

Trường đại học công nghệ  
Khoa công nghệ thông tin



Lê Thái Sơn

**XÂY DỰNG HỆ THỐNG HỖ TRỢ TƯ VẤN SINH  
VIÊN THỰC HIỆN KHÓA LUẬN TỐT NGHIỆP**

**KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**  
**Ngành: Công nghệ thông tin**

Hà nội - 2024

Trường đại học công nghệ  
Khoa công nghệ thông tin

Lê Thái Sơn

**XÂY DỰNG HỆ THỐNG HỖ TRỢ TƯ VẤN SINH  
VIÊN THỰC HIỆN KHÓA LUẬN TỐT NGHIỆP**

**KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**  
Ngành: Công nghệ thông tin

Cán bộ hướng dẫn: TS.Đặng Trần Bình

Cán bộ đồng hướng dẫn:

Hà nội - 2024

## **Lời cảm ơn**

Trước tiên, em xin bày tỏ lòng biết ơn chân thành và sâu sắc nhất tới Thầy giáo, Tiến sĩ Đặng Trần Bình đã tận tình hướng dẫn, động viên, giúp đỡ em trong suốt quá trình thực hiện đề tài.

Em xin gửi lời cảm ơn sâu sắc tới quý Thầy Cô trong Khoa Công nghệ thông tin đã truyền đạt kiến thức quý báu cho em trong những năm học vừa qua. Em xin gửi lời cảm ơn các anh chị tiền bối đã nhiệt tình chỉ bảo trong quá trình em làm khoá luận. Con xin nói lên lòng biết ơn đối với Ông Bà, Cha Mẹ luôn chăm sóc, động viên trên mỗi bước đường học vấn của con.

Xin chân thành cảm ơn các Anh Chị và Bạn bè, đặc biệt là các thành viên trong lớp K65CB đã ủng hộ, giúp đỡ và động viên tôi trong suốt thời gian học tập bốn năm trên giảng đường đại học và thực hiện đề tài.

Mặc dù đã cố gắng hoàn thành luận văn trong phạm vi và khả năng cho phép nhưng chắc chắn sẽ không tránh khỏi những thiếu sót. Em kính mong nhận được sự cảm thông và tận tình chỉ bảo của quý Thầy Cô và các Bạn.

Em xin chân thành cảm ơn!.

**Lê Thái Sơn**

## LỜI CAM ĐOAN

Tôi xin cam đoan các kết quả trình bày trong luận án là công trình nghiên cứu của tôi dưới sự hướng dẫn của cán bộ hướng dẫn. Các số liệu, các kết quả trình bày trong luận án hoàn toàn trung thực và chưa được công bố trong các công trình trước đây. Các dữ liệu tham khảo được trích dẫn đầy đủ.

*Hà Nội, ngày tháng năm 2024*

**Lê Thái Sơn**

## Tóm tắt

**Tóm tắt:** Khóa luận là môn học cuối cùng ở đại học, có số tín chỉ cao nhất và đồng thời thể hiện kiến thức của sinh viên trong suốt 4 năm học. Tuy nhiên, không ít sinh viên còn gặp phải những rắc rối trong việc thực hiện khóa luận của mình từ việc chọn chủ đề, chọn thầy cô, chọn đề tài... Việc hỗ trợ sinh viên trên chặng đường cuối cùng và khó khăn nhất này là một vấn đề cần thiết để các bạn có thể hoàn thành tốt khóa luận của mình. Trong những năm gần đây, nhiều mô hình ngôn ngữ lớn đã ra đời và đạt được những thành tựu ấn tượng. Những mô hình này ngày càng được tin dùng và triển khai nhiều hơn trong các ứng dụng. Năm bắt tinh thần này, khóa luận hướng đến việc sử dụng chúng trong việc xây dựng một hệ thống hỗ trợ sinh viên trong quá trình thực hiện khóa luận.

**Keywords:** *Transformer, LLM, Similarity search, RAG*

# Mục lục

1. Mở đầu .....	12
1.1. Đặt vấn đề .....	12
1.2. Nội dung hệ thống .....	12
1.3. Đóng góp của bản thân .....	12
1.4. Cấu trúc khóa luận .....	13
2. Kiến thức nền tảng .....	14
2.1. Token .....	14
2.2. Kiến trúc Transformer .....	14
2.2.1. Encoder và Decoder .....	15
2.2.2. Cơ chế Attention .....	15
2.2.3. Tầng FeedForward .....	17
2.2.4. Tầng Embedding .....	18
2.2.5. Tầng mã hóa vị trí (Positional Encoding) .....	18
2.2.6. So sánh .....	18
2.3. Vector nhúng .....	19
2.3.1. Biểu diễn từ .....	19
2.3.1.1. Dựa trên thống kê .....	20
2.3.1.1.1. TF-IDF .....	20
2.3.1.1.2. BM25 .....	20
2.3.1.2. Dựa trên ngữ nghĩa .....	21
2.3.2. Biểu diễn văn bản .....	21
2.3.2.1. Dựa trên thống kê .....	21
2.3.2.2. Dựa trên ngữ nghĩa .....	21
2.3.3. Truy xuất .....	22
2.4. LLM .....	22
2.4.1. Base LLM và Instruction LLM .....	23
2.4.2. Giới hạn của các mô hình ngôn ngữ lớn .....	23
2.4.3. Prompting .....	24
3. Hệ thống .....	25
3.1. Phân tích thành phần .....	25
3.1.1. RAG .....	25
3.1.1.1. RAG Framework .....	25
3.1.1.2. Dữ liệu .....	27
3.1.1.3. Phân đoạn .....	29
3.1.1.4. Nhúng .....	29
3.1.1.4.1. Dựa trên thống kê .....	29
3.1.1.4.2. Dựa trên ngữ nghĩa .....	29
3.1.1.5. Đánh chỉ mục .....	31
3.1.1.6. Mô hình sinh .....	33
3.1.1.6.1. GPT .....	33
3.1.1.6.2. Lấy mẫu .....	33
3.1.1.7. Tiền xử lý .....	35
3.1.1.8. Truy xuất .....	35
3.1.1.9. Hậu xử lý .....	35
3.1.1.9.1. Lọc .....	35

3.1.9.2. Rerank .....	35
3.1.9.3. Long context reorder .....	36
3.1.10. Lịch sử trò chuyện .....	36
3.1.11. Prompt .....	37
3.1.12. Tìm kiếm mờ .....	37
3.1.12.1. Khoảng cách chỉnh sửa .....	37
3.1.12.2. Các phép toán mở rộng .....	38
3.1.13. Bộ công cụ .....	40
3.1.14. Đánh giá .....	42
3.1.14.1. Nguyên tắc cơ bản: .....	42
3.1.14.2. Độ đo .....	42
3.1.14.3. Yêu cầu về khả năng .....	42
3.2. Biểu đồ .....	43
3.2.1. Biểu đồ ca sử dụng .....	43
3.2.2. Biểu đồ tuần tự .....	43
3.2.2.1. Hệ thống .....	43
3.2.2.2. Công cụ truy xuất thông tin cá nhân .....	44
3.2.2.3. Công cụ truy xuất thông tin thực hiện khóa luận .....	44
3.2.2.4. Công cụ tìm kiếm thông tin giáo viên .....	44
3.2.2.5. Công cụ tìm kiếm bài báo .....	45
3.2.2.6. Công cụ tìm kiếm khóa luận UET .....	45
3.2.2.7. Công cụ tìm kiếm khóa luận khác .....	46
3.2.2.8. Công cụ tìm kiếm công trình cụ thể .....	47
3.2.2.9. Công cụ tìm kiếm thầy hướng dẫn .....	48
3.2.2.10. Công cụ gợi ý chủ đề khóa luận tốt nghiệp .....	48
3.2.2.11. Thu hồi ngũ cảnh .....	49
3.2.2.12. Hậu xử lý .....	49
4. Thực nghiệm .....	50
4.1. Nền tảng .....	50
4.2. Hệ thống .....	50
4.2.1. Dữ liệu .....	50
4.2.2. Phân đoạn .....	51
4.2.3. Cơ sở dữ liệu vector .....	51
4.2.4. Nhúng .....	51
4.2.5. LLM .....	51
4.2.6. Truy xuất .....	51
4.2.7. Hậu truy xuất .....	51
4.2.8. Lịch sử trò chuyện .....	52
4.2.9. Bản mẫu .....	52
4.3. Kiểm thử .....	53
4.3.1. Tìm kiếm thông tin cá nhân .....	53
4.3.2. Tìm kiếm thông tin giáo viên .....	53
4.3.3. Hỏi về thực hiện khóa luận .....	54
4.3.4. Tìm kiếm khóa luận theo chủ đề .....	54
4.3.5. Tìm kiếm khóa luận theo giáo viên hướng dẫn .....	55
4.3.6. Tìm kiếm khóa luận theo chủ đề của giáo viên hướng dẫn .....	56
4.3.7. Tìm kiếm bài báo theo chủ đề .....	57

4.3.8. Tìm kiếm bài báo theo tác giả .....	58
4.3.9. Tìm kiếm bài báo theo chủ đề của tác giả .....	59
4.3.10. Tìm kiếm công trình cụ thể .....	60
4.3.11. Gợi ý giáo viên hướng dẫn .....	60
4.3.12. Gợi ý đề tài khóa luận .....	61
4.3.13. Đánh giá câu trả lời .....	62
4.4. Đánh giá hệ thống .....	62
4.4.1. Đánh giá hiệu quả truy xuất .....	62
4.4.1.1. Xây dựng bộ dữ liệu .....	62
4.4.1.2. Thực hiện đánh giá .....	63
4.4.1.2.1. Đánh giá mô hình nhúng .....	63
4.4.1.2.2. Đánh giá mô hình Rerank .....	64
4.4.2. Đánh giá chất lượng sinh .....	64
4.4.2.1.1. Tạo bộ dữ liệu .....	64
4.4.2.1.2. Thực hiện đánh giá .....	65
4.4.3. Nhận xét .....	65
5. Kết luận .....	65
5.1. Kết quả đạt được .....	65
5.2. Hướng phát triển .....	65
Lời kết .....	67
Tài liệu tham khảo .....	68

## List of Figures

Hình 2: Ví dụ về tokenized .....	14
Hình 3: Mô hình Transformer [1] .....	14
Hình 4: Ví dụ về attention ( <a href="#">Nguồn</a> ) .....	15
Hình 5: Trọng số Attention ( <a href="#">Nguồn</a> ) .....	15
Hình 6: Scaled Dot-Product Attention và Multi-Head Attention [1] .....	16
Hình 7: Vector nhúng từ ( <a href="#">Nguồn</a> ) .....	19
Hình 8: Các tác vụ khử nhiễu [2]. .....	21
Hình 9: Sự phát triển về kích thước của các mô hình ngôn ngữ. ....	22
Hình 10: RAG Framework [3] .....	26
Hình 11: Kiến trúc hệ thống .....	27
Hình 12: Tổ chức dữ liệu: Thông tin giáo viên .....	28
Hình 13: Tổ chức dữ liệu: Bài báo khoa học .....	28
Hình 14: Tổ chức dữ liệu: Khóa luận UET .....	29
Hình 15: Tổ chức dữ liệu: Khóa luận ngoài UET .....	29
Hình 16: BERT ( <a href="#">Nguồn</a> ) .....	30
Hình 17: Bi-EncoderEncoder .....	30
Hình 18: Tìm kiếm thất bại trên NSW .....	32
Hình 19: Kiến trúc HNWS .....	32
Hình 20: Mô hình GPT-3 .....	33
Hình 21: Sự ảnh hưởng của Temperature đến xác suất ( <a href="#">Nguồn</a> ) .....	34
Hình 22: Cross-Encoder [3] .....	35
Hình 23: Cross-Encoder .....	36
Hình 24: Ảnh hưởng của sắp xếp thông tin tới hiệu năng của mô hình [4] .....	36

Hình 25: Khoảng cách Levenshtein .....	37
Hình 26: Khoảng cách Levenshtein .....	38
Hình 27: Biểu đồ ca sử dụng .....	43
Hình 28: Biểu đồ tuần tự: Hệ thống .....	44
Hình 29: Biểu đồ tuần tự: Truy xuất thông tin cá nhân .....	44
Hình 30: Biểu đồ tuần tự: Truy xuất thông tin thực hiện khóa luận .....	44
Hình 31: Biểu đồ tuần tự: Tìm kiếm thông tin giáo viên .....	45
Hình 32: Biểu đồ tuần tự: Tìm kiếm bài báo .....	45
Hình 33: Biểu đồ tuần tự: Tìm kiếm khóa luận UET .....	46
Hình 34: Biểu đồ tuần tự: Tìm kiếm khóa luận khác .....	47
Hình 35: Biểu đồ tuần tự: Tìm kiếm công trình cụ thể .....	48
Hình 36: Biểu đồ tuần tự: Tìm kiếm thầy giáo hướng dẫn .....	48
Hình 37: Biểu đồ tuần tự: Gợi ý khóa luận tốt nghiệp .....	49
Hình 38: Biểu đồ tuần tự: Thu hồi ngũ cảnh .....	49
Hình 39: Biểu đồ tuần tự: Hậu xử lý truy xuất .....	49
Hình 40: Giao diện bản mẫu .....	52
Hình 41: Kết quả: Tìm kiếm thông tin cá nhân .....	53
Hình 42: Ngũ cảnh: Tìm kiếm thông tin cá nhân .....	53
Hình 43: Kết quả: Tìm kiếm thông tin giáo viên .....	53
Hình 44: Ngũ cảnh: Tìm kiếm thông tin giáo viên .....	53
Hình 45: Kết quả: Hỏi về thực hiện khóa luận .....	54
Hình 46: Ngũ cảnh: Hỏi về thực hiện khóa luận .....	54
Hình 47: Kết quả: Tìm kiếm khóa luận theo chủ đề .....	54
Hình 48: Ngũ cảnh: Tìm kiếm khóa luận theo chủ đề .....	55
Hình 49: Kết quả: Tìm kiếm khóa luận theo giáo viên hướng dẫn .....	55
Hình 50: Ngũ cảnh: Tìm kiếm khóa luận theo giáo viên hướng dẫn .....	56
Hình 51: Kết quả: Tìm kiếm khóa luận theo chủ đề của giáo viên hướng dẫn .....	56
Hình 52: Ngũ cảnh: Tìm kiếm khóa luận theo chủ đề của giáo viên hướng dẫn .....	57
Hình 53: Kết quả: Tìm kiếm bài báo theo chủ đề .....	57
Hình 54: Ngũ cảnh: Tìm kiếm bài báo theo chủ đề .....	58
Hình 55: Kết quả: Tìm kiếm bài báo theo tác giả .....	58
Hình 56: Ngũ cảnh: Tìm kiếm bài báo theo tác giả .....	59
Hình 57: Kết quả: Tìm kiếm bài báo theo chủ đề của tác giả .....	59
Hình 58: Ngũ cảnh: Tìm kiếm bài báo theo chủ đề của tác giả .....	60
Hình 59: Kết quả: Tìm kiếm công trình cụ thể .....	60
Hình 60: Ngũ cảnh: Tìm kiếm công trình cụ thể .....	60
Hình 61: Kết quả: Gợi ý giáo viên hướng dẫn .....	61
Hình 62: Ngũ cảnh: Gợi ý giáo viên hướng dẫn .....	61
Hình 63: Kết quả: Gợi ý đề tài khóa luận .....	61
Hình 64: Ngũ cảnh: Gợi ý đề tài khóa luận .....	62
Hình 65: Kết quả: Đánh giá câu trả lời .....	62
Hình 66: Kết quả: Đánh giá câu trả lời .....	62
Hình 67: Kết quả: Đánh giá câu trả lời .....	64

## List of Tables

Bảng 1: Công thức TF-IDF .....	20
Bảng 2: Công thức BM25 .....	21
Bảng 3: Tổng quan dữ liệu. Các bảng đều bao gồm nhiều cột. Chỉ các cột quan trọng được liệt kê ở trên. ....	28
Bảng 4: Mô tả ngắn gọn bộ công cụ .....	41
Bảng 5: Tổng quan dữ liệu .....	50
Bảng 6: Giá sử dụng API của OpenAI .....	51
Bảng 7: So sánh mô hình nhúng .....	63

## Bảng chú giải thuật ngữ

Viết tắt	Nguyên bản	Dịch/Giải nghĩa
	Token	Đơn vị đầu vào nhỏ nhất của mô hình, một từ được cấu tạo từ nhiều token
	Embedding	Vector nhúng. Biểu diễn số học của ngôn ngữ tự nhiên (Từ, câu, đoạn văn)
	Transformer	Một mô hình học sâu
	Encoder	Một trong 2 thành phần chính của mô hình Transformer
	Decoder	Một trong 2 thành phần chính của mô hình Transformer
GPT	Generative Pre-trained Transformer	Một mô hình sinh
BERT	Bidirectional Encoder Representation	Một mô hình mã hóa
	Benmark	Kiểm chuẩn. Bộ dữ liệu lớn dùng để đánh giá mô hình
LLM	Large Language Model	Mô hình ngôn ngữ lớn.
	Prompt	Đầu vào của LLM
RAG	Retriwal-Augmented Generation	Khung làm việc kết hợp tìm kiếm tri thức bên ngoài làm đầu vào cho LLM
ANN	Approximate Nearest Neighbors	Xấp xỉ hàng xóm gần nhất. Các thuật toán tìm các đối tượng gần nhất xung quanh một mục tiêu. Độ chính xác là không tuyệt đối nhưng đủ tốt và nhanh hơn nhiều so với tìm kiếm chính xác tuyệt đối.
HNSW	Hierachy navigable seach world	Một phương pháp đánh chỉ mục và tìm kiếm trong không gian vector

# **1. Mở đầu**

## **1.1. Đặt vấn đề**

Khóa luận tốt nghiệp là môn học cuối cùng trong hành trình học tập tại đại học của một sinh viên. Đây là công trình nghiên cứu khoa học do một hoặc nhiều sinh viên sắp tốt nghiệp hệ cử nhân thực hiện. Mục đích của khoá luận tốt nghiệp là để người học thể hiện được khả năng và sự hiểu biết chuyên sâu về ngành học của họ, đồng thời phản ánh việc áp dụng những kiến thức đã học trong suốt quá trình đào tạo để nghiên cứu và giải quyết một vấn đề mang tính lý thuyết hoặc thực tiễn. Do yêu cầu phải có nền tảng kiến thức sâu rộng và vững chắc, khóa luận sẽ chỉ được thực hiện sau khi sinh viên đã hoàn thành một số tín chỉ nhất định. Hội đồng trường sẽ sử dụng khóa luận tốt nghiệp như một công cụ để đánh giá lại kiến thức và kỹ năng áp dụng lý thuyết của sinh viên sau quá trình theo học tại trường. Đây cũng là môn học quan trọng nhất vì nó có số tín chỉ cao nhất trong chương trình đào tạo.

Rõ ràng, việc thực hiện khóa luận tốt nghiệp là không hề đơn giản. Thực tế, sinh viên còn gặp phải không ít khó khăn trong quá trình thực hiện khóa luận của mình, trải dài từ: Hiểu về khóa luận, chọn chủ đề, tìm kiếm giáo viên, tìm tài liệu ... Trước mỗi đợt bảo vệ, nhà trường đều sẽ hỗ trợ sinh viên tìm giáo viên phù hợp với đề tài. Việc này gây tiêu tốn nhiều thời gian của các cán bộ. Không những thế, việc tham khảo các khóa luận liên quan bị hạn chế do không được tổ chức sắp xếp một cách rõ ràng. Hiểu được điều này, khóa luận hướng đến việc hỗ trợ các bạn sinh viên thực hiện khóa luận của mình.

Ngày 30 tháng 11 năm 2022, ChatGPT ra đời và gây ra một cơn sốt trên toàn cầu. Đây là lần đầu tiên một ứng dụng sử dụng mô hình ngôn ngữ được mọi người biết và quan tâm nhiều đến thế. Thành công của chatGPT khiến cho rất nhiều nỗ lực đã đổ vào trong lĩnh vực này, cả về khía cạnh học thuật và công nghiệp. Hàng loạt mô hình nhanh chóng ra đời và cho những kết quả tuyệt vời. Không ít mô hình có hiệu suất vượt qua GPT-3 và thậm chí cả GPT-4. Việc nghiên cứu và ứng dụng các mô hình ngôn ngữ lớn hiện nay đang sôi nổi hơn bao giờ hết. Vì lý do đó, tôi muốn sử dụng mô hình ngôn ngữ trong đề tài của mình, sau cùng hướng tới việc xây dựng hệ thống hỗ trợ thực hiện khóa luận.

## **1.2. Nội dung hệ thống**

Hệ thống được thiết kế như một chatbot. Người dùng tiến hành trò chuyện với hệ thống để thu được thông tin mong muốn. Khả năng trả lời của chatbot xoay quanh 5 yêu cầu chính:

- Hỏi đáp về thực hiện khóa luận
- Tìm kiếm giáo viên hướng dẫn
- Tìm kiếm tài liệu liên quan
- Tìm kiếm thông tin về giáo viên
- Gợi ý đề tài khóa luận

## **1.3. Đóng góp của bản thân**

Trong sản phẩm này tôi đã chịu trách nhiệm tìm hiểu và thử nghiệm công nghệ RAG và từ đó ứng dụng để xây dựng hệ thống và các mô-đun liên quan. Bên cạnh đó tôi

cũng nghiên cứu về mức độ phù hợp của các mô hình có sẵn đối với bài toán. Sau khi hoàn thành xây dựng sản phẩm, tôi cũng lập ra các kịch bản thử nghiệm, tiến hành thực nghiệm và phân tích dữ liệu để đánh giá hiệu năng của hệ thống một cách chính xác nhất.

Đóng góp của khóa luận bao gồm:

- Hệ thống hỗ trợ tư vấn sinh viên thực hiện khóa luận
- Bộ dữ liệu khóa luận
- Bộ dữ liệu tương đồng từ khóa, mô tả

## 1.4. Cấu trúc khóa luận

Khóa luận gồm 5 chương. Các thuật ngữ xuất hiện lần đầu sẽ được viết kèm thuật ngữ gốc ở bên cạnh. Các thuật ngữ sẽ được liệt kê đầy đủ trong bảng chú giải.

### Bố cục của luận án

**Chương 1:** Vấn đề, động lực, giải pháp, đóng góp và cấu trúc khóa luận

**Chương 2:** Trình bày các kiến thức nền tảng để hiểu và xây dựng hệ thống.

**Chương 3:** Phân tích các thành phần của hệ thống, trình bày chi tiết các kiến thức liên quan, làm rõ cấu trúc và cách hoạt động của hệ thống.

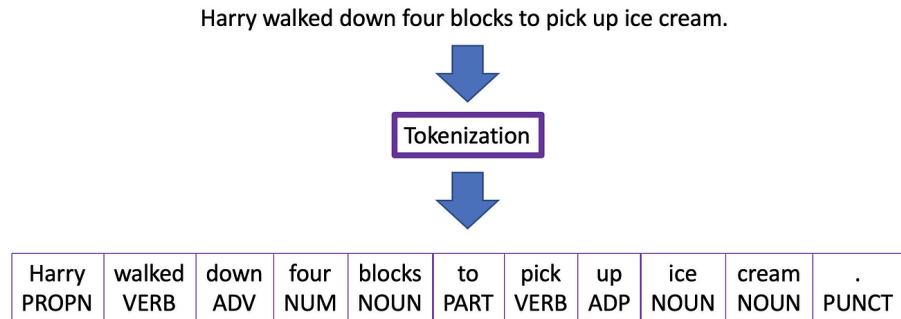
**Chương 4:** Chi tiết cơ sở vật chất và quá trình thực nghiệm cùng kết quả đánh giá hệ thống.

**Chương 5:** Trình bày kết luận được đúc kết ra trong quá trình thực hiện đề tài và hướng phát triển trong tương lai.

## 2. Kiến trúc nền tảng

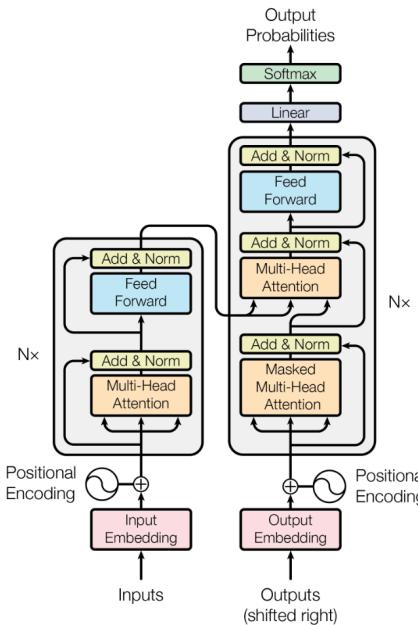
### 2.1. Token

Ngôn ngữ tự nhiên cần phải được xử lý trước khi trở thành đầu vào cho các mô hình ngôn ngữ. Quá trình này được gọi là token hóa. Văn bản đầu vào sẽ được làm sạch và chuẩn hóa, sau đó mỗi từ trong đó sẽ được chia thành nhiều phần nhỏ gọi là các token. Các token này sau đó sẽ được số hóa bằng vị trí của chúng trong từ điển đi kèm. Ngoài ra, còn có một số kí tự đặc biệt được thêm vào chuỗi trong quá trình tokenize.



Hình 2: Ví dụ về tokenized

### 2.2. Kiến trúc Transformer



Hình 3: Mô hình Transformer [1]

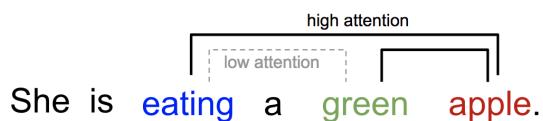
Kiến trúc Transformer được Google đề xuất vào năm 2017 trong bài báo “*Attention is all you need*” [1] để giải quyết bài toán dịch máy. Khi vừa xuất hiện, mô hình đã đạt được điểm số cao trên nhiều kiểm chuẩn và trở thành mô hình tiên tiến nhất lúc bấy giờ. Tiếp nối sự thành công ban đầu, mô hình dần được phát triển và ứng dụng trong toàn bộ miền xử lý ngôn ngữ tự nhiên và sau đó là cả các miền khác như xử lý ảnh, xử lý tiếng nói hay đa phương tiện. Sự ra đời Transformer là một bước tiến lớn trong toàn bộ lĩnh vực trí tuệ nhân tạo. Phần lớn mô hình ra đời sau đó, cho đến tận bây giờ, đều ít nhiều mang trong mình hình bóng của mô hình nguyên bản.

