ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

NGÔ PHÚ THỊNH

Deep learning in legal system: Opportunities and Challenges

LUẬN VĂN TỐT NGHIỆP
CHUYÊN NGÀNH KHOA HỌC DỮ LIỆU

Tp. Hồ Chí Minh - 2023

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

NGÔ PHÚ THỊNH

Deep learning in legal system: Opportunities and Challenges

LUẬN VĂN TỐT NGHIỆP
CHUYÊN NGÀNH KHOA HỌC DỮ LIỆU

NGƯỜI HƯỚNG DẪN KHOA HỌC
PGS. TS. Nguyễn Thanh Bình

Tp. Hồ Chí Minh - 2023

Lời cảm ơn

Khóa luận tốt nghiệp chuyên ngành Khoa học Dữ liệu với đề tài Deep Learning in the Legal System: Opportunities and Challenges là kết quả cố gắng của bản thân tôi sau 4 năm học tập tại Khoa Toán - Tin học, trường Đại học Khoa học Tự nhiên, ĐHQG-TPHCM và được sự giúp đỡ, động viên từ quý thầy cô, bạn bè và người thân. Qua đây tôi xin gửi lời cảm ơn chân thành đến những người đã giúp đỡ tôi trong quá trình học tập - nghiên cứu khoa học vừa qua.

Lời đầu tiên, tôi xin trân trọng gửi đến PGS. TS. Nguyễn Thanh Bình lời cảm ơn chân thành và sâu sắc nhất. Thầy không chỉ là người tạo cảm hứng cho tôi đến với chuyên ngành Khoa học Dữ liệu, mà còn là người nhiệt tình hướng dẫn cũng như cung cấp cho tôi những kiến thức, tài liệu khoa học cần thiết phục vụ cho luận văn này.

Tôi cũng muốn gửi lời cảm ơn chân thành đến anh Đỗ Hữu Chiến, người đã truyền cho tôi rất nhiều cảm hứng và động lực để tôi có thể hoàn thành luận văn này.

Tôi cũng tri ân đến bạn Lê Huy Hoàng, đã hỗ trợ tôi rất nhiều từ giai đoạn lên ý tưởng đến việc đề xuất những phương pháp hữu ích cho luận văn này.

Nhân dịp này tôi xin gửi lời cảm ơn đến quý thầy cô ở khoa Toán - Tin học đã nhiệt tình truyền đạt cho tôi những kiến thức từ cơ bản đến chuyên sâu trong suốt quá trình học tập tại Khoa. Những kiến thức tích lũy được ở Khoa đã giúp tôi có nền tảng vững vàng cho việc phát triển tương lai sau này.

Cuối cùng, tôi xin cảm ơn gia đình, người thân, bạn bè đã luôn bên cạnh, ủng hộ, động viên.

Tp.HCM, ngày 12 tháng 6 năm 2023

Tác giả

Ngô Phú Thịnh

Mục lục

Danh mục bảng biểu

Danh mục hình ảnh

Lời nói đầu

Trong những năm gần đây, thuật ngữ "trí tuệ nhân tạo" (artificial intelligence) đã trở nên phổ biến hơn bao giờ hết, nhờ những tiến bộ vượt bậc trong quá trình nghiên cứu và phát triển của ngành khoa học máy tính.

Gần đây nhất những tác vụ như text-to-image (sinh ảnh từ chữ) hay text generation (sinh chữ), đã cho thấy sự sáng tạo đột phá của AI. Đứng đầu danh sách của các lĩnh vực này là các công cụ như MidJourney, Stable Diffusion (text-to-image); ChatGPT, Bard, BingAI, LLaMA (text generation) đã thu hút được sự quan tâm lớn của nhiều người. ChatGPT ước tính đã có hơn 100 triệu người dùng chỉ sau 2 tháng ra mắt.

Tuy những công cụ AI này rất hữu dụng, nhưng chúng chỉ có thể xử lí được các kiến thức chung, còn bị giới hạn nhiều ở kiến thức chuyên ngành. Đối với tiếng Việt, những công cụ này hoạt động không hoàn toàn hiệu quả.

Hầu hết mọi người đều cho rằng pháp luật quá khô khan, cứng nhắc và cao siêu. Họ sợ hãi phải dính dáng đến bất cứ vấn đề gì liên quan đến pháp luật. Cứ nhắc đến pháp luật là người ta nghĩ ngay đến những điều kinh khủng như thủ tục rắc rối, bị gây khó dễ, bất công, bồi thường, ngồi tù… Nỗi sợ này có nguồn gốc từ lâu đời, từ khi pháp luật còn là công cụ để giới cầm quyền đàn áp, bóc lột nô lệ và những người dân yếu đuối[1, p. 74].

Nhưng thực tế, luật là một ngành rất rộng, có ảnh hưởng đến mọi lĩnh vực của đời sống xã hội. Pháp luật chính là cuộc sống. Luật như một công cụ để bảo vệ lợi ích của công dân. Đó là những hoạt động rất đỗi bình thường mà chúng ta vẫn làm hằng ngày. Chỉ cần chúng ta chịu tìm hiểu, mọi vấn đề đều trở nên vô cùng đơn giản.

Với mong muốn giúp mọi người hiểu rõ hơn về luật, cũng như giúp các luật sư, sinh viên luật có thêm một công cụ hỗ trợ trong công việc, đưa khái niệm "luật là cuộc sống" đi sâu vào tiềm thức nhiều người. Tôi đã tiến hành nghiên cứu các ứng dụng của AI trong lĩnh vực này.

Nội dung của luận văn ngoài phần mở đầu và kết luận được chia thành 3 phần:

- Phần 1: Các kiến thức liên quan.
- Phần 2: Cơ hội và thách thức.
- Phần 3: Thử nghiệm tra cứu văn bản luật.

1. Kiến thức chuẩn bị

1.1. Hệ thống văn bản quy phạm pháp luật

Hệ thống những văn bản quy phạm pháp luật là hình thức biểu hiện mối liên hệ bên ngoài của pháp luật thông qua các loại văn bản quy phạm pháp luật có giá trị cao thấp khác nhau được các cơ quan Nhà nước có thẩm quyền ban hành theo một trình tự, thủ tục do pháp luật quy định, nhưng đều tồn tại trong thể thống nhất.

Các văn bản quy phạm pháp luật tạo nên hệ thống pháp luật các văn bản quy phạm pháp luật có những đặc điểm:

- Nội dung của các văn bản quy phạm pháp luật là các quy phạm pháp luật do các cơ quan Nhà nước có thẩm quyền ban hành.
- Các văn bản quy phạm pháp luật đều có tên gọi khác nhau (luật, nghị định, pháp lệnh…) do Hiến pháp quy định. Giá trị pháp lý của chúng cao thấp khác nhau do vị trí của cơ quan Nhà nước trong bộ máy nhà nước có quy định.
- Các văn bản quy phạm pháp luật có hiệu lực trong không gian (hiệu lực trong phạm vi khu vực lãnh thổ) và hiệu lực theo thời gian (bắt đầu có hiệu lực hay hết hiệu lực), hiệu lực theo nhóm người (có hiệu lực đối với nhóm người này và không có hiệu lực đối với nhóm người khác).

Theo Hiến pháp năm 2013, Luật Ban hành văn bản quy phạm pháp luật năm 2015 quy định hệ thống những văn bản quy phạm pháp luật gồm các văn bản có giá trị pháp lý như sau:

1. Hiến pháp.
2. Bộ luật, luật, nghị quyết của Quốc hội.
3. Pháp lệnh, nghị quyết của Ủy ban thường vụ Quốc hội; nghị quyết liên tịch giữa Ủy ban thường vụ Quốc hội với Đoàn Chủ tịch Ủy ban trung ương Mặt trận Tổ quốc Việt Nam; nghị quyết liên tịch giữa Ủy ban thường vụ Quốc hội, Chính phủ, Đoàn Chủ tịch Ủy ban trung ương Mặt trận Tổ quốc Việt Nam.
4. Lệnh, quyết định của Chủ tịch nước.
5. Nghị định của Chính phủ; nghị quyết liên tịch giữa Chính phủ với Đoàn Chủ tịch Ủy ban trung ương Mặt trận Tổ quốc Việt Nam.
6. Quyết định của Thủ tướng Chính phủ.
7. Nghị quyết của Hội đồng Thẩm phán Tòa án nhân dân tối cao.

8. Thông tư của Chánh án Tòa án nhân dân tối cao; thông tư của Viện trưởng Viện kiểm sát nhân dân tối cao; thông tư của Bộ trưởng, Thủ trưởng cơ quan ngang bộ; quyết định của Tổng Kiểm toán nhà nước.
9. Thông tư liên tịch giữa Chánh án Tòa án nhân dân tối cao, Viện trưởng Viện kiểm sát nhân dân tối cao, Tổng Kiểm toán nhà nước, Bộ trưởng, Thủ trưởng cơ quan ngang bộ. Không ban hành thông tư liên tịch giữa Bộ trưởng, Thủ trưởng cơ quan ngang bộ.
10. Nghị quyết của Hội đồng nhân dân tỉnh, thành phố trực thuộc Trung ương (sau đây gọi chung là cấp tỉnh).
11. Quyết định của Ủy ban nhân dân cấp tỉnh.
12. Văn bản quy phạm pháp luật của chính quyền địa phương ở đơn vị hành chính - kinh tế đặc biệt.
13. Nghị quyết của Hội đồng nhân dân huyện, quận, thị xã, thành phố thuộc tỉnh, thành phố thuộc thành phố trực thuộc Trung ương (sau đây gọi chung là cấp huyện).
14. Quyết định của Ủy ban nhân dân cấp huyện.
15. Nghị quyết của Hội đồng nhân dân xã, phường, thị trấn (sau đây gọi chung là cấp xã).
16. Quyết định của Ủy ban nhân dân cấp xã.

1.2. Large Language Model

Large Language Model (LLM) là mô hình ngôn ngữ sử dụng deep neural network[1] với số lượng tham số rất lớn (thường là hàng tỷ trọng số hoặc nhiều hơn), được huấn luyện trên lượng lớn văn bản không được gán nhãn bằng cách sử dụng học tự giám sát hoặc học bán giám sát (supervised learning và unsupervised learning). LLM xuất hiện vào khoảng năm 2018 và thể hiện khả năng xử lý tốt nhiều loại nhiệm vụ khác nhau. Điều này đã thay đổi tâm điểm của nghiên cứu xử lý ngôn ngữ tự nhiên từ mô hình giám sát chuyên biệt cho từng nhiệm vụ sang mô hình đa năng có thể thích ứng với nhiều tình huống. LLM thường được áp dụng trong các ứng dụng xử lý ngôn ngữ tự nhiên (natural language processing) như hiểu, tóm tắt, dịch, sinh và dự đoán văn bản mới.

Một ví dụ của LLM là GPT, viết tắt của Generative Pre-trained Transformer. GPT là một mô hình biến đổi được tiền huấn luyện trên một tập dữ liệu văn bản rộng lớn, sau đó được tinh chỉnh cho các nhiệm vụ cụ thể như sinh văn bản, trả lời câu hỏi, phân loại văn bản và hơn thế nữa. GPT có khả năng sinh

[1]Deep neural network (DNN) là một mạng nơ-ron nhân tạo (ANN) với nhiều lớp ẩn giữa lớp đầu vào và lớp đầu ra. DNN có thể được huấn luyện với dữ liệu không được gán nhãn và được sử dụng để phân loại, phân cụm và trích xuất đặc trưng.

ra các đoạn văn bản có ý nghĩa và trôi chảy từ một đầu vào bất kỳ, chẳng hạn như một câu, một từ khóa hoặc một hình ảnh. Phiên bản mới nhất của GPT là GPT-4[2], có khoảng 100 tỷ tham số và được huấn luyện trên khoảng 10 triệu từ.

1.3. Generative Pretrained Transformer

Generative Pre-trained Transformer (GPT), một loại mô hình học sâu có khả năng sinh văn bản tự động dựa trên dữ liệu huấn luyện lớn. GPT được phát triển bởi OpenAI[2]. GPT có nhiều phiên bản khác nhau, từ GPT-1 ra mắt vào năm 2018 đến GPT-3 ra mắt vào năm 2020. Mỗi phiên bản đều có số lượng tham số và khả năng sinh văn bản cao hơn phiên bản trước. Điển hình như GPT-3 có 175 tỷ tham số và có thể sinh văn bản với độ dài tối đa là 2048 từ. GPT có thể áp dụng cho nhiều ứng dụng khác nhau, như viết tiêu đề, tóm tắt, bài luận, thơ, hội thoại và nhiều thứ khác. Ví dụ, GPT-3 có thể viết một bài luận ngắn về tác dụng của việc đọc sách hoặc một câu chuyện ngắn về một chú mèo tên Tom. GPT là một trong những mô hình học sâu tiên tiến nhất hiện nay trong lĩnh vực xử lý ngôn ngữ tự nhiên.

1.4. Embeddings

Embedding là một kỹ thuật biểu diễn các nội dung số như hình, chữ, âm thanh thành một danh sách các con số (vector). Quá trình này giúp cho các mô hình machine learning có thể "hiểu" được nội dung đó.

Embeddings thường được sử dụng trong các ứng dụng như:

- Search (kết quả được sắp xếp theo mức độ liên quan đến một chuỗi truy vấn)
- Clustering (các chuỗi văn bản được nhóm lại theo độ tương tự)
- Recommendations (các mục có chuỗi văn bản liên quan được đề xuất)
- Anomaly detection (các điểm ngoại lệ có độ tương tự thấp được xác định)
- Diversity measurement (phân tích phân phối độ tương tự)
- Classification (các chuỗi văn bản được phân loại theo nhãn tương tự nhất)

1.5. TF-IDF

TF-IDF (Term Frequency – Inverse Document Frequency) là 1 kĩ thuật sử dụng trong khai phá dữ liệu văn bản. Trọng số này được sử dụng để đánh giá tầm quan trọng của một từ trong một văn bản. Giá trị cao thể hiện độ quan trọng cao và nó phụ thuộc vào số lần từ xuất hiện trong văn bản nhưng bù lại bởi tần

[2]OpenAI là một tổ chức nghiên cứu trí tuệ nhân tạo phi lợi nhuận được thành lập vào tháng 12 năm 2015, có trụ sở tại San Francisco, California. OpenAI được thành lập bởi Elon Musk, Sam Altman và các nhà nghiên cứu khác, với mục tiêu "điều tra và thúc đẩy một trí tuệ nhân tạo thân thiện với con người

suất của từ đó trong tập dữ liệu. Một vài biến thể của tf-idf thường được sử dụng trong các hệ thống tìm kiếm như một công cụ chính để đánh giá và sắp xếp văn bản dựa vào truy vấn của người dùng. Tf-idf cũng được sử dụng để lọc những từ stopwords trong các bài toán như tóm tắt văn bản và phân loại văn bản.

TF: Term Frequency(Tần suất xuất hiện của từ) là số lần từ xuất hiện trong văn bản. Vì các văn bản có thể có độ dài ngắn khác nhau nên một số từ có thể xuất hiện nhiều lần trong một văn bản dài hơn là một văn bản ngắn. Như vậy, term frequency thường được chia cho độ dài văn bản (tổng số từ trong một văn bản).

$$\mathrm{TF}(t,d) = \frac{f(t,d)}{\max\{f(w,d) : w \in d\}}$$

Trong đó:
- $\mathrm{TF}(t,d)$: tần suất xuất hiện của từ t trong văn bản d
- $f(t,d)$: Số lần xuất hiện của từ t trong văn bản d
- $\max(\{f(w,d) : w \in d\})$: Số lần xuất hiện của từ có số lần xuất hiện nhiều nhất trong văn bản

IDF: Inverse Document Frequency(Nghịch đảo tần suất của văn bản), giúp đánh giá tầm quan trọng của một từ. Khi tính toán TF, tất cả các từ được coi như có độ quan trọng bằng nhau. Nhưng một số từ như “is”,"of" và “that” thường xuất hiện rất nhiều lần nhưng độ quan trọng là không cao. Như thế ta cần giảm độ quan trọng của những từ này xuống.

$$\mathrm{IDF}(t,D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

Trong đó:
- $\mathrm{IDF}(t,D)$: giá trị idf của từ t trong tập văn bản
- $|D|$: Tổng số văn bản trong tập D
- $|\{d \in D : t \in d\}|$: thể hiện số văn bản trong tập D có chứa từ t.

