TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**NGÔ HỮU LỄ**

**LÊ DƯƠNG MINH KHOA**

**XÂY DỰNG SQUAD 2.0 CHO BÀI TOÁN HỎI ĐÁP TỰ ĐỘNG TRONG LĨNH VỰC GIÁO DỤC TIẾNG VIỆT**

**DỰ ÁN CÔNG NGHỆ THÔNG TIN 2**

**KHOA HỌC MÁY TÍNH  
KỸ THUẬT PHẦN MỀM**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2024**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**NGÔ HỮU LỄ - 51800571**

**LÊ DƯƠNG MINH KHOA - 51800055**

**XÂY DỰNG SQUAD 2.0 CHO BÀI TOÁN HỎI ĐÁP TỰ ĐỘNG TRONG LĨNH VỰC GIÁO DỤC TIẾNG VIỆT**

**DỰ ÁN CÔNG NGHỆ THÔNG TIN 2**

**KHOA HỌC MÁY TÍNH  
KỸ THUẬT PHẦN MỀM**

Người hướng dẫn

**TS. Trần Thanh Phước**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2024**

**LỜI CẢM ƠN**

Chúng tôi xin chân thành cảm ơn thầy Trần Thanh Phước đã hỗ trợ và giúp đỡ chúng tôi hoàn thành dự án này. Khi bước vào quá trình nghiên cứu và thực hiện dự án học máy của mình, chúng tôi đã nhận ra giá trị vô giá của việc nhận được sự hướng dẫn đầy kiên nhẫn và sẻ chia kiến thức từ thầy. Thầy không chỉ đóng vai trò là một người hướng dẫn đáng kính mà còn là nguồn động viên giúp chúng tôi kiên trì vượt qua những khó khăn. Nhờ có sự hướng dẫn và giúp đỡ của thầy, chúng tôi đã tiếp cận và hiểu sâu hơn về nhiều khái niệm mới trong lĩnh vực học tập của mình.

Chúng tôi xin bày tỏ lòng biết ơn sâu sắc nhất tới thầy Trần Thanh Phước. Sự hướng dẫn nhiệt thành và sự hỗ trợ không biết mệt mỏi từ thầy đã không chỉ giúp dự án đạt được thành công mà còn giúp tôi phát triển bản thân trên con đường sự nghiệp sau này. Chúng tôi mong rằng, những kiến thức và kinh nghiệm mà thầy đã truyền đạt sẽ tiếp tục giúp đỡ và phát triển bản thân trong tương lai.

Chúng tôi xin chân thành cảm ơn.

*TP. Hồ Chí Minh, ngày 21 tháng 03 năm 2024*

*Tác giả*

*(Ký tên và ghi rõ họ tên)*

*Ngô Hữu Lễ*

*Lê Dương Minh Khoa*

**CÔNG TRÌNH ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi và được sự hướng dẫn khoa học của TS. Trần Thanh Phước. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong Dự án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung Dự án của mình**. Trường Đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 21 tháng 03 năm 2023*

*Tác giả*

*(Ký tên và ghi rõ họ tên)*

*Ngô Hữu Lễ*

*Lê Dương Minh Khoa*

**XÂY DỰNG SQUAD 2.0 CHO BÀI TOÁN HỎI ĐÁP TỰ ĐỘNG TRONG LĨNH VỰC GIÁO DỤC TIẾNG VIỆT**

**TÓM TẮT**

Đề tài nghiên cứu cách tạo ra tập dữ liệu SQuAD 2.0 cho bài toán hỏi đáp tự động trong tiếng Việt về các chủ đề giáo dục. Đề tài gồm các nội dung chính như: Giới thiệu dự án, cơ sở lý thuyết, mô hình bài toán, thực nghiệm, ứng dụng minh họa, kết luận và hướng phát triển.

Chương 1: Trình bày về lý do chọn đề tài, mục tiêu và phạm vi nghiên cứu của dự án.

Chương 2: Trình bày ý tưởng, kiến thức nền tảng, những gì đạt được trong những công trình nghiên cứu về vấn đề hỏi đáp trong những năm gần đây, từ đó rút ra hướng giải quyết vấn đề.

Chương 3: Trình bày phương pháp thực hiện bài toán như mô hình tổng quát, giải thích từng bước và ví dụ minh họa.

Chương 4: Trình bày dữ liệu thực nghiệm, công cụ đánh giá, kết quả thảo luận và thảo luận của mô hình đào tạo.

Chương 5: Trình bày ứng dụng web minh họa cho mô hình hỏi đáp trên.

Chương 6: Trình bày những kết quả mang lại từ dự án, những hạn chế cần được khắc phục và hướng nghiên cứu, phát triển trong tương lai.

**BUILDING SQUAD 2.0 FOR AUTOMATIC QUESTION AND ANSWER PROBLEMS IN THE FIELD OF VIETNAMESE EDUCATION**

**ABSTRACT**

The topic studies how to create the SQuAD 2.0 dataset for the automatic question answering problem in Vietnamese on educational topics. The topic includes the main contents such as: Introduction of the project, theoretical basis, problem model, experiments, illustrative applications, conclusions and development directions.

Chapter 1: Present the reasons for choosing the topic, the objectives, and the scope of research for the project.

Chapter 2: Presentation of the idea, the background knowledge, the achievement in the research works on the question answering problem in recent years, and from there draw the solution to the problem.

Chapter 3: Presentation of the method of solving the problem such as the general model, explanation of each step and illustrative examples.

Chapter 4: Presentation of the experimental data, evaluation tools, discussion and discussion of the trained model.

Chapter 5: Presentation of the web application illustrating the question answering model above.

Chapter 6: Presentation of the results obtained from the project, the limitations that need to be overcome and the direction of research and development in the future.

**MỤC LỤC**

[DANH MỤC HÌNH VẼ vii](#_Toc161940998)

[DANH MỤC BẢNG BIỂU viii](#_Toc161940999)

[DANH MỤC CÁC CHỮ VIẾT TẮT ix](#_Toc161941000)

[CHƯƠNG 1. GIỚI THIỆU 1](#_Toc161941001)

[1.1 Hiện trạng 1](#_Toc161941002)

[1.2 Mục tiêu của đồ án 2](#_Toc161941003)

[1.2.1 Mục tiêu nghiên cứu 2](#_Toc161941004)

[1.2.2 Kết quả dự kiến 2](#_Toc161941005)

[1.2.3 Phạm vi nghiên cứu 3](#_Toc161941006)

[CHƯƠNG 2. KIẾN THỨC NỀN TẢNG 4](#_Toc161941007)

[2.1 Công trình liên quan 4](#_Toc161941008)

[2.2 Kiến thức nền tảng 12](#_Toc161941009)

[2.2.1 Kho dữ liệu SQuAD 2.0 13](#_Toc161941010)

[2.2.2 Phân loại ý định 16](#_Toc161941011)

[2.2.3 Trả lời tự động 17](#_Toc161941012)

[2.2.4 Học chuyển giao 19](#_Toc161941013)

[2.2.5 Mô hình BERT 21](#_Toc161941014)

[CHƯƠNG 3. MÔ HÌNH BÀI TOÁN 22](#_Toc161941015)

[3.1 Mô hình tổng quát 22](#_Toc161941016)

[3.2 Kiến trúc ứng dụng 23](#_Toc161941017)

[CHƯƠNG 4. THỰC NGHIỆM 25](#_Toc161941018)

[4.1 Dữ liệu thực nghiệm 25](#_Toc161941019)

[4.2 Công cụ đánh giá 26](#_Toc161941020)

[4.2.1 F1 Score 26](#_Toc161941021)

[4.2.2 Exact Match 27](#_Toc161941022)

[4.2.3 BLEU Score 28](#_Toc161941023)

[4.3 Kết quả thực nghiệm 29](#_Toc161941024)

[4.4 Thảo luận 30](#_Toc161941025)

[4.4.1 Bài toán IC 30](#_Toc161941026)

[4.4.2 Bài toán MRC 30](#_Toc161941027)

[4.4.3 Bài toán IC kết hợp MRC 31](#_Toc161941028)

[CHƯƠNG 5. ỨNG DỤNG MINH HỌA 32](#_Toc161941029)

[5.1 Flask cho Python 32](#_Toc161941030)

[5.2 Ứng dụng trang web 33](#_Toc161941031)

[CHƯƠNG 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 35](#_Toc161941032)

[6.1 Kết luận 35](#_Toc161941033)

[6.2 Hướng phát triển 36](#_Toc161941034)

[TÀI LIỆU THAM KHẢO 38](#_Toc161941035)

# DANH MỤC HÌNH VẼ

[Hình 2.1 Ví dụ về đầu ra của hệ thống COBERT với truy vấn đầu vào 4](#_Toc161936915)

[Hình 2.2 Mô hình hệ thống QA lai diễn ngôn-tranh luận tiếng Việt. 5](#_Toc161936916)

[Hình 2.3 Số câu trả lời chấp nhận được theo nhóm câu hỏi của các mô hình. 6](#_Toc161936917)

[Hình 2.4 Ví dụ về hệ thống QA khai thác bao gồm câu hỏi, ngữ cảnh và câu trả lời. 7](#_Toc161936918)

[Hình 2.5 Kiến trúc hệ thống CoQUAD. 7](#_Toc161936919)

[Hình 2.6 Điểm F1 của tất cả các mô hình. 8](#_Toc161936920)

[Hình 2.7 Kiến trúc mô hình Transformer. 9](#_Toc161936921)

[Hình 2.8 Khung đề xuất cho Bangla RC từ khâu tiền xử lý dữ liệu đến dự đoán đáp án của RC. 10](#_Toc161936922)

[Hình 2.9 Bản so sánh về Độ chính xác và điểm F1 của một số mô hình 10](#_Toc161936923)

[Hình 2.10 Tổng quan về khung đề xuất cho hệ thống trả lời câu hỏi của MRC 11](#_Toc161936924)

[Hình 2.11 Kiến trúc của ExtGPT-QA được đề xuất cho khả năng đọc hiểu của máy (MRC). 12](#_Toc161936925)

[Hình 3.1 Mô hình hoạt động 22](#_Toc161936926)

[Hình 3.2 Kiến trúc ứng dụng 24](#_Toc161936927)

[Hình 5.1 Giao diện ứng dụng 33](#_Toc161936928)

[Hình 5.2 Lịch sử tin nhắn 34](#_Toc161936929)

# DANH MỤC BẢNG BIỂU

[Bảng 1 Cấu trúc dữ liệu SQuAD dạng JSON 15](#_Toc161936965)

[Bảng 2 Kết quả thực nghiệm 29](#_Toc161936966)

# DANH MỤC CÁC CHỮ VIẾT TẮT

|  |  |
| --- | --- |
|  |  |
| BERT | Bidirectional Encoder Representations from Transformers |
| Bi-LSTM | Bidirectional Long Short-Term Memory |
| BLEU | Bilingual Evaluation Understudy |
| CNN | Convolutional Neural Network |
| CoQUAD | COVID-19 Question Answering Dataset |
| EM | Exact Match |
| ExtGPT-QA | Extended Generative Pretrained Transformers-based Question Answering |
| GRU | Gated Recurrent Unit |
| IE | Information Extraction |
| IR | Information Retrieval |
| LSTM | Long Short-Term Memory |
| MRC | Machine Reading Comprehension |
| NER | Named Entity Recognition |
| NLG | Natural Language Generation |
| NLP | Natural Language Processing |
| NLU | Natural Language Understanding |
| QA | Question Answering |
| QAS | Question Answering System |
| RC | Reading Comprehension |
| SQuAD | Stanford Question Answering Dataset |
| T5 | Text-to-Text Transfer Transformer |

# GIỚI THIỆU

## Hiện trạng

Việc trả lời câu hỏi và yêu cầu của người dùng trong nhiều lĩnh vực là rất quan trọng. Hiện nay, việc này còn phụ thuộc vào con người và chưa có công cụ hỗ trợ. Điều này gây ra nhiều khó khăn như chậm trễ, thiếu chính xác và không minh bạch. Ngoài ra, việc chọn đúng người trả lời cũng không dễ dàng và có thể gây hiểu lầm. Người dùng cần trao đổi thông tin nhiều hơn qua các phương tiện truyền thông, nhất là Internet. Họ phải xử lý hàng ngày nhiều thông tin khác nhau. Hệ thống hỏi-đáp tự động giúp họ tìm kiếm câu trả lời ngắn gọn, súc tích và chính xác cho những thắc mắc của họ. Đó là mục tiêu của các hệ thống hỏi đáp thông tin qua mạng hiện nay.

