Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

GraphRAG-Based Indian Legal Question Answering System

Amit Jadhav
Department of Computer Science and Engineering
MIT World Peace University
Pune, India

Amit Gholasgaon
Department of Computer Science and Engineering
MIT World Peace University
Pune, India

Janavi Shinde
Department of Computer Science and Engineering
MIT World Peace University
Pune, India

Prasad Kokate
Department of Computer Science and Engineering
MIT World Peace University
Pune, India

Aryan Patil
Department of Computer Science and Engineering
MIT World Peace University
Pune, India

Dr. Jayshree Aher
Department of Computer Science and Engineering
MIT World Peace University
Pune, India

***Abstract* - Answering legal questions in the Indian legal domain poses a challenge due to long judgements, complex legal terminology and the complex relations between cases, laws, judges and parties involved. Traditional Retrieval-Augmented Generation (RAG) methods rely mainly on semantic similarity and therefore lack explicit relation between elements. This paper evaluates the effectiveness of a knowledge graph enhanced RAG approach to answer legal questions. A RAG approach is used where additional structured information is retrieved from a pre-existing legal knowledge graph. 589 Indian Supreme Court judgements were used in 2016 alongside a knowledge graph with 4462 nodes and 8363 edges. Both traditional RAG and knowledge-graph-enhanced RAG were evaluated using 500 legal questions related to 356 cases. As both methods included the same retrieval step, they achieved the same results during retrieval. Precision@5 was 10.36% and Recall@5 was 51.80%. The knowledge-graph-enhanced approach achieved marginally better results with answer relevance being 36.37% compared to 35.50%. The knowledge graph enhanced method had a much higher context support score (46.21%) than the traditional approach (29.86%). This could be due to the extra information being retrieved from the graph and longer generation context. There was no evidence to show the knowledge-graph-enhanced approach had better retrieval results, legal correctness, claim faithfulness or citation accuracy. Further testing is required with specific context limits and expert evaluation to determine if it is effective.**

Keywords - GraphRAG, Retrieval Augmented Generation, Legal Question Answering, Knowledge Graph, Large Language Models, Indian Legal NLP

I. INTRODUCTION

A. Background and Motivation

Legal Question Answering (LQA) is a specialized task that attempts to return relevant answers to questions based upon the information contained within judicial documents and statutory sources. However, this is a challenging task in the Indian jurisdiction as legal judgments tend to be verbose, filled with legal jargon, and mention various cases, Acts, sections, judges, and principles of law. Therefore, finding relevant evidence requires more than just keyword matching between a question and a document.

While Large Language Models (LLMs) are capable of producing fluent responses to legal queries, they have a tendency to produce answers that contain hallucinations when they lack sufficient grounding in the provided documents. Retrieval-Augmented Generation (RAG) overcomes this limitation by retrieving supporting passages from an external corpus and using them to inform the answer. Previous works have shown that retrieval-based methods are crucial to achieving strong performance on legal question answering tasks [1], [2].

Most previous RAG systems make use of vector embeddings to find semantically similar passages. While this method is adequate at retrieving relevant judgment content, it lacks explicit modeling of relationships between different entities. For instance, a question about which section of law was cited in a particular case or finding similar cases that the same statute cannot be retrieved merely by finding similar vectors in the embedding space.

Graph Retrieval Augmented Generation (GraphRAG) enhances standard RAG by incorporating information from a knowledge graph. The hybrid nature of this architecture allows for both semantic similarity based retrieval as well as relational inference.

B. Problem Statement

Indian legal documents contain various relationships between cases, Acts, sections, judges, and parties. Traditional vector-based RAG methods are not able to explicitly model these relationships, limiting the ability of the language model to effectively utilize the retrieved information. In this work, we attempt to build a GraphRAG based Indian Legal Question Answering system that utilizes both semantic similarity and graph relationships to enhance answer generation with evidence from relevant judgments.

C. Research Contributions

We summarize the main contributions of this work as follows:

1. Enhancing a Retrieval-Augmented Generation pipeline by augmenting Vector Search results of

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

Supreme Court Judgment passages with knowledge base information related to legal entities.

