HỆ KHUYẾN NGHỊ Y TẾ BẰNG TIẾNG VIỆT DỰA TRÊN PHƯƠNG PHÁP RETRIEVAL AUGMENTED GENERATION

Ngô Huỳnh Trưởng, Nguyễn Hữu Minh Tâm, Nguyễn Trần Gia Thế, Huỳnh Văn Tín
Faculty of Information Science and Engineering, University of Information Technology,
Ho Chi Minh City, Vietnam
Vietnam National University, Ho Chi Minh City, Vietnam
{20522085, 20521871, 20521940}gm.uit.edu.vn
tinhv@uit.edu.vn

Abstract

Trong bài báo này, chúng tôi giới thiệu một bộ dữ liệu mới có tên là "Medical3T" và một hệ thống khuyến nghị thông minh "Medical Agent" được thiết kế để hỗ trợ trong quá trình chẩn đoán y tế và gợi ý thuốc phù hợp cho bệnh nhân. Trong hệ thống này, dữ liệu được embedd bằng embedding của OpenAI[1] và được thêm vào collection của Milvus[2] thông qua LangChain[3]. Với câu truy vấn mà người dùng nhập vào sẽ được embedding bởi model embedding của OpenAI. Tiếp theo similarity search được thực hiện với câu truy vấn đã được embedd (sử dụng độ đo cos_sim) - retrieval. Cuối cùng instructions, context và query ban đầu được đưa ra cho mô hình trò chuyện OpenAI. Sử dụng ngôn ngữ tự nhiên bằng tiếng Việt, hệ thống này có khả năng dự đoán bệnh dựa trên triệu chứng của bệnh nhân và đề xuất thuốc điều trị phù hợp. Công nghệ học máy và xử lý ngôn ngữ tự nhiên (NLP) được tích hợp để phân tích và xử lý dữ liệu bệnh lý, cung cấp các gợi ý chính xác và cá nhân hóa. Hệ thống này là một bước tiến quan trọng trong việc ứng dụng trí tuệ nhân tạo trong lĩnh vực y tế, đặc biệt là trong việc cung cấp dịch vụ y tế chất lượng cao và tiếp cận được cho người dùng tiếng Việt. Chúng tôi tiến hành sử dụng hệ khuyến nghị này trên tập test, mỗi testcase sẽ đưa ra 5 khuyến nghị về bệnh hay thuốc dựa trên triệu chứng. Kết quả mà nhóm đạt được ở top@1, top@3 và top@5 accuracy lần lượt là 0.58, 0.74 và 0.8.

1 Giới thiệu

Trong những năm gần đây, sự tiến bộ của trí tuệ nhân tạo, các mô hình học máy, học sâu đã tạo điều kiện cho ngành y tế phát triển. Nhiều hệ thống, công nghệ được tạo ra và áp dụng giúp công việc của các y bác sĩ đỡ khó khăn hơn, nâng cao chất lượng và sự chính xác của việc điều trị. Trong bài báo này, nhóm chúng tôi giới thiệu một bộ dữ liệu mới là Medical3T, và Medical Agent, một hệ khuyến nghị sử dụng ngôn ngữ tiếng Việt để phục vụ cho ngành y tế.

Hệ thống này nhằm mục đích cung cấp các khuyến nghị về bệnh dựa trên triệu chứng của bệnh nhân, và đề xuất thuốc dựa trên bệnh, hướng tới việc cải thiện phương pháp chăm sóc sức khỏe theo hướng cá nhân hóa và chủ động hơn. Hệ thống này giúp khả năng đưa ra chẩn đoán của bác sĩ về bệnh được nhanh chóng hơn, giúp việc điều trị được diễn ra kịp thời. Đồng thời, khả năng xác định thuốc dựa trên bệnh cũng được phát triển, giúp tìm ra loại thuốc phù hợp với bệnh nhân thông qua một cơ sở dữ liệu về bệnh, thuốc và các triệu chứng.

Hệ khuyến nghị thuốc này không chỉ dựng lại ở việc cải thiện các quy trình chẩn đoán và điều trị. Một phần quan trọng khác của hệ thống chính là sự tối ưu hóa trải nghiệm của bệnh nhân. Chúng tôi tiến hành xây dựng một giao diện để bệnh nhân hay bác sĩ đều có thể truy cập và tra cứu bệnh hoặc thuốc một cách nhanh chóng. Bằng việc sử dụng ngôn ngữ tiếng Việt, hệ thống này không chỉ phục vụ cho ngành y tế Việt Nam mà còn góp phần giảm bớt rào cản ngôn ngữ, mang lại cơ hội tiếp cận chăm sóc y tế chất lượng cao cho một số lượng lớn bệnh nhân.

2 Các công trình liên quan

2.1 Dữ liệu

Trong lĩnh vực nghiên cứu y học và dược học, vai trò của các bộ dữ liệu đầy đủ đã trở nên quan trọng. Các bộ dữ liệu như ViMQ cho tiếng Việt và các bộ dữ liệu tiếng nước ngoài như NCBI, BC4CHEMD, Drug Combination Extraction Dataset đã trở thành điểm tập trung quan trọng trong cộng đồng nghiên cứu y tế.

ViMQ (Huy et al., 2021): Bộ dữ liệu tiếng Việt về câu hỏi y tế từ bệnh nhân, gồm 9,000 câu thu thập từ tư vấn trực tuyến tại www.vinmec.com. Bộ

[1] https://openai.com/
[2] https://milvus.io/
[3] https://github.com/langchain-ai/langchain

dữ liệu này cung cấp chú thích cho Phân loại Mục đích và Nhận dạng Thực thể, hỗ trợ nghiên cứu xử lý ngôn ngữ tự nhiên trong lĩnh vực y học tiếng Việt.

NCBI (Doğan et al., 2014): Bộ dữ liệu về bệnh lý bao gồm 793 tóm tắt PubMed, được chia thành tập huấn luyện (593), phát triển (100) và kiểm thử (100). Bộ dữ liệu này được chú thích với các đề cập về bệnh lý, sử dụng các định danh khái niệm từ MeSH hoặc OMIM.

BC4CHEMD (Krallinger et al., 2015): là một bộ dữ liệu gồm 10,000 tóm tắt PubMed[4] chứa tổng cộng 84,355 đề cập về thực thể hóa học được đánh dấu thủ công bởi các chuyên gia hóa học.

Drug Combination Extraction Dataset (Tiktinsky et al., 2022): Bộ dữ liệu này bao gồm 1,634 tóm tắt y học, được chuyên gia chú thích với mục đích trích xuất thông tin về hiệu quả của sự kết hợp các loại thuốc từ văn bản khoa học. Bộ dữ liệu này bao gồm 1,634 tóm tắt y học, được chuyên gia chú thích với mục đích trích xuất thông tin về hiệu quả của sự kết hợp các loại thuốc từ văn bản khoa học.

2.2 Phương pháp

Towards personalized healthcare-an intelligent medication recommendation system (Suryadevara, 2020) , nghiên cứu này khám phá về việc thiết kế, phát triển và đánh giá hệ thống đề xuất thông minh về đề xuất thuốc. Hệ thống này tận dụng sức mạnh của trí tuệ nhân tạo và học máy để cung cấp gợi ý điều trị cá nhân hóa, từ đó cải thiện kết quả điều trị và sức khỏe của bệnh nhân. Nó sử dụng các thuật toán phức tạp để phân tích hồ sơ bệnh nhân, lịch sử y tế và các hướng dẫn lâm sàng liên quan để đưa ra các đề xuất về thuốc.

A unified drug–target interaction prediction framework based on knowledge graph and recommendation system (Ye et al., 2021), KGE_NFM là một framework thống nhất cho dự đoán drug-target interactions (DTI), kết hợp knowledge graph (KG) và hệ thống đề xuất. Framework này sử dụng biểu diễn chiều thấp cho các thực thể trong KG và tích hợp thông tin đa dạng thông qua neural factorization machine (NFM) (He and Chua, 2017). Trong ba kịch bản thực tế, KGE_NFM đạt được dự đoán chính xác và mạnh mẽ trên bốn bộ dữ liệu kiểm thử. Kết quả của nghiên cứu này cho thấy KGE_NFM mang lại cái nhìn quan trọng để tích hợp KG và hệ thống đề xuất, giúp khám phá DTI mới.

