Chatbot Tra cứu Luật Giao thông Đường bộ Việt Nam

Đồ án môn học CS431.P22

Thực hiện bởi:

Lê Minh Nhựt[1] Trần Đình Khánh Đăng[1] Thái Ngọc Quân[1]

Giảng viên hướng dẫn: TS. Nguyễn Vinh Tiệp[1]

[1]Khoa Khoa học Máy tính
Trường Đại học Công nghệ Thông tin - ĐHQG TP.HCM

Báo cáo Tiến độ – Tháng 5, 2025

Mục lục Tổng quát

Nội dung

Giới thiệu: Bối cảnh và Thách thức

Bối cảnh Luật GTĐB Việt Nam

- Hệ thống pháp luật quan trọng, ảnh hưởng trực tiếp an toàn & trật tự giao thông.
- Phức tạp: nhiều loại văn bản (Luật, Nghị định, Thông tư...).
- Thường xuyên cập nhật, sửa đổi.

Thách thức Tra cứu Hiện tại

- Khó khăn tìm kiếm thông tin nhanh chóng, chính xác.
- Đọc hiểu văn bản gốc tốn thời gian, cần kiến thức chuyên môn.
- Nguồn online phân tán, thiếu tin cậy, có thể lỗi thời.

Giới thiệu: Động lực và Giải pháp

Động lực Thực hiện

- **Nhu cầu cao:** Nhiều người tham gia giao thông chưa nắm vững luật.
- **Hậu quả:** Vi phạm không cố ý, tranh cãi khi xử lý, tiềm ẩn mất an toàn.

Giải pháp Đề xuất: Chatbot Luật GTĐB

- Xây dựng Chatbot thông minh, chuyên biệt cho Luật GTĐB Việt Nam.
- Cung cấp kênh tra cứu tập trung, tiện lợi, nhanh chóng và thân thiện.

Giới thiệu: Động lực và Giải pháp

Hình 1: Chatbot tra cứu luật giao thông đường bộ.

Giới thiệu: Mục tiêu Dự án

Mục tiêu Phát triển Chatbot

- **Trả lời câu hỏi:** Về quy tắc, mức phạt, thủ tục cơ bản...
- **Đảm bảo Chính xác:** Dựa trên văn bản pháp luật hiện hành, nguồn tin cậy.
- **Thân thiện Người dùng:** Giao diện tương tác đơn giản, tự nhiên, trả lời mạch lạc.
- **Truy xuất Nguồn:** Chỉ rõ điều khoản, văn bản gốc liên quan.

Nội dung

Dữ liệu: Tầm quan trọng và Nguồn gốc

Tầm quan trọng của Dữ liệu

- **Nền tảng cốt lõi:** Chất lượng & độ tin cậy dữ liệu quyết định độ chính xác của Chatbot.
- Dữ liệu sai/lỗi thời $\rightarrow$ tư vấn sai, hậu quả nghiêm trọng.

Nguồn Dữ liệu Chính

- **Thư viện Pháp luật** (thuvienphapluat.vn): Uy tín, cập nhật, phổ biến.

Dữ liệu: Phạm vi và Thu thập

Phạm vi Dữ liệu

- Lĩnh vực: **Trật tự, An toàn Giao thông Đường bộ**.
- Số lượng: **29 văn bản** chọn lọc (Luật, Nghị định, Thông tư...).
- Cơ quan ban hành: Quốc hội, Chính phủ, Bộ Công an, Bộ GTVT,...
- Đảm bảo **tính cập nhật** (VD: Luật 36/2024/QH15, Nghị định 168/2024/NĐ-CP).

Thu thập Dữ liệu

- Sử dụng **Web Crawling** tự động tải nội dung text từ Thư viện Pháp luật.

