TRƯỜNG ĐẠI HỌC CẦN THƠ
TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

LUẬN VĂN TỐT NGHIỆP
NGÀNH KHOA HỌC MÁY TÍNH

Đề tài

XÂY DỰNG HỆ THỐNG TỰ ĐỘNG TRÍCH XUẤT VÀ HỎI - ĐÁP THÔNG TIN TỪ VĂN BẢN PHÁP LUẬT
(Tên tiếng Anh)

Sinh viên: Lê Tuấn Đạt
Mã số: B2113328
Khóa: 47
Giảng viên hướng dẫn: TS. Trần Nguyễn Minh Thư

Cần Thơ, 07/2025

TRƯỜNG ĐẠI HỌC CẦN THƠ
TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

LUẬN VĂN TỐT NGHIỆP
NGÀNH KHOA HỌC MÁY TÍNH

Đề tài

XÂY DỰNG HỆ THỐNG TỰ ĐỘNG TRÍCH XUẤT VÀ HỎI - ĐÁP THÔNG TIN TỪ VĂN BẢN PHÁP LUẬT
(Tên tiếng Anh)

Giảng viên hướng dẫn
TS. Trần Nguyễn Minh Thư

Sinh viên thực hiện
Họ và tên: Lê Tuấn Đạt
Mã số: B2113328
Khóa: 47

Cần Thơ, 07/2025

LỜI CẢM ƠN

Để có được bài niên luận này, em xin được bày tỏ lòng biết ơn chân thành và sâu sắc đến Cô Trần Nguyễn Minh Thư – người đã trực tiếp tận tình hướng dẫn, giúp đỡ em. Trong suốt quá trình thực hiện niên luận, nhờ những sự chỉ bảo và hướng dẫn quý giá đó mà bài niên luận này được hoàn thành một cách tốt nhất.

Em cũng xin gửi lời cám ơn chân thành đến các Thầy Cô Giảng viên Đại học Cần Thơ, đặc biệt là các Thầy Cô ở Trường CNTT & TT, những người đã truyền đạt những kiến thức quý báu trong thời gian qua.

Em cũng xin chân thành cảm ơn bạn bè cùng với gia đình đã luôn động viên, khích lệ và tạo điều kiện giúp đỡ trong suốt quá trình thực hiện để em có thể hoàn thành bài niên luận một cách tốt nhất.

Tuy có nhiều cố gắng trong quá trình thực hiện niên luận, nhưng không thể tránh khỏi những sai sót. Em rất mong nhận được sự đóng góp ý kiến quý báu của quý Thầy Cô và các bạn để bài niên luận hoàn thiện hơn.

Cần Thơ, ngày tháng 07 năm 2025

Người viết

(Ký và ghi họ tên)

Lê Tuấn Đạt

MỤC LỤC

DANH MỤC HÌNH ẢNH

DANH MỤC BẢNG BIỂU

DANH MỤC TỪ VIẾT TẮT

STT	Từ viết tắt (Abbreviation)	Từ viết đầy đủ (Origin word)
1	VPPL	Văn bản quy phạm pháp luật
1	LLMs	Large Language Models
2	QA	Question Answering
3	NLP	Natural Language Processing
4	GPT	Generative Pre-trained Transformer
5	BERT	Bidirectional Encoder Representations from Transformers
6	T5	Text-To-Text Transfer Transformer
7	PDF	Portable Document Format
8	SVM	Support Vector Machine
9	CRF	Conditional Random Fields
10	RNN	Recurrent Neural Network
11	LSTM	Long-Short Term Memory
12	GRU	Gated Recurrent Unit Networks
12	TF-IDF	Term Frequency-Inverse Document Frequency
13	BiDAF	Bi-Directional Attention Flow
14	DrQA	Document Reader for Question Answering

15	ViT5	Vietnamese Text-To-Text Transfer Transformer
16	OCR	Optical Character Recognition
17		

ABSTRACT

TÓM TẮT

I. PHẦN GIỚI THIỆU

1. Đặt vấn đề

Trong bối cảnh dữ liệu văn bản ngày càng gia tăng về mặt số lượng và độ phức tạp, việc truy xuất thông tin một cách hiệu quả từ các tài liệu như văn bản quy phạm pháp luật (VPPL) trở thành một yêu cầu thiết yếu. Các văn bản như nghị định, nghị quyết, thông tư,… thường mang tính chất kỹ thuật, có độ dài lớn, cấu trúc phức tạp, và đòi hỏi độ chính xác cao trong quá trình tiếp cận và trích xuất thông tin. Điều này khiến cho không chỉ người dân mà cả cán bộ hành chính cũng gặp nhiều khó khăn trong việc tra cứu, hiểu và sử đúng nội dung các văn bản này. Việc ghi nhớ toàn bộ nội dung, hoặc nhanh chóng tìm ra phần thông tin phù hợp khi cần thiết, vẫn là thách thức lớn trong thực tiễn áp dụng.

Khó khăn này càng thể hiện rõ hơn trong bối cảnh tiếp xúc cử tri - một hoạt động chính trị định kỳ, nơi đại biểu Quốc hội hoặc Hội đồng Nhân dân (HĐND) tiếp nhận ý kiến, phản ánh và kiến nghị từ người dân. Cử tri thường đặt ra nhiều câu hỏi liên quan đến các quy định pháp luật, đặc biệt xoay quanh những vấn đề thời sự có sự ảnh hưởng sâu rộng, như việc sáp nhập đơn vị hành chính. Hiện tại, đây là chủ đề nóng, thu hút nhiều sự quan tâm, kéo theo hàng loạt câu hỏi như: "*Sổ đỏ có phải làm lại không?*", "*Tôi đang hưởng chính sách ở phường cũ thì sau sáp nhập thế nào?*", "*Có cần phải đi làm giấy tờ lại hay không?*". Những câu hỏi này đòi hỏi phản hồi chính xác, căn cứ vào văn bản pháp luật cụ thể. Tuy nhiên, không phải lúc nào đại biểu hay cán bộ hỗ trợ cũng có đủ thời gian và công cụ để nhanh chóng tra cứu, đối chiếu, và trích dẫn chính xác các điều khoản pháp lý liên quan. Điều này dễ dẫn đến việc trả lời thiếu chính xác, gây hiểu nhầm, hoặc làm giảm hiệu quả của buổi tiếp xúc.

Hình 1. Quá trình tiếp xúc cử tri (Nguồn: baochinhphu.vn)

Song song đó, sự bùng nổ của các mô hình trí tuệ nhân tạo hiện đại - đặc biệt là những mô hình xử lý ngôn ngữ như GPT (Generative Pre-trained Transformer) - đang ngày càng chứng minh vai trò hữu ích trong việc "hiểu" và xử lý văn bản ngôn ngữ tự nhiên. Các hệ thống hỏi - đáp tự động (QA - Question Answering) tích hợp mô hình ngôn ngữ lớn (LLMs - Large Language Models) có khả năng đọc hiểu, tóm tắt nội dung, và phản hồi theo

ngữ cảnh một cách nhanh chóng, tự nhiên, và thông minh. Dù vậy, hầu hết các mô hình hiện có vẫn chủ yếu được huấn luyện cho tiếng Anh, hoặc chưa tinh chỉnh phù hợp với đặc thù của VPPL tiếng Việt - cả về ngôn ngữ, cấu trúc và định dạng văn bản (như PDF - Portable Document Format).