Trợ lý ảo, chatbot, hay hệ thống trả lời tự động là chủ đề nóng, khi các công ty lớn như Microsoft, Google, Facebook, Apple, Samsung đã giới thiệu các sản phẩm của mình. Microsoft và Facebook đã cung cấp các nền tảng cho nhà phát triển tạo ra các chatbot trên Web, Skype, hay Tin nhắn. Mục tiêu là tạo ra một trợ lý ảo thông minh trong hệ sinh thái của mình. Nhiều công ty mong muốn phát triển các trợ lý ảo hiểu ngôn ngữ tự nhiên, trả lời và tương tác tự nhiên với con người. Nhiều người tin rằng xử lý ngôn ngữ tự nhiên NLP và học sâu Deep Learning có thể nâng cao chất lượng và hiệu quả của hệ thống. Nhưng để áp dụng lý thuyết vào thực tế là khó khăn, con người cần tích hợp trí tuệ nhân tạo AI vào các sản phẩm công nghiệp.

Hệ thống trả lời tự động có vai trò trợ giúp con người trong nhiều lĩnh vực: y tế, giáo dục, thương mại điện tử … đáng để nghiên cứu và phát triển. Nhiều công ty lớn đã xây dựng chatbot trả lời hỏi đáp trực tuyến, dùng xử lý ngôn ngữ tự nhiên và học sâu Deep Learning để nâng cao chất lượng và hiệu quả của chatbot, tiết kiệm chi phí, tư vấn khách hàng liên tục mà không cần nhân viên trực tiếp.

Để phát triển phẩm chất và năng lực người học, hệ thống câu hỏi của giáo viên là yếu tố quan trọng trong dạy học Ngữ văn THPT, ảnh hưởng đến chất lượng và khả năng lĩnh hội kiến thức của học sinh. Câu hỏi kích thích não bộ học sinh suy nghĩ, tái hiện và thu nạp kiến thức mới, liên hệ và áp dụng kiến thức vào thực tiễn, đưa ra cách giải quyết vấn đề sáng tạo cá nhân. Do đó nhóm tập trung giải quyết bài toán hỏi-đáp quanh vấn đề liên quan đến kiến thức văn học phổ thông cho tiếng Việt. Thế nên nhóm đã chọn đề tài “Xây dựng SQuAD 2.0 cho bài toán hỏi đáp tự động trong lĩnh vực giáo dục tiếng Việt” nhằm tạo ra cổng thông tin hỏi đáp trực tuyến.

## Mục tiêu của đồ án

### Mục tiêu nghiên cứu

Nhóm tiến hành dự án “Triển khai SQuAD 2.0 cho bài toán hỏi đáp tự động trong lĩnh vực giáo dục tiếng Việt” với các mục tiêu sau:

+ Thu thập và xử lý dữ liệu để xây dựng một bộ dữ liệu SQuAD 2.0 phong phú, đa dạng và chất lượng cho tiếng Việt. Đảm bảo độ đối xứng giữa câu hỏi và câu trả lời để đạt được tính công bằng trong bộ dữ liệu.

+ Tiến hành xây dựng một mô hình Question Answering trong lĩnh vực giáo dục tiếng Việt: Nhóm sẽ áp dụng phương pháp tinh chỉnh (fine-tuning) trên mô hình đã được huấn luyện sẵn, sử dụng tập dữ liệu tiếng Việt, nhằm phát triển một mô hình Question Answering phù hợp với tiếng Việt.

+ Xây dựng ứng dụng Question Answering thân thiện với người dùng: Nhóm đặt mục tiêu phát triển một ứng dụng hỏi đáp có giao diện dễ sử dụng, đáp ứng được các chức năng cơ bản và hỗ trợ cuộc sống hàng ngày của người dùng.

### Kết quả dự kiến

Dự án mong muốn tạo ra một bộ dữ liệu SQuAD 2.0 chất lượng và đáng tin cậy cho tiếng Việt. Đạt được thời gian đáp ứng nhanh chóng, đảm bảo rằng người dùng nhận được câu trả lời trong khoảng thời gian ngắn. Kết quả đánh giá độ chính xác của hệ thống trên bộ dữ liệu kiểm thử, đảm bảo rằng mô hình đưa ra câu trả lời chính xác và phản ánh ý định của người dùng. Giao diện người dùng được thiết kế một cách thân thiện, giúp người dùng tương tác một cách tự nhiên và thuận tiện. Tiến hành thử nghiệm và đánh giá trên môi trường thực tế để đảm bảo rằng hiệu năng trên bảng thử nghiệm được phản ánh trong điều kiện sử dụng thực tế.

### Phạm vi nghiên cứu

Quá trình huấn luyện và tinh chỉnh mô hình sẽ được thực hiện trong phạm vi ngôn ngữ tiếng Việt. Mô hình có khả năng xử lý và trả lời các câu hỏi liên quan đến chủ đề văn học phổ thông dựa trên đầu vào được cung cấp.

Phạm vi nghiên cứu về hệ thống hỏi đáp tự động trong lĩnh vực giáo dục tiếng Việt có thể bao gồm nhiều khía cạnh quan trọng để phát triển một ứng dụng hiệu quả và linh hoạt:

+ Thu thập và xây dựng cơ sở dữ liệu chất lượng về giáo dục tiếng Việt, bao gồm tài liệu giảng dạy, sách giáo trình, bài giảng, và tài liệu học.

+ Sử dụng mô hình học sâu có sẵn để tối ưu hóa khả năng hiểu ngôn ngữ tiếng Việt.

+ Phát triển giao diện người dùng thân thiện, cho phép người dùng tương tác một cách tự nhiên và thuận tiện.

Mọi hệ thống hỏi đáp tự động sử dụng mô hình học sâu đều có thể gặp phải một số hạn chế:

+ Độ dài chuỗi đầu vào bị giới hạn.

+ Khả năng diễn đạt tiếng Việt chính xác và tự nhiên là một thách thức, đặc biệt là khi đối mặt với ngôn ngữ không chính thức hoặc ngôn ngữ học thuật.

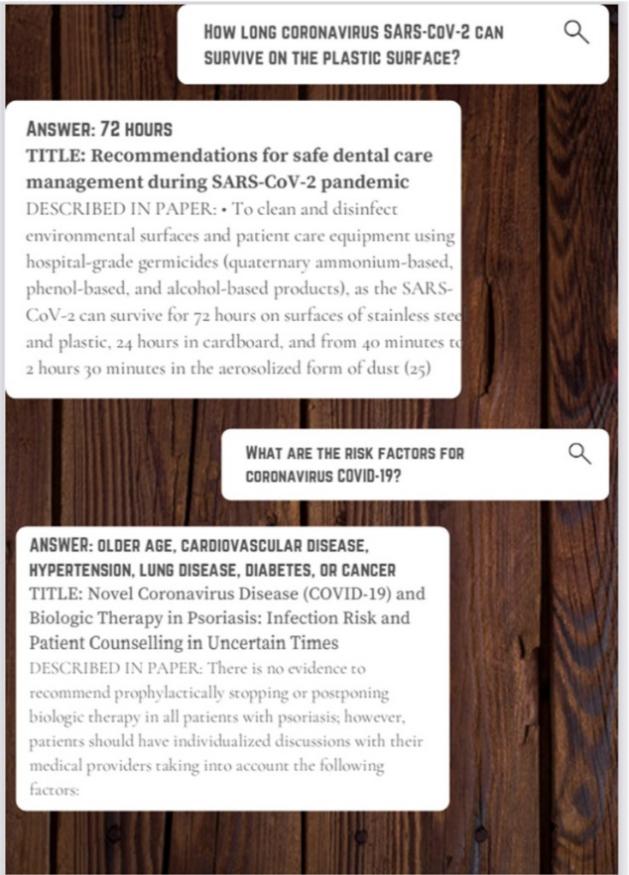
+ Dữ liệu tiếng Việt có thể không phong phú như dữ liệu tiếng Anh, điều này có thể ảnh hưởng đến khả năng huấn luyện và chất lượng của mô hình.

# KIẾN THỨC NỀN TẢNG

## Công trình liên quan

Nhóm đã tìm hiểu các công trình liên quan đến hệ thống hỏi đáp và đưa ra cái nhìn tổng quát. Từ đó rút ra những kiến thức để xây dựng mô hình cho hệ thống hỏi đáp phục vụ cho nhiệm vụ giáo dục phổ thông:

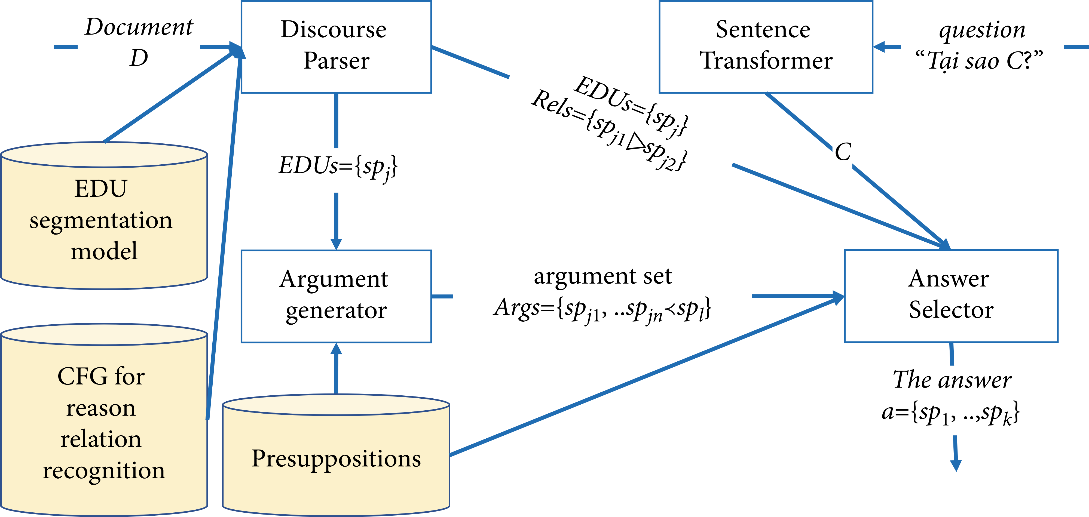
Một bài toán tìm kiếm tài liệu liên quan đến COVID – 19 (COBERT: COVID-19 Question Answering System Using BERT) trong số 59 nghìn tài liệu. Sử dụng công cụ truy xuất bộ tạo vector TF-IDF và trình đọc pre-trained của BERT, cụ thể sử dụng DistilBERT (tức là Distilled-BERT) cho nhiệm vụ tinh chỉnh (fine-tuning) trên tập dữ liệu SQuAD 1.1 để tạo ra câu trả lời ngắn gọn sau khi xếp hạng (Ranker) như trong hình 2.1 [1].



Hình 2.1 Ví dụ về đầu ra của hệ thống COBERT với truy vấn đầu vào

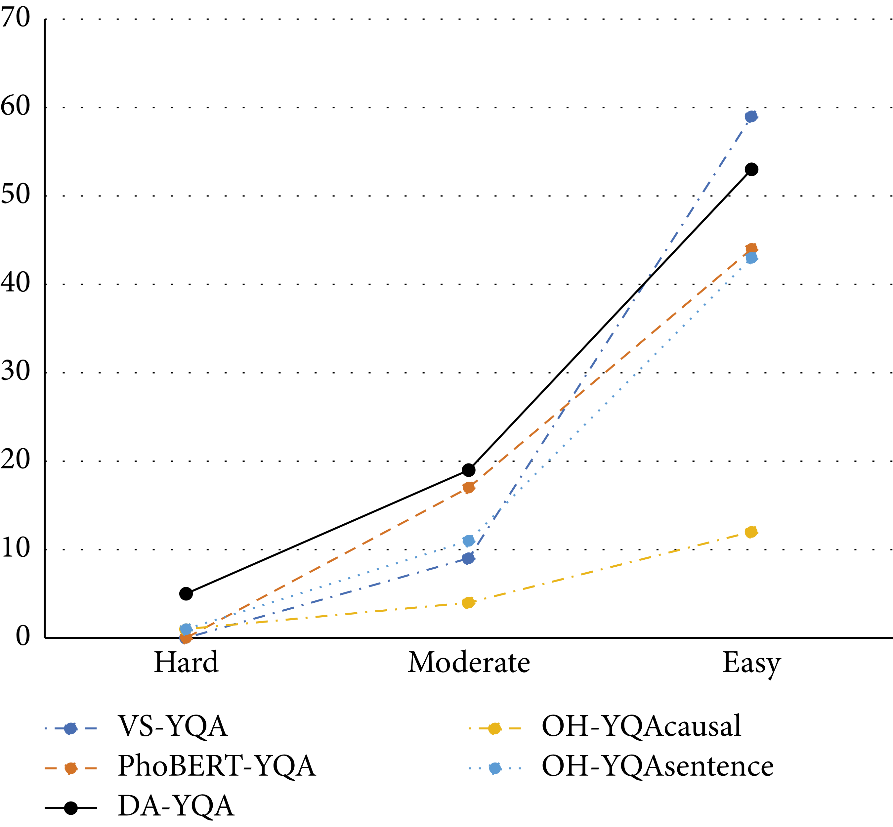
(Nguồn: (“COBERT: COVID-19 Question Answering System Using BERT,”, Fig. 5, 2023))

Một công trình khác Xây dựng hệ thống kết hợp diễn ngôn và lập luận để trả lời câu hỏi tại sao trong tiếng Việt (Building a Discourse-Argument Hybrid System for Vietnamese Why-Question Answering) tự xây dựng với 3 phần chính bộ phân tích diễn ngôn (discoure parser), bộ tạo đối số (argument generator) và bộ chọn câu trả lời (answer selector) cùng bộ bộ biến đổi câu (sentense transformer) như trong hình 2.6. Họ so sánh với các mô hình khác so với mô hình của họ là DA-YQA trên tập test SQuAD 1.1 được dịch ra tiếng Việt như hình 2.7 bên dưới, cho kết quả suy luận tốt hơn [2].



