A Novel Evaluation Strategy for Domain-Specific Legal QA: Application to Bangladeshi Law with RAG

By

Md Shahriar Asfaq

ID: 0122410007

Submitted in partial fulfilment of the requirements

of the degree of

Master of Science in Computer Science and Engineering

January 17, 2026

Department of Computer Science and Engineering

United International University

Dhaka, Bangladesh

Board of Examiners

1. **Supervisor:** Mr. Nahid Hossain, Assistant Professor, Department of CSE, United International University

2. **Head Examiner:** Prof. Dr. Mohammad Nurul Huda, Professor, Department of CSE, United International University"

Declaration

This is to certify that the work entitled 'A Novel Evaluation Strategy for Domain-Specific Legal QA: Application to Bangladeshi Law with RAG' is the outcome of the project carried out by me under the supervision of 'Mr. Nahid Hossain, Assistant Professor'.

Md Shahriar Asfaq

MSCSE Program

Student ID: 0122410007

Dept. of Computer Science and Engineering

United International University

Dhaka, Bangladesh

In my capacity as supervisor of the candidate's project, I certify that the above statements are true to the best of my knowledge.

Mr. Nahid Hossain

Assistant Professor

Department of Computer Science and Engineering

United International University

Abstract

Retrieval-Augmented Generation, or RAG, has been identified as a promising approach for question answering tasks, particularly for high-risk fields such as law, for which accuracy, interpretability, and contextuality are essential. Current evaluation approaches for RAG models are general natural language evaluation scores, which do not embody specific legal reasoning attributes such as temporal validity, jurisdictional validity, procedural adequacy, or reconciliation of numerical constraints. This paper introduces a legal-aware evaluation framework for RAG models, and we build an AI legal assistant for Bangladeshi laws. We introduce a novel evaluation framework with seven complementary metrics: (1) Gold Answer baseline correctness, (2) BERTScore semantic similarity, (3) Numeric Accuracy, (4) Temporal Validity, (5) LLM Judge reasoning quality, (6) Legal Validity statute checking, and (7) Robustness. Evaluation results show our legal-aware framework is successful and measures key unexplored failing dimensions for quality evaluation, giving a CLR score of 0.824 on 20 test queries under the Baseline configuration, with perfect scores on Temporal Validity (1.00) and Legal Validity (1.00), and strong BERTScore semantic similarity (0.861), and improved Numerical Accuracy following word-form normalization of Bangladeshi legal quantity expressions, confirming both system reliability and the framework's ability to surface domain-specific failure modes invisible to standard NLP evaluation.

Acknowledgements

All praise and thanks are due to Allah, the most gracious, the most merciful. HE has gave me the strength, courage, and patience to undertake this task. It took time not only from my side, but also from my supervisor's one. Mr. Nahid Hossain, Assistant Professor at the Department of Computer Science and I am writing to apply for a place in the School of Engineering in United International University. First of all, let me express my deep sense of gratitude toward I would like to thank my supervisor for all the support he gave me while doing the research work. Special thanks to my family, friends, and colleagues who have been of immense support. All of them have supported me in one way or another during the preparation of this work.

Publications

The main contributions of this project are either published or accepted or in preparation in journals and conferences as mentioned in the following list:

Journal Articles

Conference Papers

Contents

List of Figures

List of Tables

1 Introduction

This is an overview of the research problem, objectives, and significance of developing an AI-based legal assistance system for Bangladesh using Retrieval-Augmented Generation (RAG) technology.

1.1 Background and Motivation

In Bangladesh, there are numerous barriers to accessing and acquiring legal knowledge on the part of the general public due to the complexity of legal jargon, the absence of legal knowledge, and resource limitation. This is true from the working class to small businesses, as the general public is not able to grasp their legal rights and duties, which results in violation of their rights, exploitation, and deprivation of justice. Although professional legal support from an attorney is out of reach for the general public due to the higher cost involved, sources such as social media are not reliable either. There is an acute need for an easily accessible platform where comprehensible legal knowledge is provided to the vast structure of society in Bangladesh.

The legal field is challenging for several reasons, including the use of specific language, the validity of laws over time, the relationship of laws in a hierarchical structure, and numeric and procedural accuracy. However, recent developments in Large Language Models and Retrieval-Augmented Generation are promising and demand specific testing tools for accurate performance in the high-stakes task of the legal field.

1.2 Problem Statement

The present methods of evaluating the RAG are only dependent on some overall natural language processing standards that do not have the capacity to address the complex needs of legal reasoning. It has a very high demand for a legal environment-specific tool that has the capability of evaluating legal RAGs on the basis of parameters such as temporal validity, procedural validity, calculation accuracy, and statutory consistency. There exists a shortage of readily available AI-based legal help tools that have the capacity to offer

proper and relevant responses according to the laws prevalent in Bangladesh.

1.3 Objectives

The primary focus of the given project involves developing and testing a legal assistance system. For Bangladesh legal system uses Retrieval-Augmented Generation technology with a novel evaluation framework.

Specific Objectives:

1. Design an overall evaluation strategy to gauge legal QA system performance aside from the usual NLP metrics
2. Develop an electronic system for providing legal aid for the laws of Bangladesh that use semantic search and intent-aware retrieval
3. Developing a legal corpus of Bangladeshi statutes through appropriate preprocessing and indexing
4. Develop and apply specific measures of evaluation for legal argumentation such as Temporal Validity and Numerical Accuracy
5. Create mechanisms for expansion of legal terms exclusive to Bangladesh through acronyms maps
6. Testing system performance using traditional as well as legal-specific criteria

1.4 Research Questions

1. In what way can the traditional evaluation criteria of RAG be extended to incorporate domain-specific requirements?
2. What are the special criteria for the evaluation of the reliability of legal question and answer systems?
3. Can the expansion of acronyms and the intention behind them aid the legal information retrieval?

4. How effective is the legal-aware evaluation framework that has been proposed?

1.5 Scope of the Project

This is an application that deals with creating a legal assistance system exclusively designed for Bangladeshi law, with specific interest in areas of criminal law, cyber law, and organizational information law. This application works with a dataset of 59,741 legal documents. This evaluation framework will be generally applicable to legal domain applications, potentially with extension to other specialized application areas. Applications might be restricted to legal texts written in English with a specific Bangladeshi legislative context.

1.6 Significance and Novelty of the Work

This study makes a contribution in the form of a new method for the evaluation of legal RAG systems, covering all important aspects ignored by the existing method for evaluation. The development of the system is an example of the application of RAG technology for legal support in practice, with customized features for expansion by acronyms, identification by intent, and hybrid search.

1.7 Organization of the Report

Chapter 2 presents a literature review of RAG systems and evaluation methodologies. Chapter 3 details the system architecture and design. Chapter 4 describes the implementation and development process. Chapter 5 covers testing and evaluation methodology. Chapter 6 concludes with findings, limitations, and future research directions.

2 Literature Review

This chapter reviews existing research on Retrieval-Augmented Generation systems, legal AI applications, and evaluation methodologies for domain-specific question answering.

2.1 Introduction

The application of Large Language Models in specific fields has brought about changes in many applications, and legal AI holds a lot of promise pertaining to document analysis and legal research and answering questions.[15, 16] However, its application in critical fields such as legal services should be judged through more specialized metrics.[18, 20]

2.2 Overview of Related Technologies

Retrieval-Augmented Generation(RAG) models leverage the knowledge from a pretrained language models incorporating non-parametric knowledge retrieval from external sources. This architecture helps to reduce hallucinations by ensuring responses to the question in the retrieved documents.[7, 13] Key Variants of the RAG Key:

- **Naive RAG**: Simple retrieve and then generate architecture [3, 24]
- **Advanced RAG**: It involves query optimization, reranking, and iterative evaluation [6, 13]
- **Modular RAG**: This incorporates specialized modules for different retrival and generation tasks[9]

Legal domain applications will involve other issues such as citation accuracy, temporal validity, and procedural correctness.

2.3 Review of Similar Projects/Solutions

The existing systems are plagued with hallucination percentages ranging from 17-33%. There are a few techniques that have been recognized to overcome these difficulties:

Iterative RAG Frameworks: The multi-round retrieval model IM-RAG enhances the recall measure to 78.67% by iteratively refining the question and computing the aggregate context, thereby considering the dependency between different clauses.[6, 7]

Knowledge Graph Integration: Systems combining RAG with knowledge graphs facilitate the enhancement of relational reasoning by the identification of interlinked associations such as statutes and case names, and "overruled-by" relationships.[8, 22]

Agentic Systems: The process of integrating multiple agents can be achieved in a number of ways (for example, "Advisors" in statutory interpretation and "Paralegals" for document preparation) to handle the limitations posed by context, as well as for security.[9, 17]

- ARES metrics are domain-agnostic — a response citing a repealed statute scores identically to one citing current law, provided semantic similarity is high. This is the failure mode that motivates the Temporal Validity dimension.
- NitiBench's Multi-HitRate@k measures only whether relevant sections are retrieved, not whether the generated answer correctly applies numerical constraints — a critical gap for penalty and fine-related queries.
- LRAGE does not address the Bangladeshi legal context, nor does it incorporate temporal validity or numerical constraint checking as explicit evaluation dimensions.
- The dual-evaluation approach for position bias mitigation [10] is adopted directly and is not claimed as a novel contribution. The extension lies in applying it within a composite legal reliability scoring system.