Từ thực tiễn đó, đề tài "Xây dựng hệ thống hỏi - đáp văn bản pháp luật" ra đời với mục tiêu ứng dụng các kỹ thuật mới nhất trong lĩnh vực xử lý ngôn ngữ tự nhiên (NLP - Natural Language Processing), nhằm hỗ trợ người dùng - đặc biệt là đại biểu, cán bộ hoặc người dân - có thể truy xuất thông tin pháp luật một cách thông minh, chính xác và thuận tiện, phục vụ tốt hơn cho công tác đối thoại chính sách, tiếp xúc cử tri và áp dụng pháp luật vào thực tiễn.

2. Lịch sử giải quyết vấn đề

Việc tra cứu, trích xuất và hỏi đáp thông tin từ văn bản pháp luật luôn là một thách thức lớn, đặc biệt đối với đại biểu Quốc hội, Hội đồng Nhân dân và cán bộ hỗ trợ trong quá trình tiếp xúc cử tri. Các văn bản pháp luật thường có độ dài lớn, cấu trúc phức tạp và ngôn ngữ chuyên ngành, khiến việc tìm kiếm, đối chiếu chính xác các điều khoản liên quan để trả lời câu hỏi của người dân trở nên khó khăn và tốn nhiều thời gian. Trong thực tế, đại biểu và cán bộ thường phải đọc kỹ từng văn bản dài để tìm ra thông tin phù hợp, điều này ảnh hưởng trực tiếp đến hiệu quả và độ chính xác của các buổi tiếp xúc.

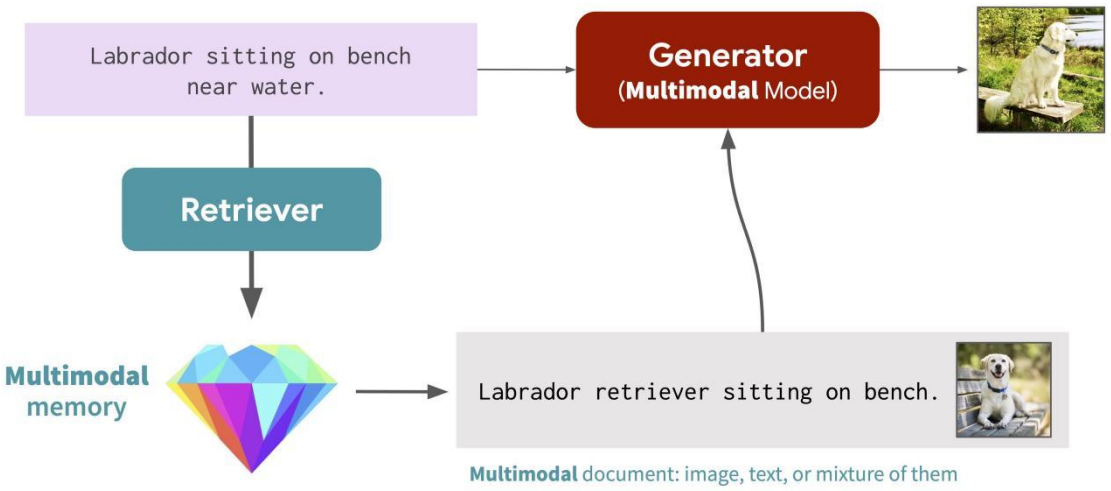
Trước đây, công cụ hỗ trợ chính là các hệ thống tìm kiếm văn bản truyền thống như *vbpl.vn*, *thuvienphapluat.vn* hay *luatvietnam.vn*, nơi người dùng - chủ yếu là cán bộ, đại biểu - sử dụng truy vấn từ khóa và bộ lọc để tìm kiếm các văn bản liên quan. Tuy nhiên, các hệ thống này chỉ giúp truy xuất văn bản theo từ khóa, không có khả năng hiểu ngữ cảnh hay tự động trích xuất thông tin cụ thể từ đoạn văn dài. Do đó, cán bộ và đại biểu vẫn phải tự mình đọc và phân tích văn bản, gây mất nhiều thời gian và có thể dẫn đến sai sót khi trả lời cử tri.

Hình 2. Giao diện tìm kiếm văn bản pháp luật với bộ lọc từ khóa và thông tin hành chính (Nguồn: vbpl.vn)

Song song đó, việc hỗ trợ trực tiếp từ các chuyên gia pháp lý hoặc luật sư vẫn được sử dụng, nhưng phương pháp này không thể đáp ứng nhu cầu xử lý nhanh, quy mô lớn trong các buổi tiếp xúc cử tri. Do vậy, nhu cầu phát triển các công cụ hỗ trợ tự động, thông minh, giúp đại biểu và các bộ nhanh chóng tra cứu, đối chiếu và trích xuất thông tin pháp luật chính xác là rất cấp thiết.

Về công nghệ, các bước đầu tiên tập trung vào các giải pháp dựa trên luật định (rule-based), sử dụng biểu thức chính quy và từ điển thuật ngữ để nhận diện thông tin trong văn bản. Tiếp đến, các mô hình học máy cổ điển như Naive Bayes, SVM và CRF được áp dụng để phân loại văn bản và trích xuất thực thể, cải thiện phần nào hiệu quả tra cứu. Tuy nhiên, những phương pháp này vẫn phụ thuộc nhiều vào bước tiền xử lý và khó xử lý các văn bản pháp luật dài, phức tạp.

Bước đột phá xuất hiện khi các mô hình học sâu (Deep Learning) và kiến trúc Transformer như BERT, GPT, T5 ra đời, nâng cao khả năng hiểu ngữ nghĩa sâu sắc và sinh ngôn ngữ tự nhiên. Các kỹ thuật như retrieval-augmented generation (RAG) kết hợp truy xuất thông tin và sinh câu trả lời đã mở ra hướng phát triển mới cho các hệ thống hỏi đáp pháp luật tự động, giúp đại biểu và cán bộ hỗ trợ trả lời nhanh chóng, chính xác các câu hỏi của cử tri dựa trên văn bản gốc.

