# Question-Answering Chatbot for Vietnamese Traffic Laws

**University of Information Technology, VNU-HCM**
**Supervisor:** PhD. Nguyen Vinh Tiep
**Author:** Le Minh Nhut, Tran Dinh Khanh Dang, Thai Ngoc Quan

**Abstract**

Addressing the complexities of Vietnam's road traffic regulations, this project introduces an intelligent chatbot to provide citizens with efficient and accurate access to legal information. Our system is built upon a Retrieval-Augmented Generation (RAG) framework, which integrates a Large Language Model (LLM) with a sophisticated Hybrid Retrieval strategy. This approach synergizes sparse (keyword-based) and dense (semantic) search techniques to precisely retrieve relevant articles from a comprehensive corpus of traffic law documents. This report details the system's end-to-end architecture, from data collection and preprocessing to evaluation. Rigorous testing demonstrates the system's high performance, notably achieving

94.68% accuracy on a set of official driver's license exam questions, validating its effectiveness as a practical tool for public use.
This project is available at: Traffic Law Chatbot on GitHub

## 1 Introduction

In 2024, Vietnam's road traffic landscape was marked by 21,532 accidents, resulting in 9,954 fatalities and 16,044 injuries, according to the Traffic Police Department [1]. A significant portion of these incidents stems from legal violations, often exacerbated by a lack of public awareness and the inherent difficulty in accessing and interpreting complex legal documents. This complexity creates a significant barrier, leading to unintentional violations and disputes. In this context, technologies in Natural Language Processing (NLP), particularly Large Language Models (LLMs), present a transformative opportunity to develop user-friendly and efficient legal information retrieval systems.

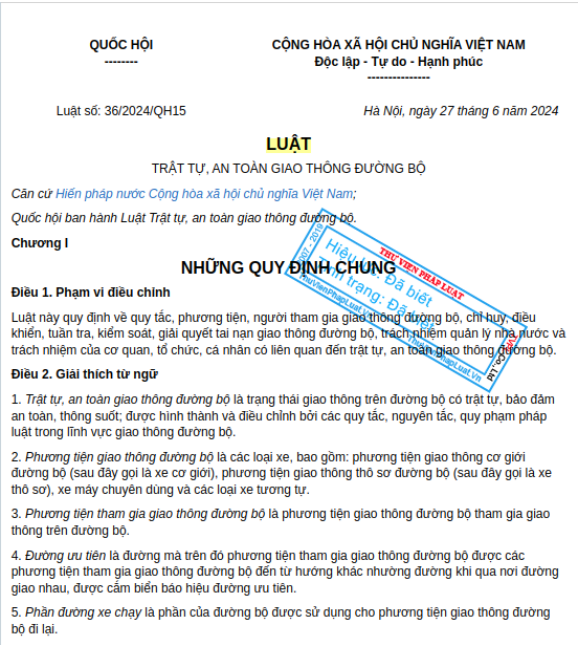
To address this challenge, this project details the development of a specialized chatbot for Vietnamese traffic law. Our system is built upon a Retrieval-Augmented Generation (RAG) architecture. RAG is a state-of-the-art framework designed to enhance the capabilities of LLMs by grounding them in external, up-to-date knowledge. Instead of relying solely on its internal, pre-trained knowledge, a RAG system first retrieves relevant information from a specified corpus and then uses that information as context to generate a more accurate and factually consistent response. This approach is particularly effective in specialized domains like law, as it significantly reduces model ”hallucination” and ensures that answers are based on verifiable sources.

Our implementation of the RAG pipeline employs a sophisticated hybrid retrieval mechanism, which synergizes a keyword-based sparse retriever (BM25) with a semantic-based dense retriever (bge-m3). The retrieved candidates are then refined by a cross-encoder reranker (bge-reranker-v2-m3) to select the most relevant legal articles. The primary objective is to provide citizens with a reliable tool for fast and accurate answers to their legal queries, ensuring every piece of information is transparently supported by citations from the authoritative legal source texts.

## 2 Dataset

The dataset was constructed by collecting legal documents from **Thư viện Pháp luật** (thuvienphapluat.vn), a reputable, up-to-date, and widely used official legal database in Vietnam. The data collection focused on documents related to *road traffic order and safety*, and involved the following:

- **Sources:** Legal documents issued by the National Assembly, Government, Ministry of Public Security, and Ministry of Transport.
- **Content:** Includes **46 legally valid documents**, such as Laws, Decrees, and Circulars, ... and **about 400 traffc signs**.
- **Examples:** Law No. 36/2024/QH15, Decree No. 168/2024/NĐ-CP, among other newly enacted or amended documents in late 2024 and early 2025.
- **Domain:** Focused on legal regulations related to road traffic safety and violations.
- **Collection method:** Legal documents are saved as text files, followed by manual verification to ensure structural and legal content accuracy.