Hình 2.2 Mô hình hệ thống QA lai diễn ngôn-tranh luận tiếng Việt.

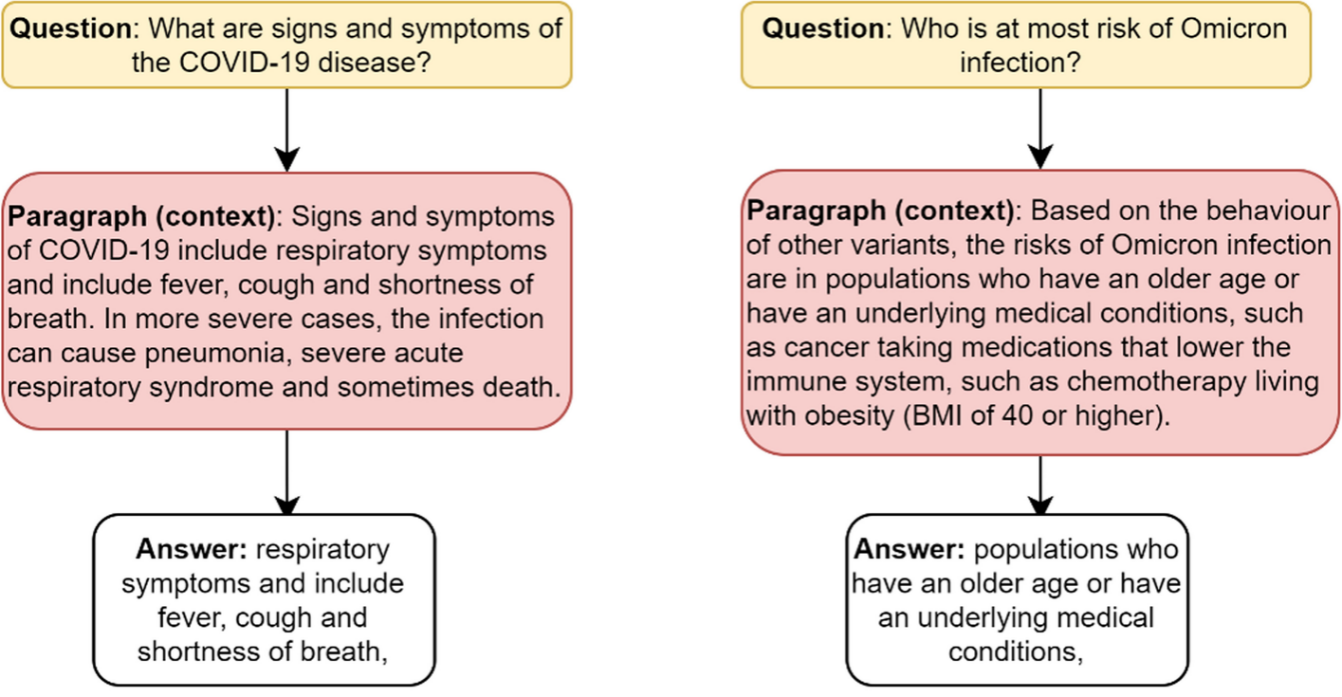
(Nguồn: (“Building a Discourse-Argument Hybrid System for Vietnamese Why-Question Answering”, Figture 4, 2021))



Hình 2.3 Số câu trả lời chấp nhận được theo nhóm câu hỏi của các mô hình.

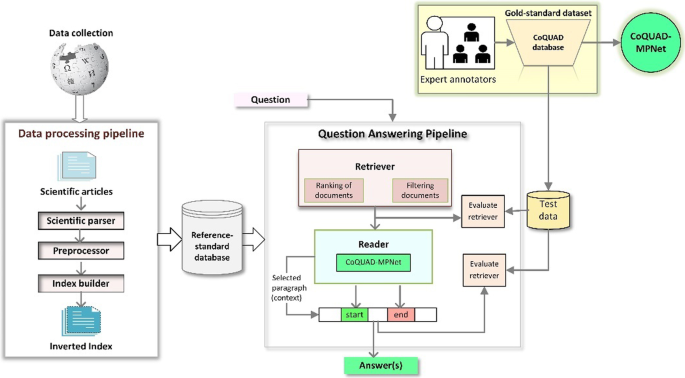
(Nguồn: (“Building a Discourse-Argument Hybrid System for Vietnamese Why-Question Answering”, Figture 8, 2021))

Một công trình nữa về COVID-19 là CoQUAD: hệ thống tập dữ liệu trả lời câu hỏi về COVID-19, hỗ trợ nghiên cứu, đo điểm chuẩn và thực hành (CoQUAD: a COVID-19 question answering dataset system, facilitating research, benchmarking, and practice). CoQUAD (COVID-19 question answering dataset) được chuẩn bị theo định dạng SQuAD 2.0, nhiệm vụ hệ thống là trích xuất câu trả lời từ văn bản được đưa ra cho một câu hỏi như hình 2.8. Hệ thống dữ liệu từ CORD-19 và LitCOVID cùng với CoQUAD, phần Retriever được đào tạo về thuật toán BM25 để tìm kiếm, tham chiếu tài liệu dựa trên câu hỏi liên quan COVID-19, phần Reader dựa trên mô hình Transformer là MPNet để tìm câu trả lời từ tài liệu được truy xuất như hình 2.9. Kết quả CoQUAD vượt trội hơn các mô hình trước đó, như trong hình 2.10 [3].



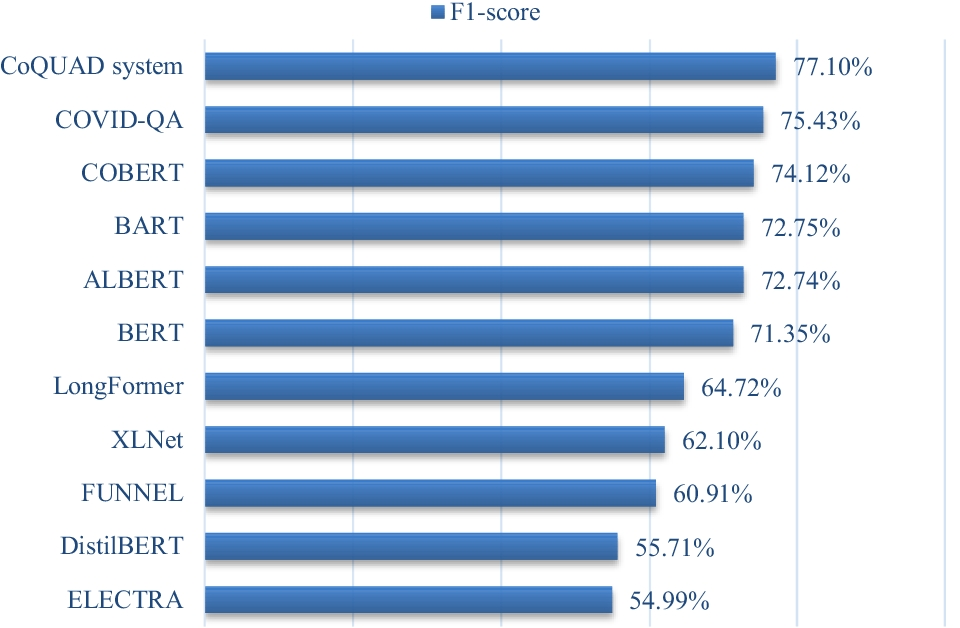
Hình 2.4 Ví dụ về hệ thống QA khai thác bao gồm câu hỏi, ngữ cảnh và câu trả lời.

(Nguồn: (“CoQUAD: a COVID-19 question answering dataset system, facilitating research, benchmarking, and practice”, Fig. 2, 2022))



Hình 2.5 Kiến trúc hệ thống CoQUAD.

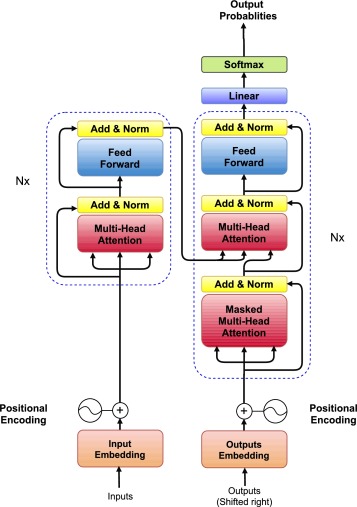
(Nguồn: (“CoQUAD: a COVID-19 question answering dataset system, facilitating research, benchmarking, and practice”, Fig. 3, 2022))



Hình 2.6 Điểm F1 của tất cả các mô hình.

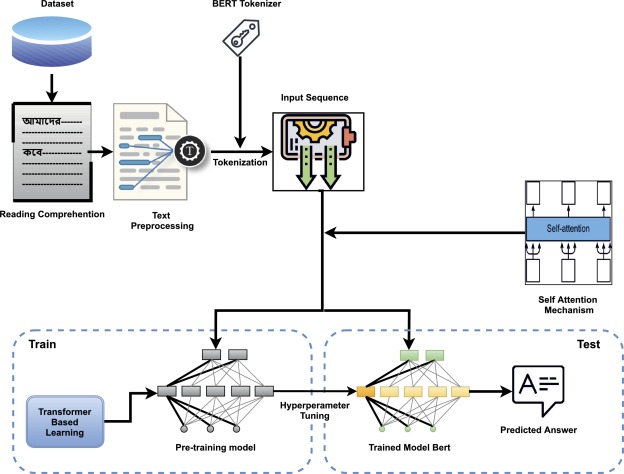
(Nguồn: (“CoQUAD: a COVID-19 question answering dataset system, facilitating research, benchmarking, and practice”, Fig. 7, 2022))

Một công trình dựa vào Transformer là Hệ thống trả lời câu hỏi dựa trên đọc hiểu bằng ngôn ngữ Bangla với phương pháp học tập dựa trên máy biến áp (Reading comprehension-based question answering system in Bangla language with transformer-based learning), họ phát triển hệ thống dựa vào RC (Reading comprehensive) trong tập dữ liệu 3636 bài đọc hiểu cùng với các câu hỏi đáp. Họ áp dụng mô hình mạng thần kinh sâu dựa trên transformer (transformer-based deep neural network) sử dụng cơ chế của attention với bộ mã hóa (encoder) và bộ giải mã (decoder) để xử lý đầu vào tuần tự như hình 2.11, các kiến trúc LSTM, Bi-LSTM with attention, RNN, ELECTRA và BERT cho việc đào tạo dữ liệu như hình 2.12. Pre-training BERT. Kết quả BERT tốt hơn, tiếp đó là ELECTRA như hình 2.13 [4].



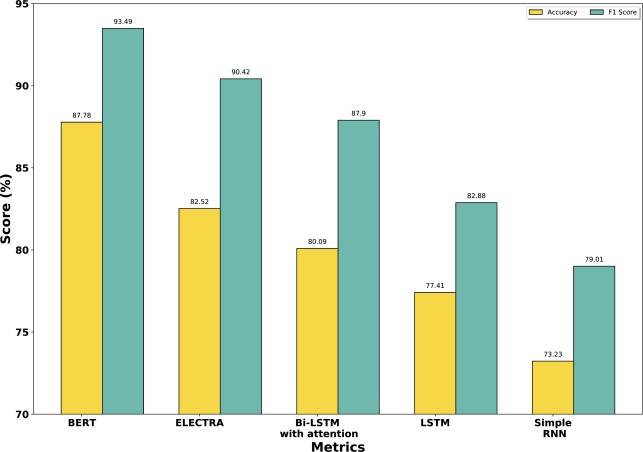
Hình 2.7 Kiến trúc mô hình Transformer.

(Nguồn: (“Reading comprehension based question answering system in Bangla language with transformer-based learning”, Figture 1, 2022))



Hình 2.8 Khung đề xuất cho Bangla RC từ khâu tiền xử lý dữ liệu đến dự đoán đáp án của RC.