Việc sử dụng logarit nhằm giúp giá trị tf-idf của một từ nhỏ hơn, do công thức tính tf-idf của một từ trong 1 văn bản là tích của tf và idf của từ đó.

Cụ thể, công thức tính tf-idf hoàn chỉnh như sau:

$$\mathrm{TDIDF}(t,d,D) = \mathrm{TF}(t,d) * \mathrm{IDF}(t,D)$$

Khi đó những từ có giá trị TF-IDF cao là những từ xuất hiện nhiều trong văn bản này, và xuất hiện ít trong các văn bản khác. Việc này giúp lọc ra những từ phổ biến và giữ lại những từ có giá trị cao (từ khoá của văn bản đó).

1.6. BM25

Phương pháp có tên BM25 (BM – best match), thường gọi "Okapi BM25", vì lần đầu tiên công thức được sử dụng trong hệ thống tìm kiếm Okapi, được sáng lập tại trường đại học London những năm 1980 và 1990[3].

Công thức tính điểm của BM25 được định nghĩa như sau:

$$\text{BM25}(D,Q) = \sum_{i=1}^{n} \text{IDF}(q_i, D) \frac{f(q_i, D) * (k_1 + 1)}{f(q_i) + k_1 * \left(1 - b + b * \frac{|D|}{d_{\text{avg}}}\right)}$$

Trong đó:

- $f(q_i, D)$: là số lần mà term q_i xuất hiện trong tất cả các tài liệu D
- $|D|$ là số từ trong tất cả các tài liệu D
- d_{avg} là số lượng từ trung bình trong mỗi tài liệu
- b và k_1 là các tham số của BM25

So với thuật toán TF-IDF, BM25 có ưu điểm là có thể xử lý được các văn bản dài. Điều này là do công thức của BM25 có thêm một số tham số như b và k_1 để điều chỉnh. Các tham số này giúp cho BM25 có thể xử lý được các văn bản dài hơn.

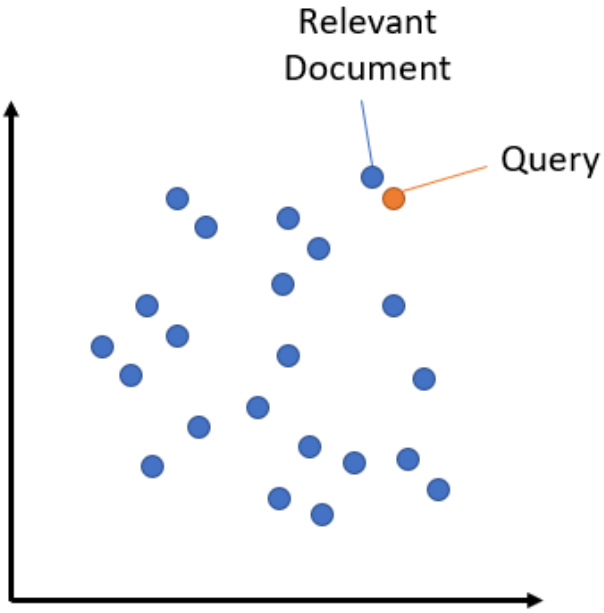
1.7. Sentence Transformers

Sentence Transformers[4] là một python framework cho tác vụ embeddings. Nó được xem là state-of-the-art[3] trong tác vụ embeddings.

Semantic Search là một ứng dụng của Sentence Transformers. Nó cho phép tìm kiếm các văn bản có nội dung tương tự với một văn bản đầu vào. Để thực hiện ứng dụng này, ta cần có một bộ dữ liệu các văn bản và một mô hình embeddings. Mô hình embeddings này sẽ nhúng các văn bản trong bộ dữ liệu thành các vector. Sau đó, ta sẽ tính khoảng cách giữa vector của văn bản đầu vào với các vector của các văn bản trong bộ dữ liệu.

Khoảng cách giữa hai vector đo lường mức độ liên quan của chúng. Khoảng cách nhỏ cho thấy mức độ liên quan cao và khoảng cách lớn cho thấy mức độ liên quan thấp.

[3] "state-of-the-art" những gì hiện đại và tiên tiến nhất



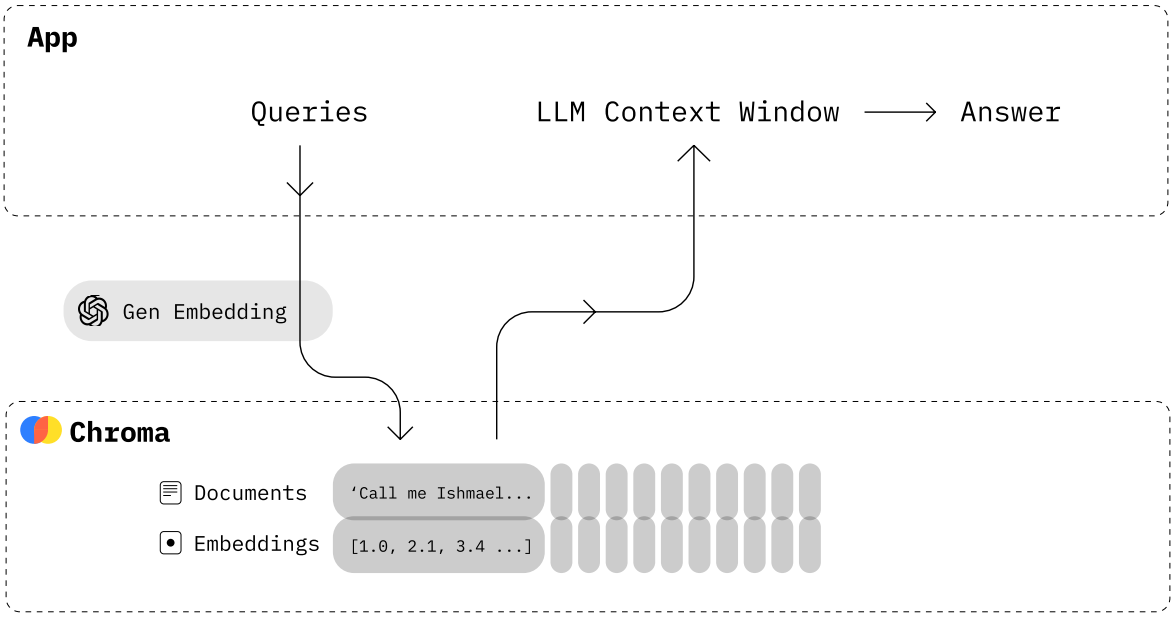
Hình 1: Semantic Search

1.8. Chroma

Chroma là một cơ sở dữ liệu nhúng mã nguồn mở được thiết kế để lưu trữ các vector nhúng (embeddings) và cho phép tìm kiếm các vector gần nhất thay vì tìm kiếm theo chuỗi con như một cơ sở dữ liệu truyền thống.

Chroma cung cấp các công cụ để:

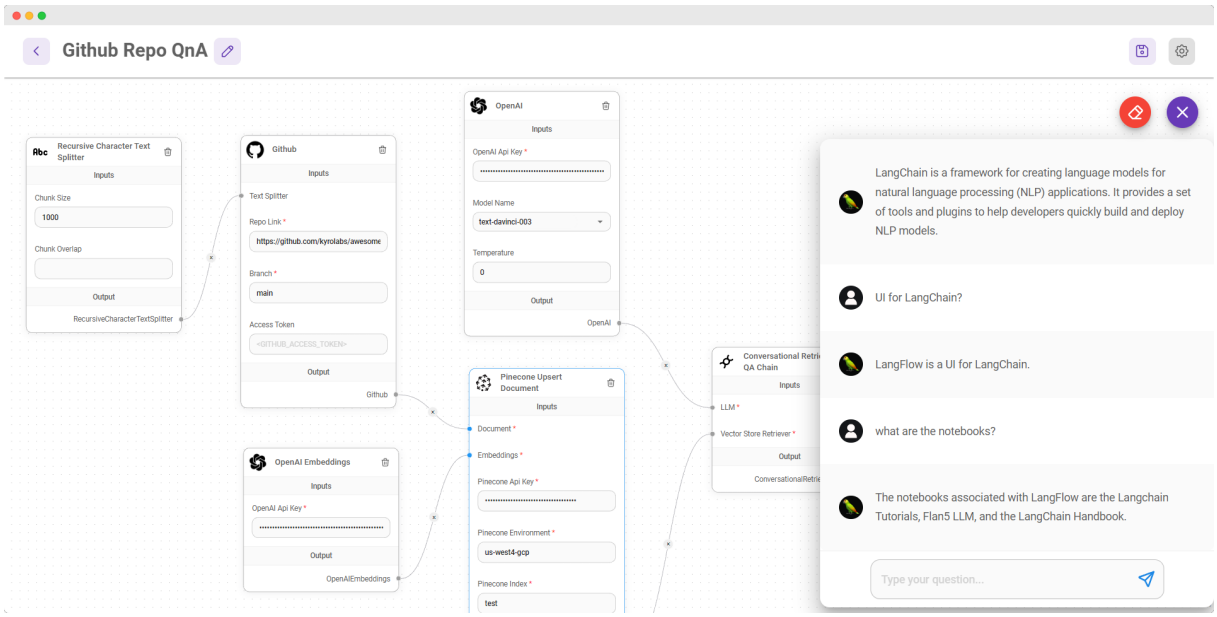
- Lưu trữ embeddings và metadata (dữ liệu mô tả) của chúng
- Nhúng tài liệu và truy vấn
- Tìm kiếm embeddings



Hình 2: Cơ sở dữ liệu Chroma

1.9. Langchain

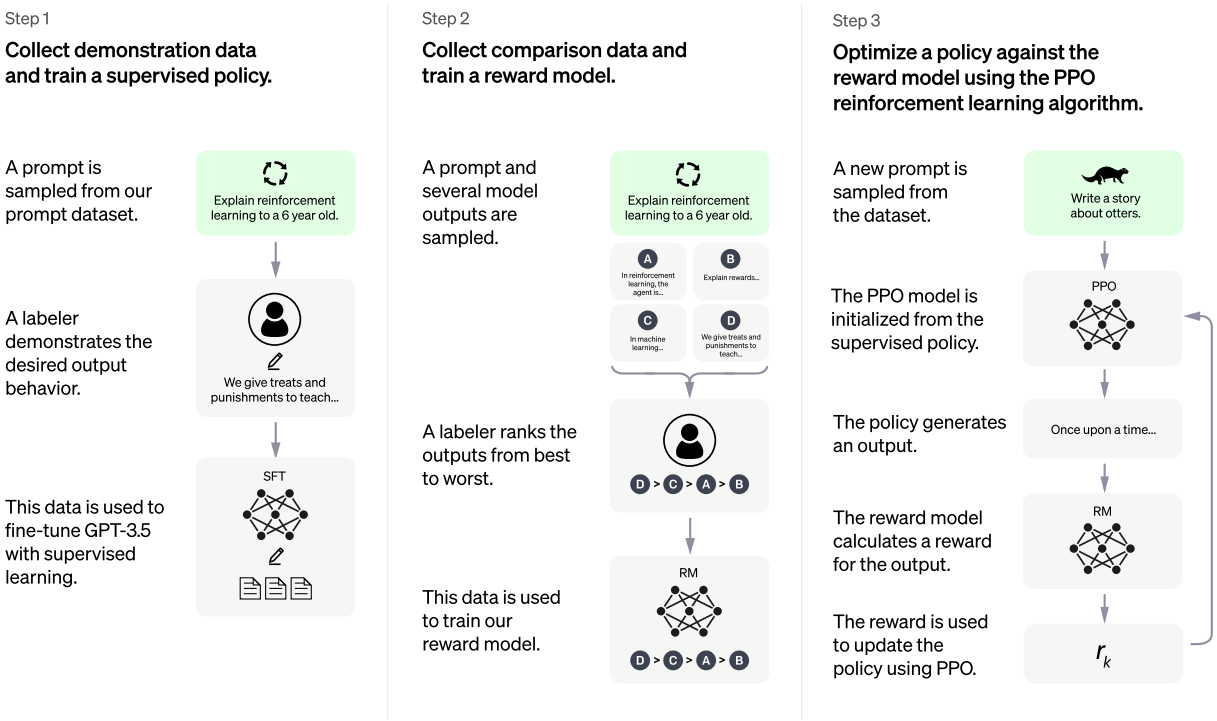
Langchain là một framework được sinh ra để tận dụng sức mạnh của các mô hình ngôn ngữ lớn LLM như ChatPGT, LLaMA… để tạo ra các ứng dụng trong thực tế. Nó giúp cho việc tương tác với các mô hình ngôn ngữ lớn trở nên dễ dàng hơn và cho phép các ứng dụng tận dụng thêm các thông tin từ nhiều nguồn data khác của bên thứ 3 như Google, Notion, Facebook… cũng như cung cấp các component cho phép sử dụng các language model trong nhiều tình huống khác nhau trên thực tế.



Hình 3: Flowise, visual tool để xây dựng các ứng dụng LLM, được xây dựng trên nền tảng Langchain

1.10. ChatGPT

ChatGPT là một chatbot AI hoạt động dựa trên mô hình GPT-3.5 được phát triển bởi OpenAI. ChatGPT có khả năng tương tác với người dùng thông qua việc trả lời các câu hỏi và hoàn thành các tác vụ liên quan đến ngôn ngữ như viết kịch bản, lời thoại, dịch thuật, tìm kiếm thông tin,… mà không giới hạn về chủ đề. ChatGPT được đào tạo bằng phương pháp Học tăng cường từ phản hồi của con người (RLHF – Reinforcement Learning from Human Feedback) [5], nên có thể hiểu ngữ cảnh, ghi nhớ thông tin người dùng nói, dự đoán nhu cầu của họ để đưa ra các phản hồi chính xác nhất. ChatGPT là một ứng dụng nổi bật của GPT-3, một trong những mô hình xử lý ngôn ngữ tự nhiên (Natural Language Processing) tiên tiến nhất hiện nay. ChatGPT có thể được áp dụng cho nhiều lĩnh vực khác nhau như chăm sóc khách hàng, sáng tạo nội dung, giáo dục,… ChatGPT là một bước tiến quan trọng trong lĩnh vực trí tuệ nhân tạo và có tiềm năng thay đổi cách con người giao tiếp và học tập trong tương lai.

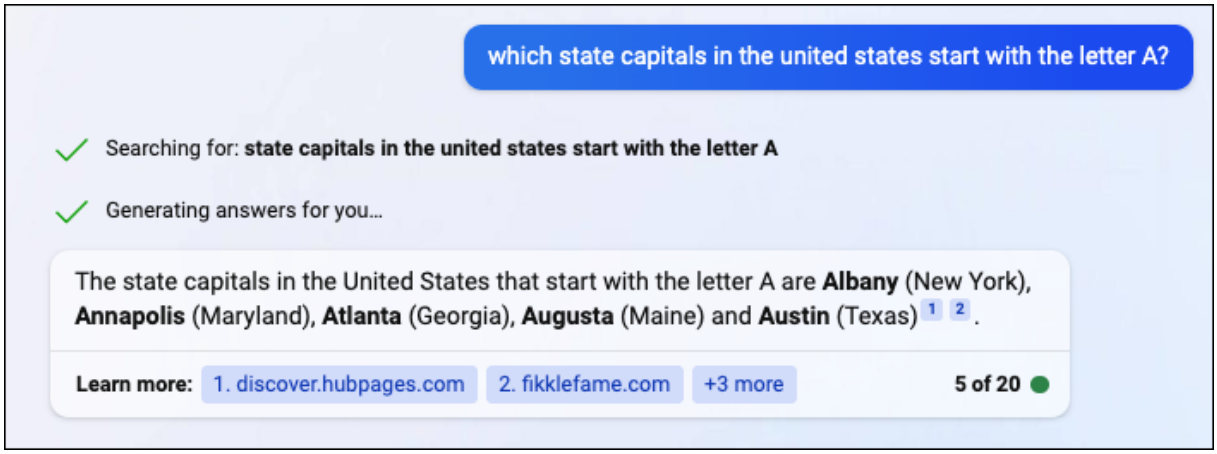


Hình 4: Sơ đồ hoạt động của ChatGPT

1.11. Bing AI

Bing AI[6] là một chatbot trí tuệ nhân tạo (AI) được phát triển bởi Microsoft và ra mắt vào năm 2023. Nó được xây dựng trên nền tảng của mô hình ngôn ngữ lớn (LLM) GPT-4 của OpenAI và đã được tinh chỉnh sử dụng cả các kỹ thuật học có giám sát và học tăng cường.