QUỐC HỘI
--------

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc
----------

Luật số: 36/2024/QH15

*Hà Nội, ngày 27 tháng 6 năm 2024*

**LUẬT**

TRẬT TỰ, AN TOÀN GIAO THÔNG ĐƯỜNG BỘ

*Căn cứ Hiến pháp nước Cộng hòa xã hội chủ nghĩa Việt Nam;*

*Quốc hội ban hành Luật Trật tự, an toàn giao thông đường bộ.*

**Chương I**

**NHỮNG QUY ĐỊNH CHUNG**

**Điều 1. Phạm vi điều chỉnh**

Luật này quy định về quy tắc, phương tiện, người tham gia giao thông đường bộ, chỉ huy, điều khiển, tuần tra, kiểm soát, giải quyết tai nạn giao thông đường bộ, trách nhiệm quản lý nhà nước và trách nhiệm của cơ quan, tổ chức, cá nhân có liên quan đến trật tự, an toàn giao thông đường bộ.

**Điều 2. Giải thích từ ngữ**

1. *Trật tự, an toàn giao thông đường bộ* là trạng thái giao thông trên đường bộ có trật tự, bảo đảm an toàn, thông suốt; được hình thành và điều chỉnh bởi các quy tắc, nguyên tắc, quy phạm pháp luật trong lĩnh vực giao thông đường bộ.

2. *Phương tiện giao thông đường bộ* là các loại xe, bao gồm: phương tiện giao thông cơ giới đường bộ (sau đây gọi là xe cơ giới), phương tiện giao thông thô sơ đường bộ (sau đây gọi là xe thô sơ), xe máy chuyên dùng và các loại xe tương tự.

3. *Phương tiện tham gia giao thông đường bộ* là phương tiện giao thông đường bộ tham gia giao thông trên đường bộ.

4. *Đường ưu tiên* là đường mà trên đó phương tiện tham gia giao thông đường bộ được các phương tiện tham gia giao thông đường bộ khác nhường đường khi qua nơi đường giao nhau, được cắm biển báo hiệu đường ưu tiên.

5. *Phần đường xe chạy* là phần của đường bộ được sử dụng cho phương tiện giao thông đường bộ đi lại.

Figure 1: Sample Legal Document (PDF)

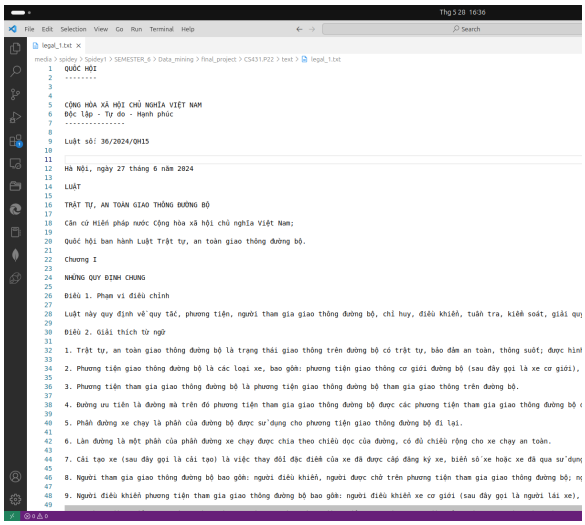


Figure 2: Sample Legal Document (Text)

# 3 Data Preprocessing

During the development of the legal-document-based question answering system, data preprocessing plays a crucial role in standardizing and organizing the collected legal texts, enabling the system to retrieve and interpret information effectively.

## 3.1 Chunking Based on Legal Document Structure

Legal documents (in PDF or plain text format) often follow a hierarchical structure as defined by Vietnamese law, typically in the form of: 'Chương' → 'Điều' → 'Khoản' → 'Điểm'. Based on this structure, the system splits each document into smaller segments (chunks), while preserving the full contextual meaning of each segment.

## 3.2 Metadata Extraction

After chunking, the system proceeds to extract accompanying metadata for each chunk, serving retrieval, management, and query tasks during both training and deployment phases. The metadata fields include:

- **source:** Document reference code (e.g., Law 36/2024/QH15, Decree 168/2024/NĐ-CP).
- **effective_date:** The effective date of the document, typically retrieved from the beginning or end of the text.
- **url:** Original source link from the official legal library website.
- **context:** Structural hierarchy information including `chapter`, `article`, `clause`, `point`, allowing the system to precisely locate where the segment belongs within the full document.
- **traffic_sign:** An illustrative traffic sign image (if available), associated with the regulation in the corresponding text segment to provide visual context.

All processed chunks, along with their metadata, are stored as individual JSON objects. Each object represents one chunk and follows a consistent structure, which is suitable for the subsequent stages such as embedding, search, and semantic querying. Below is an example of such a JSON object:


```
{
  "id": "41_2024_BGTVT_chunk_510",
  "text": "Phụ lục B\nB.2 Biển số P.102 "Cấm đi ngược chiều"
  \na) Để báo đường cấm các loại xe (cơ giới và thô sơ)...",
  "metadata": {
    "source": "Quy chuẩn kỹ thuật số 41_2024_BGTVT",
    "effective_date": "1-1-2025",
    "url": "https://luatvietnam.vn/giao-thong/q...",
    "context": {
      "khoan": "a",
      "dieu": "B.2",
      "phu_luc": "B"
```

```
    },
    "traffic_sign": ["P_102.png"]
  }
}
```