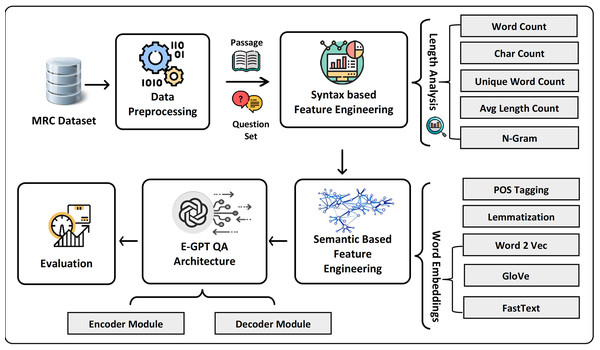
(Nguồn: (“Reading comprehension based question answering system in Bangla language with transformer-based learning”, Figture 4, 2022))



Hình 2.9 Bản so sánh về Độ chính xác và điểm F1 của một số mô hình

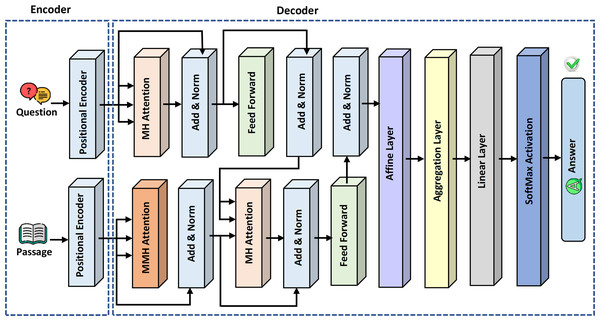
(Nguồn: (“Reading comprehension based question answering system in Bangla language with transformer-based learning”, Figture 9, 2022))

Một công trình khác cũng sử dụng transformer được đào tạo trước là Về việc giải quyết sự mơ hồ về văn bản và sự mơ hồ về ngữ nghĩa trong việc trả lời câu hỏi dựa trên MRC bằng cách sử dụng các biến áp tổng quát được đào tạo trước (On solving textual ambiguities and semantic vagueness in MRC based question answering using generative pre-trained transformers), trong đó MRC-QA (Machine Reading Comprehension) đã được kết quả cải thiện nhờ vào thuật toán học sâu hai chiều và bộ dữ liệu lớn từ việc ánh xạ và truy xuất câu trả lời như hình 2.14. Mô hình ExtGPT-QA đã giải quyết vấn đề việc đọc hiểu đoạn văn dài và tạo câu trả lời trong MRC trừu tượng từ việc sử dụng bộ mã hóa giải quyết sự mô hồ trong văn bản và bộ giải mã cùng cơ chế attention giải quyết mơ hồ về ngữ nghĩa như hình 2.15. Sử dụng bộ ba dữ liệu SQuAD, Wiki-QA và News-QA cho thực nghiệm. Kết quả mô hình ExtGPT-QA mang lại sự vượt trội hơn với F1 score là 93.25% và exact match là 90.52% [5].



Hình 2.10 Tổng quan về khung đề xuất cho hệ thống trả lời câu hỏi của MRC

(Nguồn: (“On solving textual ambiguities and semantic vagueness in MRC based question answering using generative pre-trained transformers”, Figture 2, 2023))



Hình 2.11 Kiến trúc của ExtGPT-QA được đề xuất cho khả năng đọc hiểu của máy (MRC).

(Nguồn: (“On solving textual ambiguities and semantic vagueness in MRC based question answering using generative pre-trained transformers “, Figture 4, 2023))

Từ các công trình trên, vì các mô hình hỏi đáp cần có ngữ cảnh đi kèm với các câu hỏi và câu trả lời, nhóm đã quyết định tạo ra một tập dữ liệu SQuAD 2.0 dành cho tiếng Việt. Nhóm muốn xây dựng ứng dụng cho hệ thống hỏi đáp dành riêng cho giáo dục tiếng Việt, nên không tập trung vào nghiên cứu mà sẽ dùng các mô hình đã được huấn luyện sẵn (pre-training) cho việc sinh ra câu trả lời và điều chỉnh các trọng số (fine-tuning) của mô hình để cải thiện kết quả.

## Kiến thức nền tảng

Nhóm sẽ trình bày kiến thức nền tảng liên quan đến hỏi đáp tự động, bao gồm kho dữ liệu SQuAD 2.0, phân loại ý định, trả lời tự động, học chuyển giao, mô hình BERT.

### Kho dữ liệu SQuAD 2.0

SQuAD viết tắt là Stanford Question Answering Dataset là bộ dữ liệu đọc hiểu được tổng hợp gồm hơn 100.000 các câu hỏi từ các bài Wikipedia bằng tiếng Anh. SQuAD chứa 107.785 cặp câu hỏi đáp từ 536 bài viết trên Wikipedia.

SQuAD 2.0 (Stanford Question Answering Dataset 2.0) là một bộ dữ liệu benchmark cho nhiệm vụ trả lời câu hỏi máy tính (Question Answering - QA) trong lĩnh vực Xử lý Ngôn Ngữ Tự Nhiên (Natural Language Processing - NLP). Được phát triển và công bố bởi nhóm nghiên cứu tại Đại học Stanford, SQuAD 2.0 mở rộng từ phiên bản trước của nó, SQuAD 1.1, bằng cách bổ sung khoảng 50,000 câu hỏi mà câu trả lời không xuất hiện trong bài văn cung cấp, đòi hỏi mô hình không chỉ phải tìm câu trả lời chính xác từ một đoạn văn mà còn phải nhận biết khi nào không có thông tin trong đoạn văn để trả lời câu hỏi đó.

Trong dự án này nhóm tập trung nghiên cứu, tham khảo và xây dựng dữ liệu dựa trên bộ dữ liệu SQuAD 2.0, bộ dữ liệu được chú thích bằng tay và tất cả câu trả lời là một phần nhỏ trong đoạn ngữ cảnh thay vì một tập hợp giới hạn nhiều lựa chọn. Bảng 2.1 giải thích thuật ngữ trong tệp tin JSON của SQuAD 2.0. Bên dưới là ví dụ về cặp câu hỏi đáp trong một đoạn văn của kho dữ liệu SQuAD 2.0.

{

"version": "v2.0",

"data": [

{

"title": "Tổng quan văn học Việt Nam",

"paragraphs": [

{

"qas":[

{

"question": "Các sáng tác ngôn từ văn học Việt Nam bao gồm bộ phận nào?",

"id": "e46ad3d1c7fd57eabf4cc068ac5159a1",

"answers": [

{

"text": "Văn học dân gian và văn học viết",

"answer\_start": 97

}

],

"is\_impossible": false

},

{

"question": "Văn học dân gian là gì?",

"id": "c7d8bb180c4b5eb68410ba9ca326f767",

"answers": [

{

"text": "Sáng tác tập thể và truyền miệng của nhân dân lao động",

"answer\_start": 151

}

],

"is\_impossible": false

}

],

"context": "Văn học Việt Nam bao gồm các sáng tác ngôn từ với hai bộ phận lớn có quan hệ mật thiết với nhau: văn học dân gian và văn học viết. Văn học dân gian là sáng tác tập thể và truyền miệng của nhân dân lao động. Cũng có trường hợp người trí thức tham gia sáng tác văn học dân gian, nhưng các sáng tác đó phải tuân thủ những đặc trưng của văn học dân gian và trở thành tiếng nói tình cảm chung của nhân dân."

}

]

}

]

}

Bảng 1 Cấu trúc dữ liệu SQuAD dạng JSON

|  |  |
| --- | --- |
| Tên trường | Ý nghĩa |
| version | Thông tin phiên bản tập dữ liệu SQuAD. |
| data | Thông tin toàn bộ nội dung. |
| title | Nội dung chủ đề của các đoạn văn. |
| paragraphs | Thông tin các đoạn văn cùng bộ câu hỏi đáp. |
| qas | Thông tin các cặp câu hỏi đáp trong đoạn văn. |
| question | Nội dung câu hỏi liên quan đến đoạn văn. |
| id | Mã id duy nhất cho mộĩ cặp câu hỏi đáp. |
| answers | Thông tin câu trả lời và vị trí bắt đầu của nó trong đoạn văn. |
| text | Nội dung câu trả lời. |
| answer\_start | Nội dung vị trí bắt đầu của câu trả lời trong đoạn văn. |
| is\_impossible | Nội dung cho biết câu hỏi có câu trả lời hay không. |
| context | Nội dung đoạn văn liên quan câu hỏi và chứa câu trả lời. |

SQuAD 1.1 và SQuAD 2.0 là hai phiên bản của bộ dữ liệu SQuAD (Stanford Question Answering Dataset), được sử dụng để huấn luyện và đánh giá các mô hình hỏi-đáp tự động. Dưới đây là một so sánh giữa hai phiên bản này:

Về kích thước và độ phức tạp: SQuAD 2.0 lớn hơn và phức tạp hơn so với SQuAD 1.1. SQuAD 2.0 đặc biệt khó hơn SQuAD 1.0 do yêu cầu mô hình phải xác định được cả những câu hỏi không thể trả lời, nâng cao yêu cầu về khả năng hiểu ngôn ngữ và lý luận của mô hình.

Loại câu hỏi: SQuAD 1.1 tập trung vào việc trả lời các câu hỏi dựa trên một đoạn văn bản cụ thể. Các câu hỏi trong SQuAD 1.1 đều có câu trả lời trong đoạn văn bản tương ứng. Trong khi đó, SQuAD 2.0 mở rộng phạm vi câu hỏi bằng cách đưa ra các câu hỏi "impossible" mà không có câu trả lời trong đoạn văn bản, đòi hỏi các mô hình phải nhận diện được và đưa ra câu trả lời phù hợp. Cả hai đều rất quan trọng trong việc nghiên cứu và phát triển các hệ thống trả lời câu hỏi tự động, nhưng SQuAD 2.0 cung cấp một bước kiểm tra thực tế hơn cho các mô hình về khả năng xử lý thông tin không đầy đủ hoặc không rõ ràng.

Thách thức đối với các mô hình hỏi-đáp: SQuAD 2.0 mang đến một thách thức lớn hơn cho các mô hình hỏi-đáp tự động so với SQuAD 1.1. Với sự xuất hiện của các câu hỏi "impossible", mô hình phải có khả năng xác định khi nào một câu hỏi không có câu trả lời thay vì đưa ra một câu trả lời sai.

Độ đa dạng và phân phối nguồn gốc: SQuAD 2.0 có nguồn gốc đa dạng hơn so với SQuAD 1.1, bao gồm các bài báo từ các nguồn báo cáo tin tức khác nhau. Điều này giúp đảm bảo rằng các mô hình được huấn luyện trên SQuAD 2.0 có khả năng xử lý các nguồn thông tin đa dạng và không chỉ giới hạn trong một nguồn duy nhất như Wikipedia. SQuAD 2.0 mô phỏng chặt chẽ hơn các tình huống thực tế, nơi mà không phải lúc nào thông tin cũng đầy đủ hoặc trực tiếp liên quan đến câu hỏi.

Tóm lại, SQuAD 2.0 là một phiên bản mở rộng và phức tạp hơn của SQuAD 1.1, đưa ra các thách thức mới cho các mô hình hỏi-đáp tự động. Việc sử dụng SQuAD 2.0 giúp nâng cao khả năng hiểu và trả lời các câu hỏi của các mô hình hỏi-đáp tự động.

### Phân loại ý định

Phân loại ý định (Intent Classification) là một tác vụ quan trọng trong Xử lý Ngôn Ngữ Tự Nhiên (NLP), nhằm xác định mục đích hoặc ý định của người dùng qua văn bản mà họ nhập vào. Trong các hệ thống tương tác như chatbot hoặc trợ lý ảo, phân loại ý định giúp máy tính hiểu được người dùng muốn gì từ một câu hỏi hoặc mệnh lệnh, từ đó cung cấp câu trả lời hoặc hành động phù hợp.

Đầu vào thường là câu hỏi hoặc mệnh lệnh từ người dùng, sử dụng các kỹ thuật NLP để tiền xử lý dữ liệu như tokenization, lemmatization, loại bỏ stop words… Phân loại bằng cách sử dụng các mô hình machine learning hoặc deep learning để xác định ý định dựa trên dữ liệu đã được xử lý và đầu ra là nhãn ý định, giúp hệ thống xác định được cách xử lý tiếp theo.

Trong một hệ thống trả lời câu hỏi, phân loại ý định giữ vai trò thiết yếu bằng cách giúp hệ thống phân biệt giữa các loại câu hỏi khác nhau, ví dụ: câu hỏi về thông tin cụ thể, yêu cầu giải thích, hoặc yêu cầu hướng dẫn thực hiện một hành động. Từ đó, hệ thống có thể chọn cách tiếp cận phù hợp để tìm và cung cấp câu trả lời.