### 2.2.1. Encoder và Decoder

Transformer gồm 2 phần là Encoder (phía bên trái) và Decoder (phía bên phải) như trong Hình 3. Các token của ngôn ngữ nguồn (trong bài toán dịch máy còn gọi là ngữ cảnh) sẽ được đưa qua Encoder tạo ra các vector ngữ cảnh. Các token của ngôn ngữ đích được đưa qua Decoder, kết hợp với vector ngữ cảnh thu được từ Encoder tạo ra xác suất xuất hiện của token tiếp theo. Sau đó, token thu được lại tiếp tục dùng làm đầu vào cho Decoder cho đến khi thu được token báo hiệu sự kết thúc.

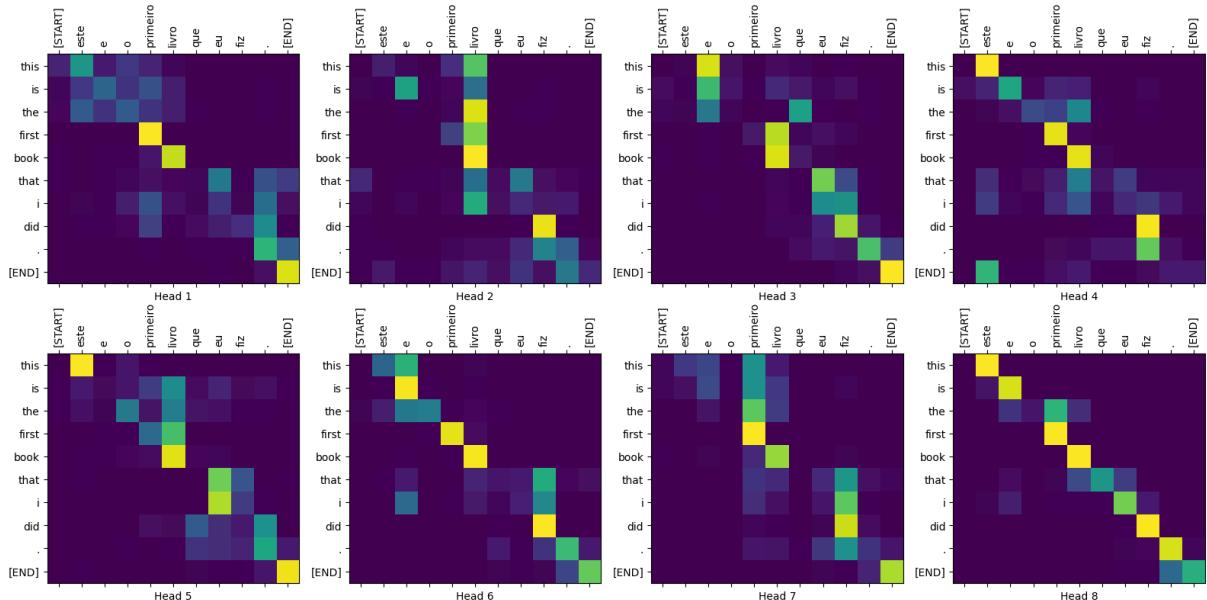
### 2.2.2. Cơ chế Attention

Attention là cốt lõi của Transformer. Đây là cơ chế mô phỏng lại sự chú ý trong nhận thức của con người. Cơ chế này ra đời vì bài toán dịch máy, hướng tới giải quyết vấn đề “quên” thông tin khi ngữ cảnh quá dài trong các mô hình Seq2Seq đương thời. Ý tưởng chính là đánh trọng số mềm (trọng số có thể biến đổi tùy theo đầu vào) cho mỗi token trong ngữ cảnh để thể hiện mức độ quan trọng của token đó đối với từ mục tiêu dưới nhu cầu giải quyết bài toán đích.



Hình 4: Ví dụ về attention (Nguồn)

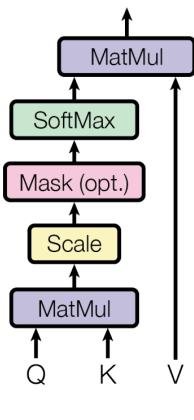
Như trong Hình 4, từ “eating” bổ sung nhiều ý nghĩa cho từ “apple” nhưng không mang lại nhiều ý nghĩa cho từ “green”. Hình 5 là ví dụ về trọng số chú ý giữa 2 câu. Màu càng đậm thể hiện từ tương ứng ở cột càng chú ý đến từ tương ứng ở hàng và ngược lại.



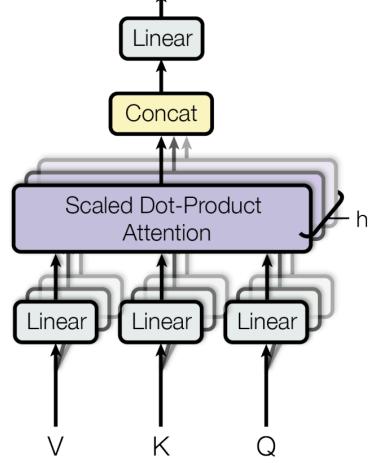
Hình 5: Trọng số Attention (Nguồn)

### Scaled Dot-Product Attention

Scaled Dot-Product Attention



Multi-Head Attention



Hình 6: Scaled Dot-Product Attention và Multi-Head Attention [1]

Scaled Dot-product Attention được giới thiệu lần đầu trong cùng bài báo. Tính toán được minh họa như trong Hình 6 và được mô tả bằng công thức:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{Q \cdot K^T}{\sqrt{d_k}} + M\right) * V \quad [1]$$

Với:

- $Q$ : Ma trận truy vấn, kích thước  $d_q * d_{\text{emb}}$ , thể hiện các token mục tiêu.
- $K$ : Ma trận đáp án, kích thước  $d_k * d_{\text{emb}}$ , thể hiện các token ngữ cảnh.
- $V$ : Ma trận giá trị, kích thước  $d_k * d_{\text{emb}}$ , thể hiện giá trị của các token ngữ cảnh.
- $M$ : Ma trận mặt nạ, kích thước  $d_q * d_k$ , chứa những giá trị 0,1 lọc những giá trị của ma trận khác.
- Softmax: Hàm chuẩn hóa các giá trị trong ma trận về khoảng 0,1.
- $\sqrt{d_k}$ : Được dùng như hệ số giảm giá trị của ma trận, ngăn phần tử của ma trận trở nên quá lớn.
- $d_q$ : Số lượng token mục tiêu.
- $d_k$ : Số lượng token ngữ cảnh.
- $d_{\text{emb}}$ : kích thước của vector nhúng từ.

$K, V$  luôn được trích xuất từ cùng một đầu vào nhưng không nhất thiết là cùng đầu vào với  $Q$ . Nếu  $Q, K, V$  cùng được trích xuất từ cùng một nguồn phép toán được gọi là Self-Attention (Tự chú ý). Nếu  $Q, K, V$  được trích xuất từ 2 nguồn khác nhau, phép toán được gọi là Cross-Attention (Chú ý chéo). Đối với bài toán sinh, trong quá trình huấn luyện, để tránh token phía trước nhìn được token phía sau trong chuỗi token kết quả dẫn đến rò rỉ dữ liệu, ma trận mặt nạ thường được áp dụng để giảm trọng số chú ý của 1 token tới token đứng sau nó xuống vô cùng bé.

### Multi-head Dot-Product Attention

Multi-head Dot-Product Attention là một biến thể của Dot-Product Attention. Tính toán được minh họa trong hình Hình 6 và được mô tả bởi công thức:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O$$

$$\text{với } \text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad [1]$$

Với:

- $\text{head}_i$ : đầu vào thứ i
- $\text{Concat}$ : phép nối ma trận trên chiều được chỉ định
- $W_i^Q$ : ma trận ánh xạ đầu vào thứ i của Q
- $W_i^K$ : ma trận ánh xạ đầu vào thứ i của K
- $W_i^V$ : ma trận ánh xạ đầu vào thứ i của V
- $W^O$ : ma trận tuyến tính
- $\text{Attention}$ : Phép toán Scaled Dot-Product Attention

Các ma trận Q, K, V thay vì được sử dụng trực tiếp trong một phép toán Scaled Dot-Product Attention duy nhất sẽ được đưa qua các ma trận tuyến tính  $W_i^Q, W_i^K, W_i^V$  để thu được các bộ  $q_i, k_i, v_i$  tương ứng có số chiều nhỏ hơn so với các ma trận ban đầu. Mỗi bộ  $q_i, k_i, v_i$  này sẽ được sử dụng như đầu vào cho các phép toán Scaled Dot-Product Attention, mỗi phép toán gọi là một head. Số lượng head, không phải luôn luôn, nhưng như trong bài báo gốc, thường thỏa mãn:

$$n\_head = d_{\text{emb}} / d_q$$

Với:

- $n\_head$ : Số lượng head
- $d_{\text{emb}}$ : Số chiều của vector nhúng từ
- $d_q$ : Số chiều của vector q

để đảm bảo đầu ra của các head sau khi được ghép lại với nhau tạo thành một ma trận tổng hợp có kích thước tương tự như ma trận Q ban đầu. Cuối cùng, ma trận tổng hợp được đưa qua một ma trận tuyến tính  $W^O$ .

Trong Hình 3, phép toán Attention dùng trong Encoder là Self-Attention. Phép toán attention thứ nhất (tính từ dưới lên) của Decoder là Self-Attention, do có sử dụng mặt nạ nên gọi là Masked Self-Attention hay Causal Attention. Phép toán thứ 2, cũng sử dụng mặt là Cross-Attention với vector Q là từ Decoder và vector K, V từ Encoder.

### 2.2.3. Tầng FeedForward

Tầng FeedForward gồm 2 ma trận tuyến tính và một hàm hoạt hóa ReLU ở giữa, đảm bảo đầu ra trước và sau có kích thước giống nhau. Tính toán được biểu diễn bởi công thức:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \quad [1]$$

Với:

- FFN: Khối FeedForward, viết tắt của FeedForward Network
- $W_1$ : Trọng số của ma trận tuyến tính đầu tiên
- $b_1$ : Hệ số tự do của ma trận tuyến tính đầu tiên
- $W_2$ : Trọng số của ma trận tuyến tính thứ 2
- $b_2$ : Hệ số tự do của ma trận tuyến tính thứ 2

## 2.2.4. Tầng Embedding

Ma trận vector nhúng với mỗi vector có kích thước:

$$d_{\text{emb}} * \text{vocab\_size}$$

Với:

- $d_{\text{emb}}$ : Kích thước vector nhúng
- vocab\_size: Kích thước từ điển

Văn bản đầu vào sau khi được tách thành các token và số hóa, sẽ thực hiện ánh xạ để có được biểu diễn vector của chính nó. Trọng số của tầng Embedding được nhân với  $\sqrt{d_{\text{emb}}}$

## 2.2.5. Tầng mã hóa vị trí (Positional Encoding)

Thêm thông tin về không gian, tức vị trí trong chuỗi của token cho vector đầu vào tương ứng. Vector mã hóa vị trí được tính toán theo công thức:

$$\begin{aligned} \text{PE}_{(\text{pos}, 2i)} &= \sin(\text{pos}/10000^{2i/d_{\text{emb}}}) \\ \text{PE}_{(\text{pos}, 2i+1)} &= \cos(\text{pos}/10000^{2i/d_{\text{emb}}}) \end{aligned} [1]$$

Với:

- pos: Vị trí trong chuỗi
- $\text{PE}_{(\text{pos}, 2i)}$ : Mã hóa vị trí chẵn
- $\text{PE}_{(\text{pos}, 2i+1)}$ : Mã hóa vị trí lẻ
- $d_{\text{emb}}$ : Kích thước vector nhúng từ, tương đương với kích thước vector mã hóa vị trí

Các mã hóa này sau đó sẽ được cộng vào các vector có vị trí tương ứng, kích thước của chúng được đảm bảo là  $d_{\text{emb}}$  để có thể thực hiện phép toán cộng. Có nhiều phương pháp mã hóa vị trí, cả có thể học hỏi và cố định, trong bài báo gốc sử dụng hàm sin và cosin với tần số khác nhau

## 2.2.6. So sánh

### Ưu điểm

Trong ngôn ngữ tự nhiên nói riêng, Transformer đã xử lý được 4 vấn đề của các mô hình Seq2Seq được sử dụng rộng rãi trước đó, bao gồm:

- Tăng khả năng xử lý song song: Không còn xử lý tuần tự từng token như mô hình Sequence-to-Sequence, trong Transformer, tất cả đều vào và đều ra được xử lý cùng lúc và chỉ tuần tự theo các tầng.
- Thực sự nhìn ngữ cảnh ở 2 chiều: Mô hình Sequence-to-Sequence chỉ có thể nhìn từ một chiều trái qua phải hoặc phải qua trái và kết hợp kết quả từ cả 2 chiều cho những tác vụ cần hiểu toàn cảnh đầu vào. Cơ chế Attention trong Transformer cho phép một token nhìn tất cả ngữ cảnh cả trước và sau nó.
- Mất mát thông tin: Mô hình Sequence-to-Sequence gấp vấn đề về nút thắt cổ chai trong thông tin(Information bottleneck). Tất cả dữ liệu đầu vào bất kể dài ngắn đều được mã hóa thành một vector ngữ cảnh có kích thước giống nhau. Nếu đầu vào quá dài, kích thước này có thể không đủ để hàm chứa thông tin, dẫn đến mất

mát dữ liệu. Transformer giữ nguyên kích thước của đầu vào sau khi đưa qua các tầng mã hóa. Đầu vào và đầu ra của các tầng mã hóa có kích thước giống hệt nhau.

- Ghi nhớ lâu hơn: Trong mô hình Seq2Seq, đầu vào được xử lý lần lượt và từ đứng càng gần cuối sẽ càng chiếm trọng số cao trong vector ngữ cảnh. Transformer giải quyết điều này bằng cách giữ nguyên kích thước đầu vào.

## Hạn chế

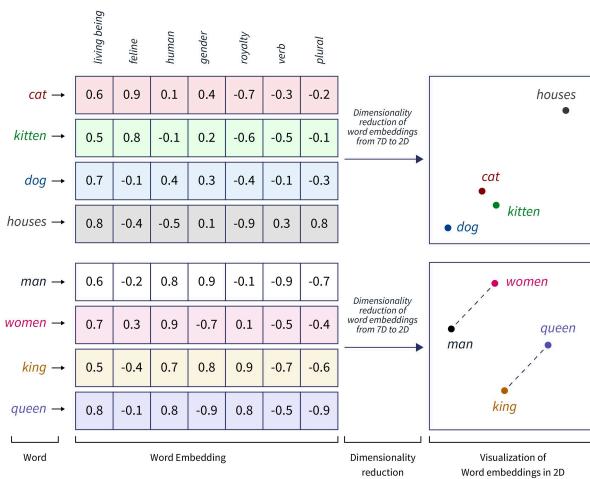
- Yêu cầu bộ nhớ lớn, tỉ lệ với độ dài câu đầu vào
- Độ phức tạp tính toán của Attention là lớn, tỉ lệ với câu đầu vào

## 2.3. Vector nhúng

### 2.3.1. Biểu diễn từ

Máy tính không hiểu được ngôn ngữ tự nhiên nên văn bản cần phải được chuyển đổi thành dạng số học trước khi thực hiện các xử lý sau đó. Cách thức đơn giản nhất là gán cho mỗi từ một số nguyên độc nhất (Ordinal Encoding). Tuy nhiên việc đánh số này dù theo bất kì quy tắc nào đều không bao hàm ngữ nghĩa và hàm chứa thiên kiến về thứ tự và độ liên quan của các từ có số gần nhau.

Một cách khác là biểu diễn mỗi từ bằng vector. Thể hiện sơ khai nhất của phương pháp này là One-hot vector. Mỗi từ được biểu diễn bằng 1 One-hot vector có kích thước tương đương với kích thước của từ điển, mang giá trị 1 tại vị trí của từ đó trong từ điển và 0 tại các vị trí khác. Ví dụ: Happy đứng thứ 64 trong từ điển sẽ có vector biểu diễn mang giá trị 1 tại vị trí 64 và 0 tại các giá trị khác. So với đánh số nguyên, One-hot vector không còn mang thiên kiến mà thay vào đó chỉ thể hiện sự xuất hiện của từ. Ngoài ra, tổng của các One-hot vector còn có thể dùng để thể hiện 1 văn bản, điều mà mã hóa thứ tự không làm được. Tuy nhiên, One-hot vector còn mang những điểm yếu: Vector quá thừa trong khi kích thước lại quá lớn, yêu cầu bộ nhớ cao nhưng không dùng hiệu quả.



Hình 7: Vector nhúng từ (Nguồn)

Cải tiến tiếp theo của biểu diễn vector là các vector ngữ nghĩa, vector nhúng từ (Word embedding). Vector nhúng từ vẫn bao gồm nhiều chiều, nhưng ít hơn nhiều so với One-hot vector, kích thước thông dụng lớn nhất là 784. Mỗi giá trị tại mỗi vị trí của vector

thể hiện giá trị của từ theo một sắc thái nào đó như: trang trọng, tươi tắn, học thuật,.. nhưng sẽ không minh bạch đối với con người. Các vector nhúng từ cần đảm bảo các từ có nghĩa giống nhau sẽ đứng gần nhau trong không gian vector và các từ trái nghĩa sẽ đứng xa nhau. Ví dụ:

$$\text{Queen} = \text{King} - \text{Man} + \text{Woman}$$

Một từ mang ý nghĩa khác nhau tùy thuộc vào ngữ cảnh, vì vậy ngoài những vector nhúng từ tĩnh còn có những Vector nhúng từ dựa trên ngữ cảnh (Contextual Word Embedding).

### 2.3.1.1. Dựa trên thống kê

Vector dựa trên việc đếm phụ thuộc vào tần số xuất hiện của từ và ma trận đồng xuất hiện (co-occurrence matrix) với giả định rằng những từ trong cùng một ngữ cảnh mang nghĩa tương đồng hoặc có liên quan tới nhau. Những mô hình này ánh xạ thống kê dựa trên việc đếm, như xuất hiện cùng nhau, tới một vector dày và nhỏ hơn.

#### 2.3.1.1.1. TF-IDF

Viết tắt của Term Frequency – Inverse Document Frequency, là giá trị thu được qua thống kê của một từ trong một văn bản, thể hiện độ quan trọng của từ này trong văn bản mà bản thân văn bản xét nằm trong 1 tập văn.

Tên	Công thức	Ý nghĩa
Term Frequency	$TF(t,d) = \sqrt{\text{frequency}}$	Tần suất xuất hiện của 1 từ trong văn bản, xuất hiện càng nhiều tương đương với độ quan trọng càng cao.
Inverse Document Frequency	$IDF(t) = \log\left(1 + \frac{\text{docNum}}{\text{docFreq}+1}\right)$	Dùng để đánh giá độ đặc biệt của từ dựa trên tần suất xuất hiện của nó trong tập văn.
Độ dài trường	$\ d\  = \frac{1}{\sqrt{\text{len}(q)}}$	Trường càng ngắn thì tức là độ quan trọng càng cao, hàm sẽ có giá trị càng cao
TF-IDF	$TF\_IDF(t) = TF\_score * IDF\_score * Field\_norm$	Độ quan trọng của một từ trong văn bản.

Bảng 1: Công thức TF-IDF

#### 2.3.1.1.2. BM25

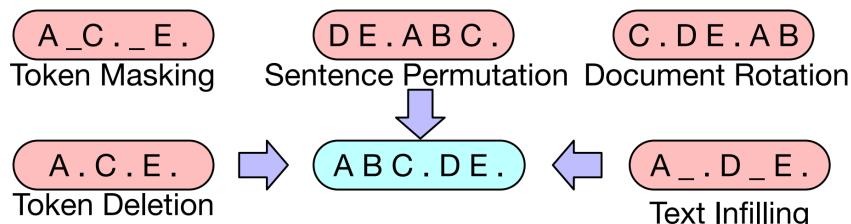
Được cải tiến từ TF-IDF, BM25 đã chỉnh sửa công thức tính lại để thêm khả năng đưa ra giá trị âm khi tần suất xuất hiện của từ trên toàn bộ tập văn quá cao (rất ít đặc biệt).

Tên	Công thức mới	Nhận xét
Term Frequency	$\frac{(k+1) * freq}{k + freq}$	k là hằng số, thường là 1.2
Inverse Document Frequency	$idf(t) = \log\left(\frac{1 + \frac{docNum - docFreq + 0.5}{docFreq + 0.5}}{docFreq + 0.5}\right)$	
Độ dài trường	$TF = \frac{(k+1) * freq}{k * (1.0 - b + b * L) + freq}$ Với: • b là hằng số mặc định là 0.75 • L là tỉ lệ giữa độ dài của trường so với độ dài trung bình của tất cả trường.	Công thức cũ cho kết quả thiếu chính xác với những tài liệu có kích thước quá ngắn hoặc quá dài
BM25	$TF\_IDF(t) = TF\_score * IDF\_score * Field\_norm$	

Bảng 2: Công thức BM25

### 2.3.1.2. Dựa trên ngữ nghĩa

Các vector nhúng từ được tạo nên bằng cách sử dụng những mô hình học máy. Các mô hình này được tiền huấn luyện bằng cách khử nhiễu (denoising), tức là bằng cách nào đó phá hủy câu đầu vào và bắt mô hình khôi phục lại.



Hình 8: Các tác vụ khử nhiễu [2].

Các chiến lược phổ biến được áp dụng phổ biến được mô tả như trong Hình 8, bao gồm:

- Ẩn từ trong văn bản (Masking)
- Xáo trộn từ trong văn bản
- Tịnh tiến câu trong văn bản
- Xóa câu trong văn bản
- Đιền từ còn thiếu

### 2.3.2. Biểu diễn văn bản

#### 2.3.2.1. Dựa trên thống kê

Vector biểu diễn văn bản là có kích thước tương đương với kích thước từ điển, và mỗi từ xuất hiện trong văn bản mang giá trị tính theo TF-IDF hoặc BM25 của chính nó.

#### 2.3.2.2. Dựa trên ngữ nghĩa

Vector ngữ nghĩa của tài liệu thu được bằng cách cho các vector nhúng từ đi qua thu được đi qua 1 tầng pooling. Tầng Pooling có nhiều cách để thực thi:

- Lấy giá trị của token đầu và cuối, là 1 token đặc biệt, thường kí hiệu là [CLS]. Vector ngữ nghĩa của token này được dùng làm vector đại diện cho văn bản.
- Lấy giá trị lớn nhất, nhỏ nhất, trung bình của các vector nhúng từ.

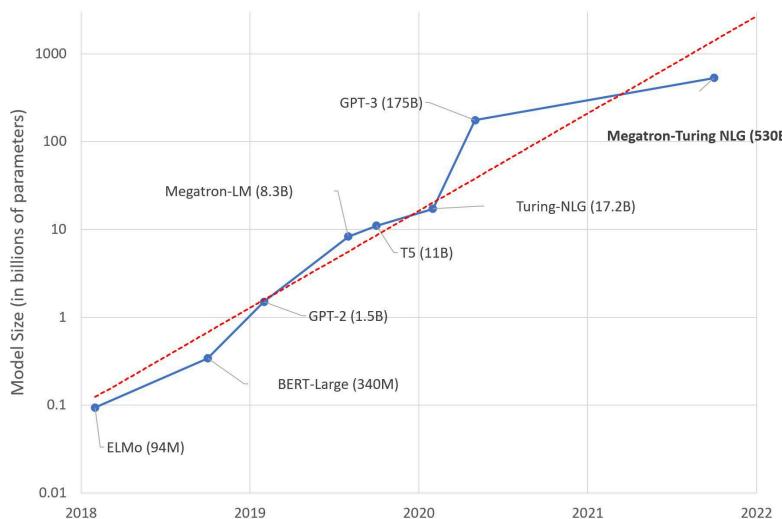
### 2.3.3. Truy xuất

Định nghĩa của các vector nhúng thể hiện rằng vector nhúng của những văn bản có khoảng cách gần nhau trong không gian vector sẽ mang ý nghĩa giống nhau. Điều này đúng với vector ngữ nghĩa dựa theo việc tối ưu hàm mục tiêu. Còn đối với các vector dựa trên thống kê, điều này là không chắc chắn, chỉ có thể dựa trên nhận xét rằng những văn bản xuất hiện nhiều từ giống nhau có khả năng cao sẽ mang ý nghĩa giống nhau. Khoảng cách giữa 2 vector được đo theo nhiều độ đo:

- Hamming
- Cosin
- Tích chấm
- Euclid

Trong đó cosin là độ đo được sử dụng nhiều nhất.

## 2.4. LLM



Hình 9: Sự phát triển về kích thước của các mô hình ngôn ngữ.

Những mô hình như BERT và GPT đã đạt được kết quả tuyệt vời trong nhiều tác vụ xử lý ngôn ngữ tự nhiên. Nhiều mô hình ngôn ngữ nhanh chóng ra đời ngay sau đó và đạt được những kết quả thậm chí còn tốt hơn. Các mô hình này đều có xu hướng tăng dần theo thời gian. Năm 2022 đã có mô hình đạt tới 530 tỉ tham số, so với 213 triệu tham số, gấp 2488 lần so với mô hình Transformer sơ khai.

Những mô hình này đã đạt được những thành tựu đáng ấn tượng và GPT-3 với 175 tỉ tham số ra đời năm 2020 đã lần đầu tiên được thương mại hóa vào 30 tháng 11 năm 2022 bởi OpenAI. GPT-3 vượt trội hơn người tiền nhiệm là GPT-2 với 1.5 tỉ tham số (gấp 167 lần).

Không lâu sau đó GPT-4 ra đời với hiệu suất vượt trội hơn GPT-3 trên mọi kiểm chuẩn. Hiệu suất tốt chứ không phải chỉ chấp nhận được của GPT-3 và GPT-4 đã bắt đầu cuộc đua sản xuất các mô hình sinh. Đầu tiên và các sản phẩm được Fine-Tune dựa trên GPT-4 như Copilot. Sau đó là các mô hình ngôn ngữ lớn vượt qua GPT 3 và thậm chí GPT4 lần lượt ra đời, có cả mã nguồn mở.

Language Model không chỉ dùng để chỉ mô hình sinh, tương tự với Large Language Model. Nhưng trong thời đại hiện nay, khi nhắc về LLM thì người ta sẽ hầu hết liên tưởng tới mô hình sinh.