2. Compare the performance of Traditional RAG and Knowledge Graph Augmented RAG on 500 legal questions belonging to 356 cases in Indian Supreme Court.
3. Understanding the performance using the evaluation metrics based on: Retrieval Accuracy, Answer Relevance, Automatic Contextual Support, Source Accuracy and Response Time along with shortcomings of the evaluation metrics.

We focus our experiments on a corpus of 589 Indian Supreme Court judgments related to the year 2016 and investigate the extent to which knowledge graph augmentation can enhance legal answer generation.

II. RELATED WORK

The recent literature on legal question answering explores retrieval-augmented generation, domain-specific language models, knowledge graphs, and hybrid retrieval approaches in the context of legal documents processing.

AILQA [1] addresses artificial intelligence for answering legal questions in the Indian judicial space, highlighting retrieval and generation approaches' effectiveness and emphasizing the need for evidence-based answers. LegalBench-RAG [2] focuses on empirical evaluation of the effectiveness of retrieval in legal question answering, with particular attention to the need for finding concise and relevant supporting passages. The Indian legal question-answering dataset described in [3] comprises 10,000 question-answer pairs collected from 1,256 Supreme Court judgments, which serves as a benchmark for legal QA systems.

Other works focus on the retrieval-based legal assistant paradigm, which uses semantic embeddings and vector search to find relevant documents. For example, the paper Democratizing Legal Knowledge in India Using Retrieval-Augmented Generation [4] presents an investigation of the retrieval paradigm in the context of a large body of legal documents. A Hybrid RAG-LLaMA Framework for Scalable and Accurate Interpretation of Legal Texts [5] describes a similar approach to the one described in the previous paragraph but focuses on the Indian tax-law domain and explores hybrid retrieval, temporal filtering, and citation-aware answer generation.

Graph-based retrieval has also attracted the attention of the research community as a promising approach to exploring document relationships beyond text similarity. GRAG [6] presents a graph-based retrieval method for knowledge-intensive generation, and the survey [7] describes graph construction, indexing, retrieval, and generation. Knowledge graph-guided retrieval methods [8] and PROPEX-RAG [9] are examples of approaches that combine language models and knowledge graphs.

Several papers focus on utilizing knowledge graphs in the Indian legal domain. NyayGraph [14] explores the application of knowledge graphs to statute-related tasks, and InLegalLLaMA [15] is a language model that incorporates relevant legal knowledge. Constructing knowledge graphs from Indian legal document collections [16] is another work that describes the process of building a knowledge graph.

The most similar works are Domain-Partitioned Hybrid RAG for Legal Reasoning [17] and NyayaRAG [18], which combine various levels of structured information with retrieval and language models. The former focuses on entity partitioning and demonstrates the effectiveness of the proposed approach on the NyayaH5 dataset, and the latter describes an end-to-end system for legal reasoning with hybrid retrieval.

Previous works have explored legal QA using techniques like retrieval augmented generation, knowledge graphs and hybrid retrieval. However, due to varying datasets, retrieval strategies and evaluation protocols, comparison becomes difficult. In this work, we examine the impact of knowledge graph assisted context enrichment by using a unified QA task on Indian Supreme Court. There are a number of prior works that investigate the use of retrieval augmented generation, knowledge graphs and hybrid retrieval in legal domains. The data, retrieval methods and evaluation processes can vary across works. In this work, we investigate the impact of incorporating knowledge graph information into vector retrieved passages within a common QA framework for Indian Supreme court. This aspect will be addressed in this paper, which compares the performance of the GraphRAG and Traditional RAG approaches in a unified experiment setting.

III. PROPOSED METHODOLOGY

A. System Architecture

The proposed GraphRAG framework has incorporated semantic retrieval and knowledge graph-based evidence enrichment for Indian legal question answering. The system involves two types of retrieval components. The first one is responsible for retrieving the relevant text passages from the given Supreme Court judgments using the embeddings. The second component provides connected entities and relations from the constructed knowledge graph.

For answering each question, the retrieved relevant judgment passages and the related entities are extracted. The knowledge graph information helps in enriching the retrieved context by including information about related cases, judges, parties, Acts, sections, and legal topics. The enriched context is then combined with the retrieved judgment passages to obtain an answer by using the Large Language Model (LLM).