Drug Recommendation System based on Sentiment Analysis of Drug Reviews using Machine Learning(Garg, 2021). Trong nghiên cứu này, tác giả xây dựng một hệ thống đề xuất thuốc sử dụng các đánh giá của bệnh nhân để dự đoán bằng cách sử dụng các quy trình vectơ hóa khác nhau như Bow, TF-IDF, Word2Vec và Manual Feature Analysis, có thể giúp đề xuất loại thuốc hàng đầu cho một số căn bệnh nhất định theo các phân loại thuật toán khác nhau.

ChatGPT and large language model (LLM) chatbots: The current state of acceptability and a proposal for guidelines on utilization in academic medicine(Kim et al., 2023). Hiện tại chưa có sự đồng thuận rõ ràng về các tiêu chuẩn sử dụng các mô hình ngôn ngữ lớn như ChatGPT trong y học hàn lâm. Trong bài báo này tác giả đã thực hiện đánh giá phạm vi các tài liệu hiện có để hiểu hiện trạng sử dụng LLM trong y học và đưa ra hướng dẫn sử dụng trong tương lai trong giới học thuật.

3 Dữ liệu

3.1 Thu Thập và tiền xử lý dữ liệu

Dữ liệu cho nghiên cứu này được chúng tôi thu thập trang web Vinmec[5] là hệ thống ý tế hàn lâm do Vingroup đầu tư và phát triển và là một nguồn thông tin y tế hàng đầu tại Việt Nam. Trang web này cung cấp thông tin chi tiết và đáng tin cậy về các loại thuốc, các bệnh lý và cách sử dụng thuốc. Điều này tạo ra một cơ sở dữ liệu đa dạng với thông tin chi tiết về sức khỏe và y tế.

Dữ liệu được thu thập thông qua quá trình tự động hóa sử dụng thư viện Python Requests[6] để tải trang web và Beautifulsoup[7] để phân tích cú pháp HTML, từ đó trích xuất thông tin cần thiết. Mục tiêu là tạo ra một tập dữ liệu phong phú và đa chiều về các bệnh lý cũng như các loại thuốc, giúp nghiên cứu sâu rộng về thuốc và các yếu tố liên quan và có thể sử dụng dữ liệu cho các hướng phát triển khác nhau.

Sau khi thu thập dữ liệu thô, nhóm chúng tôi tiến hành tiền xử lý dữ liệu qua các bước sau nhằm phục vụ cho việc phân tích dữ liệu và huấn luyện mô hình: Chuyển tất cả về từ thường; Xóa dấu câu và ký tự đặc biệt; Xóa đi các dữ liệu bị khuyết (sử dụng dropna).

Dữ liệu sau khi được xử lý bao gồm 2 file có định dạng json, đó là file drug.json chứa thông tin

[4] https://pubmed.ncbi.nlm.nih.gov/

[5] https://www.vinmec.com/vi/

[6] https://requests.readthedocs.io/en/latest/

[7] https://www.crummy.com/software/BeautifulSoup/

về các loại thuốc và illness.json chứa thông tin về các bệnh lý, với kích thước lần lượt là : (191 , 12) và (693, 11).

Quá trình thu thập và tiền xử lý được thực hiện qua các giai đoạn chính như hình 1

3.2 Thu Thập Thông Tin về Các Loại Thuốc (Drug)

Trong giai đoạn này, quá trình thu thập thông tin về các loại thuốc được thực hiện một cách tỷ mỉ để đảm bảo sự đầy đủ và chi tiết. Chúng tôi thu thập được 191 mẫu thuốc và 12 thuộc tính bao gồm: id, đường dẫn, tên thuốc, dạng bào chế, nhóm thuốc, chỉ định, chống chỉ định, thận trọng, tác dụng không mong muốn, liều và cách dùng, chú ý khi sử dụng, tài liệu tham khảo.

Dữ liệu đa dạng về tên thuốc, với chỉ định, tác dụng phụ và hướng dẫn sử dụng khác nhau cho từng thuốc. Với thuộc tính nhóm thuốc, số lượng thuốc đau chống viêm steroid nsaid là nhiều nhất với 8 thuốc, thuốc chống virus đứng thứ 2 với 4 thuốc, các nhóm thuốc còn lại có khoảng 1 đến 3 thuốc cho mỗi nhóm hình 2.

Phân bố dữ liệu về mô tả thuốc cho thấy một đặc điểm rõ ràng, đó là đa số tên các loại thuốc được mô tả một cách ngắn gọn, với độ dài chủ yếu tập trung từ 2 đến 10 từ. Điều này làm thấy rõ việc đặt tên loại thuốc thường được thực hiện một cách súc tích, dễ hiểu, dễ nhớ, mỗi từ ngắn gọn được tích hợp để truyền đạt thông tin quan trọng về loại thuốc một cách hiệu quả. Sự tập trung vào mô tả ngắn gọn không chỉ thể hiện sự dễ dàng trong việc truyền đạt thông tin, mà còn giúp tăng cường sự dễ hiểu và tiện lợi khi người đọc cần tìm hiểu về các sản phẩm y tế, cụ thể ở đây là các loại thuốc. Sự súc tích ở tên các loại thuốc là chìa khóa để các bác sĩ và bệnh nhân có thể tra cứu thông tin về các loại thuốc, để có thể hiểu rõ hơn và sử dụng cho từng bệnh, từng triệu chứng phù hợp (hình 3).

Phân bố số lượng từ trong chỉ định của các loại thuốc rõ ràng cho thấy xu hướng tập trung số lượng từ trong khoảng từ 10 đến 30 từ. Với chỉ định của các loại thuốc, các giá trị này không chỉ phản ánh mức độ trung bình đến cao về chi tiết mô tả của các loại thuốc, mà còn làm nổi bật sự rõ ràng và dễ hiểu trong việc mô tả về cách sử dụng loại thuốc. Các chỉ định không quá dài để có thể khiến bệnh nhân hay bác sĩ gặp các vấn đề khó khắn khi đọc hiểu. Độ dài vừa phải của mỗi chỉ định giúp tạo ra thông tin chính xác và đầy đủ, giúp bệnh nhân hoặc bác sĩ dễ dàng để có thể hiểu rõ và thực hiện các hướng dẫn liên quan đến việc sử dụng và xác định các loại thuốc một cách hiệu quả (hình 4).

3.3 Thu Thập Thông Tin về Các Bệnh Lý (Illness)

Giai đoạn này của quá trình nghiên cứu tập trung vào việc thu thập thông tin chi tiết về các bệnh lý, nhằm hiểu rõ về các khía cạnh y tế, cơ chế và cách điều trị hiện đại. Chúng tôi thu thập được 693 mẫu bệnh và 11 thuộc tính bao gồm: id, đường dẫn, tên bệnh, tổng quan, nguyên nhân, triệu chứng, đường lây truyền bệnh, đối tượng nguy cơ, phòng ngừa, các biện pháp chẩn đoán, các biện pháp điều trị, .

Các loại bệnh khá đa dạng, với các triệu chứng, nguyên nhân, biện pháp khác nhau, tùy thuộc vào từng bệnh.

Cũng giống như ở bộ dữ liệu về thuốc ở trên, tên của các loại bệnh cũng được mô tả một cách ngắn gọn, súc tích, đa số chỉ từ 2 đến 5 từ (hình 5).

Tuy nhiên, với mô tả của từng loại bệnh, số lượng từ là khá nhiều, lên đến hàng trăm từ, thậm chí còn có bệnh có mô tả lên tới 900 từ. Điều này cho ta thấy mức độ chi tiết của việc mô tả loại bệnh, có thể giúp người đọc hiểu rõ được định nghĩa của từng bệnh. Bộ dữ liệu mà nhóm thu thập đưa ra thông tin về các loại bệnh, và với mỗi loại bệnh đó, các định nghĩa được giải thích khá chi tiết và rõ ràng (hình 6).

Phần lớn mô tả về triệu chứng trong bộ dữ liệu tập trung chủ yếu trong khoảng từ 100 đến 300 từ, tạo ra một kết quả có tính chi tiết ở mức trung bình đến cao. Mỗi bệnh có triệu chứng được trình bày một cách chi tiết và rõ ràng, nhấn mạnh vào các triệu chứng quan trọng để bệnh nhân hay bác sĩ có thể hiểu rõ về bản chất và cách nhận diện các bệnh lý.