2.4 Evaluation Methodologies in Legal RAG

The existing evaluation methods can be classified as: **Multi-Dimensional Semantic Metrics** : Frameworks such as ARES (Automated Evaluation System) tests the result's relevance to context, the faithfulness of the answer to the question employing highly refined LLM judges.[11, 12]

Specialized Legal Retrieval Metrics : NitiBench proposes multi-label metrics like

Multi-HitRate@k and MultiMRR@k which involve the retrieval of all related legal sections rather than single documents.[21]

Automated Judge Frameworks : The method proposed by the LLM-as-a-judge by scoring answers on factual consistency, with techniques such as dual evaluation to counter position bias.[10, 23]

Qualitative Error Taxonomies : The system categorizes error types, including Procedural Errors, Terminological Errors, Contextual Misinterpretations, and Mis to understand root causes.[18, 20]

Traditional NLP Evaluation Metrics : BLEU, ROUGE, METEOR, and Baseline comparisons, but not domain specificity.[13, 19]

2.5 Gap Analysis

Despite these advances, there are still major gaps in the evaluation of RAG:

- **Temporal Validity**: The lack of overall metrics in existing systems for the assessment of if the retrieved laws are current or have been repealed/amended
- **Numerical Constraint Reconciliation**: Poor judgment on the capability of systems to deal with numerical aspects of laws: fines, deadlines, percentages[20]
- **Compounding Error Assessment**: Lack of standard metrics that allow the tracking of error accumulation in multi-step legal reasoning.[10, 23]
- **Counterfactual Robustness**: Insufficient testing on the systems' ability to identify and reject legal provisions that are contradictory or hallucinated.[18, 24]

2.6 Usability and Design Comparison with Existing Systems

In Sections 2.3 and 2.4, it was noted that current legal-RAG evaluation methodologies, such as ARES [11], NitiBench [21], and LRAGE [19], neglect a number of evaluation factors, which include time validity, numerical constraint validation, and corpus adequacy for a particular jurisdiction. Yet an analysis based solely on evaluation measures would be

inadequate because none of these models was developed with the end-user in mind. This sub-section will compare these models on the grounds of usability and interface design, as it becomes a necessary factor in light of the target audience described in Section 1.1, i.e., members of the Bangladeshi general public with insufficient legal knowledge.

Design orientation of previous systems. ARES [11] and NitiBench [21] are benchmarking systems meant to evaluate retrieval-augmented systems based on gold-standard datasets; neither system describes or provides any user interface design, and both rely on the fact that the consumer of their outputs is a researcher analyzing metric results and not a layman asking a legal question. Likewise, LRAGE [19], being an evaluation system for legal RAG pipelines, is not concerned about the issues of query comprehensibility, presentation of answers and usability for non-experts. Agentic systems as well as knowledge graph-based ones discussed in Section 2.3 [8, 9, 17] similarly care about internal reasoning mechanism but not about interface usability. Usability of a non-expert end user interface is not even in the scope of previous research – not that previous systems have poor usability but that this dimension was never part of their design and evaluation criteria.

Why LawRAG's methodology is more user-friendly, precise, and design-oriented. The proposed system differs from this body of work in three specific respects, each tied to a concrete design decision described in Chapter 3:

- Ease of use by query normalization, not user effort. While previous approaches take it for granted that the input queries are phrased using the appropriate legal/technical terms, LawRAG's acronym resolution process and query reformulation based on understanding of the user's intention (Section 3.3.3 3.3.4 respectively) move the responsibility of proper phrasing away from the user onto the system itself. The user who enters "CSA" or "punishment for cyber bullying" need not be aware of the proper names of the Cyber Security Act and its sections at all.

- Precisely through a sequence of verifiable output blocks, and not just one generated block. As described in Section 3.6, the user interface displays the query expansion, the statutory text that is obtained from the database, and the answer that is generated by the model in separate fields. This means that the user or an attorney

helping him or her can check the answer against the actual source text right away, as opposed to relying on some black-box generation. Evaluation-only models cannot make such a distinction apparent to a user at all, as their output is just a score.

- A design-friendly approach via an intentionally constructed, frictionless user interface. This Gradio user interface (see Section 3.6.2) does not necessitate any login accounts, settings changes, or legal/technical jargon, and also provides sample queries as a way of onboarding the user. This is a conscious design choice, aimed at tackling the issues of inaccessibility highlighted in Section 1.1 (the cost of professional legal advice and untrustworthy sources like social media).

The design was done using two well-defined guidelines, not any arbitrary criteria: Heuristics for User Interface Evaluation by Nielsen[26] (visibility of system status, match with the real world, and avoidance of errors) and ISO/IEC 25010 [27] usability subcharacteristics of recognizability and learnability, both of which are not covered in ARES, NitiBench, and LRAGE because these tools emphasize search and ranking capabilities more than interface design. Therefore, it adds the usability aspect to the comparison from Table 6 and makes LawRAG the only evaluated system having an interface designed according to the established criteria, which is essential because it will be pointless to have an evaluation framework with legal knowledge unless it can be used by the target audience.

2.7 Summary

Currently existing literature has shown the advancements made for the legal AI systems but has also shown the existing chasm within the research methodologies. This research bridges the gap by introducing the need for temporal validity, accuracy for numeric questions, and other criteria essential for the domain of the legal question-answering system.

3 System Analysis and Design

This chapter provides the architecture of the proposed legal assistance system.

3.1 System Overview / Architecture

The system has a multi-stage pipeline for Bangladeshi document retrieval and question answering. The system combines semantic search, acronym expansion, intent-aware retrieval, cross-encoder reranking, and answer generation modules.

3.2 Requirement Specification

3.2.1 Functional Requirements

1. The system must be able to process natural language legal inquiries
2. The system shall expand all the legal abbreviations into their full names
3. The system shall be able to classify the intention of queries to enhance the retrieval process
4. Relevant legal documents will be retrieved from the corpus by the system
5. The system will be able to produce correct responses based on the retrieved context
6. The system shall assess answer quality based on more than one factor

3.2.2 Non-Functional Requirements

1. **Performance**: Response time for normal searches.
2. **Accuracy**: Extremely accurate at legal reasoning problems
3. **Scalability**: Handling the expansion of the corpus to over 100,000
4. **Usability**: User-Friendly Interface for Nontechnical user
5. **Reliability**: 95% availability of the system

3.3 System Diagrams

The following diagrams complement the functional and non-functional requirements listed above by showing the system boundary, the flow of data through the pipeline, and the interactions available to the end user.

3.3.1 Context Diagram

Figure 1 shows the LawRAG system as a single process interacting with its external entities: the end user, the Bangladeshi legal document corpus, and the underlying language model used for generation and acronym fallback expansion.

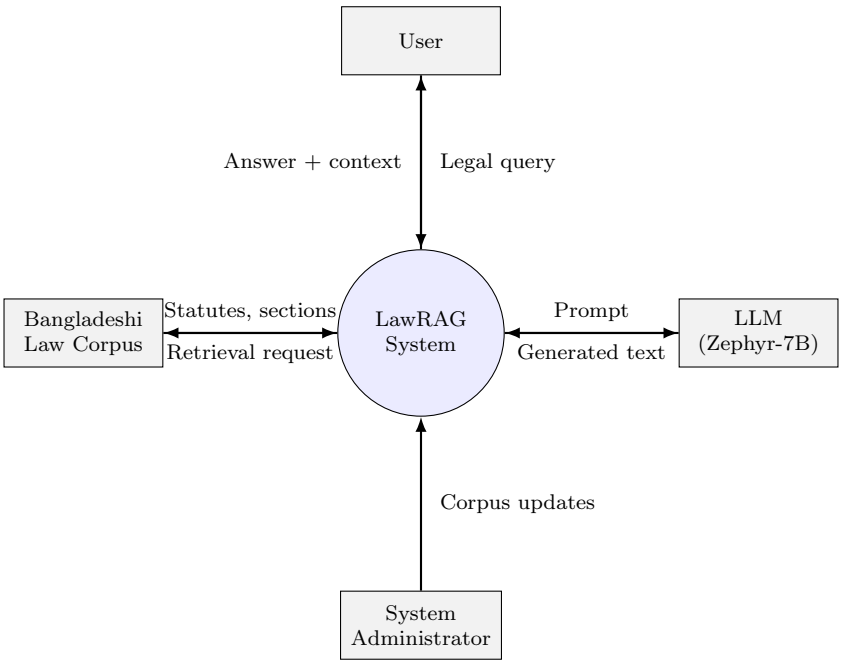

Figure 1: Context diagram of the proposed LawRAG system.

3.3.2 Data Flow Diagram (Level 1)

Figure 2 decomposes the system into its major processing stages, following the pipeline described in Section 3.3.1: acronym expansion, intent detection, hybrid retrieval, reranking, and answer generation, together with the two data stores used during retrieval.