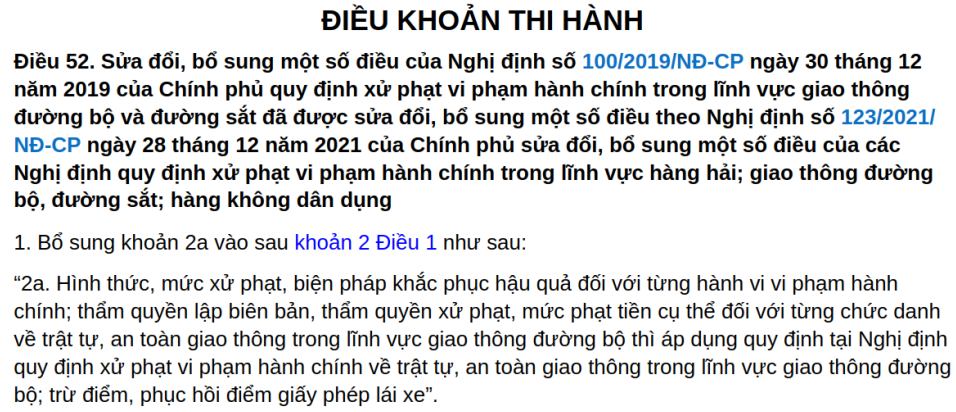

## 3.3 Challenges in Legal Data Processing

While constructing the knowledge base for the chatbot, we encountered two major challenges that required considerable manual effort to resolve. These challenges are specific to the nature of Vietnamese legal documents and their frequent cross-referencing and amendment structure:

1. **Handling amended or replaced clauses across legal documents.** Legal articles are often modified, supplemented, or replaced in separate legal texts. In the absence of officially consolidated versions, the system must manually track and incorporate updates from newer documents to reflect the most accurate legal state.

2. **Resolving indirect references between clauses.** Many legal provisions reference other articles, clauses, or points without explicitly restating the legal behavior, violation, or object in question. To make such information retrievable and understandable for the chatbot, these references must be resolved and expanded into fully self-contained chunks.

The most effective solution to address both issues is a semi-manual process: carefully reading the legal texts and rewriting or merging referenced content to produce complete, standalone passages suitable for retrieval and generation.



**ĐIỀU KHOẢN THI HÀNH**

**Điều 52. Sửa đổi, bổ sung một số điều của Nghị định số 100/2019/NĐ-CP ngày 30 tháng 12 năm 2019 của Chính phủ quy định xử phạt vi phạm hành chính trong lĩnh vực giao thông đường bộ và đường sắt đã được sửa đổi, bổ sung một số điều theo Nghị định số 123/2021/NĐ-CP ngày 28 tháng 12 năm 2021 của Chính phủ sửa đổi, bổ sung một số điều của các Nghị định quy định xử phạt vi phạm hành chính trong lĩnh vực hàng hải; giao thông đường bộ, đường sắt; hàng không dân dụng**

1. Bổ sung khoản 2a vào sau khoản 2 Điều 1 như sau:

"2a. Hình thức, mức xử phạt, biện pháp khắc phục hậu quả đối với từng hành vi vi phạm hành chính; thẩm quyền lập biên bản, thẩm quyền xử phạt, mức phạt tiền cụ thể đối với từng chức danh về trật tự, an toàn giao thông trong lĩnh vực giao thông đường bộ thì áp dụng quy định tại Nghị định quy định xử phạt vi phạm hành chính về trật tự, an toàn giao thông trong lĩnh vực giao thông đường bộ; trừ điểm, phục hồi điểm giấy phép lái xe".

Figure 3: Example of legal amendments in later documents.

For instance, as shown in Figure 3, Article 52, Clause 1 from a more recent legal document must be used to update the content of Decree No. 100/2019/NĐ-CP, which is still in effect but does not reflect the latest amendments unless a consolidated version is available.

In another example (Figures 4 and 5), Clause 12, Point a refers to Clause 2, Point đ without repeating the full context. To make this clause usable in a retrieval system, we manually merge the content as follows: *"Khoản 12, Điểm a) Thực hiện hành vi* ***xe không được quyền ưu tiên lắp đặt sử dụng thiết bị phát tín hiệu của xe được quyền ưu tiên*** *còn bị tịch thu thiết bị phát tín hiệu ưu tiên lắp đặt, sử dụng trái quy định."*

12. Ngoài việc bị áp dụng hình thức xử phạt chính, người điều khiển xe thực hiện hành vi vi phạm còn bị áp dụng các hình thức xử phạt bổ sung sau đây:

a) Thực hiện hành vi quy định tại điểm đ khoản 2 Điều này còn bị tịch thu thiết bị phát tín hiệu ưu tiên lắp đặt, sử dụng trái quy định;

b) Thực hiện hành vi quy định tại điểm a, điểm b, điểm h, điểm i, điểm k khoản 9 Điều này bị tước quyền sử dụng giấy phép lái xe từ 10 tháng đến 12 tháng;

Figure 4: Indirect reference in a legal clause (Figure 5a).