Dữ liệu: Tiền xử lý và Cấu trúc Dữ liệu

Quy trình Tiền xử lý

- **Phân đoạn (Chunking):** Chia text thành các đoạn (chunks) nhỏ, giữ ngữ cảnh.
- **Trích xuất Metadata & Ngữ cảnh:** Gắn thông tin (số hiệu VB, ngày hiệu lực, URL, vị trí: Chương, Điều, Khoản, Điểm).
- **Định dạng JSON:** Lưu từng chunk dưới dạng JSON object.

Ví dụ Cấu trúc JSON của một Chunk


```
{
  "id": "36_2024_QH15_chunk_150",
  "text": "Chương II QUY TẮC GIAO THÔNG...\n...",
  "metadata": {
    "source": "Luật số 36_2024_QH15",
    "effective_date": "1-1-2025",
    "url": "https://thuvienphapluat.vn/...",
    "context": {
      "diem": "h", "khoan": "6",
      "dieu": "14", "chuong": "II"
    }
  }
}
```


Nội dung

Quy trình: 1. Chuẩn bị Dữ liệu & Embedding

Mục tiêu: Chuyển đổi văn bản pháp luật thành vector để tìm kiếm hiệu quả hơn (thay vì so khớp từ khóa đơn thuần).

- **Đọc Dữ liệu Nguồn:**
  - Tải file JSON chứa các đoạn văn bản (chunked) kèm thông tin mô tả (metadata như điều khoản, văn bản gốc...).
- **Vector Hóa Ngữ Nghĩa (Embedding):**
  - **Khái niệm:** Mỗi đoạn văn bản sẽ được chuyển thành một vector số thực, thể hiện ý nghĩa tổng quát của đoạn văn.
  - **Ví dụ:**
    - "Xe không có biển số sẽ bị xử phạt" $\rightarrow$ `[0.13, -0.02, ..., 0.27]`
    - Các đoạn văn có ý nghĩa tương tự sẽ nằm gần nhau trong không gian vector.
  - **Công cụ:** Sử dụng SentenceTransformer `truro7/vn-law-embedding` – một mô hình được huấn luyện chuyên biệt cho tiếng Việt và văn bản pháp lý.

Quy trình: 2. Xây dựng Chỉ mục Tìm kiếm

- **Dense Index:**
  - Dựa trên vector embedding đã tạo từ bước trước.
  - Dùng thư viện FAISS để tìm kiếm gần đúng nhanh (Approximate Nearest Neighbors).
  - Cấu hình: `IndexFlatL2` – dùng khoảng cách Euclidean.
  - Mỗi vector được gắn với thông tin chunk gốc (ID, tiêu đề, metadata).
- **Sparse Index:**
  - Dựa trên mô hình truyền thống BM25 (TF-IDF cải tiến).
  - Phân tách từ (tokenize), loại bỏ stopwords, tính độ quan trọng của từ.
  - Hiệu quả tốt với truy vấn ngắn, từ khóa rõ ràng.
- **Hybrid Retriever:**
  - Kết hợp cả dense và sparse để tận dụng điểm mạnh của hai phương pháp.
  - Cho phép chọn cấu hình:
    - `sparse only`: truyền thống, phù hợp khi truy vấn rõ ràng.
    - `dense only`: tốt hơn với câu truy vấn tự nhiên, ngữ cảnh mơ hồ.
    - `hybrid`: trộn kết quả từ cả hai (có thể dùng Rank Fusion ở bước sau).

Quy trình: 3. Tiếp nhận & Phân tích Truy vấn

1. **Nhận Truy vấn:**
   - Giao diện người dùng nhập câu hỏi dạng tự nhiên (VD: "Lái xe không có bằng bị phạt bao nhiêu?").
2. **Phân tích Ngữ nghĩa & Ngữ cảnh:**
   - Dùng mô hình ngôn ngữ lớn (LLM - Gemini) hiểu mục đích, hành vi, và thực thể liên quan trong truy vấn.
3. **Kiểm tra Liên quan (Query Filtering):**
   - Nếu truy vấn không liên quan đến luật hoặc ngoài phạm vi: trả lời từ chối hoặc chuyển hướng.
   - Nếu hợp lệ: tiếp tục truy xuất thông tin.
4. **Mở rộng Truy vấn (Query Expansion):**
   - Tạo thêm các biến thể từ câu gốc (dạng rút gọn, đồng nghĩa).
   - Mục tiêu: tăng độ bao phủ (Recall) trong truy xuất tài liệu.
   - VD: "bị xử phạt"$\rightarrow$ "mức phạt", "bị phạt bao nhiêu tiền", "xử lý hành vi này như thế nào",...