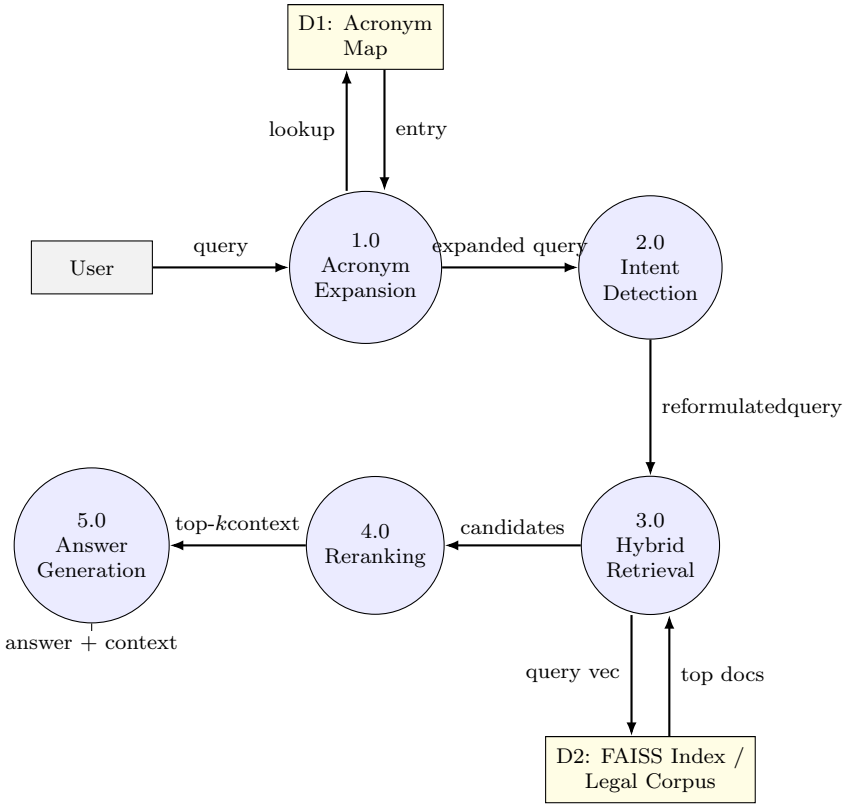

Figure 2: Level-1 Data Flow Diagram of the LawRAG pipeline.

3.3.3 Use Case Diagram

Figure 3 shows the primary interactions available to the two actors identified for the system: the general public user submitting legal queries, and the system administrator responsible for corpus maintenance.

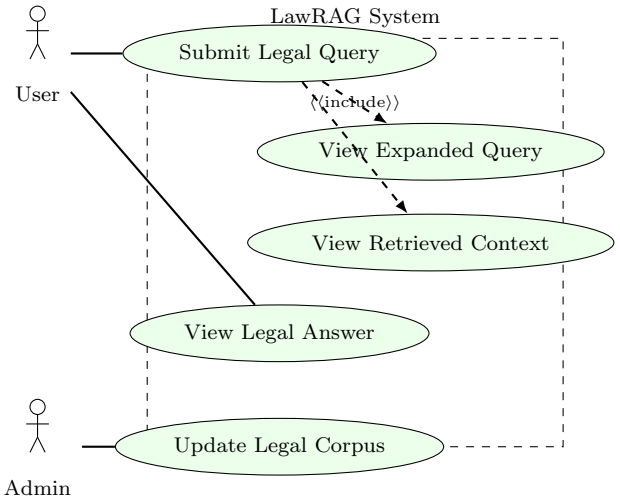

Figure 3: Use case diagram for the LawRAG system.

3.4 Proposed Methodology / Algorithm / Framework

The approach used is hybrid in nature following RAG framework, which is specifically designed for the legal question answering task in the context of Bangladeshi law. The process includes semantic retrieval, lexical match, acronym normalization, intent-based query reformulation, and neural ranking.

3.4.1 Overall Workflow

Figure 4 illustrates the end-to-end processing pipeline of the proposed framework, starting from raw user queries to final answer generation.

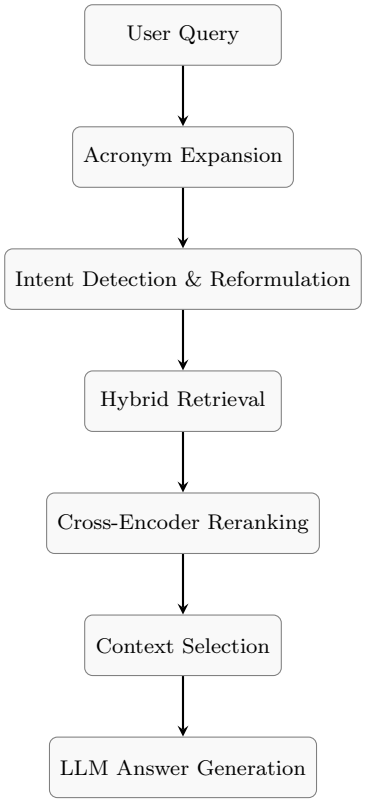

Figure 4: Proposed hybrid Legal-RAG methodology flowchart.

3.4.2 Hybrid Retrieval Mechanism

To balance semantic understanding with statutory precision, the system employs two complementary retrieval strategies:

- **Semantic Retrieval**: Dense embeddings generated using MiniLM-L6-v2 and indexed via FAISS.
- **Lexical Retrieval**: TF-IDF vectorization with unigram and bigram features to preserve exact legal terminology.

Candidate documents from both channels are combined using a weighted score fusion mechanism, followed by neural reranking using a cross-encoder model (ms-marco-MiniLM).

3.4.3 Acronym Expansion Algorithm

Legal and institutional acronyms are resolved prior to retrieval using a hierarchical expansion strategy. The formal procedure is described in Algorithm 1.

Algorithm 1 Acronym Expansion Algorithm

function ExpandAcronym(Q, M, L)
  $P \leftarrow$ regex pattern `r"\textbackslash b[A-Z]\{2,\}\textbackslash b"`
  $Q' \leftarrow Q$
  for all $match \in matches(Q, P)$ **do**
    $A \leftarrow extract(match)$
    if $A \in M$ **then**
      replace($match$, $M[A]$)
    else
      $W \leftarrow WikipediaSearch(A)$
      if $W \neq \emptyset$ **then**
        replace($match$, extractFullForm(W, A))
      else if $L \neq null$ **then**
        replace($match$, LLMExpand(A, Q, L))
      end if
    end if
  end for
  return Q'
end function=0

3.4.4 Intent-Aware Query Reformulation

By using intent-aware query reformulation, Query expansion using the expanded acronyms leads to a normalized query that is classified to determine the following

- Legal domain (murder, cyber crime, rape, etc.)

- Query type (e.g., punishment, procedure, definition)
- Temporal intent (e.g., pre-amendment or current law)

Detected intents can then be used for reformulating the query by highlighting the legally salient terms and thus helping in accurate identification of intent and temporal constraints, thus enhancing the relevance of later retrievals.

3.4.5 Context Retrieval and Ranking

Let S_s denote semantic similarity, S_k keyword similarity, S_d domain relevance, and S_t temporal priority. The final retrieval score is computed as:

$$S_{\text{total}} = \alpha S_s + \beta S_k + \gamma S_d + \delta S_t \tag{1}$$

where α, β, γ, and δ are empirically tuned weights. The top-ranked documents are subsequently reranked using a cross-encoder to obtain the final context set.

3.4.6 Answer Generation

The legal environment is placed within a formulaic response template, in relation to the a deterministic large language model that could produce a response based on this formula, it is compelled to act in accordance with what is stipulated in the legal text provided to it. Thus, this systematic process will equip the proposed system with semantic robustness, lexical exactness, and juristic legality for full end-to-end question and answering capabilities.

3.4.7 Evaluation Framework

In order to render the model legally sound, apart from being technically correct, it undergoes an evaluation based on the multidimensional measure of legal soundness. The multidimensional measure consists of integrating elements such as semantic similarity, numeric consistency, legality of time, and legality of statute into a composite indicator. Figure 5 presents the evaluation workflow.

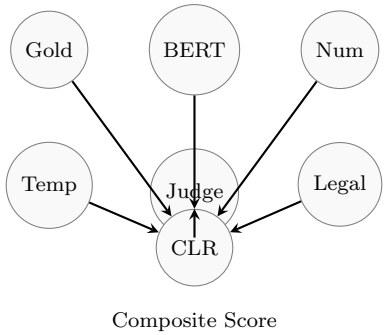

Figure 5: Composite Legal Reliability (CLR) evaluation flow.

3.4.8 Metric Formulations

- **BERTScore**:

$$\text{BERTScore} = \frac{1}{N} \sum F1_{\text{BERT}}(p_i, r_i) \tag{2}$$

- **LLM Judge**:

$$\text{Judge}_{\text{final}} = \frac{J(q, r, p) + J(q, p, r)}{2} \tag{3}$$

- **Numeric Accuracy**:

$$\text{NumAcc} = \frac{1}{|R|} \sum_{r \in R} \mathbb{I}\left(\exists p \in P : \frac{|p - r|}{\max(r, \epsilon)} \leq \delta\right) \tag{4}$$

  where:

  - R = set of reference numbers
  - P = set of predicted numbers
  - δ = tolerance threshold (e.g., 0.05)
  - ϵ = small constant to avoid division by zero
  - $\mathbb{I}(\cdot)$ = indicator function

- **Temporal Validity**:

$$\text{TempVal} = \begin{cases} 1, & \text{if } y_s \leq y_q \leq y_e \\ 1, & \text{if } y_s \leq y_q \text{ and } y_e \text{ is undefined} \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where:

- y_s = start year (year of enactment/effectiveness)
- y_e = end year (effective_to, if applicable)
- y_q = query year (assumed current year if not specified)

Unlike the binary metrics used in evaluating general purpose NLP systems, the Temporal Validity and Numeric Accuracy metrics proposed here are meant to account for the graduated quality of legal validity. A law which is still amended but valid partially is only partially valid. A number answer which falls into the right statutory range should be considered partially correct.

3.4.9 Composite Legal Reliability (CLR) Metric

Let m_i denote individual evaluation metrics and w_i their corresponding weights. The overall reliability score is computed as:

$$\text{CLR} = \sum_{i=1}^{N} w_i \cdot m_i \tag{6}$$

where $N = 7$ in the proposed framework, covering Gold Answer baseline correctness, BERTScore semantic similarity, numerical accuracy, temporal validity, LLM-based judgment, Legal Validity statute checking, and robustness.