2. Phạt tiền từ 400.000 đồng đến 600.000 đồng đối với người điều khiển xe thực hiện một trong các hành vi vi phạm sau đây:

a) Dừng xe, đỗ xe trên phần đường xe chạy ở đoạn đường ngoài đô thị nơi có lề đường;

b) Điều khiển xe chạy quá tốc độ quy định từ 05 km/h đến dưới 10 km/h;

c) Điều khiển xe chạy tốc độ thấp mà không đi bên phải phần đường xe chạy gây cản trở giao thông;

d) Dừng xe, đỗ xe ở lòng đường gây cản trở giao thông; tụ tập từ 03 xe trở lên ở lòng đường, trong hầm đường bộ; đỗ, để xe ở lòng đường, vỉa hè trái phép;

đ) Xe không được quyền ưu tiên lắp đặt, sử dụng thiết bị phát tín hiệu của xe được quyền ưu tiên;

Figure 5: Referenced clause containing full information (Figure 5b).

# 4 System Architecture

The question-answering system is implemented based on a multi-stage pipeline architecture to ensure efficient processing from the input data to the generation of accurate and contextually appropriate answers. An overview of the system architecture is illustrated in Figure 6, comprising the following main components:

- **Embedding:** Encode legal text chunks into numerical vectors to enhance retrieval effectiveness compared to traditional keyword matching methods.
- **Index Construction:** Build an index from the embedded vectors to enable fast and accurate semantic search across the legal corpus.
- **Query Reception and Parsing:** Process user queries by normalizing semantics and structure, bridging the gap between user language and legal phrasing. The system supports three different querying modes to suit various question types.
- **Information Retrieval:** Retrieve potentially relevant text chunks from the database based on semantic similarity with the parsed query.
- **Reranking:** Refine the ranking of candidate text chunks to prioritize those most relevant and accurate to the user's query.
- **Prompt Design for Answer Generation:** Generate final answers grounded solely on the retrieved context, ensuring accuracy, legal consistency, and clarity.
- **Contextual Memory:** Maintain conversation history to support contextual continuity, especially for follow-up or dependent questions.

## 4.1 Embedding

To encode text passages into dense vector representations, the system employs the `SentenceTransformer` framework, allowing flexible integration of various pretrained and fine-tuned embedding models. We experimented with the following three models to evaluate their effectiveness in representing Vietnamese legal content:

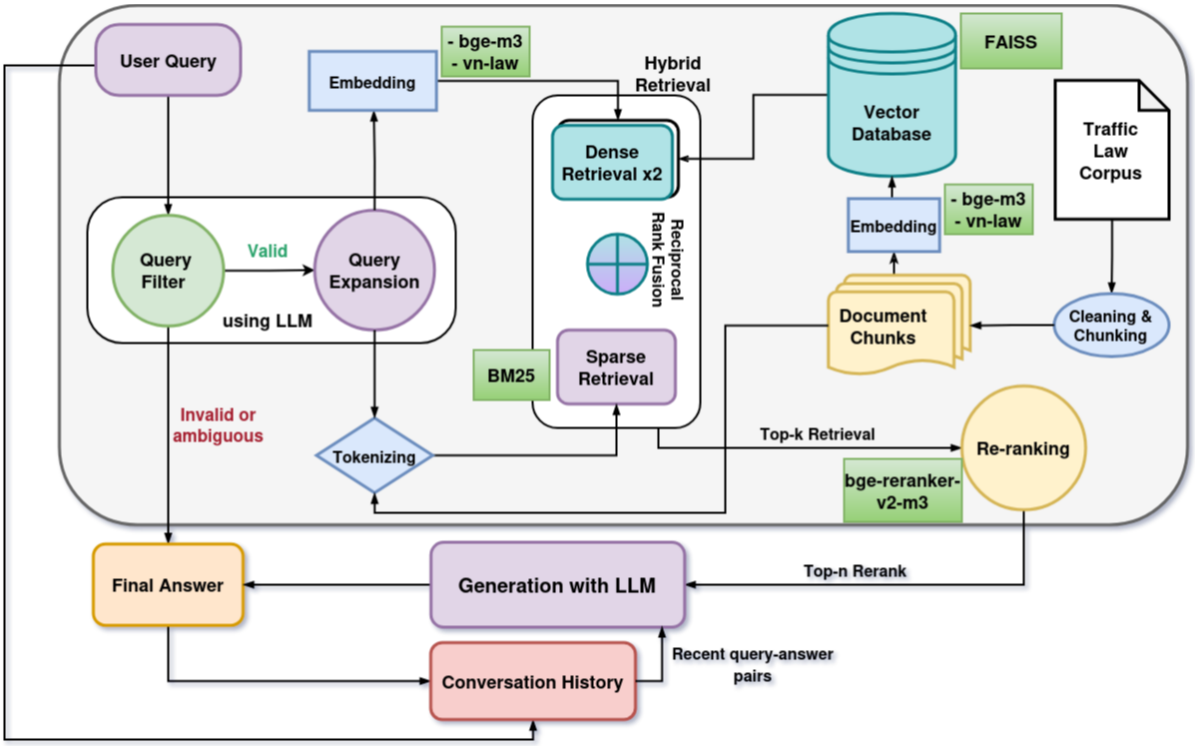



Figure 6: Pipeline of the Road Traffic Legal Chatbot System

- **vn-law-embedding**: A domain-specific model fine-tuned on Vietnamese legal corpora. It produces 768-dimensional vectors and is optimized for capturing legal semantics in the native language.
- **bge-m3**: A high-capacity multilingual model capable of handling both short and long input sequences. It generates 1024-dimensional embeddings and has demonstrated strong performance across multiple languages.
- **multilingual-e5-large**: A transformer-based embedding model supporting over 100 languages, including Vietnamese. It also produces 1024-dimensional vectors and is designed for cross-lingual retrieval and general-purpose semantic search.