#### 2.4.1. Base LLM và Instruction LLM

##### Base LLM

Là những mô hình trải qua tiền huấn luyện với dữ liệu đầu vào phong phú từ đủ nguồn và các lĩnh vực khác nhau. Những mô hình này sau khi huấn luyện chỉ đơn thuần viết tiếp câu hỏi của người dùng.

Huấn luyện:

One upon a time, there was a unicorn that lived in a magical forest with her friend.

Suy diễn:

WHAT IS THE CAPITAL OF FRANCE?

“WHAT IS THE CAPITAL OF AMERICA?”

“WHAT IS THE CAPITAL OF ENLAND?”

##### Instruction LLM

Instruction LLM là các mô hình Base LLM đã được tinh chỉnh (fine-tune) trên bộ dữ liệu chỉ dẫn chứa các cặp câu hỏi và câu trả lời, hướng dẫn mô hình đưa ra các câu trả lời hợp lý, chính xác, không thiên vị hay độc hại cho câu hỏi.

Suy diễn:

WHAT IS THE CAPITAL OF FRANCE?

“THE CAPITAL OF FRANCE IS PARIS”

#### 2.4.2. Giới hạn của các mô hình ngôn ngữ lớn

Bên cạnh những điểm mạnh, LLM vẫn còn nhiều hạn chế:

- Ảo giác: Đưa ra câu trả lời nghe có vẻ đúng nhưng thực ra là sai.
- Giới hạn ngữ cảnh: Đầu vào của mô hình bị giới hạn bởi khả năng tính toán dẫn tới việc mất mát thông tin.
- Thiên vị/Độc hại: Mô hình được huấn luyện trên một tập văn khổng lồ chứa cả nội dung độc hại: lăng mạ, phân biệt,... dẫn đến câu trả lời cũng mang theo tính độc hại
- Kiến thức lỗi thời: Kiến thức mà mô hình có được bị giới hạn bởi dữ liệu dùng trong quá trình tiền huấn luyện
- Tiêu tốn tài nguyên: Sử dụng nhiều tham số hơn đồng nghĩa với việc cần nhiều tài nguyên hơn để tính toán và thời gian đưa ra câu trả lời cũng lâu hơn.

### 2.4.3. Prompting

Đầu vào của các mô hình ngôn ngữ lớn thường được gọi là Prompt. Các mô hình ngôn ngữ lớn chưa đủ tốt để có thể luôn hiểu và đưa ra câu trả lời phù hợp cho câu hỏi của người dùng. Do vậy, việc thiết kế Prompt (Prompt Engineering) một cách cẩn thận là cần thiết để có được đáp án mong muốn. Việc tạo Prompt cần phải tuân theo các nguyên tắc:

- Viết chỉ dẫn rõ ràng và cụ thể:
  - Dùng dấu phân cách
  - Yêu cầu đầu ra theo cấu trúc
  - Kiểm tra có đủ giả thiết để thực hiện tác vụ hay không
  - Đưa ra một hoặc một vài ví dụ tương tự câu hỏi
- Cho mô hình thời gian suy nghĩ.
  - Viết các bước cụ thể để hoàn thành tác vụ
  - Hướng dẫn mô hình tự tìm ra lời giải trước khi đưa ra kết luận

Một số kỹ thuật thiết kế lời nhắc phổ biến là:

- One-shot prompt
- Few-show prompt
- Chain of thought
- Tree of thought
- React

### 3. Hệ thống

#### 3.1. Phân tích thành phần

##### 3.1.1. RAG

Những mô hình ngôn ngữ lớn đã được những thành công đáng chú ý trong lĩnh vực xử lý ngôn ngữ tự nhiên, thể hiện hiệu năng vượt trội trên nhiều kiểm chuẩn. Bên cạnh những ưu điểm, mô hình ngôn ngữ lớn còn tồn tại nhiều hạn chế đáng chú ý, cụ thể là trong việc xử lý truy vấn theo miền tri thức cụ thể hoặc có tính chuyên môn cao. Vấn đề phổ biến nhất là đưa ra thông tin sai lệch và thiếu chính xác với giọng văn đầy thuyết phục, được biết đến như là ảo giác (Hallucination), đặc biệt là khi truy vấn đòi hỏi thông tin nằm ngoài dữ liệu được huấn luyện của mô hình. Điều này hạn chế việc triển khai mô hình ngôn ngữ lớn như một giải pháp hộp đen trong môi trường sản xuất của thế giới thực mà không có những biện pháp bảo hộ bổ sung.Thêm nữa, quá trình lý luận của mô hình ngôn ngữ lớn thiếu minh bạch và không thể theo dõi. Không có gì có thể đảm bảo câu trả lời được sinh ra là chính xác, dù truy vấn hoàn toàn chỉ yêu cầu tri thức nằm trong bộ dữ liệu được huấn luyện của mô hình. Điều này là khó chấp nhận với những lĩnh vực cần độ chính xác cao như pháp luật, lập trình và sẽ ngày càng khó có thể chấp nhận trong các lĩnh vực khác trong tương lai.

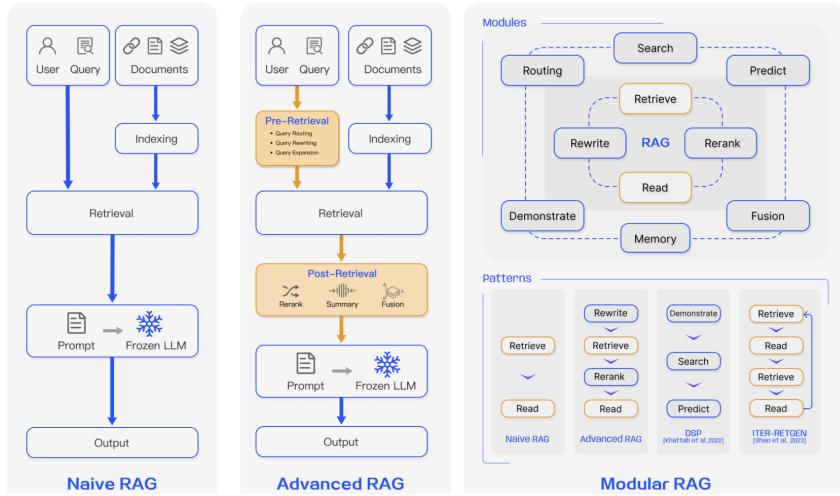
RAG (Retrieval-Augmented Generation) ra đời như một giải pháp hứa hẹn cho những vấn đề này. Bằng cách kết hợp mô hình ngôn ngữ lớn với cơ sở dữ liệu bên ngoài, kết quả thu được có độ chính xác và tính tin cậy cao hơn. Cụ thể, RAG bao gồm bước truy xuất ban đầu trong đó LLM truy xuất nguồn dữ liệu bên ngoài để thu được thông tin có liên quan trước khi tiến hành sinh câu trả lời. Quá trình này không chỉ thêm thông tin cho việc sinh mà còn đảm bảo câu trả lời được chứng thực bởi các thông tin truy xuất, từ đó tăng mạnh độ chính xác và liên quan của câu trả lời.

Một truy vấn được tạo bởi người dùng sẽ khởi động hệ thống truy xuất các thông tin có liên quan trong cơ sở dữ liệu. Những thông tin này sau đó được kết hợp với truy vấn ban đầu tạo thành đầu vào duy nhất cho LLM. Sau cùng, LLM sử dụng đầu vào với đầy đủ câu hỏi và ngữ cảnh để sinh câu trả lời.

##### 3.1.1.1. RAG Framework

Một khung làm việc RAG bao gồm 3 bước chính:

- **Dẫn nhập (Ingestion):** Thu thập và làm sạch dữ liệu, chia chúng thành các đoạn với độ dài phù hợp, chuyển hóa thành vector, đánh chỉ mục và lưu trong cơ sở dữ liệu.
- **Thu hồi (Retrieval):** Tìm kiếm các đoạn trong cơ sở dữ liệu có liên quan đến truy vấn.
- **Tổng hợp (Synthesize):** Mô hình sinh câu trả lời dựa trên thông tin ngữ cảnh thu được từ các đoạn đã truy xuất



Hình 10: RAG Framework [3]

### RAG giản đơn

Phương pháp xuất hiện sớm nhất, nổi lên sau sự lan rộng của ChatGPT. RAG giản đơn tuân theo quá trình đánh chỉ mục, truy xuất và sinh truyền thống. Hay còn được gọi là khung làm việc “Retrieve-Read”.

### RAG nâng cao

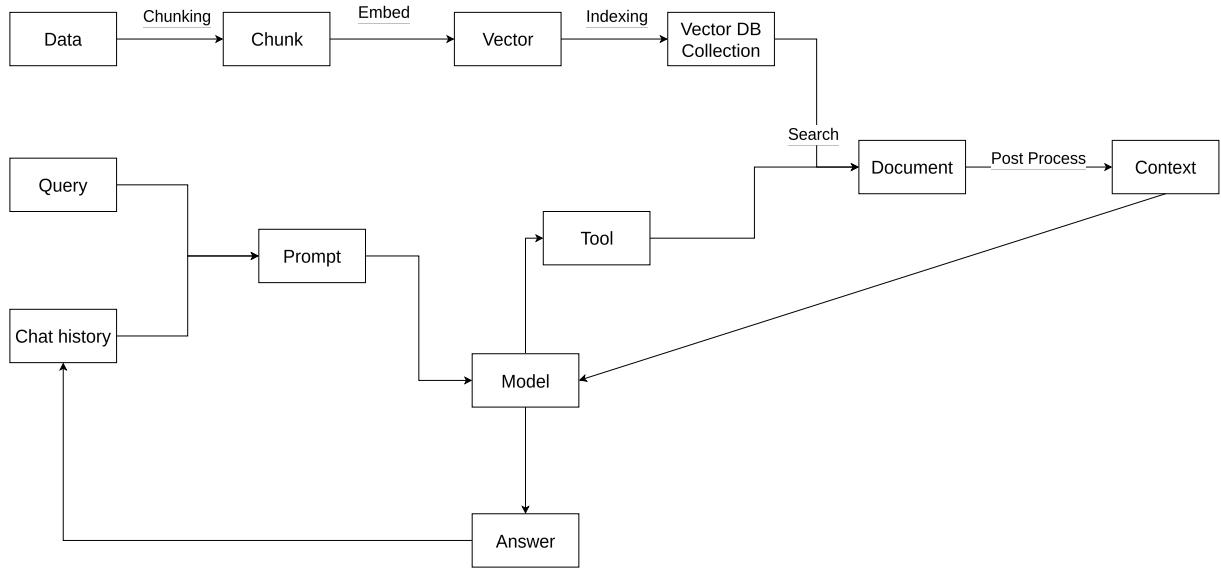
Giống như RAG giản đơn, nhưng có thêm giai đoạn xử lý tiền truy xuất và hậu truy xuất.

### RAG mô-đun

RAG mô-đun là sự tổng quát của 2 khung làm việc trước đó. Cụ thể, RAG nâng cao là một dạng đặc biệt của RAG mô-đun, RAG giản đơn là một dạng đặc biệt của RAG nâng cao. Nhưng so với 2 khung làm việc trước, nó mang lại sự linh hoạt cao hơn, tích hợp nhiều phương pháp để nâng cao các mô đun chức năng. Mô hình RAG mô-đun đang dần trở thành chuẩn trong miền RAG, cho phép quy trình được tuân tự hóa hoặc phương pháp đào tạo từ đầu đến cuối trên nhiều mô-đun.

Những mô-đun mới:

- Mô-đun tìm kiếm (Search Module):
- Mô-đun bộ nhớ (Memory Module)
- Mô-đun điều hướng (Router)
- Thành phần thích ứng tác vụ (Task adapter)
- ...



Hình 11: Kiến trúc hệ thống

Kiến trúc của hệ thống dựa trên khung làm việc RAG mô-đun với các thành phần:

- Bộ điều hướng
- Bộ nhớ
- Mô hình sinh
- Cơ sở dữ liệu
- Bộ công cụ
- Hậu xử lý

Mô tả cách hoạt động:

1. Khởi tạo:
  - Dữ liệu được tách thành các đoạn
  - Các đoạn được chuyển đổi thành vector
  - Các vector được đánh chỉ mục và lưu trữ trong cơ sở dữ liệu vector
2. Sử dụng:
  - Người dùng nhập câu hỏi
  - Hệ thống xác định thông tin cần tìm kiếm để trả lời câu hỏi
  - Mô hình gọi công cụ tương ứng để truy xuất thông tin
  - Công cụ tiến hành truy xuất thông tin và trả về
  - Mô hình trả lời người dùng với thông tin thu được
  - Câu trả lời được lưu vào trong lịch sử trò chuyện

### 3.1.2. Dữ liệu

Dữ liệu được sử dụng bao gồm:

Mô tả	Tổ chức	Kích thước
Thông tin giáo viên	Tên, liên lạc, hướng nghiên cứu	119
Khóa luận UET	Tên, mô tả, từ khóa, người hướng dẫn	2471
Khóa luận ngoài UET	Tên, mô tả, từ khóa, người hướng dẫn	94940
Bài báo khoa học của giáo viên	Gồm nhiều cột. Các thông tin quan trọng là Tên, mô tả, tác giả	1560
Thông tin cá nhân	Tên, trường, hệ đào tạo, điểm các môn, tổng tín chỉ	1
Thông tin khóa luận	Tệp văn bản chứa thông tin liên quan về việc thực hiện khóa luận	1

Bảng 3: Tổng quan dữ liệu. Các bảng đều bao gồm nhiều cột.  
Chỉ các cột quan trọng được liệt kê ở trên.

Bộ môn	Tên	Chức danh	Hướng nghiên cứu	Chủ đề /khoa luận	Liên hệ	HomePage	Link	Number of paper	Papers	Check	Name variation
0	Công nghệ phần mềm	Phạm Ngọc Hùng	PGS.TS	công nghệ phần mềm, phương pháp hình thức cho ...	Các phương pháp đặc tả và kiểm chứng cho các h...	hungpn@vnu.edu.vn	http://uet.vnu.edu.vn/~hungpn/	https://scholar.google.com/citations?hl=en&use...	42.0	Modular Conformance Testing and Assume-Guarant...	1.0 pham ngoc hung.pn hung.hn pham
1	Công nghệ phần mềm	Võ Đinh Hiếu	TS	software architecture, program analysis, kiến ...		hieuvd@vnu.edu.vn	http://www.uet.vnu.edu.vn/~hieuvd	https://scholar.google.com/citations?hl=en&use...	39.0	A technique for generating test data using gen...	1.0 vo dinh hieu.vd hieu.hd vo
2	Công nghệ phần mềm	Đặng Đức Hạnh	TS	software engineering, kỹ nghệ mô hình phần mềm...	Kỹ nghệ mô hình phần mềm (model transformation...)	hanhdd@vnu.edu.vn	http://www.uet.vnu.edu.vn/~hanhdd	https://scholar.google.com/citations?hl=en&use...	20.0	Transformation of UML models to CSP: A case st...	1.0 dang duc hanh.dd hanh.hd dang
3	Công nghệ phần mềm	Vũ Thị Hồng Nhan	TS	data mining, machine learning, moving object s...		vtnhan@vnu.edu.vn		https://scholar.google.com/citations?hl=en&use...	63.0	A Method for Predicting Location of Mobile Use...	1.0 vu thi hong nhan.vt nhan.nht vu
4	Công nghệ phần mềm	Trần Hoàng Việt	TS	software verification testing, software automa...		thv@vnu.edu.vn	http://www.tranhoangviet.name.vn/p/trang-chu.html	https://scholar.google.com/citations?hl=en&use...	27.0	On improvement of assume-guarantee verificatio...	1.0 tran hoang viet.vt vth tran

Hình 12: Tổ chức dữ liệu: Thông tin giáo viên

Title	Link	Check	Clean_Title	Authors	Publication date	Publisher	Journal	Volume	Issue	...	Total citations	Inventors	Patent office	Patent number	Application number	Institution	Conference	Source	Book
XÂY DỰNG BẢN ĐỒ PHÂN VÙNG KHẢ NĂNG HẠN HẠN TRỄ...	<a href="https://scholar.google.com/citations?view_op=view_op...">https://scholar.google.com/citations?view_op=view_op...</a>	1	xay dung ban do phan vung kha nang han tre...	Trần Thị Minh Châu, Huỳnh Văn Chung, Trần Thị...	2017/8/1	Tạp chí Khoa học và công nghệ nông nghiệp Trườ...				1.0	1	...							
Danh giá kết quả điều trị sùi daí thân dưới do...	<a href="https://scholar.google.com/citations?view_op=view_op...">https://scholar.google.com/citations?view_op=view_op...</a>	1	danh gia ket qua dieu tri soi dai than duoi do...	Phạm Ngọc Hưng, Phạm Hữu Quốc Việt, Trương Văn...	2023	Y học lâm sàng Bệnh viện Trung Ương Huế				...	Cited by 1								
Tình trạng đáp ứng kháng IgG kháng IgGs kháng SARS-Co...	<a href="https://scholar.google.com/citations?view_op=view_op...">https://scholar.google.com/citations?view_op=view_op...</a>	1	tinh trang dap ung khang the igg khang sars co...	Phan Tân Dân, Nguyễn Cơ Thạch, Nguyễn Lê Khánh...	2023/9/11	Tạp chí Y học Dự phòng				33.0	3 Phu bản	...							

Hình 13: Tổ chức dữ liệu: Bài báo khoa học

Title	Link	Type	Abstract	Keyword	Citation	Author	Advisor	Year	Collection
0 Nghiên cứu gợi ý API trong mã nguồn Java	<a href="https://repository.vnu.edu.vn/handle/VNU_123/1...">https://repository.vnu.edu.vn/handle/VNU_123/1...</a>	Thesis	Hiện nay, API là một công cụ hữu ích cho các n...	Java, Kỹ thuật phần mềm	Trần, M. C. (2023). Nghiên cứu gợi ý API trong...	Trần, Manh Cường	Võ, Đinh Hiếu	2023	UET - Conference Papers
1 Cải tiến phương pháp sinh dữ liệu kiểm thử tự ...	<a href="https://repository.vnu.edu.vn/handle/VNU_123/1...">https://repository.vnu.edu.vn/handle/VNU_123/1...</a>	Dissertation	Kiểm thử phần mềm tự động được biết đến như là...	Ngôn ngữ lập trình, Kiểm thử tự động	Trần, N. H. (2023). Cải tiến...	Trần, Nguyễn Hương	Phạm, Ngọc Hùng	2023	UET - Conference Papers
2 Algorithms and Hardware Architectures for high...	<a href="https://repository.vnu.edu.vn/handle/VNU_123/1...">https://repository.vnu.edu.vn/handle/VNU_123/1...</a>	Dissertation	This thesis aims to propose efficient solution...	Mạng thân kính, Điện tử học	Nguyễn, D. A. (2023), Algorithms and Hardware ...	Nguyễn, Duy Anh	Tran, Xuan Tuñacoppi, Francesca	2023	UET - Dissertations
3 Nghiên cứu và xây dựng hệ thống cảnh báo truyề...	<a href="https://repository.vnu.edu.vn/handle/VNU_123/1...">https://repository.vnu.edu.vn/handle/VNU_123/1...</a>	Dissertation	Ứng dụng mạng cảm biến không dây WSN (Wireless...	Kỹ thuật điện tử : Mạng cảm biến không dây	Giản, Q. A. (2023). Nghiên cứu và xây dựng hệ ...	Giản, Quốc Anh	Trần, Đức Tân&Bùi, Tiên Diệu	2023	UET - Dissertations
4 Nghiên cứu chế tạo hạt nano Ag, Au và nanocomp...	<a href="https://repository.vnu.edu.vn/handle/VNU_123/1...">https://repository.vnu.edu.vn/handle/VNU_123/1...</a>	Dissertation	Xây dựng hệ tương tác plasma chất lỏng để chế ...	Vật liệu ; Linh kiện	Nguyễn, T. T. T. (2023), Nghiên cứu chế tạo ha...	Nguyễn, Thị Thu Thủy	Nguyễn, Thế Hiện&Đỗ, Hoàng Tùng	2023	UET - Dissertations

Hình 14: Tổ chức dữ liệu: Khóa luận UET

Title	Link	Abstract	Keywords	Advisor
0 Đề xuất phương án nâng cấp hệ thống điều khiển...	<a href="https://dlib.hust.edu.vn/handle/HUST/14407">https://dlib.hust.edu.vn/handle/HUST/14407</a>	iới thiệu về mạng Profibus và cấu trúc mạng ...	Hệ thống điều khiển; Giám sát	Hoàng Minh Sơn
1 Ứng dụng thuật toán logic mờ để điều khiển tốc...	<a href="https://dlib.hust.edu.vn/handle/HUST/14406">https://dlib.hust.edu.vn/handle/HUST/14406</a>	Khái quát về động lực học và các nhiễu ảnh hưở...	Logic mờ; Tàu thủy; Ứng dụng	Phạm Thượng Hán
2 Nghiên cứu hiệu quả của công nghệ truyền tải d...	<a href="https://dlib.hust.edu.vn/handle/HUST/14470">https://dlib.hust.edu.vn/handle/HUST/14470</a>	Tổng quan về công nghệ truyền tải điện một chí...	Hệ thống điện; Tải điện; Điện một chiều	Lã Văn Út
3 Phân tích ngắn mạch và nghiên cứu bảo vệ role ...	<a href="https://dlib.hust.edu.vn/handle/HUST/14471">https://dlib.hust.edu.vn/handle/HUST/14471</a>	Giới thiệu về ngắn mạch phân tán. Tính toán n...	Bảo vệ rơ le; Ngắn mạch; Nguồn điện phân tán	Trần Bách
4 Sử dụng mô hình cháy AVL-MCC trên phần mềm mô ...	<a href="https://dlib.hust.edu.vn/handle/HUST/14468">https://dlib.hust.edu.vn/handle/HUST/14468</a>	Trình bày hệ thống luân hồi khí thải và lọc bụi...	Động cơ diesel	Lê Anh Tuấn

Hình 15: Tổ chức dữ liệu: Khóa luận ngoài UET

### 3.1.3. Phân đoạn

Độ dài đầu vào của mô hình nhúng bị giới hạn bởi khả năng tính toán. Thêm nữa, độ dài đầu vào của mô hình sử dụng trong quá trình huấn luyện là cố định. Tokenizer sẽ tự động cắt ngắn nếu đầu vào vượt quá mức cho phép. Ngoài ra, nếu cố ý sử dụng đầu vào có độ dài khác với độ dài được sử dụng để huấn luyện của mô hình thì hiệu suất có thể bị giảm xuống. Vì vậy, văn bản cần phải được chia thành các đoạn có độ dài nhỏ hơn trước khi được chuyển đổi thành các vector nhúng.

### Phân đoạn theo câu

Văn bản được tách thành các đoạn với ưu tiên đảm bảo tính toàn vẹn của câu. Với tham số là một loạt các kí tự phân tách, thuật toán cố gắng tách văn bản dựa trên mức độ ưu tiên cho tới khi các đoạn đủ nhỏ. Các kí tự phân tách thường dùng là: [“\n\n”, “\n”, “ “, “”]. Phương pháp này cho phép giữ các đoạn văn (rồi đến câu, từ) gắn kết nhất có thể vì chúng thường mang liên kết ngữ nghĩa lớn nhất trong một đoạn văn bản.

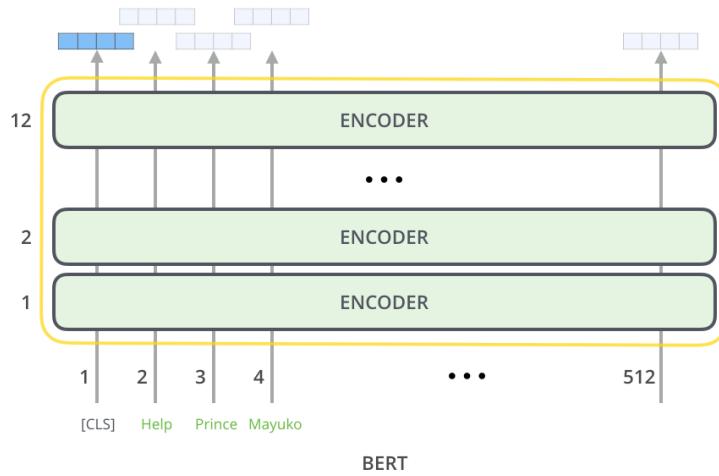
### 3.1.4. Nhúng

#### 3.1.4.1. Dựa trên thống kê

Sử dụng BM-25 để tạo ra các vector dựa trên việc đếm tần suất xuất hiện của các từ trong một văn bản và tập văn tương ứng.

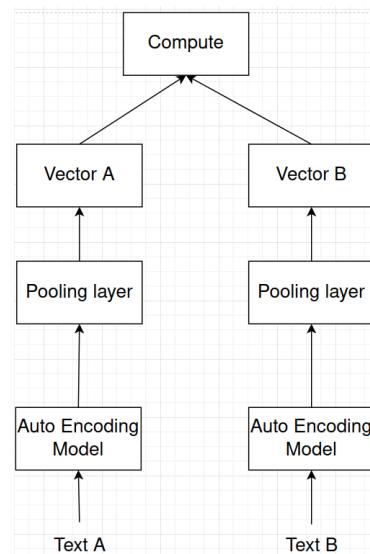
#### 3.1.4.2. Dựa trên ngữ nghĩa

Encoder trong mô hình Transformer có thể được sử dụng một cách độc lập với Decoder. Self-Attention là đặc trưng của Encoder, cho phép nhìn dữ liệu đầu vào từ cả hai chiều, phù hợp với những bài toán cần hiểu toàn bộ đầu vào như trả phân loại sắc thái, phân loại từ và cả sinh vector ngữ nghĩa. Các mô hình chỉ có bộ mã hóa thường được gọi là *Mô hình mã hóa tự động (Auto-encoding models)*. BERT là đại diện của dòng mô hình này.



Hình 16: BERT (Nguồn)

Kiến trúc của BERT gồm nhiều tầng Encoder giống như trong mô hình transformer nguyên bản xếp chồng lên nhau. Ma trận đầu ra có kích thước tương tự ma trận đầu vào, mỗi vector của ma trận đầu ra là biểu diễn vector dựa theo ngữ cảnh của từ tương ứng ở đầu vào.