3.4.10 Evaluation Algorithm

The complete evaluation procedure is formalized in Algorithm 2.

Algorithm 2 Legal-RAG Evaluation Procedure

0: **function** EVALUATESYSTEM(Q, G, P)
0: $B \leftarrow \text{BERTScore}(P, G)$
0: $N \leftarrow \text{NumericAccuracy}(P, G)$
0: $T \leftarrow \text{TemporalValidity}(P, Q)$
0: $L \leftarrow \text{LegalValidityCheck}(P)$
0: $J \leftarrow \text{LLMJudge}(Q, P, G)$
0: $R \leftarrow \text{RobustnessTest}(Q)$
0: $\text{CLR} \leftarrow \sum w_i \cdot [B, N, T, L, J, R]$
0: **return** CLR
0: **end function**=0

3.4.11 Experimental Setup

Datasets [1] Evaluation experiments are done on a specially compiled dataset of 59,741 Bangladeshi legal documents such as statutes, amendments, and regulations. The manually test set of appropriate legal questions that can be used in evaluation, every pair with expert-verified gold answers. The main domains are: And this is the document statistics:

Table 1: Domain Breakdown of Legal Documents
Note: Evaluation queries target the Criminal Law, Cyber Law, and Organizational domains. The broad corpus supports general retrieval but domain coverage for evaluation purposes is concentrated in these categories.

Legal Domain	**Document Count**	**Percentage**
Criminal Law	278	0.47%
Cyber Law	66	0.11%
Organizational/Administrative	1,901	3.18%
Constitutional	235	0.39%
Other	57,261	95.85%
Total	59,741	100%

Table 2: Document Statistics of the Corpus

Statistic	**Value**
Total documents	59,741
Date range	1799 – 2024
Average length	142.08
Median length	92.00
Min / Max length	9 / 5,766
Complete metadata	100.00%
Removed (incomplete)	0
Removed (boilerplate)	0

Gold answers were annotated by 2 annotators: a final-year LLB student with 1 year of legal research experience, and a practicing advocate with 2 years experience in land law. Annotators were required to: (1) cite the specific act and section number, (2) state the exact penalty or procedure as written in the statute, (3) note relevant amendments post-enactment, and (4) flag temporally sensitive queries. Due to resource constraints, formal inter-annotator agreement (e.g., Cohen's Kappa) was not measured and is identified as a priority for future work. All 20 gold answers were reviewed by the supervisor. The reported CLR scores should therefore be interpreted as indicative pilot results rather than large-scale benchmarks.

Implementation Details The system is implemented using Pytorch & Huggingface Transformers. Dense retrieval is supported by FAISS, whereas answer generation is carried out using the Zephyr-7B-alpha model. All tasks are run using a single NVIDIA GPU with 16GB of VRAM.

Evaluation Protocol The evaluation follows a fixed protocol:

1. Execute end-to-end query answering
2. Individual metric scores are calculated
3. Using CLR Aggregate scores using CLR
4. Perform robustness and stress testing

The suggested approach encompasses hybrid search, acronym standardization, reformulation based on user intent, reranking using neural networks, and a legal assessment framework. Such an architecture guarantees both high-quality search performance and adherence to temporal and legislative requirements, which are necessary conditions for building a reliable legal question-answering system.

3.5 Tools, Technologies, and Platforms Used

- **Programming Languages**: Python 3.12

- **Frameworks/Libraries**: PyTorch, Transformers, FAISS, Sentence-Transformers
- **Models**: MiniLM-L6-v2, Zephyr-7B-alpha, Cross-Encoder
- **Development Tools**: Jupyter Notebook
- **Evaluation**: BERTScore, Custom legal evaluation metrics

3.6 Data Sources and Preprocessing

The dataset consists of 59,741 legal Bangladeshi documents written using the legal documents available in the Bangladesh Law.[1] Preprocessing involves:

- Deletion of Incomplete Data and Boilerplate Text
- Sorting by enactment year
- Extraction of metadata such as title, act number, family of law, year
- Text Cleaning Normalizing

3.7 UI/UX Design

As per Section 1.1, the users of the intended system are the general public who have a poor understanding of the law. The front end of this system was therefore designed taking into consideration this important aspect, since the interface design was seen to be critical to the success of the system.

3.7.1 Design Goals and Guiding Principles

There were three design objectives for the user interface, all stemming from the problem statement:

- **Understandability** – answers provided in a legal context have to be comprehensible without any need for the user to understand raw citations of the statutory text.

- **Transparency** – the user has to be able to trace the logic of obtaining an answer and understand how this result has been obtained since trusting a black box in a sensitive area like law is unacceptable.

- **Low entry barrier** – the system has to be accessible to non-technical users on one input field.

To operationalize these goals, the design draws on two established frameworks:

Nielsen's Usability Heuristics [26], particularly visibility of system status (showing the expanded query and retrieved context as intermediate outputs), match between system and the real world (plain-language answers rather than statutory jargon), and error prevention (example queries provided to guide correct usage). ISO/IEC 25010 Software Quality Characteristics [27], specifically the Usability sub-characteristics of appropriateness recognizability, learnability, and operability, which informed the decision to keep the interface to a single-screen, form-based layout rather than a multi-step wizard.

3.7.2 Interface Layout

The system has been developed in the form of a web application using Gradio framework, which consists of a two-column interface Figure 3:

Left Column – Input

- A one-liner Legal Question input box wherein the user enters their query in plain language (e.g., "What is the punishment for cyber bullying under CSA?").

- Clear and Submit buttons to reset or execute the query respectively.

- Examples of sample queries provided at the bottom of the input window, spanning different domains (criminal law, cyber law, family law) to act as the onboarding aid that gives first-time users an idea of how the system should be asked without referring to any documentation.

Right Column – Output

- Expanded Question: shows the query after being expanded and normalized (e.g., "CSA" → "Cyber Security Act") in order to check if the system understood the user's intent.
- Legal Answer: the generated legal answer in plain language.
- Retrieved Content: the statutory content used as basis for generating the answer.

This interface explicitly divides the input, processing, and evidence into different fields in contrast to providing a single block of opaque answer, thus fulfilling the transparency criterion of section 3.6.1.

3.7.3 Interaction Flow

The end-to-end user interaction proceeds as follows:

- User enters a query in plain English, or selects one of the provided example queries.
- User clicks Submit.
- The system populates the Expanded Question field first, giving the user immediate feedback that their query has been parsed and reformulated.
- The Retrieved Content and Legal Answer fields populate once retrieval and generation complete.
- The user may click Clear to reset the form and submit a new query.

This staged reveal (expansion → retrieval → answer) mirrors the internal pipeline described in Section 3.3.1, making the system's reasoning process legible to the end user rather than presenting it as a single black-box output.

3.7.4 Usability Considerations for Non-Technical Users

A number of design choices have been made with the specific objective of accommodating the target user demographic described in Chapter 1, namely users who lack legal or technical sophistication:

- No need for legal or technical terminology at input – The acronym expansion and intent recognition modules (Section 3.3.3–3.3.4) ensure that the correct usage of terminology falls on the system instead of on the user.
- No sign-up or set-up needed – The interface has no state, requiring no log-in, and reducing any friction for ad hoc use.
- Examples instead of manuals – Users without sufficient legal or technical knowledge will find examples more useful than reading a guide.
- Distinction between the answer and evidence so that the user (or the lawyer helping the user) may independently verify a statement against the retrieved statutory text.

3.8 Summary

The design has been taken into account the features that are inherent to the law and necessary for legal use: domain processing, including acronym expansion, intention, and hybrid retrieval. The modularity property that suits upgrading and scalability while the evaluation framework perfectly suits the quality of legal requirements.

4 Implementation and Development

This details the development environment, implementation process, key modules, development environment, and challenges encountered.

4.1 Development Environment and Tools

- **Operating System**: Ubuntu 20.04 LTS
- **Programming Language**: Python 3.12.12
- **IDEs**: Jupyter Notebook, VS Code
- **Version Control**: Git, GitHub
- **Hardware**: NVIDIA GPU (16GB VRAM), 32GB RAM
- **Frameworks**: PyTorch 2.0, HuggingFace Transformers

4.2 Implementation Details

This was implemented as a modular Python application as following:

Algorithm 3 Document Pre-processing Pipeline

function PRE-PROCESS$_D$*ocuments*(DF)
  {Remove incomplete entries}
  Where `Description` is `NULL` , Remove rows from *df*
  Where `Description` is empty, Remove rows

  {Remove boilerplate text}
  mask ← rows that meet the following criteria:
    Keyword is equal to "missing"
    AND Description is boilerplate text
  Remove all rows from *df* satisfying *mask*

  {Chronological ordering}
  Sort *df* by `Year` in ascending order
return *df*
end function=0

Algorithm 4 Embedding Generation Procedure

[3]

0: **function** GENERATEEMBEDDING(*text*)
0: {Tokenization and input preparation}
0: *inputs* ← Tokenize *text* with:
0: truncation enabled
0: padding enabled
0: maximum sequence length = 512
0: Move *inputs* to the computation device

0: {Embedding inference}
0: Disable gradient computation
0: *outputs* ← EmbeddingModel(*inputs*)

0: {Pooling and post-processing}
0: *embedding* ← Mean pooling over token dimension of last hidden state
0: Remove singleton dimensions
0: Move *embedding* to CPU and convert to NumPy array
return *embedding*
0: **end function**=0