Phân loại ý định giúp cải thiện khả năng hiểu và xử lý yêu cầu của người dùng, từ đó nâng cao hiệu suất tổng thể của hệ thống. Giúp hệ thống có thể xử lý đa dạng các loại câu hỏi và yêu cầu, làm tăng khả năng tương tác và hài lòng của người dùng. Phân loại ý định cho phép hệ thống tập trung tài nguyên vào việc tìm kiếm và xử lý thông tin theo nhu cầu cụ thể, giảm thiểu việc lãng phí tài nguyên vào các tác vụ không cần thiết.

### Trả lời tự động

Trả lời câu hỏi (QA) là lĩnh vực nghiên cứu kết hợp truy xuất thông tin (IR), trích xuất thông tin (IE) và xử lý ngôn ngữ tự nhiên (NLP) [8]. Hệ thống trả lời câu hỏi (QAS) là loại hệ thống truy xuất thông tin có khả năng cung cấp câu trả lời cho các truy vấn dưới dạng ngôn ngữ tự nhiên, thay vì chỉ hiển thị danh sách các tài liệu liên quan (truy xuất tài liệu). Bằng cách cung cấp một giao diện đối thoại cho hệ thống trả lời câu hỏi (QAS), người dùng sẽ được khuyến khích và hỗ trợ để đặt nhiều câu hỏi có liên quan và giải quyết các yêu cầu cần làm rõ của họ [9].

Hệ thống trả lời câu hỏi (QAS) là một dạng của trí tuệ nhân tạo được phát triển để đáp ứng các câu hỏi được đặt ra bằng ngôn ngữ tự nhiên của con người. Các hệ thống này có thể được xây dựng để giải quyết các câu hỏi liên quan đến nhiều chủ đề hoặc được tinh chỉnh để chuyên về một lĩnh vực cụ thể, như chẩn đoán y khoa hoặc dịch vụ hỗ trợ khách hàng; nhóm sẽ hướng đến lĩnh vực giáo dục phổ thông trong dự án này. Hệ thông trả lời câu hỏi (QAS) được phân thành các loại chung sau đây:

* Extractive Question Answering: Hệ thống tìm câu trả lời chính xác cho một câu hỏi trong một đoạn văn hoặc tài liệu cụ thể. Cơ bản, nó trích xuất một phần của văn bản chứa câu trả lời. Ví dụ: nếu có một bài viết về Hoa Kỳ và câu hỏi là "Thủ đô của Hoa Kỳ tên gì?", hệ thống sẽ xác định và trả về phần cụ thể của văn bản, ví dụ: "Washington, D.C.".
* Generative Question Answering: Hệ thống này sinh ra câu trả lời thay vì trích xuất nó từ văn bản hiện có. Điều này tạo ra sự phức tạp hơn vì nó yêu cầu việc hiểu câu hỏi và tạo ra các câu mới, những câu mà có thể không được chỉ rõ trong dữ liệu đầu vào. Các hệ thống này thường dựa trên mô hình tuần tự (sequence-to-sequence models) để tạo ra câu trả lời dựa trên câu hỏi và ngữ cảnh đầu vào.

Các hệ thống trả lời câu hỏi (QAS) ngày càng trở nên quan trọng trong nhiều ứng dụng như tác nhân trò chuyện hoặc chatbot, công cụ tìm kiếm, trợ lý ảo (như Siri, Alexa, Google Assistant) và trong các lĩnh vực đòi hỏi việc truy cập thông tin một cách nhanh chóng và chính xác. Hệ thống QA là một lĩnh vực đang được nghiên cứu mạnh mẽ và các tiến bộ gần đây tập trung vào việc nâng cao độ chính xác, khả năng hiểu các câu hỏi phức tạp, xử lý sự mơ hồ, hiểu ngữ cảnh và duy trì hội thoại nhiều lượt trong đó ngữ cảnh và tham chiếu đến các truy vấn trước đó được duy trì.

Mô hình Machine Reading Comprehension (MRC) thuộc lĩnh vực xử lý ngôn ngữ tự nhiên (NLP), với mục đích chính là tìm hiểu và đưa ra câu trả lời cho các câu hỏi dựa trên một đoạn văn đã được cung cấp. MRC có thể phân tích và nắm bắt được ngữ cảnh của đoạn văn, bao gồm việc nhận diện các thực thể, mối quan hệ và thông tin quan trọng trong văn bản.

Sau khi xử lý dữ liệu, mô hình này có khả năng cung cấp câu trả lời cho các câu hỏi dựa vào nội dung có trong đoạn văn. Câu trả lời có thể được thể hiện dưới dạng việc chọn một phần văn bản cụ thể, tạo ra một câu trả lời hoàn toàn mới, hoặc xác định loại câu hỏi vào một số phân loại trả lời đã được định trước. MRC được áp dụng trong nhiều lĩnh vực như hệ thống trợ lý ảo, công cụ tìm kiếm, và các ứng dụng trong giáo dục, nơi mà khả năng hiểu và trả lời câu hỏi dựa trên nguồn thông tin xác định là quan trọng.

Mô hình MRC sử dụng các mô hình học sâu như BERT được huấn luyện trên tập dữ liệu lớn bao gồm các cặp câu hỏi và đoạn văn bản, dựa trên các mô hình học sâu để hiểu ngữ cảnh cụ thể của văn bản. MRC thường tập trung hơn so với QA, giới hạn trong việc hiểu và trả lời từ một đoạn văn bản cụ thể. MRC tập trung vào việc trích xuất câu trả lời từ một đoạn văn bản cụ thể, trong khi QA có thể bao gồm việc tìm kiếm câu trả lời từ nhiều nguồn thông tin khác nhau.

Mô hình MRC làm việc dựa trên các kỹ thuật học sâu như BERT, được rèn giũa qua việc huấn luyện trên các bộ dữ liệu lớn chứa đựng nhiều cặp câu hỏi và đoạn văn. Nó sử dụng những mô hình học sâu này để phân tích và hiểu rõ ngữ cảnh trong văn bản. MRC thường được thiết kế để tập trung hơn so với QA, giới hạn trong việc phân tích và trả lời các câu hỏi từ một đoạn văn bản cụ thể. Trong khi MRC chủ yếu nhấn mạnh vào việc trích xuất câu trả lời từ đoạn văn bản đã cho, QA lại mở rộng phạm vi tìm kiếm câu trả lời từ nhiều nguồn thông tin khác nhau.

### Học chuyển giao

Học chuyển giao (Transfer Learning) là một phương pháp học máy giúp một mô hình sử dụng lại kiến thức miền đã học từ một mô hình khác đã được huấn luyện hoàn chỉnh. Học chuyển giao rất cần thiết trong việc giảm nhu cầu và nỗ lực thu thập lại dữ liệu đào tạo để xây dựng lại mô hình từ các tính năng (features) và sự phân phối dữ liệu khác nhau so với dữ liệu cũ hoặc dữ liệu có thể lỗi thời [10].

Học chuyển giao được áp dụng trong các nhiệm vụ trả lời câu hỏi (QA) bằng cách sử dụng mô hình đã được huấn luyện trên một tập dữ liệu lớn và sau đó điều chỉnh nó trên một tập dữ liệu cụ thể cho QA. Phương pháp này rất phổ biến trong xử lý ngôn ngữ tự nhiên (NLP) và đóng vai trò quan trọng trong việc phát triển lĩnh vực này, đặc biệt là với sự xuất hiện của các mô hình như BERT, GPT và các mô hình kế thừa của chúng.

* Pre-Training: Quá trình bắt đầu với việc huấn luyện một mô hình trên một kho văn bản lớn. Giai đoạn này không được giám sát và tập trung vào các nhiệm vụ hiểu ngôn ngữ chung như dự đoán từ tiếp theo trong câu (language modeling) hoặc che giấu một số từ và dự đoán chúng dựa trên ngữ cảnh do các từ khác cung cấp (masked language modeling).
* Fine-Tuning: Sau khi đào tạo trước, mô hình sẽ được điều chỉnh trên tập dữ liệu QA được giám sát. Tập dữ liệu QA bao gồm các cặp câu hỏi và câu trả lời, và đôi khi còn có ngữ cảnh bổ sung nơi câu trả lời xuất hiện. Trong giai đoạn này, mô hình học các tính năng đặc biệt cho nhiệm vụ như việc hiểu câu hỏi, tìm kiếm thông tin liên quan và tạo ra hoặc trích xuất câu trả lời.
* Task Adaptation: Các nhiệm vụ QA có thể được phân loại thành hai hạng mục chính: QA khai thác (extractive QA) và QA tổng hợp (generative QA). Trong QA khai thác, mục tiêu là xác định câu trả lời chính xác trong một đoạn văn cụ thể. Trong khi đó, QA tổng hợp yêu cầu mô hình tạo ra câu trả lời mà có thể không được chỉ rõ trong đoạn văn. Phương pháp học chuyển giao cần phải điều chỉnh cách tiếp cận tinh chỉnh của nó để đáp ứng yêu cầu cụ thể của từng loại nhiệm vụ.
* Evaluation and Iteration: Sau khi điều chỉnh, hiệu suất của mô hình được kiểm tra trên một tập dữ liệu thử nghiệm riêng biệt, mà không được sử dụng trong quá trình huấn luyện. Dựa trên kết quả, các cải tiến có thể được thực hiện liên tục bằng cách điều chỉnh kiến trúc mô hình (model architecture), thay đổi tập dữ liệu điều chỉnh (fine-tuning dataset) hoặc thay đổi quy trình điều chỉnh (fine-tuning process).

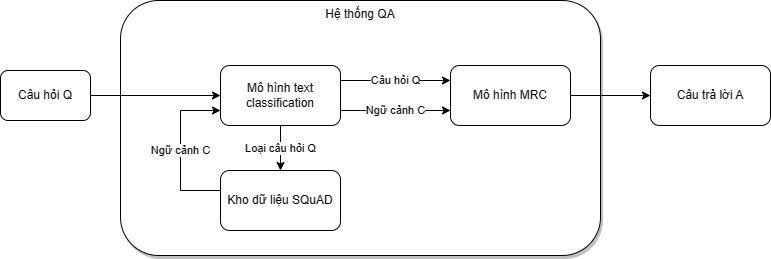
Học chuyển giao trong QA rất hiệu quả vì nó cho phép tận dụng lượng lớn kiến thức chung về ngôn ngữ mà các mô hình thu thập trong quá trình đào tạo trước. Điều này có thể cải thiện đáng kể hiệu suất của chúng đối với các nhiệm vụ QA, ngay cả khi dữ liệu tinh chỉnh cho các nhiệm vụ này tương đối nhỏ. Phương pháp này đã được chứng minh hiệu quả trong các điểm chuẩn NLP (NLP benchmarks) như Bộ dữ liệu trả lời câu hỏi Stanford (SQuAD) hoặc bộ dữ liệu Câu hỏi tự nhiên (Natural Questions dataset).

### Mô hình BERT

BERT là một mô hình ngôn ngữ hiệu quả do Google phát triển vào năm 2018, đã tạo ra một cuộc cách mạng trong lĩnh vực xử lý ngôn ngữ tự nhiên (NLP). BERT sử dụng kiến trúc Transformer [11], một loại mô hình học sâu dựa trên cơ chế attention. Điểm nổi bật của BERT là khả năng hiểu biểu diễn từ ngữ theo cả hai hướng trong văn bản. Cụ thể, BERT sử dụng một biến thể của cơ chế attention, cho phép mô hình nắm bắt được ngữ cảnh của từng từ dựa trên cả hai phía của nó bằng cách kết hợp giữa mục tiêu mô hình hóa ngôn ngữ đeo mặt nạ (Masked Language Modeling) và dự đoán câu tiếp theo (Next Sentence Prediction) trên một kho văn bản lớn bao gồm Toronto Book Corpus và Wikipedia [12]. Sau khi được đào tạo trước, BERT có thể được tinh chỉnh cho các nhiệm vụ cụ thể như phân loại văn bản, phân loại thực thể đặt tên (NER), và đặc biệt là bài toán Trả lời câu hỏi (QA).

# MÔ HÌNH BÀI TOÁN

## Mô hình tổng quát

Hình 3.1 Mô hình hoạt động

Mô hình tổng quát trong hình 3.1 bắt đầu bằng việc phân loại câu hỏi Q thông qua phương pháp phân loại văn bản, sử dụng tập dữ liệu SQuAD để xác định ngữ cảnh C phù hợp. Sau khi xác định được ngữ cảnh C, thông tin này cùng với câu hỏi Q ban đầu được truyền đến mô hình MRC. Mô hình MRC, với khả năng hiểu và xử lý thông tin chi tiết về ngữ cảnh, sẽ tiến hành phân tích và trả lời câu hỏi Q dựa trên ngữ cảnh C cụ thể đó. Quá trình này cho phép mô hình đạt được hiệu quả cao trong việc cung cấp câu trả lời chính xác, vì nó kết hợp cả bước xác định ngữ cảnh thông qua phân loại câu hỏi và bước phân tích sâu rộng của MRC.