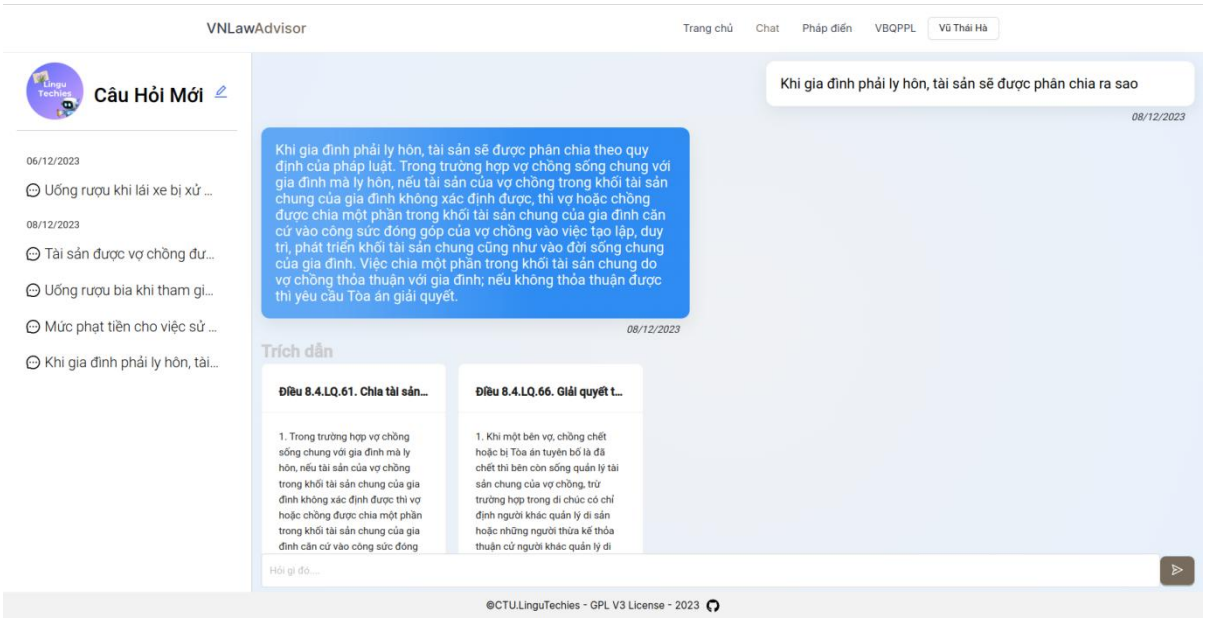


Hình 3. Mô phỏng kỹ thuật RAG (Nguồn: blog.payrollschedule.net)

Trên thế giới, nhiều dự án và hệ thống AI hỗ trợ pháp luật đã được phát triển nhằm tự động hóa quá trình tra cứu và phân tích văn bản pháp luật. Ví dụ, Siren Analytics sử dụng kỹ thuật RAG kết hợp mô hình ngôn ngữ lớn để tự động trích xuất và phân tích các văn bản luật quốc tế, hỗ trợ luật sư và chuyên gia pháp lý trong việc nghiên cứu và tư vấn. Các AI Expert Agents chuyên sâu về luật quốc tế cũng được triển khai để phân tích công ước, án lệ và tài liệu pháp lý phức tạp, giúp nâng cao hiệu quả nghiên cứu và ra quyết định. Ngoài ra, các hội thảo quốc tế như Legal Information Retrieval meets Artificial Intelligence (LIRAI) tập trung phát triển các phương pháp kết hợp AI và truy xuất thông tin pháp lý, giải quyết các thách thức về độ chính xác và khả năng giải thích trong môi trường phức tạp. Một số chatbot pháp lý tiên tiến cũng áp dụng kỹ thuật RAG để cung cấp câu trả lời chính xác dựa trên tài liệu

pháp luật chính thống, đồng thời hạn chế lỗi do mô hình sinh thông tin không chính xác (hallucination).

Tại Việt Nam, các mô hình ngôn ngữ thuần Việt như PhoBERT, ViT5 và Vietnamese-SBERT đã được phát triển, hỗ trợ hiệu quả trong các tác vụ xử lý văn bản tiếng Việt. Một ví dụ điển hình là dự án VN-Law-Advisor do nhóm CTU-LinguTechies phát triển, sử dụng mô hình phoGPT kết hợp kỹ thuật RAG và cơ sở dữ liệu vector Chroma để cung cấp hệ thống hỏi đáp pháp luật tự động, hỗ trợ đại biểu và cán bộ tra cứu, đối chiếu thông tin pháp luật một cách nhanh chóng và chính xác, đồng thời trích dẫn rõ ràng các điều khoản liên quan.



Hình 4. Giao diện ứng dụng tra cứu, hỏi đáp pháp luật của dự án VN-Law-Advisor (Nguồn: https://github.com/CTU-LinguTechies/VN-Law-Advisor)

Dự án VN-Law-Advisor đã xây dựng thành công một hệ thống tra cứu pháp luật toàn diện dựa trên Bộ pháp điển và các văn bản quy phạm pháp luật từ năm 1990 đến nay, được cập nhật liên tục. Hệ thống sử dụng kiến trúc microservices hiện đại, tích hợp các công cụ xử lý PDF (PyMuPDF), tiền xử lý ngôn ngữ tiếng Việt (VnCoreNLP, Underthesea) và lưu trữ embedding trong cơ sở dữ liệu vector Chroma để truy xuất nhanh và chính xác. Người dùng có thể đặt câu hỏi tự nhiên,, hệ thống sẽ tự động truy xuất thông tin liên quan và sinh câu trả lời mạch lạc, kèm trích dẫn điều khoản cụ thể, giúp đại biểu và cán bộ hỗ trợ trong các buổi tiếp xúc cử tri có thể phản hồi nhanh chóng, chính xác và có căn cứ pháp lý vững chắc.

3. Mục tiêu đề tài

Mục tiêu của đề tài là xây dựng một hệ thống hỏi - đáp tự động có khả năng tiếp nhận đầu vào là các văn bản pháp luật tiếng Việt dưới định dạng tập tin PDF hoặc ảnh scan, phân tích nội dung bằng các kỹ thuật xử lý ngôn ngữ tự nhiên hiện đại, và sinh ra câu trả lời phù hợp với câu hỏi người dùng. Hệ thống hướng đến việc xử lý tốt các đặc thù của văn bản pháp lý như ngôn ngữ chuyên ngành, cấu trúc phi chuẩn, độ dài lớn và yêu cầu chính xác cao.

Đồng thời, đề tài cũng tập trung vào việc tích hợp các mô hình ngôn ngữ tiên tiến nhằm nâng cao khả năng truy xuất thông tin theo ngữ cảnh, phục vụ hiệu quả cho các tình huống thực tiễn như tra cứu quy định pháp luật, tham vấn chính sách hoặc hỗ trợ công tác tiếp xúc cử tri. Mục tiêu cuối cùng là góp phần tạo ra một giải pháp có tính ứng dụng cao, giúp rút ngắn thời gian tìm kiếm và cải thiện khả năng tiếp cận thông tin pháp luật cho người dân và cán bộ chuyên môn.

4. Đối tượng và phạm vi nghiên cứu

4.1. Đối tượng nghiên cứu:

Đối tượng nghiên cứu của đề tài là các phương pháp và mô hình trong lĩnh vực xử lý ngôn ngữ tự nhiên (NLP), đặc biệt là tập trung vào bài toán hỏi - đáp tự động (Question Answering) trên ngữ liệu tiếng Việt. Ngoài ra, đề tài còn nghiên cứu các kỹ thuật xử lý tài liệu đầu vào như ảnh chụp, tập tin PDF văn bản pháp luật, bao gồm nhận dạng văn bản (OCR), trích xuất thông tin và lưu trữ có cấu trúc để phục vụ cho việc truy xuất và sinh câu trả lời.

