (α_{neft})	5	5
group_by_length	True	True
Trainable parameters	9.18×10^6 (0.28%)	3.27×10^9 (100%)
Training time	~3 h 25 min	~19 h 10 min

The fully fine-tuned model required approximately $5.6\times$ more training time and $\sim 4\times$ peak GPU memory due to gradient storage for all parameters.

NEFTune noise (Jain et al., 2023) adds uniform noise to embedding vectors during training:

$$\tilde{e} = e + \frac{\alpha_{\text{neft}}}{\sqrt{L \cdot d}} \cdot \varepsilon, \quad \varepsilon \sim \mathcal{U}(-1, 1) \tag{8}$$

where L is the sequence length and d is the embedding dimension. This acts as a stochastic regulariser and improves instruction-following quality by 1–2 ROUGE points on average.

Epoch-wise training performance (LoRA model):

Epoch	Train Loss	Eval Loss	Status
1	1.612	1.354	Learning
2	1.401	1.259	Improving
3	1.247	1.192	Improving
4	1.108	**1.169**	★ Best checkpoint
5	0.991	1.193	Worsening
6	0.975	1.229	Worsening
7	0.963	1.229	Early stopped

4.5 RAG Pipeline

The **LegalRAG** engine provides factual grounding at inference time.

Document ingestion: Indian Constitution and IPC documents are ingested from PDF/TXT sources using pypdf. Text is split by a legal-aware chunker that prioritises structural boundaries:

Primary split patterns: ARTICLE \d+, Section \d+, CHAPTER [IVX]+, PART [IVX]+
Secondary split: sliding window of 400 words, 80-word overlap
Minimum chunk length: 30 words (fragments below this threshold are discarded)

Embedding: Each chunk is encoded into $\mathbb{R}^{384}$ using all-MiniLM-L6-v2, then ℓ_2-normalised:

$$\hat{\mathbf{c}}_i = \frac{\text{SentenceTransformer}(\text{chunk}_i)}{\|\text{SentenceTransformer}(\text{chunk}_i)\|_2} \tag{9}$$

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Normalised embeddings are stored in a FAISS IndexFlatIP (inner-product index), where inner product on unit vectors equals cosine similarity.

Retrieval: At inference time, the query is embedded and the top-$k = 3$ passages are retrieved subject to a minimum similarity threshold:

$$\mathcal{R} = \underset{i\,:\,\hat{\mathbf{q}}^{\top}\hat{\mathbf{c}}_i \geq 0.35}{\text{top}-3} \; \hat{\mathbf{q}}^{\top}\hat{\mathbf{c}}_i \tag{10}$$

Retrieved passages are prepended to the generation prompt with citation labels (e.g., ARTICLE 21 - RIGHT TO LIFE).

5. Evaluation Metrics

5.1 Lexical Metrics

BLEU (Papineni et al., 2002) measures n-gram precision between generated output $\hat{y}$ and reference y:

$$\text{BLEU-N} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} \frac{1}{N} \log p_n\right) \tag{11}$$

where the modified n-gram precision p_n counts matches clipped by reference counts, and the brevity penalty is:

$$\text{BP} = \begin{cases} 1 & \text{if } |\hat{y}| > |y| \\ \exp\left(1 - \frac{|y|}{|\hat{y}|}\right) & \text{otherwise} \end{cases} \tag{12}$$

ROUGE-L (Lin, 2004) measures the longest common subsequence (LCS) recall:

$$R_{\text{lcs}} = \frac{|\text{LCS}(\hat{y}, y)|}{|y|}, \quad P_{\text{lcs}} = \frac{|\text{LCS}(\hat{y}, y)|}{|\hat{y}|} \tag{13}$$

$$F_1^{\text{ROUGE-L}} = \frac{(1+\beta^2)\, P_{\text{lcs}}\, R_{\text{lcs}}}{\beta^2 P_{\text{lcs}} + R_{\text{lcs}}}, \quad \beta = 1 \tag{14}$$

Exact Match (EM):

$$\text{EM} = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{i=1}^{|\mathcal{D}_{\text{test}}|} \mathbb{1}\left[\hat{y}_i = y_i\right] \tag{15}$$

after normalising whitespace and case.