The system also implements a Traditional RAG method that only considers the vector-based retrieval of textual evidence. By performing the comparison of methods, the contribution of using graph-based information can be analyzed.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

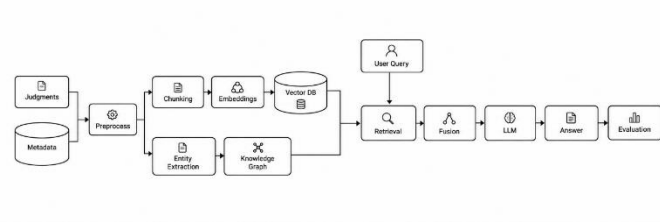

Fig. 1. Overall architecture of the proposed GraphRAG-based legal question answering system.

B. Dataset Description

The current study has utilized the legal knowledge graph that was constructed from 589 Indian Supreme Court judgments associated with 2016. The graph consists of 4,462 nodes and 8,363 edges representing the legal entities and their connections. The majority of nodes are related to cases, judges, parties, Acts, and legal topics.

The graph includes the connections between cases and their related legal topics, Acts and sections, involved parties, and deciding judges. The corresponding corpus of Supreme Court judgments was used for semantic retrieval to identify relevant text passages for the questions. As a result, the question-answering system was able to perform both structured (knowledge graph) and unstructured (judgment text) information retrieval.

The evaluation dataset includes 500 legal questions related to 356 Supreme Court of India case laws. The same question dataset was used for comparing the performance of TRAG and knowledge-graph-enhanced RAG.

C. Data Preprocessing and Text Preparation

Legal judgment documents are preprocessed in order to improve the quality of retrieval. As part of the preprocessing, unnecessary whitespace, repeated headers and footers, and OCR errors are removed when possible, while retaining legally relevant information.

The judgment documents are split into overlapping text chunks, each of which contains metadata about which case and document the text came from, allowing passages that have been retrieved to be linked back to the originating judgment.

The text chunks are then converted into dense vector embeddings using a sentence-embedding model and stored in a vector index, allowing similarity-based retrieval of the chunks given a query.

Legal entities and metadata are normalized to enable matching between the retrieved judgment passage and the knowledge graph.

D. Legal Knowledge Graph

A knowledge graph is a graph that represents legal entities as nodes and their relationships as edges. In mathematical terms it can be defined as

$$G = (V, E) \qquad (1)$$

where V is a set of legal entities and E is a set of relationships.

There are currently 5 main types of node in the legal knowledge graph: Case, Judge, Party, Act and Topic. The relationship between these nodes describes how judges decided the case, which parties were involved, which Acts were cited and the related legal topics. Legal Sections are not present as nodes in the graph. They are instead used as attributes on the relationships between Cases and Acts. In addition, the relationships of the graph store information such as which judges heard which cases, which parties were involved in which cases, what Acts were cited in the judgments of cases, and which topics were discussed.

For instance one case may cite a particular Act and be decided by a particular judge. These relationships allow us to retrieve related information from the graph that may not be captured by semantic similarity alone.

The current implementation makes use of the existing legal knowledge graph for the enrichment of the retrieved evidence. The knowledge graph consists of five main types of entities: Case, Judge, Party, Act and Topic. Relationships between these entities contain data on who decided the case, who was involved in the case, what Acts were referenced and what legal topics are related. Legal sections referenced are represented as attributes of relationships between Cases and Acts. The graph represents legal entities and their relationships in a structured form, allowing relevant information to be associated with judgment passages retrieved through semantic search.

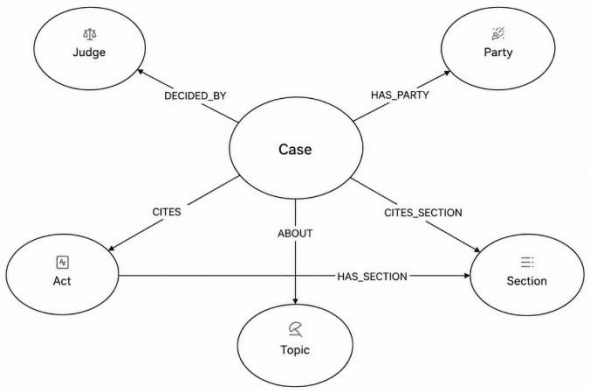

Fig. 2. Legal knowledge graph schema that illustrates the relationships between cases and legal entities.