4.2. Phạm vi nghiên cứu:

Đề tài tập trung vào việc xây dựng một hệ thống có khả năng:

- Nhận diện và trích xuất nội dung từ tài liệu văn bản pháp luật tiếng Việt dưới định dạng ảnh chụp hoặc tập tin PDF.
- Lưu trữ và tổ chức thông tin pháp lý đã trích xuất để hỗ trợ tìm kiếm và truy vấn hiệu quả.
- Phát triển một chatbot có khả năng tự động sinh câu trả lời cho các câu hỏi pháp luật dựa trên dữ liệu đã lưu trữ.

Hệ thống được triển khai thử nghiệm trên tập hợp các văn bản quy phạm pháp luật phổ biến (như nghị định, nghị quyết, thông tư, luật, …) với phạm vi ngôn ngữ tiếng Việt. Đề tài không đi sâu vào các vấn đề như diễn giải pháp lý hoặc tư vấn luật chuyên sâu.

5. Phương pháp nghiên cứu

5.1. Về lý thuyết:

Đề tài tổng hợp và ứng dụng các kiến thức liên quan đến xử lý ngôn ngữ tự nhiên (NLP), học sâu (Deep Learning), thị giác máy tính (Compuer Vision), và xây dựng hệ thống web tương tác. Các mô hình và thư viện hiện đại như Transformer, ViT5, RAG, PyMuPDF, VietOCR, cùng với các phương pháp thiết kế phần mềm … và cơ sở dữ liệu MongoDB sẽ được tích hợp để tạo thành một hệ thống hỏi - đáp hoàn chỉnh. Cụ thể, về mặt lý thuyết bao gồm các hướng nghiên cứu sau:

- Nghiên cứu các phương pháp tiền xử lý văn bản từ tập tin PDF hoặc ảnh chụp tài liệu pháp luật, bao gồm trích xuất văn bản bằng OCR (VietOCR), xử lý lỗi nhận diện, và chuẩn hóa ngữ liệu.
- Tìm hiểu các kỹ thuật phân tích và hiểu ngôn ngữ tự nhiên thông qua kiến trúc ViT5 để phục vụ trích chọn thông tin và sinh câu trả lời.

- Nghiên cứu kiến trúc RAG để kết hợp giữa khả năng truy vấn văn bản (retrieval) và sinh câu trả lời có ngữ cảnh (generation), đặc biệt phù hợp với môi trường dữ liệu văn bản pháp luật phân mảnh.
- Tìm hiểu cách xây dựng hệ thống QA chuyên biệt cho tiếng Việt, trong đó có việc đánh giá hiệu quả mô hình ViT5 trong việc hiểu ngữ nghĩa câu hỏi và văn bản pháp luật.
- Phân tích cách tổ chức dữ liệu pháp luật theo cấu trúc truy vấn hiệu quả (ví dụ như tạo chỉ mục nội dung văn bản theo điều, khoản, mục luật) để hỗ trợ quá trình tìm kiếm và đối sánh thông tin.
- Xây dựng Backend và giao diện web kết hợp Node.js để xử lý yêu cầu từ người dùng và MongoDB để lưu trữ dữ liệu văn bản đã xử lý và các câu trả lời sinh ra.
- Nghiên cứu các giao thức truyền dữ liệu thời gian thực như SSE hoặc WebSocket để hiển thị tiến trình xử lý tài liệu và phản hồi trả lời theo thời gian thực cho người dùng.

5.2. Về thực hành:

- Xây dựng pipeline xử lý đầu vào, từ việc cho phép người dùng tải lên tập tin PDF hoặc ảnh chụp, xử lý ảnh bằng OCR, đến trích xuất và chuẩn hóa nội dung để đưa vào hệ thống lưu trữ.
- Cài đặt và fine-tuning các mô hình ngôn ngữ pre-trained như ViT5 để tối ưu khả năng sinh câu trả lời phù hợp với ngữ cảnh văn bản pháp luật tiếng Việt.
- Kết hợp mô hình truy xuất văn bản (BM25, FAISS, hay Elasticsearch) với mô hình sinh văn bản để hiện thực hóa kiến trúc RAG trong môi trường tiếng Việt.
- Triển khai hệ thống web cho phép người dùng: (1) tải tài liệu, (2) đặt câu hỏi, và (3) nhận câu trả lời tự động từ hệ thống dựa trên văn bản đã lưu trữ.
- Đánh giá hệ thống qua các tiêu chí như độ chính xác câu trả lời, thời gian phản hồi, và mức độ hài lòng của người dùng trải nghiệm.

6. Kết quả đạt được:

7. Bố cục quyển báo cáo

Phần 1. Phần giới thiệu

Phần 2. Phần nội dung

Chương 1: Mô tả bài toán.

Chương 2: Thiết kế và cài đặt giải pháp

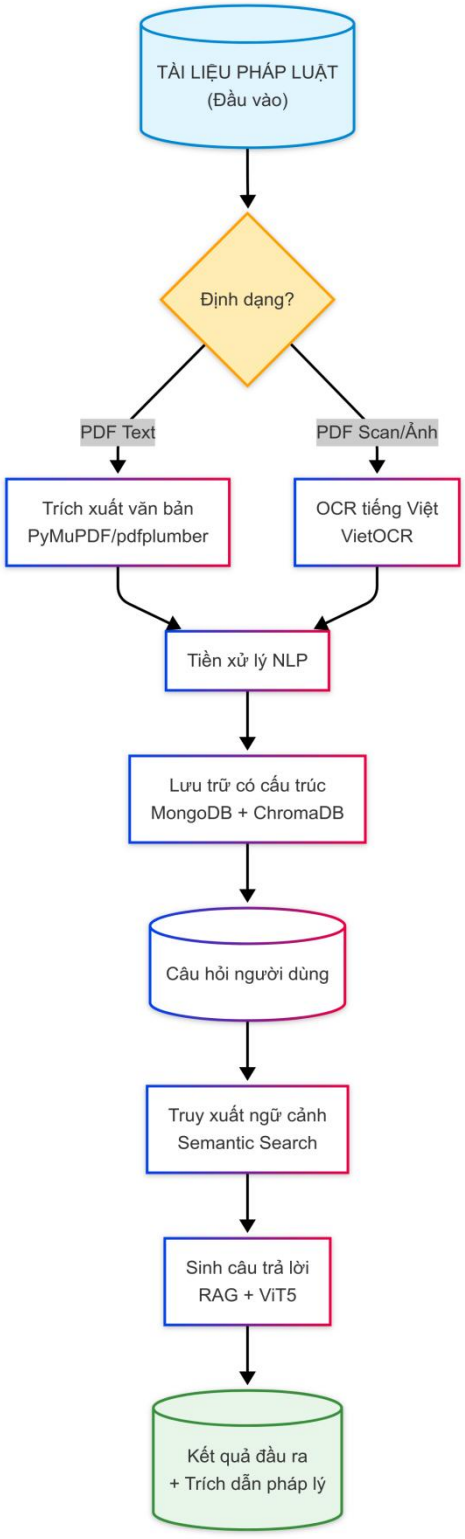
Chương 3: Kiểm thử và đánh giá

Phần 3. Phần kết luận và hướng phát triển.