These embeddings serve as the foundation for the dense indexing stage, where similarity search is later conducted using FAISS.

Table 1: Comparison of embedding model performance

| Metric | vn-law-embedding | | | | bge-m3 | | | | multilingual-e5-large | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | @3 | @5 | @10 | @15 | @3 | @5 | @10 | @15 | @3 | @5 | @10 | @15 |
| **Recall** | 0.4677 | 0.5381 | 0.5938 | 0.6124 | **0.5477** | **0.6308** | **0.7058** | **0.7346** | 0.5173 | 0.6124 | 0.6864 | 0.7126 |
| **MRR** | 0.6717 | 0.6812 | 0.6849 | 0.6862 | **0.7117** | **0.7203** | **0.7244** | **0.7260** | 0.6967 | 0.7083 | 0.7132 | 0.7148 |
| **nDCG** | 0.5450 | 0.5484 | 0.5633 | 0.5701 | **0.6150** | **0.6255** | **0.6490** | **0.6597** | 0.5919 | 0.6087 | 0.6306 | 0.6407 |

## 4.2 Index Construction

To support different retrieval strategies, we constructed both dense and sparse indexes, each tailored for specific types of user queries and downstream fusion mechanisms.

The **dense index** was built using the embedding vectors generated in the previous stage. We employed the FAISS library for fast and scalable similarity search, configured with the `IndexFlatL2` option, which performs exhaustive comparison over all vectors using Euclidean distance. Each embedding vector is linked to metadata about its corresponding chunk, including the chunk ID, document title, and relevant attributes. This setup allows for efficient semantic retrieval and reranking.

In parallel, we constructed a **sparse index** based on the BM25 algorithm—a probabilistic retrieval model that improves upon traditional TF-IDF by incorporating term frequency saturation and document length normalization. The indexing pipeline includes tokenization, stopword removal, and term weighting. This method is especially effective for handling keyword-centric queries, which are common in legal and regulatory domains.

The combination of both index types enables a hybrid retrieval strategy, where semantic and lexical signals can be integrated to improve coverage and relevance during information retrieval.

## 4.3 Query Reception and Parsing

To enhance the system's ability to interpret user input and retrieve relevant documents, we implemented a three-level query processing pipeline. This process is initiated once a query is submitted through the user interface (e.g., *"Chạy xe máy trên lề đường có bị phạt không?"*). A large language model (Gemini) is used to determine whether the query is valid (Query Filtering) and, if so, to generate variations through Query Expansion. We distinguish three types of query formulations:

- **Simple Query:** The original user question is used without modification. *Example: "Chạy xe máy trên lề đường có bị phạt không?"* This baseline approach provides fast processing but limited coverage.
- **Expanded Query:** A longer version of the original query is generated by enriching it with semantically related terms (e.g., including synonyms or legal terminology). *Example: "Đi xe máy, xe mô tô hai bánh và các loại xe tương tự trên lề đường, vỉa hè có bị xử phạt theo Luật Giao thông đường bộ không?"* This helps improve retrieval accuracy by increasing lexical and semantic overlap with relevant documents, while still maintaining the structure of a single, unified query.
- **Diverse Query:** Multiple semantically diverse variations (typically three) are generated in addition to the original query. These versions capture different phrasings or aspects of the same question, improving recall by covering a broader semantic space. *Examples:*
  - "Chạy xe máy trên lề đường có bị phạt không?" (original)
  - "Đi xe máy trên vỉa hè có vi phạm luật không?"
  - "Có bị xử lý nếu lái xe máy trên lề đường?"
  - "Luật nói gì về chạy xe máy lên vỉa hè?"

As shown in Table 2, **the diverse query approach** achieves **the highest performance across most metrics,** particularly in terms of MRR and nDCG. This is attributed to **its broader semantic coverage and robustness to variations in language.** However, **generating and processing multiple query variations incurs higher computational costs.** For this reason, the expanded query strategy—offering a single enriched query—was selected as the default configuration in

Table 2: Comparison of performance by query type

| Metric | Simple | | | | Diverse | | | | Expanded | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | @3 | @5 | @10 | @15 | @3 | @5 | @10 | @15 | @3 | @5 | @10 | @15 |
| **Recall** | 0.5771 | **0.6733** | **0.7630** | 0.7922 | **0.5808** | 0.6698 | 0.7576 | **0.7978** | 0.5675 | 0.6707 | 0.7582 | 0.7958 |
| **MRR** | 0.7211 | 0.7296 | 0.7349 | 0.7359 | **0.7483** | **0.7570** | **0.7630** | **0.7647** | 0.7306 | 0.7414 | 0.7465 | 0.7479 |
| **nDCG** | 0.6347 | 0.6559 | 0.6882 | 0.6998 | **0.6496** | **0.6604** | **0.6892** | **0.7036** | 0.6400 | 0.6571 | 0.6861 | 0.7004 |

most experiments, as it provides a practical balance between retrieval effectiveness and efficiency.

