symmetry

MDPI

Article

Optimizing Legal Text Summarization Through Dynamic Retrieval-Augmented Generation and Domain-Specific Adaptation

S Ajay Mukund *,† and K. S. Easwarakumar †

Department of Computer Science and Engineering, College of Engineering Guindy, Anna University, Chennai 600025, India; easwara@annauniv.edu

* Correspondence: ajaymukund1998@gmail.com

† These authors contributed equally to this work.

Abstract: Legal text summarization presents distinct challenges due to the intricate and domain-specific nature of legal language. This paper introduces a novel framework integrating dynamic Retrieval-Augmented Generation (RAG) with domain-specific adaptation to enhance the accuracy and contextual relevance of legal document summaries. The proposed Dynamic Legal RAG system achieves a vital form of symmetry between information retrieval and content generation, ensuring that retrieved legal knowledge is both comprehensive and precise. Using the BM25 retriever with top-3 chunk selection, the system optimizes relevance and efficiency, minimizing redundancy while maximizing legally pertinent content. with top-3 chunk selection, the system optimizes relevance and efficiency, minimizing redundancy while maximizing legally pertinent content. A key design feature is the compression ratio constraint (0.05 to 0.5), maintaining structural symmetry between the original judgment and its summary by balancing representation and information density. Extensive evaluations establish BM25 as the most effective retriever, striking an optimal balance between precision and recall. A comparative analysis of transformer-based (Decoder-only) models—DeepSeek-7B, LLaMA 2-7B, and LLaMA 3.1-8B—demonstrates that LLaMA 3.1-8B, enriched with Legal Named Entity Recognition (NER) and the Dynamic RAG system, achieves superior performance with a BERTScore of 0.89. This study lays a strong foundation for future research in hybrid retrieval models, adaptive chunking strategies, and legal-specific evaluation metrics, with practical implications for case law analysis and automated legal drafting.

Keywords: legal text summarization; domain-specific adaptation; Retrieval-Augmented Generation; BM25 retriever; Decoder-only language models

Academic Editor: Jie Yang

Received: 12 March 2025
Revised: 15 April 2025
Accepted: 17 April 2025
Published: 23 April 2025

Citation: Ajay Mukund, S.; Easwarakumar, K.S. Optimizing Legal Text Summarization Through Dynamic Retrieval-Augmented Generation and Domain-Specific Adaptation. *Symmetry* **2025**, *17*, 633. https://doi.org/10.3390/sym17050633

1. Introduction

The proliferation of legal documents and judicial rulings has underscored the need for sophisticated systems capable of efficiently retrieving and summarizing relevant legal information. Legal texts are often characterized by their verbosity and intricate references to statutes, provisions, and precedents, making the task of distilling essential information both time-consuming and prone to factual inaccuracies. Traditional Natural Language Processing (NLP) models, despite their advances, struggle with the specificity and formalism of legal language, leading to challenges in maintaining factual consistency and contextual integrity.

This paper presents a novel framework that integrates a Dynamic Legal Retrieval-Augmented Generation (RAG) system with a Legal-Aware Text Summarization model. Our approach addresses the twin challenges of efficient legal information retrieval and hallucination-free summarization by dynamically incorporating contextual information in real time from domain-specific knowledge repositories. The RAG system is designed to retrieve the most pertinent legal context from resources such as the Constitution of India, Civil Procedure Code (CPC), and Supreme Court judgments using the BM25 [1] algorithm and a top-3 chunking strategy. This ensures that the retrieved context is semantically relevant and legally accurate, minimizing redundancy while maximizing legally pertinent content.

Standard Retrieval-Augmented Generation (RAG) approaches and non-domain-specific summarization models exhibit key limitations when applied to legal documents. Generic summarization models, including fine-tuned large language models (LLMs) trained on general-purpose corpora, often hallucinate legal content or omit crucial statutory references that were not part of their training distribution. Similarly, traditional RAG frameworks typically rely on surface-level semantic similarity for retrieval, which may not capture the nuanced relevance of statutes, procedural codes, or precedent cases required in the legal domain.

To address these challenges, we propose a Dynamic Legal RAG framework that performs real-time, entity-aware augmentation using outputs from a Legal Named Entity Recognition (NER) module. This framework identifies "dark zones" in legal judgments—such as unexplained statute–provision pairs or judicial precedents—and dynamically retrieves authoritative references to fill those gaps. By anchoring retrieval on structured legal entities rather than lexical similarity, the system enhances both factual consistency and contextual relevance in downstream summarization.

Building on this foundation, our legal text summarization model fine-tunes the LLaMA 3.1 8B architecture [2] to generate concise and contextually enriched summaries. The proposed system achieves a vital form of symmetry between information retrieval and content generation, ensuring that retrieved legal knowledge is both comprehensive and precise. A key design feature is the compression ratio constraint (0.05 to 0.5), maintaining structural symmetry between the original judgment and its summary by balancing representation and information density. By integrating the real-time context of the RAG system, our model minimizes hallucinations and enhances factual grounding, a critical requirement for legal text processing. Extensive evaluations establish BM25 as the most effective retriever, striking an optimal balance between precision and recall. The contextual flow facilitated by the RAG system ensures that the summarization model remains informed by the most up-to-date and relevant legal information, thereby improving both accuracy and interpretability. The key contributions of this paper are as follows:

1. The development of a **Dynamic Legal RAG system** tailored to the complexities of Indian legal texts.
2. The implementation of a **Factually-Aligned Summarization model** that leverages real-time retrieval of legal context.

Together, these components advance the state-of-the-art in legal information retrieval and summarization, providing a robust framework to support legal professionals in their analytical and decision-making processes. The remainder of this paper is organized as follows: Section 2 reviews related work on legal text summarization, and retrieval-augmented methods. Section 3 describes our methodology, including the design and implementation of the Dynamic RAG System and its integration with the Fine-Tuned Summarization Model. Section 4 outlines the experimental setup and dataset curation for RAG and for Summarization. Section 5 presents the results and performance analysis of each system

component. Finally, Section 6 discusses our findings and outlines potential directions for future research.

2. Related Work

Research in legal text processing has made notable progress, particularly in information retrieval (IR) and legal Text Summarization, driven by the increasing complexity and volume of judicial documents. Traditional IR methods such as Boolean search and TF-IDF [3], along with extractive summarization techniques, have laid the foundation for legal document analysis. However, the advent of transformer-based models and neural architectures has significantly enhanced the precision and contextual understanding required in legal applications. This section reviews key developments in legal IR and summarization, highlighting the transition from classical approaches to state-of-the-art techniques.

2.1. Information Retrieval for Legal Texts

Legal Information Retrieval (IR) is integral to legal research and facilitates access to judicial documents, case law, and statutory materials. Unlike general IR, legal IR requires high precision in retrieving relevant information from extensive legal texts while addressing their hierarchical and interconnected nature. The following subsections examine traditional and neural retrieval approaches, evaluating their efficacy and limitations in legal IR.

2.1.1. Traditional Retrieval Methods

Traditional legal IR methods, including Boolean search, TF-IDF, and BM25, have provided foundational mechanisms for the retrieval of legal documents and precedents. Boolean search employs logical operators to filter documents based on keyword expressions but is constrained by the complexity of the query [4]. Studies indicate that while the Boolean search remains relevant, it is increasingly outperformed by TF-IDF and BM25, which offer improved adaptability and retrieval efficiency [5].

TF-IDF, a statistical model that assesses the significance of a term within a corpus, has been widely applied in legal document ranking [6,7]. However, its effectiveness diminishes with long documents and term saturation. BM25, an extension of probabilistic retrieval models, mitigates these limitations through document length normalization and term frequency adjustments, leading to superior retrieval performance in case law and statutory retrieval [8,9]. Hybrid models that integrate TF-IDF and BM25 further enhance precision and recall, reinforcing their complementary strengths [10].

2.1.2. Neural Information Retrieval in Legal Domain

Neural Information Retrieval (Neural IR) has transformed legal document retrieval by employing deep learning models to capture semantic relationships beyond keyword matching. Transformer-based models [11,12] such as BERT, Legal-BERT, and CaseLaw-BERT have significantly improved contextual understanding in legal IR tasks [13]. Dense retrieval techniques like DPR enhance retrieval accuracy by leveraging domain-specific embeddings trained on legal corpora [14,15].

ColBERT [16] refines the retrieval by enabling contextual interactions at token level, improving fine-grained semantic matching in legal queries [17]. Furthermore, graph-based retrieval frameworks like SGPT [18] enhance retrieval precision by capturing structural dependencies within legal statutes and case law [19]. Hybrid models that integrate sparse and dense retrieval mechanisms, such as Retrieval-Augmented Generation (RAG) and GAR, further optimize retrieval efficacy, underscoring the growing significance of neural approaches in modern legal information systems [20].

Generic Retrieval-Augmented Generation (RAG) frameworks typically function by retrieving semantically similar document chunks based on the input query or paragraph,

which works well in open-domain question answering or search-based tasks. However, in the context of legal document summarization, this approach often fails to capture domain-specific dependencies. Legal documents frequently refer to dense statutory provisions, precedents, or procedural hierarchies, which may not be semantically similar to the surrounding text but are essential for accurate summarization.

2.2. Legal Text Summarization

Summarization techniques play a crucial role in condensing lengthy legal documents while preserving essential information. Extractive summarization, which selects key sentences from source documents, is widely used due to its ability to maintain the integrity of legal language. Abstractive summarization, on the other hand, generates novel text representations, posing additional challenges due to the need for factual accuracy and domain-specific coherence.

2.2.1. Extractive Summarization for Legal Documents

Extractive summarization is widely utilized in the legal domain due to its ability to retain crucial segments of legal texts while preserving their original structure. Given the verbosity and complexity of legal documents, extractive approaches provide an efficient means of distilling essential information, aiding legal practitioners in expediting case analysis. TextRank [21] and LexRank [22], two prominent unsupervised graph-based algorithms, are widely applied in legal text summarization. TextRank, inspired by PageRank, constructs a sentence graph and determines importance based on inter-sentence connectivity. Optimization strategies, including Bayesian parameter tuning, have improved its performance, achieving higher ROUGE scores [23]. Similarly, LexRank employs sentence similarity for ranking, demonstrating the efficacy in extracting legally salient information [24].

Supervised models, such as GIST [25], SummaRuNNer [26], and BERT-SUMM [27], leverage annotated legal datasets to extract informative sentences with higher precision. Domain-specific adaptations, such as Legal-BERT, enhance contextual understanding and semantic relevance, outperforming unsupervised methods [28]. Comparative studies highlight the advantages of supervised learning in capturing intricate legal nuances while ensuring contextual fidelity. Extractive summarization remains a cornerstone for legal document processing, with unsupervised models providing robust baselines and supervised approaches achieving greater accuracy and legal contextualization.

2.2.2. Abstractive Summarization in Legal NLP

Abstractive summarization has gained traction with the advent of transformer-based models, including BART [29], T5 [30], and PEGASUS [31]. These models generate concise summaries by paraphrasing source documents, offering greater flexibility compared to extractive methods. Domain-specific adaptations, such as Legal-PEGASUS and fine-tuned T5 models, have improved factual accuracy in legal summarization tasks. Recent studies have explored hybrid approaches, integrating extractive and abstractive models to optimize information quality while preserving legal validity. Transformer models have significantly advanced abstractive summarization in the legal domain. BART, a denoising autoencoder, effectively generates concise summaries but requires adaptation for long legal texts [32]. T5, a unified text-to-text transformer, benefits from domain-specific fine-tuning, while PEGASUS, pretrained with a summarization-focused objective, improves coherence. Hybrid models that integrate these architectures demonstrate superior accuracy and fluency of information [33].

Fine-tuning transformer models for legal texts enhances summary coherence and fidelity. Legal-PEGASUS and Legal-BERT, trained on extensive legal corpora, mitigate the challenges posed by verbosity and domain-specific terminology. The Big Bird architec-

ture [34], optimized for long documents, improves fluency and factual accuracy. Cross-domain analyses underscore the need for fine-tuning models to accommodate jurisdiction-specific variations in legal discourse [35]. The integration of transformer-based models and domain-specific adaptations has significantly improved abstractive summarization, facilitating efficient legal document analysis.

2.2.3. Decoder-Only Transformer Architecture for Abstractive Summarization in Legal NLP

Decoder-only architectures optimize summarization efficiency by eliminating the encoder component, reducing computational complexity while maintaining high-quality outputs. Mistral 7B and Zephyr 7B Beta employ advanced decoding techniques to enhance summary generation for lengthy legal texts [36]. The LLaMA series and DeepSeek 7B leverage pretrained checkpoints to improve contextual accuracy in legal summarization. Transfer learning facilitates domain adaptation, accelerates fine-tuning, and enhances summary coherence [37]. These architectures offer computational efficiency and scalability, enabling the processing of ultra-long legal documents. Fine-tuning further enhances their ability to capture critical case details and legal precedents [38].

2.2.4. Enhancing Summarization with Legal-Domain-Specific Knowledge

Named Entity Recognition [39] ensures the retention of essential legal elements by identifying key entities such as statutes, case numbers, and precedents. Tools like CaseSummarizer [40] integrate structured legal knowledge, refining the coherence and informativeness of generated summaries. Retrieval-Augmented Generation (RAG) enhances legal summarization by grounding outputs in legally relevant contexts. Methods such as DELSumm [41] incorporate precedent and statute references, improving the factual precision and legal validity. The integration of Legal NER and precedent-based retrieval increases summary quality, addressing the complexities inherent in legal text processing.

3. Methodology

The proposed approach for context-aware summarization of legal judgments integrates Dynamic Retrieval-Augmented Generation (RAG) for contextual knowledge retrieval and fine-tuned transformer models for summarization. The objective is to improve the relevance and factual accuracy of generated summaries by incorporating domain-specific named entities and retrieved legal knowledge.

3.1. Dynamic Legal RAG System

Legal document understanding necessitates a balance between precise retrieval of domain-specific knowledge and the generation of coherent, legally grounded responses. Conventional language models, despite their efficacy in natural language processing, often suffer from hallucination and lack domain-specific expertise when applied to legal texts. Retrieval-Augmented Generation (RAG) addresses these limitations by integrating a retrieval mechanism with a generative model, ensuring that generated responses are anchored in relevant legal precedents, statutes, and case laws.

In the legal domain, the dynamic nature of jurisprudence—where statutes evolve, precedents are established, and legal arguments are constructed based on prior rulings—necessitates an adaptive retrieval mechanism. Unlike static knowledge retrieval, a Dynamic Legal RAG System continuously refines its retrieval strategies by incorporating real-time legal updates, case-specific constraints, and domain-aware query expansion. This enhances both factual consistency and contextual relevance in generated legal responses.