Bing AI không chỉ sinh văn bản dựa theo xác suất như ChatGPT của OpenAI, mà còn có thể dẫn được nguồn của văn bản mà nó tham chiếu tới do đó nội dung có tính xác thực cao hơn. Ngoài ra, Bing AI còn có thể trả lời các câu hỏi phức tạp, tương tác với người dùng qua chat, và tạo ra nội dung sáng tạo như thơ, truyện, mã nguồn, bài viết, bài hát và nhiều thứ khác.



Hình 5: Giao diện của Bing AI

1.12. Open-Domain Question Answering

Open-domain Question Answering (ODQA) là một loại nhiệm vụ ngôn ngữ, yêu cầu mô hình tạo ra câu trả lời cho các câu hỏi bằng ngôn ngữ tự nhiên. Câu trả lời đúng là khách quan, vì vậy ta có thể dễ dàng đánh giá hiệu suất của mô hình.

Ví dụ:

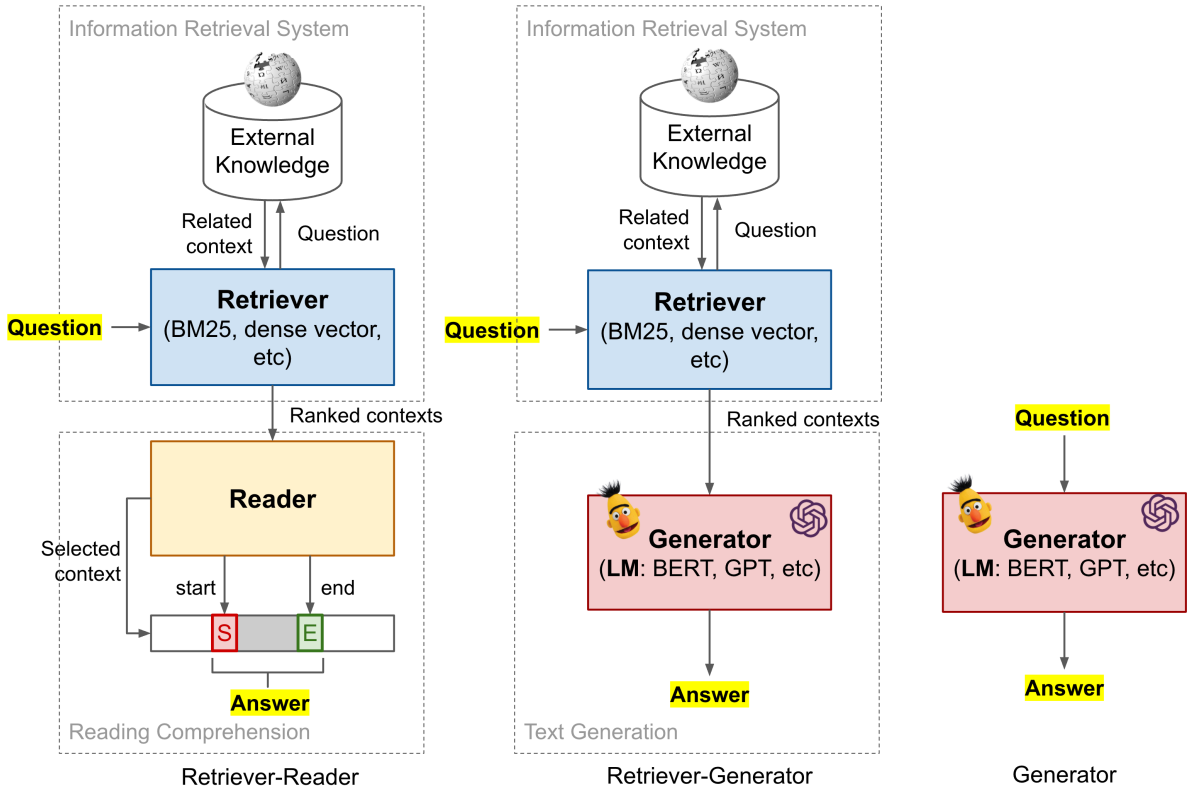
```
Question: What did Albert Einstein win the Nobel Prize for?
Answer: The law of the photoelectric effect.
```

Phần "open-domain" đề cập đến việc thiếu bối cảnh liên quan đến bất kỳ câu hỏi thực tế nào được hỏi một cách tùy ý. Trong trường hợp trên, mô hình chỉ lấy câu hỏi làm đầu vào nhưng không cung cấp bất kỳ dẫn chứng nào về "What did Albert Einstein win the Nobel Prize for", trong đó thuật ngữ "The law of the photoelectric effect" có thể được đề cập. Trong trường hợp cả câu hỏi và bối cảnh được cung cấp, nhiệm vụ được gọi là **Reading comprehension (RC)**.

Một mô hình ODQA có thể hoạt động với hoặc không có quyền truy cập vào nguồn tri thức bên ngoài (ví dụ: Wikipedia) và hai điều kiện này được gọi là open-book hoặc closed-book, trả lời câu hỏi mở hoặc đóng.

Khi xét về các loại câu hỏi open-domain khác nhau, phân loại của Lewis, et al., 2020[7] được xem là khá phù hợp, phân loại theo thứ tự tăng dần độ khó:

1. Một mô hình có thể trả lời đúng với câu trả lời cho một câu hỏi đã được thấy trong quá trình huấn luyện.
2. Một mô hình có thể trả lời đúng với câu trả lời cho một câu hỏi mới ở thời gian kiểm tra và chọn một câu trả lời từ tập các câu trả lời mà nó đã thấy trong quá trình huấn luyện.
3. Một mô hình có thể trả lời đúng với các câu hỏi mới có câu trả lời không có trong tập dữ liệu huấn luyện.



Hình 6: Sơ lược về 3 mô hình ODQA

Như đã đề cập từ trước, ChatGPT là Generator, Bing AI là retriever-Generator.

1.13. Multimodal model

Multimodal Model là một hệ thống trí tuệ nhân tạo xử lý nhiều dạng dữ liệu cảm quan cùng lúc. Học trong Multimodal Model kết hợp các dữ liệu từ các cảm biến và nguồn khác vào một mô hình, tạo ra các dự đoán linh hoạt hơn.

Multimodal Model gồm nhiều mạng nơ-ron unimodal, xử lý từng dạng dữ liệu riêng biệt. Sau đó, các đặc trưng được mã hóa từ các mạng unimodal được kết hợp lại để tạo ra một đại diện chung cho tất cả các dạng dữ liệu. Cuối cùng, đại diện chung này được sử dụng để thực hiện các nhiệm vụ mong muốn.

Multimodal Model là đề tài nóng của trí tuệ nhân tạo. Ví dụ nổi bật là GPT-4 của OpenAI, một mô hình lớn xử lý văn bản và hình ảnh và tạo ra văn bản. GPT-4 đã đạt được hiệu suất ở mức con người trên nhiều tiêu chuẩn chuyên nghiệp và học thuật. Multimodal Model có tiềm năng ứng dụng trong nhiều lĩnh vực khác nhau.

User Answer question I.1.a. Think step-by-step.

I. Principe de la détection de rayonnement avec un bolomètre

Comme illustré sur la figure 1 un bolomètre est constitué d'un absorbeur qui reçoit le rayonnement que l'on désire détecter. Sa température T, supposée uniforme, est mesurée à l'aide d'un thermomètre incorporé, constitué d'un matériau conducteur dont la résistance $R(T)$ varie avec la température T ; cette variation est caractérisée par le coefficient $\alpha = \frac{1}{R}\frac{dR}{dT}$. L'ensemble possède la capacité thermique C_{th}.

Un barreau, conducteur thermique, homogène, de longueur L, de section S et de conductivité thermique λ et sans échanges thermiques latéraux, relie le bolomètre à un thermostat de température T_b fixe.

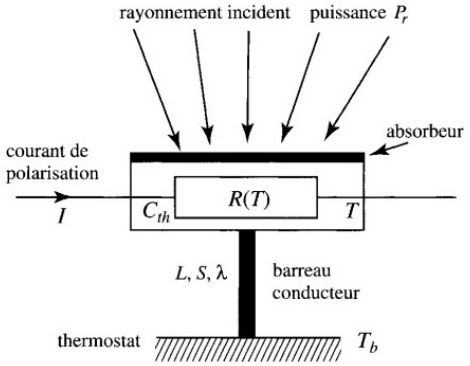


Figure 1 - Schéma d'un bolomètre

I.1. Dans cette première partie, on étudie le comportement du bolomètre en régime stationnaire, à la température de fonctionnement $T = T_0$.

I.1.a) Exprimer la température $T(x)$ en tout point du barreau conducteur repéré par son abscisse x ; on prendra $T(0) = T_0$ et $T(L) = T_b$.

GPT-4 To answer question I.1.a, we need to express the $T(x)$ in every point of the conductive bar, which is characterized by the abscissa x. We are given that $T(0) = T_0$ and $T(L) = T_b$.

For a stationary regime, the heat equation for the one-dimensional heat conduction in the bar is:
$d^2T/dx^2 = 0$

Integrating once, we get:
$dT/dx = C_1$

Integrating a second time:
$T(x) = C_1x + C_2$

To determine the constants C_1 and C_2, we use the boundary conditions:
$T(0) = T_0$, which gives $C_2 = T_0$
$T(L) = T_b$, which gives $C_1 = (T_b - T_0)/L$

Finally, the expression for $T(x)$ is:
$T(x) = (T_b - T_0) * (x/L) + T_0$

This represents a linear temperature profile along the conductive bar, with a slope of $(T_b - T_0)/L$.

,

Hình 7: Ví dụ về Multimodal Model, người dùng yêu cầu GPT4 trả lời câu hỏi về bài tập vật lý được viết bằng tiếng Pháp, bài tập là một hình ảnh.

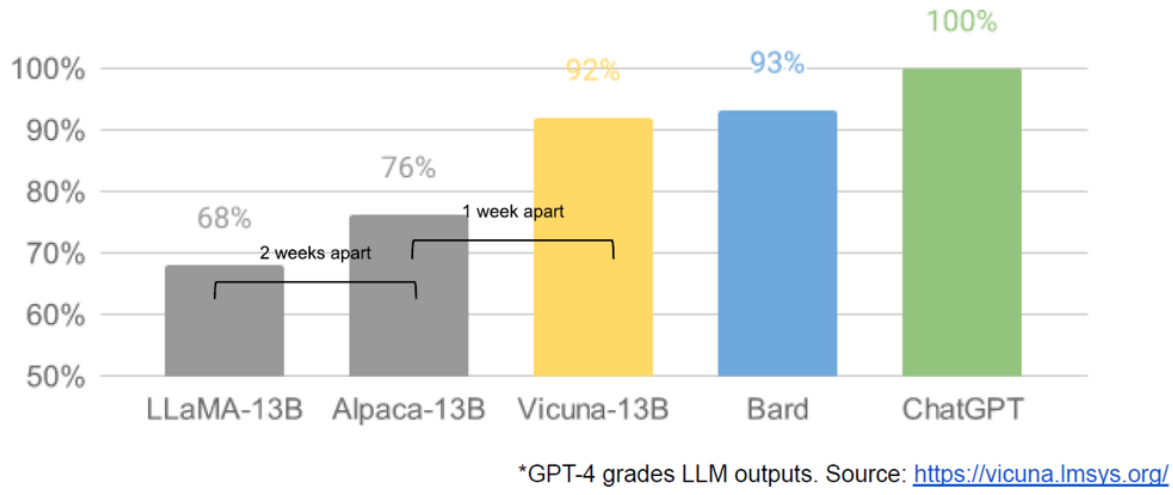
2. Cơ hội và thách thức

Điểm sơ qua quá trình hình thành và phát triển của các mô hình ngôn ngữ lớn:

- Ngày 30 tháng 11 năm 2022, **OpenAI** phát triển và ra mắt **ChatGPT**, chatbot trí tuệ nhân tạo đầu tiên.
- Ngày 6 tháng 2 năm 2023, **Google** cho ra mắt **Bard** cạnh tranh trực tiếp với **ChatGPT**.
- Ngày 7 tháng 2 năm 2023, **Microsoft** công bố phiên bản search engine **Bing** dựa trên mô hình GPT tương tự như **ChatGPT**
- Ngày 24 tháng 2 năm 2023, **Meta** cũng đã ra mắt **LLaMA**, mã nguồn mỡ, nhưng không công bố trọng số.

Trong vòng một tuần, **LLaMA** bị rò rỉ ra công chúng. Từ thời điểm này trở đi, những đổi mới trở nên mạnh mẽ và nhanh chóng hơn.

Điển hình như mô hình Vicuna[8] được xây dựng dựa trên mô hình LLaMA, chỉ sau 3 tuần kể từ khi trọng số LLaMA bị rò rỉ. Độ chính xác so với ChatGPT đạt tới 90%, với chi phí 300$.

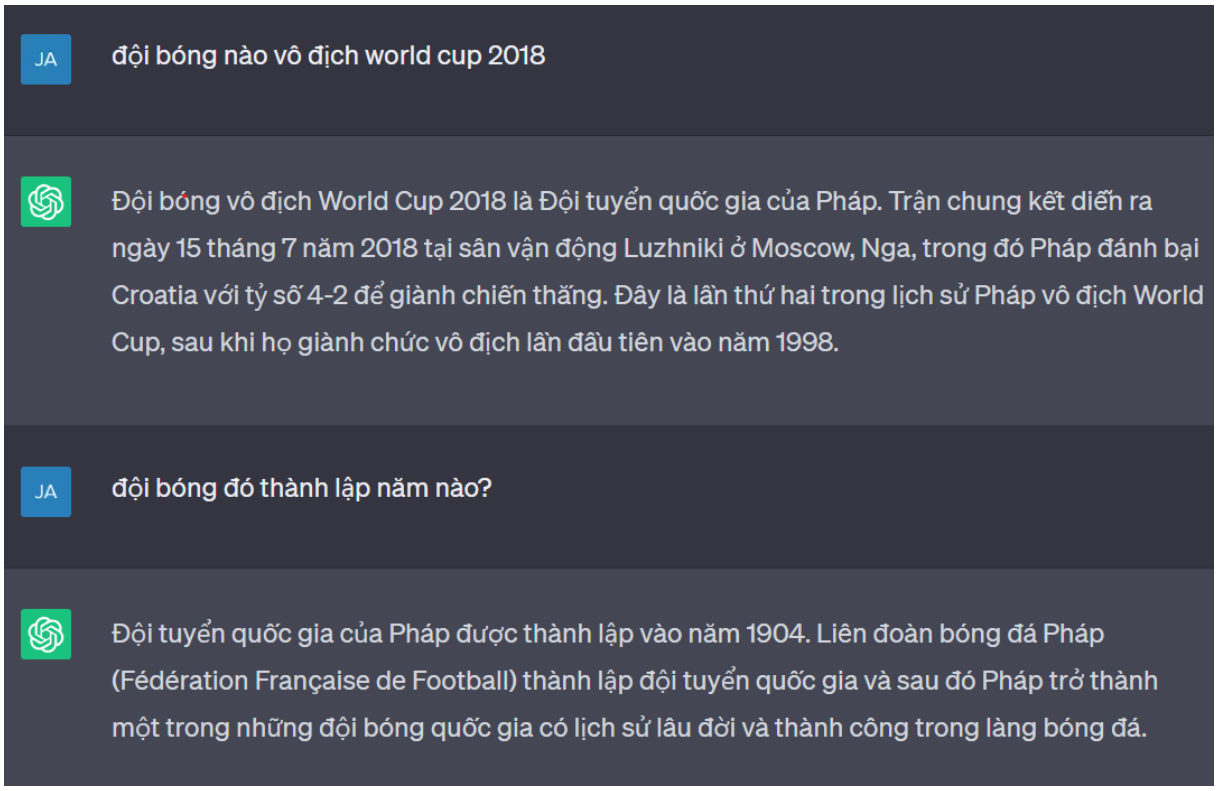


Hình 8: Chất lượng phản hồi của các mô hình được đánh giá bởi GPT-4

Để hiểu rõ thêm về ứng dụng của các mô hình này trong lĩnh vực pháp luật, ta sẽ đi vào phân tích chi tiết các ứng dụng của chúng.