II. PHẦN NỘI DUNG

CHƯƠNG 1. MÔ TẢ BÀI TOÁN

1.1. Mô tả chi tiết bài toán



Hình 5. Quy trình tổng thể hệ thống hỏi - đáp pháp luật

Bài toán được đặt ra là xây dựng một hệ thống tự động xử lý và hỏi đáp dựa trên các văn bản pháp luật do người dùng cung cấp dưới dạng tập tin PDF **hoặc ảnh scan**. Hệ thống cần thực hiện các nhiệm vụ chính gồm: trích xuất nội dung văn bản, lưu trữ dữ liệu có tổ chức và trả lời chính xác các câu hỏi liên quan đến nội dung các văn bản đó. Các câu hỏi có thể xoay quanh các quy định pháp lý, điều khoản, quyền lợi hoặc nghĩa vụ của công dân trong các tình huống cụ thể.

Quy trình xử lý tổng quan của hệ thống bao gồm:

1. Người dùng tải lên văn bản pháp luật (có thể là PDF dạng ảnh scan hoặc văn bản số).
2. Hệ thống sử dụng các công cụ như PyMuPDF, pdfplumber để trích xuất nội dung nếu là PDF text, hoặc sử dụng OCR (VietOCR) để xử lý nếu là ảnh. Sau đó, nội dung sẽ được làm sạch và phân tích ngữ nghĩa (dùng VnCoreNLP).
3. Nội dung đã xử lý được lưu trữ trong MongoDB theo cấu trúc phù hợp (chia theo chương, điều khoản, …), phục vụ cho việc truy vấn sau này.
4. Người dùng nhập câu hỏi dưới dạng ngôn ngữ tự nhiên liên quan đến nội dung văn bản.
5. Mô hình ViT5 kết hợp RAG sẽ hiểu câu hỏi, truy xuất các đoạn văn bản liên quan và sinh trả lời tự nhiên, chính xác.
6. Câu trả lời được trả về cho người dùng thông qua giao diện web, kèm theo trích dẫn đoạn văn bản gốc liên quan để minh chứng.

Dữ liệu đầu vào có thể là các văn bản pháp luật đã được số hóa (PDF text) hoặc các ảnh chụp, ảnh scan tài liệu giấy. Đối với tài liệu ảnh, yêu cầu cần có độ phân giải tối thiểu 300 DPI, chữ rõ nét, không bị nhòe, nghiêng quá 10 độ và tương phản cao giữa nền và chữ, nhằm đảm bảo độ chính xác cao trong quá trình nhận dạng ký tự quang học (OCR). Các tập tin PDF dạng ảnh cũng cần đáp ứng các yêu cầu tương tự để hệ thống có xử lý chính xác.

Sau khi người dùng tải lên tài liệu, hệ thống sẽ tự động thực hiện các bước xử lý chính xác như sau: Nếu là PDF văn bản số, nội dung sẽ được trích xuất trực tiếp bằng công cụ PyMuPDF; nếu là ảnh hoặc PDF dạng scan, hệ thống sẽ sử dụng VietOCR để thực hiện OCR, chuyển đổi hình ảnh thành văn bản. Toàn bộ văn bản sau đó sẽ được làm sạch, loại bỏ các ký tự dư thừa, phân đoạn theo logic pháp luật (Chương, Điều, Khoản), đồng thời thực hiện tách câu và gán nhãn các thành phần pháp lý (như tên điều luật, nội dung, các khoản) bằng các công cụ xử lý tiếng Việt như VnCoreNLP. Ngoài ra, các thông tin như tên văn bản, ngày ban hành, cơ quan ban hành, số hiệu, v.v... cũng sẽ được trích xuất để lưu trữ metadata phục vụ truy vấn.

Hình 6. Hai loại đầu vào phổ biến: tập tin PDF có thể trích xuất trực tiếp và ảnh scan cần xử lý bằng OCR

Phần văn bản sau khi xử lý sẽ được lưu trữ dưới dạng có cấu trúc trong MongoDB, với tổ chức phân tầng logic theo chương, điều, khoản. Ví dụ, một điều luật sẽ được lưu kèm số điều, tiêu đề điều, danh sách các khoản bên trong cùng nội dung chi tiết. Song song đó, hệ thống sẽ mã hóa toàn bộ các đoạn văn bản thành vector embedding bằng mô hình như Vietnamese SBERT hoặc PhoSimCSE, lưu trữ trong cơ sở dữ liệu vector như Chroma hoặc FAISS để phục vụ việc tìm kiếm ngữ nghĩa cho các câu hỏi sau này.

Hình 7. Dữ liệu được tổ chức theo cấu trúc pháp lý và ánh xạ sang vector để phục vụ truy hồi ngữ nghĩa

Khi người dùng đặt câu hỏi bằng tiếng Việt tự nhiên, hệ thống sẽ thực hiện bước hiểu và xử lý câu hỏi. Câu hỏi sẽ được đưa vào pipeline hỏi - đáp gồm hai giai đoạn chính: truy hồi và sinh câu trả lời. Đầu tiên, hệ thống sử dụng vector hóa câu hỏi để tìm kiếm các đoạn văn bản pháp luật liên quan nhất trong cơ sở dữ liệu vector. Sau khi truy hồi được các đoạn phù hợp, hệ thống sẽ kết hợp các đoạn này với câu hỏi, đưa vào mô hình sinh (generation) như ViT5, vận hành theo cơ chế RAG (Retrieval-Augmented Generation) để tạo ra câu trả lời tự nhiên, dễ hiểu, đúng trọng tâm và đi kèm với trích dẫn nguồn gốc cụ thể (ví dụ: "Theo Điều 13 Luật Đất đai năm 2013...").

Văn bản pháp luật có một số đặc điểm đặc thù cần được chú ý trong toàn bộ pipeline: nó thường có cấu trúc phân tầng rõ ràng nhưng phức tạp (gồm phần, chương, điều, khoản, điểm), sử dụng ngôn ngữ chính quy và pháp lý chặt chẽ, và có tính phi thời gian nhưng vẫn cần đảm bảo thông tin đúng với phiên bản mới nhất. Do đó, hệ thống cần phải lưu trữ dữ liệu một cách có tổ chức, phân tầng chính xác và dễ truy vấn. Bên cạnh đó, phần mô hình sinh cũng cần xử lý cẩn trọng để đảm bảo câu trả lời không chỉ chính xác về mặt pháp lý mà còn dễ hiểu với người dùng không chuyên.

Về mặt tổng thể, đây là một bài toán tổng hợp, bao gồm cả trích xuất thông tin (Information Extraction) và hỏi - đáp văn bản (Document-based QA). Mỗi khâu xử lý đều có vai trò quan trọng: từ OCR và xử lý ngôn ngữ cho ảnh và PDF, đến phân tích ngữ nghĩa và tổ chức lưu trữ dữ liệu, cho tới việc truy vấn và sinh phản hồi từ mô hình ngôn ngữ. Thách thức lớn nhất của hệ thống là làm sao hiểu được nội dung pháp luật có cấu trúc phức tạp, từ đó sinh ra được phản hồi tự nhiên, chính xác và có trích dẫn rõ ràng để hỗ trợ người dùng trong việc tra cứu và áp dụng pháp luật vào các tình huống thực tế.