The core principle of RAG involves retrieving the most pertinent legal documents from a structured knowledge base and leveraging them to condition the generation process. The retrieved documents provide contextual grounding, mitigating the risk of factual

inaccuracies and ensuring that the generated text aligns with established legal principles. Specifically, in the legal domain, retrieval spans various sources, including

- **Statutory Law**—Constitutional provisions, legislative acts, and codified regulations.
- **Case Law and Precedents**—Judicial rulings that establish authoritative interpretations of legal provisions.
- **Legal Commentaries and Doctrinal Writings**—Expert analyses that offer nuanced interpretations of legal principles.

The effectiveness of a Legal RAG system hinges on the retrieval strategy employed. Given that our approach utilizes Legal Named Entity Recognition (Legal NER) output, specifically provision–statute pairs and precedents, as input queries for the retrieval mechanism, a keyword-based search strategy is well suited for ensuring precision. Traditional lexical-based retrieval methods, such as BM25, are particularly effective in this context, as they rank documents based on term frequency and inverse document frequency while considering query-document relevance. As legal provisions and precedents are often explicitly referenced using standardized terminology, BM25 provides a highly efficient means of retrieving legally pertinent documents with minimal semantic drift. Additionally, by leveraging structured indexing of legal texts, the retrieval process can be further optimized to ensure that case-specific provisions and cited precedents are accurately retrieved.

By integrating BM25-based retrieval with generation, the Legal Dynamic RAG system provides a robust mechanism for generating legally sound, contextually accurate, and jurisdiction-specific responses. This approach is particularly advantageous for tasks such as case law summarization, legal question-answering, and automated brief drafting, where factual accuracy and direct reference to legal statutes and precedents are paramount. The subsequent subsections further delineate the components of this system, detailing the construction of a structured legal knowledge base, query formulation based on Legal NER outputs, and optimization techniques for improving keyword-based retrieval and generation efficiency.

3.1.1. Legal Corpus Construction and Indexing

A well-structured legal knowledge base is fundamental to the efficacy of a Legal Dynamic RAG system, ensuring comprehensive coverage of statutory provisions, judicial precedents, and legal terminology. To facilitate robust retrieval, our legal corpus consists of primary legal sources, including foundational statutes, procedural codes, judicial interpretations, and domain-specific references. Specifically, the corpus incorporates the following:

- Landmark Supreme Court Judgments (Volumes 1 and 2);
- Legal Maxims and Phrases (Volumes 1 and 2);
- The Constitution of India;
- Indian Penal Code (IPC);
- Criminal Procedure Code (CrPC);
- Civil Procedure Code (CPC);
- Indian Evidence Act;
- Legal Dictionary.

These documents collectively provide a comprehensive legal framework, ensuring that the retrieval system has access to authoritative legal knowledge across multiple domains.

Indexing Methodology for BM25-Based Retrieval

To enable efficient keyword-based retrieval, the legal corpus is preprocessed and indexed using BM25, a probabilistic ranking function widely used for information retrieval. The indexing pipeline consists of the following stages:

1. **Text Preprocessing**: Each document undergoes tokenization, stopword removal, and stemming, ensuring that common legal terms are retained while reducing redundancy.
2. **Segmentation and Structuring**: Given the hierarchical nature of legal texts, sections, provisions, and case citations are indexed separately, allowing for granular retrieval of specific legal clauses and precedents.
3. **BM25 Index Construction**: The retrieval model ranks documents using the BM25 ranking function, defined as follows:

$$BM25(D,Q) = \sum_{t \in Q} IDF(t) \cdot \frac{f(t,D) \cdot (k_1 + 1)}{f(t,D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{\text{avgDL}})} \quad (1)$$

where

- $BM25(D,Q)$ is the relevance score of document D for query Q.
- $f(t,D)$ is the term frequency of keyword t in document D.
- $|D|$ is the document length, and avgDL is the average document length in the corpus.
- k_1 and b are hyperparameters controlling term saturation and length normalization.
- $IDF(t)$ (inverse document frequency) is given by

$$IDF(t) = \log\left(\frac{N - n_t + 0.5}{n_t + 0.5} + 1\right) \quad (2)$$

where

- N is the total number of documents in the corpus.
- n_t is the number of documents containing term t.

4. **Entity-Aware Indexing**: To further enhance retrieval, Legal Named Entity Recognition (Legal NER) is employed to extract provision–statute pairs and precedents, which are indexed as queryable entities. These extracted entities are assigned higher weights within the BM25 ranking, ensuring that statutory provisions and case law references are prioritized during retrieval.

The factor 0.5 used in the IDF Formula (2) serves two key purposes:

- It prevents division by zero when $n_t = 0$.
- It acts as a Bayesian smoothing factor to moderate the influence of rare or overly frequent terms. This smoothing helps prevent extreme values in the score when terms appear in either all or very few documents in the corpus.

Although we follow the canonical BM25 implementation that uses 0.5, we acknowledge that this factor can be tuned based on corpus characteristics. For instance, setting the factor to 1.0 could offer stronger smoothing, which might be useful in noisy or highly redundant document collections. Conversely, reducing the factor below 0.5 could enhance sensitivity to rare terms, which may be desirable when legal arguments hinge on specific uncommon statutes or precedents. In our current experiments, we retain the standard value (0.5) for consistency and reproducibility, but we highlight this as a potential hyperparameter for future domain-specific optimization.

Impact of the Indexing Strategy

This indexing methodology enables high-precision legal retrieval, ensuring that query responses maintain both factual accuracy and jurisdictional relevance. By structuring the knowledge base in an efficiently indexed format, the system enhances retrieval performance while maintaining scalability for expanding legal datasets.

3.1.2. Contextualized Legal Document Retrieval

Legal document retrieval poses unique challenges due to the intricate interdependencies between statutes, provisions, and case law. Unlike general-purpose retrieval systems, legal information retrieval must account for the hierarchical structure of legal texts, domain-specific terminologies, and precedential influences. To address these complexities, we employ a Contextualized Legal Document Retrieval framework within our Dynamic Retrieval-Augmented Generation (RAG) system, ensuring that retrieved documents provide relevant legal context for the summarization model.

Our retrieval system is dynamically triggered when the summarization model encounters a provision–statute pair or a precedent, identified through Legal Named Entity Recognition (Legal NER). These entities serve as query anchors, guiding the retrieval process to fetch relevant statutory clauses, case laws, and judicial interpretations from the curated Legal Knowledge Base.

Retrieval Mechanism for Provision–Statute Pairs

Statutory provisions often reference other legal sections and judicial interpretations, necessitating an expansive retrieval strategy. When a provision–statute pair is detected, the retrieval system constructs a structured query:

$$Q = \text{Provision} + \text{Statute} + \sum_{i=1}^{n} R_i \tag{3}$$

where:

- Provision represents the specific clause or section of a legal act (e.g., *"Section 304B"*).
- Statute denotes the governing legal framework (e.g., *"Indian Penal Code"*).
- R_i represents related legal provisions, inferred using cross-references from the legal knowledge base.

The BM25 retriever is then applied to fetch semantically and contextually relevant legal texts, ensuring that the retrieved documents contain provisions, amendments, or precedential rulings relevant to the detected entity.

Retrieval Mechanism for Precedents

Legal precedents play a critical role in jurisprudence, influencing judicial reasoning across cases. When the summarization model identifies a precedent, the retrieval process is adapted to extract judicial interpretations and case law references. The structured query formulation is given by

$$Q = \text{Case Citation} + \sum_{i=1}^{n} J_i \tag{4}$$

where

- Case Citation uniquely identifies the legal precedent (e.g., *"Kesavananda Bharati v. State of Kerala (1973)"*).
- J_i represents judicial references, including *ratio decidendi*, *obiter dicta*, and citations appearing in related judgments.

The BM25 scoring mechanism, as previously formulated, ensures that case law retrieval is ranked based on contextual relevance, emphasizing key terms, statutory references, and precedential weight.

Dynamic Integration into Summarization

Once the relevant legal texts are retrieved, they are integrated into the summarization model in real time. The process follows a dynamic legal context injection strategy:

1. **Provision–Statute Pair Integration**: Retrieved statutory texts supplement the summarization pipeline, ensuring that legal provisions are accurately represented within the generated summaries.
2. **Precedent-Aware Summarization**: Case law retrieval provides precedential context, reinforcing the legal argumentation presented in the summarized text.

By leveraging contextualized legal retrieval, our framework enhances factual accuracy, reduces hallucination, and ensures jurisprudential coherence in automated legal text generation.

3.1.3. Role of Legal Named Entity Recognition (Legal NER)

Legal Named Entity Recognition (NER) plays a pivotal role in our retrieval-augmented summarization framework. We use a fine-tuned RoBERTa-based model [39] trained on legal corpora to extract structured entities such as *Statutes*, *Provisions*, *Case Numbers*, *Precedents*, *Judges*, *Lawyers*, and *Organizations*.

These extracted entities serve two key functions:

1. **Entity-Aware Query Formulation:** Extracted legal entities are used to dynamically generate focused search queries that guide the BM25 retriever. For example, if a provision such as "Section 197 CrPC" or a precedent like "Maneka Gandhi v. Union of India" is identified in the judgment, it is used to retrieve authoritative references and explanations from the legal corpus.
2. **Contextual Enrichment of Summarization:** The retrieved legal information is integrated into the input passed to the summarization model. This ensures that the model has access to definitions, related precedents, or statutory explanations that may not have been seen during training.

By incorporating Legal NER, the framework is able to mitigate hallucination risks, enhance factual grounding, and improve legal interpretability in the generated summaries. Furthermore, the NER-based filtering mechanism enables the system to prioritize legally significant content while discarding generic or irrelevant context.

3.1.4. Dynamic Retrieval Design: Thresholds, Chunk Size, and Entity-Aware Queries

The threshold of 0.75 was empirically determined during system validation, balancing recall and precision for legal entity-based queries. Lower thresholds introduced redundant or loosely related content, while higher thresholds often excluded relevant statutory or case-specific context. A similarity threshold of 0.75 offered the optimal trade-off across evaluation metrics such as ROUGE-L, cosine similarity, and legal coverage.

Chunks are extracted using a fixed size of 256 tokens with a 25-token overlap. This configuration was empirically selected to balance context preservation with computational efficiency. The chunk size ensures that legal clauses or arguments are not fragmented across retrieval windows, while the slight overlap helps maintain continuity of meaning between adjacent chunks. Other configurations (e.g., 128, 200, 300 tokens) were tested, but the 256-token setting offered the best retrieval accuracy with minimal redundancy in dynamic integration.

The "dynamic" nature of our Legal RAG system refers to its real-time, entity-aware retrieval mechanism. Instead of issuing generic queries based on input text, the system dynamically extracts legal entities (statutes, precedents, case numbers) using Legal NER, and triggers retrieval based specifically on those entities. This allows the model to target and integrate only the most relevant legal context into the summarization pipeline.

3.1.5. Dynamic Legal RAG System Workflow

The workflow of the Dynamic Legal RAG (Retrieval-Augmented Generation) System begins with the processing of an input legal judgment through the fine-tuned RoBERTa model [39]. The extracted legal entities undergo a rigorous Entity Post-Processing phase, which includes Precedent Coreference Resolution, Other Person Coreference Resolution, Provision–Statute Pair Coreference Resolution, and Entity Consolidation and Overlap Resolution. The complete system workflow is visualized in Figure 1.

Once the entities are refined, the Dynamic Legal RAG System is engaged to provide enriched contextual knowledge for legal summarization. To establish this system, a meticulously curated Legal Corpus serves as the knowledge base, comprising foundational legal documents, including, but not limited to, the Constitution of India, Indian Penal Code (IPC), Criminal Procedure Code (CrPC), Civil Procedure Code (CPC), landmark Supreme Court judgments, legal maxims, and dictionaries. These sources are systematically indexed and annotated to facilitate efficient retrieval.

The indexing methodology and the retrieval mechanisms are discussed in Section 3.1.1 and Section 3.1.2, respectively. Using this structured retrieval methodology, the Dynamic Legal RAG system fetches three highly relevant document chunks with a similarity threshold of approximately 0.75. These retrieved segments are dynamically integrated with the input legal judgment before passing into the summarization model.

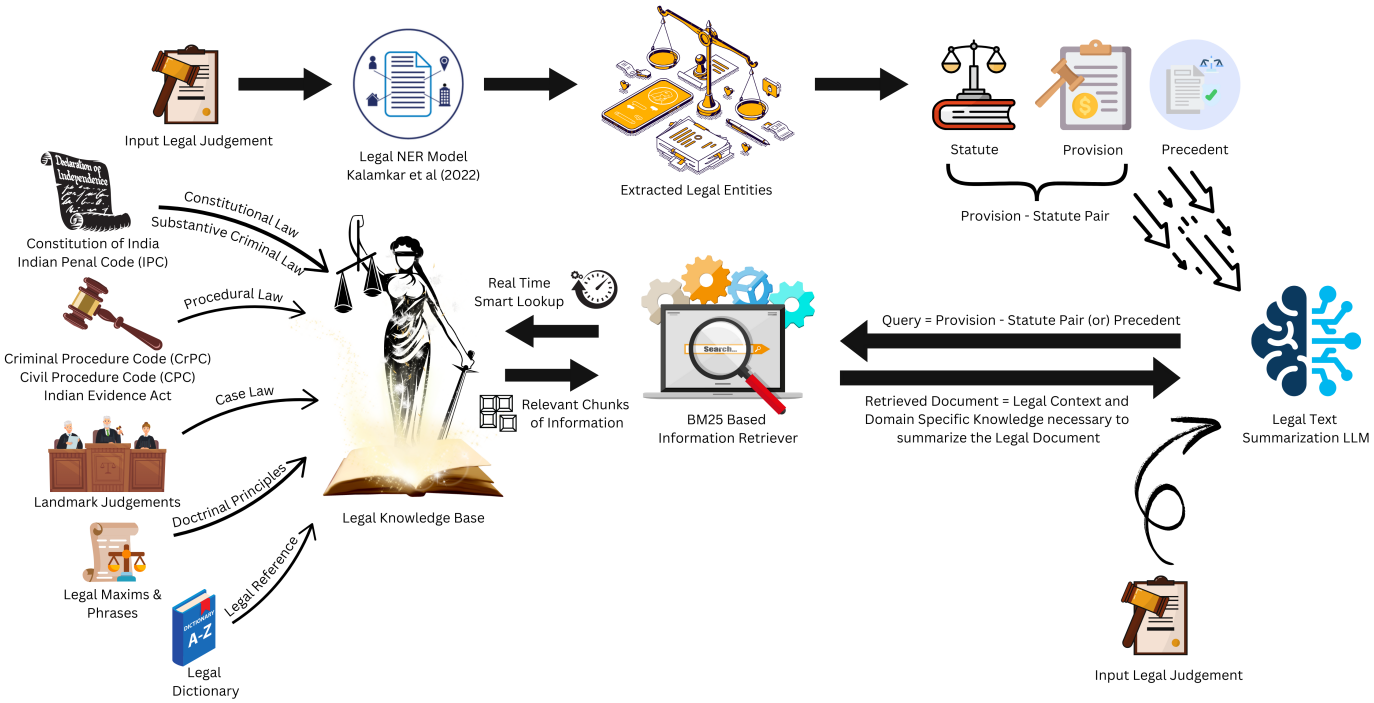


Figure 1. Dynamic Legal RAG workflow visualization [39].