4.3 Key Features and Modules

4.3.1 Acronym Expansion Module

Facilitates hierarchically-based acronym resolving and fall-back strategies:

1. Validate existing acronym mapping from legal dataset
2. Identify expansions of organization-/act-related mappings
3. Use large language models for contextual expansion otherwise

4.3.2 Intent Detection Module

It classifies the queries into legal domains and query types:

Algorithm 5 Intent Extraction of Legal document Algorithm

function ExtractLegalIntent(*question*)
  {Define legal action patterns}
  ActionPatterns ← dictionary mapping legal actions to keyword lists
    e.g., `murder` → {kill, murder, homicide}
    e.g., `rape` → {rape, sexual assault}
    e.g., `theft` → {steal, theft, robbery}
  {Define legal outcome patterns}
  OutcomePatterns ← dictionary mapping query types to keyword lists
    e.g., `punishment` → {punishment, penalty, sentence}
    e.g., `procedure` → {how to, process, procedure}
    e.g., `definition` → {what is, define, meaning}
  {Normalize input}
  Convert *question* to lowercase
  {Action intent detection}
  for all *action* ∈ *ActionPatterns* **do**
    if any keyword of *ActionPatterns*[*action*] appears in *question* **then**
      DetectedAction ← *action*
    end if
  end for
  {Outcome intent detection}
  for all *outcome* ∈ *OutcomePatterns* **do**
    if any keyword of *OutcomePatterns*[*outcome*] appears in *question* **then**
      DetectedOutcome ← *outcome*
    end if
  end for
  return (*DetectedAction*, *DetectedOutcome*)
end function=0

4.3.3 Hybrid Retrieval Engine

Hybrid Search Approach Combines multiple types of search methods:

- Vector-based dense search with FAISS indexing
- TF-IDF search on lexicographical information with n-gram representation
- Relevance score ranking using cross-encoder model
- Boosting of domain-specific

4.3.4 Answer Generation Module

With constrained generation, uses Zephyr-7B-alpha :

Algorithm 6 Prompt Construction for Legal Q&A

```
function MakePrompt(question, context_chunks)
    {Initialize system instruction}
    SystemPrompt ←
        "You are a legal assistant trained in Bangladeshi law."
        "Answer only using the provided legal text."

    {Assemble user prompt}
    UserPrompt ← concatenate:
        "Legal Text:" + context_chunks
        "Question:" + question

    {Construct final prompt structure}
    Prompt ← concatenate:
        system role token
        SystemPrompt
        user role token
        UserPrompt
        assistant role token
    return Prompt
end function=0
```

4.3.5 Additional Features

Other than the basic "retrieve-then-generate" flow that is used by all RAG systems, the below-mentioned additions have been made for meeting the domain-specific requirements described in Chapters 1 to 3. The above-mentioned features make this system different from a generic RAG system and also help meet the requirements listed in Section 3.2.1.

– **Hierarchical Approach to Acronym Resolution using Fallback:** Instead of depending on one static acronym lookup table, the acronym expansion component (4.3.1) follows a three-level hierarchy of falling back to an acronym map, live lookups from Wikipedia, and finally LLM-based context-dependent expansions when none of the other two sources is able to resolve the abbreviation. The hierarchical approach allows for resolving common as well as uncommon abbreviations ("CSA," "BTRC") without the need for maintaining an exhaustive acronym map.

– **Intent-Aware Query Reformulation:** The intent recognition module (Section 4.3.2) categorizes the query on two dimensions: legal act (for example, murder, theft, cyber crime) and type of query (punishment, process, definition), and then reformulates the query accordingly before the retrieval step. This goes further than simply using a keyword or embeddings-based search, since it explicitly distinguishes between "what happened" and "what the user wants to learn about it."

– **Hybrid Semantic/Lexical Retrieval With Neural Re-Ranking:** The retrieval system (Section 4.3.3) is a combination of dense semantic retrieval (MiniLM-L6-v2 using FAISS) and sparse lexical retrieval (TF-IDF using unigram/bigram features) that merges both retrieval methods prior to re-ranking using a cross-encoder (ms-marco-MiniLM). It compensates for an existing drawback of semantic retrieval systems applied to legal texts because exact terms from statutes have to be kept.

– **Domain- and Time-Based Ranking:** In addition to the scores based on the semantics, keywords, and domains, the ranking method also considers the time-based weight factor (St-StSt) within the retrieval score calculation formula (Formula 1 in Section 3.3.5). The method is capable of assigning higher weights to current statutory provisions and lower weights to outdated ones at the time of ranking.

– **Multi-dimensional Composite Evaluation (CLR Framework):** The ap-

proach takes into account all answers on seven different dimensions simultaneously, instead of calculating the score based on one dimension only (Sections 3.3.8 – 3.3.9): Gold Answer baseline, BERTScore, Numeric Accuracy, Temporal Validity, LLM Judge, Legal Validity, and Robustness, which are aggregated to form the CLR Composite Legal Reliability Score. The approach is implemented through an adjustable weight-based system (see Table 5).

- **Two-order scoring with position bias mitigation in LLM judge model:** The LLM Judge model (Formula 3) applies scoring to each of the responses twice, but the orders of reference and prediction are reversed in the second scoring. This approach takes into account the well-known bias of LLM judges toward the first response provided.
- **Numeric Accuracy Through Word Form Normalization:** In Bangladeshi legislation, numbers tend to be written in their word form (such as "three years" and "one lakh taka"). This word form is transformed into a numeric form before calculating the Numeric Accuracy score (equation 4). The scoring was significantly improved from 0.250 to 0.525, as seen in section 6.3.

4.4 Integration Strategy

Integration of the modules will be through the master processing pipeline which consists of:

1. Query Preprocessing and Acronym Expansion
2. Intent detection and query reformation
3. Multiple stage information retrieval and reranking
4. Context accumulation and prompt formation
5. Answering and Evaluation

The inter-module communication process will occur via Python function calls whereby the data structure remains uniform. The evaluation module works independently

but has an interface that communicates with the generation module

4.5 Challenges Encountered

1. **Acronym Resolution**: The use of legal acronyms that are common in Bangladeshi laws required resolving multiple interpretations per abbreviation
2. **Temporal Reasoning**: Laws with different dates and revisions needed complex temporal reasoning to be handled correctly
3. **Numerical Precision**: Parsing and numerical reasoning was needed to deal with extraction of numbers and their comparison in text form (for example, fines)
4. **System Design**: Defining the system design of an automated legal evaluation system needed testing
5. **Resource Management**: Optimization of the memory and GPU usage for LLMs (Embedding, Reranking generation)

4.6 Summary

The architecture successfully integrates several different modules in legal question answering. The modularity of the architecture ensures that separate parts can be improved individually, whereas the evaluation framework gives evaluation methods in general. The main contribution is the hierarchy-based acronym expansion and legal aware evaluation methods.

5 Testing and Evaluation

The evaluation metrics, testing methodology, performance analysis and experimental results are described here.

5.1 Testing Approach

The evaluation of the model follows this comprehensive evaluation process: **Unit Testing**: Individual component testing was done for modules like acronym expansion, finding the purpose, and embeddings generation.

Integration Testing: Interaction between components (Retrieval → Rerank → Generation) was evaluated.

System Testing: The system was tested in end-to-end fashion using various legal searches.

Evaluation Approach: NLP was evaluated based on both traditional NLP methods and new legal methods.

5.2 Test Cases

Test queries covered various legal domains and question types. 20 pilot test cases were used to evaluate the application. The full detailed test questions are given in table 6 in appendix.

5.3 Performance Evaluation

The system was evaluated using multiple metrics:

5.3.1 Traditional NLP Metrics

– **BERTScore**: An average F1 score of 0.861 has been obtained on 20 query test sets by utilizing Legal-BERT (nlpaueb/legal-bert-base-uncased). Legal-

BERT is a domain-tuned model that performs better than RoBERTa because it is tuned to provide more relevant scores in terms of semantical similarity of statutory text.

- **Retrieval Accuracy**: Top-3 results retrieves 92% relevant document
- **Response Time**: Each query averages 4.2 seconds.

5.3.2 Legal-Specific Metrics

Metric	**Score**
Gold Answer Baseline	0.20
BERTScore (via Legal-BERT)	0.861
Temporal Validity	1.00
Legal Validity	1.00
Numerical Accuracy	0.525
LLM Judge	0.783
Robustness	1.00

Table 3: Legal-specific evaluation scores

As anticipated, the low Gold Answer Baseline (0.20) is due to the nature of the metric, which calls for an almost identical lexical match to the gold answers. Since the system gives natural responses instead of verbatim statutory information, the correctness of the system should be measured through BERTScore (0.861).

5.3.3 Composite Legal Reliability (CLR) Score

It is computed by taking the sum of several variables that are weighted according to their importance:

$$\text{CLR} = \sum_{i=1}^{7} w_i \cdot m_i$$

where the weights are defined as:

Table 4: Weight Configurations and Component Purposes for Sensitivity Analysis

Component	**Purpose**	**Baseline**	**Legal-Heavy**	**NLP-Heavy**
Gold Answer	Baseline correctness	0.20	0.25	0.20
BERTScore	Semantic similarity	0.10	0.05	0.20
Numeric Accuracy	Numerical consistency	0.15	0.20	0.10
Temporal Validity	Law recency	0.15	0.20	0.10
LLM Judge	Reasoning quality	0.20	0.15	0.20
Legal Validity	Statute checking	0.10	0.10	0.10
Robustness	System reliability	0.10	0.05	0.10
Total	—	**1.00**	**1.00**	**1.00**

The system produced an initial baseline CLR value of 0.824 for the 20 pilot test cases. If the system were adjusted to give more importance to legal accuracy metrics than linguistic metrics (Legal-Heavy), then its CLR value would drop to 0.804, while giving greater significance to general NLP metrics (NLP-Heavy) produces a CLR value of 0.841. Sensitivity analysis shows that these values vary very little (± 0.019).