Ngoài ra, đáng chú ý là một số bệnh có mô tả triệu chứng chi tiết và phức tạp hơn, có trường hợp lên đến hơn 2000 từ. Điều này có thể cho thấy sự quan trọng của việc hiểu rõ các triệu chứng của các bệnh lý phức tạp. Các triệu chứng cần được trình bày rõ ràng và đầy đủ, để bệnh nhân và bác sĩ có thể xác định đúng được loại bệnh cũng như thuốc để điều trị. Sự đa dạng trong độ dài mô tả triệu chứng làm tôn lên sự linh hoạt trong việc trình bày thông tin y tế, đảm bảo rằng cả người chuyên nghiệp y tế, như các bác sĩ và bệnh nhân đều có khả năng tiếp cận thông tin phù hợp với nhu cầu và mức độ chi tiết mong muốn (hình 7).

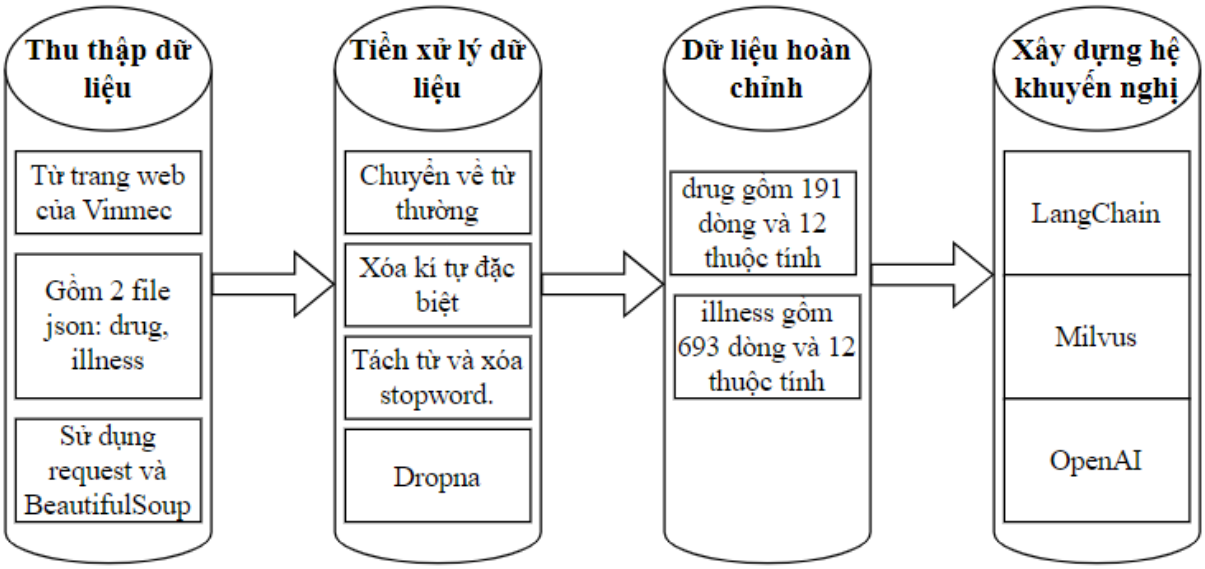

Figure 1: Quy trình thu thập và tiền xử lý.

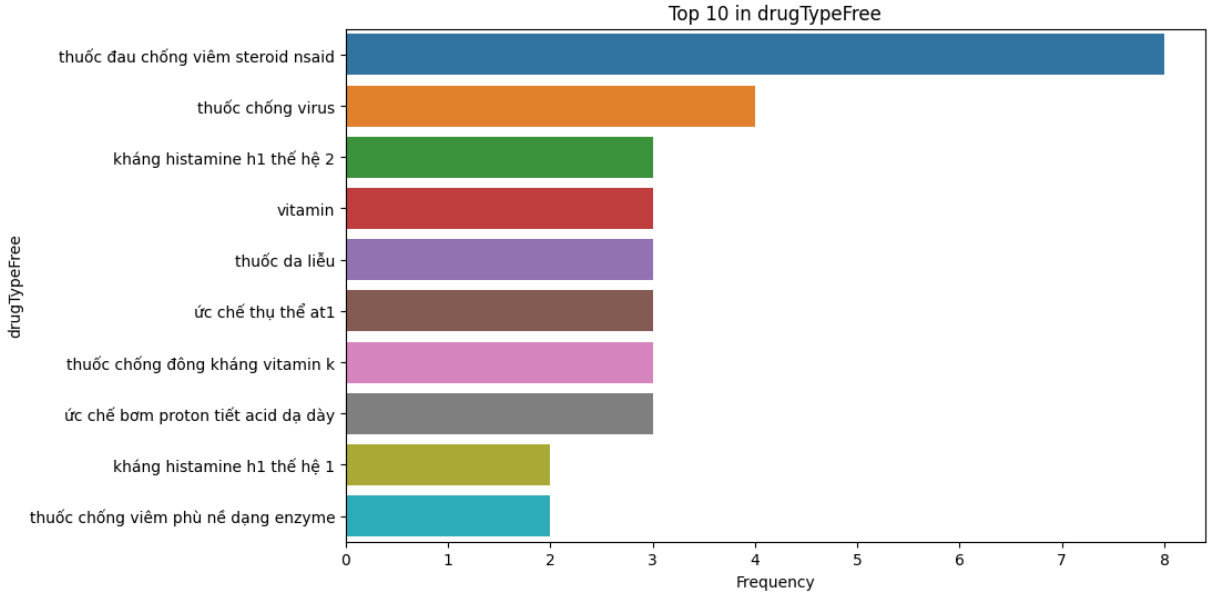

Figure 2: Top 10 nhóm thuốc phổ biến nhất.

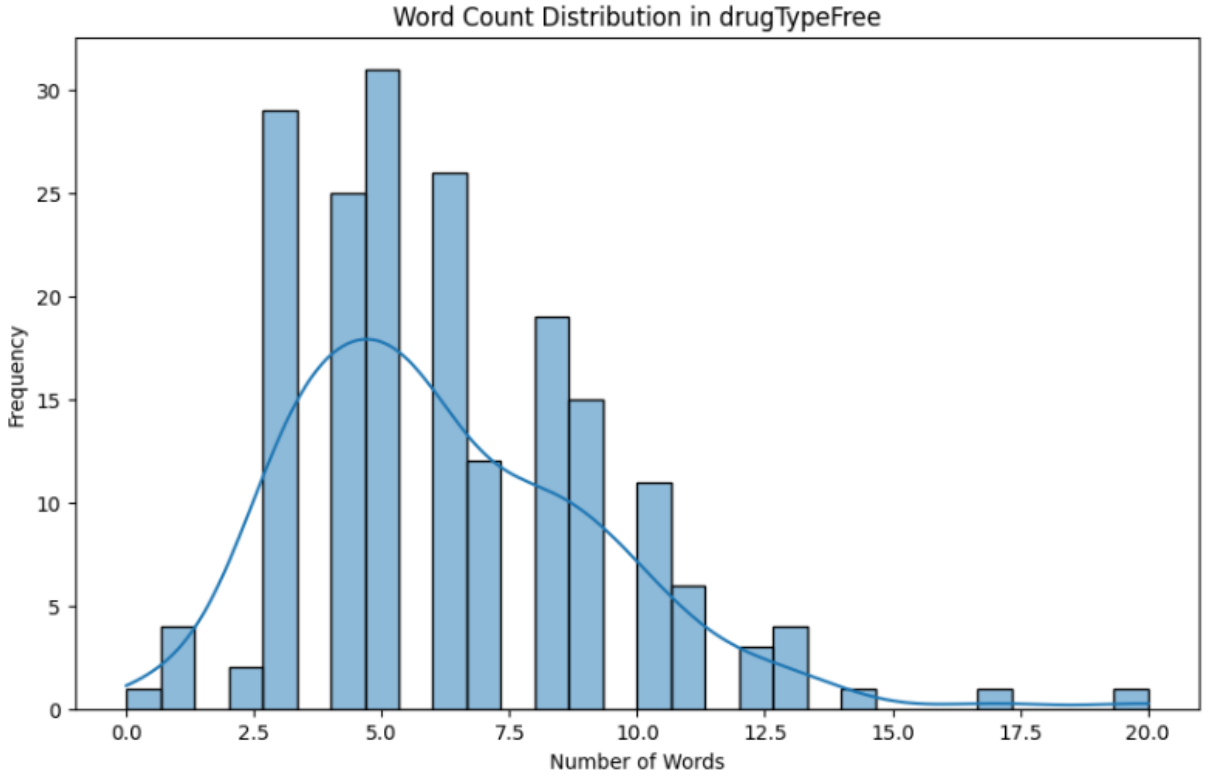

Figure 3: Số lượng từ của tên các loại thuốc.