Hình 17: Bi-EncoderEncoder

BERT được tiền huấn luyện trên tác vụ khử nhiễu và chỉ dùng lại ở việc sinh ra các vector nhúng từ. Để có thể sinh ra các vector nhúng văn bản, BERT cần phải được tinh chỉnh. Bằng việc thêm một tầng Pooling và đổi hàm mục tiêu, các mô hình dựa trên BERT hay Auto-Encoding model có thể sinh ra các vector nhúng văn bản phục vụ tìm kiếm tương đồng dựa trên ngữ nghĩa. Hàm mục tiêu trong tác vụ này là tối ưu hóa khoảng cách dựa trên một độ đo nào đó của văn bản đầu vào. Hàm mục tiêu thường được dụng là khoảng cách cosin.

Các mô hình này được gọi là Bi-Encoder (Mô hình mã hóa song song). Hai bên trái phải của mô hình trong Hình 17 giống hệt nhau về tham số. Dựa chuỗi token của 2

văn bản A, B qua mô hình lần lượt thu được 2 vector tương ứng. Thực hiện tính toán giá trị mất mát trên 2 vector này để có kết quả như mong muốn.

### 3.1.5. Đánh chỉ mục

Khi số lượng vector là quá nhiều, việc tìm kiếm với độ chính xác 100% bằng các thuật toán như kNN sẽ tiêu tốn rất nhiều thời gian. Điều này là không thể chấp nhận khi thực tế yêu cầu những truy vấn cần được xử lý và đưa ra kết quả nhanh chóng, thậm chí là ngay lập tức. Các thuật toán xấp xỉ hàng xóm gần nhất (Approximate Nearest Neighbors - ANN) ra đời để giải quyết vấn đề trên, đánh đổi một lượng nhỏ tỉ lệ chính xác để thỏa mãn yêu cầu về thời gian.

HNSW (Hierarchical Navigable Small Worlds) cũng là một thuật toán như thế. Được giới thiệu lần đầu tiên vào năm 2016 trong bài báo *Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs* [5], HNSW được tin dùng vì đạt được hiệu năng tuyệt vời, tốc độ tìm kiếm nhanh, độ hồi tưởng tốt. Đến bây giờ vẫn là một trong những thuật toán tốt nhất.

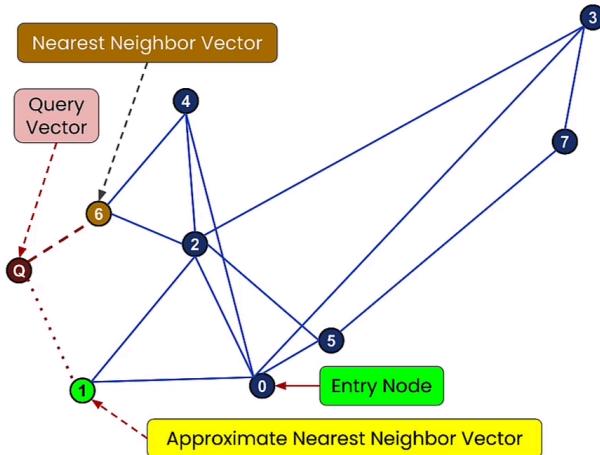
Thuật toán lấy ý tưởng từ một hiện tượng xã hội: hai cá thể bất kì có thể được kết nối thông qua 8 liên kết thân quen. Ví dụ: Với A và B là người lạ, có thể tìm được 6 người C, D, E, F, G, H thỏa mãn A quen C, C quen D, D quen E, E quen F, F quen G, G quen B.

Các thuật toán ANN có thể được chia thành 3 loại: cây, bảng băm và đồ thị. HNSW là loại đồ thị, cụ thể hơn là đồ thị lân cận các đỉnh được liên kết dựa trên khoảng cách giữa chúng, thường là khoảng cách Euclidean.

### Đồ thị NSW (Navigable Small World Graph)

Nền tảng của thuật toán là các đồ thị NSW (Navigable Small World Graph) - đồ thị mà mỗi đỉnh sẽ liên kết tới một số đỉnh gần nhất quanh nó. Trong đồ thị này, với một đỉnh cho trước, ta cần tìm các đỉnh gần nó nhất. Để làm được điều này, ta sử dụng tìm kiếm định tuyến tham lam để di chuyển giữa các đỉnh. Tại mỗi bước, ta xác định đỉnh gần nhất trong danh sách liên kết của đỉnh hiện tại rồi di chuyển đến đỉnh đó. Cuối cùng, khi không tìm được đỉnh nào gần hơn đỉnh hiện tại, ta dừng thuật toán. Điểm thu được là cực trị địa phương. Độ phức tạp của việc thực hiện điều hướng tham lam là (poly) logarithmic. Hiệu quả của định hướng tham lam bị phá vỡ nếu mạng có số lượng đỉnh lớn (1-10k+ đỉnh) và mất khả năng định hướng.

Quá trình định hướng gồm 2 giai đoạn. Bắt đầu với giai đoạn thu nhỏ khi đi qua các đỉnh có bậc thấp. Tiếp đó là giai đoạn phóng to với các đỉnh có bậc cao. Điều kiện dừng là không tìm thấy đỉnh nào có khoảng cách gần hơn trong danh sách kề. Do vậy mà thuật toán thường bị dừng quá sớm ở giai đoạn phóng to khi ít liên kết, ít có khả năng tìm được đỉnh gần hơn.

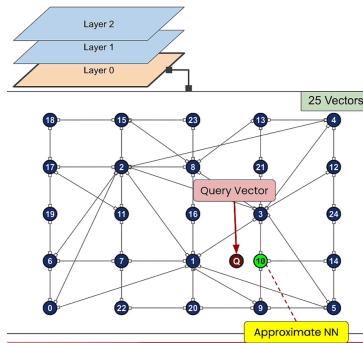


Hình 18: Tìm kiếm thất bại trên NSW

NSW là một thuật toán gần đúng nên vẫn có khả năng tìm kiếm cực trị toàn cục thất bại như trong Hình 18. Để giảm thiểu khả năng dừng sớm và tăng độ hồi tưởng, có thể tăng trung bình bậc của các đỉnh, nhưng điều này dẫn tới đồ thị phức tạp hơn và thời gian tìm kiếm lâu hơn. Cần phải cân bằng giữa cấp bậc trung bình của đỉnh với độ hồi tưởng và tốc độ tìm kiếm.

### Tiêu đề giới khả dối phân tầng (HNSW)

HNSW phiên bản nâng cấp của NSW. Gồm nhiều tầng, tầng dưới cùng là một NSW có đầy đủ liên kết của không gian vector. Các tầng phía trên là một NSW chứa một phần liên kết của tầng phía dưới. Tầng cao nhất có các liên kết xa nhất và tầng thấp nhất có các liên kết gần nhất.



Hình 19: Kiến trúc HNWS

Trong quá trình tìm kiếm, tiến vào từ tầng cao nhất, các đỉnh ở đây thường có bậc cao, tức là bắt đầu ở giai đoạn thu nhỏ. Sử dụng định hướng tham lam, tìm kiếm cực trị địa phương trên mỗi tầng. Sau đó, di chuyển xuống đỉnh đó ở tầng thấp hơn và tiếp tục tìm kiếm. Thuật toán dừng lại khi tìm thấy cực trị địa phương ở tầng 0.

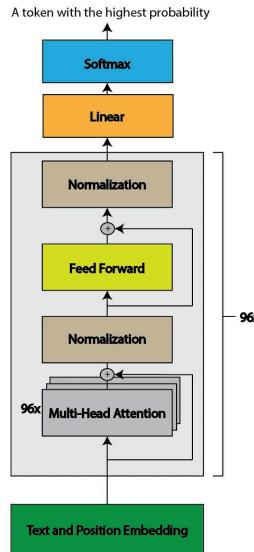
Khi khởi tạo với số tầng là L, bắt đầu từ tầng cao nhất, các vector được lần lượt đưa vào. Xác xuất 1 điểm xuất hiện ở 1 tầng sẽ là m\_L. Tác giả của HNSW thấy rằng hiệu năng tốt nhất đạt được khi tối thiểu trùng lặp của hàng xóm được chia sẻ giữa các lớp. Giảm m\_L giúp giảm sự chồng chéo vì tất cả các nút đều ở tầng 0 nhưng tăng thời

gian tìm kiếm. Cần phải có sự cân bằng và nguyên tắc chung cho giá trị tối ưu này là  $m_L = \frac{1}{\ln(M)}$

### 3.1.6. Mô hình sinh

Mô hình sinh trong ngôn ngữ tự nhiên nhận đầu vào là một đoạn văn bản đưa ra token tiếp theo trong văn bản đó.

#### 3.1.6.1. GPT



Hình 20: Mô hình GPT-3

Tương tự như Encoder, Decoder cũng có thể được sử dụng một cách độc lập. Các mô hình chỉ có Decoder được gọi là “Autoregressive models”, được đặc trưng bởi Casual-Attention, phù hợp với các bài toán sinh và được tiền huấn luyện bằng cách dự đoán token tiếp theo trong chuỗi đầu vào. Đại diện của loại mô hình này là GPT (Generative Pre-trained Transformer). Mô hình sinh cũng có thể bao gồm đầy đủ cả Encoder và Decoder nhưng kiến trúc này ít phổ biến hơn cho bài toán này.

Kiến trúc của GPT gồm nhiều tầng Decoder xếp chồng lên nhau, tuy nhiên tầng Cross-Attention bị loại bỏ do không còn Encoder. GPT-1, GPT-2, GPT-3 cơ bản có cùng kiến trúc nhưng khác nhau về số lượng tham số, công thức tính trọng số Attention, dữ liệu... Kiến trúc của GPT-4 chưa được công bố.

Sau khi đi qua các tầng Decoder, Vector nhúng ngữ cảnh của token cuối cùng trong đầu ra sẽ được đưa qua một tầng tuyến tính rồi dùng hàm Softmax để tính xác xuất trở thành từ tiếp theo trong chuỗi. Hàm softmax ít khi dùng trong huấn luyện mà chỉ dùng trong quá trình suy diễn. Đầu ra thực tế của mô hình là một vector có kích thước bằng với kích thước từ điển gọi là logit thể hiện mức độ phù hợp để trở thành token tiếp theo. Giá trị logit càng cao thì xác xuất một từ có vị trí tương ứng trong từ điển trở thành token tiếp theo sẽ càng cao.

#### 3.1.6.2. Lấy mẫu

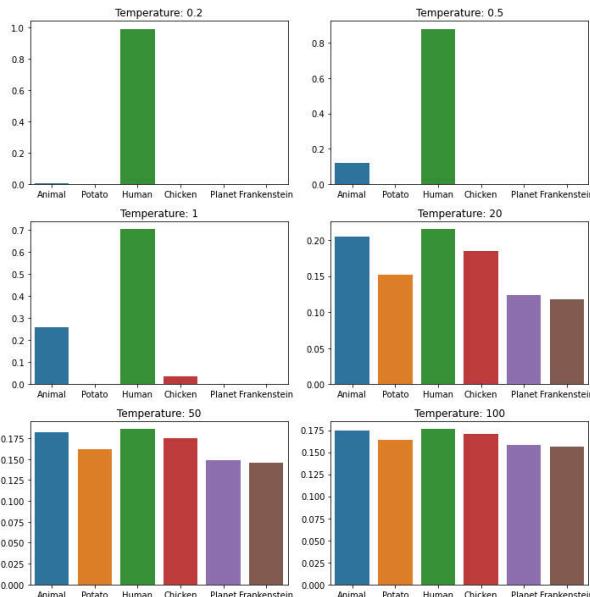
Việc luôn lấy token có giá trị logit cao nhất hay xác suất cao nhất, đảm bảo văn bản luôn sinh ra cùng một câu trả lời với mỗi lần nhập một đầu vào giống nhau. Điều

này là tốt trong những công việc yêu cầu độ chính xác cao nhưng gây ra nhảm chán và kém hiệu quả với những công việc yêu cầu tính sáng tạo. Do vậy, thay vì luôn chọn token có xác suất lớn nhất thì sẽ chọn ngẫu nhiên theo tỉ lệ của từng token. Ngoài ra, các mô hình sinh còn sử dụng một tham số để có thể điều khiển mức độ sáng tạo của mô hình là Temperature. Tham số này được dùng trong hàm Softmax để biến đổi xác suất của từng token. Công thức cụ thể được mô tả như sau:

$$\text{Softmax}(x_i) = \frac{e^{x_i/T}}{\sum_{j=1}^n e^{x_j/T}}$$

Với:

- Softmax: hàm chuyển đổi giá trị về xác suất
- n: Độ dài từ điển
- $x_i$ : Giá trị logit của từ thứ i
- T: Temperature



Hình 21: Sự ảnh hưởng của Temperature đến xác suất ([Nguồn](#))

Trong Hình 21 ta thấy giá trị Temperature càng lớn thì khoảng cách giữa xác suất của các token đối với nhau càng giảm tức tính sáng tạo càng tăng và ngược lại.

Ngoài việc thực hiện lấy mẫu như bình thường, còn có một số kĩ thuật lấy mẫu khác thường được sử dụng.

### Top k

Thay vì lấy mẫu trên toàn bộ từ điển, ta có thể thực hiện lấy mẫu trên k token có giá trị logit lớn nhất.

### Top p

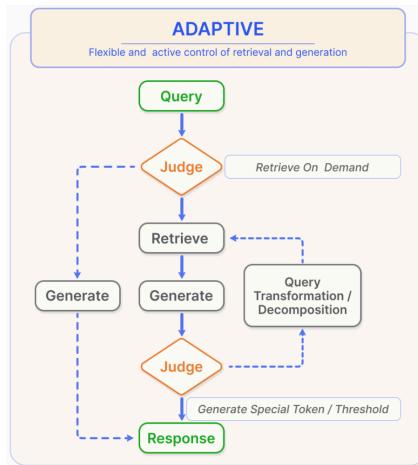
Với các giá trị xác suất thu được, lần lượt lấy các token với xác suất từ cao tới thấp cho tới khi tổng các xác suất vượt quá P rồi thực hiện lấy mẫu.

### 3.1.7. Tiền xử lý

Trích xuất thông tin cần thiết để thực hiện truy xuất từ Prompt

### 3.1.8. Truy xuất

#### Truy xuất thích ứng (Adaptive Retrieval)



Hình 22: Cross-Encoder [3]

Mô hình có thông tin về các công cụ thu thập thông tin và sẽ dùng chúng một khi xác định thông tin do công cụ mang lại là cần thiết để trả lời câu hỏi. Sau khi có được thông tin từ công cụ, mô hình sẽ lại đánh giá xem có cần dùng công cụ khác không. Nếu không, tiến hành sinh câu trả lời. Nếu có tiến hành gọi công cụ tiếp theo cho đến khi mô hình quyết định rằng đã có đủ thông tin, hoặc không có công cụ phù hợp, hoặc đạt giới hạn sử dụng công cụ.

#### Top k

Chỉ truy xuất K giá trị có độ tương đồng cao nhất.

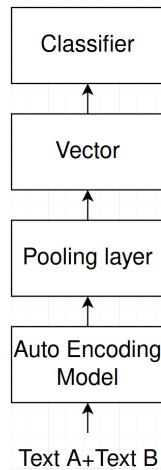
### 3.1.9. Hậu xử lý

#### 3.1.9.1. Lọc

Chỉ những thu hồi những đoạn có độ tương đồng lớn hơn một ngưỡng nào đó

#### 3.1.9.2. Rerank

Việc so sánh độ tương đồng giữa 2 văn bản theo vector nhúng của chúng là tốt nhưng độ chính xác của việc so sánh sẽ cao hơn khi đặt cả 2 văn bản ở cạnh nhau rồi đưa qua một mô hình phân loại. Điều này đã được chứng minh trong bài báo *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*[6]. Tuy nhiên, do phải có cả 2 văn bản để tiến hành so sánh sự tương đồng nên chỉ có thể thực hiện khi số lượng cặp văn bản cần so sánh là nhỏ. Vì vậy, phương pháp này chỉ có được thể sử dụng như một bước hậu xử lý truy xuất, khi số lượng văn bản cần kiểm tra giảm xuống.

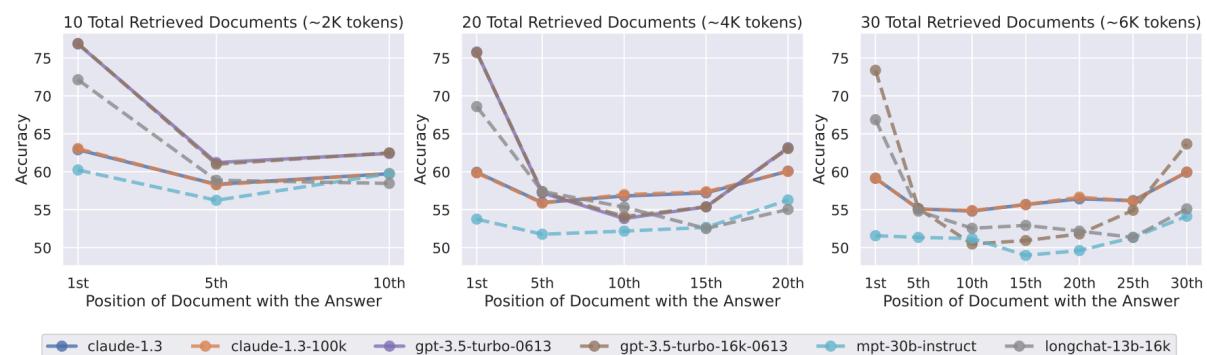


Hình 23: Cross-Encoder

Các mô hình có kiến trúc được mô tả như trong Hình 23 được gọi là Cross-Encoder vì đầu vào là sự kết hợp của 2 văn bản. Giống như mô hình Bi-Encoder, Cross-Encoder cũng dựa trên một mô hình *Auto Encoding*. Hai văn bản sẽ được ghép loại với nhau trở thành một đầu vào duy nhất. Các token của chúng được ngăn cách bởi một token đặc biệt, thường là [SEP]. Vector thu được sau khi đi qua mô hình sẽ được dùng để phân loại liệu 2 văn bản A và B có tương đồng hay không. Classifier là một hay nhiều ma trận tuyến tính.

### 3.1.9.3. Long context reorder

Khi ngữ cảnh quá dài, mô hình có xu hướng tập chung và phần đầu và phần cuối của ngữ cảnh mà bỏ qua thông tin ở giữa. Điều này đã được chỉ ra trong bài báo *Lost in the Middle: How Language Models Use Long Contexts* [4].



Hình 24: Ảnh hưởng của sắp xếp thông tin tới hiệu năng của mô hình [4]

Để mô hình có thể sử dụng ngữ cảnh một cách hiệu quả hơn, việc sắp xếp lại vị trí của các thông tin được truy xuất là cần thiết. Các thông tin có mức độ tương đồng cao sẽ được đặt tại vị trí đầu và cuối ngữ cảnh. Trong khi đó, các thông tin có mức độ tương đồng thấp sẽ được bố trí ở giữa.

### 3.1.10. Lịch sử trò chuyện

Lịch sử trò chuyện cần được lưu lại để mô hình có khả năng đưa ra câu trả lời phù hợp với ngữ cảnh.

### 3.1.11. Prompt

Đầu vào của mô hình được kiến tạo theo công thức:

$$\text{Prompt} = \text{Instruction} + \text{Chat\_history} + \text{User\_input} + \text{Context}$$

Với:

- Prompt: Đầu vào của LLM
- Instruction: Chỉ dẫn thực hiện tác vụ (VD: You are useful assistance)
- Chat\_history: Lịch sử trò chuyện
- User\_input: Câu hỏi của người dùng
- Context: Ngữ cảnh thu được

### 3.1.12. Tìm kiếm mờ

Việc tìm kiếm dữ liệu theo định danh như tên, tiêu đề... là rất cần thiết. Cách đơn giản nhất việc này được tiến hành bằng cách so khớp 2 chuỗi để xem liệu chúng có giống nhau hoàn toàn hoặc chuỗi được so sánh có chứa chuỗi mục tiêu hay không. Tuy nhiên cách tìm kiếm này trong thực tế là không hiệu quả. Yêu cầu tìm kiếm của người dùng có thể có sai sót đến từ lỗi chính tả, nhầm lẫn hoặc chỉ nhớ 1 phần định danh. Điều này khiến định danh được mong muốn không thể được tìm ra nếu thực hiện so khớp. Đây là lúc các thuật toán so khớp chuỗi mờ phát huy tác dụng. Các thuật toán này không so sánh 2 chuỗi một cách tuyệt đối mà dựa trên các độ đo tính bằng khoảng cách chỉnh sửa để đánh giá mức độ tương đồng của chúng.

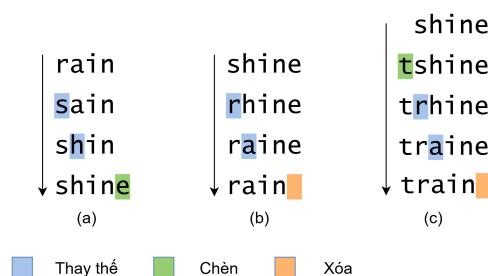
#### 3.1.12.1. Khoảng cách chỉnh sửa

Khoảng cách Levenshtein giữa 2 xâu là số phép biến đổi ít nhất để biến đổi một xâu thành xâu còn lại, các phép biến đổi bao gồm:

- Xóa
- Chèn
- Thay thế

Ví dụ: Khoảng cách Levenshtein giữa 2 chuỗi “kitten” và “sitting” là 3, vì phải dùng ít nhất 3 lần biến đổi.

- kitten -> sitten (thay “k” bằng “s”)
- sitten -> sittin (thay “e” bằng “i”)
- sittin -> sitting (thêm ký tự “g”)



Hình 25: Khoảng cách Levenshtein

Khoảng cách Indel là khoảng cách chỉnh sửa chỉ bao gồm 2 phép thêm và xóa. Vì Thay thế tương đương với việc sử dụng liên tục 2 phép Thêm và Xóa nên có thể coi chi phí cho phép Thay thế là 2.

Để giải bài toán này chúng ta sử dụng thuật toán Wagner–Fischer. Đây là một thuật toán quy hoạch động. Tư tưởng của thuật toán được mô tả bằng đoạn mã bên dưới.

		k	i	t	t	e	n	
	0	1	2	3	4	5	6	
s	1	1	2	3	4	5	6	
i	2	2	1	2	3	4	5	
t	3	3	2	1	2	3	4	
t	4	4	3	2	1	2	3	
i	5	5	4	3	2	2	3	
n	6	6	5	4	3	3	2	
g	7	7	6	5	4	4	3	

		S	a	t	u	r	d	a	y
	0	1	2	3	4	5	6	7	8
S	1	0	1	2	3	4	5	6	7
u	2	1	1	2	2	3	4	5	6
n	3	2	2	2	3	3	4	5	6
d	4	3	3	3	3	4	3	4	5
a	5	4	3	4	4	4	4	3	4
y	6	5	4	4	5	5	5	4	3

Hình 26: Khoảng cách Levenshtein

```
if (a[i]==b[j]) f[i][j]=min(f[i][j]=min(f[i-1][j],f[i][j-1],f[i-1][j-1]))
else f[i][j]=min(f[i][j]=min(f[i-1][j],f[i][j-1],f[i-1][j-1]))+1
```

Với  $f[i][j]$  là khoảng cách tối ưu để biến chuỗi con từ 0 đến  $i$  của xâu a thành chuỗi con từ 0 đến  $j$  của xâu b.

### 3.1.12.2. Các phép toán mở rộng

#### Độ tương đồng - Ratio

Độ tương đồng là độ đo mức độ giống nhau của 2 xâu được tính dựa trên khoảng cách chỉnh sửa.

Độ tương đồng tính theo khoảng cách Levenshtein:

```
dist = Levenshtein_distance(s1,s2)
normalized_distance = dist / max(len(s1), len(s2))
normalized_similarity = 1 - normalized_distance
Example:
Levenshtein.distance('controlled', 'comparative')
8
Levenshtein.similarity('controlled', 'comparative')
3
Levenshtein.normalized_distance('controlled', 'comparative')
0.7272727272727273
Levenshtein.normalized_similarity('controlled', 'comparative')
0.2727272727272727
```

Độ tương đồng tính theo khoảng cách Idel

```

dist = Indel_distance(s1, s2)
normalized_distance = dist / (len(s1) + len(s2))
similarity = 1 - normalized_distance

Example:
Indel.distance('controlled', 'comparative')
13
Indel.similarity('controlled', 'comparative')
8
Indel.normalized_distance('controlled', 'comparative')
0.6190476190476191
Indel.normalized_similarity('controlled', 'comparative')
0.38095238095238093

```

## Fuzz Ratio

Fuzz Ratio là điểm tương đồng được tỉ lệ theo 100.

Ví dụ:

```

String 1: "cat"
String 2: "hat"
> Fuzz Ratio: 66.67

```

## Fuzz Partial Ratio

Gọi hàm ratio giữa 2 xâu ngắn nhất (độ dài n) và các xâu con (độ dài n) của các xâu còn lại. Trả về giá trị điểm cao nhất.

Ví dụ:

```

String 1: "apple pie"
String 2: "pie"
> Fuzz Ratio: 60
> Fuzz Partial Ratio: 100

```

## Token Set Ratito

Các bước thực hiện:

- Tìm tất cả token trong chuỗi, coi chúng như một tập
- Tạo 2 chuỗi theo mẫu: <sorted\_intersection><sorted\_remainder>
- Lấy độ tương đồng theo 3 cách và trả về giá trị cao nhất:
  - [Phần giao] vs [Phần giao + phần dư] của chuỗi 1
  - [Phần giao] vs [Phần giao + phần dư] của chuỗi 2
  - [Phần giao + phần dư] của chuỗi 1 vs [Phần giao + phần dư] của chuỗi 2

Ví dụ:

```

string 1 = "fuzzy was a bear"
string 2 = "fuzzy fuzzy was a bear"
> Token Set Ratio: 84

```

Thuật toán:

```

s1 = " ".join(sorted(s1.split()))
s2 = " ".join(sorted(s2.split()))
intersection = s1.intersection(s2)
remain_1 = tokens1.difference(tokens2)
remain_2 = tokens2.difference(tokens1)

sorted_sect = " ".join(sorted(intersection))
sorted_1 = " ".join(sorted(remain_1))
sorted_2 = " ".join(sorted(remain_2))

combined_1 = sorted_sect + " " + sorted_1
combined_2 = sorted_sect + " " + sorted_2

pairwise = [
    ratio_func(sorted_sect, combined_1),
    ratio_func(sorted_sect, combined_2),
    ratio_func(combined_1, combined_2)
]
> max(pairwise)]

```

### Token Sort Ratio

Token hóa cả hai chuỗi đầu vào, sắp xếp các token theo thứ tự bảng chữ cái và sử dụng Fuzz Ratio với 2 chuỗi mới vừa thu được.