## 4.4 Information Retrieval

To retrieve relevant information from the knowledge base, we explored and compared three retrieval strategies: dense retrieval, sparse retrieval, and a hybrid of both. Dense retrieval relies on semantic similarity in the embedding space (via FAISS), while sparse retrieval leverages keyword-based matching through BM25. The hybrid method integrates the outputs from multiple retrievers using Reciprocal Rank Fusion (RRF), which allows the system to benefit from complementary retrieval signals. In our experiments, the hybrid configuration combining two dense retrievers and one sparse retriever yielded the best results (see Table 3).

For each strategy, the top-$k$ passages ($k = 30$) were collected as candidates for downstream processing. In the hybrid setup, each retriever contributed its own top-$k$ list. These were merged using RRF to produce a unified ranking, from which the final top-$k$ passages were selected. This fusion process not only enhances retrieval accuracy but also reduces reranking cost, as it avoids reranking all candidates from each retriever independently.

To further refine the fusion, we applied a weighted version of RRF, assigning importance scores to each retriever. The final score for a document $d$ is computed as:

$$\text{score}(d) = \sum_i \frac{w_i}{r + \text{rank}_i(d)},$$

where $w_i$ is the weight of retriever $i$, $\text{rank}_i(d)$ is the rank position of $d$ from retriever $i$, and $r = 10$ is a smoothing factor. The value $r = 10$ was selected based on empirical evaluation. For instance, with weights $w_{\text{sparse}} = 0.5$ and $w_{\text{dense}} = 0.4$, a document ranked 3rd by the sparse retriever and 1st by the dense retriever achieves a score of approximately 0.0749. In contrast, a document ranked 1st by the sparse retriever but only 10th by the dense retriever scores around 0.0655, highlighting the advantage of consistent relevance across retrievers.

Table 3: Comparison of model performance by query strategy

| Metric | Sparse | | | | Dense | | | | Hybrid (1 Dense, 1 Sparse) | | | | Hybrid (2 Dense, 1 Sparse) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | @3 | @5 | @10 | @15 | @3 | @5 | @10 | @15 | @3 | @5 | @10 | @15 | @3 | @5 | @10 | @15 |
| **Recall** | 0.4926 | 0.5682 | 0.6312 | 0.6570 | 0.5477 | 0.6308 | 0.7058 | 0.7346 | 0.5657 | 0.6606 | 0.7455 | 0.7813 | **0.5675** | **0.6707** | **0.7582** | **0.7958** |
| **MRR** | 0.6428 | 0.6486 | 0.6516 | 0.6521 | 0.7117 | 0.7203 | 0.7244 | 0.7260 | 0.7189 | 0.7286 | 0.7332 | 0.7342 | **0.7306** | **0.7414** | **0.7465** | **0.7479** |
| **nDCG** | 0.5619 | 0.5705 | 0.5925 | 0.6021 | 0.6150 | 0.6255 | 0.6490 | 0.6597 | 0.6348 | 0.6490 | 0.6775 | 0.6904 | **0.6400** | **0.6571** | **0.6861** | **0.7004** |

## 4.5 Reranking

To enhance the accuracy of retrieval results, a reranking stage is applied after the initial candidate chunks are retrieved. This process refines the ordering of passages by evaluating their semantic relevance to the query using powerful cross-encoder models.

First, the system selects the top-$K$ candidate chunks from the retrieval stage. Each candidate is then paired with the original user query to form a `(query, chunk)` input pair. These pairs are fed into a cross-encoder model, which jointly encodes both components and computes a relevance score reflecting their semantic alignment.

We experimented with several reranking models, including:

- `ViRanker` (`namdp-ptit/ViRanker`): a Vietnamese cross-encoder fine-tuned for semantic ranking.
- `bge-reranker-v2-m3`: a multilingual model supporting dense reranking across diverse languages.
- `Vietnamese_Reranker`: another model tailored for Vietnamese text relevance evaluation.

Once relevance scores are computed, the system reorders the candidate chunks accordingly and selects the top-$N$ for answer generation. Based on empirical tuning and resource constraints, we chose $N = 15$ as a practical balance between accuracy and efficiency.
*Note:* If no reranker is used, the system falls back to selecting the top-$N$ chunks directly from the original retrieval ranking.

Table 4: Comparison of reranker model performance

| Metric | No Reranker | | | | ViRanker | | | | bge-reranker-v2-m3 | | | | Vietnamese_Reranker | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | @3 | @5 | @10 | @15 | @3 | @5 | @10 | @15 | @3 | @5 | @10 | @15 | @3 | @5 | @10 | @15 |
| **Recall** | 0.4837 | 0.5715 | 0.6881 | 0.7569 | 0.5448 | 0.6290 | 0.7439 | 0.7895 | **0.5675** | **0.6707** | **0.7582** | **0.7958** | 0.5661 | 0.6589 | 0.7495 | 0.7952 |
| **MRR** | 0.6017 | 0.6138 | 0.6242 | 0.6275 | 0.6956 | 0.7037 | 0.7128 | 0.7142 | **0.7306** | **0.7414** | **0.7465** | **0.7479** | 0.7128 | 0.7219 | 0.7279 | 0.7298 |
| **nDCG** | 0.5171 | 0.5333 | 0.5780 | 0.6030 | 0.5990 | 0.6126 | 0.6543 | 0.6717 | **0.6400** | **0.6571** | **0.6861** | **0.7004** | 0.6225 | 0.6369 | 0.6683 | 0.6851 |

*bge-reranker-v2-m3* achieved the best performance, whereas the no-rerank baseline yielded significantly lower results.