The proposed Legal RAG system is dynamic in nature due to its ability to adapt retrieval and knowledge augmentation based on entity types and contextual dependencies. Unlike conventional RAG frameworks [42–44], which typically retrieve semantically similar chunks from a static corpus, our system addresses specific gaps in legal judgments, such as provision–statute pairs, precedents, or domain-specific references that may not be part of the summarization model's training distribution.

This dynamicity is achieved through real-time, entity-aware retrieval, where extracted legal entities act as anchors for generating precise queries over the legal corpus. The system reformulates these queries intelligently, prioritizing statutory and precedent-based infor-

mation relevant to the judgment at hand. The adaptive weighting of extracted entities in the BM25 ranking further enhances the prioritization of authoritative references, mitigating hallucination risks and reinforcing the legal credibility of generated summaries.

By augmenting factual gaps in real time, the system ensures that the generated content remains contextually enriched and aligned with jurisprudential reasoning. While Retrieval-Augmented Generation is not new in itself, this domain-specific adaptation—where dynamic entity-driven augmentation is paired with fine-tuned legal summarization—offers a meaningful advancement in the application of RAG for legal text processing.

3.2. Legal Text Summarization

Legal text summarization is a crucial task aimed at generating factually grounded and concise summaries of judicial decisions while minimizing hallucination. Unlike conventional summarization tasks, legal summarization necessitates the preservation of domain-specific terminology, complex legal reasoning, and statutory references. To achieve this, we adopt a structured multi-source input paradigm that integrates (i) the full-text judgment, (ii) extracted legal entities from our Legal Named Entity Recognition (Legal NER) model, and (iii) contextual augmentation from our Dynamic Legal RAG System, which dynamically retrieves relevant text chunks corresponding to legal precedents and provision–statute pairs. By leveraging these three components, we ensure that the generated summary maintains both legal coherence and contextual enrichment.

For this task, we systematically explore and fine-tune five state-of-the-art decoder-only architectures: Mistral-7B, Zephyr-7B Beta, DeepSeek-7B Base, LLaMA 2-7B Chat, and LLaMA 3.1-8B Instruct. These models were selected based on their strong performance in open-ended text generation, their scalability for long-context summarization, and their robust transformer-based architectures. Table 1 presents a comparative overview of their key architectural configurations, including decoder layers, attention heads, embedding size, MLP dimension, and key/value dimensions. Notably, all models maintain 32 attention heads for efficient self-attention mechanisms, while variations in embedding size and MLP dimensions reflect differences in computational efficiency and model expressiveness. In the subsequent subsubsections, we provide a detailed analysis of the architectural configurations and fine-tuning methodologies applied to each of the selected models, evaluating their efficacy in generating legally sound and factually reliable summaries.

Table 1. Architectural comparison of the fine-tuned models.

Model Category	Model Variant	Decoder Layers	Attention Heads	Embedding Size	MLP Dimension	Key/Value Dimension
LLaMA	LLaMA 2 7B	32	32	32,000 × 4096	14,336	1024
	LLaMA 3.1 8B	32	32	128,256 × 4096	14,336	1024
	DeepSeek 7B	30	30	100,015 × 4096	11,008	4096
Mistral	Mistral 7B	32	32	32,000 × 4096	14,336	1024
	Zephyr 7B	32	32	Pad'ng idx = 2	14,336	1024

3.2.1. Selected Decoder-Only Model Architectures

Decoder-only transformer models have proven highly effective for generative tasks such as legal summarization due to their autoregressive design, unidirectional attention, and efficient token-by-token decoding. This section provides a concise comparative overview of the selected models used in our study.

As seen in Table 1, most models share similar transformer specifications (32 decoder layers, 32 attention heads), with differences mainly in vocabulary size and embedding initialization. LLaMA 3.1 and DeepSeek exhibit expanded embedding matrices and higher

MLP dimensions to support extended context windows and cross-lingual capabilities. Zephyr represents a distilled, instruction-aligned variant of Mistral.

Model Selection and Justification of Decoder-Only Architecture

We selected multiple decoder-only transformer models—LLaMA 2 [45], LLaMA 3.1 [2], DeepSeek [46], Mistral [47], and Zephyr [48] —based on a combination of performance, open-weight availability, and architectural suitability for abstractive summarization. Decoder-only models have been shown to be highly effective for generative tasks like summarization due to their unidirectional attention and token-by-token prediction ability. These models are pretrained on large-scale general corpora and exhibit strong zero-shot and fine-tuning capabilities. Our inclusion of LLaMA 3.1 8B and Mistral was motivated by their proven performance across instruction-following tasks and recent competitive benchmarks. DeepSeek, being an open-source alternative with bilingual pretraining (English-Chinese), offered interesting insights into generalizability. Zephyr was included to observe performance on a distilled instruction-tuned lightweight variant.

We chose decoder-only architectures instead of encoder–decoder models (e.g., BART, PEGASUS) because our pipeline involves RAG-based contextual augmentation, where the summarizer receives a long, integrated input containing both the judgment and retrieved legal content. Decoder-only models handle such long generative sequences more naturally, without needing separate encoding of context and summary. Moreover, they exhibit lower latency at inference and scale better with LoRA-based fine-tuning. Thus, our choice reflects a balance of architectural strength, domain adaptability, and integration efficiency within the legal summarization framework.

3.2.2. Fine-Tuning Process of Large Language Models

The fine-tuning process for Large Language Models (LLMs) follows a structured and methodical approach to adapt pretrained models for the task of legal text summarization. The models considered in this study include Mistral-7B, Zephyr-7B Beta, DeepSeek-7B Base, Llama-2-7B, and Llama-3.1-8B Instruct. Each of these models was subjected to an identical fine-tuning pipeline, ensuring a controlled evaluation of performance enhancements through domain-specific adaptation.

Dataset Preparation and Preprocessing

The dataset employed for fine-tuning consists of full-text legal judgments along with their corresponding abstractive summaries. These texts were preprocessed into a structured conversation format, wherein the system prompt explicitly instructs the model to generate precise, well-structured summaries without introducing hallucinations. The conversation format aligns with the instruction-tuned paradigm, where the input consists of the judgment, and the expected output is a logically sequenced summary that maintains chronological coherence.

To ensure compatibility with different models, tokenization was performed using the respective tokenizers of each LLM, with truncation applied at an optimal sequence length. The dataset was then converted into a DatasetDict containing separate training and test splits, which facilitated structured training and evaluation.

Model Selection and Quantization

The fine-tuning process was performed on GPUs, with CUDA acceleration enabled where available. Model selection was based on their foundational capabilities in handling legal text, with configurations tailored accordingly. Given the substantial computational requirements, quantization techniques were employed to optimize memory usage. Specifically, 4-bit Normal Float (NF4) quantization [49] was applied, leveraging BitsAndBytes

(bnb) configurations. This optimization not only reduced memory overhead but also ensured computational efficiency without significant degradation in performance.

Optimization Strategies and Hyperparameter Tuning

Fine-tuning was conducted using the AdamW optimizer with a cosine learning rate scheduler. The learning rate was set to $\eta = 10^{-5}$ with a warm-up phase to stabilize training dynamics. Gradient accumulation and gradient checkpointing were enabled to handle large batch computations while maintaining computational feasibility. The training regime incorporated mixed-precision floating-point operations (bf16) to enhance throughput on compatible hardware.

Parameter-Efficient Fine-Tuning (PEFT) [50] with LoRA

To fine-tune these large models efficiently, we employed Low-Rank Adaptation (LoRA) [51], which injects lightweight trainable matrices into specific transformer layers. Combined with gradient checkpointing, LoRA allows task adaptation with significantly fewer trainable parameters.

Table 2 highlights the progression of trainable parameters before and after applying gradient checkpointing and Low-Rank Adaptation (LoRA). It also shows LoRA's effectiveness in significantly reducing trainable parameters while preserving model expressivity. LLaMA 3.1 and DeepSeek required more adaptation capacity due to their larger embedding matrices and multilingual pretraining.

Table 2. Comparison of model parameters at different stages for various models.

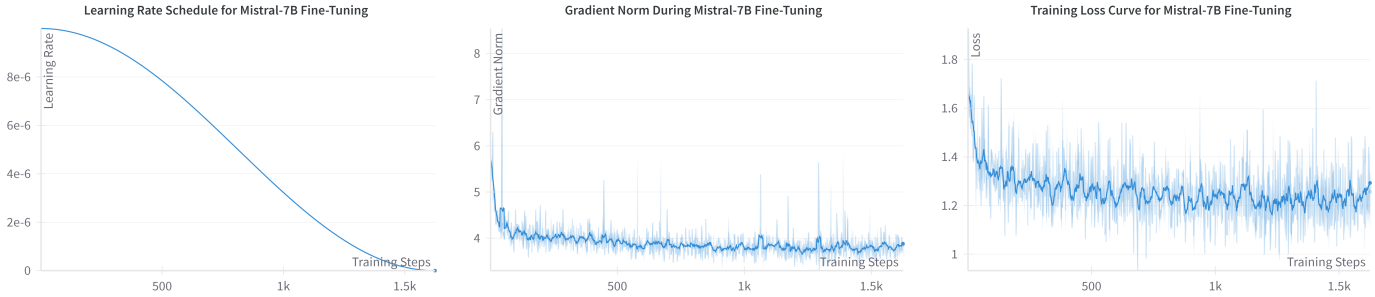
Model	Checkpointing Stage	Trainable Parameters	Total Parameters	Trainable %
Mistral-7B & Zephyr-7B	Before Gradient Checkpointing	0	3,752,071,168	0.00%
	After Gradient Checkpointing	262,410,240	3,752,071,168	6.99%
	After Applying LoRA	346,030,080	4,098,101,248	8.44%
DeepSeek-7B	Before Gradient Checkpointing	0	3,855,200,256	0.00%
	After Gradient Checkpointing	819,572,736	3,855,200,256	21.26%
	After Applying LoRA	894,279,680	4,749,479,936	18.83%
Llama-2-7B	Before Gradient Checkpointing	0	3,500,412,928	0.00%
	After Gradient Checkpointing	262,410,240	3,500,412,928	7.50%
	After Applying LoRA	342,097,920	3,842,510,848	8.90%
Llama-3.1-8B-Instruct	Before Gradient Checkpointing	0	4,540,600,320	0.00%
	After Gradient Checkpointing	1,050,939,392	4,540,600,320	23.15%
	After Applying LoRA	1,134,559,232	5,675,159,552	19.99%

Summary of Fine-Tuning Outcomes

Figures 2–6 provide additional insights into the fine-tuning dynamics by illustrating the learning rate schedules, gradient norm variations, and training loss curves for each model. The learning rate schedules exhibit a smooth cosine decay, ensuring stable convergence while preventing abrupt shifts in weight updates. The gradient norm plots reveal the stability of updates across iterations, with minor fluctuations indicative of efficient optimization. The training loss curves demonstrate consistent declines, affirming the effectiveness of the parameter-efficient fine-tuning strategy. Collectively, these visualizations validate the controlled progression of fine-tuning and underscore the impact of adaptive techniques in optimizing model performance.

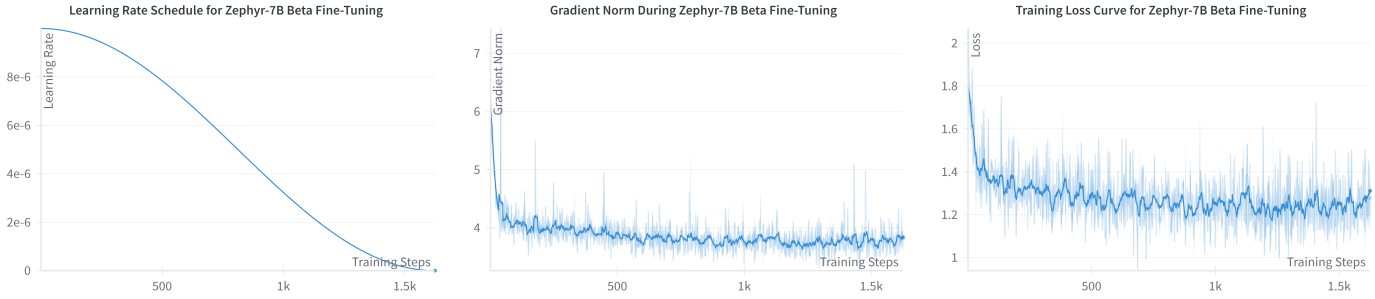
The fine-tuning framework established in this study provided a systematic approach to optimizing LLMs for legal summarization. The primary findings underscore the importance of structured dataset preprocessing, quantization for efficiency, PEFT techniques

for scalability, and robust hyperparameter selection for stable convergence. By ensuring model-specific compatibility while maintaining a uniform training protocol, the adaptation of these models successfully enhanced their ability to generate concise, logically ordered legal summaries.



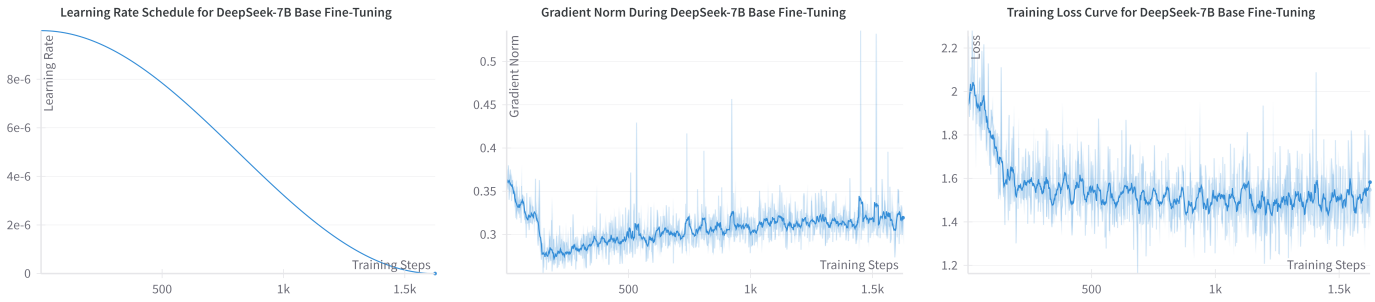
(**a**) (**b**) (**c**)

Figure 2. Fine-tuning performance metrics for Mistral-7B. This figure showcases the (**a**) learning rate dynamics, (**b**) gradient norm fluctuations, and (**c**) training loss progression.