E. Traditional RAG Baseline

The Traditional RAG is implemented as a baseline for comparison purposes. It performs a passage retrieval task based on the semantic similarity between a user query and the judgment chunks. Let us define the user query as Q and each judgment chunk as D_i, then the calculation of their embeddings can be represented as:

$$q = f(Q),\ d_i = f(D_i) \qquad (2)$$

where f stands for the embedding function.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

The cosine similarity is applied to calculate how relevant each retrieved document is to the query by taking the dot product of the query embedding and the document embedding and dividing it by the product of their norms:

$$S_{ve}\square(q, d_i) = (q \cdot d_i) / (\|q\| \|d_i\|) \quad (3)$$

The top-K most relevant judgment chunks are retrieved and provided as context to the language model. The language model then proceeds to generate an answer based on the retrieved evidence.

F. GraphRAG Retrieval and Context Enrichment

The GraphRAG approach, as an extension of Traditional RAG, takes advantage of information from the knowledge graph to improve the retrieval and answer-generating process.

First, the system retrieves relevant judgment chunks using vector search. Then, the legal entities related to the retrieved judgment chunks are matched with entities stored in the knowledge graph. Subsequently, related graph information is retrieved and employed to enrich the textual context and yield more relevant information.

A general k-hop neighborhood of an entity vq is defined as:

$$N_k(vq) = \{v \in V \mid dist(vq, v) \leq k\} \quad (4)$$

where k is the maximum graph traversal distance.

The framework proposed in this project also allows for a combination of vector and graph relevance scores via a weighted fusion function:

$$Sfinal = \alpha Svec + \beta Sgraph \quad (5)$$

subject to:

$$\alpha + \beta = 1, \ \alpha, \beta \in [0, 1]. \quad (6)$$

In this case, Svec represents vector-based relevance, Sgraph represents graph-based relevance, while α and β are their respective weights.

We enrich the context using facts from the knowledge graph. By linking the retrieved judgment texts to their respective legal entities, we add the relevant relationships in the knowledge graph to the input provided to the language model. We do not change the order of the documents in our retrieval. Graph-aware re-ranking, weighted score combination, and multi-hop graph navigation are not evaluated in our implementation. However, they can be promising ways to improve our application in the future.

G. Answer Generation

The final context comprises information retrieved using semantic search and enriched with knowledge graph data:

$$C = Cvec \cup Cgraph \quad (7)$$

with Cvec representing the retrieved judgment passages and Cgraph being the corresponding knowledge graph evidence.

The final answer is subsequently generated as:

$$A = LLM(Q, C) \quad (8)$$

where Q is the posed legal question, C is the context, and A is the resulting answer. The language model is prompted to utilize the provided evidence when responding to legal inquiries. Additionally, the retrieval process preserves source information to facilitate references to relevant judgments and legal entities.

By combining textual information with a structured knowledge graph, the framework aims to enhance the contextuality and evidential support of legal answers.

IV. EXPERIMENTAL SETUP

A. Implementation Environment

The traditional RAG and GraphRAG pipelines are implemented through Python natural language processing and retrieval components. The pipeline involves text preprocessing, embedding generation, vector similarity search, and language-model-based answer generation.

The traditional RAG baseline is implemented on the semantic retrieval of relevant sections of text from the indexed legal corpus. The GraphRAG approach extends the traditional RAG pipeline with additional steps involving structured information from the legal knowledge graph.

The experimental comparison includes the measurement of the contribution of graph-based context enrichment while holding the evaluation settings constant across both systems.

B. Evaluation Dataset and Procedure

The experimental evaluation reported in this study consists of 500 legal questions associated with 356 cases. Both traditional RAG and GraphRAG pipelines are evaluated using the same question set.