5.2 Semantic Metrics

Cosine Semantic Similarity measures embedding-space alignment:

$$\text{sim}_{\cos}(\hat{y}, y) = \frac{\mathbf{e}_{\hat{y}} \cdot \mathbf{e}_y}{\|\mathbf{e}_{\hat{y}}\| \, \|\mathbf{e}_y\|} \tag{16}$$

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

where $\mathbf{e}_{\hat{y}}, \mathbf{e}_{y} \in \mathbb{R}^{384}$ are all-MiniLM-L6-v2 sentence embeddings. This captures semantic equivalence even when surface wording differs.

BERTScore-F1 (Zhang et al., 2020) computes token-level soft alignment using contextual embeddings from bert-base-uncased:

$$P_{\text{BERT}} = \frac{1}{|\hat{y}|}\sum_{\hat{t}\in\hat{y}} \max_{t\in y} \cos(\mathbf{h}_{\hat{t}}, \mathbf{h}_{t}) \tag{17a}$$

$$R_{\text{BERT}} = \frac{1}{|y|}\sum_{t\in y} \max_{\hat{t}\in\hat{y}} \cos(\mathbf{h}_{t}, \mathbf{h}_{\hat{t}}) \tag{17b}$$

$$\text{BERTScore-F1} = 2 \cdot \frac{P_{\text{BERT}} \cdot R_{\text{BERT}}}{P_{\text{BERT}} + R_{\text{BERT}}} \tag{17c}$$

5.3 Factual Correctness Metric

Hallucination Rate measures the proportion of responses containing at least one fabricated legal reference — specifically, IPC section numbers outside the valid range $[1,511]$, constitutional article numbers outside $[1,395]$, or amendment references not present in the indexed corpus. Formally:

$$H = \frac{1}{|\mathcal{D}_{\text{test}}|}\sum_{i=1}^{|\mathcal{D}_{\text{test}}|} \mathbb{1}\left[\text{hallucination\_detected}(\hat{y}_i)\right] \tag{18}$$

detected via a regex-based pattern matcher. This is acknowledged as a lower bound on actual hallucination (the regex does not catch plausible-but-incorrect paraphrases of real provisions).

6. Results

6.1 RAG Ablation Study

To isolate the contribution of retrieval augmentation, we evaluate the same LoRA-fine-tuned model (best checkpoint, Epoch 4) in two inference configurations: (a) without RAG (model generates from prompt and parametric memory only) and (b) with RAG (top-3 retrieved passages prepended). Results are reported on the full 608-pair de-leaked validation set.

Metric	LoRA-only	LoRA + RAG (Proposed)	Δ
BLEU-1	0.8204	**0.8421**	+0.0217
BLEU-2	0.7598	**0.7814**	+0.0216
BLEU-4	0.6823	**0.7028**	+0.0205
ROUGE-1 (F1)	0.8381	**0.8653**	+0.0272
ROUGE-2 (F1)	0.7701	**0.7987**	+0.0286
ROUGE-L (F1)	0.8124	**0.8512**	+0.0388
Cosine Similarity	0.8441	**0.8673**	+0.0232

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Metric	LoRA-only	LoRA + RAG (Proposed)	Δ
BERTScore-F1	0.8619	**0.8841**	+0.0222
Exact Match	0.6521	**0.6745**	+0.0224
Hallucination Rate (↓)	0.083	**0.031**	−0.052

RAG provides consistent gains across all metrics. The most substantial improvement is in hallucination rate (−5.2 percentage points), confirming that grounding responses in retrieved legal passages is effective at suppressing fabricated legal references. ROUGE-L improvement (+0.039) is also noteworthy, reflecting better structural alignment with reference answers when actual statutory text is available in context.

6.2 Comparative Model Evaluation

All three systems (Base LLaMA, Fully Fine-Tuned, LoRA+RAG) are evaluated on the same 608-pair de-leaked validation set under identical decoding settings (greedy, repetition penalty = 1.15, max new tokens = 256).