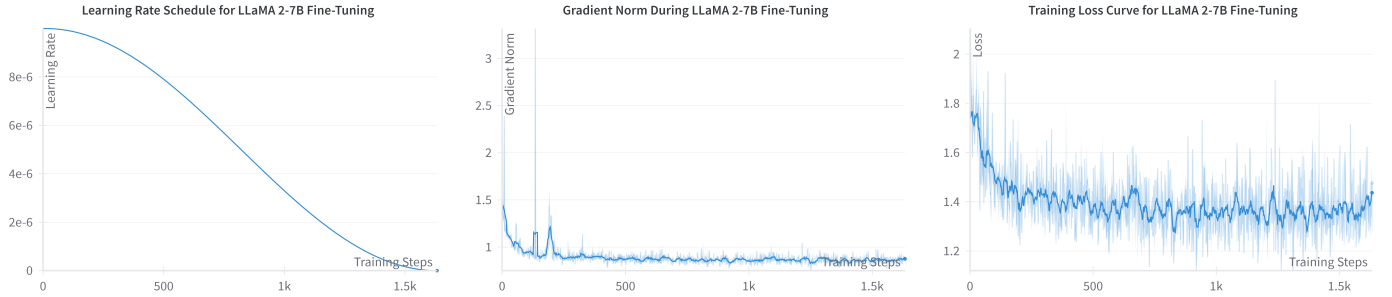
(**a**) (**b**) (**c**)

Figure 3. Fine-tuning performance metrics for Zephyr-7B Beta. This figure showcases the (**a**) learning rate dynamics, (**b**) gradient norm fluctuations, and (**c**) training loss progression.



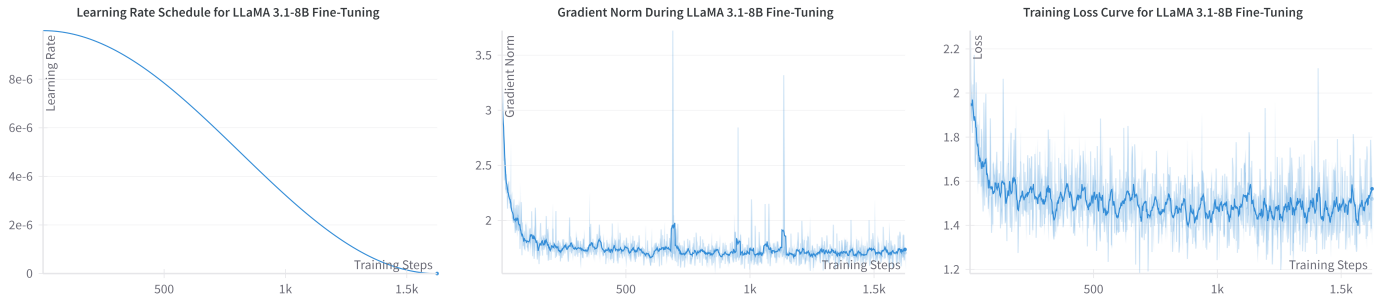
(**a**) (**b**) (**c**)

Figure 4. Fine-tuning performance metrics for DeepSeek-7B. This figure showcases the (**a**) learning rate dynamics, (**b**) gradient norm fluctuations, and (**c**) training loss progression.



(**a**) (**b**) (**c**)

Figure 5. Fine-tuning performance metrics for LLaMA 2-7B. This figure showcases the (**a**) learning rate dynamics, (**b**) gradient norm fluctuations, and (**c**) training loss progression.



(**a**) (**b**) (**c**)

Figure 6. Fine-tuning performance metrics for LaMA 3.1-8B. This figure showcases the (**a**) learning rate dynamics, (**b**) gradient norm fluctuations, and (**c**) training loss progression.

4. Experimental Procedure

This section delineates the experimental framework employed for the development, evaluation, and integration of the Legal Retrieval-Augmented Generation (Legal RAG) and legal text summarization components within our proposed system. The experimental setup is designed to ensure a systematic and reproducible workflow, facilitating the assessment of model performance across distinct yet interrelated tasks in legal document processing. Section 4.1 details the Environmental Setup, specifying the computational infrastructure, software dependencies, and model execution environment to provide a standardized foundation for experimentation. Section 4.2 elaborates on Data Preparation and Preprocessing, outlining the dataset curation strategies, annotation methodologies, and preprocessing techniques tailored for the Legal RAG and legal text summarization tasks. This ensures consistency in data handling and optimizes model performance. By establishing a rigorous experimental framework, this section lays the groundwork for evaluating the efficacy and robustness of our proposed methodology.

4.1. Environmental Setup

The effectiveness of deep learning models in legal text processing is significantly influenced by the computational environment in which they are deployed. Given the intricate nature of legal documents, which often exhibit high linguistic complexity and domain-specific terminology, a well-optimized computational setup is essential to ensure efficient training, inference, and evaluation of machine learning models. This section provides a detailed overview of the computational infrastructure utilized for the implementation of our legal text processing tasks, specifically Legal Retrieval-Augmented Generation (Legal RAG) and legal text summarization.

Environmental Setup for Dynamic Legal RAG System and Legal Text Summarization

For the implementation of Legal Retrieval-Augmented Generation (Legal RAG) and legal text summarization, a high-performance computational environment was established to ensure efficient model execution, retrieval augmentation, and text summarization processes. The experimental setup was designed to handle the complexity of large-scale legal text processing, leveraging a powerful GPU-accelerated framework.

Initial development was carried out on a local workstation equipped with an NVIDIA Quadro P5000 GPU (16 GB VRAM). However, due to significant memory and latency constraints when fine-tuning large-scale decoder-only transformer models, this setup was found to be suboptimal. As a result, we transitioned to a subscription-based cloud compute environment providing access to an NVIDIA A100 GPU (40 GB VRAM, Ampere architecture), which offered the necessary computational throughput and memory capacity for efficient training and inference.

The A100 GPU environment, running on CUDA 12.2, delivered high-throughput parallel computing capabilities essential for deep learning and transformer-based models. The Intel Xeon CPU (2.20 GHz, 6 physical cores, 12 logical threads) facilitated efficient multi-threaded operations, ensuring smooth preprocessing and model orchestration. The system was equipped with 83.5 GB of total memory (79 GB available), enabling the execution of memory-intensive tasks such as document retrieval and RAG-based summarization. Disk space configuration included 113 GB of total storage, with 81 GB available for dataset handling and model checkpoint storage.

The software stack was structured using Python 3.10.12 and pip-managed virtual environments. Key libraries included huggingface-hub (v0.26.5) and sentence-transformers (v3.2.1) for retrieval and embedding tasks, langchain (v0.3.11) for efficient RAG workflows, and spaCy (v3.7.5) along with pandas (v2.2.2) for preprocessing and data handling. Visualization and experimentation tools included matplotlib (v3.8.0) and Weights & Biases (wandb v0.18.7). Deep learning operations were powered by TensorFlow (v2.17.1) with CUDA-optimized CuDNN (v9.6.0.74) acceleration.

This upgraded computational infrastructure ensured reliable scaling of the Legal RAG framework, facilitating faster training, better memory utilization, and high-accuracy processing across legal summarization workflows.

4.2. Dataset Preparation

In the context of developing robust and high-performing models for legal Natural Language Processing (NLP) tasks, the quality and comprehensiveness of the dataset play a pivotal role. Given the intricacies and domain-specific nature of legal texts, it is imperative to curate datasets that accurately capture the nuances of legal language and information embedded in judicial documents. In this section, we provide a detailed account of the datasets employed for the two key components present in our proposed framework. Each dataset was carefully selected and preprocessed to align with the objectives of the respective tasks, ensuring that the models trained on these resources are equipped to handle the complexity and specificity of legal discourse. Furthermore, the datasets encompass a diverse range of judicial documents, including court judgments, preambles, and human-authored headnotes, thus offering a holistic representation of legal texts essential for comprehensive model development and evaluation.

4.2.1. Dataset for Legal Retrieval-Augmented Generation (Legal RAG)

The dataset curated for this study comprises authoritative legal documents essential for jurisprudential research, statutory interpretation, and legal precedent analysis.

These documents were sourced from India Code (https://www.indiacode.nic.in, accessed on 23 November 2024)—Digital Repository of Laws and Legal Information Institute of India (http://www.liiofindia.org, accessed on 16 June 2024) (LIIofIndia), ensuring that the corpus remains jurisdictionally relevant and doctrinally robust. The dataset was preprocessed into 1877 textual chunks, thereby enhancing its indexability and retrieval efficiency within the RAG framework.

A structured overview of the dataset and its legal significance is provided in Table 3, detailing the relevance of each document within the Indian legal framework, along with its role in case law interpretation, statutory exegesis, and legal research. As RAG systems operate dynamically, newer legal documents can be incrementally integrated into the vector database, ensuring that the model remains adaptable to evolving judicial interpretations and legislative amendments. This continual knowledge augmentation enhances the utility of the system for real-time legal research, automated case law retrieval, and statutory analysis, making it an indispensable tool in legal text summarization.

Table 3. Overview of the Legal Corpus and its role in Retrieval-Augmented Generation (RAG).

Legal Document	Type of Legal Framework	Legal Significance	Application in RAG
Landmark Judgments	Case Law	Establishes binding precedents Serves as a primary source for case law analysis	Supports retrieval of judicial interpretations Aids in legal reasoning
Legal Maxims and Phrases	Doctrinal Principles	Provides interpretative principles Standardizes legal terminology	Enhances contextual understanding Improves semantic enrichment
Constitution of India	Constitutional Law	Supreme legal document defining rights Governs fundamental governance structure	Facilitates constitutional law queries Supports jurisprudential research
Indian Penal Code (IPC)	Substantive Criminal Law	Defines criminal offenses Prescribes penalties for crimes	Supports criminal case law retrieval Enhances legal document classification
Indian Evidence Act	Procedural Law	Governs admissibility of evidence Establishes legal standards for proof	Enables retrieval of evidentiary rules Assists in legal argumentation
Criminal Procedure Code (CrPC)	Procedural Law	Regulates criminal trials and investigations Ensures due process in criminal justice	Provides procedural case insights Supports RAG-based legal queries
Civil Procedure Code (CPC)	Procedural Law	Governs civil litigation procedures Regulates dispute resolution mechanisms	Identifies procedural precedents Enhances case law retrieval
Legal Dictionary	Legal Reference	Defines legal terminology Standardizes legal definitions	Ensures precision in information retrieval Supports document annotation

4.2.2. Dataset for Legal Text Summarization

The dataset employed for the legal text summarization task is the IN-Abs dataset, originally introduced by Shukla et al. [28]. It comprises 7130 judgment–summary pairs, featuring judgments delivered by the Supreme Court of India. The dataset was primarily

sourced from the Legal Information Institute of India (LIIofIndia) (http://www.liiofindia.org/in/cases/cen/INSC/ (accessed on 16 April 2025)), encompassing judgments from 1950 to 1993.

Each summary in the dataset is a human-authored headnote, encapsulating key legal principles, precedents, and critical issues discussed in the case, making them an ideal gold-standard reference for fine-tuning summarization models. On average, the judgments contain 4373 tokens, while the summaries average 841 tokens.

The dataset is publicly available at https://zenodo.org/record/7152317 (accessed on 13 April 2024) and includes the following directory structure:

- `train-data/`—Contains 7030 judgment–summary pairs used for training.
  - `judgement/`—Raw full-text judgments.
  - `summary/`—Corresponding abstractive summaries.
  - `stats-IN-train.txt`—Word and sentence count statistics.
- `test-data/`—Contains 100 judgment–summary pairs for evaluation.
  - `judgement/`, `summary/`, `stats-IN-test.txt`.

This dataset provides a legally rich, high-quality training and evaluation corpus. While it spans a historical period, its linguistic consistency and expert annotations make it highly suitable for training legal summarization models. However, the dataset exhibits certain vulnerabilities and inconsistencies, necessitating rigorous preprocessing and filtering to ensure optimal quality for model training.

Temporal Validity of Legal Language and Dataset Justification

When discussing the temporal limitation of the IN-Abs dataset with legal professionals, it was noted that while Indian law has undergone significant amendments over time, the core structure and phrasing of legal language has remained largely consistent from the 1950s through to the 2020s. Legal judgments are still written in a formal, precedent-driven manner with little stylistic deviation, making the IN-Abs dataset a valid training corpus for learning summarization patterns.

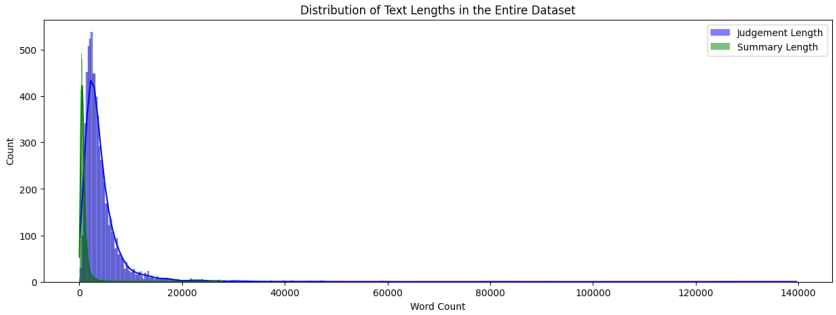
Furthermore, our framework compensates for any factual or statutory drift by integrating a Dynamic Retrieval-Augmented Generation (RAG) pipeline. This component ensures that up-to-date statutes, amendments, and precedents are retrieved in real time, while the language model focuses solely on generating contextually faithful summaries. Thus, even if laws evolve, the system maintains alignment with current legal standards without requiring complete retraining on recent data.

Statistical Analysis and Filtering

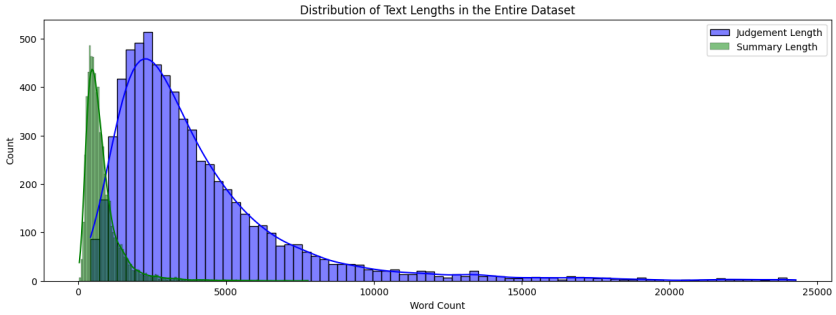
A preliminary statistical analysis of the dataset reveals significant skewness and outliers in the distribution of judgment lengths, as illustrated in Figure 7a. Upon closer examination, the following anomalies were identified:

1. **Outliers in Judgment Lengths**: Approximately 7% of the dataset (499 judgments) comprises extremely long texts, with the 99th percentile averaging 38,858 tokens. To maintain consistency, these are removed.
2. **Very Short Judgments**: Judgments in the 0.5th percentile (mean $\tilde{3}12$ tokens) are deemed too short for a meaningful summarization and are excluded.

By restricting the dataset within the 0.5th to 99th percentiles based on the lengths of the judgment, we effectively mitigate skewness and reduce variance. Figure 7b presents the improved distribution after applying this preprocessing step.