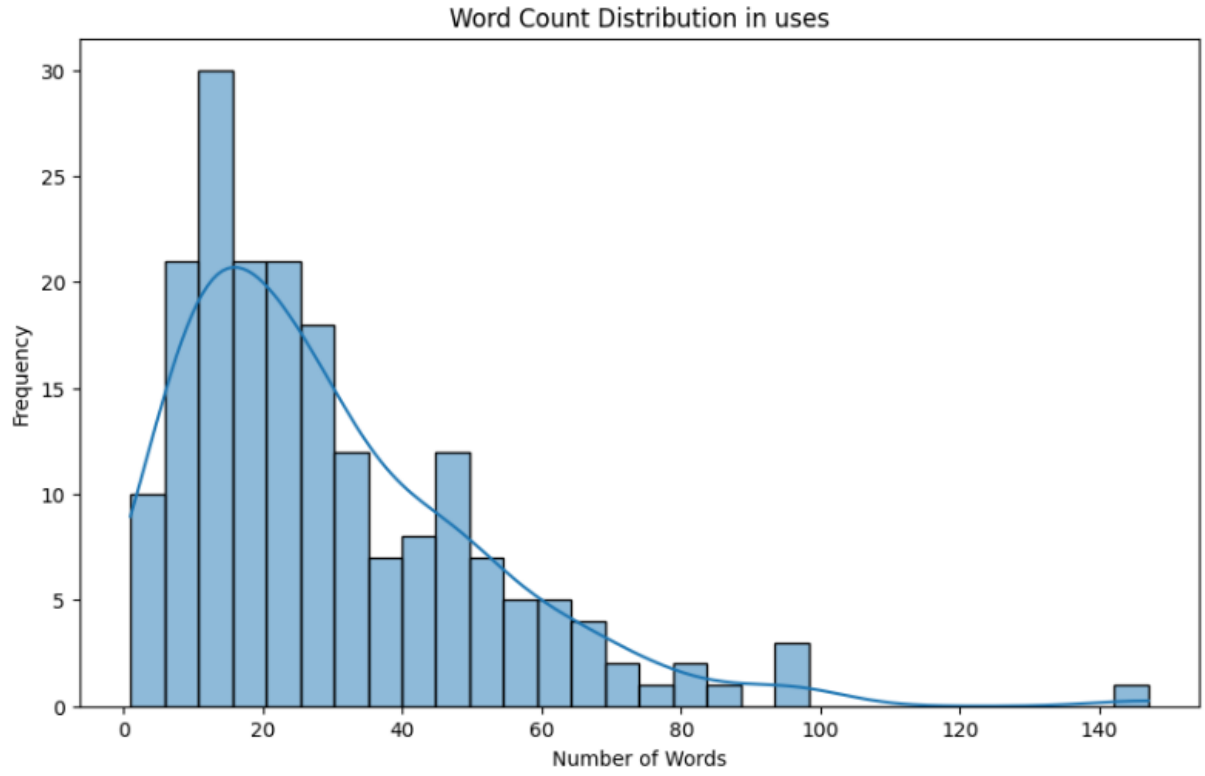

Figure 4: Số lượng từ của các chỉ định.

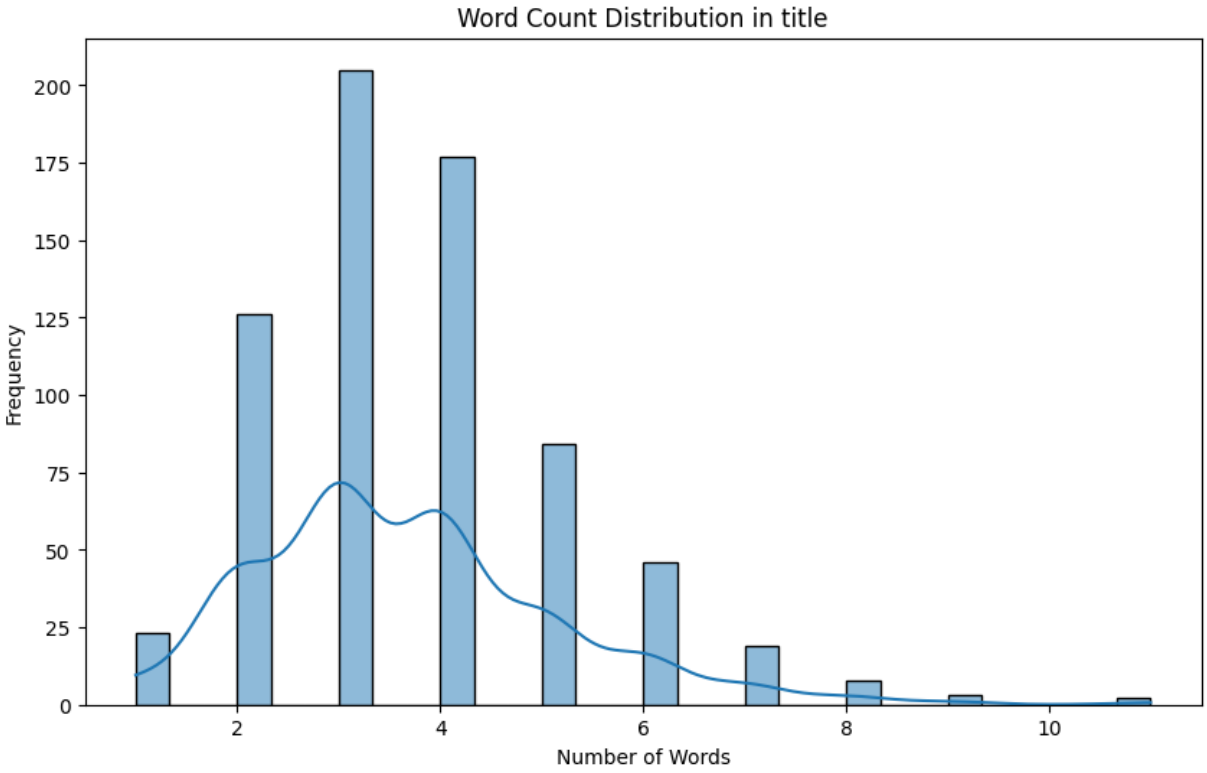

Figure 5: Số lượng từ của tên bệnh.