1.2. Vấn đề và giải pháp liên quan đến bài toán

1.2.1. Nhận dạng ký tự quang học (Optical Character Recognition - OCR):

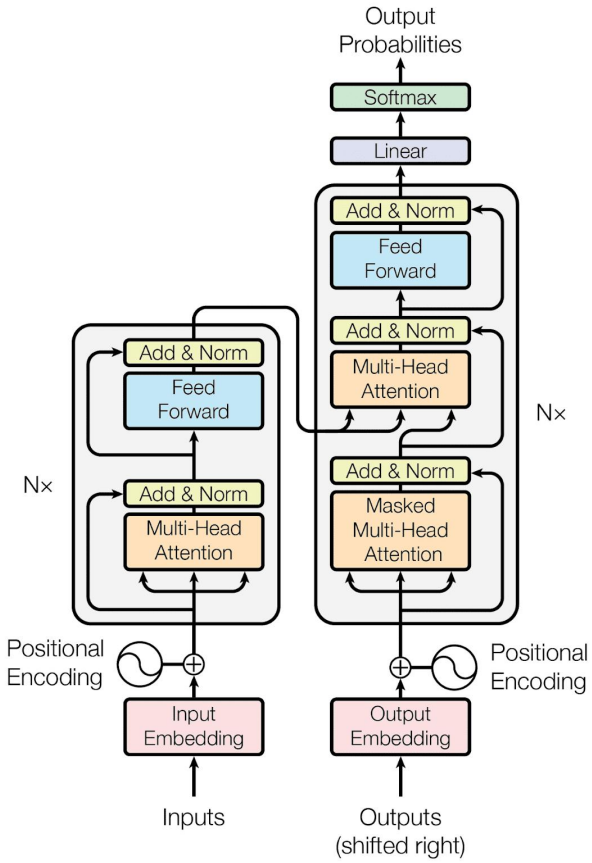
Nhận dạng kí tự quang học (Optical Character Recognition - OCR) là một công nghệ quan trọng cho phép chuyển đổi các loại tài liệu chứa văn bản dưới dạng hình ảnh - chẳng hạn như ảnh scan, ảnh chụp tài liệu giấy, hoặc các tập tin PDF chỉ chứa ảnh - thành dữ liệu văn bản mà máy tính có thể đọc, tìm kiếm và chỉnh sửa được. Nguyên lý hoạt động cơ bản của OCR bao gồm việc phân tích hình ảnh để xác định các ký tự riêng lẻ và sau đó "dịch" các ký tự này thành mã máy tính tương ứng (ví dụ: ASCII hoặc Unicode). Quá trình này

đóng vai trò cầu nối thiết yếu giữa thế giới tài liệu vật lý hoặc tài liệu số dạng ảnh với môi trường dữ liệu số có thể xử lý được.

Quy trình xử lý OCR điển hình thường trải qua nhiều giai đoạn. Đầu tiên là **tiền xử lý ảnh** (Image Processing), bao gồm các thao tác như khử nhiễu (noise reduction), nhị phân hóa (binarization - chuyển ảnh xám hoặc màu thành ảnh đen trắng), làm thẳng hàng (deskewing - chỉnh sửa độ nghiêng của văn bản), và phân đoạn bố cục (layout analysis) để xác định các vùng chứa văn bản, hình ảnh, bảng biểu. Tiếp theo là **phân đoạn kí tự** (Character Segmentation), nơi hệ thống cố gắng tách từng kí tự riêng lẻ ra khỏi dòng văn bản. Sau đó, mỗi ký tự được đưa vào giai đoạn **nhận dạng ký tự** (Character Recognition). Giai đoạn này thường sử dụng các mô hình học máy, đặc biệt là các mạng nơ-ron sâu như mạng nơ-ron tích chập (Convolutional Neural Networks - CNNs) hoặc các kiến trúc kết hợp CNN với mạng nơ-ron tuần tự (Recurrent Neural Networks - RNNs) như Long Short-Term Memory (LSTM), để phân loại hình ảnh của ký tự cụ thể trong bảng chữ cái. Cuối cùng, **hậu xử lý** (Post-processing) được áp dụng để sửa lỗi nhận dạng dựa trên ngữ cảnh ngôn ngữ (ví dụ: sử dụng từ điển, mô hình ngôn ngữ) và định dạng lại văn bản đầu ra.

Tuy nhiên, OCR đối mặt với nhiều thách thức, đặc biệt khi xử lý tài liệu tiếng Việt và tài liệu scan. Tiếng Việt có bảng chữ cái phức tạp với nhiều dấu phụ (thanh điệu và dấu nguyên âm), làm tăng số lượng lớp ký tự cần nhận dạng và dễ gây nhầm lẫn. Chất lượng tài liệu scan thường không đồng đều, có thể bị mờ, nhiễu, độ tương phản thấp, hoặc văn bản bị nghiêng, dính liền, gây khó khăn cho tất cả các giai đoạn của OCR. Các phông chữ đa dạng, kích thước chữ khác nhau và bố cục phức tạp của văn bản pháp luật cũng là những yếu tố làm tăng độ khó của bài toán.

1.2.3. Kiến trúc Transformer:



Hình 8. Kiến trúc Transformer (Nguồn: Vaswani et al., 2017)

Kiến trúc Transformer được giới thiệu lần đầu tiên trong bài báo nổi tiếng "Attention is All You Need" của Vaswani và cộng sự năm 2017. Nó đã tạo nên một cuộc cách mạng trong lĩnh vực xử lý ngôn ngữ tự nhiên (NLP) và nhanh chóng trở thành nền tảng cho hầu hết các mô hình ngôn ngữ lớn (LLMs) hiện đại như BERT, GPT, T5 hay ViT5. Điểm đặc biệt là nó hoàn toàn dựa trên một cơ chế gọi là "Attention" (Chú ý), loại bỏ hoàn toàn sự cần thiết của các thành phần tuần tự (recurrent layers) như trong RNN hay LSTM, vốn là kiến trúc chủ đạo trước đó cho các bài toán xử lý chuỗi. Điều này cho phép Transformer xử lý song song các phần của chuỗi đầu vào, giúp tăng tốc độ huấn luyện và giải quyết hiệu quả vấn đề phụ thuộc xa (long-range dependencies).