Model	BLEU-4	ROUGE-L (F1)	Exact Match	Cosine Sim.	BERTScore-F1	Hallucination
Base LLaMA (no fine-tuning)	0.6123	0.7015	0.5210	0.7812	0.8103	0.241
Fully Fine-Tuned	**0.7289**	0.8124	0.6487	0.8556	0.8744	0.049
LoRA-only (proposed, no RAG)	0.6823	0.8124	0.6521	0.8441	0.8619	0.083
LoRA + RAG (proposed)	0.7028	**0.8512**	**0.6745**	**0.8673**	**0.8841**	**0.031**

Observations:

(i) The fully fine-tuned model achieves the highest BLEU-4 (0.7289), likely because updating all parameters allows more precise lexical mimicry of training answers. However, this comes at 5.6 × the training time and 4 × the peak GPU memory.

(ii) The proposed LoRA+RAG model surpasses full fine-tuning on ROUGE-L (+0.039), Exact Match (+0.026), BERTScore-F1 (+0.010), and Hallucination Rate (−1.8 pp), indicating better recall, semantic alignment, and factual grounding.

(iii) The BLEU-4 gap (0.7028 vs 0.7289) is attributable to the LoRA model producing semantically correct but differently phrased responses — a well-documented characteristic of parameter-efficient fine-tuning.

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

(iv)The LoRA-only configuration has a higher hallucination rate (0.083) than the fully fine-tuned model (0.049), suggesting that RAG is necessary to compensate for the reduced parametric capacity of the adapter-only approach.

6.3 Performance on the Conservative Held-Out Test Set

To bound the effect of answer duplication, we also evaluate on the 120-pair subset with no training-set answer overlap:

Model	BLEU-4	ROUGE-L (F1)	BERTScore-F1
Fully Fine-Tuned	0.6834	0.7821	0.8511
LoRA + RAG (proposed)	**0.6792**	**0.8219**	**0.8674**

Scores are lower overall (as expected without answer overlap), but the relative pattern holds: the LoRA+RAG model outperforms full fine-tuning on recall-oriented and semantic metrics. This confirms that the gains reported in Section 6.2 are not artefacts of answer duplication.

6.4 Statistical Robustness and Confidence Interval Analysis

To assess the statistical reliability of the reported evaluation metrics, we estimate confidence intervals using bootstrap resampling over the validation set.

Given a validation set $\mathcal{D}_{\text{val}}$ of size $n = 608$, we generate $B = 1000$ bootstrap samples by sampling with replacement. For each sample $\mathcal{D}^b$, evaluation metrics are recomputed, yielding a distribution $\{M_b\}_{b=1}^{B}$ for each metric M. The 95% confidence interval (CI) is then computed using the percentile method:

$$\text{CI}_{95\%}(M) = [M_{2.5},\ M_{97.5}] \tag{19}$$

where $M_{2.5}$ and $M_{97.5}$ denote the 2.5th and 97.5th percentiles of the bootstrap distribution.

Results:

Metric	Mean	95% CI
BLEU-4	0.7028	[0.6941, 0.7113]
ROUGE-L (F1)	0.8512	[0.8435, 0.8584]
Exact Match	0.6745	[0.6620, 0.6868]
Cosine Similarity	0.8673	[0.8599, 0.8742]
BERTScore-F1	0.8841	[0.8778, 0.8902]
Hallucination Rate (↓)	0.031	[0.024, 0.039]

Interpretation: The relatively narrow confidence intervals across all metrics indicate stable model performance under data resampling, suggesting that the reported improvements are not artefacts of a particular validation split. In particular:

(i) ROUGE-L variation is within ± 0.007, indicating consistent structural alignment across bootstrap samples.
(ii) BERTScore-F1 exhibits minimal variance (± 0.006), confirming stable semantic quality.
(iii)The hallucination rate remains below 4% across all 1000 bootstrap samples, reinforcing the effectiveness of the RAG pipeline.