Quy trình: 4. Truy xuất Thông tin

Tìm các đoạn văn bản (chunk) tiềm năng từ kho dữ liệu.

- **Lựa chọn Chiến lược Truy xuất:**
  - Dựa trên **query gốc**, **tóm tắt** hoặc **biến thể** sinh từ LLM.
  - Chọn chế độ **Dense** (vector), **Sparse** (BM25), hoặc **Hybrid**.
- **Thực hiện Truy xuất:**
  - **Dense Search:** Dùng FAISS để tìm các vector gần nhất trong không gian embedding.
  - **Sparse Search:** Dùng BM25 để truy vấn theo từ khóa.
  - **Hybrid Search:** Kết hợp cả hai kết quả bằng kỹ thuật **Rank Fusion (RRF)**:
    - Gộp các bảng xếp hạng, tính điểm dựa vào thứ hạng.
    - Ưu tiên kết quả xuất hiện trong nhiều phương pháp.
- **Thu thập Kết quả Ứng viên:**
  - Trả về danh sách các chunk tiềm năng cho bước Reranking hoặc sinh câu trả lời.

Quy trình: 5. Xếp hạng lại (Reranking)

Tinh chỉnh thứ tự các đoạn văn bản ứng viên để nâng độ chính xác.

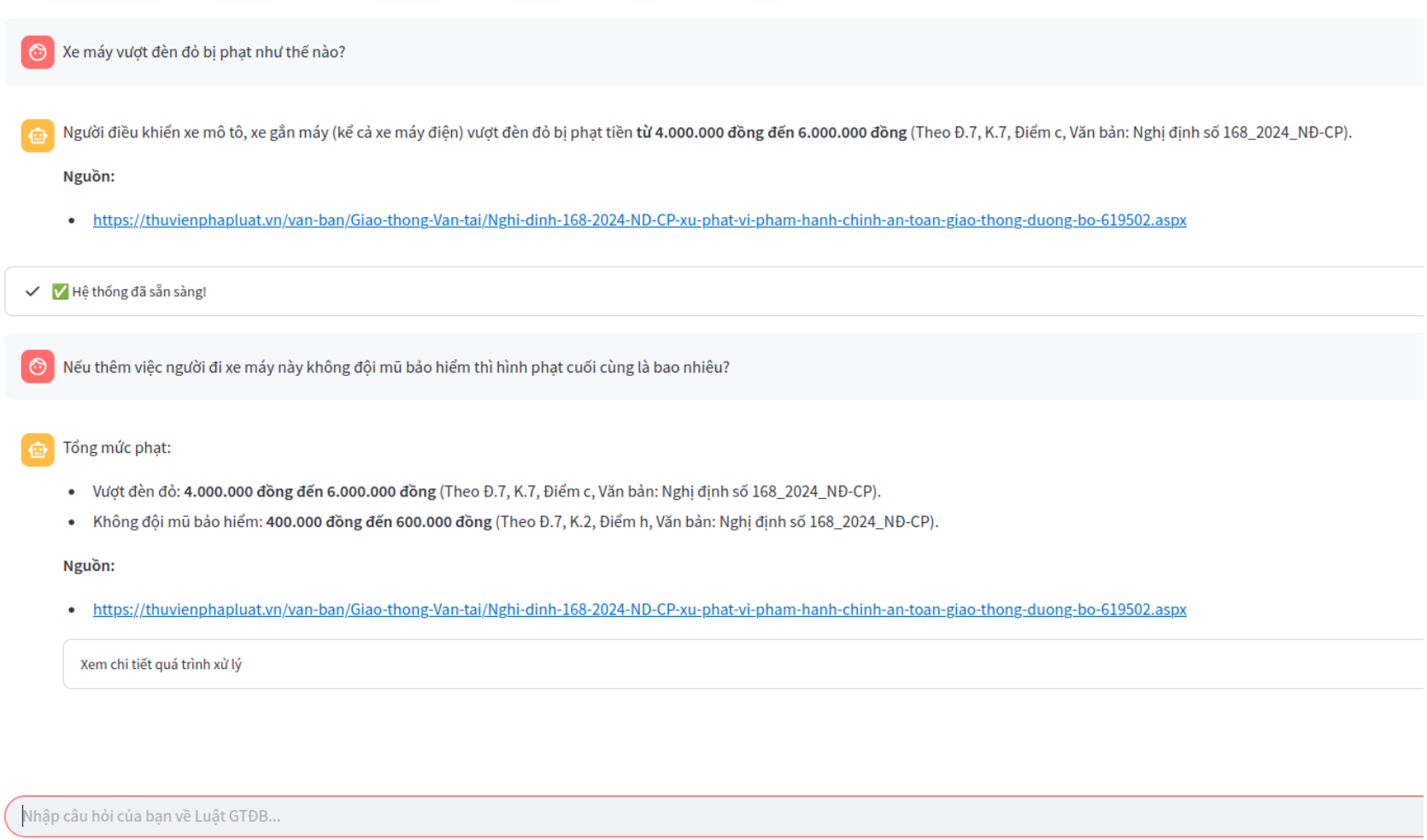
- **Chọn ứng viên ban đầu:**
  - Lấy Top **N** chunk từ kết quả truy xuất ban đầu.
- **Áp dụng CrossEncoder để đánh giá lại:**
  - Kết hợp từng cặp **(query, chunk)** làm đầu vào cho mô hình.
  - Mô hình (CrossEncoder như `namdp-ptit/ViRanker`) trả về **score** phản ánh độ liên quan.
  - Ưu điểm: chính xác hơn so với truy xuất thô.