For each question, the system retrieved relevant legal information and produced an answer. The retrieved information compared to source documents associated with each question, while answers were evaluated using reference-answer similarity and context-support measurements.

Evaluation considered both the retrieval effectiveness and the quality of generated answers to reflect the influence of knowledge graph information.

C. Evaluation Metrics

Retrieval effectiveness is quantified using Precision@K and Recall@K. Precision@K represents the ratio of relevant documents retrieved in response to a query, while Recall@K reflects the ratio of relevant documents retrieved as a fraction of the total relevant documents for the given ground truth.

$$Precision_K = \frac{Number\ of\ relevant\ documents\ in\ Top-K}{K} \quad (9)$$

$$Recall_K = \frac{Number\ of\ relevant\ documents\ retrieved\ in\ Top-K}{Total\ number\ of\ relevant\ documents} \quad (10)$$

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (11)$$

Answer relevance is quantified using semantic similarity against the reference answer.

Context support is defined by semantic correspondence analysis of generated answers and the content of retrieved supporting information. This measure approximates answer grounding, but does not substitute for an explicit claim-level faithfulness assessment.

The source retrieval accuracy evaluates the presence of the ground truth source document among the retrieved results. It is important to distinguish this measurement from evaluation of the legal correctness of sources cited in the generated answer.

Average response time is measured as a proxy of computational performance of the two approaches during answer generation.

V. RESULTS AND DISCUSSION

A. Comparative Analysis of Performance Indicators

The traditional approach and the graph-based method to retrieval-augmented generation were compared in terms of retrieval performance, answer quality, context support, source retrieval accuracy, and response time. The evaluation was conducted using 500 legal questions associated with 356 cases.

Evaluation Metric	Traditional RAG	GraphRAG
Precision@K	10.36%	10.36%
Recall@K	51.80%	51.80%
Answer Relevance	35.50%	36.37%
Context Support	29.86%	46.21%
Source Retrieval Accuracy	51.80%	51.80%
Average Response Time	2.853 s	1.90 s

The results demonstrate that both methods have identical retrieval performance with a Precision@K of 10.36% and a Recall@K of 51.80%. In other words, the utilization of graph-based context did not impact the set of relevant source documents retrieved in the evaluated configuration.

GraphRAG reached an answer relevance score of 36.37%, as compared to 35.50% in the case of Traditional RAG. Therefore, the graph-based method contributed to a slight increase in semantic correspondence with the reference answers.

A more significant difference was observed in terms of context support, with 46.21% in GraphRAG as compared to 29.86% in Traditional RAG. The generated legal judgments are the same. However, in addition to using the same context, we have enriched the context of the model with additional legal facts in the form of a knowledge graph. This might explain why this method achieved better automated context support. However, the graph enriched model also had a longer context, therefore we cannot be certain that it has greater faithfulness and legal correctness.

The observed average response times were 2.853 seconds for Traditional RAG and 1.90 seconds for knowledge-graph-enriched RAG. It should be noted that these figures only illustrate the observed execution time in the reported experiment, which cannot be expanded to general graph-based retrieval methods.

B. Discussion of Performance Differences

Overall, the results of the experiment demonstrate that the application of knowledge graph enrichment can improve the context support of answers generated on the basis of retrieved legal documents. Notably, such an improvement was more significant in this regard than answer relevance, which implies that additional information contributes to contextual grounding without transforming the overall meaning of generated text to the same extent.

The identical retrieval precision and recall values are consistent with the current implementation methodology, within which knowledge graph information supplements vector retrieved passages rather than modifying document retrieval rankings directly. As a result, further improvements in this regard would require the implementation of graph-aware reranking or multi-hop graph traversal.

In turn, the evaluation has several limitations, including the utilization of a restricted set of Supreme Court judgments and the possibility of OCR or entity extraction errors impacting retrieval performance. Similarly, automatic semantic similarity assessment and context-support measurement do not guarantee the legal correctness of generated answers. Accordingly, the results of the experiment should not be considered proof of the complete legal reliability of graph augmented answers but rather evidence of the potential contribution of structured information.