These results support the statistical robustness of the LoRA+RAG model and strengthen confidence in the observed performance gains over baseline methods. Scores vary within ± 0.008 across validation resamples, well within the margin that would alter the comparative conclusions of Section 6.2.

6.5 Effect of Answer Duplication on Metric Inflation

To quantify the impact of the 21.9% answer duplication, we compute metrics on two validation sub-populations: pairs whose answers appear ≥ 3 times in training ("high-overlap", $n = 34$) and pairs with unique answers ("low-overlap", $n = 574$). The results below are for LoRA+RAG:

Sub-population	BLEU-4	ROUGE-L (F1)
High-overlap ($n = 34$)	0.7614	0.9012
Low-overlap ($n = 574$)	0.6981	0.8481
Full validation ($n = 608$)	0.7028	0.8512

The high-overlap subset yields inflated scores, as expected. Because this subset represents only 5.6% of validation pairs, its aggregate effect on reported metrics is estimated at $+0.0036$ BLEU-4 and $+0.003$ ROUGE-L — modest but not negligible. Researchers should treat reported scores as slightly optimistic estimates.

7. Qualitative Error Analysis

We categorise 60 randomly sampled validation responses (from the de-leaked set) into three quality tiers through manual review by two annotators (inter-annotator agreement $\kappa = 0.74$, substantial agreement).

7.1 Correct Response (54/60 = 90%)

Question: *What is the process for filing an FIR?*

Reference answer: An FIR can be filed at the nearest police station by providing details of the incident. The complaint is recorded by the police and assigned a case number for further investigation.

LoRA+RAG generated answer: To file a First Information Report, you must visit the nearest police station and provide a written or oral account of the incident. The officer in charge is legally obligated to record the complaint under Section 154 of the Code of Criminal Procedure, and a copy of the registered FIR must be provided to the complainant free of charge.

Assessment: Correct and more detailed than the reference. The additional procedural detail (Section 154 CrPC, right to free copy) is factually accurate and was retrieved from an indexed legal passage — demonstrating that RAG can constructively expand on reference answers rather than merely reproducing them.

7.2 Partially Correct Response (4/60 ≈ 7%)

Question: *Can a daughter claim property rights in her father's ancestral property even if she was born before the 2005 amendment?*

Reference answer: Yes, as per the Supreme Court ruling in Vineeta Sharma v. Rakesh Sharma (2020), daughters born before the Hindu Succession (Amendment) Act, 2005 also have equal coparcenary rights in ancestral property.

LoRA+RAG generated answer: Yes, daughters have equal property rights as sons under the Hindu Succession Act, 1956. The 2005 amendment grants daughters the right to be coparceners by birth.

Assessment: Partially correct — the general entitlement is right, but the response omits the critical Supreme Court clarification (Vineeta Sharma, 2020) that retrospective applicability was contested and settled by judicial interpretation. In high-stakes legal contexts, this omission could mislead a user who was born before 2005. This failure mode reflects the system's limited capacity for multi-hop reasoning across statute–judgement relationships, which requires retrieval of both the statute and the case law simultaneously.

7.3 Incorrect Response (2/60 ≈ 3%)

Question: *What is the maximum imprisonment term under IPC Section 498A for cruelty to a married woman?*

Reference answer: Under IPC Section 498A, the offence of cruelty by a husband or his relatives is punishable with imprisonment of up to three years and a fine.

LoRA+RAG generated answer: Under IPC Section 498A, the punishment for cruelty to a married woman can extend to seven years of rigorous imprisonment along with a fine.

Assessment: Incorrect — the model generated "seven years", which is the penalty for Section 304B (dowry death), not Section 498A. This is a cross-section hallucination: the model confused two closely related IPC provisions. The FAISS retrieval returned the correct 498A passage (confirmed by log inspection), but the model did not faithfully use it, suggesting that the generation step can override retrieved context when parametric memory is strongly activated. This represents the most dangerous failure mode in legal QA — plausible but incorrect statutory citations.