2.1. Sử dụng LLM để tra cứu và soạn thảo

Sự ra đời của ChatGPT đã thay đổi cách chúng ta tìm kiếm nội dung trên internet. Thay vì phải nhập từ khóa, chúng ta có thể đặt câu hỏi và nhận được câu trả lời ngay lập tức. Và ChatGPT còn có thể hiểu được cuộc hội thoại đang diễn ra và đưa ra những câu trả lời phù hợp. Điều này giúp chúng ta tiết kiệm thời gian và tăng cường hiệu quả tìm kiếm.



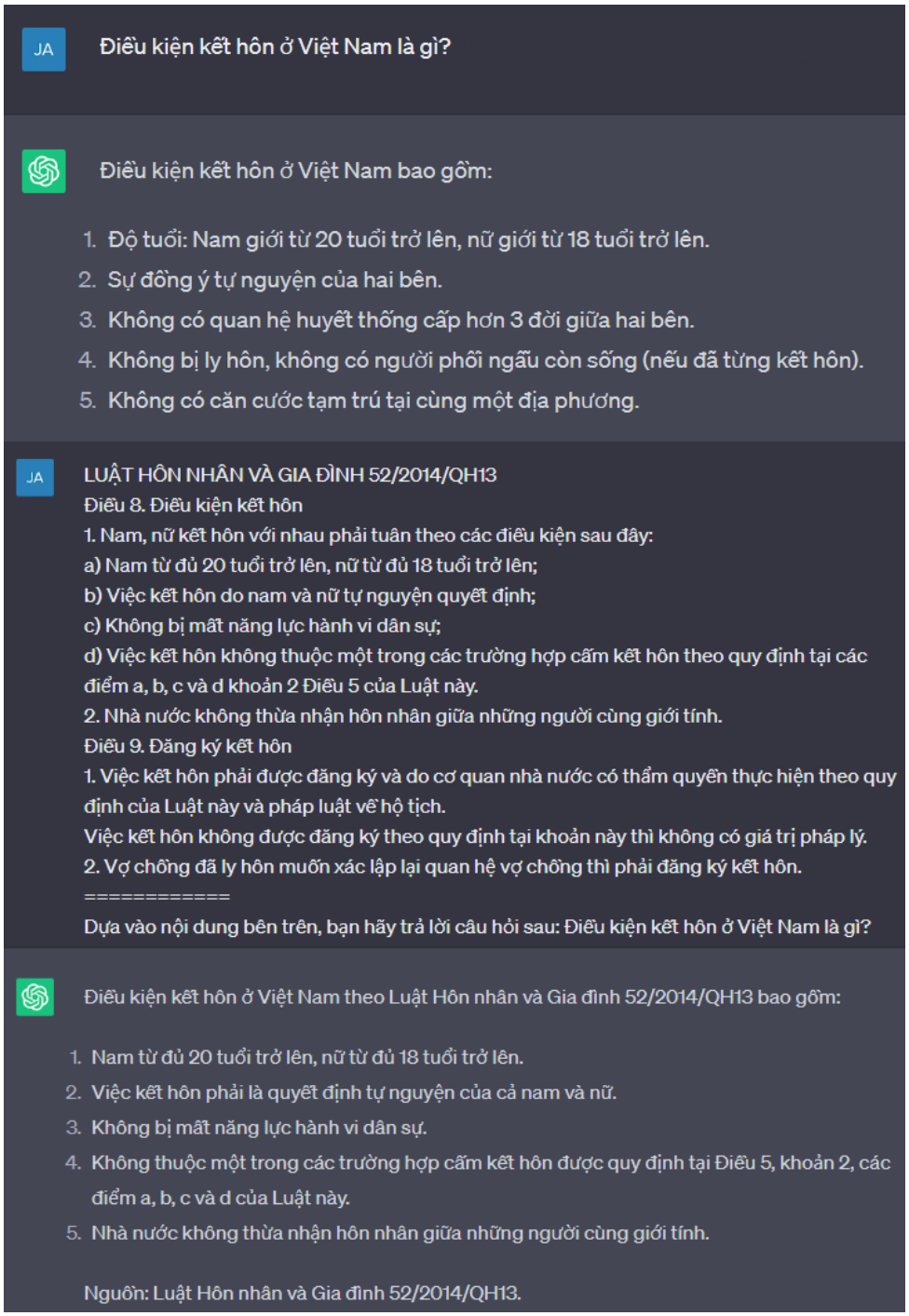
Hình 9: Một cuộc hội thoại với ChatGPT

Về bản chất thì ChatGPT là một mô hình ngôn ngữ, cách hoạt động có thể hiểu đơn giản là nó dự đoán xác suất của từ tiếp theo dựa theo các từ đã xuất hiện trước đó. Ví dụ:

$$\text{Con mèo đang đuổi theo con \_\_\_\_} \begin{cases} \text{chó } 5\% \\ \text{chuột } 70\% \\ \text{sóc } 20\% \\ \text{dế } 5\% \\ \text{ngôi nhà } 0\% \end{cases}$$

Do đó, không có gì chứng minh được câu trả lời là chính xác tuyệt đối. Để giải quyết vấn đề này, nhiều nhà nghiên cứu đã đưa thêm nội dung từ search engine (công cụ tìm kiếm) để ChatGPT có thể thông minh hơn. Cụ thể, trước khi trả lời cho một câu hỏi, ta sẽ thực hiện thêm một bước tìm kiếm các nội dung liên quan đến câu hỏi, sau đó đưa các nội dung này vào mô hình để tạo ra câu trả lời, theo format:

```
{nội dung tìm kiếm}
========
Dựa vào nội dung bên trên, bạn hãy trả lời câu hỏi sau: {câu hỏi}
```

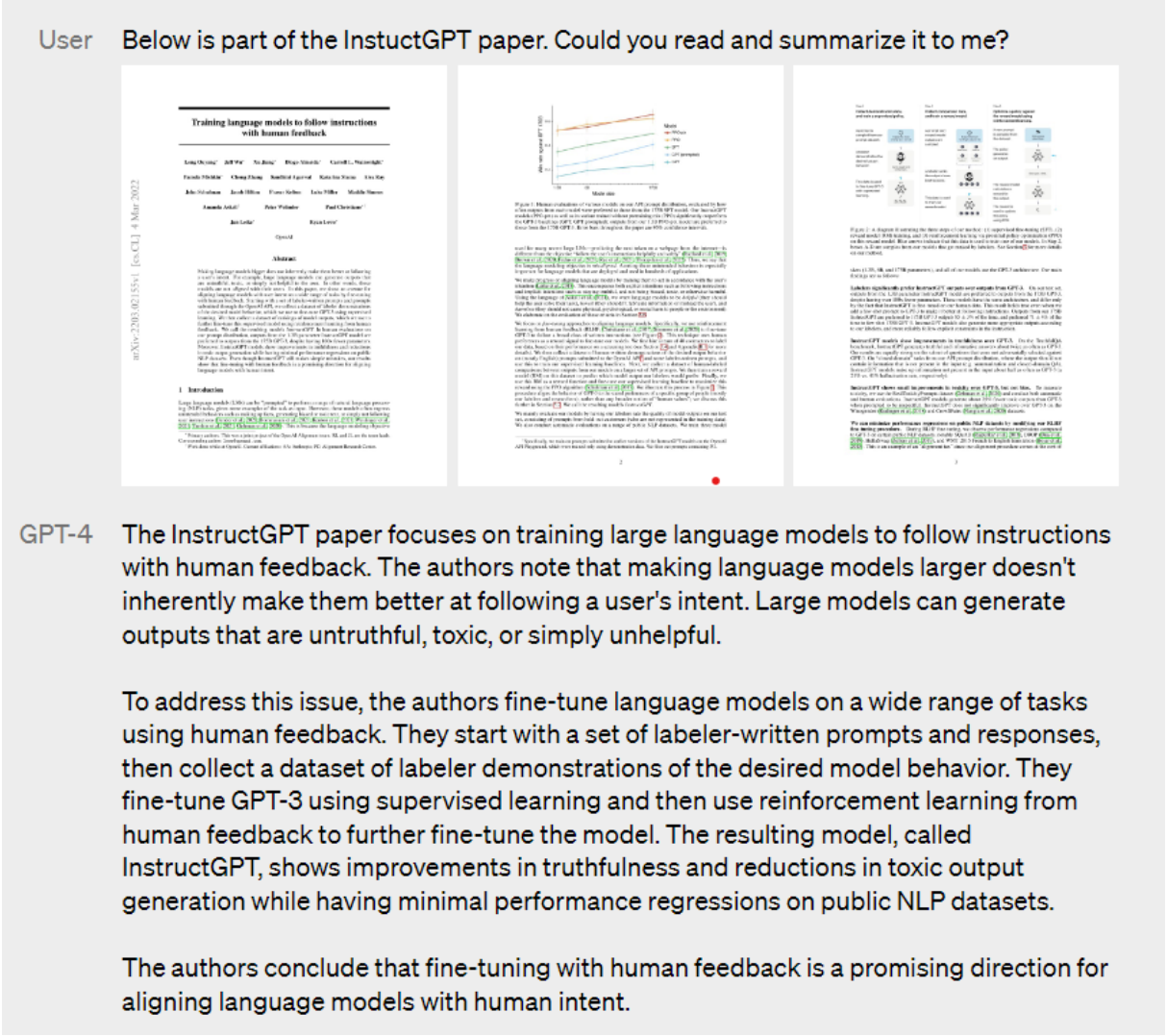


Hình 10: So sánh câu trả lời của ChatGPT không có và có nội dung tìm kiếm

Cách làm này đã tạo ra một định nghĩa, ngành nghề hoàn toàn mới: **prompt engineering**, là quá trình tìm kiếm, lựa chọn và sắp xếp các từ, cụm từ hoặc câu văn để hướng dẫn mô hình ngôn ngữ tạo ra các nội dung hữu ích và phù hợp với mục đích và yêu cầu của người dùng.

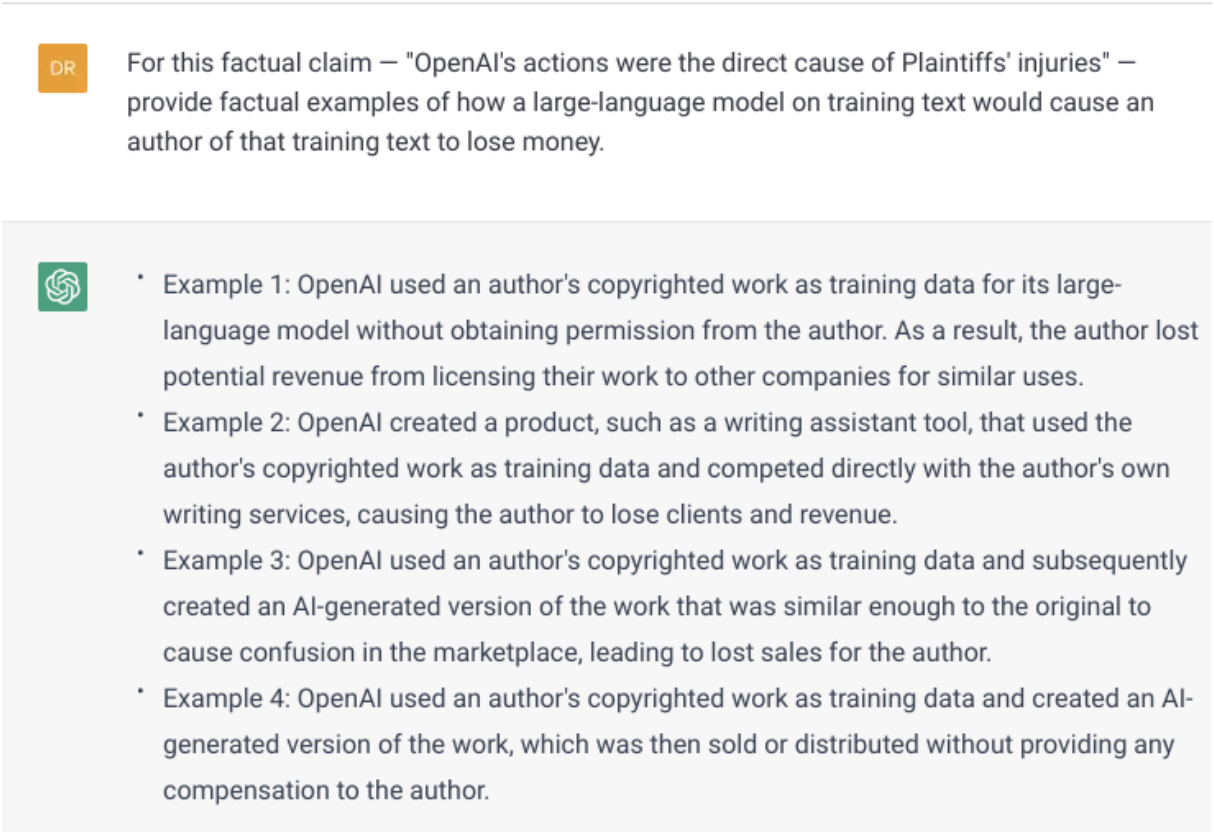
Phiên bản GPT-4 được ra mắt vào ngày 9 tháng 3 năm 2023, mang lại vô số tính năng cải tiến. Trong đó, tính năng đặc biệt nhất là người dùng có thể thêm hình ảnh vào câu hỏi. Từ đó ta có thể upload tài liệu của mình và hỏi chatbot

những nội dung xoay quanh đó. Như hợp đồng, báo cáo tài chính, bản thỏa thuận…



Hình 11: GPT-4 tóm tắt nội dung của một bài báo khoa học

Các nhà nghiên cứu tại MIT đã có một thử nghiệm nhỏ trên lĩnh vực luật[9]. Họ đã yêu cầu ChatGPT đưa ra các lập luận pháp lý để đáp lại "Kiến nghị của OpenAI, về việc bác bỏ vụ kiện Copilot". Tóm tắt kết quả như sau: từ một văn bản dài 25 trang, các nhà nghiên cứu yêu cầu ChatGPT tóm tắt lại nội dung của văn bản, và đưa ra ví dụ chứng minh cho một tuyên bố: "Hành động của OpenAI là nguyên nhân trực tiếp dẫn đến tổn thất của Nguyên đơn". Thời gian để ChatGPT tạo ra kết quả trong vỏn vẹn một phút, trong khi đó thời gian để một luật sư hay tư vấn viên có thể đưa ra kết quả tương tự là hơn 1 giờ, với chi phí từ 500$/giờ.



Hình 12: Kết quả cuối cùng của thí nghiệm ở MIT

Với sức mạnh của GPT-4, ta có thể thấy rõ ràng rằng, trong tương lai, các công việc liên quan đến tìm kiếm thông tin, tìm hiểu vấn đề, đọc hiểu văn bản, sẽ được thay thế bởi các chatbot thông minh. Điều này sẽ giúp con người tiết kiệm được rất nhiều thời gian và chi phí, đồng thời tăng cường hiệu quả công việc.

2.2. Robot luật sư

Một xu hướng mới đang nổi lên trong lĩnh vực AI là "cá nhân hóa" các mô hình ngôn ngữ lớn (LLM), tức là huấn luyện riêng biệt cho từng người dùng cụ thể thay vì cho một mô hình chung chung.

Xu hướng này được thể hiện rõ nét qua sản phẩm Claude của Anthropic. Thay vì chỉ huấn luyện trên các tập dữ liệu công khai, Anthropic đã thu thập dữ liệu riêng từ các tình nguyện viên để Claude có thể hiểu rõ và phù hợp với từng cá nhân. Một số công ty khác cũng đang thử nghiệm các phiên bản LLM cá nhân hóa, ví dụ Bard của Google có thể được huấn luyện riêng cho người dùng nếu họ cho phép chia sẻ dữ liệu cá nhân.