(**a**)



(**b**)

Figure 7. Comparison of judgment and summary length distributions before and after filtering. (**a**) Original distribution of text lengths, highlighting skewness and extreme outliers. (**b**) Refined distribution after removing extremely long and very short judgments, resulting in a more balanced dataset.

Compression Ratio Filtering

The **compression ratio (CR)** is a measure of how much a judgment text is condensed into its corresponding summary. It is defined as the ratio of the total number of tokens in the summary to the total number of tokens in the original judgment. Mathematically, it can be expressed as follows:

$$CR = \frac{\sum_{i=1}^{N_{\text{summary}}} t_i}{\sum_{j=1}^{N_{\text{judgment}}} T_j} \tag{5}$$

where

- t_i represents the token count of the i-th token in the summary.
- T_j represents the token count of the j-th token in the judgment.
- N_{summary} and N_{judgment} denote the total number of tokens in the summary and judgment, respectively.

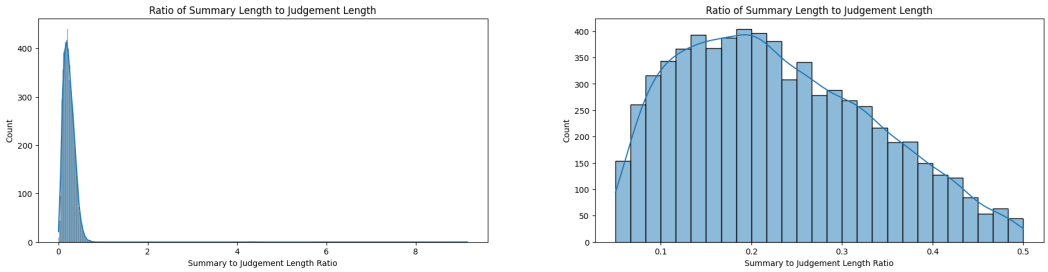
A detailed inspection, illustrated in Figure 8a, highlights specific anomalies in the dataset:

1. **Compression ratios > 1** indicate that summaries are longer than judgments, contradicting the fundamental principle of summarization.
2. **Compression ratios = 0** indicate that the summary is identical to the judgment, providing no meaningful compression.

To ensure that the dataset maintains an appropriate level of summarization without excessive compression or verbosity, we filter samples based on the following constraint:

$$0.05 \leq CR \leq 0.5 \tag{6}$$

This step ensures the exclusion of overly compressed or excessively verbose summaries, thereby improving dataset quality. The compression ratio (CR) constraint also enforces a vital form of structural symmetry between the original legal judgment and its summary. By maintaining a balanced representation and information density, the summaries preserve essential legal context while avoiding disproportionate detail reduction or expansion. This structural symmetry ensures that the summarized content remains proportionate, informative, and aligned with the source material's intent and significance. The effectiveness of this filtering is depicted in Figure 8b.



(**a**) (**b**)

Figure 8. Comparison of compression ratio distributions before and after filtering. (**a**) Initial distribution of compression ratios, highlighting anomalies such as overly verbose summaries (CR > 1) and redundant summaries (CR = 0). (**b**) Refined distribution after filtering out extreme cases, ensuring a more balanced and meaningful summarization ratio.

Upon execution of Algorithm 1, the dataset reduces from 7130 to 6757 pairs, reflecting a data loss of 5.23% due to the removal of outliers and inconsistencies, thereby improving the quality of the dataset and the reliability of the model. As shown in Table 4, the dataset exhibits a wide range of judgment and summary lengths, with a mean compression ratio of 0.229, ensuring that the dataset provides meaningful summarization while avoiding excessive compression or verbosity.

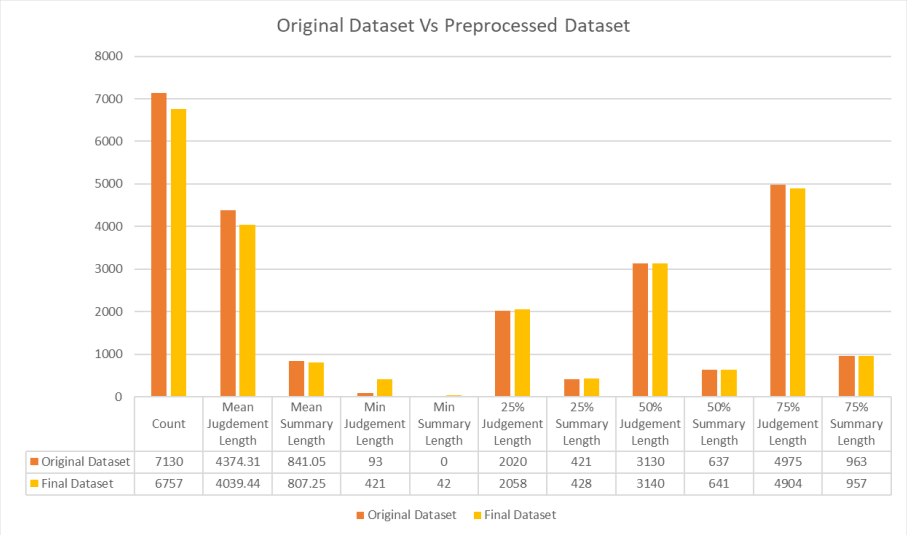
Algorithm 1 Preprocessing and filtering of IN-Abs dataset for legal text summarization.

Input: IN-Abs dataset D containing judgment–summary pairs (J_i, S_i)
Output: Filtered dataset D' with improved quality for summarization models
Initialize $D' \leftarrow \emptyset$
Step 1: Remove Outliers in Judgment Length
Compute judgment lengths $L = \{|J_1|, |J_2|, \ldots, |J_n|\}$
Determine 0.5th percentile ($P_{0.5}$) and 99th percentile (P_{99})
for all $(J_i, S_i) \in D$ **do**
    if $|J_i| < P_{0.5}$ or $|J_i| > P_{99}$ **then**
        Remove (J_i, S_i) from D
    end if
end for
Step 2: Compute Compression Ratio (CR)
for all $(J_i, S_i) \in D$ **do**
    Compute $CR_i = \frac{|S_i|}{|J_i|}$
end for
Step 3: Apply Compression Ratio Filtering
for all $(J_i, S_i) \in D$ **do**
    if $CR_i < 0.05$ or $CR_i > 0.5$ **then**
        Remove (J_i, S_i) from D
    end if
end for
Store the final filtered dataset as D'
return D'

Table 4. Descriptive statistics of the filtered IN-Abs dataset.

Statistic	Judgment Length (Tokens)	Summary Length (Tokens)	Compression Ratio (CR)
Total Samples	6757	6757	6757
Mean (Average)	4039.44	807.25	0.229
Standard Deviation	3180.25	643.38	0.105
Minimum	421	42	0.050
25th Percentile (Q1)	2058	428	0.144
Median (50th Percentile)	3140	641	0.216
75th Percentile (Q3)	4904	957	0.305
Maximum	24,268	7771	0.500

Figure 9 presents a comparative analysis between the original dataset and the final preprocessed dataset across multiple statistical parameters. The chart demonstrates the impact of preprocessing techniques, including outlier removal and compression ratio filtering, on the characteristics of the dataset. Notably, the mean judgment and summary lengths are reduced post-processing, contributing to a more uniform and reliable dataset. Furthermore, the interquartile ranges (Q1, Q2, and Q3) indicate that the refined dataset maintains a representative distribution while eliminating extreme variations. These enhancements ensure that the dataset is well suited for training a robust legal text summarization model, mitigating biases introduced by excessive text lengths and inconsistencies in summarization patterns.



	Count	Mean Jugdement Length	Mean Summary Length	Min Judgement Length	Min Summary Length	25% Judgement Length	25% Summary Length	50% Judgement Length	50% Summary Length	75% Judgement Length	75% Summary Length
Original Dataset	7130	4374.31	841.05	93	0	2020	421	3130	637	4975	963
Final Dataset	6757	4039.44	807.25	421	42	2058	428	3140	641	4904	957

Figure 9. Comparison of original and preprocessed dataset statistics. This bar chart highlights key differences in dataset size, judgment length, and summary length before and after preprocessing.

Randomization and Data Splitting

To mitigate potential biases arising from sequential dependencies within the dataset, a rigorous randomization process is employed prior to dataset partitioning. The filtered dataset is randomly shuffled to ensure a uniform distribution of textual characteristics across subsets, thereby enhancing the generalizability of the trained model.

Following the shuffling process, the dataset is stratified into distinct training and testing sets as follows:

- **Training Set:** Consisting of 6500 judgment–summary pairs, this subset is utilized to fine-tune the legal text summarization model, allowing it to learn domain-specific linguistic structures and summarization patterns.

- **Testing Set:** Comprising 257 judgment–summary pairs, this subset serves as an independent evaluation benchmark to assess model performance and generalization capabilities.

The finalized and processed dataset is subsequently made publicly available through the Hugging Face Hub under the designated repository, facilitating reproducibility and further advancements in the domain of legal text summarization. This structured and methodologically sound approach ensures the integrity, diversity, and reliability of the dataset, thereby contributing to the development of robust and high-quality summarization models for legal discourse.

5. Results and Discussion

This section on Results and Discussion presents a comprehensive evaluation of our proposed approach, integrating the Dynamic Legal RAG system and domain-specific fine-tuned language models for legal text summarization. This section is structured to highlight the performance of different retrieval and summarization mechanisms, providing an in-depth analysis of their effectiveness in enhancing the quality and accuracy of legal document summarization. Through systematic experimentation, we assess the impact of top-K chunk selection and the choice of retriever models on the Retrieval-Augmented Generation pipeline, identifying optimal configurations for legal knowledge retrieval. Furthermore, we conduct a comparative performance analysis of various baseline and fine-tuned large language models (LLMs), demonstrating the substantial performance gains achieved through domain-specific adaptation and real-time knowledge augmentation. The findings underscore the critical role of legal entity extraction and retrieval-based augmentation in ensuring high-fidelity, contextually rich legal summaries. Thus, this section serves as a pivotal discussion of the empirical results, offering insights into the strengths and limitations of different model configurations in the legal domain.

5.1. Dynamic Legal RAG Results

The Dynamic Legal RAG system is employed in real time to augment the summarization model whenever it encounters a case precedent or a provision–statute pair for which it lacks sufficient contextual information. To ensure effective retrieval of legally relevant content, we compare various retrieval mechanisms based on their performance across multiple evaluation metrics. The results of different retriever models with varying top-K chunk selections are presented in Table 5.

Table 5. Performance comparison of different retrievers with various top-K chunks.

Retriever Model	Top-K Chunks	ROUGE-1	ROUGE-2	ROUGE-L	Cosine Similarity	Legal Coverage (1–5)	Irrelevant Retrieval (%)	Redundancy (%)
BM25 [1]	Top 1	0.7805	0.6203	0.7234	0.85	4.4	5.8	9.3
	Top 3	0.9102	0.7647	**0.8894**	**0.94**	4.9	2.3	3.9
	Top 5	0.8810	0.7208	0.8543	0.92	4.6	3.6	5.1
DPR [14]	Top 1	0.6521	0.5017	0.6123	0.79	4.0	8.3	12.1
	Top 3	0.9214	0.7789	0.8612	0.91	4.7	4.8	7.5
	Top 5	0.8745	0.7180	0.8326	0.89	4.5	6.1	9.0
ColBERT [16]	Top 1	0.7032	0.5509	0.6640	0.81	4.2	7.0	10.4
	Top 3	0.8743	0.7316	0.8129	0.90	4.8	3.2	**3.5**
	Top 5	**0.9321**	0.7930	0.8821	0.88	**5.0**	5.0	6.8
SGPT [18]	Top 1	0.7235	0.5832	0.6824	0.83	4.3	6.2	9.1
	Top 3	0.8806	0.7452	0.8227	0.91	4.8	**2.1**	5.2
	Top 5	0.8619	**0.8013**	0.8521	0.90	4.7	3.5	6.0

5.1.1. Performance Evaluation of Different Retrievers

To determine the most suitable retriever and an optimal value of the number of retrieved chunks, we performed a comparative study using four state-of-the-art retrieval techniques, namely, BM25, Dense Passage Retrieval (DPR), ColBERT, and SGPT. The performance of these retrievers is evaluated based on standard ROUGE scores, cosine similarity, legal coverage, irrelevant retrieval percentage, and redundancy percentage.

(A) Description of Retrieval Models

1. BM25 (Best Matching 25) [1]: BM25 is a probabilistic ranking function that scores documents based on term frequency and inverse document frequency, adjusted by a length normalization factor. It is defined as follows:

$$\mathrm{S}(D,Q) = \sum_{t \in Q} \log\left(\frac{N - n_t + 0.5}{n_t + 0.5} + 1\right) \times \frac{(k_1+1) f_{t,D}}{k_1\left(1 - b + b \cdot \frac{|D|}{\mathrm{avgdl}}\right) + f_{t,D}} \tag{7}$$

where $f_{t,D}$ is the frequency of term t in document D, N is the total number of documents, and n_t is the number of documents containing t. The parameters k_1 and b control term saturation and document length normalization.

2. DPR (Dense Passage Retrieval) [15]: DPR employs a dual-encoder model in which both the query and the documents are embedded into a dense vector space, and similarity is computed using dot product similarity. The retrieval score is given by

$$S(Q,D) = E_Q(Q) \cdot E_D(D) \tag{8}$$

where E_Q and E_D represent the query and document encoders, respectively.

3. ColBERT (Contextualized Late Interaction Over BERT) [16]: ColBERT utilizes a BERT-based representation with late interaction between token-level embeddings. It refines retrieval by computing maximum similarity matching across query and document token embeddings, defined as

$$S(Q,D) = \sum_{q \in Q} \max_{d \in D} \cos(E_Q(q), E_D(d)) \tag{9}$$

where Q and D are the sets of token embeddings for the query and document, respectively; E_Q and E_D are the embedding functions; and cos denotes the cosine similarity.

4. SGPT (Sentence Graph Pretrained Transformer) [18]: SGPT leverages transformer-based embeddings optimized for sentence retrieval, incorporating semantic relationships beyond term matching. SGPT optimizes a contrastive learning objective for query-document matching. Given matching query–document pairs $(q^{(i)}, d^{(i)})$, the cost function $J_{CL}(\theta)$ is

$$J_{CL}(\theta) = \frac{1}{M} \sum_{i=1}^{M} \log \frac{\exp(\tau \cdot \sigma(f_\theta(q^{(i)}), f_\theta(d^{(i)})))}{\sum_{j=1}^{M} \exp(\tau \cdot \sigma(f_\theta(q^{(i)}), f_\theta(d^{(j)})))} \tag{10}$$

where f_θ are the model's output embeddings, σ is the cosine similarity, and τ is a temperature parameter.

(B) Evaluation Metrics

To objectively assess retrieval quality, we employ the following metrics:

- **ROUGE-1, ROUGE-2, and ROUGE-L** [52]: These measure lexical overlap between the retrieved documents and the ground truth.
- **Cosine Similarity**: Evaluates the contextual relevance of retrieved results.
- **Legal Coverage**: Assessed using a 5-point Likert scale by six legal annotators, measuring how well a retrieved document aligns with the legal query.