- **Chọn kết quả cuối cùng:**
  - Sắp xếp các chunk theo điểm số.
  - Lấy Top **K** chunk (ví dụ $K = 10$) để đưa vào LLM sinh câu trả lời.

Nếu không dùng Reranker: Lấy trực tiếp Top K từ kết quả truy xuất ban đầu (FAISS hoặc BM25).

Quy trình: 6. Tạo & Hoàn thiện Câu trả lời

- **RAG - LLM giai đoạn 2:**
  - Tạo **prompt đầy đủ** gồm:
    - Truy vấn người dùng (query).
    - Lịch sử hội thoại trước đó (nếu có).
    - Các đoạn văn bản liên quan (chunks từ truy xuất).
    - Hướng dẫn phong cách trình bày (ví dụ: ngắn gọn, dễ hiểu, văn phong pháp lý...).
  - Gọi LLM (Gemini) để sinh ra câu trả lời mạch lạc, chính xác.
- **Hậu xử lý (Post-processing):**
  - Trích xuất và định dạng trích dẫn (citation) $\rightarrow$ liên kết đến tài liệu gốc.
  - Kiểm tra sự đầy đủ của câu trả lời (có đúng, rõ, nguồn không).
  - Gửi câu trả lời cùng link nguồn về giao diện người dùng.
  - Ghi lại lịch sử truy vấn phản hồi

Quy trình: 6. Ghi nhớ ngữ cảnh



Hình 3: Khả năng nhớ ngữ cảnh

Nội dung

Đánh Giá: Quy trình và Bộ dữ liệu

Mục tiêu Đánh giá

- Đo hiệu quả Retrieval trong RAG.
- Khả năng tìm đúng/chọn đủ chunk liên quan.

Bộ dữ liệu Đánh giá

- 96 câu hỏi tổng hợp thủ công về Luật GTĐB.
- Ground truth: chunk ID liên quan.