Các nhà phát triển hy vọng rằng với dữ liệu cá nhân hóa, LLM sẽ hiểu ngữ cảnh và ngôn ngữ tốt hơn cho từng cá nhân, từ đó có thể đưa ra phản hồi chính xác và thông minh hơn. Tuy nhiên, điều này cũng khiến người dùng lo ngại về việc liệu dữ liệu của họ có thể bị lộ ra ngoài hay không. Các công ty cần

tìm cách cân bằng giữa lợi ích mang lại và việc bảo mật thông tin người dùng trong xu hướng cá nhân hóa LLM.

DoNotPay[10] là một công ty khởi nghiệp công nghệ đứng sau ứng dụng được gọi là "robot luật sư đầu tiên trên thế giới", sử dụng trí tuệ nhân tạo để bảo vệ quyền lợi của người tiêu dùng. Trí tuệ nhân tạo này sẽ hướng dẫn các bị cáo cách trả lời trước tòa án bằng cách sử dụng một tai nghe có khả năng kết nối Bluetooth, theo bài báo của Matthew Sparkes trên tạp chí New Scientist[11].

Joshua Browder, người sáng lập DoNotPay, trong bài phỏng vấn với David Lumb của CNET[12], cho biết các dịch vụ và phí pháp lý có thể đắt đỏ, ngăn cản một số người thuê luật sư truyền thống để đấu tranh cho họ tại tòa án. "Hầu hết mọi người không đủ khả năng đại diện pháp lý". Luật sư AI "sẽ là một bằng chứng về khái niệm cho các tòa án cho phép sử dụng công nghệ trong phòng xử án".

Robot luật sư của DoNotPay sẽ được cung cấp âm thanh của các quá trình tố tụng tại tòa án khi chúng diễn ra, sau đó nó sẽ phản hồi bằng các lập luận pháp lý. Các bị cáo đã đồng ý lặp lại với thẩm phán chính xác những gì chatbot nói.

Robot luật sư sẽ sử dụng GPT-J[13], một mô hình ngôn ngữ mã nguồn mở được phát hành bởi EleutherAI[4]. DoNotPay đã huấn luyện chatbot để tranh luận bằng cách sử dụng các sự kiện thay vì bịa ra mọi thứ để thắng kiện. Họ cũng lập trình nó để đôi khi giữ im lặng, vì không phải mọi thứ trước tòa đều cần có phản hồi.

Vụ án đầu tiên liên quan đến luật sư robot được ấn định vào ngày 22 tháng 2, Browder tiết lộ trên Twitter vào ngày 21 tháng 1. Vụ án này liên quan đến một cáo buộc vi phạm tốc độ khi tham gia giao thông.

Hình 13: Công bố vụ án đầu tiên của robot luật sư trên Twitter

[4]EleutherAI là một phòng thí nghiệm nghiên cứu AI phi lợi nhuận tập trung vào khả năng diễn giải và căn chỉnh của các mô hình lớn.

Tuy còn nhiều tranh cãi xoay quanh vấn đề này, nhưng có thể nói rằng đây là một bước tiến quan trọng trong việc ứng dụng trí tuệ nhân tạo vào lĩnh vực pháp lý. Với tốc độ phát triển của ngành AI hiện nay, khả năng robot luật sư trở thành một thứ gì đó "bình dân" không còn là điều quá xa vời.

2.3. Thách thức

Tuy các ứng dụng của các mô hình ngôn ngữ lớn mang lại nhiều lợi ích, nhưng cũng có những thách thức cần được giải quyết. Một số vấn đề chung thế giới đang phải đối mặt đó là:

Bảo mật thông tin:

Hiện nay các model LLM có độ chính xác cao nhất đều là do bên thứ 3 cung cấp: OpenAI, Cohere, Stability AI… Do đó, những doanh nghiệp có dữ liệu nhạy cảm rất khó để sử dụng các công cụ này. Theo tạp chí Fortune[14], một số công ty như Apple, Samsung… đã cấm nhân viên của mình sử dụng ChatGPT vì lo ngại các thông tin nhạy cảm có thể bị rò rỉ.

Do vậy, để dùng được các công cụ này, các doanh nghiệp phải tự chủ được các công nghệ AI. Mà để tự chủ được các công nghệ này thì cần một chi phí cực kì cao. Theo Business Insider, một ngày OpenAI có thể phải trả tới 700,000 USD để duy trì hệ thống của mình.[15]

Chi phí cao:

Chi phí để huấn luyện một mô hình ngôn ngữ lớn được The Next Platform[16] thống kê tại Bảng 1. Mô hình càng lớn thì chi phí và thời gian huấn luyện càng cao.

Model Name	Parameter (billion)	Tokens (billion)	Days to train	Price to train	Cost per 1M parameters
GPT-3XL	1.3	26	0.4	$2500	$1.92
GPT-J	6	120	8	$45000	$7.50
GPT-3 6.7B	6.7	134	11	$40000	$5.97
T-5 11B	11	34	9	$60000	$5.45
GPT-3 13B	13	260	39	$150000	$11.54
GPT NeoX	20	400	47	$525000	$26.25
GPT 70B	70	1400	85	$2500000	$33.71
GPT 175B	175	3500	110	$8750000	$50.00

Bảng 1: Chi phí để huấn luyện một mô hình ngôn ngữ lớn

Chi phí để sử dụng các model LLM do bên thứ 3 cung cấp được tính bằng tokens. Tokens là một đơn vị đo lường được sử dụng để đo lường số lượng từ

được sử dụng trong một câu. Ví dụ: "Con mèo đang đuổi theo con chuột" có 7 tokens. Do đó, nếu lưu lượng sử dụng công cụ AI càng nhiều thì chi phí phải bỏ ra để vận hành càng cao.

Name	Model	Input	Output
GPT-4	8K context	$0.03 / 1K token	$0.06 / 1K token
GPT-4	32K context	$0.06 / 1K token	$0.12 / 1K token
Chat	4K context	$0.0015 / 1K token	$0.002 / 1K token
Chat	16K context	$0.003 / 1K token	$0.004 / 1K token
InstructGPT	Ada	$0.0004 / 1K token	$0.0004 / 1K token
InstructGPT	Babbage	$0.0005 / 1K token	$0.0005 / 1K token
InstructGPT	Curie	$0.0020 / 1K token	$0.0020 / 1K token
InstructGPT	Davinci	$0.0200 / 1K token	$0.0200 / 1K token

Bảng 2: Bảng giá của OpenAI theo số token đầu vào và đầu ra

Ta có thể sử dụng các model LLM open source có độ hiệu quả tương tự như Bloom, GPT-J… Nhưng để chạy được những model này thì cần một lượng lớn phần cứng. Điển hình như Bloom[17], để chạy model Bloom 176B (176 billion parameters) cần đến 8 card đồ họa A100 80GB. Với chi phí khoản 15000 USD cho một card, tổng chi phí để chạy model này lên tới 120,000 USD chưa tính đến chi phí vận hành và bảo trì.

Các nhà nghiên cứu đã sáng tạo ra nhiều phương pháp để khắc phục các nhược điểm này. Điển như hình LoRa[18] dùng để fine-tune các model lớn, giúp model học thêm các kiến thức mới mà không tốn quá nhiều tài nguyên. Hay GPTQ[19] viết tắt của Generative Pre-Training Quantized, một phương pháp "nén" mô hình để có thể chạy trên các thiết bị yếu hơn, không cần GPU. Đây cũng là một trong các mục tiêu mà các nhà nghiên cứu đang hướng đến: "chạy mô hình ngôn ngữ lớn trên mọi thiết bị" như điện thoại, laptop, các thiết bị điện tử trong công nghiệp…

Đạo đức và pháp lý:

Theo Reuters ngày 27/1, Science Po, một trong những trường đại học hàng đầu của Pháp, đã tuyên bố cấm sử dụng ChatGPT[20]. Hình phạt đối với hành vi sử dụng phần mềm này có thể nặng tới mức bị đuổi khỏi trường, thậm chí là toàn bộ nền giáo dục đại học của Pháp. Lý do cho quyết định này là ChatGPT có thể tạo ra những câu trả lời sai lầm hoặc vô nghĩa, gây hiểu lầm và nhầm lẫn cho người học. Ngoài ra, ChatGPT cũng có thể bị lợi dụng để gian lận trong các bài kiểm tra hoặc thi cử.

Science Po không phải là trường đầu tiên làm như vậy. Vào đầu năm 2023, nhiều trường học ở Mỹ đã có những biện pháp tương tự để ngăn chặn việc gian lận bằng ChatGPT. Họ đã giảm bớt bài tập về nhà và yêu cầu học sinh và giáo viên không sử dụng ChatGPT trong quá trình học tập. Họ cũng đã thiết lập các hệ thống để kiểm soát quyền truy cập vào ChatGPT trên các thiết bị hoặc Internet do trường quản lý. Những biện pháp này nhằm bảo vệ chất lượng giáo dục và khuyến khích học sinh tự học mà không phụ thuộc vào AI.

"Nhiều thập kỷ trước, các trường đại học phải đối mặt với một vấn đề nhức nhối là đạo văn và tình trạng vay mượn ý tưởng. Giờ đây, cộng đồng giáo dục tiếp tục đối mặt với một thách thức mới liên quan đến việc sử dụng hệ thống mạng và trí tuệ nhân tạo trong các hoạt động khoa học và giáo dục", tuyên bố của trường đại học Russische Staatliche Geisteswissenschaftliche Universität ở Nga

Getty Images, còn được biết đến là kho ảnh trực tuyến, vào ngày 17/1 đã cáo buộc công ty công nghệ Stability AI sử dụng hình ảnh của hãng và của các đối tác để kiếm lời.

Nhiều đơn vị phát triển AI như OpenAI không công bố nguồn dữ liệu họ thu thập để huấn luyện mô hình. Còn Stability AI nói rằng quy trình huấn luyện Stable Diffusion dựa vào nguồn dữ liệu mở. Đã có bên độc lập phân tích những nguồn dữ liệu này và đi đến kết luận Stability AI thu thập rất nhiều hình ảnh từ Getty Images và những nguồn hình stock khác trên mạng internet.

Về phần Getty Images, CEO đơn vị cung cấp bản quyền hình ảnh này nói rằng công ty không quan tâm tới những khoản bồi thường về mặt tài chính, mà cũng không có ý định ngăn chặn bất kỳ nhà phát triển thuật toán AI nào, mà thay vào đó là tạo ra một án lệ, một nền tảng để những thủ tục pháp lý tương tự về sau có thể dựa vào, và đương nhiên không loại trừ khả năng Getty Images sẽ có được cho họ một thoả thuận sử dụng hình ảnh từ các nhà phát triển AI.

Từ 2 ví dụ trên, ta có thể thấy vấn đề đạo đức và pháp lý đối với các ứng dụng AI vẫn còn khá mập mờ. Chưa có một đạo luật nào để quy định, hướng dẫn, bảo vệ cho những nhân tố trong lĩnh vực này.

Rất khó khăn để có thể thương mại hóa một ứng dụng AI với điều kiện hiện nay. Vì để gia nhập cuộc chơi này thì cần một số tiền đầu tư rất lớn, nhưng kết quả thu được lại không thể đảm bảo. Nên những công nghệ này hiện nay vẫn chỉ nằm trong các phòng thí nghiệm, hay được sử dụng nội bộ trong các công ty lớn, chưa thể tiếp cận rộng rãi được với đại đa số người dùng.

Thách thức đối với tiếng Việt:

Ngoài các thách thức chung như đã nêu ở trên thì đối với các mô hình ngôn ngữ sử dụng tiếng Việt còn phải đối mặt với một số thách thức riêng.

Phần quan trọng nhất để tạo nên các mô hình ngôn ngữ lớn chính là **dữ liệu**. Hiện nay chưa có một nguồn dữ liệu mở chính thống có kiểm duyệt nào cho tiếng Việt. Các nguồn dữ liệu hiện có đều là từ các trang web, các bài báo, các bài viết trên mạng internet. Những nguồn dữ liệu này chỉ mang tính chất thông tin thông thường, trong đó còn có nhiễu bởi các thông tin độc hại, sai sự thật. Do đó, các mô hình ngôn ngữ lớn cho tiếng Việt hiện nay đều có độ chính xác thấp hơn các mô hình cho tiếng Anh.

Các tri thức tiếng Việt đa số đều ở trong sách vở, các nguồn tại liệu đóng… mà những nguồn này đều không được công khai, nếu khai thác các nguồn dữ liệu này có thể vi phạm bản quyền, sở hữu trí tuệ của các tác giả có liên quan.

Để có thể phát triển mảng AI ở Việt Nam, ta cần một chính sách nới lỏng hơn cho các nhà nghiên cứu có thể tiếp cận các nguồn dữ liệu này. Theo Technomancers.ai[21], chính phủ Nhật Bản đã công bố một chính sách cho phép các ứng dụng AI có thể dùng bất kì loại dữ liệu nào "bất kể đó là vì mục đích phi lợi nhuận hay thương mại, cho dù đó là một hành động không phải là sao chép hay đó là nội dung thu được từ các trang web bất hợp pháp hay cách khác". Cho thấy sự sẵn sàng để cạnh tranh với các nước khác trong lĩnh vực AI.

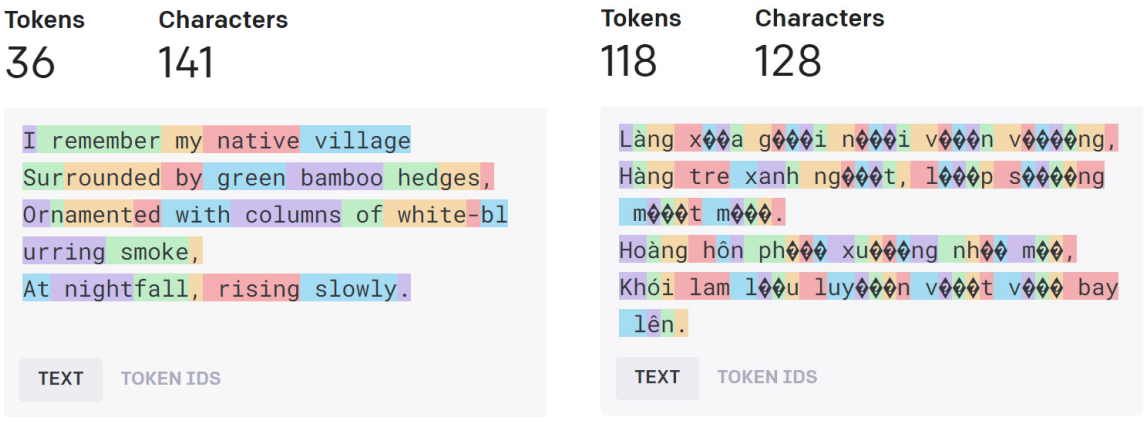
Đối với dữ liệu về văn bản vi phạm pháp luật ở Việt Nam, những văn bản này được nhà nước ban hành và miễn phí sử dụng cho người dân. Vì vậy việc sử dụng các văn bản này sẽ không có vấn đề về pháp lý. Tuy nhiên, dữ liệu còn ở dạng thô (raw data), chưa phù hợp để huấn luyện mô hình vì thế trong Phần 3.1 tôi có đề xuất xây dựng một bộ dữ liệu về văn bản vi phạm pháp luật.

Với dữ liệu hỏi đáp liên quan tới luật thì khó hơn, vì nó cần qua sự kiểm duyệt của con người để đảm bảo tính chính xác. Và người kiểm duyệt phải có chuyên môn về luật. Do đó, việc xây dựng một bộ dữ liệu hỏi đáp liên quan tới luật hiện tại vẫn đang là một việc làm khó khăn và tốn kém.

Một vấn đề khác cũng quan trọng không kém đó là kiến trúc của các model LLM. Hay cụ thể hơn là phần tokenizer[5] của các model này, nó chỉ phù hợp

[5]Tokenizer là một quá trình chuyển đổi văn bản thành các token (đơn vị nhỏ nhất trong xử lý ngôn ngữ tự nhiên) để có thể xử lý dữ liệu bằng máy tính. Các token có thể là từ, ký tự hoặc sub-word. Tokenizer được sử dụng để tạo từ vựng trong một kho ngữ liệu (một tập dữ liệu trong NLP). Từ vựng này sau đó được chuyển thành số (ID) và giúp chúng ta lập mô hình

cho nội dung là tiếng Anh vì cách dùng từ, cách đặt câu của tiếng Anh và tiếng Việt khác nhau. Và bảng mã sử dụng cho hai ngôn ngữ cũng khác, tiếng Anh sử dụng bảng mã ASCII, tiếng việt sử dụng bảng mã Unicode. Ở Hình 14 ta có thể thấy được sự khác biệt này, tuy đoạn chữ tiếng Việt có ít kí tự hơn nhưng số lượng token lại nhiều hơn. Do đó, khi sử dụng các model này với tiếng Việt thì thời gian chạy rất lâu và độ chính xác rất thấp.