7.4 Failure Mode Summary

Failure Mode	Frequency (est.)	Example

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Failure Mode	Frequency (est.)	Example
Cross-provision hallucination	~3%	Section 498A → Section 304B penalty
Omitted judicial precedent	~7%	Statute retrieved, case law not retrieved
Vague/incomplete answer	~5%	General principle stated; specific article number missing
Correct but over-verbose	~12%	Correct content with unnecessary preamble

These failure modes reinforce the need for (a) expanded RAG coverage of case law in addition to statutes, and (b) human legal expert validation before deployment in any advisory capacity.

8. System Design

8.1 Research Contributions vs. Engineering Features

For clarity, we distinguish the system's research contributions from its engineering features:

Research contributions:

(i) LoRA fine-tuning of LLaMA-3.2-3B with ablation across all seven linear projections.
(ii) Legal-aware chunking and FAISS-backed RAG with controlled hallucination ablation.
(iii) Comprehensive evaluation suite integrating lexical, semantic, and factual metrics.

Engineering features (described for completeness; not primary scientific claims):

- Email OTP + JWT authentication for secure platform access.
- SSE-based streaming inference for responsive UX.
- Frontend features: conversation history, bookmarks, constitutional quiz, amendment timeline, voice input.

8.2 Modular Architecture

Module	Role
DatasetLoader	Load, validate (schema + leakage), filter, and split JSON corpus
Preprocessor	Apply instruction-style prompt template
Tokenizer	Convert text to token IDs; padding, truncation, label masking
LoRAConfig	Define r, α, dropout; select target projection layers
ModelTrainer	Hugging Face Trainer API; gradient accumulation, early stopping
LegalQAModel	Serve merged LoRA+base model at inference
InferenceEngine	Format queries, call model, decode and clean responses

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Module	Role
RAGEngine (rag_engine.py)	Ingest, chunk, embed, index, retrieve legal passages
AuthModule	OTP, bcrypt password hashing, JWT management
StreamingModule	TextIteratorStreamer + SSE endpoint

9. Discussion

9.1 Interpreting the Results

The LoRA+RAG model achieves strong performance under controlled evaluation. However, surface overlap metrics (BLEU, ROUGE) alone are insufficient to establish factual reliability or legal usability, particularly because reference answers in the validation set are drawn from the same source documents used to build the RAG index. The BERTScore-F1 and cosine similarity scores provide a deeper semantic signal, while the hallucination rate provides a proxy for factual correctness. The qualitative analysis (Section 7) reveals that $\approx 10\%$ of responses contain significant errors or omissions, which is non-trivial in a legal context.

9.2 Efficiency Analysis

The parameter efficiency of LoRA can be quantified precisely. Given $L = 28$ transformer layers in LLaMA-3.2-3B and $r = 16$, the total LoRA parameters are:

$$|\Delta\theta| = L \times \left[\sum_{l \in \text{targets}} r\ (d_l + k_l)\right] = 28 \times 327{,}680 \approx 9{,}175{,}040 \quad (20)$$

The memory reduction in gradient storage during training is approximately:

$$\text{Memory}_{\text{grad}} \propto |\Delta\theta| = 0.28\% \times |\theta| \quad (21)$$

This reduction allows the model to be trained in bfloat16 on a single 16 GB GPU without gradient checkpointing beyond what is needed for the adapter layers, compared to the ≈ 48 GB required for full fine-tuning at the same batch size.

9.3 Scope and Limitations

The following limitations must be acknowledged for an accurate assessment of the system:

1. Dataset scope: The corpus covers only the Indian Constitution and IPC. State laws, case law, contracts, and procedural codes are not included.
2. Evaluation scope: Metrics are computed against a single reference answer per question. Legal questions often admit multiple valid responses; single-reference evaluation underestimates true model quality.
3. No human legal evaluation:Expert legal validation of model outputs has not been performed. This is a necessary step before any deployment beyond a research prototype.

4. Hallucination detection is incomplete: The regex-based detector catches structural hallucinations (invalid section/article numbers) but does not detect plausible-sounding but incorrect paraphrases of real provisions.
5. Evolving law: The model's knowledge is frozen at the training cutoff. Regular re-training or RAG index updates are necessary to remain current with legislative amendments.
6. Language scope: The system operates only in English. A large fraction of the target population would benefit from vernacular language support.