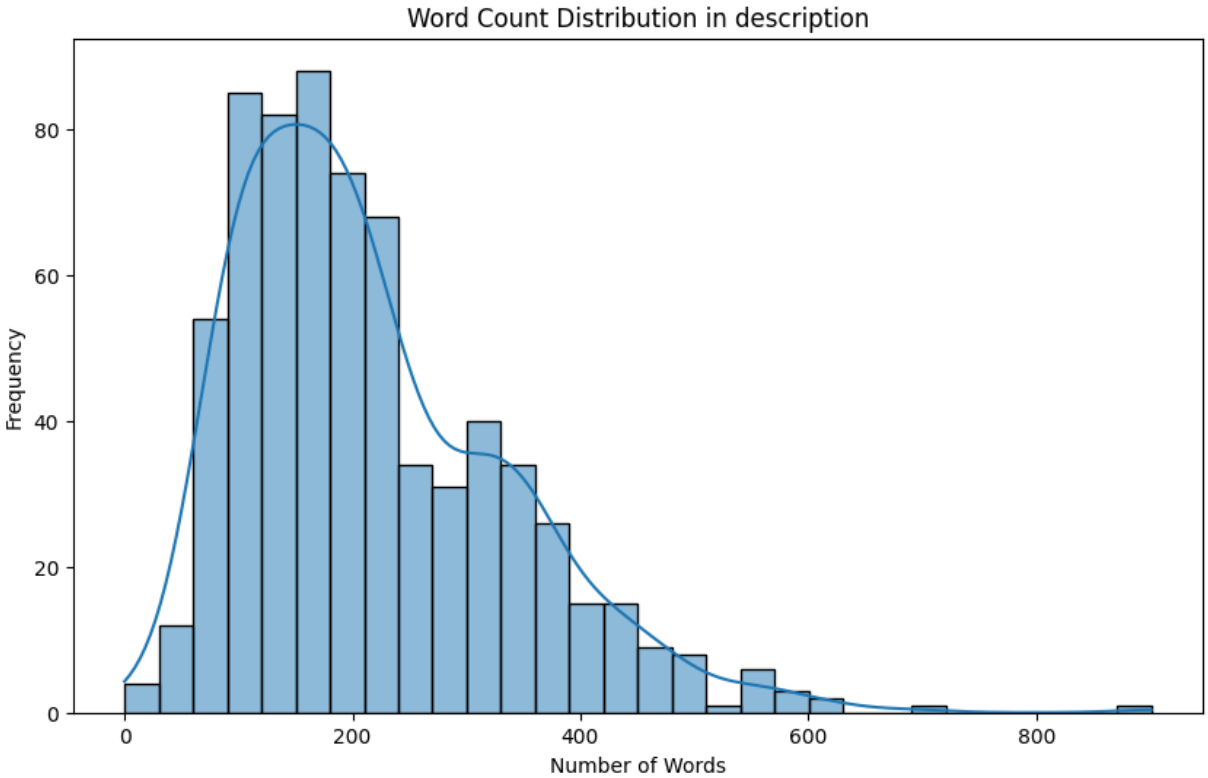

Figure 6: Số lượng từ của mô tả các bệnh.

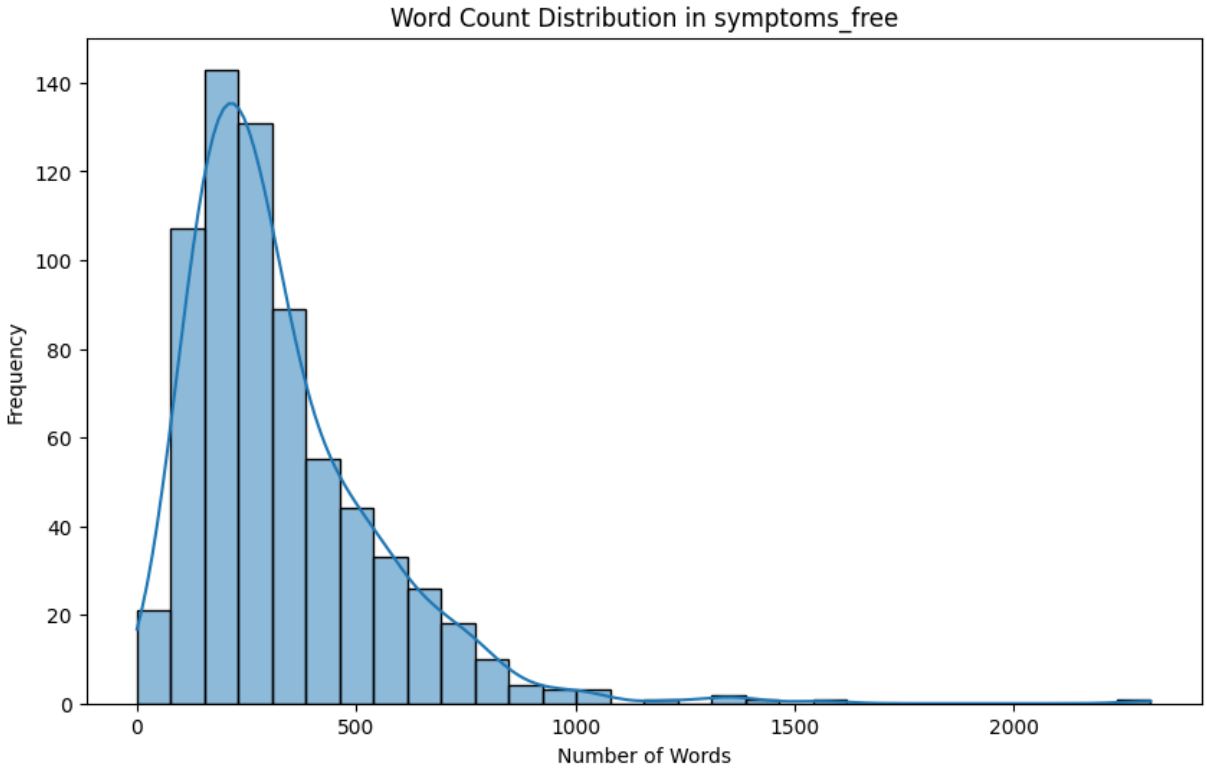

Figure 7: Số lượng từ của triệu chứng bệnh.

4 Thí nghiệm

4.1 LangChain

LangChain[8], xuất hiện vào tháng 10 năm 2022, là một framework mã nguồn mở cho phép các nhà phát triển làm việc với trí tuệ nhân tạo kết hợp các mô hình ngôn ngữ lớn với các nguồn tính toán và dữ liệu bên ngoài.

LangChain là cầu nối để giữa các mô hình ngôn ngữ lớn với các ứng dụng, các hệ cơ sở dữ liệu từ bên thứ ba, ta có thể nói LangChain có khả năng nhận thức được về dữ liệu cũng như các tác nhân. LangChain bao gồm các thành phần như sau:

- LLM Wrappers giúp kết nối và sử dụng các mô hình ngôn ngữ lớn.
- Tiếp theo là Prompt Templates, đây là nơi ta có thể thiết lập các prompts, chính là các input của mô hình ngôn ngữ lớn.
- Thành phần thứ ba là Indexes, dùng để trích xuất thông tin từ các mô hình ngôn ngữ lớn.
- Cuối cùng là Memory, đây là nơi để lưu trữ và lấy các thông tin cần thiết. Có 2 loại memory, đó là short-term memory và long-term memory. Short-term memory liên quan đến một câu trả lời riêng biệt nào đó, còn long-term memory thể hiện mối quan hệ giữa các câu trả lời khác nhau.

LangChain Chains cho phép kết hợp nhiều thành phần ở trên lại với nhau để giải quyết một bài toán cụ thể, và xây dựng một ứng dụng của mô hình ngôn ngữ lớn hoàn chỉnh. Với Sequential Chains, ta có thể thực hiện một loạt các kết nối đến với các mô hình ngôn ngữ lớn. Kết quả đầu ra của một Chain có thể dùng để làm đầu vào cho một Chain khác. Sequential Chains bao gồm hai loại:

- SimpleSequentialChain
- General form of sequential chains

Trong đó, SimpleSequentialChain đại diện cho một chuỗi các Chains, trong đó mỗi Chain riêng lẻ có một đầu vào và một đầu ra duy nhất, và đầu ra đó được sử dụng làm đầu vào cho Chain tiếp theo.

LangChain Agents tạo điều kiện cho sự tương tác giữa các mô hình ngôn ngữ lớn và API. Chúng đóng vai trò quan trọng trong việc đưa ra các quyết định, xác định những hành động mà mô hình ngôn ngữ lớn nên thực hiện. Quá trình này bao gồm việc