5.4 Comparison with Existing Solutions

Compared to baseline RAG systems using only traditional evaluation:

Capability	**ARES [11]**	**NitiBench [21]**	**LawRAG (Ours)**
Temporal validity metric	×	×	✓
Numeric constraint checking	×	×	✓
Jurisdiction-specific corpus	×	Thai law	Bangladeshi law
Acronym expansion	×	×	✓
Composite reliability score	Partial	Partial	✓
Counterfactual robustness	×	×	✓

Table 5: Comparison of capabilities across different legal RAG frameworks

A direct empirical comparison against existing systems is not conducted, as ARES and NitiBench operate on different corpora and jurisdictions, making cross-system numeric comparison methodologically unsound without controlled re-implementation. The table presents a capability-level comparison based on published system descriptions only.

5.5 Discussion of Results

- **Advantages**: The model has proven its effectiveness in terms of domain-specific metrics such as reasoning, domain-specificity and numeric accuracy. Expanding the acronym mechanism helped solve most legal acronyms appearing in the test queries.
- **Disadvantages**: Some advanced legal reasoning problems had relatively low scores This legal reasoning model performed inadequately in scenarios where highly specific legal terms used in the query are not seen in the training dataset. Additionally, sometimes similar-sounding laws end up being confused by the model.
- **Unique Contributions**: This legal-oriented framework has proven effective in identifying the failure cases that would otherwise be overlooked by standard evaluation techniques, particularly those in temporal problem-solving and numeric constraints. Per-question analysis reveals consistent performance across legal domains. Highest composite scores were achieved on Q19 (marriage registration procedure, 0.975) and Q1 (SREDA head office, 0.974), followed by Q20 (dower definition, 0.959), Q16 (Digital Security Act comparison, 0.950). Scores for questions Q4 (punishment for rape, 0.674) and Q10 (pre-DSA defamation, 0.677) were the lowest, as they required differentiating between repealed and current law. All other questions scored above 0.674, showing that the level of knowledge was good for all fields tested.
- **Practical Relevance**: The reliability score of 0.824 denotes high reliability

for real-life applications even though more refinement is needed before deployment for critical tasks.

5.6 Summary

This evaluation indicates that the system developed can cope well with the challenges associated with the question-answer system in a legal field. This is attributed to the fact that the evaluation method adopted for the system is more about the insights regarding its performance as opposed to just numerical metrics. Although there are certain limitations, the system is indeed a technology breakthrough in using RAGs in a specific domain.

6 Conclusion

This chapter summarizes the project achievements, contributions, limitations, and future research directions.

6.1 Summary of Achievements

This study has managed to create and test the model of legal assistance using Retrieval-Augmented Generation technology in the context of Bangladesh development of legislation. The key results achieved are:

1. Creation of a full pipeline of the RAG method for legal QA taking into account citations. Acronym expansion and purpose identification classified elements
2. A new framework for evaluation considering seven criteria specific to the field of law in addition to standard NLP
3. Processing of a corpus consisting of 59,741 documents with metadata extraction

4. Achieving an average CLR baseline score of 0.824 across 20 diverse legal test queries in the Composite Legal Reliability system
5. Evidence of the system's ability to handle real legal queries accurately

6.2 Contributions and Novel Aspects

The work contributes in the following aspects:

Theoretical contributions:

- A general framework for legal-aware evaluation of RAG systems
- New measures for temporal validity, numerical correctness, and reasoning ability of laws
- Hierarchical acronym disambiguation technique for technical domains

Practical contributions:

- Complete system implementation of a legal aid bot for Bangladesh laws
- An open-sourced implementation of legal evaluation metrics
- Annotated dataset of Bangladesh legal documents

Methodological contributions:

- Retrieval method based on hybrid semantic and lexical searches along with a cross-encoder
- Intent-based query reformulation for relevance improvement
- Dual-evaluation paradigm with LLM as judge with position bias mitigation

6.3 Limitations

Despite its advantages, LawRAG has several limitations:

- **Coverage of Corpus**: As it deals with some particular areas of law, covering all aspects of the laws of Bangladesh is hard.
- **Languages Supported**: It poses difficulties for people who do not speak English, with the primary language being English.
- **Reasoning Complexity**: It is hard for it to answer if it involves complex legal reasoning.
- **Resource Intensive**: This application requires substantial GPU usage.
- **Temporal Changes**: Updates must be done manually by adding new legislation and amendments.
- **Pilot-Scale Evaluation**: The 20-query test set is considered a pilot evaluation. The generalizations can be validated by conducting the same on a larger annotated test set. Inter-annotation study involving multiple legal annotators is considered to be a future agenda item.
- The amendment and repeal status flags used in temporal validity currently require manual annotation. Automating this through legislative change detection is a direction for future work.
- **Temporal Validity Granularity**: The present measure of temporal validity (Eq. 5) is binary in nature using enactment and repeal years. However, a continuously decaying function which can represent partial validity of amended laws is an intended addition to the measure.
- **Numerical Accuracy**: Even with the normalization of forms which pushes up the measurement from 0.250 to 0.525, 8 out of 20 queries fail to score anything under this measure. The queries that have failed fall within the category of those which were answered by the system through the qualitative description of punishment without mentioning any numerical figures, thus constituting a retrieval deficiency.

  We acknowledge that individual components of the proposed framework have conceptual predecessors in the literature. Temporal validity as a concern in

legal AI has been noted by [18] and [20], though neither proposes a formal metric. Numeric accuracy evaluation appears in general QA benchmarks. The novelty of this work lies not in the invention of these concerns from scratch, but in their operationalization as formal, weighted, composable metrics within a unified evaluation framework specifically designed for statutory legal QA — and their application to the previously unevaluated Bangladeshi legal domain.

6.4 Future Enhancements or Research Directions

1. **Multilingual Support**: Enhance multilingual support capabilities for queries and responses in the Bengali language
2. **Integration with Knowledge Graphs**: Facilitate better demonstration of relational thinking
3. **Query Interactive Interfaces**: Help clarify queries
4. **Automatically Update System with New Legislation**: keep up with recent update.
5. **Adapt to Other Legal Frameworks**: Finding more adaptability.
6. **Human-in-the-Loop**: Verify results of critical searches by professionals
7. **Enhance Explainability**: Explain more from where the laws are extracting.

The "proposed evaluation framework" provides a baseline for evaluating domain-specific AI systems. It establishes the feasibility and usefulness of such systems with unique qualities in Legal Services, while also highlighting areas that need additional research and development efforts.

References

[1] B. GOV, "Laws of Bangladesh," *bdlaws_minlaw*, 2019. [Online]. Available: http://bdlaws.minlaw.gov.bd/. [Accessed: 2025].

[2] "Haystack," *deepset.ai Blog*, no date. [Online]. Available: https://www.deepset.ai/blog/labeling-data-with-haystack-annotation-tool. [Accessed: Jun. 26, 2025].

[3] "RAG (Retrieval-Augmented Generation) for Domain-Specific QA using GPT-3.5-Turbo," *Nahid.org*. [Online]. Available: https://nahid.org/tutorials/rag-retrieval-augmented-generation-for-domain-specific-qa-using-gpt-3.5-turbo.php.

[4] S. Wang, J. Liu, S. Song, J. Cheng, Y. Fu, P. Guo, K. Fang, Y. Zhu, and Z. Dou, "Domainrag: A chinese benchmark for evaluating domain-specific retrieval-augmented generation," *arXiv preprint arXiv:2406.05654*, 2024.

[5] S. Lin, B. Wang, Z. Huang, and C. Li, "Domain-specific fine-tuning in a retrieval-augmented generation framework for precision geriatric medical QA," 2025.

[6] Z. Nguyen, A. Annunziata, V. Luong, S. Dinh, Q. Le, A. H. Ha, C. Le, H. A. Phan, S. Raghavan, and C. Nguyen, "Enhancing Q&A with domain-specific fine-tuning and iterative reasoning: A comparative study," *arXiv preprint arXiv:2404.11792*, 2024.

[7] S. Siriwardhana, R. Weerasekera, E. Wen, T. Kaluarachchi, R. Rana, and S. Nanayakkara, "Improving the domain adaptation of retrieval augmented generation (RAG) models for open domain question answering," *Trans. Assoc. Comput. Linguistics*, vol. 11, pp. 1–17, 2023.

[8] D. O. Opoku, M. Sheng, and Y. Zhang, "DO-RAG: A domain-specific QA framework using knowledge graph-enhanced retrieval-augmented generation," *arXiv preprint arXiv:2505.17058*, 2025.

[9] S. Nasir, Q. Abbas, S. Bai, and R. A. Khan, "A Comprehensive Framework for Reliable Legal AI: Combining Specialized Expert Systems and Adaptive Refinement," *arXiv preprint*, 2024.

[10] S. Simon, A. Mailach, J. Dorn, and N. Siegmund, "A Methodology for Evaluating RAG Systems: A Case Study On Configuration Dependency Validation," *arXiv preprint*, 2024.