## 4.6 Prompt Design for Answer Generation

Following the reranking stage, the system generates the final answer using a structured prompt tailored for legal accuracy, contextual grounding, and response consistency. The prompt is carefully constructed to guide the language model in producing reliable and legally sound outputs. It consists of the following key components:

- **Conversation History**: A record of the most recent question–answer exchanges is included to preserve dialogue coherence and enable contextual continuity in multi-turn interactions.
- **Contextual Evidence**: A curated set of legal text segments retrieved and reranked from the knowledge base. These chunks serve as the factual foundation upon which the answer is formulated.
- **User Query**: The current question posed by the user, typically reflecting a real-world legal concern or scenario requiring clarification.
- **Instructional Prompt (LLM Directive)**: A carefully phrased set of instructions guiding the large language model to generate responses that are legally precise, concise, and appropriately scoped. This may include constraints such as "respond only based on the provided context" or "cite specific articles if available."

This prompt formulation strategy ensures that the model's responses remain grounded in the retrieved legal content, while also adapting dynamically to user intent and conversational flow.

## 4.7 Contextual Memory

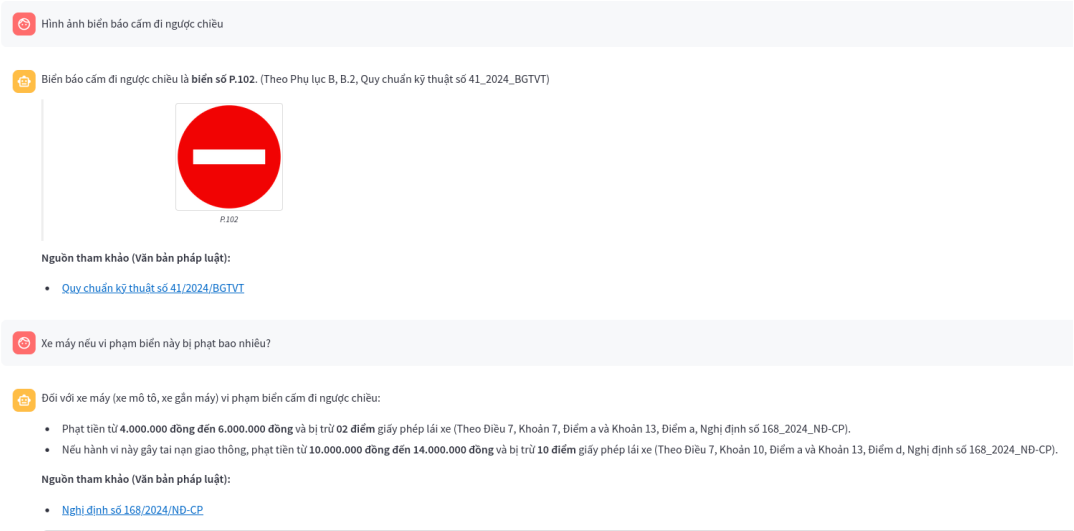



Figure 7: Contextual Memory

To maintain consistency in responses, the system stores a history of recent query-answer pairs (about 4 pairs). It also supports multi-turn conversations by leveraging this interaction history as context for follow-up questions, as illustrated in Figure 7.

# 5 Evaluation and Results

## 5.1 Experiment Settings

To evaluate the performance of different retrieval and reranking strategies, we conducted a series of experiments under consistent baseline configurations. Unless otherwise specified, all experiments were performed using the following settings:

- **Question Type**: We used the *expanded query* format by default, which provides a semantically enriched formulation of the original user question to enhance retrieval effectiveness.
- **Reranker**: The `bge-reranker-v2-m3` model was selected as the default reranker due to its strong empirical performance across legal tasks.
- **Retrieval Strategy**: Multiple strategies were tested, including:
  - **Dense-only**, using the best-performing dense model: `bge-m3`.
  - **Hybrid (1 Dense + 1 Sparse)**: combining `bge-m3` with BM25.
  - **Hybrid (2 Dense + 1 Sparse)**: combining `bge-m3`, `vn-law-embedding`, and BM25.
- **Final Configuration for Benchmarking**: The full hybrid strategy using `bge-m3+vn-law-embedding+BM25` was used in most benchmark experiments. This combination yielded the highest overall performance in retrieval recall and MRR.

## 5.2 Evaluation Dataset

300 manually compiled questions about the Road Traffic Law. Ground truth: IDs/information of the relevant chunks.

```
[ {"query_id": "eval_001",
   "query": "Mức phạt không đội mũ BH xe máy 2025?",
   "relevant_chunk_ids": ["168_2024_NĐ-CP_161", ...] } ]
```

Using 282 out of a total of 600 questions from the question set compiled by the Traffic Police Department and effective from June 1, 2025, for evaluation.