[8] https://github.com/langchain-ai/langchain

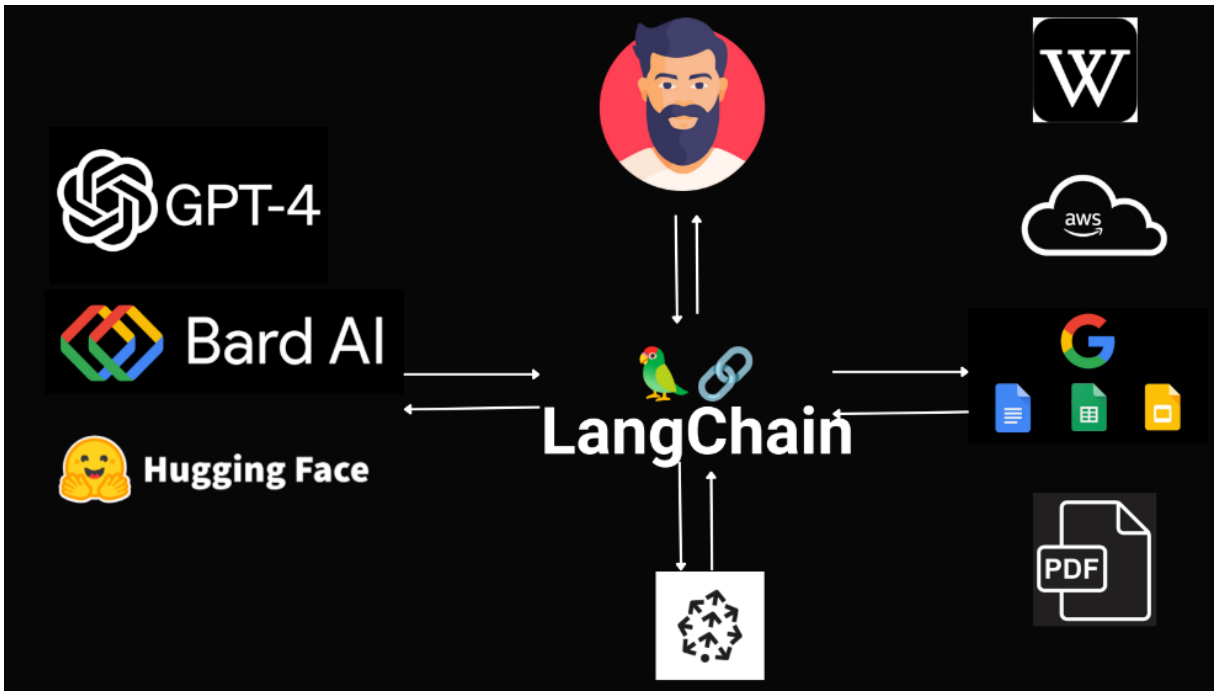

Figure 8: LangChain

thực hiện một hành động, quan sát các kết quả và sau đó lặp lại chu trình cho đến khi hoàn thành.

Chính vì những đặc điểm nổi bật ở trên, LangChain hiện nay đang được sử dụng rộng rãi, và có thể ứng dụng để xây dựng các hệ thống như Chat Bots, hệ thống Question-Answering, các công cụ tóm tắt, hay trong bài nghiên cứu này, chính là hệ thống khuyến nghị thuốc, một ứng dụng để giúp đỡ các bác sĩ cũng như bệnh nhân.

4.2 Milvus

Milvus là một công cụ mạnh mẽ để tìm kiếm sự tương đồng trong tập dữ liệu vectơ dày đặc chứa hàng triệu hoặc thậm chí hàng tỷ vectơ. Nó sử dụng kiến trúc phân tán tách biệt lưu trữ và điện toán, cho phép khả năng mở rộng theo chiều ngang trong các nút điện toán.

Cấu trúc Milvus được biểu diễn ở hình 9. Hệ thống bao gồm 4 lớp : access layer, coordinator service, worker nodes và storage layer.

- Access layer : bao gồm một nhóm các proxy không trạng thái và đóng vai trò là front layer của hệ thống mà người dùng tương tác. Được thiết kế để đơn giản hóa giao tiếp, Access Layer giúp người dùng truy cập các dịch vụ của hệ thống một cách thuận tiện.

- Coordinator service : đóng vai trò là bộ não của hệ thống, phân công nhiệm vụ cho các worker nodes. Chúng đóng vai trò quan trọng trong việc xử lý và tính toán các thao tác trên dữ liệu, đảm bảo sự hiệu quả của hệ thống.

- Worker nodes : có nhiệm vụ làm theo hướng dẫn từ dịch vụ điều phối và thực thi các lệnh DML/DDL do người dùng kích hoạt. Chúng đóng vai trò quan trọng trong việc xử lý và tính toán các thao tác trên dữ liệu, đảm bảo sự hiệu quả của hệ thống.

- Storage layer : là xương sống của hệ thống và chịu trách nhiệm về sự bền vững của dữ liệu. Bao gồm meta storage, log broker và object storage.

4.3 OpenAI

OpenAI là một mô hình ngôn ngữ lớn rất phổ biến hiện nay. Tuy nhiên, OpenAI còn khá nhiều vấn để liên quan đến việc bảo mật. Vì vậy mà một dịch vụ mới đã ra đời, với khả năng bảo mật cao hơn, đó là Azure OpenAI[10].

Azure OpenAI là một dịch vụ trong hệ sinh thái Azure được phát triển bởi Microsoft. Azure OpenAI Service có khả năng cung cấp các REST API để truy cập đến mô hình ngôn ngữ lớn của OpenAI. Hiện nay, khá nhiều mô hình khả dụng trên Azure OpenAI Service:

- GPT - 3.5.

[9] https://milvus.io/docs/architecture_overview.md

[10] https://azure.microsoft.com/en-us/products/ai-services/openai-service

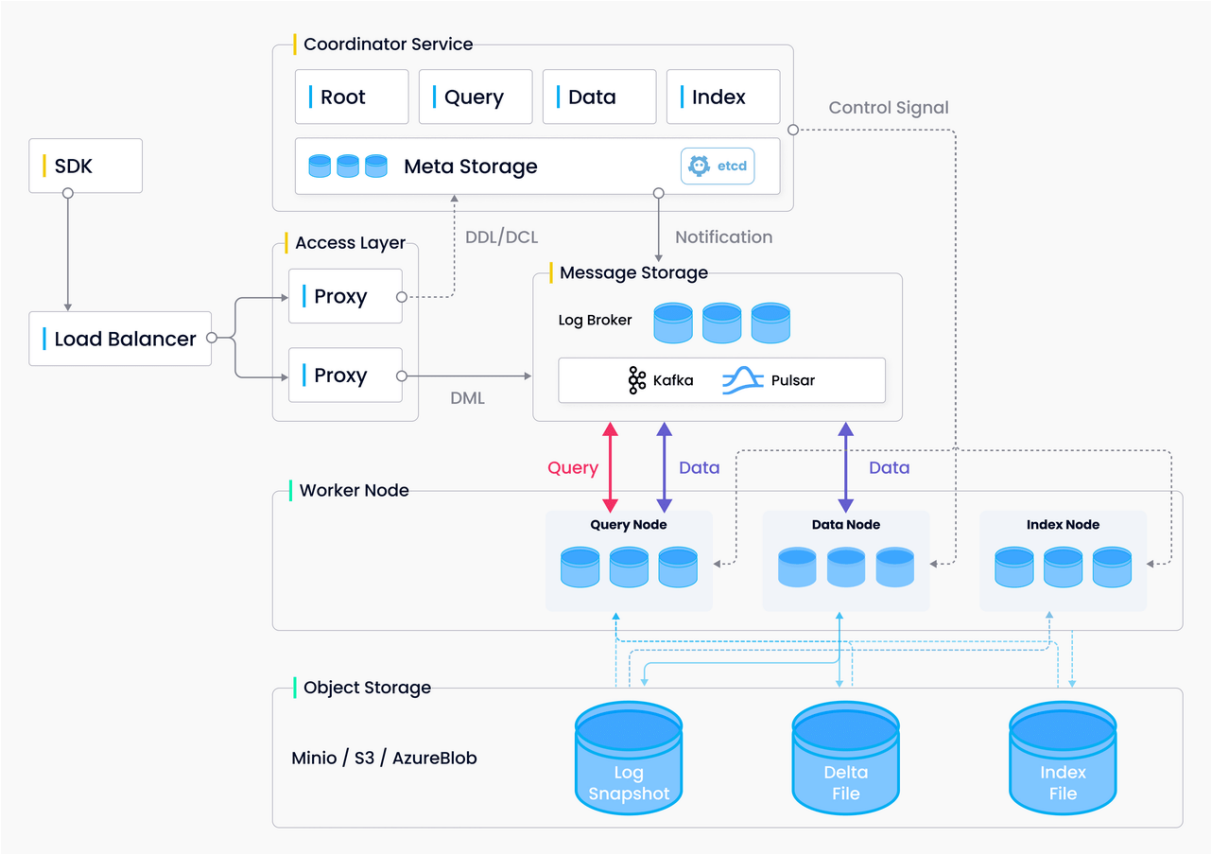

Figure 9: Kiến trúc Milvus[9]

- GPT - 4.
- Codex.
- Embedding.

Trong nghiên cứu này, nhóm tập trung vào embedding có trong Azure OpenAI Service. Embedding nghĩa là đưa dữ liệu về một vector để có thể huấn luyện các mô hình học máy một cách hiệu quả. Với các dữ liệu có độ tương quan với nhau càng cao thì độ tương đồng giữa hai vector được embedding là càng lớn.