Để thấu hiểu mô hình tổng hợp từ việc phân loại câu hỏi, qua tìm kiếm ngữ cảnh từ SQuAD, đến việc trả lời bằng mô hình MRC, quá trình này có thể được phân tích thành từng bước cụ thể như sau:

* Phân loại câu hỏi: Mục đích là xác định ngữ cảnh hoặc chủ đề liên quan đến câu hỏi. Dùng mô hình học máy hoặc học sâu để phân loại câu hỏi theo chủ đề. Mô hình này được rèn luyện trên tập dữ liệu lớn để có khả năng nhận diện loại câu hỏi.
* Sử dụng tập dữ liệu SQuAD: Mục đích lựa chọn ngữ cảnh phù hợp với câu hỏi. SQuAD, một bộ sưu tập dữ liệu lớn, bao gồm nhiều đoạn văn và câu hỏi tương ứng. Tùy thuộc vào loại của câu hỏi đã được xác định thông qua phân loại, mô hình sẽ thực hiện việc tìm kiếm trong SQuAD để chọn lựa ra đoạn văn bản phù hợp nhất với chủ đề được nêu trong câu hỏi. Lựa chọn ngữ cảnh phù hợp từ nhiều ngữ cảnh dựa trên việc phân tích từ khóa, ngữ nghĩa của câu hỏi, hoặc sự liên quan với ngữ cảnh hoặc kết hợp từ nhiều ngữ cảnh.
* Kết hợp câu hỏi và ngữ cảnh với mô hình MRC: Mục đích phân tích và trả lời câu hỏi dựa trên ngữ cảnh cụ thể. Mô hình MRC nhận đầu vào là câu hỏi và đoạn văn từ SQuAD. Mô hình này được thiết kế để hiểu sâu về ngữ cảnh và mối liên hệ của các yếu tố trong đoạn văn với câu hỏi.
* Xử lý và trả lời câu hỏi: Mục đích đưa ra câu trả lời chính xác dựa trên hiểu biết về ngữ cảnh. Mô hình MRC xem xét đoạn văn và câu hỏi, sau đó định hình câu trả lời thích hợp. Câu trả lời này có thể là một phần của đoạn văn, câu trả lời tự sinh dựa trên thông tin hiện có, hoặc phân loại câu hỏi vào một loạt câu trả lời cố định.

Quá trình này yêu cầu khả năng hiểu biết sâu rộng về ngữ nghĩa và ngữ cảnh, cùng với khả năng xử lý thông tin phức tạp. Nhất là trong việc kết hợp các bước để cung cấp câu trả lời chính xác và có thông tin.

## Kiến trúc ứng dụng

Hình 3.2 là kiến trúc ứng dụng cho hệ thống trả lời tự động cho giáo dục tiếng Việt. Flask cung cấp một cách đơn giản để thiết lập một web server trên máy cục bộ (localhost), điều này làm cho nó trở nên lý tưởng cho việc phát triển và kiểm thử. Khi chạy ứng dụng (py app.py), Flask sẽ khởi động một web server trên localhost (thường là http://127.0.0.1:5000/) mà nhóm có thể truy cập qua trình duyệt web để tương tác với hệ thống hỏi đáp của mình.

Jinja là một template engine cho Python, được sử dụng rộng rãi trong việc phát triển web với Flask. Template Jinja cho phép bạn tạo các trang HTML động bằng cách tích hợp dữ liệu Python vào trong mã HTML một cách sạch sẽ và linh hoạt. Sử dụng Jinja trong Flask giúp tách biệt logic ứng dụng Python với HTML, làm cho mã nguồn dễ đọc, dễ bảo trì và thân thiện với các nhà phát triển front end. Người dùng tương tác trực tiếp thông qua trình duyệt web từ việc nhập câu hỏi. Front end cũng xử lý các sự kiện do người dùng kích hoạt, như nhấp chuột và nhập liệu. Trong một ứng dụng Flask, nhóm sẽ thiết kế giao diện của mình sử dụng các mẫu HTML, và sử dụng CSS và JavaScript để cải thiện tính thẩm mỹ và tương tác.

Back end xử lý dữ liệu và logic ứng dụng thông qua nhận yêu cầu từ front end, xử lý yêu cầu đó như truy cập dữ liệu, tải mô hình từ Hungging Face và trả về kết quả cho front end để hiển thị. Flask là một framework cho Python được sử dụng để xây dựng back end, bao gồm việc viết các hàm xử lý yêu cầu (route handlers), logic ứng dụng, và tương tác với dữ liệu. Sử dụng Flask để xây dựng một hệ thống hỏi đáp trên localhost là một cách tiện lợi để phát triển, kiểm thử, và thử nghiệm ý tưởng của nhóm trước khi đưa chúng lên một môi trường sản xuất thực sự.

A diagram of a computer flow

Description automatically generated

Hình 3.2 Kiến trúc ứng dụng

# THỰC NGHIỆM

## Dữ liệu thực nghiệm

Khi thu thập dữ liệu thực nghiệm cho một mô hình học máy liên quan đến danh sách câu hỏi và trả lời về chủ đề giáo dục phổ thông từ nguồn thông tin bộ sách trắc nghiệm “Cánh diều” và “Chân trời sáng tạo”, nhóm đã tập trung vào việc đảm bảo chất lượng và độ tin cậy của dữ liệu. Xác định rõ các chủ đề giáo dục phổ thông mà nhóm muốn có mô hình của mình, ví dụ như toán học, địa lý, lịch sử, văn học… bao quát cấp độ học lớp 10 đến lớp 12 và đi vào chi tiết các chủ đề cụ thể như nguyên tử, Tháp nước Niagara, Bản Tuyên ngôn độc lập, Romeo và Juliet…

Tìm kiếm nguồn dữ liệu đáng tin cậy như sách giáo khoa, wikipedia, trắc nghiệm online từ web giáo dục được công nhận bởi cộng đồng giáo dục. Dữ liệu thu thập cả câu hỏi, câu trả lời và ngữ cảnh chứa câu trả lời. Đảm bảo rằng câu trả lời là chính xác và đã được xác minh bởi các chuyên gia giáo dục hoặc nguồn thông tin đáng tin cậy. Bao gồm một loạt các loại câu hỏi khác nhau như lựa chọn, điền vào chỗ trống, câu hỏi mở và chủ đề để tăng cường khả năng tổng quát hóa của mô hình.

Loại bỏ dữ liệu không liên quan hoặc thông tin gây nhiễu không cần thiết khỏi tập dữ liệu. Chuẩn hóa định dạng văn bản, chính tả, và cú pháp để đảm bảo tính nhất quán trong tập dữ liệu. Sử dụng các kỹ thuật tăng cường dữ liệu như paraphrasing (tái phrasing câu hỏi) để tạo ra các biến thể của câu hỏi nhằm tăng độ đa dạng và khối lượng dữ liệu huấn luyện.

Đoạn văn trong trường ngữ cảnh có độ dài khoảng từ 2 đến 5 câu để mô hình có thể hiểu được ngữ cảnh một cách chính xác. Mỗi ngữ cảnh cần đi kèm với ít nhất 5 bộ câu hỏi và câu trả lời, đảm bảo rằng quá trình huấn luyện mô hình không gặp phải tình trạng thiếu hụt dữ liệu. Câu hỏi được tạo ra liên quan đến nội dung trong trường ngữ cảnh, và câu trả lời cho câu hỏi đó có thể được tìm thấy trong văn bản. Câu trả lời cho mỗi câu hỏi nên là một phần văn bản liên tục được trích xuất trực tiếp từ nội dung ngữ cảnh, bảo đảm sự chính xác và liên quan.

Chia dữ liệu thành các tập huấn luyện, validation và kiểm tra. Điều này giúp đánh giá chính xác hiệu suất và khả năng tổng quát hóa của mô hình trên dữ liệu chưa thấy. Chuẩn bị và xử lý dữ liệu một cách cẩn thận là bước quan trọng nhất trong việc phát triển một mô hình học máy hiệu quả và đáng tin cậy, đặc biệt là trong lĩnh vực giáo dục, nơi đòi hỏi độ chính xác và độ tin cậy cao.

Bộ câu hỏi và câu trả lời có tổng cộng 5000 cặp câu qua 858 đoạn nội dung văn bản với 265 chủ đề. Đối với bài toán phân loại ý định, đầu vào là một câu hỏi và đầu ra là danh sách đoạn nội dung thuộc một trong 265 chủ đề, để đánh giá các mô hình nhóm chia tập dữ liệu theo tỉ lệ 80 % tập huấn luyện, 10 % tập xác thực, 10 % tập kiểm tra. Hình 2.4 minh hoạ một ví dụ trong kho dữ liệu.

## Công cụ đánh giá

Đánh giá mô hình cho hệ thống hỏi đáp (QA) thường sử dụng các chỉ số như F1 Score, Exact Match (EM), và BLEU Score. Mỗi chỉ số này đo lường các khía cạnh khác nhau của hiệu suất và độ chính xác của mô hình.

Khi áp dụng các công cụ này, quan trọng là phải xem xét đến bối cảnh và mục tiêu cụ thể của hệ thống hỏi đáp. Một số hệ thống có thể yêu cầu độ chính xác tuyệt đối (EM), trong khi những hệ thống khác có thể coi trọng việc dự đoán các từ chính xác mà không cần khớp hoàn toàn (F1, BLEU). Thông thường, một sự kết hợp của các chỉ số này sẽ cung cấp cái nhìn toàn diện nhất về hiệu suất của mô hình QA.

### F1 Score

F1 Score là một chỉ số đánh giá phổ biến dùng trong nhiều lĩnh vực của máy học và xử lý ngôn ngữ tự nhiên (NLP), bao gồm cả hệ thống hỏi đáp (QA). F1 Score cung cấp một cách để đo lường sự cân bằng giữa Precision (độ chính xác) và Recall (độ bao phủ), hai chỉ số quan trọng trong đánh giá hiệu suất mô hình.

Precision (Độ chính xác) là tỷ lệ giữa số câu trả lời đúng mà mô hình dự đoán chính xác và tổng số câu trả lời mà mô hình dự đoán. Nói cách khác, đây là phần trăm của các câu trả lời dự đoán chính xác so với tổng số câu trả lời mà mô hình đưa ra.

Recall (Độ bao phủ) là tỷ lệ giữa số câu trả lời đúng mà mô hình dự đoán chính xác và tổng số câu trả lời đúng trong dữ liệu. Nói cách khác, đây là phần trăm của các câu trả lời dự đoán chính xác so với tổng số câu trả lời đúng mà mô hình nên dự đoán.

Trong hệ thống hỏi đáp, F1 Score tính đến số lượng từ chính xác (overlap) giữa câu trả lời dự đoán và câu trả lời thực tế. F1 Score được tính bằng công thức:

F1 = 2 × ​

Trong hệ thống hỏi đáp, F1 Score giúp đánh giá mức độ chính xác của các câu trả lời dựa trên các từ khóa hoặc thông tin chính xác được trích xuất từ nguồn dữ liệu. Đối với mỗi câu hỏi, câu trả lời dự đoán của mô hình được so sánh với câu trả lời thực tế để tính Precision và Recall. Sau đó, F1 Score được tính toán để đánh giá tổng quan hiệu suất mô hình trong việc trả lời chính xác các câu hỏi.

F1 Score cung cấp một phép đo cân bằng giữa Precision và Recall, đặc biệt hữu ích khi cần một hệ thống vừa phải chính xác, không đưa ra quá nhiều câu trả lời sai, vừa phải bao phủ đầy đủ, không bỏ sót câu trả lời đúng. F1 Score cao chỉ ra rằng mô hình vừa có độ chính xác cao vừa có độ bao phủ tốt, là mục tiêu trong nhiều hệ thống QA. Một F1 Score cao không phải lúc nào cũng có nghĩa là mô hình hoạt động tốt trong mọi tình huống. Trong một số trường hợp, có thể cần ưu tiên Precision hoặc Recall tùy thuộc vào yêu cầu cụ thể của ứng dụng.

### Exact Match

Exact Match đo lường tỷ lệ phần trăm mà mô hình trả lời chính xác hoàn toàn so với câu trả lời thực tế. Đây là một chỉ số nghiêm ngặt vì nó yêu cầu sự khớp hoàn hảo giữa câu trả lời dự đoán và câu trả lời đúng.

Exact Match (EM) là một chỉ số đánh giá được sử dụng rộng rãi trong hệ thống hỏi đáp (QA) để đo lường mức độ chính xác tuyệt đối của các câu trả lời do mô hình tạo ra so với câu trả lời thực tế. EM đánh giá xem câu trả lời dự đoán của mô hình có khớp hoàn hảo với câu trả lời chính xác hay không, không có sự linh hoạt nào cho phép lệch lạc hoặc sự không chính xác về mặt kỹ thuật.