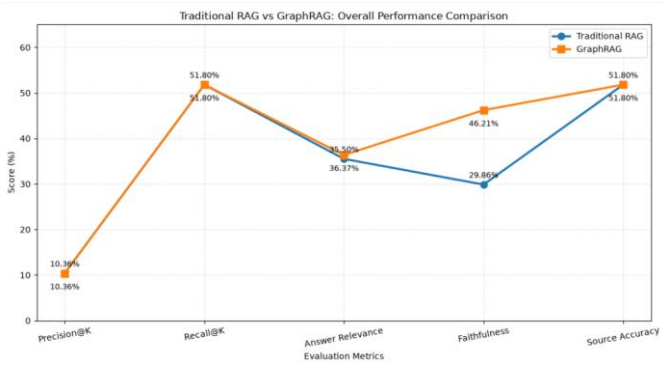

Fig. 3. Performance comparison of Traditional RAG and GraphRAG across the evaluated metrics.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

VI. CONCLUSION AND FUTURE WORK

In this study we have explored how to enhance context through knowledge graphs for Retrieval Augmented Question Answering tasks using Indian Supreme Court decisions. We augmented vector-based retrieved decision text with additional structured knowledge about entities relevant to the legal domain. During evaluation, Traditional RAG and Knowledge Graph-enhanced RAG models had identical retrieval results, both achieving a Precision@5 and Recall@5 of 10.36% and 51.80% respectively, as both were using the same retrieval method. In terms of answer relevance, Knowledge Graph-enhanced RAG models showed slightly better results (35.50% increased to 36.37%). Automated scores for contextual support improved from 29.86% to 46.21%. However, whilst there is evidence that Knowledge Graph-enhanced RAG models produce different answers with more alignment to the retrieved context, there is no evidence that they are producing more correct answers from a legal standpoint, nor that they provide answer faithfulness at the claim level. Further, it is difficult to tell the extent to which the graph knowledge is influencing the results over just providing extra context. Future work will focus on comparing models with limited context budgets, graph aware re-ranking, multi-hop reasoning, multilingual legal question answering, and asking experts to evaluate the accuracy and citing support of answers.

REFERENCES

[1] S. K. Nigam, S. K. Mishra, N. Shallum, K. Ghosh, and A. Bhattacharya, "AILQA: Evaluating AI-driven legal question answering systems for the Indian legal system," *Artificial Intelligence and Law*, 2026, doi: 10.1007/s10506-026-09537-2.

[2] N. Pipitone and G. Houir Alami, "LegalBench-RAG: A benchmark for retrieval-augmented generation in the legal domain," *arXiv preprint arXiv:2408.10343*, 2024, doi: 10.48550/arXiv.2408.10343.

[3] K. Veningston and A. Mishra, "Dataset for legal question answering system in the Indian judiciary context," *Data in Brief*, vol. 60, Art. no. 111647, 2025, doi: 10.1016/j.dib.2025.111647.

[4] A. P. Singh, M. P. Singh, M. Sharma, L. Dang, and A. Gera, "Democratizing legal knowledge in India using retrieval-augmented generation: A scalable large-corpus framework for legal question answering," *SSRN*, 2026.

[5] V. R. Karna, R. R. M, B. S. Babu, N. S, M. M, and H. V, "A hybrid RAG-LLaMA framework for scalable and accurate interpretation of legal texts," *Applied Artificial Intelligence*, vol. 40, no. 1, Art. no. e2626097, 2026, doi: 10.1080/08839514.2026.2626097.

[6] Y. Hu, Z. Lei, Z. Zhang, B. Pan, C. Ling, and L. Zhao, "GRAG: Graph retrieval-augmented generation," *arXiv preprint arXiv:2405.16506*, 2024.

[7] B. Peng, Y. Zhu, Y. Liu, X. Bo, H. Shi, C. Hong, Y. Zhang, and S. Tang, "Graph retrieval-augmented generation: A survey," *arXiv preprint arXiv:2408.08921*, 2024.