4.4 Quy trình xây dựng

Quy trình xây dựng hệ khuyến nghị của chúng tôi được mô tả trong hình 10. Gồm các phần như sau :

4.4.1 Vector database

Đây là nơi lưu trữ các vector biểu diễn cho các dữ liệu dùng để đưa ra khuyến nghị dành cho người dùng. Dữ liệu được embedded bằng embedding của OpenAI, một công cụ có khả năng biểu diễn dữ liệu dưới dạng các vector số. Các vector này có thể đo lường được mức độ tương đồng giữa các dữ liệu khác nhau. Dữ liệu được nhúng sau đó được thêm vào collection của Milvus, một nền tảng quản lý vector có khả năng lưu trữ, truy vấn và phân tích các vector một cách hiệu quả và nhanh chóng. Quá trình trên được thực hiện thông qua LangChain, một nền tảng kết nối các nguồn dữ liệu và các mô hình ngôn ngữ lớn một cách linh hoạt.

Các câu truy vấn được nhập vào cũng sẽ được embedded, lúc này trong vector database chứa cả các vector embedding của câu truy vấn được nhập vào và vector embedding của dữ liệu được thu thập từ VinMEC đã đề cập ở trên. Similarity search (tìm kiếm tương tự) được thực hiện với câu truy vấn đã embedded để truy xuất các dữ liệu có liên quan nhất với câu hỏi của người dùng. Các dữ liệu được truy xuất có thể chứa các thông tin cần thiết cho việc đưa ra khuyến nghị.

4.4.2 Al Model

Chúng tôi đã triển khai OpenAI GPT-3.5 kết hợp với mô hình prompt để phát triển hệ thống khuyến nghị bệnh dựa trên triệu chứng.

Để tận dụng hiệu suất của GPT-3.5, chúng tôi đã thiết kế một promt thông minh và hiệu quả. Bằng cách này, chúng tôi có thể chỉ đạo mô hình tìm kiếm trong vector database triệu chứng. Sau đó,

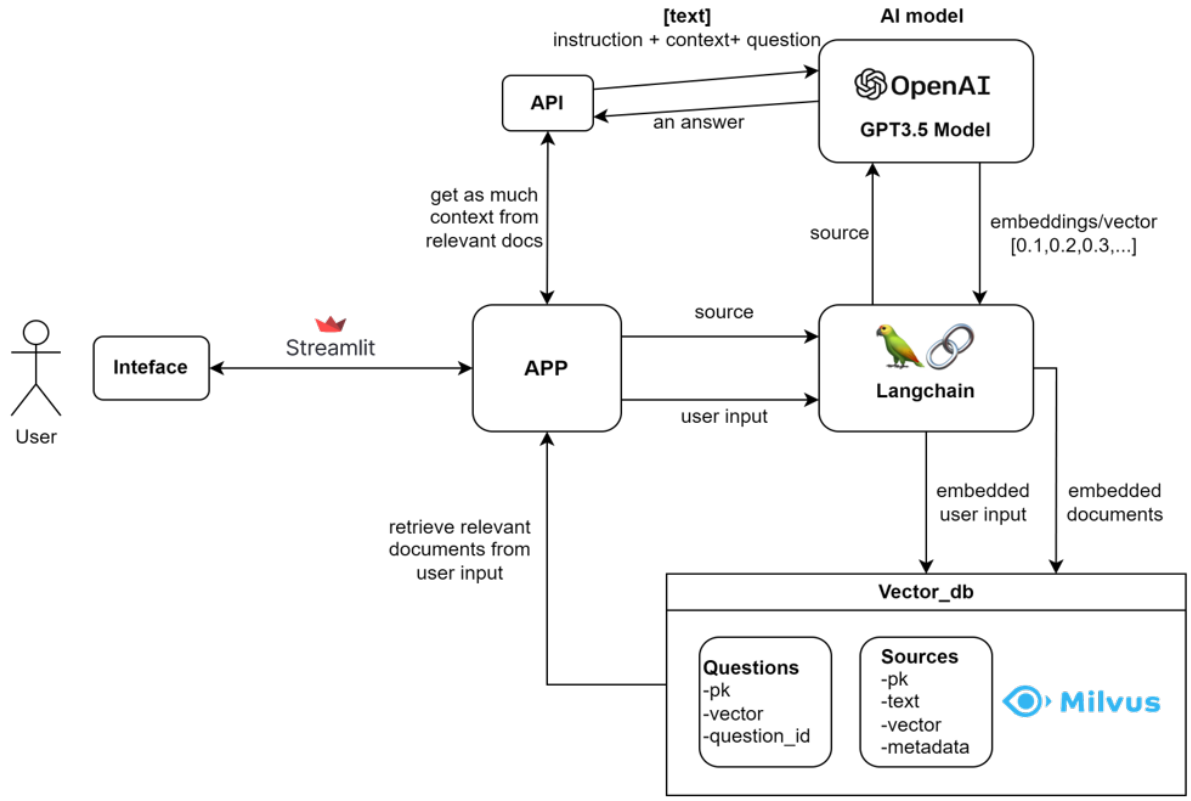

Figure 10: Quy trình xây dựng hệ khuyến nghị

đưa ra và biến đổi chúng thành thông tin hữu ích để đưa ra các khuyến nghị về bệnh lý. Dưới đây là 2 promt chúng tôi đã sử dụng :

"PROMT" của dự đoán bệnh

DEFAULT_PROMPT = "You are helpful AI assistance. User question: query"

DIAGNOSE_PROMPT =
"""Bác sĩ sẽ cung cấp một số thông tin về các bệnh liên quan và triệu chứng của một bệnh nhân.
Yêu cầu của bạn là đưa ra chẩn đoán bệnh có tỉ lệ mắc phải cao nhất, dựa trên các thông tin về các bệnh được cung cấp và triệu chứng của bệnh nhân. Giải thích chẩn đoán đó và ghi chú thông tin tham khảo từ nguồn tài liệu nào.
Kết luận cần tuân thủ theo định dạng json dưới đây
{"diagnose": "string", "info": "explain": "string [source_number]", "source": ["string"]}}
"""

Ví dụ:
Tài liệu bệnh
Bệnh A: Thông tin bệnh A
Bệnh B: Thông tin bệnh B
Triệu chứng: bệnh nhân có triệu chứng C
Chẩn đoán:
{"diagnose": "Bệnh A", "info": "explain": "Do những biểu hiện C được ghi trong tài liệu A [1]", "source": ["Tài liệu a"]}}

DIAGNOSE_TEMPLATE =
"""
Tài liệu bệnh : {disease_info}
Triệu chứng: {symptom}
Chẩn đoán:
"""

"PROMT" của dự đoán thuốc

SUGGEST_MEDICINE_PROMPT =
"""Bác sĩ sẽ cung cấp tên một số thông tin về bệnh cần điều trị và tài liệu về bệnh có liên quan.
Yêu cầu của bạn là kết luận 01 loại thuốc phù hợp để điều trị bệnh nêu kể trên, dựa trên các thông tin về bệnh và thuốc được cung cấp. Đồng thời giải thích lý do bạn đưa ra kết luận này.
Kết luận cần tuân thủ theo định dạng json dưới đây
{"suggestion": "string", "explain": "string"}

Ví dụ:
Thông tin bệnh
Bệnh A: Cần điều trị bằng thuốc B
Bệnh cần xem xét: Bệnh A
Gợi ý:

{"suggestion": "thuốc B", "explain": "Thuốc B phù hợp để điều trị bệnh A vì ..."}
"""

SUGGEST_MEDICINE_TEMPLATE =
"""
Thông tin bệnh : {disease_info}
Bệnh cần xem xét: {disease}
Gợi ý:
"""

4.4.3 API

API là giao diện liên kết giữa các thành phần của hệ thống. Trong nghiên cứu này, nhóm sử dụng Azure OpenAI API, có nhiệm vụ lấy nội dung từ các tài liệu, hay thông tin nhập vào từ người dùng. Sau đó gửi các instruction, context và question dạng text đến cho AI Model. Cuối cùng, sau khi AI Model thực hiện các công việc của mình, một khuyến nghị sẽ được đưa ngược lại cho API để gửi đến app streamlit, hiển thị trên giao diện người dùng.