**Câu 9. Trong nhóm các phương tiện giao thông đường bộ dưới đây, nhóm phương tiện nào là xe thô sơ?**

1. Xe đạp, xe đạp máy, xe đạp điện; xe xích lô; xe lăn dùng cho người khuyết tật; xe vật nuôi kéo và các loại xe tương tự.

2. Xe đạp (kể cả xe đạp máy, xe đạp điện), xe gắn máy, xe cơ giới dùng cho người khuyết tật và xe máy chuyên dùng.

3. Xe ô tô, máy kéo, rơ moóc hoặc sơ mi rơ moóc được kéo bởi xe ô tô, máy kéo.

Figure 8: An Example of questions from the 600 question set

## 5.3 Evaluation method

**For retrieval evaluation**, the system is assessed using common information retrieval metrics such as **Recall@k**, **MRR@k**, and **nDCG@k** to measure the accuracy and relevance of the retrieved text chunks.

Meanwhile, answer evaluation is conducted through two approaches: *semi-automatic evaluation* using the **GPT 4.o** language model to analyze the appropriateness of the answers in relation to the question context, and *automatic evaluation* by checking the chatbot's ability to correctly answer standardized multiple-choice questions. **Recall@k** is defined as:

$$\text{Recall@k} = \frac{|S_q \cap R_q|}{|R_q|}$$

It measures the proportion of relevant documents ($R_q$) that are actually retrieved in the top-$k$ results ($S_q$). This metric is important for evaluating the *coverage* of the search system — i.e., whether it misses any relevant documents.

**MRR (Mean Reciprocal Rank)** is defined as:

$$\text{MRR} = \frac{1}{|Q|} \sum_{q=1}^{|Q|} \frac{1}{\text{rank}_q}$$

MRR focuses on the position of the first relevant result. A higher value indicates that the correct result appears earlier in the ranked list.

**nDCG@k (Normalized Discounted Cumulative Gain)** is defined as:

$$\text{nDCG@k} = \frac{1}{\text{IDCG@k}} \sum_{i=1}^{k} \frac{\text{rel}_i}{\log_2(i+1)}$$

nDCG@k prioritizes placing highly relevant documents at the top of the ranking. It applies a discount factor based on position to reflect the quality of the ranking order.

Both MRR and nDCG@k help evaluate the *ranking quality* of the results, not just their quantity. These metrics are typically calculated at thresholds $k = 3, 5, 10$ for comparison purposes. In this evaluation, we also include $k = 15$ because some legal articles contain a large number of points.

**For answers evaluation**, answers from the road traffic chatbot are paired with relevant legal text chunks and evaluated using a standardized prompt. The evaluation is semi-automated with the GPT 4.o model, which simulates a Vietnamese legal expert assessing responses based on current road traffic laws. The scoring is guided by four clear criteria on a 1–5 scale: (1) accuracy and completeness, (2) relevance, (3) clarity and language, and (4) proper legal citation.

Table 5: Average score of criteria

| Criteria | Accuracy and Completeness | Relevance | Structure and Language | Citation Ability |
|---|---|---|---|---|
| Average | 3.041 | 4.048 | 4.696 | 3.027 |

Additionally, in our second evaluation method, we used 282 multiple-choice questions selected from the official 600-question set used for the driving license exam. The chatbot correctly answered 267 questions, achieving an accuracy rate of 94.68%.

# 6 Conclusion and Future Work

This project presents a legal information retrieval system that **combines large language models (LLMs) with a hybrid retrieval approach** to provide accurate and efficient access to Vietnam's traffic laws. **The integration of dense and sparse retrieval methods** allows the system to **handle both keyword-based and natural language queries effectively**. By enabling fast and reliable access to legal content, the system helps reduce the risk of unintentional legal violations due to lack of information.

In practice, the system shows strong potential to assist citizens especially students and young people who frequently participate in traffic yet often lack sufficient legal awareness in understanding and complying with traffic regulations.

For future work, we plan to **enhance the system with voice-based interaction, support for multiple languages, and expansion beyond traffic laws to other areas of legal knowledge.** These improvements aim to make the system **more accessible, user-friendly, and applicable in a broader legal context.**

# References


[1] National Traffic Safety Committee. "Traffic Accident Report 2024."

[2] Evaluation Metrics for Search and Recommendation Systems. Available at: `https://weaviate.io/blog/retrieval-evaluation-metrics`

[3] Vietnam Law Library. "Official Portal for Vietnamese Legal Documents.". Available at: `https://thuvienphapluat.vn/`

[4] LLM-Engineers-Handbook. Available at: https://github.com/PacktPublishing/LLM-Engineers-Handbook?tab=readme-ov-file

[5] RAG_Techniques. Available at: https://github.com/NirDiamant/RAG_Techniques

[6] Wikipedia. *Road signs in Vietnam*. Available at: https://en.wikipedia.org/wiki/Road_signs_in_Vietnam

[7] Hugging Face. *RAG — Retrieval-Augmented Generation*. Available at: https://huggingface.co/docs/transformers/model_doc/rag