Hình 14: So sánh tokenizer của giữa tiếng Việt và tiếng Anh

3. Thử nghiệm

Từ những gì đã làm được trong bài báo **Intelligent Retrieval System on Legal Information** [22] ở hội thảo khoa học quốc tế **ACIIDS**[6], cùng với thầy Nguyễn Thanh Bình và các cộng sự, tôi tiếp tục phát triển các bộ dữ liệu về luật với chủ đề là bảo hiểm xã hội.

Phần cứng sử dụng:

- CPU: Intel Core i5-12400F
- GPU: RTX 3060, riêng fine-tuning sử dụng 4 GPU RTX 4090
- RAM: 32GB DDR4

Trong phạm vi của bài luận này, tôi chỉ sử dụng các văn bản liên quan tới lĩnh vực bảo hiểm xã hội và việc làm, bao gồm 761 điều luật để thử nghiệm và đánh giá. Một số văn bản luật:

- Luật Bảo hiểm xã hội 2014
- Luật việc làm 2013
- Bộ luật Lao động 2019
- Luật Bảo hiểm y tế sửa đổi 2014

3.1. Xây dựng bộ dữ liệu văn bản vi phạm pháp luật

3.1.1. Sơ lược về dữ liệu

Theo dữ liệu từ Thư viện pháp luật[7], hiện nay Việt Nam có trên dưới 300000 văn bản vi phạm pháp luật. Bao gồm 20 loại văn bản và 27 lĩnh vực khác nhau.

Các thuộc tính của một văn bản quy phạm pháp luật gồm: tên văn bản, số hiệu văn bản, loại văn bản, nơi ban hành, người ký, ngày ban hành, ngày hiệu lực, ngày công báo, số công báo.

Ngoài ra các thuộc tính trên, còn có lược đồ thể hiện mối quan hệ giữa các văn bản quy phạm pháp luật dựa trên *văn bản đang tham chiếu*:

- Văn bản được hướng dẫn: là văn bản ban hành trước, có hiệu lực pháp lý cao hơn *Văn bản tham chiếu* và được *Văn bản tham chiếu* hướng dẫn hoặc quy định chi tiết nội dung của nó.

[6] ACIIDS (Asian Conference on Intelligent Information and Database Systems) là hội thảo khoa học quốc tế về nghiên cứu trong lĩnh vực hệ thống thông tin và cơ sở dữ liệu thông minh, được tổ chức từ ngày 24 đến ngày 26 tháng 7 năm 2023 tại Phuket, Thái Lan.

[7] thuvienphapluat.vn là trang chuyên cung cấp cơ sở dữ liệu, tra cứu và thảo luận pháp luật

- Văn bản được hợp nhất: Là văn bản ban hành trước, bao gồm các văn bản được sửa đổi, bổ sung và văn bản sửa đổi, bổ sung, được *Văn bản tham chiếu* hợp nhất nội dung lại với nhau.
- Văn bản bị sửa đổi bổ sung: Là văn bản ban hành trước, bị *Văn bản tham chiếu* sửa đổi, bổ sung một số nội dung.
- Văn bản bị đính chính: Là văn bản ban hành trước, bị *Văn bản tham chiếu* đính chính các sai sót như căn cứ ban hành, thể thức, kỹ thuật trình bày,…
- Văn bản bị thay thế: Là văn bản ban hành trước, bị *Văn bản tham chiếu* quy định thay thế, bãi bỏ toàn bộ nội dung.
- Văn bản được dẫn chiếu: Là văn bản ban hành trước, trong nội dung của *Văn bản tham chiếu* có quy định dẫn chiếu trực tiếp đến điều khoản hoặc nhắc đến nó.
- Văn bản được căn cứ: Là văn bản ban hành trước *Văn bản tham chiếu*, bao gồm các văn bản quy định thẩm quyền, chức năng của cơ quan ban hành *Văn bản tham chiếu* văn bản có hiệu lực pháp lý cao hơn quy định nội dung, cơ sở để ban hành *Văn bản tham chiếu*.
- Văn bản liên quan ngôn ngữ: Là bản dịch Tiếng Anh của *Văn bản tham chiếu*.
- Văn bản hướng dẫn: Là bản tiếng Việt của *Văn bản tham chiếu*.
- Văn bản hợp nhất: Là văn bản ban hành sau, hợp nhất lại nội dung của *Văn bản tham chiếu* và văn bản sửa đổi, bổ sung của *Văn bản tham chiếu*.
- Văn bản sửa đổi bổ sung: Là văn bản ban hành sau, sửa đổi, bổ sung một số nội dung của *Văn bản tham chiếu*.
- Văn bản đính chính: Là văn bản ban hành sau, nhằm đính chính các sai sót như căn cứ ban hành, thể thức, kỹ thuật trình bày,…của *Văn bản tham chiếu*.
- Văn bản thay thế: Là văn bản ban hành sau, có quy định đến việc thay thế, bãi bỏ toàn bộ nội dung của *Văn bản tham chiếu*
- Văn bản liên quan cùng nội dung: Là văn bản có nội dung tương đối giống, hoặc có phạm vi điều chỉnh, đối tượng điều chỉnh tương tự *Văn bản tham chiếu*.

Mục lục của văn bản là phần quan trọng không thể thiếu. Tuy nhiên không phải văn bản nào cũng có mục lục, và cũng không có một định dạng chuẩn cho mục lục. Các chỉ mục thường thấy là: phần, chương, mục, điều, khoản, điểm.

Một số nhận xét và thống kê về dữ liệu:

Loại văn bản	Số lượng
Quyết định	188360
Nghị quyết	30709
Kế hoạch	23301
Thông tư	15067
Thông báo	13588
Chỉ thị	13438
Nghị định	5191
Văn bản khác	2608
Thông tư liên tịch	2605
Văn bản hợp nhất	2162

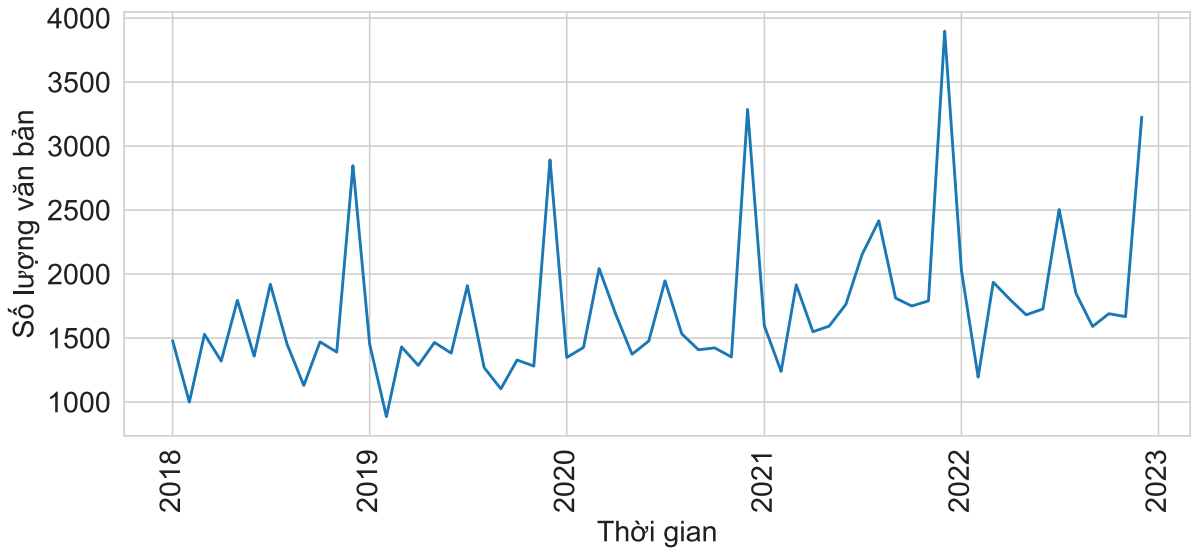
Loại văn bản	Số lượng
Hướng dẫn	1772
Báo cáo	1494
Điều ước quốc tế	1331
Công điện	1244
Sắc lệnh	997
Lệnh	526
Luật	486
Pháp lệnh	228
Văn bản WTO	68
Hiến pháp	5

Bảng 3: Số lượng văn bản vi phạm pháp luật theo loại văn bản

Lĩnh vực	Số lượng
Bộ máy hành chính	105445
Tài chính nhà nước	42216
Văn hóa - Xã hội	39014
Tài nguyên - Môi trường	25490
Thương mại	22388
Xây dựng - Đô thị	21410
Bất động sản	21149
Thể thao - Y tế	19734
Thuế - Phí - Lệ Phí	17592
Giáo dục	16278
Giao thông - Vận tải	14825
Lao động - Tiền lương	14374
Doanh nghiệp	12744
Đầu tư	12718

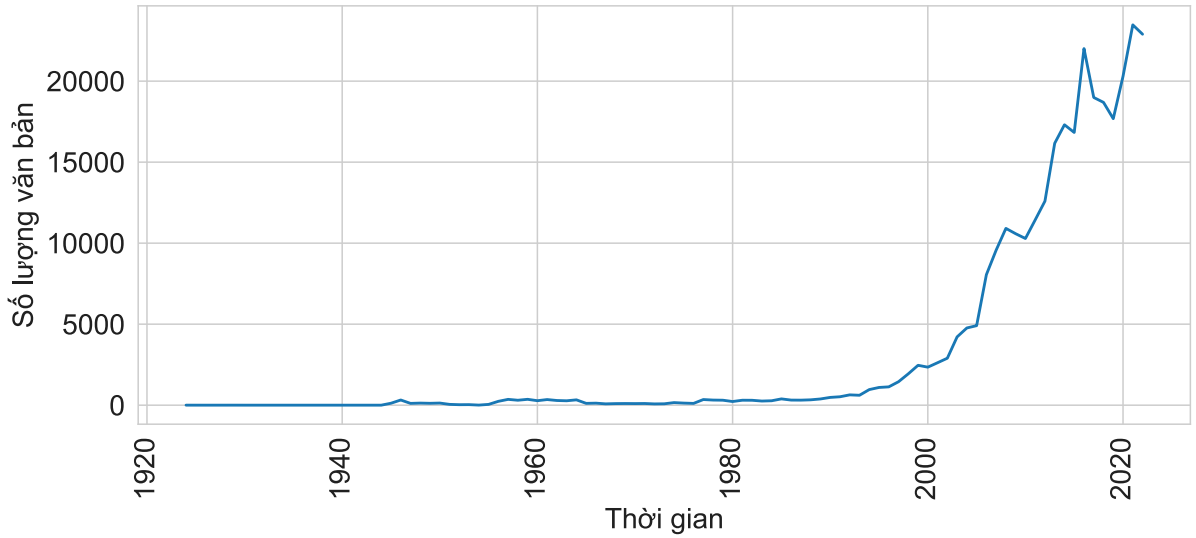
Lĩnh vực	Số lượng
Công nghệ thông tin	12217
Xuất nhập khẩu	11535
Lĩnh vực khác	8607
Quyền dân sự	5505
Tiền tệ - Ngân hàng	4954
Bảo hiểm	2697
Dịch vụ pháp lý	2619
Thủ tục Tố tụng	2350
Vi phạm hành chính	2225
Kế toán - Kiểm toán	1752
Trách nhiệm hình sự	1515
Sở hữu trí tuệ	965
Chứng khoán	771

Bảng 4: Số lượng văn bản vi phạm pháp luật theo lĩnh vực



Hình 15: Số lượng văn bản vi phạm pháp luật ban hành theo tháng của 5 năm gần đây

Từ biểu đồ trên ta thấy: số lượng văn bản thường tăng đột biến vào các tháng giữa và cuối năm. Vì đây là các thời điểm diễn ra Kỳ họp Quốc hội hằng năm, thời điểm các quyết định được nhà nước ban hành nhiều nhất.



Hình 16: Số lượng văn bản vi phạm pháp luật ban hành theo năm

Tuy nhiên, khoảng 10 năm trở lại đây, số lượng văn bản được ban hành tăng rất nhanh. Điển hình như số lượng văn bản được ban hành ở năm 2021 nhiều gấp đôi năm 2011.

3.1.2. Xây dựng cơ sở dữ liệu

Cấu trúc dữ liệu của dữ liệu gồm 3 bảng chính: `VanBan`, `LuocDo`, `ChiMuc` được mô tả như sau:

Tên trường	Kiểu dữ liệu	Mô tả
ten_chi_muc	string	Tên của chỉ mục
loai_chi_muc	string	Loại của mục lục. VD: phần, chương, mục, điều, khoản, điểm…
start_index	integer	Vị trí bắt đầu của nội dung của chỉ mục trong văn bản
end_index	integer	Vị trí kết thúc của nội dung của chỉ mục trong văn bản
parent_id	integer (FK)	ID của chỉ mục cha (nếu có), thể hiển tree structure[8].
vanban_id	integer (FK)	ID của văn bản

Bảng 5: Bảng `ChiMuc`: chứa thông tin về mục lục của văn bản vi phạm pháp luật.

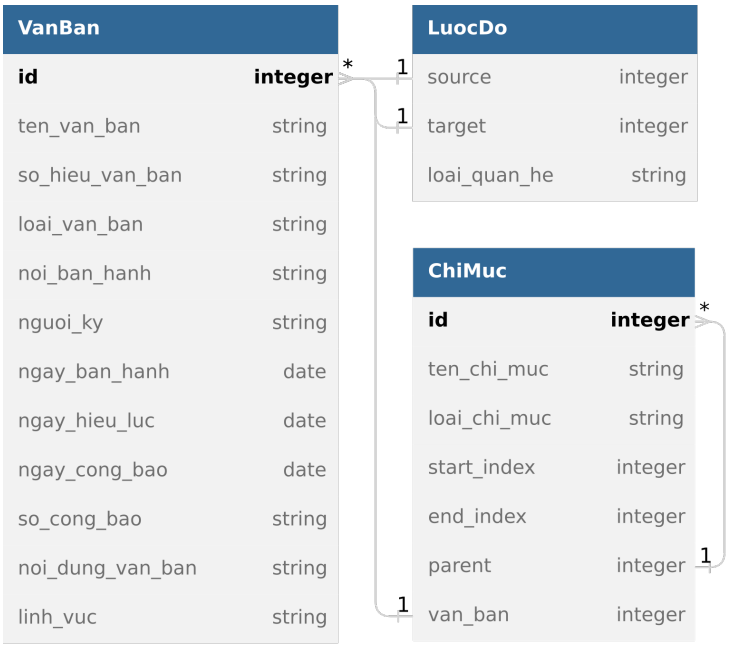
Tên trường	Kiểu dữ liệu	Mô tả
source	integer (FK)	ID của văn bản nguồn
target	integer (FK)	ID của văn bản đích
loai_quan_he	string	Loại quan hệ giữa văn bản nguồn và văn bản đích. VD: thay thế, hướng dẫn, sửa đổi bổ sung…

Bảng 6: Bảng `LuocDo` chứa thông tin về mối quan hệ giữa các văn bản vi phạm pháp luật

Tên trường	Kiểu dữ liệu	Mô tả
id	integer (PK)	ID của văn bản
ten_van_ban	string	Tên văn bản
so_hieu_van_ban	string	Số hiệu văn bản
loai_van_ban	string	Loại văn bản
noi_ban_hanh	string	Nơi ban hành
nguoi_ky	string	Người ký
ngay_ban_hanh	date	Ngày ban hành
ngay_hieu_luc	date	Ngày hiệu lực
ngay_cong_bao	date	Ngày công báo
so_cong_bao	string	Số công báo
noi_dung_van_bang	string	Nội dung văn bản dạng text
linh_vuc	string	Lĩnh vực của văn bản

Bảng 7: Bảng `VanBan` chứa thông tin về văn bản vi phạm pháp luật

[8]Tree structure hay cây là một cấu trúc dữ liệu được sử dụng rộng rãi gồm một tập hợp các nút (node) được liên kết với nhau theo quan hệ cha-con



Hình 17: Cấu trúc dữ liệu của cơ sở dữ liệu văn bản vi phạm pháp luật

Xử lý văn bản: Văn bản sau khi tải xuống có định dạng HTML[9], do đó cần phải xử lý để lấy được nội dung văn bản dạng text. Để làm được điều này, tôi sử dụng thư viện BeautifulSoup[23] để lấy nội dung dạng text của văn bản.