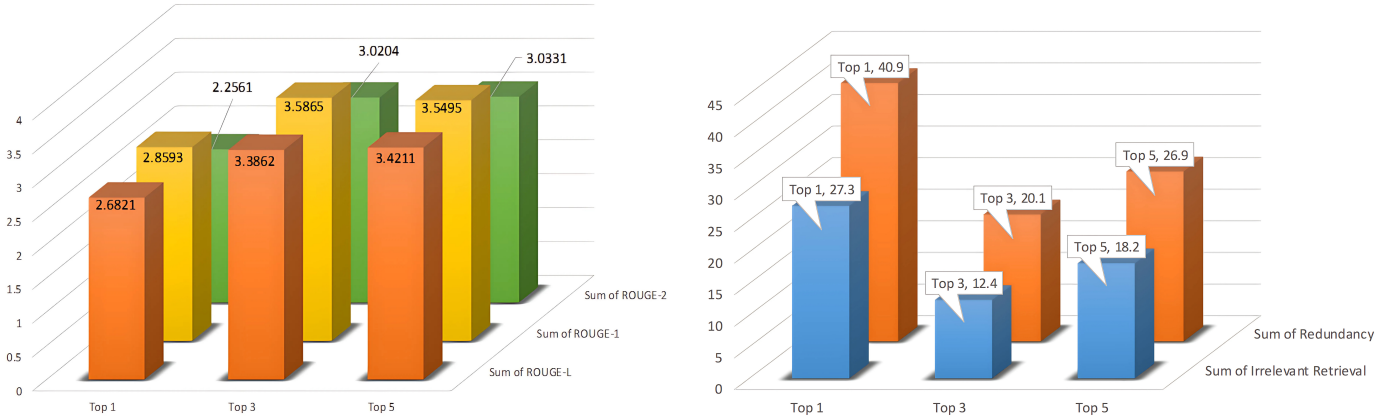
- **Irrelevant Retrieval Percentage**: Computes the proportion of retrieved documents that are semantically unrelated to the query.
- **Redundancy Percentage**: Measures how frequently the same legal excerpts appear across multiple retrieved results.

5.1.2. Analysis of Top-K Chunk Selection

To determine the optimal chunk retrieval size (K), we analyze the aggregated retrieval performance for top-1, top-3, and top-5 chunk selections. The results are illustrated in Figure 10.

From the analysis, we observe that top-3 retrieval consistently outperforms both top-1 and top-5 selections across all retrievers. While top-1 retrieval lacks sufficient legal context (Figure 10a), top-5 retrieval introduces unnecessary redundancy and increases the likelihood of retrieving irrelevant information (Figure 10b). Top-3 retrieval strikes the best balance between contextual relevance and retrieval precision, ensuring that retrieved chunks contain sufficient yet non-redundant legal information.

The selection of top-3 retrieval exhibits a form of harmonic balance, maintaining equilibrium between informativeness and conciseness. By avoiding the extremes of overly limited or excessive retrieval, this approach ensures that the summarization model captures the essential legal context without introducing noise or redundancy. This harmonic balance enhances both the quality and interpretability of the generated summaries, making the model well-suited for the nuanced demands of legal text processing.



(**a**) (**b**)

Figure 10. Performance and retrieval quality of top-K chunks. (**a**) Aggregated ROUGE scores for top-1, top-3, and top-5 chunks. (**b**) Combined redundancy and irrelevant retrieval analysis.

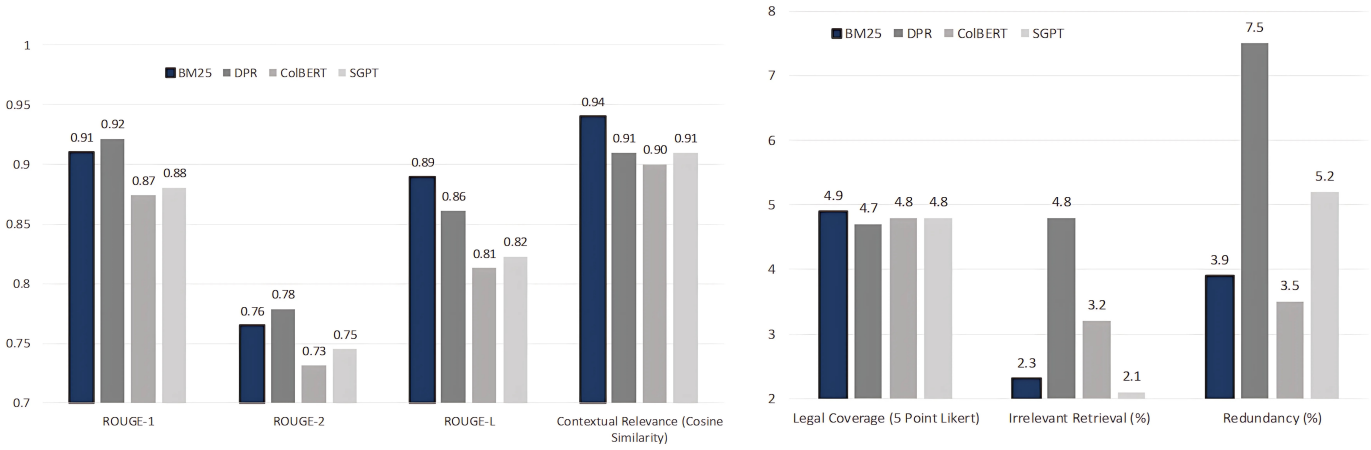
5.1.3. Selection of the Optimal Retriever

Having established that top-3 retrieval is optimal, we now compare retrievers based on their performance for top-3 chunk selection (Figure 11).

BM25 emerges as the best-performing retriever across all evaluation metrics. It achieves the highest ROUGE-L (0.8894), the highest cosine similarity (0.94), and the lowest irrelevant retrieval (2.3%). ColBERT, DPR, and SGPT show competitive performance, but BM25 offers the best trade-off between relevance, redundancy, and retrieval quality. While DPR and ColBERT achieve high ROUGE-2 scores (Figure 11a), they suffer from increased irrelevant retrieval percentages (4.8% and 3.2%, respectively—Figure 11b), making them less reliable for legal precedents where precision and factual consistency are crucial.

The legal domain involves structured and standardized language—for instance, statutory references like "Section 300 IPC" or precedents like "Kesavananda Bharati v. State of Kerala"—that benefit more from sparse keyword-based retrieval than from dense semantic

embeddings. BM25, by leveraging exact term frequencies and inverse document frequencies, is well suited for such contexts. It aligns naturally with the format of legal documents, where even minor variations in wording can imply significant legal differences.



(**a**) (**b**)

Figure 11. Retriever comparison for top-3 chunks. (**a**) ROUGE scores and cosine similarity (Y-axis starts from 0.7 for better differentiation). (**b**) Legal coverage, irrelevant retrieval, and redundancy (Y-axis starts from 2 for better differentiation) across BM25, DPR, ColBERT, and SGPT.

Additionally, BM25 does not require pretraining or fine-tuning on domain-specific corpora, offering a lightweight yet powerful retrieval mechanism. In contrast, dense retrievers like DPR and SGPT require domain adaptation and significantly more computational resources. ColBERT, although strong in late interaction, incurs a high memory overhead and latency, which is not ideal for real-time summarization tasks.

Based on an extensive evaluation of retrieval performance using a sample of 20 legal queries (10 case precedents and 10 provision–statute pairs), we conclude that BM25 with top-3 retrieval chunks is the most effective configuration for our Dynamic Legal RAG System. This setting ensures that the RAG system retrieves the most legally relevant content with minimal redundancy and irrelevance, thereby enhancing the quality of summarization when encountering unfamiliar legal precedents or statutory provisions.

Thus, we employ BM25 (top-3) as the default retriever within our RAG pipeline to dynamically enrich the summarization model with legal knowledge, ensuring enhanced factual consistency, contextual relevance, and domain-specific accuracy.

5.1.4. Qualitative Analysis of Retrieved Context in Dynamic Legal RAG

To complement our evaluation of summary outputs, we present a qualitative analysis of the legal context retrieved by the Dynamic Legal RAG system prior to summarization. This analysis aims to demonstrate that the top-3 chunks retrieved via the BM25 retriever—guided by Legal Named Entity Recognition (NER) anchors such as statutes, provisions, and precedents—contain highly relevant and factually accurate legal information. By evaluating the semantic and statutory alignment of these retrieved chunks with the legal query and corresponding judgment, we assess whether the retrieval step effectively supplies the summarization model with grounded, hallucination-free context. This layer of analysis validates the core hypothesis that real-time, entity-aware legal retrieval substantially reduces hallucination risks and improves the legal and factual consistency of generated summaries.

Sample Query 1—Statutory Anchor: Section 300 IPC

This query focuses on the provision defining "murder" under the Indian Penal Code. The Legal NER system identifies Section 300 as a relevant anchor, triggering the retrieval of statutory and precedential content related to culpable homicide. As seen in Table 6, the retrieved chunks collectively reinforce statutory, doctrinal, and precedential grounding for Section 300 IPC. This layered retrieval mitigates factual drift and supports structured legal summarization.

Sample Query 2—Precedent: *Kesavananda Bharati v. State of Kerala (1973)*

This query pertains to one of the most significant constitutional precedents in Indian legal history. The Legal NER module detects the case name as a precedent, prompting the BM25-based retriever to fetch contextually relevant segments related to the judgment. The retrieved chunks include statutory references to Article 368, doctrinal interpretations of the Basic Structure Doctrine, and precedential lineage supporting its enduring constitutional authority. As shown in Table 7, legal experts verified that the retrieved context not only reinforces the doctrine's foundational scope but also supplies factual details critical for summarization without hallucination.

Table 6. Sample Query 1—Section 300 IPC.

Top-3 Retrieved Chunks	Expert Evaluation and Remarks
Chunk 1: Section 300 of the Indian Penal Code defines murder. Except in cases falling under the exceptions specified in the section, culpable homicide is murder if the act is done with the intention of causing death, or if it is done with the intention of causing such bodily injury as is likely to cause death, or if the offender knows that his act is imminently dangerous to life.	**Highly Relevant** • Provides a direct definition of "murder" from Section 300 IPC. • Anchors the summarizer in accurate statutory language, reducing risk of paraphrased misrepresentation. • Retrieved from: Indian Penal Code, Bare Act.
Chunk 2: The classification of homicide into culpable homicide not amounting to murder and murder is elaborated under Sections 299 and 300 IPC. The distinction hinges upon the degree of intention and knowledge involved in the act.	**Contextually Supportive** • Offers doctrinal distinction between Sections 299 and 300 IPC. • Helps clarify gradation of offenses for the summarizer—reinforcing legal specificity. • Retrieved from: Standard Criminal Law Treatise.
Chunk 3: In *Virsa Singh v. State of Punjab* (1958), the Supreme Court clarified that for a conviction under Section 300, it is essential to establish both the intention to inflict the injury and the sufficiency of that injury in the ordinary course of nature to cause death.	**Precedential Grounding** • Backs the statute with case law that interprets intent and causation in murder charges. • Reinforces factual consistency and helps the summarizer avoid hallucinated legal standards. • Retrieved from: Landmark Judgments Volume I.

Table 7. Sample Query 2—Precedent: *Kesavananda Bharati v. State of Kerala (1973).*

Top-3 Retrieved Chunks	Expert Evaluation and Remarks
Chunk 1: In *Kesavananda Bharati v. State of Kerala*, the Supreme Court ruled that although Parliament has the power to amend the Constitution under Article 368, it cannot alter or destroy the basic structure of the Constitution. This judgment laid the foundation for the Basic Structure Doctrine in Indian constitutional law.	**Highly Relevant** • Clearly states the central doctrinal principle established in the case. • Helps the summarizer stay grounded in accurate constitutional limitations. • Retrieved from: Landmark Judgments Volume I.
Chunk 2: The majority opinion in the case (7:6 split) held that features such as judicial review, federalism, secularism, and the rule of law form part of the Constitution's basic structure and cannot be amended, even through constitutional amendments.	**Contextually Supportive** • Provides additional granularity on which constitutional features are protected. • Clarifies scope of judicial restraint, improving factual completeness in summaries. • Retrieved from: Indian Constitutional Law Compendium.
Chunk 3: The Basic Structure Doctrine introduced in Kesavananda Bharati has been consistently upheld in subsequent rulings, including in *Indira Nehru Gandhi v. Raj Narain* and *Minerva Mills v. Union of India*, reinforcing its foundational status in constitutional jurisprudence.	**Precedential Continuity** • Demonstrates the long-term legal validity and application of the doctrine. • Enhances temporal grounding for summaries referencing constitutional amendment cases. • Retrieved from: Judicial Precedent Tracker, Vol II.

5.2. Results on Legal Text Summarization

This section presents a comparative analysis of the performance of various legal text summarization models, including baseline pretrained transformer models, fine-tuned models, and fine-tuned models integrated with legal-domain-specific knowledge. The results, as detailed in Table 8, highlight the progressive improvements achieved through fine-tuning and integration of domain knowledge.

Table 8. Performance comparison of baseline, fine-tuned, and fine-tuned models integrated with legal-domain-specific knowledge obtained as output from Legal NER and Legal RAG.

Model Type	Model	ROUGE-1	ROUGE-2	ROUGE-L	METEOR [53]	BERTScore Precision	BERTScore Recall	BERTScore F1
Pretrained Baseline (Zero-Shot)	LLaMA 3.1-8B	0.3986	0.2402	0.2458	0.3689	0.8134	0.8312	0.8223
	DeepSeek-7B	**0.4405**	**0.2756**	**0.2803**	**0.4053**	**0.8391**	**0.8542**	**0.8463**
	Zephyr-7B Beta	0.3892	0.2321	0.2384	0.3504	0.8105	0.8274	0.8189
	Mistral-7B	0.3541	0.2036	0.2101	0.3178	0.7903	0.8065	0.7981
	LLaMA 2-7B	0.3324	0.1857	0.1902	0.2987	0.7713	0.7872	0.7786
Fine-Tuned Models (LoRA)	**LLaMA 3.1-8B**	**0.4528**	0.2874	0.2929	**0.4189**	0.8493	0.8647	**0.8658**
	DeepSeek-7B	0.4521	**0.2892**	**0.2956**	0.4105	**0.8501**	**0.8652**	0.8573
	Zephyr-7B Beta	0.4109	0.2525	0.2583	0.3805	0.8232	0.8417	0.8323
	Mistral-7B	0.3674	0.2159	0.2217	0.3295	0.7992	0.8153	0.8071
	LLaMA 2-7B	0.3457	0.1983	0.2042	0.3120	0.7821	0.7986	0.7902
Fine-Tuned Models + Domain Knowledge Integration	**LLaMA 3.1-8B**	**0.5214**	**0.3572**	**0.3652**	**0.4951**	**0.8834**	**0.8979**	**0.8906**
	DeepSeek-7B	0.4593	0.2981	0.3049	0.4267	0.8567	0.8703	0.8631
	Zephyr-7B Beta	0.4328	0.2675	0.2746	0.4012	0.8317	0.8472	0.8393
	Mistral-7B	0.4462	0.2898	0.2969	0.4154	0.8498	0.8635	0.8561
	LLaMA 2-7B	0.4883	0.3236	0.3312	0.4508	0.8712	0.8856	0.8784

These qualitative insights reaffirm that RAG-based retrieval—when driven by entity-aware legal indexing—offers targeted, factually grounded input that bolsters summary accuracy while substantially reducing hallucination risk.