Ví dụ dữ liệu đánh giá

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

Đánh Giá: Precision, Recall, F1

System trả về S_q top-k, ground truth R_q.

- **Precision@k:** $\frac{|S_q \cap R_q|}{k}$
- **Recall@k:** $\frac{|S_q \cap R_q|}{|R_q|}$
- **F1@k:** $2\frac{\text{P@k} \times \text{R@k}}{\text{P@k} + \text{R@k}}$

(Tính tại k=3,5,10)

Đánh Giá: MRR và nDCG

- **MRR:** $\frac{1}{|Q|}\sum_q \frac{1}{\text{rank}_q}$
- **nDCG@k:**

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

(Tính tại k=3,5,10)

Đánh Giá: Kết quả Retrieval

Bảng 1: So sánh hiệu suất giữa truy vấn *Đơn giản*, *Tổng quát* và *Sâu*. Cấu hình chung: Hybrid Search + Rerank

Độ đo	**Đơn giản**			**Tổng quát**			**Sâu**		
	@3	@5	@10	@3	@5	@10	@3	@5	@10
Precision	0.3611	0.2708	0.1740	**0.3692**	**0.3011**	**0.1892**	0.0000	0.0000	0.0000
Recall	**0.5365**	0.6059	0.7255	0.5328	**0.6586**	**0.7501**	0.0000	0.0000	0.0000
F1-score	0.3941	0.3423	0.2626	**0.3986**	**0.3798**	**0.2814**	0.0000	0.0000	0.0000
MRR	0.5851	0.5924	0.6031	**0.6075**	**0.6215**	**0.6294**	0.0000	0.0000	0.0000
nDCG	0.5387	0.5571	0.6050	**0.5401**	**0.5860**	**0.6254**	0.0000	0.0000	0.0000

Đánh Giá: Kết quả Retrieval

Bảng 2: So sánh hiệu suất giữa khi **sử dụng Reranker** và khi **không sử dụng**. Cấu hình chung: Truy vấn Tổng quát + Hybrid Search

Độ đo	Không Reranker			Có Reranker		
	@3	@5	@10	@3	@5	@10
Precision	0.2660	0.2191	0.1479	**0.3692**	**0.3011**	**0.1892**
Recall	0.3801	0.4867	0.6331	**0.5328**	**0.6586**	**0.7501**
F1-score	0.2876	0.2789	0.2244	**0.3986**	**0.3798**	**0.2814**
MRR	0.4628	0.4899	0.5094	**0.6075**	**0.6215**	**0.6294**
nDCG	0.3947	0.4334	0.4897	**0.5401**	**0.5860**	**0.6254**

Đánh Giá: Kết quả Retrieval

Bảng 3: So sánh hiệu suất giữa **Sparse**, **Dense** và **Hybrid Search**. Cấu hình chung: Truy vấn Tổng quát + Rerank

Độ đo	**Sparse**			**Dense**			**Hybrid**		
	@3	@5	@10	@3	@5	@10	@3	@5	@10
Precision	0.3082	0.2495	0.1602	0.3014	0.2383	0.1426	**0.3692**	**0.3011**	**0.1892**
Recall	0.4622	0.5671	0.6719	0.4443	0.5445	0.6283	**0.5328**	**0.6586**	**0.7501**
F1-score	0.3387	0.3199	0.2433	0.3298	0.3054	0.2182	**0.3986**	**0.3798**	**0.2814**
MRR	0.5663	0.5728	0.5832	0.5089	0.5238	0.5353	**0.6075**	**0.6215**	**0.6294**
nDCG	0.4775	0.5148	0.5588	0.4572	0.4920	0.5243	**0.5401**	**0.5860**	**0.6254**

Xin Cảm Ơn!

Q&A

Tài liệu tham khảo I

- Luật Giao Thông 2025 và các nghị định thông tư hướng dẫn mới nhất.
- vn-law-embedding on Hugging Face.
- ViRanker on Hugging Face.
- LLM Engineers Handbook.
- AI Engineering Book.