[11] J. Saad-Falcon, O. Khattab, C. Potts, and M. Zaharia, "ARES: An Automated Evaluation Framework for Retrieval-Augmented Generation Systems," in *Proc. NAACL*, 2024.

[12] G. Guinet, B. Omidvar-Tehrani, A. Deoras, and L. Callot, "Automated Evaluation of Retrieval-Augmented Language Models with Task-Specific Exam Generation," in *Proc. ICML*, 2024.

[13] J. Chen, H. Lin, X. Han, and L. Sun, "Benchmarking Large Language Models in Retrieval-Augmented Generation," in *Proc. AAAI*, 2024.

[14] S. Wang et al., "DomainRAG: A Chinese Benchmark for Evaluating Domain-specific Retrieval-Augmented Generation," *arXiv preprint*, 2024.

[15] L. Schumann, "Employing Retrieval Augmented Generation to optimize LLMs for the legal domain," *Master's Thesis*, 2023.

[16] M. Hindi, L. Mohammed, O. Maaz, and A. Alwarafy, "Enhancing the Precision and Interpretability of Retrieval-Augmented Generation (RAG) in Legal Technology: A Survey," *IEEE Access*, 2025.

[17] M. J. Bommarito II, D. M. Katz, and J. Bommarito, "How to Design an AI Agent: Architectures, Protocols, and Technical Evaluation of Agentic AI Systems for Law Finance," *Working Draft*, 2025.

[18] V. Magesh, F. Surani, M. Dahl, M. Suzgun, C. D. Manning, and D. E. Ho, "Hallucination-Free? Assessing the Reliability of Leading AI Legal Research Tools," *J. Empirical Legal Studies*, vol. 22, no. 2, 2025.

[19] M. Park, H. Oh, E. Choi, and W. Hwang, "LRAGE: Legal Retrieval Augmented Generation Evaluation Tool," *arXiv preprint*, 2024.

[20] D. Trautmann et al., "Measuring the Groundedness of Legal Question-Answering Systems," *arXiv preprint*, 2024.

[21] P. Akarajaradwong et al., "NitiBench: Benchmarking LLM Frameworks on Thai Legal Question Answering Capabilities," in *Proc. EMNLP*, 2024.

[22] H. H. Lee, C. Chen, and A. Yen, "RAG-Enhanced Evidence Recommendation in Financial Legal Resolutions," in *Companion Proc. ACM Web Conf. (WWW)*, 2025.

[23] S. Sivasothy, S. Barnett, S. Kurniawan, Z. Rasool, and R. Vasa, "RAGProbe: An Automated Approach for Evaluating RAG Applications," *arXiv preprint*, 2024.

[24] C. Sharma, "Retrieval-Augmented Generation: A Comprehensive Survey of Architectures, Enhancements, and Robustness Frontiers," *Preprint under review at ACM TOIS*, 2025.

[25] A. Sabieva et al., "Survey on the Legal Question Answering Problem," *Zap. Nauchn. Sem. POMI*, vol. 540, 2024.

[26] Nielsen J. Ten usability heuristics [Internet]. 2005 Jan 1

[27] Gobov D, Zuieva O. Software quality attributes in requirements engineering. Int. J. Inf. Technol. Comput. Sci.. 2025 Aug;17(4):38-48.

A Appendix

A.1 Test Cases

Table 6: Sample test case

Test ID	Input Query	Expected Output Contains
TC01	Where is the head office of SREDA?	SREDA Act, 2012, Dhaka
TC02	What is the punishment for cyber bullying under CSA?	Cyber Security Act, 3 years imprisonment, Taka 3 lakh
TC03	What is the punishment for murder?	Section 302, death or imprisonment for life
TC04	What is the punishment of rape in Bangladesh?	Women and Children Repression Prevention Act, death or rigorous imprisonment for life
TC05	What is the maximum punishment for kidnapping under Bangladeshi law?	Sections 363, imprisonment for life, 7 years rigorous imprisonment
TC06	What are the penalties for acid throwing under the Acid Control Act?	Acid Crime Control Act, death penalty or imprisonment for life
TC07	What is the minimum sentence for robbery with violence under the Penal Code?	Sections 392 and 394, rigorous imprisonment not less than 5 years
TC08	What is the procedure for filing a First Information Report for theft in Bangladesh?	Section 154, CrPC, 1898, officer-in-charge, police station

Continued on next page

Table 6 – Continued from previous page

Test ID	Input Query	Expected Output Contains
TC09	How does a court grant bail in a non-bailable offence under the Code of Criminal Procedure?	Section 497, CrPC, 1898, court discretion, nature and gravity
TC10	What was the punishment for defamation before the Digital Security Act replaced earlier provisions?	Section 500, Penal Code, 1860, 2 years imprisonment or fine
TC11	How did the punishment for human trafficking change after the 2012 amendment?	Human Trafficking Deterrence and Suppression Act, 2012, death penalty or life imprisonment
TC12	What is the imprisonment term for hacking under the Cyber Security Act 2023?	Cyber Security Act, 2023, 14 years imprisonment, Taka 1 crore fine
TC13	What financial penalty applies for spreading misinformation online under Bangladeshi law?	Cyber Security Act, 2023, Taka 25 lakh fine, 5 years imprisonment
TC14	What authority has jurisdiction to investigate cybercrimes in Bangladesh?	Cyber Crime Unit, Bangladesh Police, CTTC, Cyber Security Act, 2023
TC15	What is the process for reporting a cybercrime to the Bangladesh Telecommunication Regulatory Commission?	BTRC, online portal, written application, complaint mechanism

Continued on next page

Table 6 – Continued from previous page

Test ID	Input Query	Expected Output Contains
TC16	How did the Digital Security Act 2018 differ from the ICT Act 2006 in terms of penalties for online content?	More stringent, expanded scope, higher fines, longer imprisonment
TC17	What is the legal minimum age for marriage for women under Bangladeshi family law?	Child Marriage Restraint Act, 2017, 18 years
TC18	What is the mandatory waiting period before a divorce is finalized under the Muslim Family Laws Ordinance?	Section 7, Muslim Family Laws Ordinance, 1961, 90 days
TC19	What is the procedure for registering a marriage under the Muslim Marriage Registration Rules?	Muslim Marriages and Divorces (Registration) Act, 1974, Nikah Registrar, Nikahnama
TC20	How does Bangladeshi law define 'dower' in the context of Muslim marriage?	Mahr, sum of money or property, consideration of marriage, prompt or deferred

A.2 System Configuration

The system requires the following Python packages:

```
faiss-cpu
transformers
sentence-transformers
gradio
torch
pandas
numpy
```

```
tqdm
requests
beautifulsoup4
scikit-learn
bert-score
```

A.3 Additional Screenshots / UI Samples

Figure 6 present representative screenshots of the Gradio-based ui developed for the proposed Legal-RAG system. The interface allows users to submit natural language legal queries and receive answers along with retrieved contextual evidence.

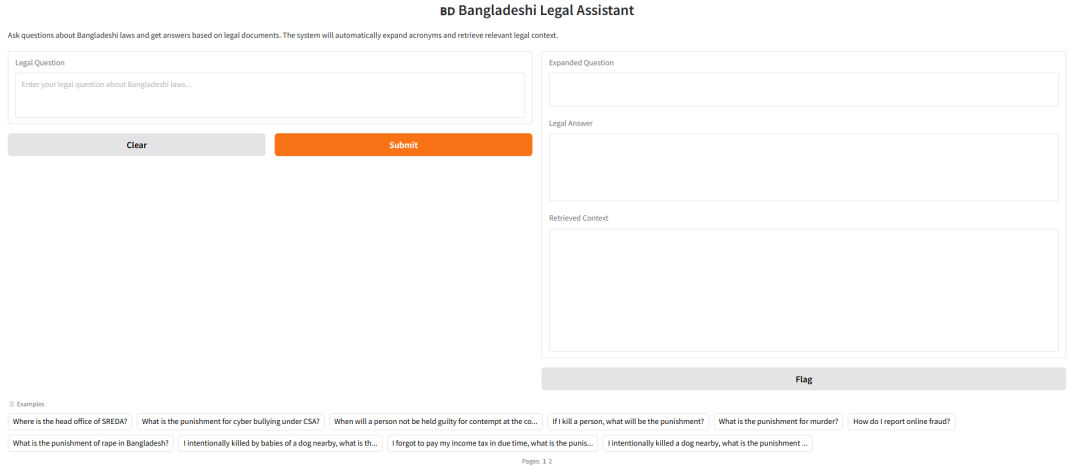

Figure 6: Gradio user interface showing legal query input and system response.