Exact Match (Khớp chính xác) là một câu trả lời dự đoán được coi là khớp chính xác nếu nó giống hệt câu trả lời thực tế, kể cả dấu câu và khoảng trắng. Không có sự dung thứ cho bất kỳ sai lệch nào từ câu trả lời chính xác.

EM được biểu diễn qua tỷ lệ phần trăm câu trả lời dự đoán khớp chính xác với câu trả lời thực tế trong toàn bộ tập dữ liệu. EM được tính bằng công thức:

EM = ​

EM là một chỉ số dễ hiểu và cung cấp một đánh giá rõ ràng về khả năng của mô hình trong việc tạo ra câu trả lời chính xác tuyệt đối. Điều này làm cho EM trở thành một chỉ số quan trọng trong các hệ thống QA, đặc biệt là khi độ chính xác là yếu tố quan trọng nhất. EM không tính đến trường hợp mà câu trả lời dự đoán có thể chứa thông tin chính xác nhưng không được diễn đạt theo cách giống hệt với câu trả lời thực tế. Do đó, nó có thể không hoàn toàn phản ánh khả năng của mô hình trong việc hiểu và xử lý thông tin.

Trong thực tế, việc đạt được điểm số EM cao có thể khó khăn đối với các hệ thống hỏi đáp tự động do sự đa dạng về cách biểu đạt ngôn ngữ và câu trả lời. EM thường được sử dụng cùng với các chỉ số khác như F1 Score để cung cấp một cái nhìn toàn diện hơn về hiệu suất của mô hình.

### BLEU Score

BLEU Score (Bilingual Evaluation Understudy Score) là một chỉ số đánh giá tự động được sử dụng rộng rãi trong lĩnh vực máy dịch để đo lường chất lượng của văn bản được dịch tự động so với một hoặc nhiều bản dịch tham chiếu do con người tạo ra. Mặc dù được thiết kế ban đầu cho máy dịch, BLEU Score cũng được áp dụng trong các hệ thống hỏi đáp (QA) tự động và tác vụ NLP khác để đánh giá mức độ tương đồng giữa câu trả lời dự đoán và câu trả lời tham chiếu.

BLEU Score tính điểm dựa trên sự trùng khớp của n-gram giữa câu trả lời dự đoán và câu trả lời tham chiếu, từ đó đo lường độ chính xác và độ mượt (fluency) của văn bản dự đoán. Điểm BLEU thường nằm trong khoảng từ 0 đến 1, với 1 chỉ ra sự trùng khớp hoàn hảo.

Điểm BLEU được tính toán theo độ chính xác của n-gram, nó đo lường tỷ lệ phần trăm của các n-gram trong câu trả lời dự đoán trùng khớp với các n-gram trong câu trả lời tham chiếu. Brevity penalty (BP), phạt đoạn văn bản quá ngắn bằng cách so sánh độ dài của câu trả lời dự đoán với độ dài của câu trả lời tham chiếu. Mục đích là để tránh việc đánh giá cao câu trả lời dự đoán ngắn gọn mà thiếu thông tin. Kết hợp độ chính xác của n-gram và brevity penalty để tính điểm BLEU tổng thể:

BLEU =

​

Trong đó là độ chính xác của n-gram, là trọng số cho mỗi n-gram (thường bằng nhau), và BP là brevity penalty. N thường được chọn là 4.

Trong hệ thống QA, BLEU Score có thể được sử dụng để đánh giá độ chính xác của câu trả lời dự đoán so với câu trả lời tham chiếu. Điều này giúp kiểm tra khả năng của mô hình trong việc sinh ra câu trả lời một cách mượt mà và chính xác, gần giống với cách con người trả lời.

BLEU Score có một số hạn chế, bao gồm khả năng không phản ánh đầy đủ mức độ linh hoạt trong cách biểu đạt ngôn ngữ của con người. Một câu trả lời có thể hoàn toàn chính xác nhưng không trùng khớp về mặt từ ngữ với câu trả lời tham chiếu và do đó nhận điểm BLEU thấp.

## Kết quả thực nghiệm

Kết quả thực nghiệm cho bài toán IC và MRC được ghi nhận trong bảng dưới đây.

Bảng 2 Kết quả thực nghiệm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mô hình | F1-Score | Exact Match | BLEU Score |
| 1 | LogisticRegression cho IC | 0.55 | - | - |
| 2 | BERT cho MRC | 0.053 | 0.029 | 0.027 |
| 3 | IC + MRC | 0.17 | - | - |

## Thảo luận

### Bài toán IC

Một mô hình phân loại ý định sử dụng Logistic Regression với điểm F1 Score bằng 0.55 cho thấy mô hình có hiệu suất trung bình trong việc phân loại đúng các ý định. Nó không quá thấp để coi mô hình là không hữu ích, nhưng cũng không đủ cao để xem mô hình là rất hiệu quả. F1 Score ở mức 0.55 chỉ ra rằng có một số lượng lớn các dự đoán sai hoặc một số ý định không được phân loại chính xác.

Có thể cần xem xét lại dữ liệu huấn luyện và các tính năng được sử dụng. Bổ sung thêm dữ liệu, sử dụng kỹ thuật tái chế mẫu (oversampling) cho các lớp ít dữ liệu, hoặc tạo thêm các tính năng mới có thể giúp cải thiện F1 Score. Logistic Regression là một mô hình tuyến tính đơn giản, có thể không đủ để nắm bắt được các mối quan hệ phức tạp trong dữ liệu. Việc thử nghiệm với các mô hình phức tạp hơn như Random Forest, SVM hoặc thậm chí là các mô hình Deep Learning có thể cung cấp hiệu suất tốt hơn.

### Bài toán MRC

F1 Score là trung bình điều hòa của precision và recall, cho biết mức độ chính xác và độ bao phủ của câu trả lời so với câu trả lời tham khảo. Huấn luyện mô hình BERT cho hệ thống hỏi đáp với 4000 cặp câu hỏi và trả lời trong thời gian 37 giờ nhưng chỉ đạt được điểm F1 là 0.053 cho thấy mô hình có sự chính xác và độ bao phủ rất thấp trong việc tìm ra câu trả lời đúng, có thể do khó khăn trong việc hiểu ngữ cảnh hoặc xử lý câu hỏi phức tạp. Mô hình có thể không được huấn luyện với đủ các trường hợp đa dạng hoặc các cặp câu hỏi và trả lời không đủ phong phú, dẫn đến khả năng tổng quát hóa kém.

EM đo lường phần trăm câu trả lời của mô hình khớp chính xác hoàn toàn với ít nhất một câu trả lời tham khảo. Điểm EM là 0.029 chỉ ra rằng chỉ có một phần rất nhỏ của câu trả lời từ mô hình là hoàn toàn chính xác, điều này nói lên rằng khả năng của mô hình trong việc trả lời chính xác các câu hỏi là rất thấp.

BLEU Score đo lường mức độ tương đồng giữa câu trả lời mô hình và câu trả lời tham khảo dựa trên sự trùng lặp của n-grams. Một BLEU Score thấp như 0.027 cho thấy câu trả lời của mô hình khác biệt đáng kể so với câu trả lời tham khảo, có thể do thiếu chính xác về ngôn từ hoặc không hiểu được câu hỏi.

Việc đánh giá và cải thiện một mô hình hỏi đáp dựa trên các chỉ số này là một quá trình liên tục, đòi hỏi phải thích ứng và cập nhật kỹ thuật để đáp ứng tốt nhất với nhu cầu và tiêu chuẩn ngày càng cao trong lĩnh vực xử lý ngôn ngữ tự nhiên. Bằng cách tinh chỉnh mô hình trên tập dữ liệu lớn hơn và đa dạng hơn, việc huấn luyện trên một tập dữ liệu có chất lượng cao và đa dạng có thể giúp mô hình hiểu được nhiều ngữ cảnh và kiểu câu hỏi hơn. Cải thiện kỹ thuật xử lý ngôn ngữ tự nhiên, sử dụng các mô hình NLP tiên tiến và kỹ thuật như BERT, GPT, hoặc các biến thể của chúng có thể cải thiện đáng kể khả năng hiểu ngữ cảnh và sinh câu trả lời. Tối ưu hóa mô hình cho SQuAD 2.0, điều chỉnh mô hình để tối ưu hóa với các yêu cầu đặc biệt của SQuAD 2.0, bao gồm việc xử lý câu hỏi không có câu trả lời trong văn bản.

Ví dụ trường hợp dự đoán đúng cho câu hỏi “Cấu tạo của độ cơ đốt trong gồm?” thì hệ thống đưa ra câu trả lời dự đoán “năm 1877. Cấu tạo của động cơ đốt trong gồm có 2 cơ cấu và 4 hệ thống. Hai cơ cấu gồm có cơ cấu phân phối khí và cơ cấu trục khuỷu thanh truyền. Bốn hệ thống gồm có hệ thống bôi trơn,” so với câu trả lời đúng “2 cơ cấu và 4 hệ thống” cho thấy hệ thống đưa ra một phần của câu trả lời chính xác và trích dẫn từ đoạn ngữ cảnh. Tuy nhiên vẫn còn trường hợp hệ thống đưa ra câu dự đoán sai cho câu hỏi “Hai câu thơ 'Gió theo lối gió, mây đường mây - Dòng nước buồn thiu, hoa bắp lay' (Đây thôn Vĩ Dạ, Hàn Mặc Tử) gợi lên nỗi niềm gì?” có câu dự đoán là “a. Cảm hứng bài thơ Đây thôn Vĩ Dạ của Hàn Mặc Tử được bắt đầu từ tấm thiếp phong cảnh của Hoàng Cúc,” nhưng không chứa câu trả lời đúng “Nỗi buồn chia lìa” bên trong dữ liệu.

### Bài toán IC kết hợp MRC

Nhóm nhận thấy có sự cải thiện đáng kể từ việc chỉ sử dụng mô hình MRC với F1 score tăng từ 0.055 lên 0.17 khi kết hợp với phân loại ý định là một sự cải thiện đáng kể, cho thấy việc kết hợp phân loại ý định với MRC mang lại lợi ích rõ ràng trong việc xử lý bài toán hỏi đáp. Điều này có thể là do mô hình phân loại ý định giúp xác định rõ ràng hơn mục tiêu của câu hỏi, từ đó giúp mô hình MRC tập trung vào phần thông tin cần thiết để tìm câu trả lời chính xác.

Một F1 score thấp trong cả hai trường hợp cho thấy có thể dữ liệu huấn luyện không đủ đa dạng hoặc không phong phú về thông tin, hoặc có thể là do sự không chính xác trong việc gán nhãn. Dù đã có sự cải thiện, nhưng F1 score ở mức 0.17 vẫn còn khá thấp cho một hệ thống hỏi đáp hiệu quả. Cần có thêm nghiên cứu và tối ưu về mô hình, bao gồm việc điều chỉnh siêu tham số, tăng cường dữ liệu huấn luyện, hoặc thậm chí kết hợp thêm các kỹ thuật NLP khác.

# ỨNG DỤNG MINH HỌA

## Flask cho Python

Flask là một microframework cho Python được thiết kế để tạo nhanh các ứng dụng web nhỏ đến trung bình, việc sử dụng Flask cho bài toán hỏi đáp mang lại nhiều ưu điểm và cơ hội, nhưng cũng có những thách thức cần giải quyết. Được phát triển bởi Armin Ronacher và ra mắt công chúng lần đầu vào năm 2010, Flask đã nhanh chóng trở thành một trong những framework web phổ biến nhất cho Python, cạnh tranh với Django, một framework lớn hơn và đầy đủ chức năng hơn. Flask được đánh giá cao vì sự đơn giản, linh hoạt, và dễ sử dụng của nó.

Flask mang lại những tính năng hay như Flask cung cấp một lõi micro đơn giản, tự do mở rộng theo nhu cầu của ứng dụng, điều này cho phép các nhà phát triển tự do lựa chọn các công cụ và thư viện muốn sử dụng cùng với Flask. Flask hỗ trợ việc định tuyến URL một cách dễ dàng, giúp kết nối các hàm xử lý với các địa chỉ web cụ thể. Flask sử dụng Jinja2 cho phần template, giúp tạo ra các trang HTML động một cách dễ dàng từ dữ liệu. Flask đi kèm với một máy chủ phát triển tích hợp, cho phép kiểm tra ứng dụng web một cách nhanh chóng mà không cần tới một máy chủ web thực sự. Flask cung cấp một chế độ debug mạnh mẽ giúp tìm và sửa lỗi dễ dàng hơn. Mặc dù là một microframework, Flask có một hệ sinh thái mở rộng rất lớn, với các extension cho phép thêm vào các chức năng như ORM, xác thực người dùng, xử lý form, và nhiều hơn nữa mà không làm mất đi sự đơn giản của Flask.