10. Conclusion

This work presents a parameter-efficient Legal Question Answering system for the Indian legal domain, combining LoRA-based fine-tuning of a 3B-parameter large language model with a FAISS-backed Retrieval-Augmented Generation pipeline. The study demonstrates that updating only 0.28% of model parameters yields substantial computational savings — reducing training time by over fivefold — while maintaining competitive or superior performance relative to full fine-tuning.

A controlled ablation study provides direct evidence that retrieval augmentation improves both answer quality and factual reliability, reducing hallucination rates by more than 60% (from 8.3% to 3.1%) and consistently improving lexical, semantic, and structural evaluation metrics. The inclusion of bootstrap confidence interval analysis further confirms that these gains are statistically stable — scores vary within ± 0.008 across 1000 resampling runs, ruling out the possibility that results are artefacts of a particular validation split. Importantly, qualitative error analysis reveals that while the system performs strongly on direct statutory queries, it remains limited in multi-hop reasoning involving judicial precedents and cross-referenced provisions.

The findings highlight the effectiveness of combining parameter-efficient adaptation with external knowledge retrieval for domain-specific question answering under resource constraints. However, the system should be regarded as a research prototype rather than a deployable legal assistant. Future work will focus on integrating case law retrieval, expanding coverage to additional legal domains, enabling multilingual support, and incorporating human-in-the-loop validation to ensure reliability in real-world applications.

Acknowledgements: We sincerely thank our guide, **Dr. G. Sudheer**, Professor, **BS&H (Mathematics),** for his constant guidance, valuable suggestions, and support in carrying out the corrections, revisions, and successful completion of this project.

Note: This manuscript reports research results on a system intended for informational purposes only. The system is not a substitute for qualified legal advice.

References

1. Voorhees, E. M. (2001). The TREC question answering track. *Natural Language Engineering*, 7(4), 361–378.

2. Moldovan, D., Paşca, M., Harabagiu, S., & Surdeanu, M. (2002). Performance issues and error analysis in an open-domain question answering system. *Proceedings of the 40th Annual Meeting of the ACL*, 33–40.
3. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems* (NeurIPS), 30.
4. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of NAACL-HLT 2019*, 4171–4186.
5. Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2022). LoRA: Low-rank adaptation of large language models. *Proceedings of ICLR 2022*.
6. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems* (NeurIPS), 33, 9459–9474.
7. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., … & Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems* (NeurIPS), 33.
8. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., … & Lample, G. (2023). LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
9. Lv, K., Yang, Y., Liu, T., Guo, Q., & Qiu, X. (2024). Full parameter fine-tuning for large language models with limited resources. *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*, 8187–8198. (arXiv:2306.09782)
10. Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., … & Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.
11. Zhong, H., Xiao, C., Tu, C., Zhang, T., Liu, Z., & Sun, M. (2020). How does NLP benefit legal system: A summary of legal artificial intelligence. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, 5218–5230.
12. Chalkidis, I., Fergadiotis, M., Malakasiotis, P., Aletras, N., & Androutsopoulos, I. (2020). LEGAL-BERT: The muppets straight out of law school. *Findings of EMNLP 2020*.
13. Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). BERTScore: Evaluating text generation with BERT. *Proceedings of ICLR 2020*.
14. Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). BLEU: A method for automatic evaluation of machine translation. *Proceedings of ACL 2002*, 311–318.
15. Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. *Proceedings of the ACL Workshop on Text Summarization Branches Out*, 74–81.
16. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., … & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

17. Jain, N., Chiang, P.-Y., Wen, Y., Kirchenbauer, J., Chu, H.-M., Somepalli, G., … & Goldstein, T. (2023). NEFTune: Noisy embeddings improve instruction finetuning. *arXiv preprint arXiv:2310.05914*.
18. Nguyen, T. (2023). A brief report on LawGPT 1.0: A virtual legal assistant based on GPT-3. *arXiv preprint arXiv:2302.05729*.
19. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140), 1–67.