Hình 9. Kiến trúc tổng quan

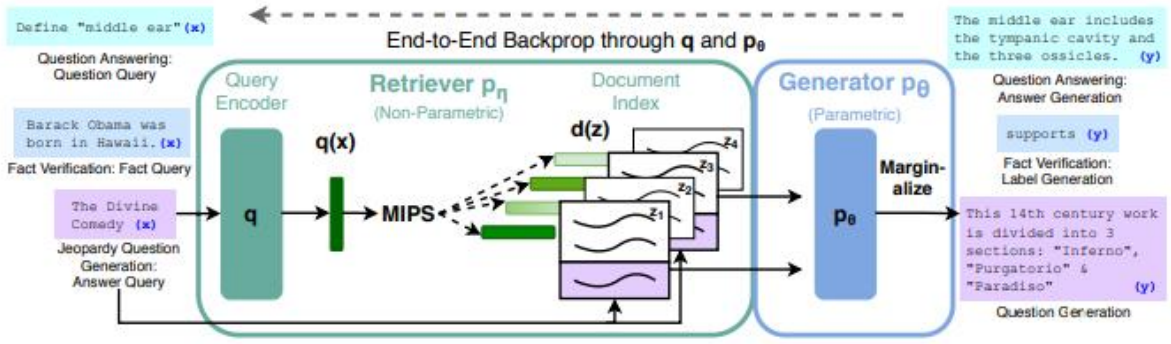
Transformer ban đầu được thiết kế cho bài toán dịch máy (sequence-to-sequence task) và bao gồm hai thành phần chính: Khối mã hóa (Encoder) và khối giải mã (Decoder). Cả Encoder và Decoder đều được cấu tạo từ nhiều tầng (layers) giống hệt nhau được xếp chồng lên nhau. Trong bài báo gốc, số lượng lớp cho cả Encoder và Decoder là sáu (N=6). Khối mã hóa tiếp nhận một chuỗi đầu vào và tạo ra một chuỗi các biểu diễn liên tục, giàu ngữ cảnh (contextualized representations) cho chuỗi đó. Những biểu diễn này nắm bắt ý nghĩa của từng từ trong mối quan hệ với các từ khác trong câu. Khối giải mã tiếp nhận đầu ra của khối mã hóa (các biểu diễn ngữ cảnh) cùng với chuỗi đầu ra đã được tạo ra ở bước trước đó để tạo ra từ tiếp theo trong chuỗi đầu ra một cách tự hồi quy.

1.2.4. Mô hình Vietnamese Text-To-Text Transfer Transformer (ViT5):

1.2.5. Kỹ thuật nhúng văn bản (Text Embedding):

1.2.6. Hệ thống hỏi đáp (Question Answering - QA):

1.2.6. Retrieval-augmented Generation (RAG):



Hình . Kiến trúc kỹ thuật RAG (Nguồn: Lewis et al., 2020)

Retrieval-Augmented Generation (RAG) là một khung trí tuệ nhân tạo hoặc một kỹ thuật học máy được giới thiệu lần đầu tiên trong bài báo "*Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*" (Lewis et al., 2020) nhằm nâng cao khả năng của các mô hình ngôn ngữ lớn (LLM) bằng cách tích hợp chúng với khả năng truy xuất thông tin từ bên ngoài. Điều này có nghĩa là LLM không còn chỉ dựa vào dữ liệu huấn luyện tĩnh của chúng nữa. Nguyên tắc cốt lõi của RAG là "neo" các LLM vào thông tin chính xác, cập nhật nhất từ một cơ sở tri thức bên ngoài trước khi tạo ra phản hồi. Cách tiếp cận này giúp giải quyết "khoảng trống kiến thức" trong cách hoạt động của các LLM truyền thống.

Động lực chính đằng sau sự phát triển của RAG bao gồm việc cho phép tạo ra văn bản chính xác, cập nhật và có thể kiểm chứng được. Quan trọng hơn, RAG giải quyết một số hạn chế chính của LLM, chẳng hạn như việc tạo ra "ảo giác" (halluciation) - tức là thông tin nghe có vẻ hợp lý nhưng thực tế lại không chính xác hoặc gây hiểu lầm. Bằng cách neo quá trình tạo sinh vào các tài liệu được truy xuất và đã được xác minh, RAG giảm thiểu việc tạo ra nội dung sai lệch. Ngoài ra, RAG còn cung cấp cho người dùng cái nhìn sâu sắc về quy trình tạo sinh của LLM bằng cách liên kết các phản hồi với thông tin nguồn.

Hơn nữa, động lực đằng sau RAG không chỉ là dừng lại ở việc đạt được độ chính xác thực tế; nó còn hướng đến việc xây dựng niềm tin và khả năng kiểm chứng trong nội dung do AI tạo ra, điều này rất quan trọng đối với việc áp dụng trong doanh nghiệp và các ứng dụng quan trọng. Hiện tượng ảo giác của LLM là một rào cản lớn đối với niềm tin. RAG nhằm mục đích giảm thiểu những điều này bằng cách đưa các phản hồi vào các tài liệu có thể kiểm chứng. Khả năng truy xuất nguồn gốc của nội dung được tạo ra từ một nguồn sự thật là một kết quả then chốt. Khả năng truy vết này giúp củng cố niềm tin của người dùng vào giải pháp AI.

Từ góc độ vĩ mô, các hệ thống RAG có thể được chia thành hai thành phần chính:

- Phần truy xuất (Retriever): Bao gồm các hoạt động đa dạng như tiền xử lý dữ liệu bên ngoài, các cơ chế truy xuất dày đặc (dense) hoặc thưa thớt (sparse) để tìm kiếm tài liệu liên quan, và có thể bao gồm cả việc xếp hạng lại (reranking) và tỉa bớt (pruning) các tài liệu này. Thành phần truy xuất sẽ tìm kiếm trong các cơ sở dữ liệu / nguồn bên ngoài dựa trên truy vấn người dùng.
- Phần tạo sinh: Bao gồm các thành phần như lập kế hoạch truy xuất (quyết định cái gì / làm thế nào để truy xuất), tích hợp kiến thức từ nhiều nguồn, và suy luận logic bởi LLM để tổng hợp một phản hồi dựa trên câu lệnh đã được tăng cường. Mô-đun tạo sinh xử lý thông tin được truy xuất để tạo ra văn bản giống như con người.

Ngoài ra, các yếu tố liên kết khác bao gồm việc phân đoạn tài liệu (document chunking), tạo vector nhúng (embedding generation), và các cơ chế đảm bảo an ninh và độ tin cậy.

Bảng 1. Các thành phần cốt lõi của hệ thống RAG

Thành phần	Mô tả / Chức năng
Nguồn dữ liệu	Kho lưu trữ kiến trúc bên ngoài mà RAG sẽ truy xuất thông tin.
Bộ chỉ mục / Bộ nhúng (Indexer / Embedder)	Xử lý và chuyển đổi dữ liệu nguồn thành các biểu diễn số (vector nhúng) và lưu trữ chúng trong một cơ sở dữ liệu vector để tìm kiếm hiệu quả.
Bộ truy xuất (Retriever)	Tìm kiếm và lấy các đoạn thông tin liên quan nhất từ cơ sở dữ liệu vector dựa trên truy vấn của người dùng.

Bộ tăng cường (Augmenter)	Kết hợp thông tin được truy xuất với truy vấn ban đầu của người dùng để tạo ra một câu lệnh tăng cường cho LLM.
Bộ tạo sinh / LLM (Generator / LLM)	Nhận câu lệnh tăng cường và tạo ra một phản hồi mạch lạc, dựa trên ngữ cảnh, kết hợp cả kiến thức được truy xuất và kiến thức nội tại của nó.