Ví dụ:

```

string 1 = "fuzzy wuzzy was a bear"
string 2 = "wuzzy fuzzy was a bear"
> Token Sort Ratio: 91

```

Mã giả:

```

s1 = " ".join(sorted(s1.split()))
s2 = " ".join(sorted(s2.split()))
> fuzz.ratio(s1, s2)

```

Example:

```

string 1 = "fuzzy wuzzy was a bear"
string 2 = "wuzzy fuzzy was a bear"
> Token Sort Ratio: 91

```

### 3.1.13. Bộ công cụ

Công cụ	Đầu vào	Chức năng
Thu hồi thông tin cá nhân		Trả về thông tin cá nhân của người dùng
Tìm kiếm thông tin thực hiện khóa luận	Truy vấn	Trả về các thông tin về thực hiện khóa luận có liên quan tới truy vấn của người dùng.
Tìm kiếm thông tin giáo viên	Tên giáo viên	Trả về các thông tin có trong cơ sở dữ liệu của giáo viên được tìm kiếm
Tìm kiếm bài báo	Chủ đề Tác giả	<ul style="list-style-type: none"> <li>Trả về các khóa luận có liên quan tới chủ đề của tác giả.</li> <li>Nếu không có tác giả trả về các khóa luận có liên quan tới chủ đề.</li> <li>Nếu không có chủ đề trả về các khóa luận của tác giả.</li> </ul>
Tìm kiếm khóa luận UET	Chủ đề Người hướng dẫn	<ul style="list-style-type: none"> <li>Trả về các khóa luận thuộc UET có liên quan tới chủ đề của người hướng dẫn.-</li> <li>Nếu không có người hướng dẫn trả về các khóa luận có liên quan tới chủ đề thuộc UET.</li> <li>Nếu không có chủ đề trả về các khóa luận thuộc UET của người hướng dẫn</li> </ul>
Tìm kiếm khóa luận ngoài UET	Chủ đề Người hướng dẫn	<ul style="list-style-type: none"> <li>Trả về các khóa luận ngoài UET có liên quan tới chủ đề của người hướng dẫn.</li> <li>Nếu không có người hướng dẫn trả về các khóa luận ngoài UET có liên quan tới chủ đề.</li> <li>Nếu không có chủ đề trả về các khóa luận ngoài UET của người hướng dẫn</li> </ul>
Tìm kiếm công trình cụ thể	Tên bài báo hoặc khóa luận	Trả về thông tin của bài báo hoặc khóa luận có tên được tìm kiếm.
Gợi ý thầy hướng dẫn	Chủ đề	Trả về danh sách các thầy có hướng nghiên cứu liên quan tới chủ đề.
Gợi ý chủ đề khóa luận	Chủ đề Người hướng dẫn	Gợi ý chủ đề khóa luận phù hợp với yêu cầu và hướng nghiên cứu của thầy giáo.

Bảng 4: Mô tả ngắn gọn bộ công cụ

Các công cụ được mô hình dùng để thu thập thông tin cần thiết nhằm trả lời câu hỏi của người dùng. Một công cụ sẽ gồm 3 thành phần chính:

- Lược đồ mô tả các đầu vào của công cụ
- Mô tả chức năng của công cụ
- Đoạn mã thực hiện truy xuất thông tin

### **3.1.14. Đánh giá**

Truy xuất được thực hiện dựa trên các thuật ANN nên không đảm bảo luôn thu được ngữ cảnh phù hợp. Dù hiệu suất của các thuật toán này đã được khẳng định dựa trên nhiều kiểm chuẩn, việc đạt được kết quả tốt trên một hệ thống cụ thể còn liên quan tới việc tinh chỉnh siêu tham số của các thành phần cấu tạo nên công cụ truy xuất như tốp k, ngữ lọc,... Và dù cho đã được cung cấp thêm ngữ cảnh, không thể chắc chắn được rằng câu trả lời của mô hình là luôn chính xác. Việc kiểm tra kết quả sinh là cần thiết để đánh giá hiệu năng của hệ thống cũng như từng thành phần trong nó.

#### **3.1.14.1. Nguyên tắc cơ bản:**

Việc đánh giá kết quả sinh được dựa trên 2 nguyên tắc cơ bản:

- Chất lượng thu hồi (Retrieval Quality): Những độ đo tiêu chuẩn được từ các miền công cụ tìm kiếm, hệ thống đề xuất, và hệ thống thu hồi thông tin được triển khai để đo hiệu năng của mô-đun thu hồi của RAG
- Chất lượng sinh (Generation Quality): Độ đo đánh giá chất lượng sinh phụ thuộc vào kết quả mục tiêu là có nhãn hay không có nhãn. Với kết quả không có nhãn, thẩm định bao gồm tính chân thực, tính liên quan và tính vô hại của câu trả lời được sinh ra. Với kết quả có nhãn, độ đo được sử dụng thường là độ chính xác (Accuracy).

Việc đánh giá cả chất lượng thu hồi và chất lượng sinh đều có thể tiến hành thông qua phương pháp thủ công hoặc tự động.

#### **3.1.14.2. Độ đo**

Có nhiều rất độ đo để đánh giá RAG, nhưng có 3 độ đo được sử dụng nhiều hơn cả là: **Context Relevancy**

Đánh giá mức độ liên quan giữa ngữ cảnh thu được và truy vấn.

$$\text{Contextual Relevancy} = \frac{\text{Number of Relevant Statements}}{\text{Total Number of Statements}} [7]$$

#### **Answer Relevancy**

Đánh giá mức độ liên quan giữa câu trả lời được sinh ra và truy vấn.

$$\text{Answer Relevancy} = \frac{\text{Number of Relevant Statements}}{\text{Total Number of Statements}} [8]$$

#### **Faithfulness**

Đánh giá mức độ khẳng định giữa ngữ cảnh thu được và câu trả lời được sinh ra.

$$\text{Faithfulness} = \frac{\text{Number of Truthful Claims}}{\text{Total Numbers of Claims}} [9]$$

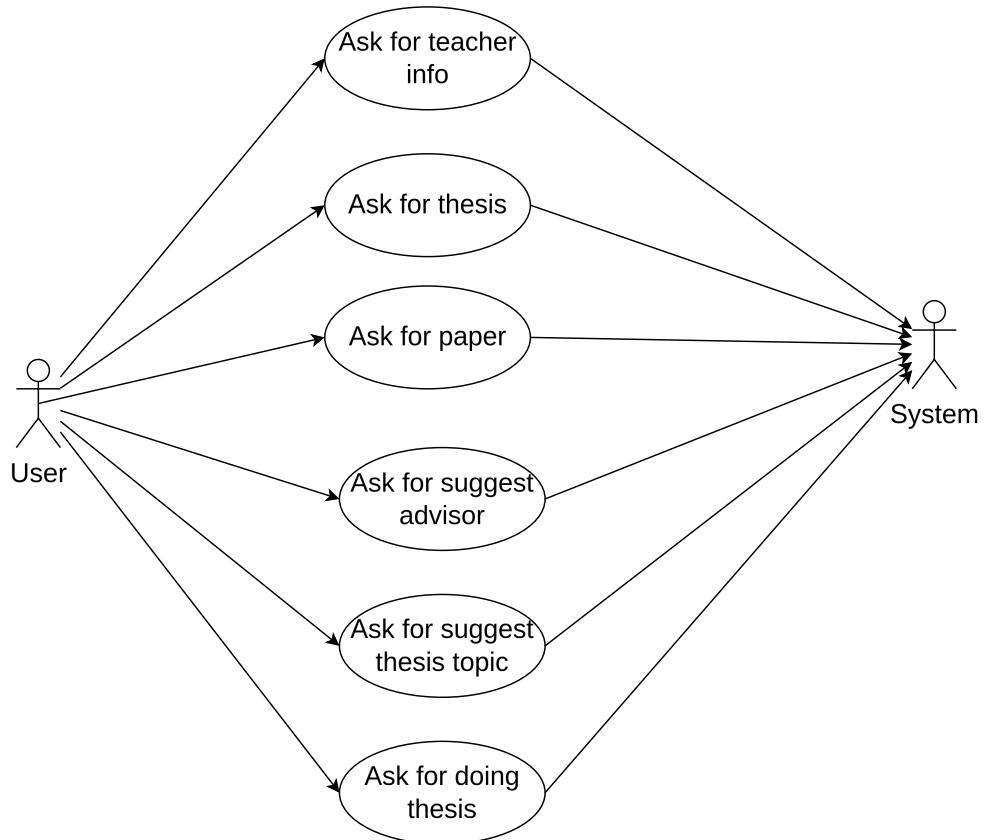
#### **3.1.14.3. Yêu cầu về khả năng**

Ngoài việc đánh giá chất lượng thu hồi và chất lượng sinh, một hệ thống RAG còn được đánh giá qua 4 khả năng thiết yếu thể hiện khả năng thích ứng và hiệu quả:

- Bên vững trước nhiễu (Noise robustness): đánh giá khả năng của mô hình trong việc quản lý các tài liệu nhiễu liên quan đến câu hỏi nhưng thiếu thông tin thực chất.
- Từ chối phủ định (Negative Rejection): Đánh giá khả năng nhận thức của mô hình trong việc kiểm chế không trả lời khi tài liệu được truy xuất không chứa kiến thức cần thiết để trả lời một câu hỏi.
- Tích hợp thông tin (Information Intergration): Đánh giá mức độ thành thạo của mô hình trong việc tổng hợp thông tin từ nhiều tài liệu để giải quyết các câu hỏi phức tạp.
- Bên vững trước phản thực (Counterfactual Robustness): Kiểm tra khả năng của mô hình trong việc nhận biết và bỏ qua những điểm không chính xác đã biết trong tài liệu, ngay cả khi được hướng dẫn về thông tin sai lệch tiềm ẩn.

## 3.2. Biểu đồ

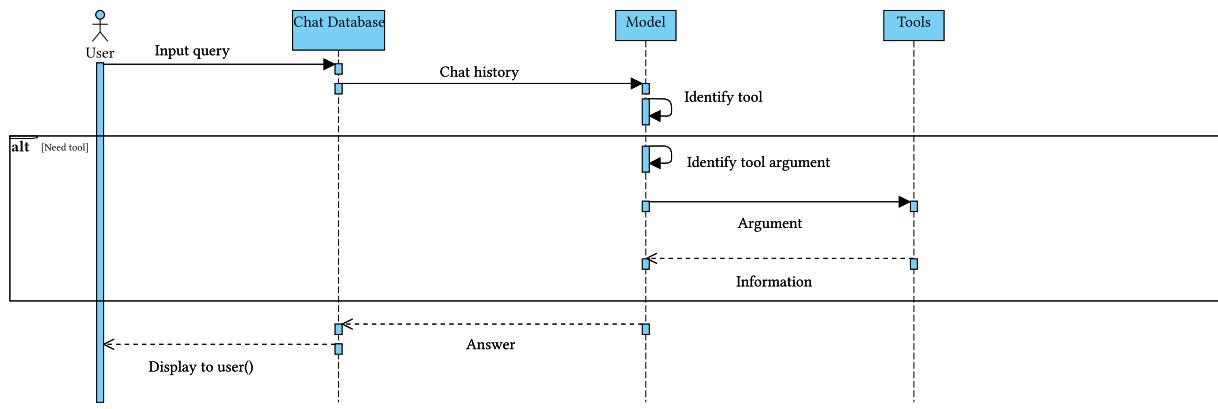
### 3.2.1. Biểu đồ ca sử dụng



Hình 27: Biểu đồ ca sử dụng

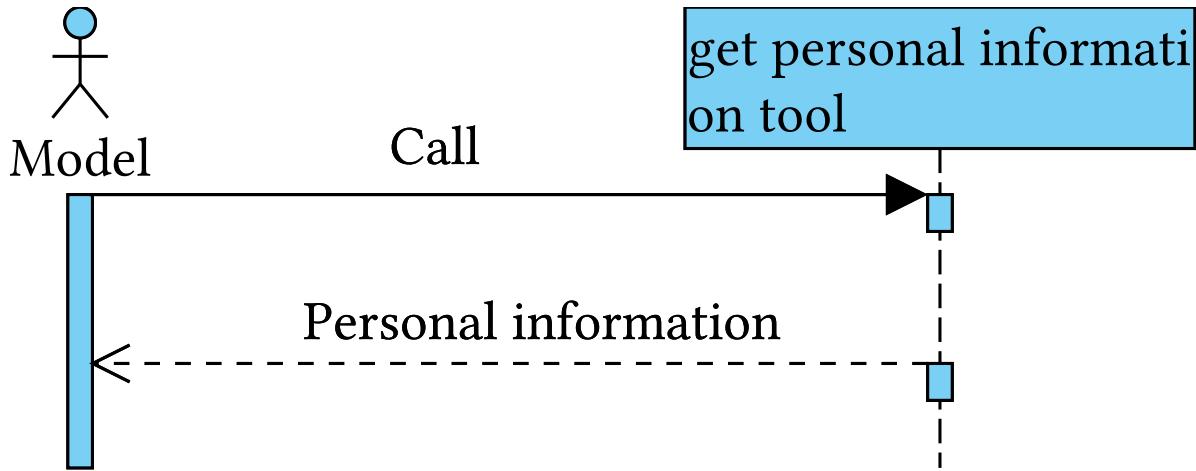
### 3.2.2. Biểu đồ tuần tự

#### 3.2.2.1. Hệ thống



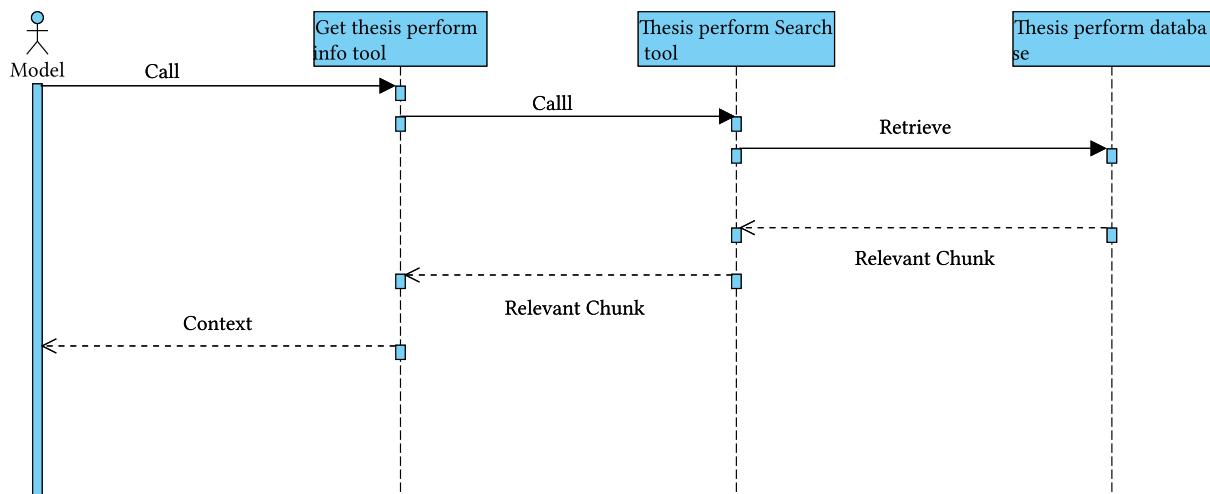
Hình 28: Biểu đồ tuần tự: Hệ thống

### 3.2.2.2. Công cụ truy xuất thông tin cá nhân



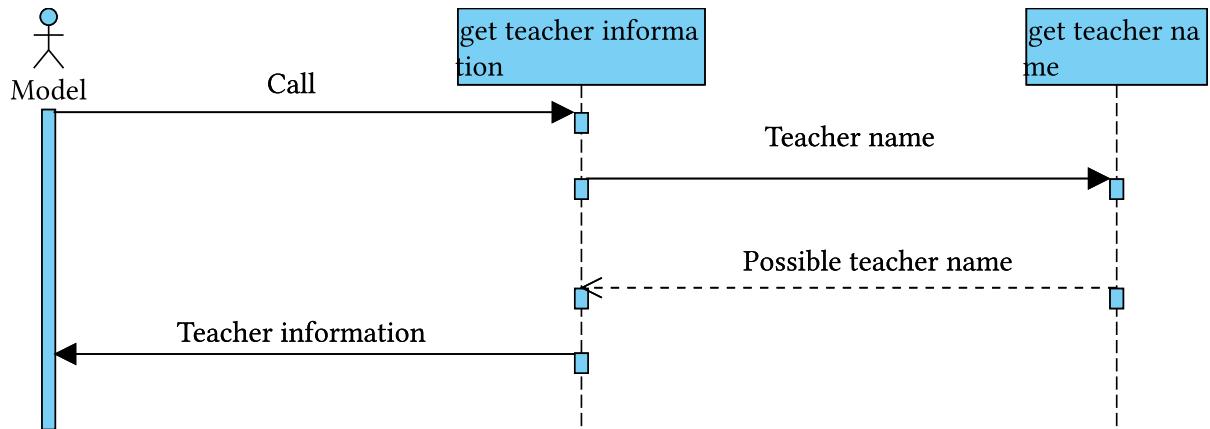
Hình 29: Biểu đồ tuần tự: Truy xuất thông tin cá nhân

### 3.2.2.3. Công cụ truy xuất thông tin thực hiện khóa luận



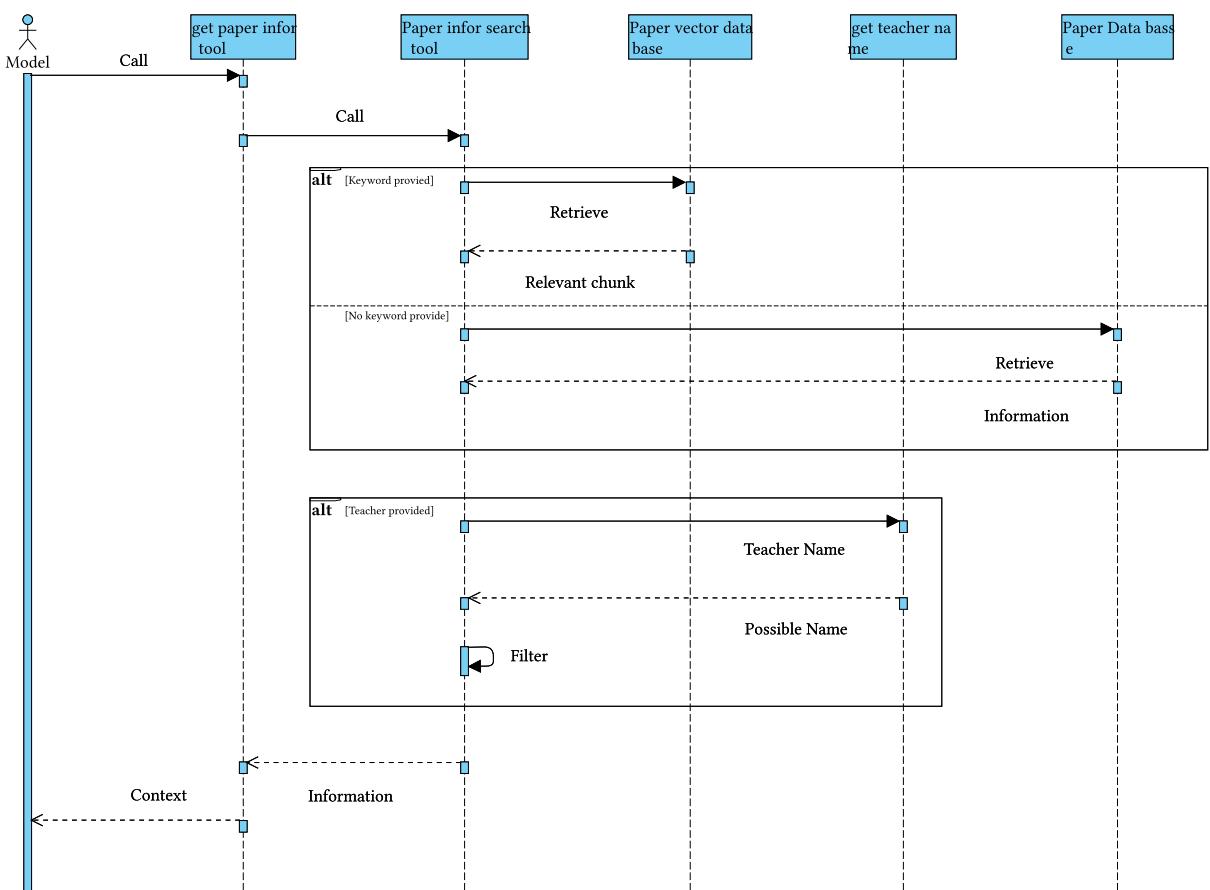
Hình 30: Biểu đồ tuần tự: Truy xuất thông tin thực hiện khóa luận

### 3.2.2.4. Công cụ tìm kiếm thông tin giáo viên



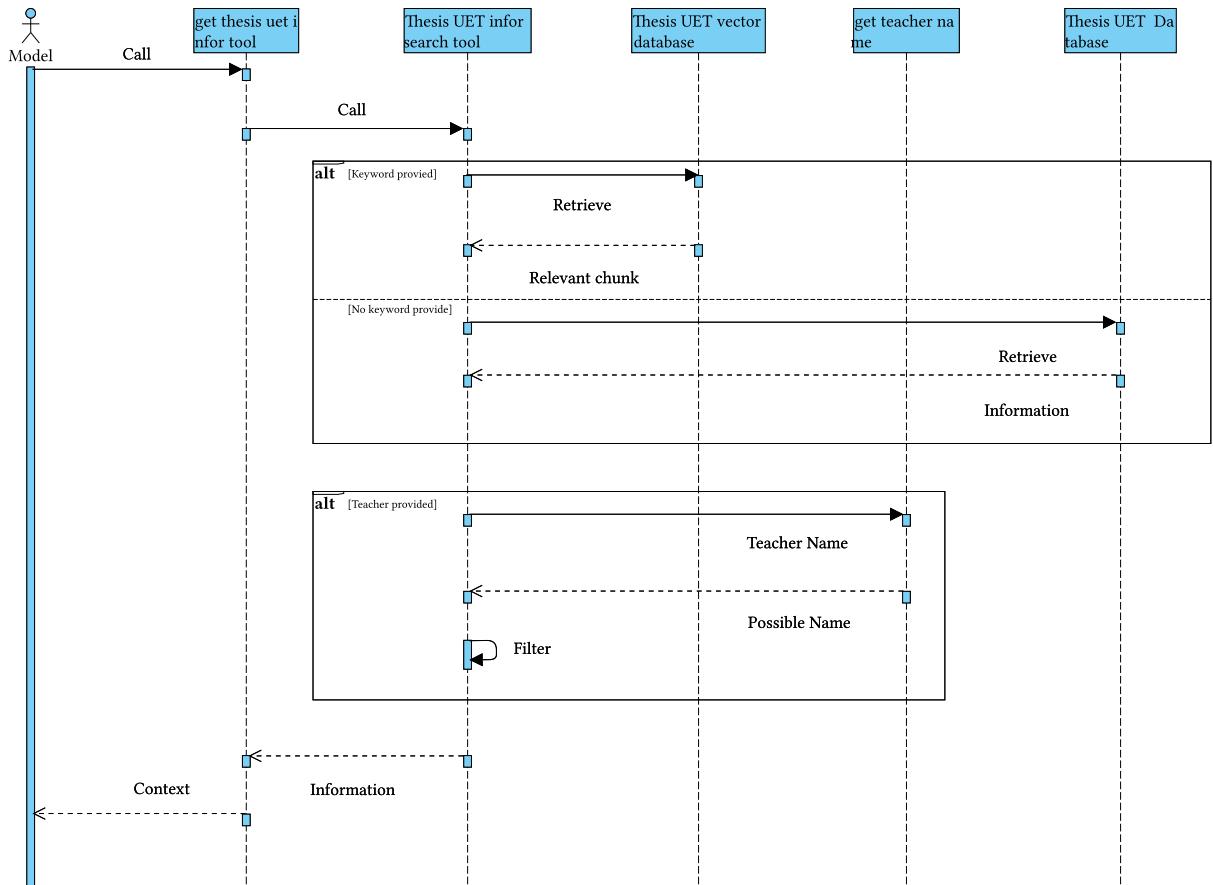
Hình 31: Biểu đồ tuần tự: Tìm kiếm thông tin giáo viên

### 3.2.2.5. Công cụ tìm kiếm bài báo



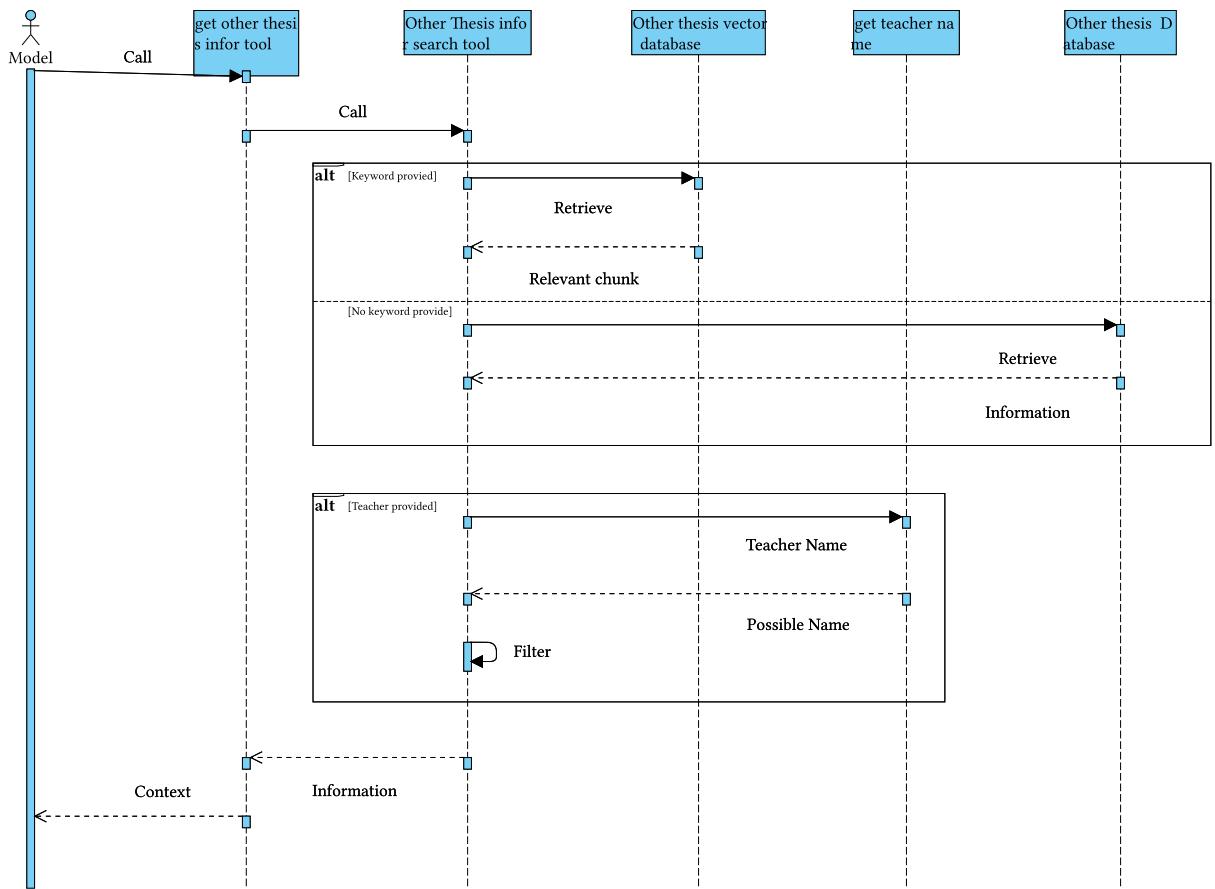
Hình 32: Biểu đồ tuần tự: Tìm kiếm bài báo