5.2.1. Performance of Pretrained Transformer Models (Zero-Shot)

Among the pretrained baseline models evaluated, the DeepSeek-7B model exhibits the best performance across all metrics. With a ROUGE-1 score of 0.4405, a ROUGE-2 score of 0.2756, and a BERTScore F1 [54] of 0.8463, DeepSeek-7B surpasses other baseline models. This superior performance can be attributed to its large-scale reinforcement learning (RL) training, which enhances its reasoning capabilities. Additionally, its extended context window of 16,000 tokens allows it to process lengthy legal documents more effectively compared to models such as LLaMA 2-7B, which has a significantly shorter context window of 4096 tokens.

However, LLaMA 2-7B emerges as the weakest baseline model, with a ROUGE-1 score of 0.3324 and a ROUGE-2 score of 0.1857. This lower performance can be attributed to its relatively outdated architecture and smaller context handling capacity. Mistral-7B also underperforms compared to DeepSeek-7B, achieving a ROUGE-1 score of 0.3541, reflecting its limitations in processing lengthy and intricate legal texts without domain adaptation.

5.2.2. Performance of Fine-Tuned Models (LoRA-Based Adaptation)

Fine-tuning significantly improves the performance of all models. Among the fine-tuned models, LLaMA 3.1-8B emerges as the most effective, achieving a ROUGE-1 score of 0.4528 and a BERTScore F1 of 0.8658. The superior performance of LLaMA 3.1-8B can be attributed to its robust multilingual and long-context capabilities, supporting up to 128K tokens. These properties enable the model to retain and understand longer legal arguments, citations, and statutory references, making it particularly suitable for legal text summarization. pSeek-7B, though slightly behind LLaMA 3.1-8B, maintains competitive performance with a ROUGE-1 score of 0.4521 and a BERTScore F1 of 0.8573. The slight

decline in performance relative to LLaMA 3.1-8B may be attributed to its lack of explicit fine-tuning for legal text structures and terminology, despite its strong reasoning abilities. Meanwhile, Zephyr-7B Beta, Mistral-7B, and LLaMA 2-7B exhibit moderate improvements after fine-tuning, with LLaMA 2-7B remaining the weakest performer even after adaptation.

5.2.3. Performance of Fine-Tuned Models Integrated with Legal Domain Knowledge

Integrating domain-specific legal knowledge via Named Entity Recognition (NER) and Retrieval-Augmented Generation (RAG) yields substantial performance gains. LLaMA 3.1-8B, when combined with legal entity recognition and precedent-statute querying, significantly outperforms all other models, achieving a ROUGE-1 score of 0.5214 and a BERTScore F1 of 0.8906. The notable increase in performance underscores the impact of dynamically retrieving relevant legal provisions and case precedents during summarization, thereby enhancing the factual accuracy and contextual coherence.

Interestingly, LLaMA 2-7B, which had previously underperformed in both baseline and fine-tuned settings, exhibits a marked improvement upon domain knowledge integration. With a ROUGE-1 score of 0.4883 and a BERTScore F1 of 0.8784, it surpasses even DeepSeek-7B, suggesting that its subpar performance in prior settings was largely due to its inability to effectively process legal-specific nuances rather than inherent model limitations.

DeepSeek-7B and Mistral-7B also experience notable improvements post integration, with DeepSeek-7B achieving a ROUGE-1 score of 0.4593 and Mistral-7B reaching 0.4462. The enhancement in performance further validates the effectiveness of augmenting LLMs with domain-adaptive retrieval mechanisms to address challenges inherent in legal text summarization, such as maintaining legal precision and contextual integrity.

5.2.4. Summary of Comparative Analysis

The findings from Table 8 reinforce the necessity of model adaptation and domain knowledge integration for effective legal text summarization. The key takeaways include the following:

- **DeepSeek-7B is the best performing baseline model** due to its extended context window and strong reasoning capabilities, while LLaMA 2-7B is the weakest.
- **Fine-tuning significantly improves model performance, with LLaMA 3.1-8B emerging as the strongest fine-tuned model**, leveraging its superior long-context handling and multilingual capabilities.
- **Domain knowledge integration via NER and RAG leads to the highest performance gains**, particularly for LLaMA 3.1-8B, which benefits from real-time retrieval of legal precedents and statutes.
- **LLaMA 2-7B, despite its poor baseline performance, improves drastically when enhanced with domain knowledge**, suggesting that its primary limitation lies in contextual understanding rather than model architecture.

These results highlight the effectiveness of combining fine-tuned LLMs with structured legal entity extraction and retrieval mechanisms to ensure a high-fidelity, legally accurate summarization.

5.3. Legal-Domain-Specific Evaluation Metrics

While standard metrics such as ROUGE and BERTScore offer insights into surface-level similarity, they do not fully capture the factual and legal fidelity expected in judicial summaries. To address this, we introduce a suite of legal-domain-specific evaluation metrics aimed at assessing factual grounding, statutory precision, and domain alignment. These include the following:

- **Citation Density (CD)**: The number of legal citations (e.g., case names, statute references) per 100 tokens.
- **Statutory Inclusion Rate (SIR)**: The percentage of statutes/provisions mentioned in the ground truth that are also included in the generated summary.
- **Precedent Alignment Score (PAS)**: Measures the inclusion and accuracy of cited precedents with respect to the gold summary.
- **Factual Consistency Index (FCI)**: Proportion of factual claims in the summary that are supported by retrieved legal content.
- **Legal Entity Preservation Rate (LEPR)**: The percentage of extracted legal entities retained in the summary, relative to the input judgment.

These metrics are computed for the top-performing configuration—LLaMA 3.1-8B (Fine-Tuned + RAG)—and compared with its zero-shot and fine-tuned-only variants, highlighting improvements in legal relevance, precision, and factual consistency.

The results in Table 9 clearly demonstrate the advantages of integrating domain-specific retrieval with fine-tuned summarization models. The RAG-based model significantly outperforms both the zero-shot and fine-tuned configurations across all legal-domain-specific metrics. Specifically, it achieves the highest citation density (2.21), indicating a strong inclusion of relevant legal citations per 100 tokens, and a statutory inclusion rate of 87.2%, reflecting robust incorporation of pertinent legal provisions in the generated summaries.

Table 9. Comparison of legal-domain-specific evaluation metrics across model configurations.

Evaluation Metric	Zero-Shot Model	Fine-Tuned Model	RAG-Based Model
Citation Density (CD)	0.92	1.63	**2.21**
Statutory Inclusion Rate (SIR)	42.5%	68.4%	**87.2%**
Precedent Alignment Score (PAS)	28.3%	59.1%	**85.6%**
Factual Consistency Index (FCI)	61.0%	75.3%	**92.7%**
Legal Entity Preservation Rate (LEPR)	52.8%	79.6%	**93.8%**

Notably, the precedent alignment score (85.6%) highlights the model's ability to retain and reference judicial precedents accurately, a critical factor in legal summarization. Similarly, the factual consistency index (92.7%) confirms the model's grounding in truth-preserving generation, minimizing hallucinations and factual drift. The legal entity preservation rate (93.8%) further underscores the effectiveness of Legal NER and RAG in maintaining key actors, statutes, and procedural references.

These results validate our methodology's core hypothesis—that domain-aware augmentation and structured legal entity guidance are essential for producing accurate, trustworthy, and legally usable summaries.

5.4. Qualitative Results: Summary Comparisons Across Models

While quantitative metrics such as ROUGE, BERTScore, and cosine similarity provide valuable insights into model performance, they may not fully capture the nuanced legal fidelity required in judicial summarization tasks. Therefore, we supplement our evaluation with a qualitative analysis, comparing summaries generated by three different settings—zero-shot LLMs, fine-tuned LLMs, and our proposed RAG-augmented framework.

For each sample case, we provide the excerpted legal judgment, the human-annotated (ground-truth) summary, and the outputs from the three summarization approaches. We then offer expert commentary highlighting how each method performs in terms of legal completeness, factual grounding, and doctrinal relevance. This comparison reinforces the effectiveness of our entity-driven, dynamically retrieved RAG pipeline in enhancing both accuracy and legal interpretability (see Tables 10 and 11).

5.4.1. Sample Case 1: Indra Sawhney v. Union of India (1992)

Excerpt from Judgment:

"This case involves the implementation of 27% reservations for OBCs in central government jobs and discusses the permissible extent of reservations under Article 16(4). The Court deliberated on the meaning of 'backward classes', the ceiling limit of 50%, and the exclusion of the creamy layer from reservation benefits. The role of Mandal Commission's recommendations was also evaluated in light of constitutional provisions."

Gold-Standard (Human Annotated) Summary:

The Supreme Court upheld 27% reservation for OBCs under Article 16(4) but introduced the concept of the creamy layer to exclude the more advanced among the backward classes. It also imposed a 50% cap on total reservations and emphasized periodic review of backward class lists. The judgment validated the core recommendations of the Mandal Commission.

Table 10. Qualitative summary comparison—Indra Sawhney v. Union of India (1992).

Generated Summary	Expert Justification/Observation
Zero-Shot Summary: The judgment discusses the issue of reservation in government jobs for backward classes and interprets constitutional rights. The Court referred to the Mandal Commission and gave directions on equality in employment.	**Limitations:** • Lacks mention of key doctrines such as creamy layer and the 50% cap. • Vague phrasing like "gave directions" doesn't convey binding legal principles. • Fails to cite Article 16(4) or highlight judicial reasoning.
Fine-Tuned Summary: The Court upheld 27% reservations for OBCs under Article 16(4) and ruled that the 'creamy layer' should be excluded. It also imposed a 50% ceiling on reservations and addressed the validity of Mandal Commission's recommendations.	**Improvements:** • Covers essential legal elements like creamy layer and Article 16(4). • Specifies the reservation ceiling and key actors involved. • Lacks interpretive insight and doesn't mention periodic review provisions.
RAG-Based Summary: The Supreme Court, while interpreting Article 16(4), upheld 27% reservation for OBCs based on the Mandal Commission's findings but excluded the creamy layer to ensure equality. It instituted a 50% reservation ceiling, asserting that excessive quotas violate constitutional balance. The Court also recommended periodic reviews of backward class status to prevent misuse and ensure targeted benefits.	**Most Complete:** • Captures deeper legal reasoning such as "constitutional balance". • Cites interpretive doctrine and periodic review—grounded in retrieval. • Closely matches the gold-standard in both structure and factual richness.

5.4.2. Sample Case 2: Maneka Gandhi v. Union of India (1978)

Excerpt from Judgment:

"This landmark case questioned the arbitrary impounding of Maneka Gandhi's passport under the Passport Act, 1967, and challenged the violation of Article 21 of the Constitution. The Court interpreted the scope of 'personal liberty' broadly and emphasized that any procedure established by law must be fair, just, and reasonable."

Gold-Standard (Human Annotated) Summary:

The Supreme Court ruled that the right to travel abroad is protected under Article 21. It expanded the interpretation of personal liberty, holding that procedures depriving liberty must be fair, just, and reasonable. The decision laid the foundation for procedural due process in Indian constitutional law.

Table 11. Qualitative Summary Comparison—Maneka Gandhi v. Union of India (1978).

Generated Summary	Expert Justification/Observation
Zero-Shot Summary: The case is about a passport issue where the Court discussed personal liberty and gave a ruling on constitutional rights under Article 21.	**Limitations:** • Very generic and oversimplified—does not reflect the constitutional significance of the case. • Lacks any mention of "procedural due process" or the Court's interpretive shift. • Phrases like "gave a ruling" do not reflect judicial reasoning or doctrinal development.
Fine-Tuned Summary: The Court held that impounding a passport violates Article 21 and must follow a fair procedure. It expanded the meaning of personal liberty and emphasized that laws must be reasonable and just.	**Improvements:** • Correctly identifies the expansion of Article 21 and mentions fairness and reasonableness. • Provides better legal framing than zero-shot output. • Misses crucial terminology such as "procedural due process" and unified interpretation of Articles 14, 19, and 21.
RAG-Based Summary: In Maneka Gandhi v. Union of India, the Supreme Court held that the right to travel abroad falls under personal liberty in Article 21. The judgment revolutionized Indian constitutional law by introducing the doctrine of procedural due process, asserting that any law depriving personal liberty must be fair, just, and reasonable. The case overruled narrow interpretations of liberty and linked Articles 14, 19, and 21 in a unified reading.	**Most Complete:** • Reflects the broader doctrinal shift introduced by the case, including "procedural due process". • Introduces the unification of constitutional rights under Articles 14, 19, and 21. • Goes beyond surface facts to express the legal theory and long-term impact.

6. Conclusions and Future Work

This paper presents a focused exploration of dynamic Retrieval-Augmented Generation (RAG) and legal text summarization, advancing the state of legal document understanding through domain-specific knowledge integration. Our Dynamic Legal RAG system, leveraging the BM25 retriever with top-3 chunk selection, exemplifies the principle of symmetry in information flow by striking an optimal balance between contextual relevance and retrieval precision. This balanced approach ensures that the information flow remains neither sparse nor overloaded, surfacing legally pertinent content without introducing unnecessary redundancy or irrelevant data. As a result, this configuration consistently outperforms alternative retrieval mechanisms and aligns the generated summaries closer to the factual and contextual truth, minimizing the risk of hallucination. Through rigorous evaluation across multiple legal queries, BM25 emerges as the most effective retriever, making it the default choice for real-time legal knowledge augmentation. The top-3 retrieval strategy further enhances this by maintaining harmonic balance in the selection of information, ensuring proportional and well-distributed content representation. This synergy between dynamic retrieval and controlled information flow underpins the reliability and coherence of the retrieved legal content.

In the realm of legal text summarization, our experimental results highlight the critical importance of domain adaptation and knowledge augmentation. Among the baseline models, DeepSeek-7B stands out for its extended context window and robust reasoning capabilities, while LLaMA 2-7B shows the weakest performance. Fine-tuning significantly improves summarization quality, with LLaMA 3.1-8B excelling due to its advanced long-context handling and multilingual capabilities. The integration of domain-specific knowledge through Legal Named Entity Recognition (NER) and the Dynamic Legal RAG system results in the highest performance gains. This approach not only enriches the context but also aligns the generated summaries with the underlying legal facts, ensuring that the summarization model remains hallucination-free and legally accurate. Ultimately, the principle of symmetry in information flow emerges as a cornerstone of our methodology, harmonizing the interplay between retrieval precision and contextual completeness. By balancing information representation and ensuring the seamless integration of domain-

specific knowledge, our approach sets a new benchmark for legal text summarization and real-time knowledge augmentation.