RAG mang lại nhiều ưu điểm đáng kể, nổi bật nhất là khả năng tăng cường độ chính xác và giảm thiểu tình trạng "ảo giác" của mô hình ngôn ngữ lớn bằng cách truy xuất thông tin thực tế từ các nguồn dữ liệu bên ngoài. Điều này cũng cho phép RAG linh hoạt cập nhật kiến thức mới mà không cần huấn luyện lại toàn bộ mô hình, đồng thời tăng tính minh bạch khi câu trả lời thường đi kèm nguồn trích dẫn để người dùng kiểm chứng, và cải thiện khả năng xử lý kiến thức chuyên ngành hiệu quả. Tuy nhiên, RAG cũng tồn tại một số hạn chế. Hiệu quả của nó phụ thuộc lớn vào chất lượng của nguồn dữ liệu được truy xuất; nếu dữ liệu không tốt, kết quả cũng sẽ bị ảnh hưởng. Bên cạnh đó, việc triển khai RAG làm tăng độ phức tạp của hệ thống và có thể gây ra độ trễ trong thời gian phản hồi do quá trình truy xuất. Cuối cùng, luôn có nguy cơ cơ chế truy xuất không chọn được thông tin tối ưu hoặc bị ảnh hưởng bởi các thiên kiến tiềm ẩn trong dữ liệu. Mặc dù vậy, RAG vẫn là một giải pháp mạnh mẽ để nâng cao chất lượng và độ tin cậy của LLM, nhưng cần cân nhắc kỹ về nguồn dữ liệu và độ phức tạp khi triển khai.

1.2.7. Cơ sở dữ liệu vector (Vector database/stores):

1.2.8. Cơ sở dữ liệu NoSQL:

1.2.9. Kỹ thuật tinh chỉnh mô hình (Fine-tuning techniques):

- Khái niệm fine-tuning.
- LoRA.

1.2.10. Phương pháp đánh giá hệ thống:

1.2.10.1. Đánh giá module OCR:

1.2.10.2. Đánh giá module truy xuất thông tin (Retrieval Performance của RAG)

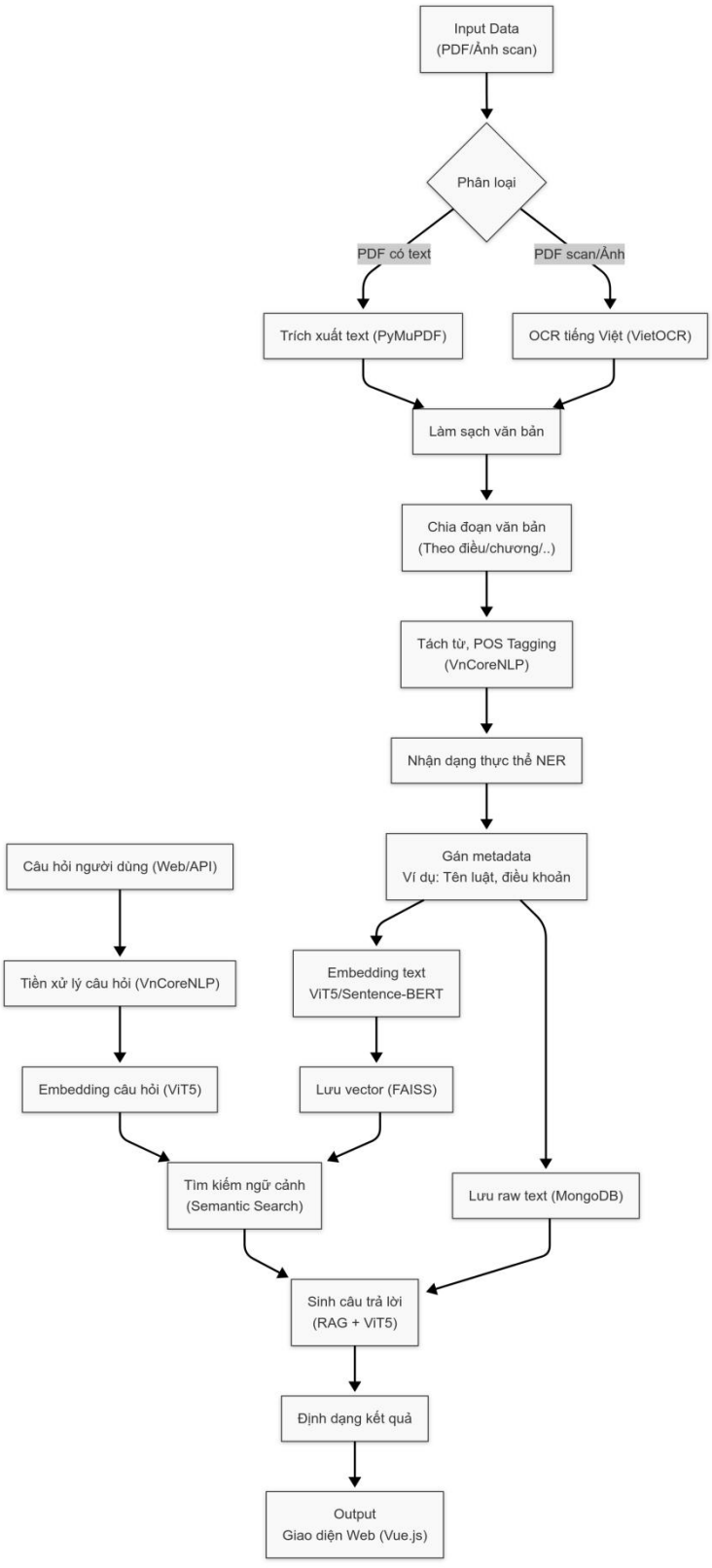
1.2.10.3. Đánh giá module sinh câu trả lời / hỏi - đáp (QA Performance):

1.2.11. Các thư viện và công cụ hỗ trợ:

// Các thư viện được sử dụng khi code, ghi thêm pycharm, visual studio code

CHƯƠNG 2. THIẾT KẾ VÀ CÀI ĐẶT GIẢI PHÁP

2.1. Thiết kế hệ thống



Hình . Quy trình tổng thể của hệ thống

2.2. Cài đặt giải pháp

2.2.1. Thu thập dữ liệu

2.2.2. Tiền xử lý dữ liệu

Bảng 1. ……..

CHƯƠNG 3. KIỂM THỬ VÀ ĐÁNH GIÁ

3.1. Giao diện sản phẩm (nếu có)

3.2. Kết quả thực nghiệm

3.3. Thảo luận về kết quả đạt được

III. PHẦN KẾT LUẬN

1. Kết quả đạt được

2. Hướng phát triển

1. TÀI LIỆU THAM KHẢO

[1] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N, Kaiser, L., & Polosukhin, I. (2017). ***Attention is All You Need***. arXiv preprint arXiv:1706.03762.

[2] Collin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter Jj.. Liu. (2020). ***Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer***. arXiv preprint arXiv:1910.10683

[3] Long Phan, Hieu Tran, Hieu Nguyen, Trieu H. Trinh, (2022) ***ViT5: Pretrained Text-to-Text Transformer for Vietnamese Language Generation.*** arXiv preprint arXiv:2205.06456

[4] Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, Wen-tau Yih, Tim Rocktashcel, Sebastian Riedel, Douwe Kiela, (2020). ***Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks***. arXiv preprint:2005.11401

[5] CTU-LinguTechies (2023) ***VN-Law-Advisor: Hệ thống hỏi đáp pháp luật dựa trên mô hình ngôn ngữ lớn.***

[6] Trần Nguyễn Nhật Huy (2024). ***Xây dựng ứng dụng trích xuất thông tin từ Căn cước công dân***

[7] VietOCR. (n.d). ***VietOCR: Open source OCR software for Vietnamese language***. https://pbcquoc.github.io/vietocr/

[8]

[9]

[10]

[11]

2. PHỤ LỤC