QUỐC HỘI
CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM Độc lập - Tự do - Hạnh phúc
Luật số: 58/2014/QH13
Hà Nội, ngày 20 tháng 11 năm 2014
LUẬT BẢO HIỂM XÃ HỘI
Căn cứ Hiến pháp nước Cộng hòa xã hội chủ nghĩa Việt Nam;
Quốc hội ban hành Luật bảo hiểm xã hội.
Chương I:NHỮNG QUY ĐỊNH CHUNG
Điều 1. Phạm vi điều chỉnh
Luật này quy định chế độ, chính sách bảo hiểm xã hội; quyền và trách nhiệm của người lao động, người sử dụng lao động; cơ quan, tổ chức, cá nhân có liên quan đến bảo hiểm xã hội, tổ chức đại diện tập thể lao động, tổ chức đại diện người sử dụng lao động; cơ quan bảo hiểm xã hội; quỹ bảo hiểm xã hội; thủ tục thực hiện bảo hiểm xã hội và quản lý nhà nước về bảo hiểm xã hội.
Điều 2. Đối tượng áp dụng
1. Người lao động là công dân Việt Nam thuộc đối tượng tham gia bảo hiểm xã hội bắt buộc, bao gồm:
……

Hình 18: Văn bản sau khi xử lý

[9]HTML là viết tắt của cụm từ Hypertext Markup Language (tạm dịch là Ngôn ngữ đánh dấu siêu văn bản). HTML được sử dụng để tạo và cấu trúc các thành phần trong trang web hoặc ứng dụng, phân chia các đoạn văn, heading, titles, blockquotes…

Tạo mục lục: để tạo chỉ mục cho văn bản, tôi sử dụng regex[10] để tìm kiếm các chỉ mục trong văn bản. Tuy nhiên, phương pháp này chỉ là semi-auto, vì có những trường hợp regex không thể tìm được chỉ mục thì phải thêm thủ công.

Regex	**Loại chỉ mục**	**Chú giải**
^(Phần thứ [\d\w]+.*)$	Phần	Tìm các chỉ mục có dạng "Phần thứ <số | chữ> <nội dung>". Ví dụ: "Phần thứ nhất: Những quy định chung"
^(Chương [\d\w]+.*)$	Chương	Tìm các chỉ mục có dạng "Chương <số | chữ> <nội dung>". Ví dụ: "Chương I: ĐIỀU KHOẢN CƠ BẢN"
^(Mục [\d | I | V | X | L | C | D | M]+.*)$	Mục	Tìm các chỉ mục có dạng "Mục <số | chữ số la mã> <nội dung>". Ví dụ: "Mục 1. QUY ĐỊNH CHUNG VỀ QUYẾT ĐỊNH HÌNH PHẠT"
^(Điều \d+.*)$	Điều	Tìm các chỉ mục có dạng "Điều <số> <nội dung>". Ví dụ: "Điều 51. Các tình tiết giảm nhẹ trách nhiệm hình sự"
^(\d+\. .*)$	Khoản	Tìm các chỉ mục có dạng "<số>. <nội dung>". Ví dụ: 1. Người phạm tội phải trả lại tài sản đã…
^(\w\).*)$	Điểm	tìm các chỉ mục có dạng "<chữ>. <nội dung>". Ví dụ: a) Người phạm tội đã ngăn chặn h…

Bảng 8: Các regex để tìm chỉ mục trong văn bản

Để đơn giản khi lập trình, tôi lưu kết quả sau khi xử lý thành định dạng JSON[11]:

[10]Regex là một chuỗi các ký tự đặc biệt được định nghĩa để tạo nên các mẫu (pattern) và được sử dụng để tìm kiếm và thay thế các chuỗi trong một văn bản

[11]JSON là viết tắt của Javascript Object Notation, là một bộ quy tắc về cách trình bày và mô tả dữ liệu trong một chuỗi lớn thống nhất được gọi chung là chuỗi JSON. Chuỗi JSON được bắt đầu bằng ký tự { và kết thúc bởi ký tự }

```
{
  "id":1,
  "ten_van_ban": "Luật Bảo hiểm xã hội 2014",
  "so_hieu_van_ban": "58/2014/QH13",
  "loai_van_ban": "Luật",
  "linh_vuc": "Bảo hiểm, Lao động - Tiền lương",
  "noi_ban_hanh": "Quốc hội",
  "nguoi_ky": "Nguyễn Sinh Hùng",
  "ngay_ban_hanh": "20/11/2014",
  "ngay_hieu_luc": "01/01/2016",
  "ngay_cong_bao": "29/12/2014",
  "so_cong_bao": "Từ số 1163 đến số 1164",
  "noi_dung_van_ban":"...",
  "tree": {
    "ten_chi_muc": "Luật Bảo hiểm xã hội 2014",
    "loai_chi_muc": "root",
    "start_index": 0,
    "end_index": 110927,
    "children": [
      {
        "ten_chi_muc": "Chương I:NHỮNG QUY ĐỊNH CHUNG",
        "loai_chi_muc": "chương",
        "start_index": 280,
        "end_index": 14742,
        "children": [
          {
            "ten_chi_muc": "Điều 1. Phạm vi điều chỉnh",
            "loai_chi_muc": "điều",
            "start_index": 307,
            "end_index": 681,
            "children": []
          },
          {
            "ten_chi_muc": "Điều 2. Đối tượng áp dụng",
            "loai_chi_muc": "điều",
            "start_index": 708,
            "end_index": 3086,
            "children": [...]
          },
          ...
        ]
      }
      ...
    ]
  }
}
```


Hình 19: Kết quả sau khi xử lý văn bản ở định dạng JSON

Sau khi xây dựng được datasets về luật, tôi có tạo thêm một python package dùng để truy vấn dữ liệu một cách thuận tiện hơn[24]. Hình 20 là một đoạn code mẫu để truy vấn dữ liệu.

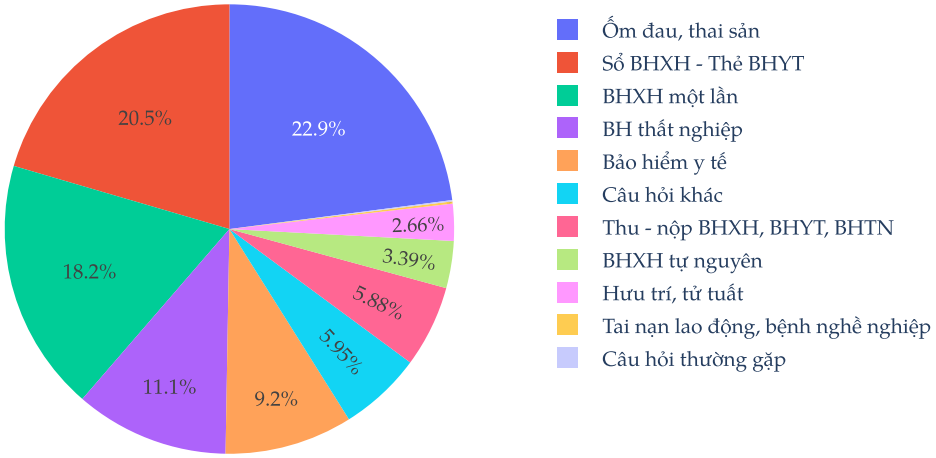
```python
from lawquery import Engine

# create engine, law_id là số hiệu của văn bản luật
engine = Engine(law_id='58/2014/QH13')
# query single node
engine.query(node_type='điều', node_id='1')
# => [Điều 1. Phạm vi điều chỉnh]
engine.query(node_type='phần')
# => [Phần thứ nhất..., Phần thứ hai...]
engine.query(name='hôn nhân')
# => [Điều 67. Các trường hợp hưởng trợ cấp tuất hằng tháng]
# query by path: from parent to child
node = engine.query_by_path([
    {
        'node_type': 'phần',
        'node_id': 'hai'
    },
    {
        'node_type': 'chương',
        'node_id': 'I'
    },
    {
        'node_type': 'mục',
        'node_id': '1'
    },
    {
        'node_type': 'điều',
        'node_id': '50'
    }])
# => [Điều 50. Trợ cấp phục vụ]
node.content
# => Nội dung của điều luật
```

Hình 20: Sử dụng package `lawquery` để truy vấn dữ liệu

3.2. Xây dựng bộ dữ liệu hỏi đáp luật

Bộ câu hỏi được lấy từ website baohiemxahoi.gov.vn, cổng thông tin điện tử bảo hiểm xã hội Việt Nam. Bao gồm 19330 bộ câu hỏi-trả lời được phân vào nhiều lĩnh vực khác nhau, xem Hình 21.

Sau khi có được bộ dữ liệu hỏi đáp luật, tôi tiếp tục sử dụng Label Studio[25] để gán nhãn cho câu trả lời. Nhãn của câu trả lời là một danh sách các chỉ mục chứa nội dung liên quan tới câu trả lời. Có định dạng: `[id luật] > [chỉ mục level 0] > [chỉ mục level 1] > ... > [chỉ mục level n]`. Tôi chỉ lấy các data point mà câu trả lời có trích dẫn đến các văn bản luật, do đó số lượng thật sự của bộ dữ liệu là 4368 câu hỏi. Format của 1 câu hỏi-trả lời được thể hiện trong Hình 22



Hình 21: Số lượng câu hỏi theo lĩnh vực

Nội dung câu hỏi:
Sổ BHXH của tôi đã được chốt tại BHXH Ba Đình - Hà Nội. Hiện tại tôi bị mất 2 tờ rời của sổ , tôi đang sinh sống ở tỉnh Long An thì có thể ra cơ quan BHXH của tỉnh để xin cấp lại tờ rời BHXH hay không? hay phải ra cơ quan BHXH đã chốt sổ thì mới có thể xin cấp lại được? Xin cảm ơn

Câu trả lời:
Theo quy định tại **Tiết a Điểm 2.1** và **Tiết a Điểm 2.2 Khoản 2 Điều 3 Văn bản hợp nhất số 2089/VBHN-BHXH** thì:

- BHXH huyện được cấp lại sổ BHXH cho người đang bảo lưu thời gian đóng BHXH, BHTN, BHTNLĐ, BNN ở huyện, tỉnh khác.
- BHXH tỉnh được cấp lại sổ BHXH cho người đã hưởng BHXH hoặc đang bảo lưu thời gian đóng BHXH, BHTN, BHTNLĐ, BNN ở huyện, tỉnh khác.

Đồng thời, theo quy định tại **Tiết a Điểm 1.1 Khoản 1 Điều 27 Văn bản hợp nhất số 2089/VBHN-BHXH** ngày 26/6/2020 của BHXH Việt Nam ban hành Quy trình thu BHXH, BHYT, BHTN, BHTNLĐ, BNN; quản lý sổ BHXH, thẻ BHYT thì hồ sơ để cấp lại sổ BHXH gồm Tờ khai tham gia, điều chỉnh thông tin BHXH, BHYT (Mẫu TK1-TS). Vì vậy, nếu Bạn thuộc các trường hợp nêu trên thì có thể nộp hồ sơ xin cấp lại sổ BHXH tại cơ quan BHXH ở Long An nơi Bạn đang sinh sống.

Label:
2089/VBHNBHXH > Điều 3 > Điểm 2.2 > Tiết a
2089/VBHNBHXH > Điều 3 > Điểm 2.1 > Tiết a
2089/VBHNBHXH > Điều 27 > Khoản 1 > Điểm 1.1 > Tiết a

Hình 22: Ví dụ câu hỏi và câu trả lời

3.3. Truy xuất thông tin

Dựa vào dữ liệu đã thu thập được ở Phần 3.1 và Phần 3.2. Ta thiết lập một bài toán:

- Đầu vào: câu hỏi cần tra cứu
- Đầu ra: danh sách các nội dung liên quan có thể trả lời cho câu hỏi đó

Metric Evaluation: Để đánh giá kết quả, metric Top_K@acc được sử dụng. Độ chính xác được tính bằng tỷ lệ các nội dung đúng (nội dung dùng để trả lời cho câu hỏi) xuất hiện trong K kết quả trả về. Cụ thể công thức:

$$\text{Top}_K\text{@acc} = \frac{1}{n}\sum_{1}^{n}\begin{cases}1, l_q \subseteq L_K \\ 0, \text{otherwise}\end{cases}$$

Trong đó:

- L_K: là tập hợp chứa K nhãn có nội dung với độ tương đồng lớn nhất với câu hỏi q
- l_q: là tập hợp các nội dung đúng của câu hỏi q

Hướng tiếp cận thứ 1: Sử dụng các thuật toán cơ bản như TF-IDF và BM25, để tính toán độ tương đồng giữa câu hỏi và các nội dung trong dataset. Sau đó, sắp xếp các nội dung theo độ tương đồng giảm dần và trả về kết quả.

Để kết quả tốt hơn, tôi có sử dụng thêm một số kỹ thuật để chuẩn hóa nội dung như: loại bỏ các ký tự đặc biệt, dùng các công cụ của underthesea[26] để chỉnh dấu câu, phân tách từ…

```
from underthesea import text_normalize,word_tokenize
import re
import string
def
format_text(text,word_segmentation=False,remove_punctuation=False):
    text = re.sub(r'\s+', ' ', text)
    text = text.strip()
    text = text_normalize(text)
    if remove_punctuation:
        text = text.translate(str.maketrans('', '',
string.punctuation))
    if word_segmentation:
        text = word_tokenize(text, format="text")
    return text
```

Chương trình 1: Hàm format_text dùng để chuẩn hóa nội dung

Bảng 9 là kết quả của cách tiếp cận đầu tiên, sử dụng 2 thuật toán cơ bản là TF-IDF và BM25. Với 2 dạng chuẩn hóa: sử dụng word segmentation và không sử dụng word segmentation. Kết quả của phương pháp này chưa được tốt.

Name	$\mathbf{Top_5}$**@acc**	$\mathbf{Top_{10}}$**@acc**	$\mathbf{Top_{20}}$**@acc**	$\mathbf{Top_{50}}$**@acc**
TDIDF	0.1037	0.201	0.347	0.5289
BM25	0.079	0.1474	0.2556	0.4485
TDIDF_WS	0.1094	0.199	0.3344	0.5187
BM25_WS	0.0944	0.1746	0.2908	0.4709

Bảng 9: Kết quả cách tiếp cận thứ nhất

Hướng tiếp cận thứ 2: Sử dụng Sentence Transformers. Cụ thể ở đây là model InstructorEmbedding[27] được coi là state-of-the-art trong mảng này.

Model này sẽ nhận đầu vào là một string và trả về một vector 768 chiều. Các câu hỏi có nội dung tương đồng sẽ có vector tương tự nhau. Do đó, ta có thể tính toán độ tương đồng giữa câu hỏi và các nội dung trong dataset bằng cách tính *cosine similarity* giữa vector của câu hỏi và vector của các nội dung trong dataset.

Cosine similarity được tình dựa trên công thức sau:

$$\text{similarity}(q, d_i) = \frac{q * d_i}{\|q\|\|d_i\|}$$

trong đó q là vector của câu hỏi, d_i là vector của nội dung thứ i trong dataset.

Tuy nhiên model này không được train trên tập dataset có nhiều tiếng Việt cho nên ta cần fine-tune lại model này trên tập dataset về luật để có kết quả tốt nhất.

Theo tác giả của Instructor Embedding dữ liệu để fine-tune model có format là file JSON, gồm danh sách các ví dụ có format như trong Hình 23. Trong đó, `query` là câu hỏi, `pos` là nội dung có thể trả lời cho câu hỏi, `neg` là nội dung không thể trả lời cho câu hỏi, `task_name` là tên của dataset (có thể có nhiều dataset trong file JSON này).