### 3.2.2.6. Công cụ tìm kiếm khóa luận UET



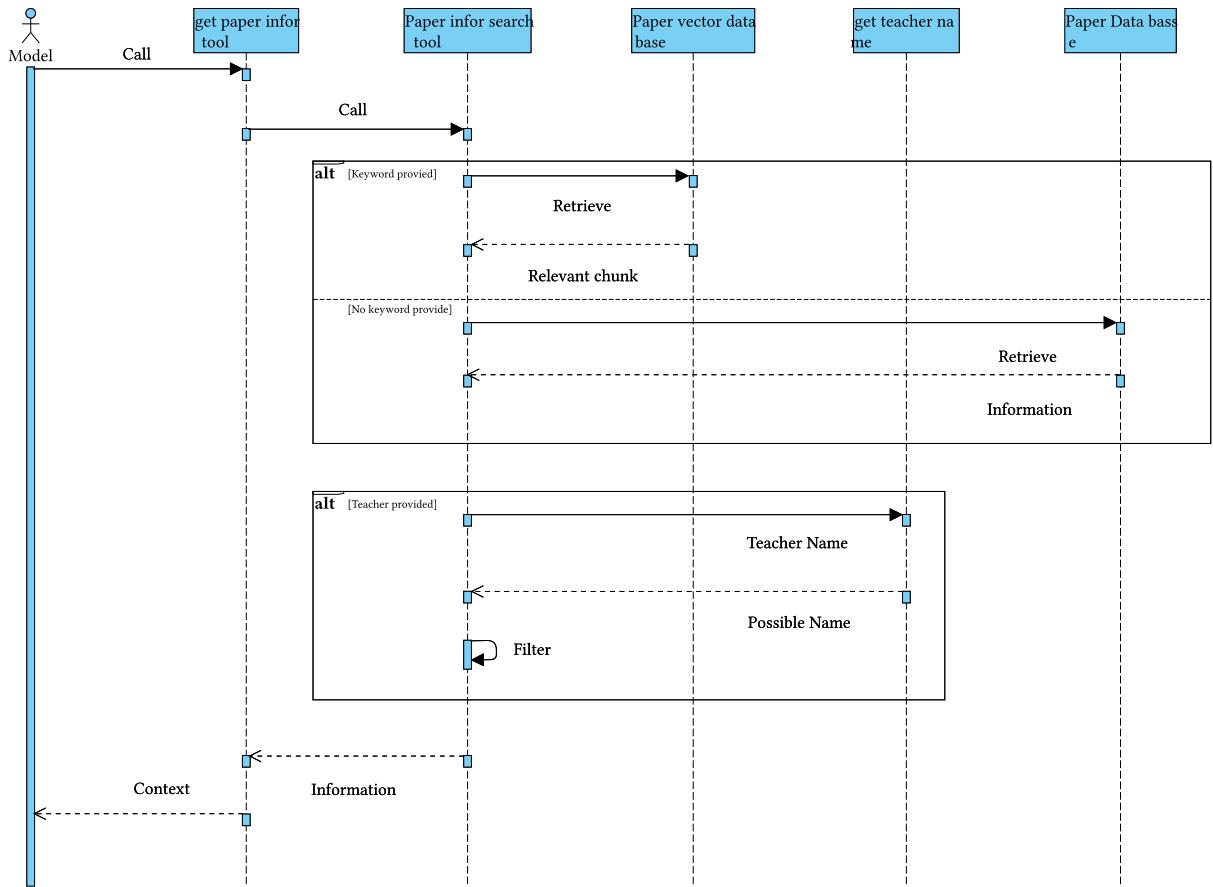
Hình 33: Biểu đồ tuần tự: Tìm kiếm khóa luận UET

### 3.2.2.7. Công cụ tìm kiếm khóa luận khác



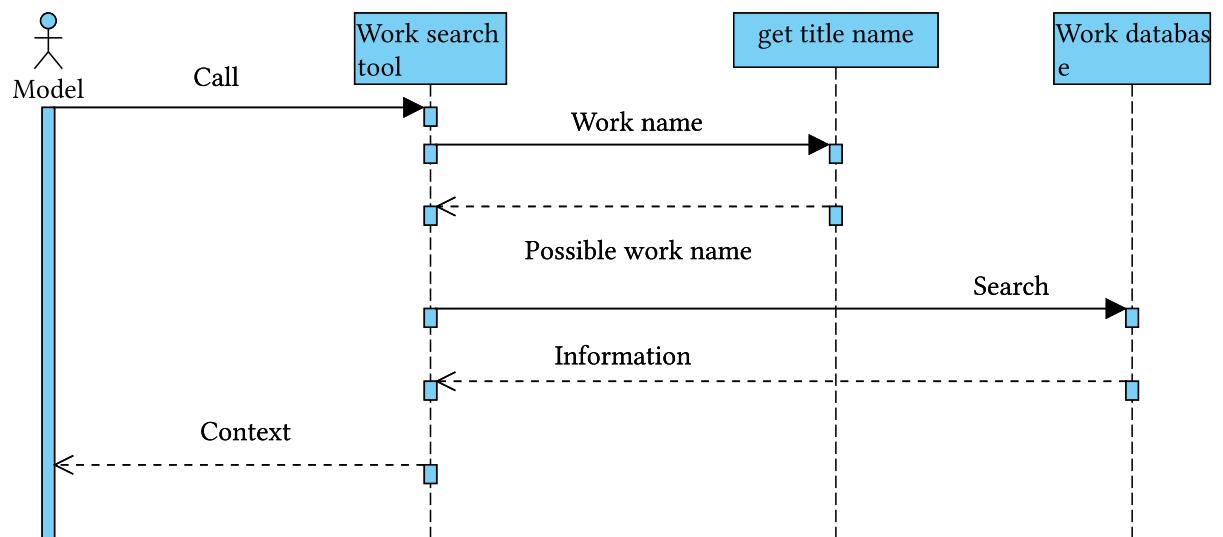
Hình 34: Biểu đồ tuần tự: Tìm kiếm khóa luận khác

### 3.2.2.8. Công cụ tìm kiếm công trình cụ thể



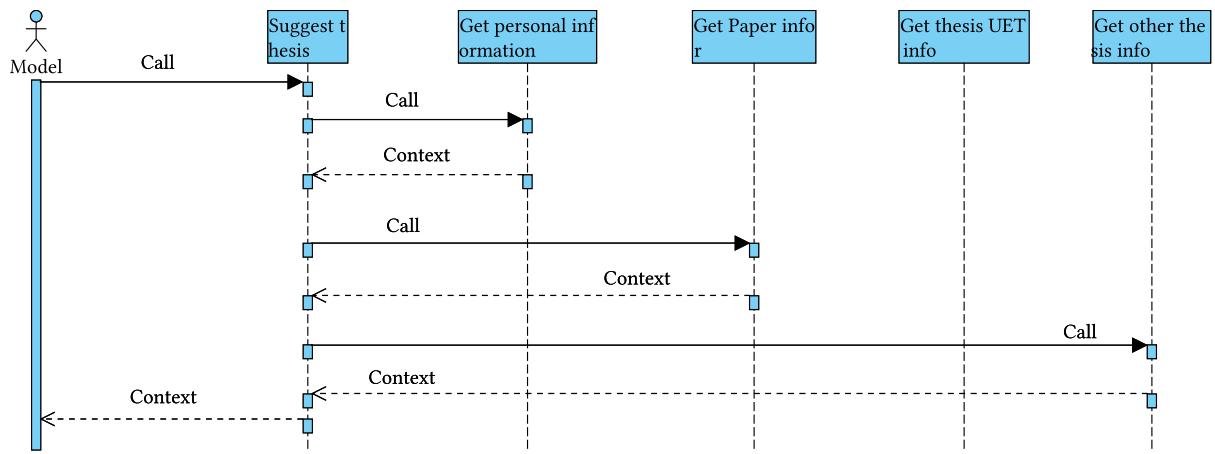
Hình 35: Biểu đồ tuần tự: Tìm kiếm công trình cụ thể

### 3.2.2.9. Công cụ tìm kiếm thầy hướng dẫn



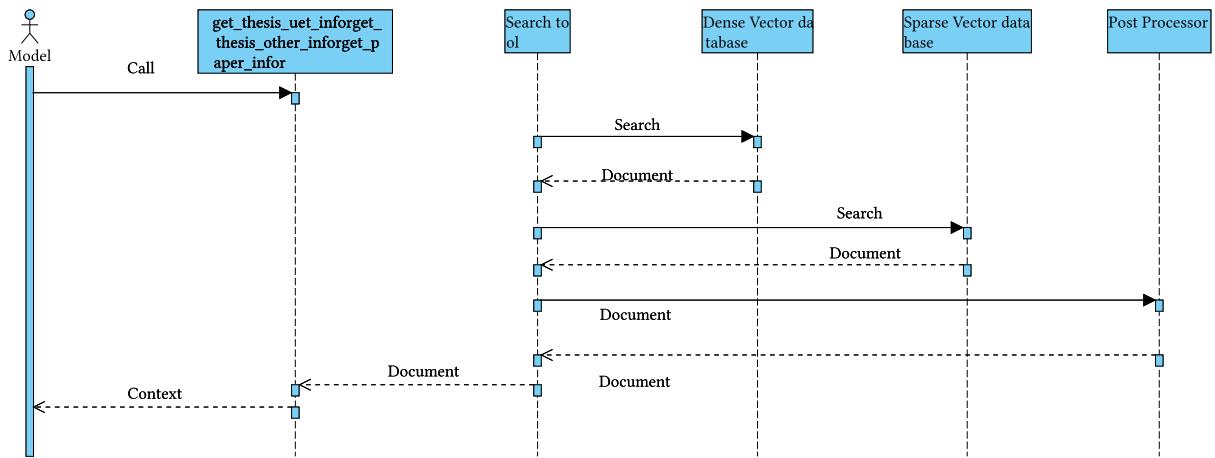
Hình 36: Biểu đồ tuần tự: Tìm kiếm thầy giáo hướng dẫn

### 3.2.2.10. Công cụ gợi ý chủ đề khóa luận tốt nghiệp



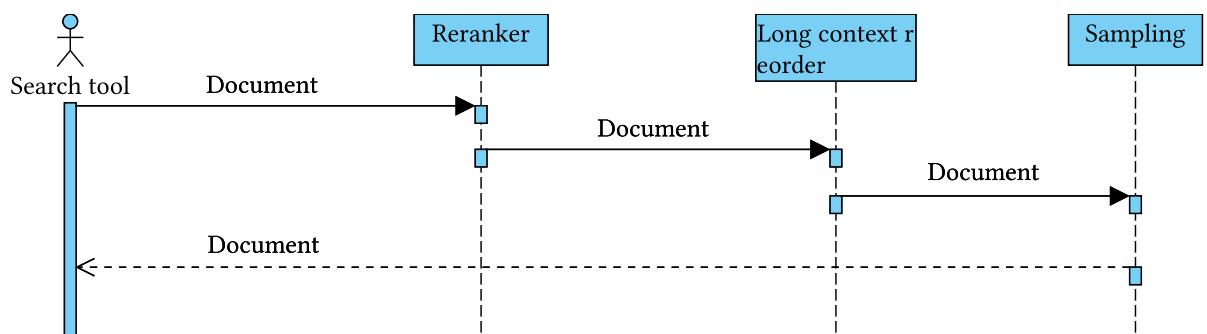
Hình 37: Biểu đồ tuần tự: Gợi ý khóa luận tốt nghiệp

### 3.2.2.11. Thu hồi ngữ cảnh



Hình 38: Biểu đồ tuần tự: Thu hồi ngữ cảnh

### 3.2.2.12. Hậu xử lý



Hình 39: Biểu đồ tuần tự: Hậu xử lý truy xuất

## 4. Thực nghiệm

### 4.1. Nền tảng

Ngôn ngữ lập trình: Python

Môi trường: Google Colaboratory

Phần cứng:

- GPU: Nvidia Tesla T4 16GB
- CPU: Intel Xeon CPU @2.20 GHz

Các thư viện chính:

- Selenium: Thư viện dùng cho thu thập dữ liệu
- Pandas: Thư viện dùng cho xử lý dữ liệu
- Llama-Index: Thư viện dùng xây dựng cơ sở dữ liệu
- theFuzz: Thư viện dùng thực hiện tìm kiếm mờ
- Langchain: Thư viện cho việc xây dựng RAG
- DeepEval: Thư viện hỗ trợ đánh giá LLM
- Gradio: Thư viện dùng để viết Demo

### 4.2. Hệ thống

#### 4.2.1. Dữ liệu

Dữ liệu	Nguồn	Kích thước	Thu thập
Thông tin thầy cô	<a href="#">FIT</a>	119 mẫu	Thủ công
Khóa luận UET	<a href="#">VNU Repository</a>	2470 mẫu	Crawler
Khóa luận HUST	<a href="#">Thư viện điện tử HUST</a>	16927 mẫu	Crawler
Khóa luận Oxford	<a href="#">Thư viện điện tử Oxford</a>	19435 mẫu	Crawler
Khóa luận Stanfoxd	<a href="#">Thư viện điện tử Stanford</a>	14911 mẫu	Crawler
Khóa luận Toronto	<a href="#">Thư viện điện tử Toronto</a>	36409 mẫu	Crawler
Khóa luận Berkeley	<a href="#">Thư viện điện tử Berkeley</a>	10000 mẫu	Crawler
Bài báo khoa học	Google scholar	1560 mẫu	Crawler
Thông tin cá nhân	Tự tạo	1 mẫu	Tự tạo
Thông tin thực hiện khóa luận	<a href="#">FIT</a>	7 trang	Thủ công

Bảng 5: Tổng quan dữ liệu

Nhận xét:

- Dữ liệu giáo viên chưa đầy đủ
- Danh sách giáo viên giới hạn ở khoa công nghệ thông tin
- Dữ liệu bài báo chưa đầy đủ: thông tin công trình của một số giáo viên không được cập nhật
- Dữ liệu khóa luận UET có nhiều nhiễu
- Nguồn dữ liệu khóa luận UET chỉ lưu trữ 1 phần nhỏ khóa luận so với thực tế

Xử lý:

- Xóa các mẫu có phần mô tả quá ngắn.

- Đối với dữ liệu khóa luận và bài báo, tạo cột clean\_title là giá trị chuẩn hóa của tên (đưa về kí tự latin, viết thường, xóa dấu câu và kí tự đặc biệt).
- Đối với dữ liệu thông tin giáo viên, tạo cột name\_variation gồm các biến thể của tên. Ví dụ: Phạm Thị Kiều Trang sẽ có các biến thể: “pham thi kieu trang”, “trang ptk”, “pham tkt”, ...
- Thực hiện làm sạch thủ công với dữ liệu khóa luận UET. Ví dụ: “Nghiên cứu các giao thức định tuyến trong mạng truyền thông : Luận văn ThS. Công nghệ thông tin : 1.01.10” thành “Nghiên cứu các giao thức định tuyến trong mạng truyền thông”

#### 4.2.2. Phân đoạn

Đối với khóa luận, bài báo, mỗi đoạn được tạo bằng công thức:

$$\text{Chunk} = \text{"document: "} + \text{Tiêu đề} + \text{Mô tả}$$

Đối với văn bản chứa thông tin thực hiện khóa luận thực hiện phân đoạn theo câu. Với mỗi đoạn thu được thêm xâu “document: “ vào trước.

#### 4.2.3. Cơ sở dữ liệu vector

Hệ quản trị cơ sở dữ liệu vector: Chroma  
Phương pháp đánh chỉ mục: HNSW

#### 4.2.4. Nhúng

Mô hình: intfloat/multilingual-e5-small

Ngôn ngữ: Đa ngôn ngữ

Kích thước vector: 384

Giới hạn đầu vào: 256 token

Yêu cầu đầu vào: Yêu cầu thêm “query: “ vào trước văn bản truy xuất và “document: “ vào trước văn bản nhúng trong cơ sở dữ liệu vector.

#### 4.2.5. LLM

Mô hình LLM được sử dụng là GPT-3 VÀ GPT-4 do OpenAI cung cấp. GPT-3 là mô hình phổ biến nhất hiện nay. Đây không phải là mô hình tốt nhất nhưng là mô hình trả lời tự nhiên nhất. Các mô hình này có thể được sử dụng qua API với chi phí được mô tả trong Bảng 6

	Giá token đầu vào	Giá token đầu ra
GPT-3-turbo-instruct	\$1.50 / 1 triệu tokens	\$2.00 / 1 triệu tokens
GPT-4	\$30.00 / 1 triệu tokens	\$60.00 / 1 triệu tokens

Bảng 6: Giá sử dụng API của OpenAI

#### 4.2.6. Truy xuất

Top k = 100

#### 4.2.7. Hậu truy xuất

Mô hình rerank: BGE/Reranker-v2-m3

Ngôn ngữ: Đa ngôn ngữ

Giới hạn đầu vào: 8192 token

Top k: Trả về 20 văn bản với độ tương đồng cao nhất

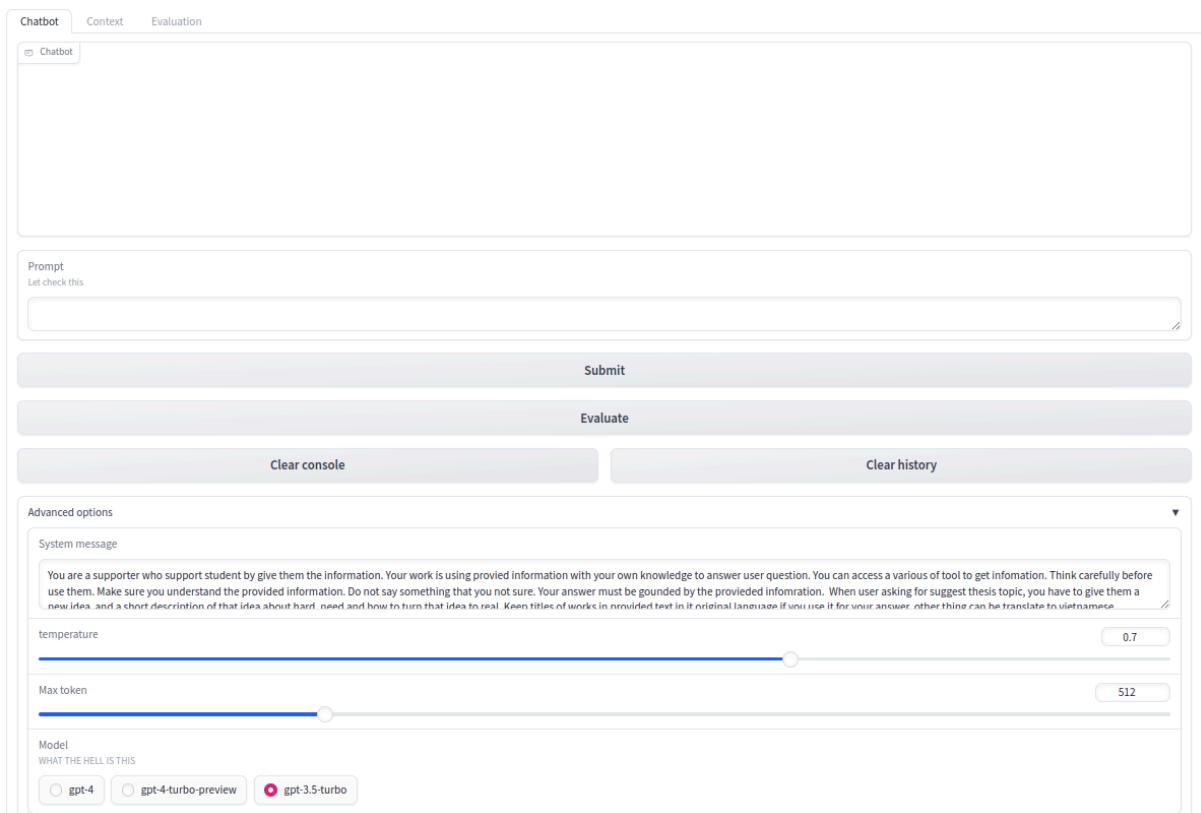
Nguồng lọc: 0.01

#### 4.2.8. Lịch sử trò chuyện

Lưu trữ 10 cặp câu hỏi và câu trả lời gần nhất.

#### 4.2.9. Bản mẫu

Bản mẫu được viết bằng thư viện gradio, có thể triển khai trên web và gửi liên kết đến những người dùng khác để họ sử dụng hệ thống. Bản mẫu thể hiện đầy đủ tinh thần của RAG từ khả năng trả lời câu hỏi cho đến đưa ra tài liệu được sử dụng và đánh giá câu trả lời.



Hình 40: Giao diện bản mẫu

Giao diện của bản mẫu cho phép:

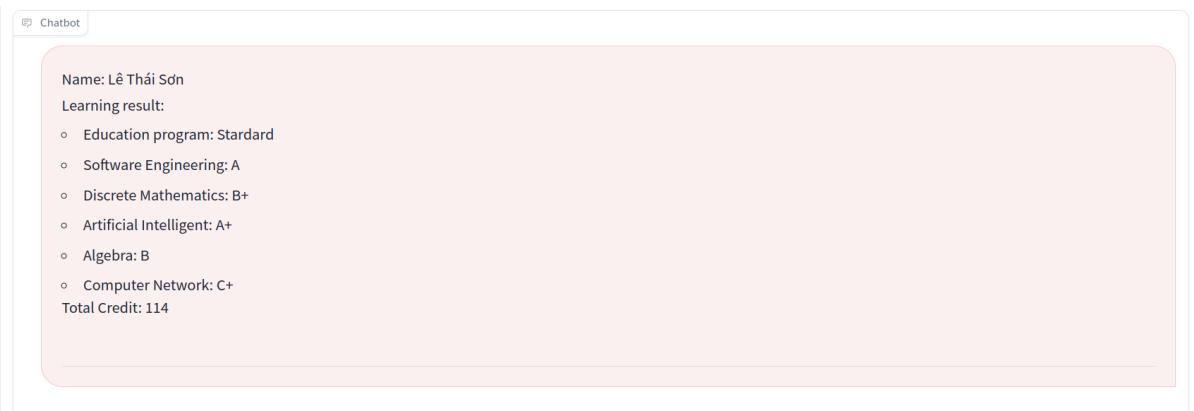
- Nhập đầu vào
- Hiển thị lịch sử trò chuyện
- Streaming
- Xóa màn hình
- Xóa lịch sử trò chuyện
- Xem ngữ cảnh được truy xuất
- Đánh giá kết quả sinh
- Lựa chọn mô hình sinh
- Tùy chỉnh số lượng token tối đa
- Tùy chỉnh Temperature

## 4.3. Kiểm thử

### 4.3.1. Tìm kiếm thông tin cá nhân

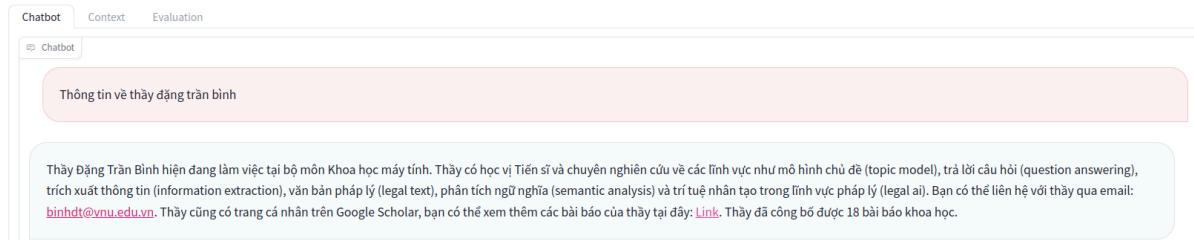


Hình 41: Kết quả: Tìm kiếm thông tin cá nhân

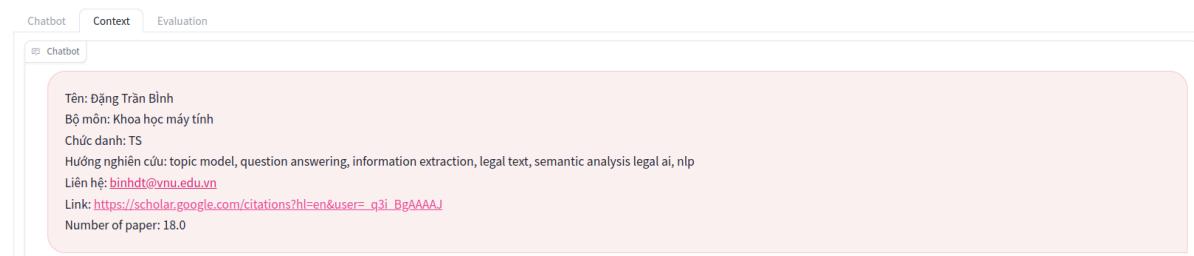


Hình 42: Ngữ cảnh: Tìm kiếm thông tin cá nhân

### 4.3.2. Tìm kiếm thông tin giáo viên



Hình 43: Kết quả: Tìm kiếm thông tin giáo viên



Hình 44: Ngữ cảnh: Tìm kiếm thông tin giáo viên

### 4.3.3. Hỏi về thực hiện khóa luận

Chatbot Context Evaluation

Ngày nộp khóa luận là ngày bao nhiêu

Ngày nộp khóa luận tại trường là ngày 25/05/2023. Trước ngày này, sinh viên cần nộp quyền bia cứng và bản điện tử của khóa luận sau khi đã chỉnh sửa và có xác nhận đã chỉnh sửa của cán bộ hướng dẫn. Sau khi hết thời hạn nếu sinh viên không nộp đúng quy định, họ sẽ nhận điểm F. Đề nghị các sinh viên và đơn vị thực hiện đúng thời gian quy định để hoàn thành quy trình nộp khóa luận.