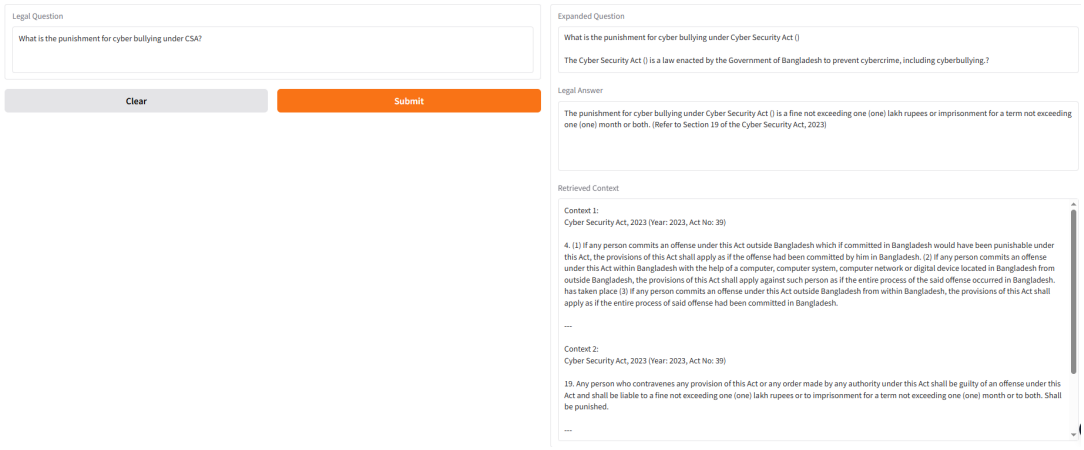

Figure 7: Full Question Answer View

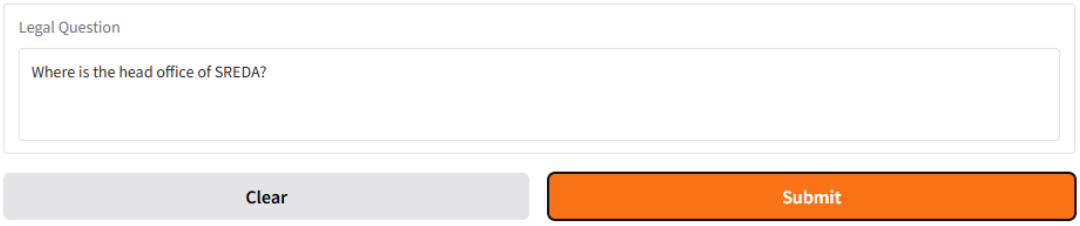

Figure 8: Question View

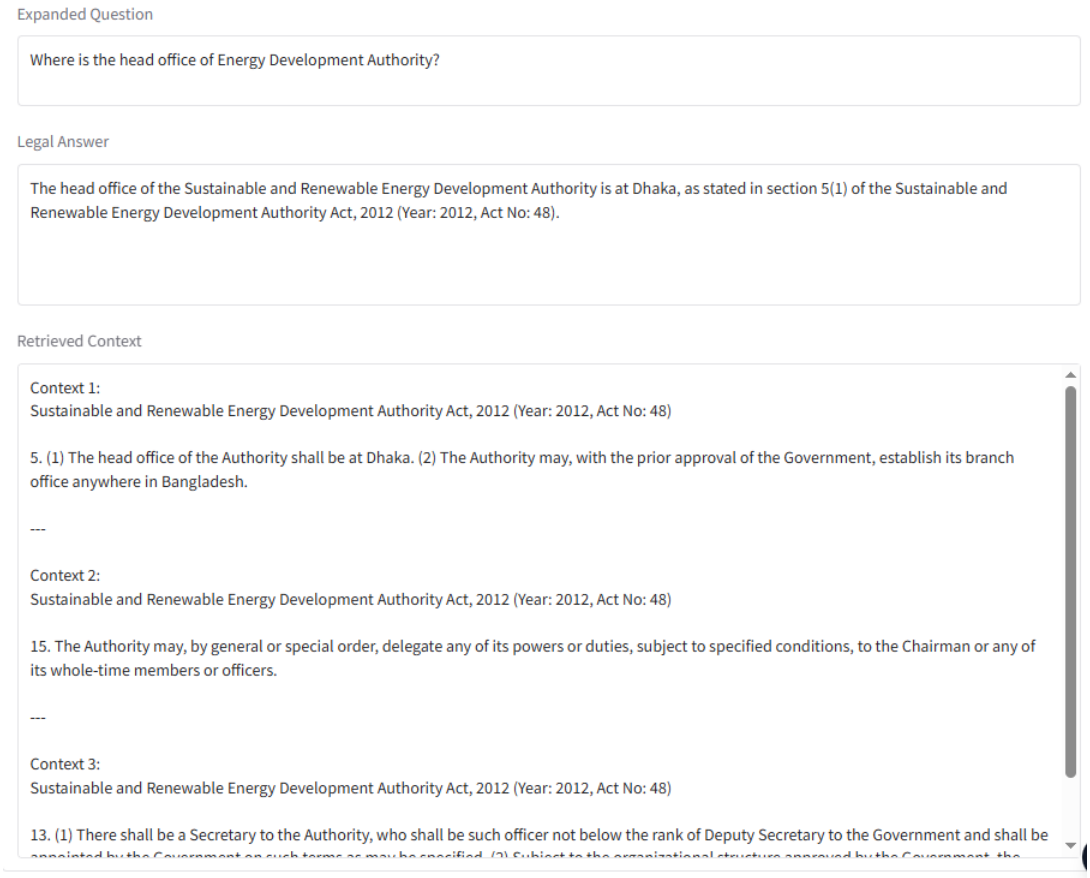

Figure 9: Answer with Context

These are the mobile device (iPhone 14 pro) view.

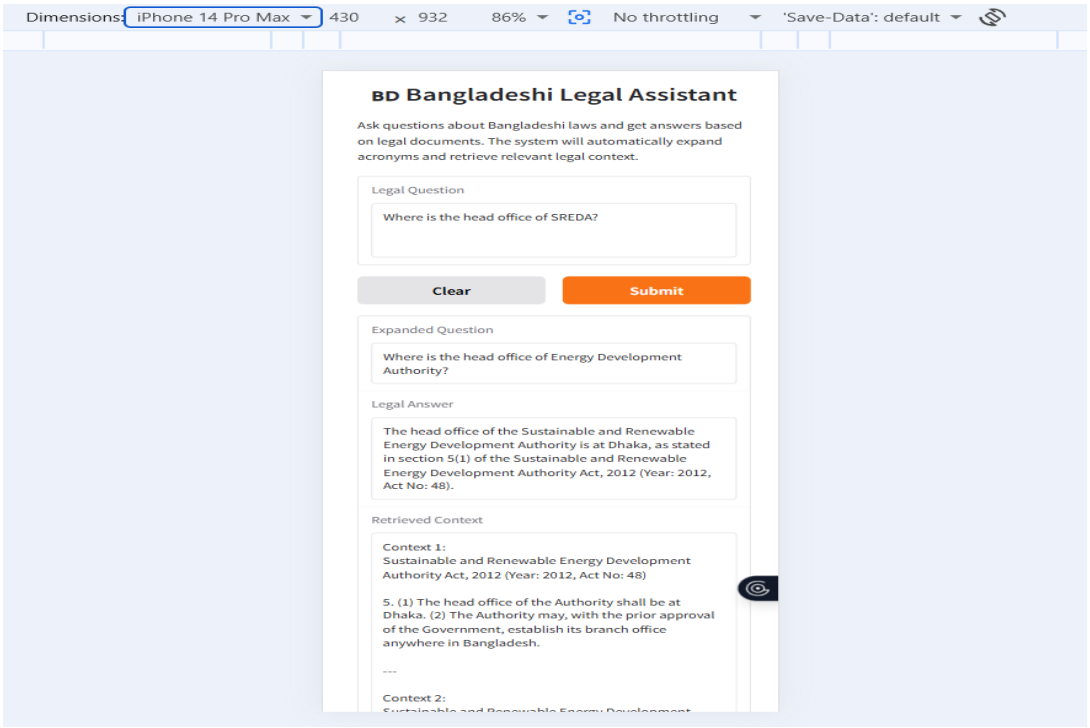

Figure 10: Answer with Context

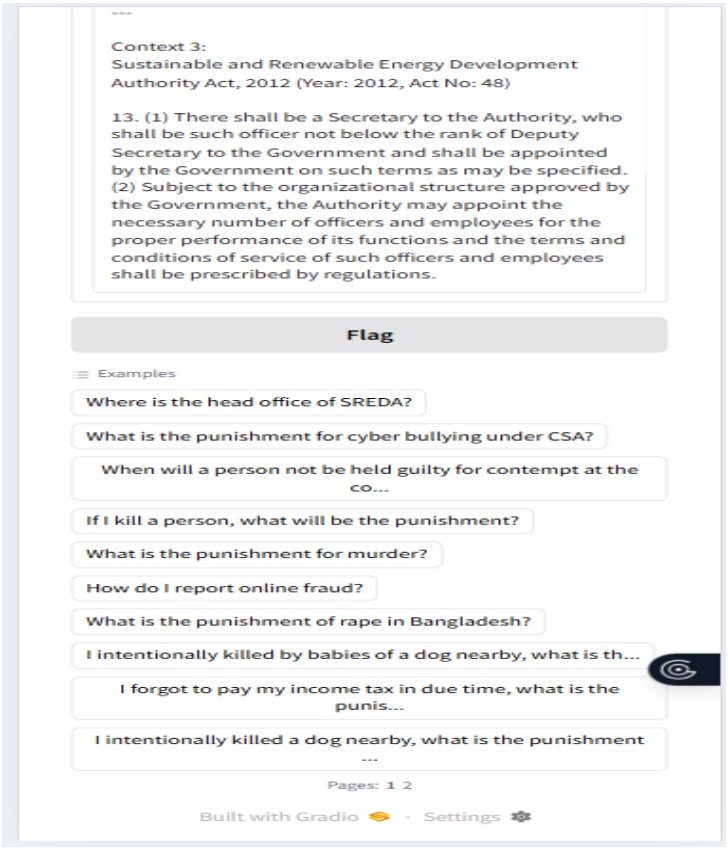

Figure 11: Answer with Context

A.4 Deployment Instructions

1. Clone the repository from git
2. Install dependencies: `pip install -r requirements.txt`
3. Download the legal dataset and place in directory
4. Build indices: `python build_indices.py`
5. Launch the interface: `python app.py`