Để tạo dataset cho việc fine-tune, chúng ta sẽ tận dụng dataset về hỏi đáp luật và **Hướng tiếp cận thứ 1** để tạo ra các ví dụ cho việc fine-tune. Cụ thể:

- Với mỗi hỏi đáp trong dataset, `query` sẽ là câu hỏi, `pos` sẽ là nội dung của các chỉ mục đã được gán nhãn ở Phần 3.2

- Để tạo `neg`, ta sẽ sử dụng thuật toán đã nói ở **Hướng tiếp cận thứ 1** để tìm ra top k nội dung. Sau đó kiểm tra xem nội dung nào chưa nằm trong `pos`, thì nội dung đó sẽ là `neg`.

Quá trình trên sẽ là **stage 1** của quá trình fine-tune. Sau khi fine-tune xong, ta sẽ tiến hành **stage 2**, tạo dataset tương tự ở stage 1 nhưng thay vì sử dụng các thuật toán cơ bản, ta sẽ sử dụng model đã được fine-tune để tạo `neg`. Xem thêm tại Hình 24.

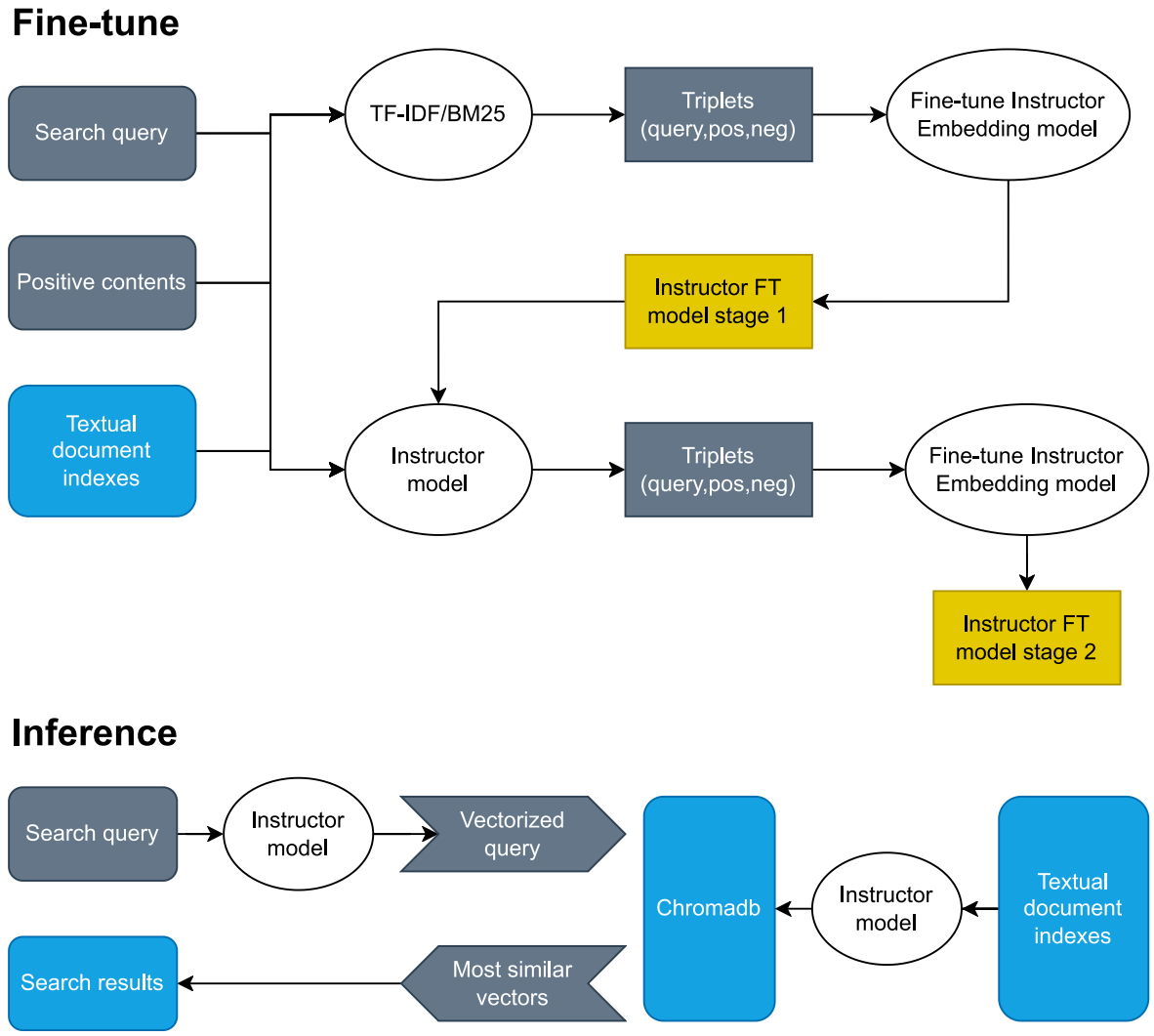

```
{
  "query": [
    "Represent the Wikipedia question for retrieving relevant
documents;",
    "big little lies season 2 how many episodes"
  ],
  "pos": [
    "Represent the Wikipedia document for retrieval;",
    "Big Little Lies (TV series) series garnered several accolades.
It received 16 Emmy Award nominations and won eight, including
Outstanding Limited Series and acting awards for Kidman, Skarsgård,
and Dern. The trio also won Golden Globe Awards in addition to a
Golden Globe Award for Best Miniseries or Television Film win for
the series. Kidman and Skarsgård also received Screen Actors Guild
Awards for their performances. Despite originally being billed as a
miniseries, HBO renewed the series for a second season. Production
on the second season began in March 2018 and is set to premiere in
2019. All seven episodes are being written by Kelley"
  ],
  "neg": [
    "Represent the Wikipedia document for retrieval;",
    "Little People, Big World final minutes of the season two-A
finale, Farm Overload. A crowd had gathered around Jacob, who was
lying on the ground near the trebuchet. The first two episodes of
season two-B focus on the accident, and how the local media reacted
to it. The first season of Little People, Big World generated solid
ratings for TLC (especially in the important 18-49 demographic),
leading to the show\"s renewal for a second season. Critical
reviews of the series have been generally positive, citing the
show\"s positive portrayal of little people. Conversely, other
reviews have claimed that the show has a voyeuristic bend"
  ],
  "task_name": "NQ"
}
```


Hình 23: Format của file JSON chứa dữ liệu fine-tune

Kết quả của các model được thể hiện ở Bảng 10. Model gốc của Instructor gồm có 3 model: `base`, `large`, `xl`. Vì giới hạn phần cứng nên tôi chỉ tiến hành finetune model nhỏ nhất là `base`. Tuy là model nhỏ nhưng kết quả sau khi finetune rất tốt.

Name	Top_5@acc	Top_{10}@acc	Top_{20}@acc	Top_{50}@acc
INSTRUCTOR-BASE	0.0119	0.0221	0.0416	0.0944
INSTRUCTOR-LARGE	0.0138	0.0247	0.0421	0.1023
INSTRUCTOR-XL	0.0188	0.0312	0.0537	0.1427
INSTRUCTOR-BASE FTS1	0.4832	0.5741	0.6621	0.7765
INSTRUCTOR-BASE FTS2	0.6431	0.7432	0.8123	0.8912

Bảng 10: Kết quả cách tiếp cận thứ hai



Hình 24: Sơ đồ tổng quan về phương pháp tiếp cận thứ hai

Như vậy, tôi đã giải quyết được bài toán truy xuất thông tin với các công cụ cơ bản như TF-IDF, BM25 và model Instructor Embedding. Có thể cải thiện kết quả bằng cách sử dụng các model có kích thước lớn hơn, hoặc sử dụng các mô hình khác như Dense Passage Retrieval[28], Haystack[29]…

4. Kết luận

Tóm lại, AI là một lĩnh vực rất rộng và mới nổi lên gần đây. Tuy tạo ra nhiều kết quả đáng kinh ngạc, nhưng AI vẫn còn nhiều hạn chế trên các bài toán mang tính chuyên sâu. Cơ hội để Việt Nam có thể tham gia vào cuộc đua này là rất lớn, nhưng cần có sự đầu tư nghiêm túc từ các cơ quan chức năng, các doanh nghiệp và các nhà nghiên cứu.

Qua luận văn này, tôi đã nghiên cứu và hiểu hơn về lĩnh vực luật và các công cụ AI mới nhất hiện nay. Và thấy được tính khả thi khi áp dụng các công cụ AI này vào luật. Vì thiếu điều kiện, kinh phí để có thể thí nghiệm thêm, tôi chỉ có thể đưa ra một số ý tưởng và đề xuất cho các nghiên cứu sau này. Gồm 4 ý chính:

- **Xây dựng các bộ dữ liệu liên quan tới luật** như đã làm trong Phần 3, tuy nhiên bộ dữ liệu mà tôi đã xây dựng chỉ mới là một phần nhỏ trong tất cả văn bản (chỉ bao gồm các văn bản liên quan tới bảo hiểm xã hội). Cần có một bộ dữ liệu đầy đủ hơn, bao gồm nhiều lĩnh vực pháp luật khác nhau, để có thể đưa ra được những kết quả chính xác nhất. Và còn thiếu liên kết giữa các điều luật với nhau, ví dụ: Điều 28 luật hôn nhân 2014 ***được hướng dẫn*** bởi điều Điều 7 Nghị định 126/2014/NĐ-CP.
- **Xây dựng mô hình ngôn ngữ lớn (LLM) cho tiếng Việt**. Các mô hình hiện nay chỉ hoạt động tốt trên tiếng Anh, hay các ngôn ngữ la-tin. Cần có một mô hình dành riêng cho tiếng Việt.
- **Dùng LLM để sinh câu trả lời cho các câu hỏi liên quan tới luật**. Trong bài luận này, tôi có thử nghiệm để tìm ra các nội dung liên quan đến câu hỏi. Tuy nhiên, để đưa ra được câu hỏi có ý nghĩa, chính xác, cần ứng dụng các mô hình LLM.
- **Các vấn đề liên quan tới đạo đức, pháp lý khi sử dụng AI**. Cần có 1 phương pháp để kiểm soát những nội dung sinh ra từ AI. Sao cho nó không vi phạm đạo đức, pháp luật. Và cần có một cơ chế để kiểm soát, giám sát các mô hình AI. Để tránh việc các mô hình này bị lợi dụng, hay bị sử dụng sai mục đích.

Tài liệu tham khảo

[8] Chiang, W.-L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., Gonzalez, J. E., Stoica, I., & Xing, E. P. (2023, March). *Vicuna: an open-source chatbot impressing gpt-4 with 90%* chatgpt quality*. https://lmsys.org/blog/2023-03-30-vicuna/

[1] Chiến, Đ. H. (2021). *Tự sự: từ ước mơ đến hiện thực king attorney – vua app luật sư 5.0*.

[19] Frantar, E., Ashkboos, S., Hoefler, T., & Alistarh, D. (2022). GPTQ: accurate post-training compression for generative pretrained transformers. *Arxiv preprint arxiv:2210.17323*.

[9] Greenwood, D., & Riehl, D. (2023, February 14). Legal Prompt Engineering - Examples and Tips. *Mit computational law report*.

[22] Hoang, L. H., Thanh, N. C., Thinh, N. P., Vinh, P. V., Nguyen, B. T., Huynh, A. T., & Nguyen, H. D. (2023). Intelligent retrieval system on legal information [Paper presentation]. In *Intelligent information and database systems*. Springer International Publishing.

[18] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2022). LoRA: low-rank adaptation of large language models [Paper presentation]. In *International conference on learning representations*. https://openreview.net/forum?id=nZeVKeeFYf9

[28] Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., & Yih, W.-t. (2020). *Dense passage retrieval for open-domain question answering*.

[5] Lambert, N., Castricato, L., von Werra, L., & Havrilla, A. (2022). Illustrating reinforcement learning from human feedback (rlhf). *Hugging face blog*.

[7] Lewis, P., Stenetorp, P., & Riedel, S. (2020). *Question and answer test-train overlap in open-domain question answering datasets*.

[12] Lumb, D. (2023). *Ai lawyer: it's starting as a stunt, but there's a real need*. CNET. https://www.cnet.com/tech/computing/an-ai-lawyer-will-challenge-speeding-tickets-in-court-next-month/

[14] MCGLAUFLIN, P. (2023). *Apple, goldman sachs, samsung, and 10 others clamp down on chatgpt at work*. Fortune. https://fortune.com/2023/05/19/chatgpt-banned-workplace-apple-goldman-risk-privacy/

[6] Mehdi, Y. (2023, May). *Reinventing search with a new ai-powered microsoft bing and edge, your copilot for the web*. https://blogs.microsoft.com/blog/2023/02/07/reinventing-search-with-a-new-ai-powered-microsoft-bing-and-edge-your-copilot-for-the-web/

[15] Mok, A. (2023). *Chatgpt could cost over $700,000 per day to operate. microsoft is reportedly trying to make it cheaper.* Business Insider. https://www.businessinsider.com/how-much-chatgpt-costs-openai-to-run-estimate-report-2023-4

[16] Morgan, T. P. (2022). *Counting the cost of training large language models*. The Next Platform. https://www.nextplatform.com/2022/12/01/counting-the-cost-of-training-large-language-models/

[24] Ngo, T. (n. d.). *LawQuery*. https://github.com/Th1nhNg0/law-query

[2] OpenAI. (2023). *Gpt-4 technical report*.

[29] Pietsch, M., Möller, T., Kostic, B., Risch, J., Pippi, M., Jobanputra, M., Zanzottera, S., Cerza, S., Blagojevic, V., Stadelmann, T., Soni, T., & Lee, S. (2019, November). *Haystack: the end-to-end NLP framework for pragmatic builders*. https://github.com/deepset-ai/haystack

[21] Prime, D. (2023). *Japan goes all in: copyright doesn't apply to ai training*. Technomancers.ai. https://technomancers.ai/japan-goes-all-in-copyright-doesnt-apply-to-ai-training/

[4] Reimers, N., & Gurevych, I. (2019). Sentence-bert: sentence embeddings using siamese bert-networks [Paper presentation]. In *Proceedings of the 2019 conference on empirical methods in natural language processing*. Association for Computational Linguistics. https://arxiv.org/abs/1908.10084

[20] Reuters. (2023). Top french university bans use of chatgpt to prevent plagiarism. *Reuters*. https://www.reuters.com/technology/top-french-university-bans-use-chatgpt-prevent-plagiarism-2023-01-27/

[23] Richardson, L. (2007). Beautiful soup documentation. *April*.

[3] Robertson, S., Walker, S., Hancock-Beaulieu, M., Gatford, M., & Payne, A. (1995). *Okapi at trec-4* [Paper presentation].

[11] Sparkes, M. (2023). *Ai legal assistant will help defendant fight a speeding case in court*. New Scientist. https://www.newscientist.com/article/2351893-ai-legal-assistant-will-help-defendant-fight-a-speeding-case-in-court/

[27] Su, H., Shi, W., Kasai, J., Wang, Y., Hu, Y., Ostendorf, M., Yih, W.-t., Smith, N. A., Zettlemoyer, L., & Yu, T. (2022). *One embedder, any task: instruction-finetuned text embeddings* [Paper presentation]. https://arxiv.org/abs/2212.09741

[13] Wang, B., & Komatsuzaki, A. (2021, May). *GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model*. (\url{https://github.com/kingoflolz/mesh-transformer-jax})

[17] Workshop, B., :, Scao, T. L., Fan, A., Akiki, C., Pavlick, E., Ilić, S., Hesslow, D., Castagné, R., Luccioni, A. S., Yvon, F., Gallé, M., Tow, J., Rush, A. M., Biderman, S., Webson, A., Ammanamanchi, P. S., Wang, T., Sagot, B., ... , Wolf, T. (2023). *Bloom: a 176b-parameter open-access multilingual language model*.

[25] *Open source data labeling platform*. (n. d.). https://labelstud.io/

[10] *Save time and money with donotpay!* (n. d.). https://donotpay.com/about/

[26] *Underthesea - Vietnamese NLP Toolkit*. (n. d.). https://github.com/undertheseanlp/underthesea