Hình 45: Kết quả: Hỏi về thực hiện khóa luận

Chatbot Context Evaluation

- Đề xuất số lượng Hội đồng chấm ĐA/KLTN và giới thiệu nhân sự, đề nghị Nhà trường ra quyết định thành lập Hội đồng chấm ĐA/KLTN, Hội đồng thẩm định kết quả chấm ĐA/KLTN cấp Khoa/Viện (mỗi Hội đồng gồm 5 thành viên trở lên).

Hạn cuối  
14/04/2023  
(Thứ sáu)

- Bản trên giấy (qua Phòng HC-QT).

- Bản điện tử gửi về địa chỉ email: [dungpn09@vnu.edu.vn](mailto:dungpn09@vnu.edu.vn)

3  
Phòng ĐT rà soát các điều kiện, trình Ban Giám hiệu ban hành các Quyết định Duyệt danh sách sinh viên bảo vệ tại Hội đồng, thành lập Hội đồng chấm ĐA/KLTN, Hội đồng thẩm định cấp Khoa/Viện.

Hạn cuối  
21/04/2023  
(Thứ sáu)

4  
Các đơn vị cử cán bộ đến Phòng ĐT nhận các giấy tờ, văn bản liên quan đến buổi bảo vệ ĐA/KLTN.  
25/04/2023

5  
Các đơn vị thu ĐA/KLTN (có xác nhận của cán bộ hướng dẫn), bัน tóm tắt ĐA/KLTN và phát cho sinh viên:  
- Giấy mời phản biện, Bản nhận xét hướng dẫn/phản biện.  
26/04 , 28/04/2023  
(Thứ tư , Thứ sáu)

Hình 46: Ngữ cảnh: Hỏi về thực hiện khóa luận

### 4.3.4. Tìm kiếm khóa luận theo chủ đề

Chatbot Context Evaluation

Một số khóa luận liên quan tới học máy

Dưới đây là danh sách các khóa luận liên quan đến học máy:

1. Echocardiography segmentation using machine learning algorithms

- Mô tả: Phân đoạn siêu âm tim bằng thuật toán học máy. Nghiên cứu về phân vùng tâm thất trái từ video siêu âm tim sử dụng tiến bộ trong lĩnh vực học sâu.
- Advisor: Trần Quốc Long, Phạm Huy Hiệu

2. Sử dụng Active Learning trong việc lựa chọn dữ liệu gần nhau cho bài toán Speech Recognition

- Mô tả: Đánh giá mô hình nhận dạng tiếng nói và đề xuất phương pháp lựa chọn dữ liệu "quan trọng" cho việc gần nhau dữ liệu bài toán nhận dạng tiếng nói sử dụng phương pháp học chủ động.
- Advisor: Nguyễn Xuân Hoài

3. On the problem of synthesizing gestures for 3D virtual human

- Mô tả: Nghiên cứu về học tăng cường và xây dựng hệ thống tự học sử dụng thuật toán học tăng cường - Q-learning.
- Advisor: Bùi Thế Duy

4. Deep learning-based approach for water crystal classification

- Mô tả: Phân loại tinh thể nước dựa trên học sâu và cung cấp bộ dữ liệu chất lượng cao để dạy thuật toán hàm ánh xạ.
- Advisor: Trần Quốc Long, Andres Frederic

5. Machine learning-based extraction of semantic relations from biomedical literature

- Mô tả: Trích xuất mối quan hệ ngữ nghĩa trong văn bản y sinh dựa trên học máy

Hình 47: Kết quả: Tìm kiếm khóa luận theo chủ đề

Chatbot Context Evaluation

Chatbot

Here is the list of UET thesis that related to học máy:

document: Xây dựng mô hình học máy hỗ trợ chẩn đoán bước đầu bệnh mạch vành dựa trên ảnh chụp SPECT tưới máu cơ tim. Mục tiêu của đề tài này hướng đến việc giải quyết bài toán xây dựng một mô hình giúp chẩn đoán bệnh mạch vành dựa trên ảnh chụp SPECT tưới máu cơ tim. Mô hình sẽ nhận đầu vào là các ảnh chụp SPECT và có thể kết hợp với các chẩn đoán lâm sàng ban đầu và đưa ra xác suất dự đoán về khả năng mắc bệnh động mạch vành. Các phương pháp, mô hình khác nhau sẽ lần lượt được đánh giá để tìm ra giải pháp tối ưu cho bài toán.

Advisor: Trần Quốc Long  
Nguyễn Chí Thành  
Keywords: Học máy, Bệnh mạch vành -- Chẩn đoán, Học máy

document: Machine learning-based extraction of semantic relations from biomedical literature = Trích xuất mối quan hệ ngữ nghĩa trong văn bản y sinh dựa trên học máy. In this Dissertation, we consider Relation Extraction as two text mining sub-tasks, i.e., Named Entity Recognition (NER) and Relation Classification (RC). The task of biomedical named entity recognition (NER) seeks to locate named entities from free-form biomedical text and classify them into a set of pre-defined categories/types such as gene/protein, phenotype, disease, and chemical, or 'none-of-the-above'. Relation classification (RC) is the task of discovering semantic connections between biomedical entities

Advisor: Collier, Nigel  
Đặng Thanh Hải  
Keywords: Hệ thống thông tin, Công nghệ thông tin, Học máy

document: Một số mô hình học máy trong phân loại câu hỏi. Trình bày tổng quan về phân loại câu hỏi, giới thiệu về hệ thống hỏi đáp, bài toán phân loại câu hỏi, cách tiếp cận giải quyết bài toán, tổng quan về các tiếp cận học máy như: biểu diễn câu hỏi, phân lớp câu hỏi, các đặc trưng câu hỏi. Đồng thời trình bày về 3 bộ phân loại thường được sử dụng: Naïve Bayes, K-lắng giềng gần, Máy vector hỗ trợ và liệt kê một số bộ phân loại khác. So sánh hiệu suất phân loại của các bộ phân loại đó dựa trên kết quả tham khảo. Từ đó Áp dụng bộ phân loại SVM thực hiện thí nghiệm trên tập dữ liệu UIUC, lựa chọn đặc trưng bag-of-word. Nhận xét kết quả trả về

Advisor: Lê Hồng Phương  
Keywords: Máy tính, Mô hình học máy, phân loại câu hỏi

Hình 48: Ngữ cảnh: Tìm kiếm khóa luận theo chủ đề

#### 4.3.5. Tìm kiếm khóa luận theo giáo viên hướng dẫn

Chatbot Context Evaluation

Chatbot

Các khóa luận được hướng dẫn bởi thầy Hà Quang Thụy

Dưới đây là danh sách một số khóa luận được hướng dẫn bởi thầy Hà Quang Thụy:

- "Phát triển kỹ thuật chắt lọc tri thức trong học suốt đời đối với miền dữ liệu văn bản" - Đề xuất thuật toán và MH CDSĐ miền gần CD-AMC phát triển từ MH CDSĐ AMC của Z. Chen và B. Liu [18] với giải pháp chắt lọc tri thức phải-di-cùng và tri thức không-thể-di-cùng chỉ từ các miền quá khứ gần thay vì từ tất cả các miền quá khứ [ChamNT1].
- "Các phương pháp trừu tượng hóa mô hình quy trình kinh doanh và thực nghiệm" - Tổng quan về quản lý mô hình quy trình kinh doanh, sự cần thiết của mô hình hóa quy trình kinh doanh và các hướng tiếp cận, phương pháp giải quyết bài toán trừu tượng hóa mô hình quy trình kinh doanh, giới thiệu các ngôn ngữ mô hình kinh doanh.
- "Nhận dạng thực thể định danh từ văn bản ngắn tiếng Việt và đánh giá thực nghiệm" - Xây dựng mô hình CRFs để nhận dạng thực thể trong văn bản Tiếng Việt áp dụng học suốt đời.
- "Nghiên cứu bộ đo tin cậy và áp dụng vào hệ thống tư vấn" - Trình bày về sự tin cậy, tin cậy trong mạng xã hội và độ đo tin cậy. Giới thiệu ch

Hình 49: Kết quả: Tìm kiếm khóa luận theo giáo viên hướng dẫn

Chatbot Context Evaluation

Chatbot

Here is the list of UET thesis that related to and were advised by Hà Quang Thuy:

document: Phát triển kỹ thuật chắt lọc tri thức trong học suốt đời với miền dữ liệu văn bản. Đề xuất thuật toán và MH CDSĐ miền gần CD-AMC phát triển từ MH CDSĐ AMC của Z. Chen và B. Liu [18] với giải pháp chắt lọc tri thức phải-di-cùng và tri thức không-thể-di-cùng chỉ từ các miền quá khứ gần thay vì từ tất cả các miền quá khứ [ChamNT1]. Đề xuất hai cách thức xác định miền gần đối với miền dữ liệu hiện tại (dựa trên tập từ - tập chủ đề trong CD-AMC và dựa trên các bộ phân lớp văn bản quá khứ trong CCDAMC (Classifier-based CD-AMC)), áp dụng vào tác vụ phân lớp đa nhân tiếng Việt [ChamNT1] và tác vụ phân lớp quan điểm tiếng Anh [ChamNT2], đồng thời, tiến hành đánh giá thực nghiệm các mô hình đề xuất. Hơn nữa, luận án đã tiến hành kiểm định thống kê một mẫu theo phân phối-t (one-sample t test) về kỳ vọng quần thể giả thuyết khi chưa biết độ lệch chuẩn quần thể để minh chứng mô hình đề xuất thực sự có hiệu năng cao hơn so với AMC [ChamNT1]. Đề xuất MH CDSĐ miền gần hướng đích TCD-AMC (Targeted CD - AMC) kết hợp MH CDSĐ miền gần CD-AMC của luận án với mô hình chủ đề hướng đích TTM (Targeted Topic Model) của S. Wang và cộng sự [98] và áp dụng vào tác vụ phân lớp đa nhân trích xuất khía cạnh trong khai phá quan điểm tiếng Việt [ChamNT3]. - Đề xuất mô hình HMSĐ chắt lọc tri thức tham số mô hình học sâu BiLSTM-KD-NER cho tác vụ nhận dạng thực thể y sinh tiếng Việt và tiến hành thực nghiệm kiểm chứng, đánh giá đề xuất này [ChamNT4].

Advisor: Hà, Quang Thuy

Keywords: Kỹ thuật chắt lọc tri thức; Miền dữ liệu văn bản; Dữ liệu

document: Các phương pháp trừu tượng hóa mô hình quy trình kinh doanh và thực nghiệm. Tìm hiểu chung về trừu tượng hóa mô hình quy trình kinh doanh. Ở chương đầu tiên mở đầu này nêu tổng quan về quản lý mô hình quy trình kinh doanh, sự cần thiết của mô hình hóa quy trình kinh doanh và các hướng tiếp cận, phương pháp giải quyết bài toán trừu tượng hóa mô hình quy trình kinh doanh, giới thiệu các ngôn ngữ mô hình kinh doanh. Giới thiệu một số phương pháp trừu tượng hóa mô hình quy trình kinh doanh, cụ thể là giới thiệu các quy tắc trừu tượng cũng như các nguyên tắc chuyển đổi mô hình và ý tưởng về mô hình giải bài toán trong luận văn. Trình bày bài toán trừu tượng hóa mô hình quy trình kinh doanh cùng mô hình giải quyết và thực nghiệm giải quyết bài toán Trừu tượng hóa mô hình quy trình kinh doanh. Chương này là chương chính của luận văn, giới thiệu bài toán cụ thể mà luận văn cần giải quyết, sau đó là đưa ra mô hình giải quyết bài toán.

Advisor: Hà, Quang Thuy

Keywords: Công nghệ thông tin, Khai thác dữ liệu, Trí tuệ nhân tạo

document: Nhận dạng thực thể định danh từ văn bản ngắn tiếng Việt và đánh giá thực nghiệm. Xây dựng mô hình CRFs để nhận dạng thực thể trong văn bản Tiếng Việt áp dụng học suốt đời. Tiến hành đánh giá thực nghiệm để so sánh giữa nhiều trường hợp, từ đó chứng minh được áp dụng học suốt đời có thể làm tăng hiệu suất của việc học cũng như chỉ ra vai trò quan trọng của dữ liệu có được thông

Hình 50: Ngữ cảnh: Tìm kiếm khóa luận theo giáo viên hướng dẫn

#### 4.3.6. Tìm kiếm khóa luận theo chủ đề của giáo viên hướng dẫn

Chatbot Context Evaluation

Chatbot

Các khóa luận liên quan tới học máy và được hướng dẫn bởi thầy hà quang thụy

Dưới đây là danh sách các luận văn tại UET liên quan đến học máy và được hướng dẫn bởi thầy Hà Quang Thụy:

- Tên Luận Văn:** Transductive support vector machines for cross-lingual sentiment classification.
  - Tóm tắt:** Nghiên cứu về bài toán phân loại cảm xúc đa ngôn ngữ sử dụng kỹ thuật học máy.
  - Từ khóa:** Khoa học máy tính, Công nghệ thông tin, Dữ liệu, Ngôn ngữ
  - Hướng dẫn:** Hà Quang Thụy
- Tên Luận Văn:** Advanced Deep Learning Models and Applications in Semantic Relation Extraction.
  - Tóm tắt:** Nghiên cứu về các mô hình học sâu tiên tiến và ứng dụng trong trích chọn quan hệ ngữ nghĩa.
  - Từ khóa:** Khoa học máy tính, Trí tuệ nhân tạo, Công nghệ thông tin
  - Hướng dẫn:** Hà Quang Thụy
- Tên Luận Văn:** Advanced Deep learning Methods and Applications in Open-domain Question Answering.
  - Tóm tắt:** Đề xuất phương pháp học biểu diễn tài liệu dựa vào thông tin câu hỏi và áp dụng vào bài toán hệ hỏi-dáp minden mở.
  - Từ khóa:** Công nghệ thông tin, Khoa học máy tính, Trí tuệ nhân tạo
  - Hướng dẫn:** Hà Quang Thụy

Bạn có thể xem chi tiết từng luận văn để biết thêm thông tin.

Hình 51: Kết quả: Tìm kiếm khóa luận theo chủ đề của giáo viên hướng dẫn

Chatbot Context Evaluation

Chatbot

Các khóa luận liên quan tới học máy và được hướng dẫn bởi thầy Hà Quang Thụy

Dưới đây là danh sách các luận văn tại UET liên quan đến học máy và được hướng dẫn bởi thầy Hà Quang Thụy:

1. **Tên Luận Văn:** Transductive support vector machines for cross-lingual sentiment classification.
  - **Tóm tắt:** Nghiên cứu về bài toán phân loại cảm xúc đa ngôn ngữ sử dụng kỹ thuật học máy.
  - **Từ khóa:** Khoa học máy tính, Công nghệ thông tin, Dữ liệu, Ngôn ngữ
  - **Hướng dẫn:** Hà Quang Thụy
2. **Tên Luận Văn:** Advanced Deep Learning Models and Applications in Semantic Relation Extraction.
  - **Tóm tắt:** Nghiên cứu về các mô hình học sâu tiên tiến và ứng dụng trong trích chọn quan hệ ngữ nghĩa.
  - **Từ khóa:** Khoa học máy tính, Trí tuệ nhân tạo, Công nghệ thông tin
  - **Hướng dẫn:** Hà Quang Thụy
3. **Tên Luận Văn:** Advanced Deep learning Methods and Applications in Open-domain Question Answering.
  - **Tóm tắt:** Đề xuất phương pháp học biểu diễn tài liệu dựa vào thông tin câu hỏi và áp dụng vào bài toán hệ hỏi-dáp miền mở.
  - **Từ khóa:** Công nghệ thông tin, Khoa học máy tính, Trí tuệ nhân tạo
  - **Hướng dẫn:** Hà Quang Thụy

Bạn có thể xem chi tiết từng luận văn để biết thêm thông tin.

Hình 52: Ngũ cảnh: Tìm kiếm khóa luận theo chủ đề của giáo viên hướng dẫn

#### 4.3.7. Tìm kiếm bài báo theo chủ đề

Chatbot Context Evaluation

Chatbot

Các bài báo liên quan tới học máy

Dưới đây là danh sách các bài báo liên quan đến học máy:

1. **Phân loại câu hỏi sử dụng nhiều loại đặc trưng**
  - Tác giả: Van Tu Nguyen, Anh Cuong Le, Ha Nam Nguyen
2. **Examining Students' Learning Styles Impacted on Learning Outcome in the MOOC Course: A Case Study**
  - Tác giả: Viet Anh Nguyen
3. **Một thuật toán hiệu quả dựa trên giải thuật tối ưu đan kiến giải bài toán r/p trung tâm**
  - Tác giả: Duc Quang Vu, Xuan Huan Hoang, Thanh Mai Do
4. **Một chỉ số đánh giá số cụm mới cho thuật toán C-means mờ**
  - Tác giả: Trung Duc Nguyen, Xuan-Huan Hoang
5. **Nâng cao hiệu năng xử lý giao tác với mô hình CSDL trong bộ nhớ**
  - Tác giả: Bá Duy Vũ, Phương Hạnh Vũ
6. **Góp bàn về chiến lược phát triển trí tuệ nhân tạo của Việt Nam**
  - Tác giả: Nguyễn Thanh Thủy, Hà Quang Thụy, Phan Xuân Hiếu, Nguyễn Trí Thành
7. **Giải pháp xếp hạng và tính toán song song trên nền tảng Apache Spark**
  - Tác giả: Đức Nguyễn Đông
8. **Phân Loại Mã Độc Dựa Trên Các Kỹ Thuật N-GRAM Và Học Máy**

Hình 53: Kết quả: Tìm kiếm bài báo theo chủ đề

Chatbot Context Evaluation

Chatbot

Here is the list of paper that related to học máy:

document: Phân loại câu hỏi sử dụng nhiều loại đặc trưng. Phân loại câu hỏi là một thành phần quan trọng trong các hệ thống hồi đáp tự động. Nhiệm vụ chính của phân loại câu hỏi là dự báo kiểu thực thể của câu trả lời của các câu hỏi viết bằng ngôn ngữ tự nhiên. Phân loại câu hỏi có thể được thực hiện bằng các tiếp cận khác nhau như: tiếp cận dựa trên luật, tiếp cận dựa trên học máy. Các đặc trưng khác nhau về từ vựng, cú pháp và ngữ nghĩa có thể được trích xuất tự động từ các câu hỏi để phục vụ việc phân loại. Trong nghiên cứu này chúng tôi kết hợp các đặc trưng về từ vựng, cú pháp, ngữ nghĩa trong phân loại câu hỏi. Chúng tôi đã xuất sử dụng mẫu câu hỏi (Question pattern) như là một đặc trưng mới để kết hợp với các đặc trưng khác trong phân loại câu hỏi. Chúng tôi cũng đã xuất sử dụng các tập đặc trưng khác nhau cho mỗi nhóm câu hỏi với các từ để hỏi khác nhau. Chúng tôi nhận thấy rằng khi sử dụng mẫu câu hỏi như là một đặc trưng và kết hợp với các đặc trưng từ vựng, cú pháp, ngữ nghĩa khác có thể cải thiện đáng kể độ chính xác của phân loại câu hỏi. Chúng tôi đã kiểm tra những đặc trưng của mình bằng cách sử dụng bộ phân loại Support Vector Machine trên bộ dữ liệu TREC và đã đạt được độ chính xác phân loại câu hỏi cao hơn so với những nghiên cứu trước đó trên cùng nguyên tắc phân loại và tập dữ liệu.

Authors: Van Tu Nguyen, Anh Cuong Le, Ha Nam Nguyen

document: Giáo trình kiểm thử phần mềm. 1 Tổng quan về kiểm thử 2 Một số ví dụ 3 Cơ sở toán rời rạc cho việc kiểm thử 4 Khảo sát đặc tả và mã nguồn 5 Kiểm thử chức năng 6 Kiểm thử dòng điều khiển 7 Kiểm thử dòng dữ liệu 8 Kiểm thử dựa trên mô hình 9 Kiểm thử tự động và công cụ hỗ trợ 10 KT tích hợp, hệ thống & chấp nhận

Authors: Phạm Ngọc Hùng, Trương Anh Hoàng, Đặng Văn Hưng

document: Điều khiển tốc độ dựa trên thông số tầng MAC cho giao thức tầng giao vận trong mạng không dây da bước. Điều khiển tốc độ dựa trên thông số tầng MAC cho giao thức tầng giao vận trong mạng không dây da bước - VNU-UET Repository Logo VNU-UET Repository Home About Browse by Year Browse by Subject Browse by Division Browse by Author Login Điều khiển tốc độ dựa trên thông số tầng MAC cho giao thức tầng giao vận trong mạng không dây da bước Tran, Truc Mai and Duong, Le Minh (2013) Điều khiển tốc độ dựa trên thông số tầng MAC cho giao thức tầng giao vận trong mạng không dây da bước. Tạp chí công nghệ thông tin và truyền thông, 29 . Full text not available from this repository. Item Type: Article Subjects: Information Technology (IT) Divisions: Faculty of Information Technology (FIT) Depositing User: Ms. Cam Le Tran Thi Date Deposited: 03 Nov 2015 10:34 Last Modified: 03 Nov 2015 11:14 URI: <http://eprints.uet.vnu.edu.vn/eprints/id/eprint/781> Actions (login required)

View Item View EPrints ...

Authors: Truc Mai Tran, Le Minh Duong

document: Examining Students' Learning Styles Impacted on Learning Outcome in the MOOC Course: A Case Study. This article presents the results of a study examining the influence of learning styles on

Hình 54: Ngữ cảnh: Tìm kiếm bài báo theo chủ đề

#### 4.3.8. Tìm kiếm bài báo theo tác giả

Chatbot Context Evaluation

Chatbot

các bài báo của thầy ha quang thuy

Dưới đây là danh sách các bài báo liên quan và đóng góp của Thầy Hà Quang Thuy:

- Tên bài báo:** Môi trường và giải pháp cho chiến lược quốc gia về trí tuệ nhân tạo của Việt Nam  
Tác giả: Quang Thuy Ha, Thanh Thuy Nguyen, Bao Son Pham, Xuan Hieu Phan, Trong Hieu Tran, Mai Vu Tran, Quoc Long Tran, Tri Thanh Nguyen, Hoang Tung Ly
- Tên bài báo:** Using cross-layer heuristic and network coding to improve throughput in multicast wireless mesh networks  
Tác giả: Nguyen Hai Chau, Wonjun Lee, Ha Quang Thuy
- Tên bài báo:** Self-MI: Efficient Multimodal Fusion via Self-Supervised Multi-Task Learning with Auxiliary Mutual Information Maximization  
Tác giả: Ngoc-Hoa Nguyen Thi, Duc-Trong Le, Quang Thuy Ha
- Tên bài báo:** A novel similarity measure for trace clustering based on normalized google distance  
Tác giả: Hong Nhung Bui, Quang Thuy Ha, Tri Thanh Nguyen
- Tên bài báo:** Sentiment analysis and user similarity for social recommender system: An experimental study  
Tác giả: Thi-Ngan Pham, Thi-Hong Vuong, Thi-Hoai Thai, Mai-Vu Tran, Quang-Thuy Ha
- Tên bài báo:** Vietnamese Multidocument Summarization Using Subgraph Selection-Based Approach with Graph-Informed Self-attention Mechanism  
Tác giả: Tam Doan-Thanh, Cam-Van Thi Nguyen, Huu-Thin Nguyen, Mai-Vu Tran, Quang Thuy Ha
- Tên bài báo:** A hybrid genetic algorithm for the traveling salesman problem with drone  
Tác giả: Quang Minh Ha, Yves Deville, Quang Dung Pham, Minh Hoàng Hà
- Tên bài báo:** Parameter Distribution Ensemble Learning for Sudden Concept Drift Detection

Hình 55: Kết quả: Tìm kiếm bài báo theo tác giả

Chatbot Context Evaluation

Chatbot

Here is the list of paper that related to and contributed by Hà Quang Thúy:

document: Môi trường và giải pháp cho chiến lược quốc gia về trí tuệ nhân tạo của Việt Nam. Trí tuệ nhân tạo (TTNT) đã tạo ra những chuyển đổi lớn về kinh tế, xã hội, đời sống của con người, và nhiều nước đã xây dựng chiến lược quốc gia về TTNT. Phân tích môi trường và xây dựng chiến lược quốc gia về TTNT là một công việc đầy thách thức đối với nhiều quốc gia, trong đó có Việt Nam. Bài báo tìm hiểu một số nghiên cứu phân tích về môi trường và xây dựng chiến lược quốc gia về TTNT. Từ đó đưa ra giải pháp định hướng cho chiến lược quốc gia về TTNT của Việt Nam  
Authors: Quang Thuy Ha, Thanh Thuy Nguyen, Bao Son Pham, Xuan Hieu Phan, Trong Hieu Tran, Mai Vu Tran, Quoc Long Tran, Tri Thanh Nguyen, Hoang Tung Ly

document: Using cross-layer heuristic and network coding to improve throughput in multicast wireless mesh networks. Wireless mesh networks (WMNs) receive much research interests because of their reliability, scalability and low cost. Obtaining high-throughput for multicast applications (e.g. video streaming broadcast) in WMNs is challenging due to the interference and the change of channel quality. Cross-layer design and network coding are approaches which have been recently received considerable attention for high-throughput problem in wireless networks. In this paper, we propose an approach namely CLNC (Cross-Layer Network Coding) which is a combination of the above approaches to improve throughput in multicast wireless mesh networks. Our simulation results show that when the number of receivers is high CLNC's throughput is higher at least 30% than that of known methods such as AODV, DSDV and DSR and higher than that of MAODV. Moreover, PDR (Packet Delivery Ration) of CLNC is higher than that of ...  
Authors: Nguyen Hai Chau, Wonjun Lee, Ha Quang Thuy

document: Self-MI: Efficient Multimodal Fusion via Self-Supervised Multi-Task Learning with Auxiliary Mutual Information Maximization. Multimodal representation learning poses significant challenges in capturing informative and distinct features from multiple modalities. Existing methods often struggle to exploit the unique characteristics of each modality due to unified multimodal annotations. In this study, we propose Self-MI in the self-supervised learning fashion, which also leverage Contrastive Predictive Coding (CPC) as an auxiliary technique to maximize the Mutual Information (MI) between unimodal input pairs and the multimodal fusion result with unimodal inputs. Moreover, we design a label generation module, ULGMI for short, that enables us to create meaningful and informative labels for each modality in a self-supervised manner. By maximizing the Mutual Information, we encourage better alignment between the multimodal fusion and the individual modalities, facilitating improved multimodal fusion. Extensive experiments on three benchmark datasets including CMU-MOSI, CMU-MOSEI, and SIMS, demonstrate the effectiveness of Self-MI in enhancing the multimodal fusion task.  
Authors: Ngoc-Hoa Nguyen Thi, Duc-Trong Le, Quang Thuy Ha

document: A novel similarity measure for trace clustering based on normalized google distance.. In trace clustering, a problem of process mining, traditional distance measures only focus on the local

Hình 56: Ngũ cảnh: Tìm kiếm bài báo theo tác giả

#### 4.3.9. Tìm kiếm bài báo theo chủ đề của tác giả

Chatbot Context Evaluation

Chatbot

Các bài báo có liên quan tới học máy của thầy hà quang thùy

Dưới đây là danh sách các bài báo liên quan đến học máy và được đóng góp bởi Hà Quang Thúy:

1. **Tiêu đề:** Learning to recognize phenotype candidates in the auto-immune literature using SVM re-ranking  
**Tác giả:** Nigel Collier, Mai-vu Tran, Hoang-quynh Le, Quang-Thuy Ha, Anika Oellrich, Dietrich Rebholz-Schuhmann  
**Mô tả:** Bài báo này trình bày các kỹ thuật mới để xác định các đề cập tiềm năng về hình thái sinh học bằng cách tận dụng một mô hình kết hợp dựa trên học máy, quy tắc và so khớp từ điển. Nghiên cứu này tập trung vào việc kết hợp nhau chuỗi từ các module này cũng như ưu điểm của các tài nguyên về bản đồ ontological khác nhau.