The findings of this study open up several promising directions for future research. Expanding the retrieval framework by exploring hybrid models that combine BM25's efficiency with neural retrievers such as ColBERT or DPR could further enhance retrieval quality. Adaptive fragmentation strategies based on legal document structure and query complexity merit investigation to optimize the granularity of content. In legal text summarization, developing reinforcement learning with human feedback (RLHF) [55] and legal-specific evaluation metrics beyond ROUGE and BLEU could better align the generated summaries with expert expectations. Furthermore, extending this framework to support multilingual legal texts and jurisdiction-specific legal interpretations would broaden its applicability. Many legal systems across the world—including the European Union, South Asia, and Africa—function in multiple official languages. Future work could involve incorporating translation-aware retrieval mechanisms, multilingual pretrained LLMs (e.g., mBERT [56], XLM-R [57]), or fine-tuning bilingual LLMs like DeepSeek to handle cross-lingual judgment summarization. Incorporating multilingual Legal NER pipelines would also help in recognizing language-specific statutes and provisions. These steps would support a scalable, inclusive framework adaptable to diverse legal jurisdictions.

While the present system is tailored to the Indian legal domain, we acknowledge that adapting this framework to other jurisdictions (e.g., the United States, European Union, or United Kingdom) would require substantial domain-specific reengineering. Legal systems differ not only in structure (common law vs. civil law), but also in citation standards, case referencing, and drafting conventions. To generalize effectively, future work would involve integrating localized statutes, region-specific case law corpora, and retraining Legal NER systems for jurisdictional entity schemas. Finally, real-world deployment of this integrated system for practical applications such as case law analysis, automated legal drafting, and contract review would provide valuable insights into its operational effectiveness. Collaborations with legal professionals across multiple legal systems would ensure that technological advancements remain closely aligned with the nuanced requirements of global legal practice, paving the way for more sophisticated and reliable legal AI systems.

Author Contributions: Conceptualization, S.A.M.; Software, S.A.M.; Validation, S.A.M.; Data curation, S.A.M.; Writing—original draft, S.A.M.; Writing—review & editing, K.S.E.; Visualization, S.A.M.; Supervision, K.S.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the University Grants Commission (UGC) of India under the Junior Research Fellowship (JRF) scheme, awarded based on the UGC NET Examination conducted in December 2019, which is gratefully acknowledged. Grant Number: 190520718400.

Data Availability Statement: The data presented in this study are openly available at https://zenodo.org/record/7234359#.Y2tShdJByC1 [IN-Abs Dataset] (accessed at 11 March 2025).

Acknowledgments: The authors would like to thank the Centre for Research, Anna University, for facilitating the proper disbursement of funds from the University Grants Commission (UGC) under the Junior Research Fellowship (JRF) scheme. The authors also extend their gratitude to the Department of Computer Science and Engineering, College of Engineering Guindy, Anna University, for providing the necessary administrative and technical support, including access to the GPU Lab facility, which was instrumental in the successful completion of this research article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Robertson, S.; Zaragoza, H. The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.* **2009**, *3*, 333–389. [CrossRef]
2. Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. The llama 3 herd of models. *arXiv* **2024**, arXiv:2407.21783.
3. Ramos, J. Using tf-idf to determine word relevance in document queries. In *First Instructional Conference on Machine Learning*; Citeseer: Princeton, NJ, USA, 2003; Volume 242, pp. 29–48.
4. Koubarakis, M.; Skiadopoulos, S.; Tryfonopoulos, C. Logic and computational complexity for Boolean information retrieval. *IEEE Trans. Knowl. Data Eng.* **2006**, *18*, 1659–1666. [CrossRef]
5. Gomes, T.; Ladeira, M. A new conceptual framework for enhancing legal information retrieval at the Brazilian Superior Court of Justice. In Proceedings of the 12th International Conference on Management of Digital EcoSystems, Virtual, 2–4 November 2020; pp. 26–29.
6. Costa, W.M.; Pedrosa, G.V. Legal Information Retrieval Based on a Concept-Frequency Representation and Thesaurus. In Proceedings of the 25th International Conference on Enterprise Information Systems, Prague, Czech Republic, 24–26 April 2023; pp. 303–311.
7. Mandal, A.; Ghosh, K.; Ghosh, S.; Mandal, S. Unsupervised approaches for measuring textual similarity between legal court case reports. *Artif. Intell. Law* **2021**, *29*, 417–451. [CrossRef]
8. Liu, L.; Liu, L.; Han, Z. Query Revaluation Method For Legal Information Retrieval. In Proceedings of the Forum for Information Retrieval Evaluation (FIRE 2020) Working Notes, Hyderabad, India, 16–20 December 2020; pp. 18–21.
9. Balaji, N.N.A.; Bharathi, B.; Bhuvana, J. Legal Information Retrieval and Rhetorical Role Labelling for Legal Judgements. In Proceedings of the Forum for Information Retrieval Evaluation (FIRE 2020) Working Notes, Hyderabad, India, 16–20 December 2020; pp. 26–30.
10. Kanapala, A.; Jannu, S.; Pamula, R. Passage-based text summarization for legal information retrieval. *Arab. J. Sci. Eng.* **2019**, *44*, 9159–9169. [CrossRef]
11. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; Volume 1 (Long and Short Papers), pp. 4171–4186.
12. Chalkidis, I.; Fergadiotis, M.; Malakasiotis, P.; Aletras, N.; Androutsopoulos, I. LEGAL-BERT: The muppets straight out of law school. *arXiv* **2020**, arXiv:2010.02559.
13. Anand, D.; Wagh, R. Effective deep learning approaches for summarization of legal texts. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 2141–2150. [CrossRef]
14. Althammer, S.; Askari, A.; Verberne, S.; Hanbury, A. DoSSIER@ COLIEE 2021: Leveraging dense retrieval and summarization-based re-ranking for case law retrieval. *arXiv* **2021**, arXiv:2108.03937.
15. Karpukhin, V.; Oğuz, B.; Min, S.; Lewis, P.; Wu, L.; Edunov, S.; Chen, D.; Yih, W.t. Dense passage retrieval for open-domain question answering. *arXiv* **2020**, arXiv:2004.04906.
16. Khattab, O.; Zaharia, M. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Xi'an, China, 25–30 July 2020; pp. 39–48.
17. Wang, X.; Macdonald, C.; Tonellotto, N.; Ounis, I. ColBERT-PRF: Semantic pseudo-relevance feedback for dense passage and document retrieval. *Acm Trans. Web* **2023**, *17*, 1–39. [CrossRef]
18. Muennighoff, N. Sgpt: Gpt sentence embeddings for semantic search. *arXiv* **2022**, arXiv:2202.08904.
19. Louis, A.; Van Dijck, G.; Spanakis, G. Finding the law: Enhancing statutory article retrieval via graph neural networks. *arXiv* **2023**, arXiv:2301.12847.
20. Mao, Y.; He, P.; Liu, X.; Shen, Y.; Gao, J.; Han, J.; Chen, W. Generation-augmented retrieval for open-domain question answering. *arXiv* **2020**, arXiv:2009.08553.
21. Mihalcea, R.; Tarau, P. Textrank: Bringing order into text. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, Barcelona, Spain, 25–26 July 2004; pp. 404–411.
22. Erkan, G.; Radev, D.R. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.* **2004**, *22*, 457–479. [CrossRef]
23. Jain, D.; Borah, M.D.; Biswas, A. Fine-tuning textrank for legal document summarization: A Bayesian optimization based approach. In Proceedings of the 12th Annual Meeting of the Forum for Information Retrieval Evaluation, Hyderabad, India, 16–20 December 2020; pp. 41–48.
24. Kumar, H.; Jayanth, P.; Anand Kumar, M. Large Language Models for Indian Legal Text Summarisation. In Proceedings of the 2024 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), Bangalore, India, 12–14 July 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 1–5.

25. Liu, C.L.; Chen, K.C. Extracting the gist of Chinese judgments of the supreme court. In Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law, Montreal, QC, Canada, 17–21 June 2019; pp. 73–82.
26. Nallapati, R.; Zhai, F.; Zhou, B. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31.
27. Liu, Y.; Lapata, M. Text summarization with pretrained encoders. *arXiv* **2019**, arXiv:1908.08345.
28. Shukla, A.; Bhattacharya, P.; Poddar, S.; Mukherjee, R.; Ghosh, K.; Goyal, P.; Ghosh, S. Legal case document summarization: Extractive and abstractive methods and their evaluation. *arXiv* **2022**, arXiv:2210.07544.
29. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv* **2019**, arXiv:1910.13461.
30. Ni, J.; Abrego, G.H.; Constant, N.; Ma, J.; Hall, K.B.; Cer, D.; Yang, Y. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. *arXiv* **2021**, arXiv:2108.08877.
31. Zhang, J.; Zhao, Y.; Saleh, M.; Liu, P. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In Proceedings of the International Conference on Machine Learning. PMLR, Virtual, 3–18 July 2020; pp. 11328–11339.
32. Kale, A.R.; Deshmukh, P.R. Abstractive Text Summarization: A Transformer Based Approach. In Proceedings of the 2024 IEEE 9th International Conference for Convergence in Technology (I2CT), Pune, India, 5–7 August 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 1–4.
33. Myla, S.D.; Saini, R.; Kapoor, N. Enhanced Text Summarization through Hybrid Integration of RoBERTa, T5, and Pegasus Models. In Proceedings of the 2024 International Conference on Advances in Modern Age Technologies for Health and Engineering Science (AMATHE), Shivamogga, India, 9–10 May 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 1–8.
34. Zaheer, M.; Guruganesh, G.; Dubey, K.A.; Ainslie, J.; Alberti, C.; Ontanon, S.; Pham, P.; Ravula, A.; Wang, Q.; Yang, L.; et al. Big bird: Transformers for longer sequences. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 17283–17297.
35. Kumar, V.B.; Bhattacharjee, K.; Gangadharaiah, R. Towards cross-domain transferability of text generation models for legal text. In Proceedings of the Natural Legal Language Processing Workshop 2022, Abu Dhabi, United Arab Emirates, 8 December 2022; pp. 111–118.
36. de Oliveira, L.; Rodrigo, A.L. Repurposing decoder-transformer language models for abstractive summarization. *arXiv* **2019**, arXiv:1909.00325.
37. Rothe, S.; Narayan, S.; Severyn, A. Leveraging pre-trained checkpoints for sequence generation tasks. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 264–280. [CrossRef]
38. Jo, S.G.; Park, S.H.; Kim, J.J.; On, B.W. Learning cluster patterns for abstractive summarization. *IEEE Access* **2023**, *11*, 146065–146075. [CrossRef]
39. Kalamkar, P.; Agarwal, A.; Tiwari, A.; Gupta, S.; Karn, S.; Raghavan, V. Named entity recognition in indian court judgments. *arXiv* **2022**, arXiv:2211.03442.
40. Polsley, S.; Jhunjhunwala, P.; Huang, R. Casesummarizer: A system for automated summarization of legal texts. In Proceedings of the COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations, Osaka, Japan, 11–16 December 2016; pp. 258–262.
41. Bhattacharya, P.; Poddar, S.; Rudra, K.; Ghosh, K.; Ghosh, S. Incorporating domain knowledge for extractive summarization of legal case documents. In Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law, São Paulo, Brazil, 21–25 June 2021; pp. 22–31.
42. Khandelwal, U.; Levy, O.; Jurafsky, D.; Zettlemoyer, L.; Lewis, M. Generalization through memorization: Nearest neighbor language models. *arXiv* **2019**, arXiv:1911.00172.
43. Borgeaud, S.; Mensch, A.; Hoffmann, J.; Cai, T.; Rutherford, E.; Millican, K.; Van Den Driessche, G.B.; Lespiau, J.B.; Damoc, B.; Clark, A.; et al. Improving language models by retrieving from trillions of tokens. In Proceedings of the International Conference on Machine Learning. PMLR, Baltimore, MD, USA, 17–23 July 2022; pp. 2206–2240.
44. Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.t.; Rocktäschel, T.; et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 9459–9474.
45. Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv* **2023**, arXiv:2307.09288.
46. Bi, X.; Chen, D.; Chen, G.; Chen, S.; Dai, D.; Deng, C.; Ding, H.; Dong, K.; Du, Q.; Fu, Z.; et al. Deepseek llm: Scaling open-source language models with longtermism. *arXiv* **2024**, arXiv:2401.02954.
47. Chaplot, D.S. Albert q. jiang, alexandre sablayrolles, arthur mensch, chris bamford, devendra singh chaplot, diego de las casas, florian bressand, gianna lengyel, guillaume lample, lucile saulnier, lélio renard lavaud, marie-anne lachaux, pierre stock, teven le scao, thibaut lavril, thomas wang, timothée lacroix, william el sayed. *arXiv* **2023**, arXiv:2310.06825.
48. Tunstall, L.; Beeching, E.; Lambert, N.; Rajani, N.; Rasul, K.; Belkada, Y.; Huang, S.; Von Werra, L.; Fourrier, C.; Habib, N.; et al. Zephyr: Direct distillation of lm alignment. *arXiv* **2023**, arXiv:2310.16944.

49. Dettmers, T.; Pagnoni, A.; Holtzman, A.; Zettlemoyer, L. Qlora: Efficient finetuning of quantized llms. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 10088–10115.
50. Han, Z.; Gao, C.; Liu, J.; Zhang, J.; Zhang, S.Q. Parameter-efficient fine-tuning for large models: A comprehensive survey. *arXiv* **2024**, arXiv:2403.14608.
51. Hu, E.J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. Lora: Low-rank adaptation of large language models. *ICLR* **2022**, *1*, 3.
52. Lin, C.Y. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop, Barcelona, Spain, 25 July 2004*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2004; pp. 74–81.
53. Banerjee, S.; Lavie, A. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Ann Arbor, MI, USA, 29 June 2005; Association for Computational Linguistics: Stroudsburg, PA, USA, 2005; pp. 65–72.
54. Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K.Q.; Artzi, Y. Bertscore: Evaluating text generation with bert. *arXiv* **2019**, arXiv:1904.09675.
55. Bai, Y.; Jones, A.; Ndousse, K.; Askell, A.; Chen, A.; DasSarma, N.; Drain, D.; Fort, S.; Ganguli, D.; Henighan, T.; et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv* **2022**, arXiv:2204.05862.
56. Pires, T.; Schlinger, E.; Garrette, D. How multilingual is multilingual BERT? *arXiv* **2019**, arXiv:1906.01502.
57. Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; Stoyanov, V. Unsupervised cross-lingual representation learning at scale. *arXiv* **2019**, arXiv:1911.02116.