4.4.4 APP

Cuối cùng, chúng tôi thiết kế giao diện người dùng với Streamlit[11], giúp người dùng dễ dạng tương tác và sử dụng. Hệ khuyến nghị mà nhóm đề xuất sẽ đưa ra top 5 thông tin phù hợp nhất, trong nghiên cứu này là thuốc hoặc bệnh được chẩn đoán dựa trên triệu chứng. Kết quả được hiển thị một cách rõ ràng, giúp cho bác sĩ hoặc bệnh nhân có thể dễ dàng tra cứu được các thông tin cần thiết.

5 Kết quả và phân tích lỗi

5.1 Kết quả

Top1@Acc	Top3@Acc	Top5@Acc
0.58	0.74	0.8

Table 1: Kết quả

Trong bài báo này, việc lựa chọn giữa Top 1 Accuracy và Top N Accuracy là quan trọng để đảm bảo sự hiệu quả và linh hoạt trong quá trình đề xuất thuốc. Top 1 Accuracy, với tính chính xác tuyệt đối, đem lại lợi ích trong việc đề xuất một loại thuốc duy nhất, tối ưu hóa sự chắc chắn trong quyết định và giúp người dùng dễ dàng hiểu và áp dụng. Ngược lại, Top N Accuracy, với tính đa dạng và linh hoạt, mở rộng phạm vi lựa chọn thuốc cũng như đưa ra dự các bệnh khác nhau , và cung cấp sự linh hoạt cho người dùng trong quá trình quyết định. Vì vậy, chúng tôi đã sử dụng cả 2 thang đo, trong đó sử dụng Top N Accuracy với N là 3 và 5.

Để thực hiện đánh giá hiệu suất của hệ thống, chúng tôi đã xây dựng tập kiểm thử với 50 mẫu bao gồm các bệnh và triệu chứng của các bệnh đó. Kết quả đạt được trình bày trong bảng 1 với top1@acc là 0.58. Và kết quả khá cao với top3@acc là 0.74 và top5@acc là 0.8.

5.2 Phân tích lỗi

Mặc dù hệ thống của chúng tôi có hiệu suất khá tốt với top 3 và top 5 accuracy, nhưng việc top 1 accuracy thấp có thể phản ánh một số vấn đề cụ thể trong quá trình dự đoán :

- Đa ý nghĩa của triệu chứng : một số triệu chứng có thể xuất hiện trong nhiều bệnh khác nhau, làm tăng độ khó trong việc dự đoán chính xác bệnh cụ thể. Ví dụ đầu tiên trong bảng 2 đã chỉ rõ vấn đề này, các bệnh trong top 3 đều có những triệu chứng như đổ mồ hôi, cảm thấy sợ hãi... Vì vậy , các triệu chứng chung chưa đủ để loại trừ các bệnh tương tự, dẫn đến sự nhầm lẫn trong việc chọn ra bệnh chính xác.

- Sự thiếu hiểu biết về mối quan hệ nội dung: mô hình có thể chưa đủ hiểu biết về mối quan hệ nội dung giữa các triệu chứng và bệnh, đặc biệt là khi có sự tương đồng giữa các triệu chứng của các bệnh.

- Khả năng dự doán thấp khi có triệu chứng : hệ thống có thể có xu hướng dự đoán chính xác hơn khi có nhiều triệu chứng xuất hiện, như trong trường hợp top 3 và top 5, nhưng gặp khó khăn khi chỉ có 1 hay 2 triệu chứng. Ví dụ thứ hai trong bảng 2, chỉ có 2 triệu chứng thì dự đoán đúng chỉ nằm trong top 5 là "bạch tạng".

- Dữ liệu chưa nhiều và chưa đa dạng.

Việc nâng cao top 1 accuracy là một mục tiêu để cải thiện trong tương lai của chúng tôi.

6 Kết luận và hướng phát triển

Trong bài báo cáo này, chúng tôi đã xây dựng được bộ dữ liệu đa dạng và phong phú, chứa thông tin chi tiết về bệnh và thuốc. Cùng với đó, chúng tôi đã trình bày chi tiết về việc xây dựng một hệ thống hệ khuyến nghị tập trung vào bệnh và thuốc dựa trên triệu chứng. Sự tích hợp của Milvus database,

[11] https://docs.streamlit.io/

Table 2: Các mẫu được khuyến nghị

Triệu chứng/Bệnh	Top 1	Top 3	Top 5
Đổ mồ hôi, Không thể hít thở sâu, Cảm thấy sợ hãi cực độ, Đau nhức đầu hay toàn thân	viêm não nhật bản	viên não nhật bản rối loạn hoảng sợ ám ảnh sợ hãi	viên não nhật bản rối loạn hoảng sợ ám ảnh sợ hãi hoảng sợ khi ngủ tằng thông khí
da dễ bị rám nắng, thị lực kém	rách giác mạc	rách giác mạc tắc ống dẫn tinh hội chứng mệt mỏi	rách giác mạc tắc ống dẫn tinh hội chứng mệt mỏi bạch tạng nám da

LangChain và OpenAI API đã tạo ra một cấu trúc mạnh mẽ cho hệ thống, từ việc quản lý cơ sở dữ liệu đến xử lý ngôn ngữ tự nhiên và học máy. Với độ chính xác 50% trên top 1 accuracy, 74% trên top 3 accuracy và 80% trên top 5 accuracy, thể hiện khả năng dự đoán ấn tượng của hệ thống.

Để nâng cao hệ thống, chúng tôi đề xuất mở rộng nguồn dữ liệu để đảm bảo tính đa dạng và chi tiết. Đồng thời, tối ưu hóa mô hình và tích hợp thông tin từ các nguồn phác đồ điều trị có thể cung cấp những khuyến nghị cá nhân hóa và hữu ích hơn cho người dùng.

Đối với giao diện người dùng, việc tối ưu hóa và thêm tính năng tương tác sẽ làm cho trải nghiệm người dùng trở nên thuận tiện hơn. Đồng thời, tích hợp tính năng giải thích sẽ giúp người dùng hiểu rõ hơn về cơ sở lý do của mỗi khuyến nghị, tăng cường sự tin cậy và tương tác.

Chúng tôi tin rằng sự phát triển theo những hướng này sẽ định hình hệ thống hệ khuyến nghị của chúng tôi thành một công cụ mạnh mẽ và linh hoạt, hỗ trợ người dùng một cách toàn diện trong quá trình quản lý sức khỏe và tìm kiếm thông tin y tế.

References

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. Ncbi disease corpus: a resource for disease name recognition and concept normalization. *Journal of biomedical informatics*, 47:1–10.

Satvik Garg. 2021. Drug recommendation system based on sentiment analysis of drug reviews using machine learning. In *2021 11th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, pages 175–181.

Xiangnan He and Tat-Seng Chua. 2017. Neural factorization machines for sparse predictive analytics. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 355–364.

Ta Duc Huy, Nguyen Anh Tu, Tran Hoang Vu, Nguyen Phuc Minh, Nguyen Phan, Trung H Bui, and Steven QH Truong. 2021. Vimq: A vietnamese medical question dataset for healthcare dialogue system development. In *Neural Information Processing: 28th International Conference, ICONIP 2021, Sanur, Bali, Indonesia, December 8–12, 2021, Proceedings, Part VI 28*, pages 657–664. Springer.

Jin K Kim, Michael Chua, Mandy Rickard, and Armando Lorenzo. 2023. Chatgpt and large language model (llm) chatbots: the current state of acceptability and a proposal for guidelines on utilization in academic medicine. *Journal of Pediatric Urology*.

Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M Lowe, et al. 2015. The chemdner corpus of chemicals and drugs and its annotation principles. *Journal of cheminformatics*, 7(1):1–17.

Chaitanya Krishna Suryadevara. 2020. Towards personalized healthcare-an intelligent medication recommendation system. *IEJRD-International Multidisciplinary Journal*, 5(9):16.

Aryeh Tiktinsky, Vijay Viswanathan, Danna Niezni, Dana Meron Azagury, Yosi Shamay, Hillel Taub-Tabib, Tom Hope, and Yoav Goldberg. 2022. A dataset for n-ary relation extraction of drug combinations. *arXiv preprint arXiv:2205.02289*.

Qing Ye, Chang-Yu Hsieh, Ziyi Yang, Yu Kang, Jiming Chen, Dongsheng Cao, Shibo He, and Tingjun Hou. 2021. A unified drug–target interaction prediction framework based on knowledge graph and recommendation system. *Nature communications*, 12(1):6775.