2. **Tiêu đề:** An experimental study on Vietnamese POS tagging  
**Tác giả:** Oanh Thi Tran, Cuong Anh Le, Thuy Quang Ha, Quynh Hoang Le  
**Mô tả:** Nghiên cứu thực nghiệm về việc gán nhãn phần của tiếng Việt (POS tagging) sử dụng các đặc điểm dựa trên ý tưởng về cấu trúc từ. Kết quả thực nghiệm cho thấy rằng các đặc điểm dựa trên cấu trúc từ luôn cho độ chính xác cao hơn so với các phương pháp trước đó - thường là các đặc điểm dựa trên từ.

3. **Tiêu đề:** QASA: advanced document retriever for open-domain question answering by learning to rank question-aware self-attentive document representations  
\*\*Tác

Hình 57: Kết quả: Tìm kiếm bài báo theo chủ đề của tác giả

Here is the list of paper that related to machine learning and contributed by Hà Quang Thuy:

document: Learning to recognize phenotype candidates in the auto-immune literature using SVM re-ranking. The identification of phenotype descriptions in the scientific literature, case reports and patient records is a rewarding task for bio-medical text mining. Any progress will support knowledge discovery and linkage to other resources. However because of their wide variation a number of challenges still remain in terms of their identification and semantic normalisation before they can be fully exploited for research purposes.

This paper presents novel techniques for identifying potential complex phenotype mentions by exploiting a hybrid model based on machine learning, rules and dictionary matching. A systematic study is made of how to combine sequence labels from these modules as well as the merits of various ontological resources. We evaluated our approach on a subset of Medline abstracts cited by the Online Mendelian Inheritance of Man database related to auto-immune diseases.

Using partial matching the best micro-averaged F-score for phenotypes and five other entity classes was 79.9%. A best performance of 75.3% was achieved for phenotype candidates using all semantics resources. We observed the advantage of using SVM-based learn-to-rank for sequence label combination over maximum entropy and a priority list approach. The results indicate that the identification of simple entity types such as chemicals and genes are robustly supported by single semantic resources, whereas phenotypes require combinations. Altogether we conclude that our approach coped well with the compositional structure of phenotypes in the auto-immune domain.

Authors: Nigel Collier, Mai-vu Tran, Hoang-quynh Le, Quang-Thuy Ha, Anika Oellrich, Dietrich Rebholz-Schuhmann

document: An experimental study on Vietnamese POS tagging. In Natural Language Processing (NLP), Part-of-speech tagging is one of the important tasks. It, however, has not drawn much attention of Vietnamese researchers all over the world. In this paper, we present an experimental study on Vietnamese POS tagging. Motivated from Chinese research and Vietnamese characteristics, we present a new kind of features based on the idea of word composition. We call it morpheme based features. To verify the effectiveness of these features, we use three powerful machine learning techniques - MEM, CRF and SVM. In addition, we also built a Vietnamese POS-tagged corpus with approximately 8000 sentences of different genres to conduct experiments. Experimental results showed that morpheme-based features always give higher precision in comparison with previous approaches - usually word-based features. We achieved the precision of 91.64% by using these morpheme ...

Authors: Oanh Thi Tran, Cuong Anh Le, Thuy Quang Ha, Quynh Hoang Le

document: QASA: advanced document retriever for open-domain question answering by learning to rank question-aware self-attentive document representations. For information consumers, being able to obtain a short and accurate answer for a query is one of the most desirable features. This motivation, along with the rise of deep learning, has led to a boom in open-domain Question Answering (QA) ...

Hình 58: Ngữ cảnh: Tìm kiếm bài báo theo chủ đề của tác giả

#### 4.3.10. Tìm kiếm công trình cụ thể

Thông tin cụ thể về khóa luận Machine learning-based extraction of semantic relations from biomedical literature

Khóa luận "Trích xuất mối quan hệ ngữ nghĩa trong văn bản y sinh dựa trên học máy" xem xét việc Trích xuất Mối quan hệ như hai tác vụ của khai thác văn bản, tức là Nhận dạng Thực thể Đã Đặt Tên (NER) và Phân loại Mối quan hệ (RC). Nhiệm vụ của việc nhận dạng thực thể đã đặt tên trong lĩnh vực y sinh (NER) nhằm xác định các thực thể đã đặt tên từ văn bản y sinh không cấu trúc và phân loại chúng vào một tập hợp các danh mục/loại được xác định trước như gene/protein, hiện tượng, bệnh tật và hóa chất, hoặc 'không phải là cái nào trong số đó'. Phân loại mối quan hệ (RC) là nhiệm vụ phát hiện các kết nối ngữ nghĩa giữa các thực thể y sinh.

Người hướng dẫn: Collier, Nigel, Đặng, Thanh Hải

Bạn có thể tìm hiểu thêm thông tin chi tiết và tài liệu của khóa luận tại đường link sau: [Link](#)

Hình 59: Kết quả: Tìm kiếm công trình cụ thể

Title: Machine learning-based extraction of semantic relations from biomedical literature = Trích xuất mối quan hệ ngữ nghĩa trong văn bản y sinh dựa trên học máy

Abstract: In this Dissertation, we consider Relation Extraction as two text mining sub-tasks, i.e., Named Entity Recognition (NER) and Relation Classification (RC). The task of biomedical named entity recognition (NER) seeks to locate named entities from free-form biomedical text and classify them into a set of pre-defined categories/types such as gene/protein, phenotype, disease, and chemical, or 'none-of-the-above'. Relation classification (RC) is the task of discovering semantic connections between biomedical entities

Advisor: Collier, Nigel  
Đặng, Thanh Hải

Link: [https://repository.vnu.edu.vn/handle/VNU\\_123/140236](https://repository.vnu.edu.vn/handle/VNU_123/140236)

Hình 60: Ngữ cảnh: Tìm kiếm công trình cụ thể

#### 4.3.11. Gợi ý giáo viên hướng dẫn

Chatbot Context Evaluation

Gợi ý thầy giáo hướng dẫn khóa luận liên quan tới học máy

Here is the list of teachers that have research directions related to machine learning:

1. Lê Hồng Hải
2. Nguyễn Thu Trang
3. Bùi Thế Duy
4. Trần Quốc Long
5. Ngô Thị Duyên
6. Phạm Bảo Sơn
7. Tạ Việt Cường
8. Hà Quang Thụy
9. Cấn Duy Cát
10. Nguyễn Hà Nam
11. Lê Đức Trọng
12. Trần Trường Thủy
13. Hoàng Xuân Huân
14. Trần Mai Vũ
15. Nguyễn Tuệ

Hình 61: Kết quả: Gợi ý giáo viên hướng dẫn

Chatbot Context Evaluation

Gợi ý thầy giáo hướng dẫn

Here is the list of teachers that have research directions related to machine learning :

Lê Hồng Hải, Nguyễn Thu Trang, Bùi Thế Duy, Trần Quốc Long, Ngô Thị Duyên, Phạm Bảo Sơn, Tạ Việt Cường, Hà Quang Thụy, Cấn Duy Cát, Nguyễn Hà Nam, Lê Đức Trọng, Trần Trường Thủy, Hoàng Xuân Huân, Trần Mai Vũ, Nguyễn Tuệ, Nguyễn Ngọc Hoá, Nguyễn Việt Anh, Nguyễn Nam Hải

Hình 62: Ngữ cảnh: Gợi ý giáo viên hướng dẫn

#### 4.3.12. Gợi ý đề tài khóa luận

Chatbot Context Evaluation

gợi ý một chủ đề khóa luận phù hợp với tôi

Based on your learning results in Artificial Intelligence and Software Engineering, as well as your interest in these fields, I would suggest the following thesis topic for you:

Title: "Enhancing Software Development with Machine Learning Techniques"

Advisor: Lê Khánh Trình

Description:

This thesis will focus on exploring the integration of machine learning techniques into the software development process to improve various aspects of software engineering. You will delve into the application of machine learning algorithms in tasks such as code analysis, bug detection, and software testing. By utilizing machine learning, you can automate repetitive tasks, enhance code quality, and optimize software performance.

Effectiveness:

This topic is highly relevant in the current technological landscape where automation and optimization are key factors in software development. It offers a practical approach to enhancing traditional software engineering practices and can lead to significant improvements in productivity and software quality.

Difficulty:

The topic can be challenging as it requires a solid understanding of both software engineering principles and machine learning algorithms. Implementing machine learning in software development involves complex data processing and model training, which may pose technical challenges.

Instructions:

1. Start by conducting a literature review to understand the existing research on machine learning in software engineering.

Hình 63: Kết quả: Gợi ý đề tài khóa luận

Chatbot   Context   Evaluation

Chatbot

Learning result:

- Education program: Standard
- Software Engineering: A
- Discrete Mathematics: B+
- Artificial Intelligence: A+
- Algebra: B
- Computer Network: C+

Total Credit: 114

Here is the list of papers related to Artificial Intelligence, Machine Learning, Software Engineering and contributed by Lê Khánh Trinh:

document: Suggesting natural method names to check name consistencies. Misleading names of the methods in a project or the APIs in a software library confuse developers about program functionality and API usages, leading to API misuses and defects. In this paper, we introduce MNire, a machine learning approach to check the consistency between the name of a given method and its implementation. MNire first generates a candidate name and compares the current name against it. If the two names are sufficiently similar, we consider the method as consistent. To generate the method name, we draw our ideas and intuition from an empirical study on the nature of method names in a large dataset. Our key finding is that high proportions of the tokens of method names can be found in the three contexts of a given method including its body, the interface (the method's parameter types and return type), and the enclosing class' name. Even when such tokens are not there, MNire uses the ...

Authors: Son Nguyen, Hung Phan, Trinh Le, Tien N Nguyen

document: BM-BronchoLC-A rich bronchoscopy dataset for anatomical landmarks and lung cancer lesion recognition. Flexible bronchoscopy has revolutionized respiratory disease diagnosis. It offers direct visualization and detection of airway abnormalities, including lung cancer lesions. Accurate identification of airway lesions during flexible bronchoscopy plays an important role in the lung cancer diagnosis. The application of artificial intelligence (AI) aims to support physicians in recognizing anatomical landmarks and lung cancer lesions within bronchoscopy images. This work describes the ...

Hình 64: Ngữ cảnh: Gợi ý đề tài khóa luận

#### 4.3.13. Đánh giá câu trả lời

Chatbot   Context   Evaluation

evaluation

evaluation_cost	score	reason
0.003326	1	The score is 1.00 because the output provided a highly relevant and appropriate response to your query. Great job!
0.004140000000000005	0.9375	The score is 0.94 because the actual output incorrectly states that the advisor for the thesis is Lê Khánh Trinh, instead of Lê Thái Sơn and Lê Văn Hùng.
0.000936000000000001	0	The score is 0.00 because the context provided is a list of learning results and total credits of a student, which is irrelevant to suggesting a research topic.

Hình 65: Kết quả: Đánh giá câu trả lời

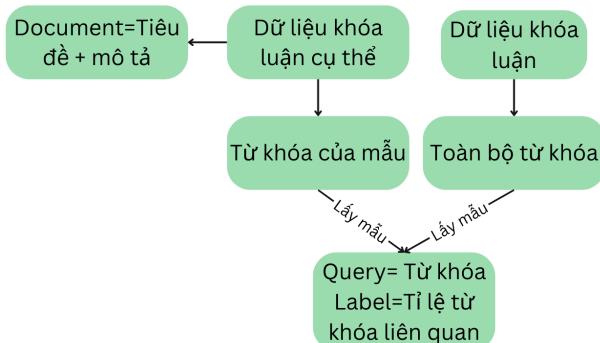
### 4.4. Đánh giá hệ thống

#### 4.4.1. Đánh giá hiệu quả truy xuất

##### 4.4.1.1. Xây dựng bộ dữ liệu

Bộ dữ liệu dùng để đánh giá có cấu trúc:

- Query: Truy vấn
- Document: Tài liệu
- Label: Mức độ tương đồng giữa query và document



Hình 66: Kết quả: Đánh giá câu trả lời

Dữ liệu bài báo không bao gồm từ khóa nên phải tiến hành thu thập bên ngoài từ trang github [LIAAD/KeywordExtractor-Datasets](#). Các bộ dữ liệu con được sử dụng là:

- Inspec [10]
- SemEval2017 [11]
- WikiNews [12]
- pak2018 [13]
- theses100

Việc xây dựng bộ dữ liệu được thực hiện qua các bước:

- Kết hợp danh sách bài báo và khóa luận thu được
- Lọc bỏ những mẫu không chứa từ khóa trong dữ liệu khóa luận
- Với mỗi mẫu, thực hiện khuyếch đại dữ liệu
- Trả về 2 bộ dữ liệu riêng biệt cho đánh giá mô hình nhúng và mô hình Rerank

Quá trình khuyếch đại dữ liệu gồm các bước:

- Thu thập tất cả từ khóa trong bộ dữ liệu tạo thành một tập
- Tiến hành khuếch đại dữ liệu trên mỗi mẫu:
  1. Kết hợp tiêu đề và mô tả của mẫu tạo thành document,
  2. Lấy mẫu một số từ khóa ngẫu nhiên trong tập từ khóa của mẫu. Dùng tập từ khóa vừa thu được tạo thành query. Thêm chuỗi đánh dấu nêu cần, gán nhãn chúng là 1
  3. Tạo một tập các từ khóa không có phần tử trùng với tập từ khóa của mẫu đang xét. Thực hiện tương tự bước 2, nhưng gán nhãn là 0
  4. Thực hiện lấy mẫu cả trên tập từ khóa của mẫu và tập từ khóa ngoài mẫu, nhãn của mẫu sẽ là tỉ lệ số lượng từ khóa trong tập từ khóa của mẫu được lấy trên tổng từ khóa của mẫu.

Số lượng điểm dữ liệu thu được phụ thuộc vào số lần thực hiện lấy mẫu. Với mỗi loại điểm dữ liệu (nhãn 0, 1, hoặc giữa 0 và 1) lấy mẫu 4 lần thì dữ liệu thu được sẽ có khoảng 800 nghìn điểm dữ liệu.

#### 4.4.1.2. Thực hiện đánh giá.

##### 4.4.1.2.1. Đánh giá mô hình nhúng

Thực hiện thử nghiệm với 2 mô hình nổi tiếng sử dụng hàm mất mát cosin:

	multilingual-e5-small	all-MiniLM-L6-v2
Ngôn ngữ	Đa ngôn ngữ	Đa ngôn ngữ
Tokennizer	Unigram tokenizer. Tách theo từ đối với tiếng việt	BPE tokenizer. Tách theo byte đối với tiếng việt
Kích thước	12 tầng	6 tầng
Đánh giá	$\approx 0.002/1$	$\approx 1/1$
Yêu cầu chỉnh sửa văn bản	Thêm “query: “ vào trước văn bản truy xuất. Thêm “document: “ trước văn bản cần truy xuất	Không

Bảng 7: So sánh mô hình nhúng

Dựa theo thiết kế dữ liệu đánh giá, truy vấn và kết quả trả về chỉ có thể liên quan hoặc không liên quan, không thể tồn tại đối nghịch nên giá trị sai số lớn nhất kì vọng là 1. Thực tế, mô hình có thể sai sót và trả về giá trị lớn hơn 1.

Mô hình multilingual-e5-small cho ra kết quả tốt hơn mô hình all-MiniLM-L6-v2 rất nhiều. Việc này là do multilingual-e5-small là một mô hình lớn hơn và có bộ tokenizer đổi với tiếng việt tốt hơn và cũng do multilingual-e5-smal có kích thước lớn hơn.

### Fine tune

Thêm vào mỗi mô hình 2 tầng tuyển tính có kích thước lần lượt là (384, 128) và (128, 384) cùng một tầng hoạt hóa GELU ở giữa, đóng băng mô hình gốc và tiến hành huấn luyện với learning\_rate = 1e-5. Mô hình all-MiniLM-L6-v2 đạt được độ mất mát trung bình khoảng 0.013 sau 1 epoch và không có dấu hiệu giảm thêm. Kết quả của multilingual-e5-small không thay đổi.

#### 4.4.1.2.2. Đánh giá mô hình Rerank.

Thực hiện đánh giá với bộ dữ liệu trên, độ mất mát thu được vào khoảng 0.002/1. Độ mất mát này tương đồng với độ mất mát của mô hình multilingual-e5-small nhưng kết quả của Cross-Encoder vẫn luôn được đảm bảo hơn khi so sánh với Bi-Encoder hơn theo bài báo *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks* [6]

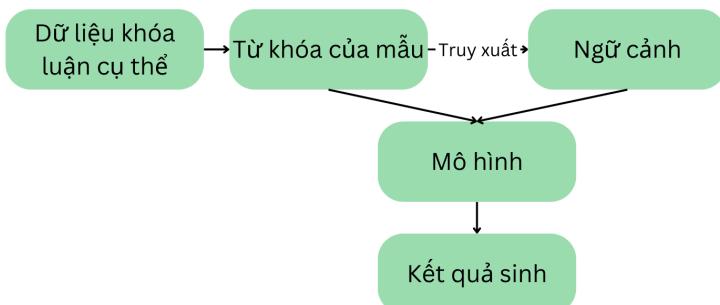
#### 4.4.2. Đánh giá chất lượng sinh

Thực hiện sử dụng thư viện DeepEval. Các độ đo được sử dụng là:

- Answer Relevance: Độ liên quan giữa câu trả lời và câu hỏi
- Context Relevance: Độ liên quan giữa câu trả lời và ngữ cảnh
- Faithfulness: mức độ khẳng định của ngữ cảnh đối với câu trả lời

Chỉ thực hiện đánh giá gợi ý chủ đề khóa luận do hạn chế về thời gian và cũng do tính sáng tạo của các ca sử dụng còn lại là thấp.

#### 4.4.2.1.1. Tạo bộ dữ liệu



Hình 67: Kết quả: Đánh giá câu trả lời

Lấy ra một 100 điểm dữ liệu thuộc khóa luận UET. Tạo bộ dữ liệu gồm:

- Truy vấn: Được tạo từ từ khóa
- Ngữ cảnh: thu hồi từ 2 trường hợp có thầy hướng dẫn hoặc không. Nếu có thầy hướng dẫn đảm bảo tên thầy có trong danh sách giáo viên
- Câu trả lời: được sinh từ truy vấn và ngữ cảnh sử dụng LLM

#### 4.4.2.1.2. Thực hiện đánh giá

Thực hiện đánh giá bằng DeepEval trên các độ đo đã nêu. Kết quả thu được là:

- Context\_relevant: 1/1
- Answer\_relevant: 0.9462/1
- Groundness: 1/1

Bộ dữ liệu tuy nhỏ nhưng phần nào đánh giá khả năng của mô hình. Giá trị Context Relevant luôn bằng 1 cho thấy chất lượng của quá trình truy xuất thông tin là cao.

#### 4.4.3. Nhận xét

Hệ thống nhìn chung cho kết quả tốt, nhưng kết quả này còn phụ thuộc nhiều vào việc viết chủ giải công cụ và instruction prompt. Điều này cho thấy các mô hình ngôn ngữ lớn vẫn còn nhiều hạn chế trong việc hiểu ý của người dùng.

## 5. Kết luận

### 5.1. Kết quả đạt được

Trong khóa luận này, dựa trên những kiến thức về RAG tôi đã xây dựng được *Hệ thống hỗ trợ tư vấn sinh viên thực hiện khóa luận tốt nghiệp*. Đối tượng hướng đến ở đây chủ yếu là học sinh thực hiện khóa luận nhưng một số đối tượng khác như: những người muốn tìm hiểu thông tin về giáo viên hay những thầy cô hỗ trợ học sinh tìm kiếm giáo viên hướng dẫn... cũng có thể được hưởng lợi từ hệ thống. Tôi cũng đã tiến hành kiểm thử các thành phần và đánh giá kết quả của hệ thống. Các thành phần hoạt động trơn tru không phát hiện lỗi. Kết quả đánh giá cho thấy chất lượng của hệ thống trên lý thuyết là tốt. Ngoài ra, đi kèm với hệ thống, tôi đã tổng hợp được bộ dữ liệu về khóa luận từ nhiều nguồn khác nhau.

Các tài nguyên có thể được tìm thấy tại:

- Hệ thống hỗ trợ sinh viên thực hiện khóa luận tốt nghiệp, chương trình tạo dữ liệu đánh giá, crawler: [Link](#)
- Bộ dữ liệu liên quan tới khóa luận tốt nghiệp: [Link](#)

### 5.2. Hướng phát triển

Hệ thống còn cần được phát triển thêm kể cả cho việc giải quyết bài toán được khóa luận đặt ra hay mở rộng sang bài toán khác. Một số hướng phát triển khả thi có thể kể đến như:

- Thu thập dữ liệu người dùng: Mô hình sẽ gửi nhiều kết quả đến người dùng để nhờ người dùng đánh giá xem câu trả lời nào tốt hơn. Từ đó tạo ra bộ dữ liệu giúp cải tiến hệ thống.
- Thêm trọng số thời gian: Công nghệ phát triển rất nhanh tương ứng với việc tốc độ lỗi thời của tài liệu viết về chúng cũng rất nhanh. Việc truy xuất nên ưu tiên những tài liệu có thời gian gần để mang lại thông tin hữu ích cho người dùng.
- Tự động thực thi thu thập dữ liệu: Cho phép chương trình thu thập dữ liệu chạy một cách tự động khi xác định có dữ liệu mới được tải lên hoặc sau một khoảng thời gian cố định

- Thu thập nhiều dữ liệu hơn: Dữ liệu về người dùng còn hạn chế, hiện tại chỉ là bản mẫu. Bên cạnh đó dữ liệu về giáo viên và thực hiện khóa luận còn chưa được đầy đủ. Dữ liệu giáo hiện giới hạn trong khoa công nghệ thông tin.
- Huấn luyện mô hình đặc trưng: Huấn luyện mô hình đặc trưng với dữ liệu thu được để có được câu trả lời tốt hơn và hiệu suất tốt hơn.

## **Lời kết**

Khóa luận này đã được tác giả dành nhiều công sức để thực hiện. Tuy vậy, những thiếu sót là điều không thể tránh khỏi. Mọi lời nhận xét và góp ý của hội đồng về khóa luận từ cách thực hiện cho đến sản phẩm đều sẽ được tác giả hoan nghênh và tiếp thu cho những công trình sau này. Tôi xin chân thành cảm ơn.

## Tài liệu tham khảo

- [1] A. Vaswani và c.s., “Attention Is All You Need”. 2023.
- [2] M. Lewis và c.s., “BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension”. 2019.
- [3] Y. Gao và c.s., “Retrieval-Augmented Generation for Large Language Models: A Survey”. 2024.
- [4] N. F. Liu và c.s., “Lost in the Middle: How Language Models Use Long Contexts”. 2023.
- [5] Y. A. Malkov và D. A. Yashunin, “Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs”. 2018.
- [6] N. Reimers và I. Gurevych, “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks”. 2019.
- [7] [Online]. Available at: <https://docs.confident-ai.com/docs/metrics-contextual-relevancy>
- [8] [Online]. Available at: <https://docs.confident-ai.com/docs/metrics-answer-relevancy>
- [9] [Online]. Available at: <https://docs.confident-ai.com/docs/metrics-faithfulness>
- [10] A. Hulth, “Improved Automatic Keyword Extraction Given More Linguistic Knowledge”. tr , 2003. doi: [10.3115/1119355.1119383](https://doi.org/10.3115/1119355.1119383).
- [11] I. Augenstein, M. Das, S. Riedel, L. Vikraman, và A. McCallum, “SemEval 2017 Task 10: ScienceIE - Extracting Keyphrases and Relations from Scientific Publications”. 2017.
- [12] O. Medelyan, I. Witten, và D. Milne, “Topic Indexing with Wikipedia”. tr , 2010.
- [13] R. Campos, V. Mangaravite, A. Pasquali, A. Jorge, C. Nunes, và A. Jatowt, “YAKE! Keyword extraction from single documents using multiple local features”, *Information Sciences*, vol 509, tr 257–289, 2020, doi: <https://doi.org/10.1016/j.ins.2019.09.013>.