[8] X. Zhu, Y. Xie, Y. Liu, Y. Li, and W. Hu, "Knowledge graph-guided retrieval augmented generation," in *Proc. 2025 Conf. Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Vol. 1*, Albuquerque, NM, USA, 2025, pp. 8912–8924, doi: 10.18653/v1/2025.naacl-long.449.

[9] T. Sarnaik, M. Shah, and R. Hegde, "PROPEX-RAG: Enhanced GraphRAG Using Prompt-Driven Prompt Execution," in Pattern Recognition and Machine Intelligence (PReMI 2025), Lecture Notes in Computer Science, vol. 16357, Springer, Cham, pp. 200–207, 2026, doi: 10.1007/978-3-032-18477-1_21.

[10] M. Tadayon and M. Gupta, "Graphs RAG at scale: Beyond retrieval-augmented generation with labeled property graphs and Resource Description Framework for complex and unknown search spaces," *arXiv preprint arXiv:2603.22340*, 2026.

[11] J. Lelong, A. Errazine, and A. Blangero, "Agentic RAG with knowledge graphs for complex multi-hop reasoning in real-world applications," *arXiv preprint arXiv:2507.16507*, 2025.

[12] J. Bose, "Falkor-IRAC: Graph-constrained generation for verified legal reasoning in Indian judicial AI," *arXiv preprint arXiv:2605.14665*, 2026.

[13] Z. Chen, Q. Zhang, Z. Xiang, Z. Wei, L. Gao, X. Huang, Z. Zhang, and J. Su, "LegalGraphRAG: Multi-agent graph retrieval-augmented generation for reliable legal reasoning," in *Proc. 64th Annual Meeting of the Association for Computational Linguistics (ACL), Vol. 1: Long Papers*, San Diego, CA, USA, 2026, pp. 37455–37484, doi: 10.18653/v1/2026.acl-long.1738.

[14] S. Shukla, T. Tyagi, A. S. Bisht, A. Sharma, and B. Agarwal, "NyayGraph: A knowledge graph enhanced approach for legal statute identification in Indian law using large language models," in *Proc. Natural Legal Language Processing Workshop (NLLP 2025)*, Suzhou, China, 2025, pp. 147–156, doi: 10.18653/v1/2025.nllp-1.11.

[15] S. Ghosh, D. Verma, B. Ganesan, P. Bindal, V. Kumar, and V. Bhatnagar, "InLegalLLaMA: Indian Legal Knowledge Enhanced Large Language Models," IJCAI 2024, 2024.

[16] S. Jain, P. Harde, N. Mihindukulasooriya, S. Ghosh, A. Dubey, and A. Bisht, "Constructing a knowledge graph from Indian legal domain corpus," in *Proc. ESWC 2022 Workshops*, 2022.

[17] R. Goel, S. P. Kumar, A. Agrawal, D. Poddar, P. Narang, and D. Kumar, "Domain-partitioned hybrid RAG for legal reasoning: Toward modular and explainable legal AI for India," *arXiv preprint arXiv:2602.23371*, 2026, doi: 10.48550/arXiv.2602.23371.

[18] S. K. Nigam, B. D. Patnaik, S. Mishra, A. V. Thomas, N. Shallum, K. Ghosh, and A. Bhattacharya, "NyayaRAG: Realistic legal judgment prediction with RAG under the Indian common law system," *arXiv preprint arXiv:2508.00709*, 2025.

[19] V. Vishnuprabha, D. M. Viswanathan, R. Rajesh, and A. V. Pillai, "Augmented question-guided retrieval (AQgR) of Indian case law with LLM, RAG, and structured summaries," *arXiv preprint arXiv:2508.04710*, 2025.

[20] S. Knollmeyer, O. Caymazer, and D. Grossmann, "Document GraphRAG: Knowledge graph enhanced retrieval augmented generation for document question answering within the manufacturing domain," *Electronics*, vol. 14, no. 11, Art. no. 2102, 2025, doi: 10.3390/electronics14112102.

[21] VaidhyaMegha, "Indian Supreme Court Judgments Knowledge Graph," Hugging Face Datasets, version 1.0, 2026. Source data: Indian Supreme Court Judgments, curated by Dattam Labs, AWS Open Data Registry, licensed under CC BY 4.0.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)