Flask nên được sử dụng bởi Flask cung cấp một khởi đầu nhanh chóng và đơn giản cho việc phát triển web. Khả năng tùy chỉnh cao và sự linh hoạt cho phép các nhà phát triển sử dụng Flask cho một loạt các dự án, từ các trang web tĩnh đơn giản đến các ứng dụng web phức tạp. Cùng với Flask có một cộng đồng người dùng và nhà phát triển lớn, cung cấp rất nhiều tài nguyên, hướng dẫn, và thư viện mở rộng.

## Ứng dụng trang web

A screenshot of a computer

Description automatically generated

Hình 5.1 Giao diện ứng dụng

Để thiết kế một trang web hỏi đáp đơn giản với Flask, nhóm đã tạo một giao diện người dùng (UI) đơn giản, nơi người dùng có thể đặt câu hỏi và xem các chủ đề kiến thức đã huấn luyện bên cạnh, hình 5.1. Tập tin app.py sẽ chứa logic của Flask app, bao gồm routing và xử lý các yêu cầu. File chat.html trong thư mục templates sẽ chứa form để người dùng nhập câu hỏi và hiển thị các chủ đề kiến thức, cũng như câu trả lời cho câu hỏi. Câu hỏi “Động cơ đốt trong là gì?” hệ thống dự đoán câu trả lời là “Mã lực, tốc độ 800 vòng/ phút là động cơ xăng. động cơ xăng có 5 hệ thống chính là hệ thống bôi trơn, hệ thống làm mát, hệ thống khởi động. Và 4 hệ thố@@.” có điểm F1 score là 0.22 là trường hợp tương đối đúng khi đề cập đến động cơ xăng. Tuy nhiên câu hỏi “Tác giả Rừng xà nu là ai?” được dự đoán là “Cảm hứng chủ yếu của 'rừng xà nu'. Vào nhau.” có F1 score là 0.31 thì hệ thống trả lời chưa chính xác câu hỏi và cần khai thác thông tin nhiều hơn.

Trong việc phát triển hệ thống hỏi đáp (Q&A) sử dụng Flask Python, việc xây dựng một trang lịch sử nhắn tin là một tính năng hữu ích, không chỉ cho phép nhóm xem lại các câu hỏi và câu trả lời trước đó mà còn hỗ trợ trong việc phân tích hiệu suất của hệ thống qua thời gian. F1 Score là một chỉ số đánh giá mô hình quan trọng cần xem xét trong quá trình này. Hiển thị lịch sử qua template Jinja trong Flask, cho phép tạo các trang HTML động dựa trên dữ liệu truy vấn được trong hình 5.2.

A screenshot of a computer

Description automatically generated

Hình 5.2 Lịch sử tin nhắn

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

Việc kết hợp sử dụng Logistic Regression để phân loại câu hỏi và tìm ngữ cảnh phù hợp, cùng với việc áp dụng mô hình BERT để dự đoán câu trả lời là một chiến lược mạnh mẽ trong việc xây dựng hệ thống hỏi đáp. Logistic Regression là một mô hình tuyến tính, đơn giản nhưng hiệu quả, đặc biệt khi số lượng tính năng (features) không quá lớn và mối quan hệ giữa các tính năng và kết quả là tương đối tuyến tính. Việc huấn luyện và triển khai mô hình Logistic Regression thường nhanh chóng và không đòi hỏi nhiều tài nguyên tính toán, làm cho nó phù hợp với việc phân loại câu hỏi theo ngữ cảnh hoặc loại.

BERT, với kiến trúc Transformer và khả năng hiểu ngữ cảnh hai chiều, mạnh mẽ trong việc hiểu ý nghĩa sâu sắc của văn bản, giúp dự đoán câu trả lời chính xác từ ngữ cảnh đã được xác định. BERT có thể được tinh chỉnh cho một loạt các tác vụ NLP, bao gồm cả hệ thống hỏi đáp, làm cho nó trở thành công cụ linh hoạt và mạnh mẽ.

Kết hợp sức mạnh của Logistic Regression trong việc xác định nhanh chóng ngữ cảnh hoặc loại câu hỏi với khả năng hiểu và xử lý ngữ cảnh sâu sắc của BERT tạo ra một hệ thống hỏi đáp mạnh mẽ, có khả năng hiểu và trả lời các câu hỏi phức tạp. Sự kết hợp này tối ưu hóa cả hiệu suất và độ chính xác của hệ thống hỏi đáp, giúp nó nhanh chóng xác định ngữ cảnh phù hợp và sau đó dự đoán câu trả lời một cách chính xác. Từ hỗ trợ khách hàng đến trợ lý ảo, hệ thống giáo dục, và hơn thế nữa, sự kết hợp này có thể được áp dụng cho nhiều loại ứng dụng hỏi đáp.

Từ đánh giá và thảo luận về hiệu suất của mô hình hỏi đáp dựa trên F1 Score, EM (Exact Match), và BLEU Score trên tập dữ liệu SQuAD 2.0, có thể thấy mô hình này gặp phải nhiều thách thức trong việc hiểu và trả lời chính xác các câu hỏi. Các chỉ số đánh giá thấp cho thấy rằng mô hình hiện tại cần được cải thiện đáng kể để có thể hoạt động hiệu quả. Việc kết hợp phân loại ý định với mô hình MRC mang lại sự cải thiện đáng kể so với việc chỉ sử dụng MRC. Tuy nhiên, F1 score vẫn còn thấp, cho thấy cần có nhiều cải tiến và tối ưu hóa hơn nữa. Việc này yêu cầu một cách tiếp cận toàn diện, từ việc phân tích kỹ lưỡng dữ liệu, tối ưu hóa mô hình, cho đến việc áp dụng các kỹ thuật NLP tiên tiến.

Mô hình cần phải cải thiện khả năng hiểu ngữ cảnh và ngữ nghĩa của văn bản để tạo ra câu trả lời chính xác hơn. SQuAD 2.0 đặc biệt khó khăn do bao gồm cả câu hỏi không có câu trả lời trong văn bản, yêu cầu mô hình phải nhận diện và xử lý loại câu hỏi này một cách chính xác. Cần cải thiện phương pháp đánh giá mô hình bằng cách sử dụng các chỉ số đánh giá đa chiều và huấn luyện mô hình trên một tập dữ liệu đa dạng và lớn hơn.

## Hướng phát triển

Dù BERT đã mạnh mẽ, việc tinh chỉnh mô hình dựa trên tập dữ liệu cụ thể có thể giúp cải thiện hiệu suất đáng kể. Tìm hiểu và thử nghiệm với các cấu hình và siêu tham số khác nhau để tối ưu mô hình cho tác vụ cụ thể.

Mở rộng tập dữ liệu huấn luyện với nhiều mẫu câu hỏi và ngữ cảnh khác nhau, đặc biệt là những mẫu từ các nguồn không được đại diện rộng rãi. Đa dạng hóa dữ liệu giúp mô hình có khả năng tổng quát hóa tốt hơn.

Đối với các hệ thống hỏi đáp trực tuyến, thời gian phản hồi là yếu tố quan trọng. Tìm cách tối ưu hóa đoạn mã và cơ sở hạ tầng để giảm thiểu độ trễ. Áp dụng kỹ thuật xử lý đồng thời để cải thiện khả năng xử lý nhiều yêu cầu cùng lúc, nâng cao hiệu quả phục vụ.

Phát triển giao diện người dùng trực quan và thân thiện, giúp người dùng dễ dàng tương tác và sử dụng hệ thống. Mở rộng khả năng tương tác của hệ thống qua nhiều kênh giao tiếp như web, ứng dụng di động, hoặc thậm chí qua các nền tảng nhắn tin.

Áp dụng các kỹ thuật NLP và AI tiên tiến khác như nhận diện thực thể đặt tên (Named Entity Recognition), tích hợp NER để cải thiện khả năng hiểu các thực thể cụ thể trong câu hỏi. Sử dụng tích hợp ngôn ngữ tự nhiên NLG để tạo ra các câu trả lời tự nhiên và phong phú hơn, thay vì chỉ trích xuất văn bản.

Xây dựng cơ chế để thu thập và phân tích phản hồi từ người dùng, sử dụng thông tin này để liên tục cải thiện mô hình. Áp dụng kỹ thuật học máy để mô hình học hỏi và tự điều chỉnh từ mỗi tương tác với người dùng, từ đó trở nên thông minh hơn theo thời gian.

Sử dụng mô hình NLP tiên tiến, áp dụng và tinh chỉnh các mô hình ngôn ngữ lớn và tiên tiến như BERT, GPT-3, hoặc các biến thể của Transformer có thể giúp mô hình hiểu sâu hơn về ngữ cảnh và ngữ nghĩa, từ đó cải thiện khả năng trả lời câu hỏi. Áp dụng kỹ thuật huấn luyện chuyên sâu như học chuyển giao (transfer learning), học tăng cường (reinforcement learning), hoặc học đa nhiệm (multi-task learning) để cải thiện khả năng tổng quát hóa và hiểu biết của mô hình.

Tối ưu hóa kiến trúc mô hình và kỹ thuật xử lý, bao gồm việc xử lý câu hỏi và văn bản nguồn, để cải thiện tốc độ và độ chính xác của mô hình. Mở rộng tập dữ liệu huấn luyện với nhiều loại câu hỏi và văn bản từ nguồn khác nhau để mô hình có thể học được nhiều ngữ cảnh và ngữ nghĩa hơn, giúp cải thiện khả năng trả lời chính xác trong các tình huống thực tế. Việc áp dụng những hướng phát triển này đòi hỏi sự đầu tư về thời gian, nguồn lực, và nghiên cứu, nhưng chúng có tiềm năng lớn trong việc cải thiện đáng kể hiệu suất của mô hình hỏi đáp tự động, làm cho nó trở nên hữu ích hơn trong các ứng dụng thực tế.

# TÀI LIỆU THAM KHẢO

Tiếng Anh

[1] J. A. Alzubi, R. Jain, A. Singh, P. Parwekar, and M. Gupta, “COBERT: COVID-19 Question Answering System Using BERT,” *Arab J Sci Eng*, vol. 48, no. 8, pp. 11003–11013, Aug. 2023, doi: 10.1007/s13369-021-05810-5.

[2] C. T. Nguyen and D. T. Nguyen, “Building a Discourse-Argument Hybrid System for Vietnamese Why-Question Answering,” *Computational Intelligence and Neuroscience*, vol. 2021, p. e6550871, Dec. 2021, doi: 10.1155/2021/6550871.

[3] S. Raza, B. Schwartz, and L. C. Rosella, “CoQUAD: a COVID-19 question answering dataset system, facilitating research, benchmarking, and practice,” *BMC Bioinformatics*, vol. 23, no. 1, p. 210, Jun. 2022, doi: 10.1186/s12859-022-04751-6.

[4] T. T. Aurpa, R. K. Rifat, M. S. Ahmed, M. M. Anwar, and A. B. M. S. Ali, “Reading comprehension based question answering system in Bangla language with transformer-based learning,” *Heliyon*, vol. 8, no. 10, Oct. 2022, doi: 10.1016/j.heliyon.2022.e11052.

[5] M. Ahmed, H. Khan, T. Iqbal, F. K. Alarfaj, A. Alomair, and N. Almusallam, “On solving textual ambiguities and semantic vagueness in MRC based question answering using generative pre-trained transformers,” *PeerJ Comput. Sci.*, vol. 9, p. e1422, Jul. 2023, doi: 10.7717/peerj-cs.1422.

[6] “D16-1.pdf.” Accessed: Dec. 16, 2023. [Online]. Available: https://aclanthology.org/D16-1.pdf

[7] “P18-2.pdf.” Accessed: Dec. 16, 2023. [Online]. Available: https://aclanthology.org/P18-2.pdf

[8] A. Allam, “The Question Answering Systems: A Survey”, Accessed: Dec. 18, 2023. [Online]. Available: https://www.academia.edu/13265598/The\_Question\_Answering\_Systems\_A\_Survey

[9] S. Manandhar and S. Quarteroni, “A chatbotbased interactive question answering system,” Jan. 2007, Accessed: Dec. 18, 2023. [Online]. Available: https://www.academia.edu/18195382/A\_chatbotbased\_interactive\_question\_answering\_system

[10] S. J. Pan and Q. Yang, “A Survey on Transfer Learning,” *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.

[11] A. Vaswani *et al.*, “Attention Is All You Need,” arXiv.org. Accessed: Dec. 30, 2023. [Online]. Available: https://arxiv.org/abs/1706.03762v7

[12] “BERT.” Accessed: Dec. 30, 2023. [Online]. Available: https://huggingface.co/docs/transformers/model\_doc/bert

[13] “T5.” Accessed: Dec. 30, 2023. [Online]. Available: https://huggingface.co/docs/transformers/model\_doc/t5