COMPARING QUESTION SIMILARITY IN FORUMS

Vo Tran Tien, Luong Tran Ngoc Khiet*, Nguyen Phuong Nam, Huynh Thi Tuong Vi, Nguyen Huynh Phuc Khang, Phan Thi Nam Anh, Luong Tran Hy Hien
Ho Chi Minh University of Education

ARTICLE INFO

Received: 09/4/2025
Revised: 26/6/2025
Published: 28/6/2025

KEYWORDS

PhoBERT
Similar question
Natural language processing
Forum
Text extraction

ABSTRACT

This study aims to develop a system for comparing the similarity of questions on online forums using the PhoBERT model combined with natural language processing techniques. The goal is to improve the recognition of similar questions, thereby automatically suggesting appropriate and timely responses. The research methodology includes collecting data from forums, student confessions, and fan pages, followed by data preprocessing steps such as duplicate removal, text normalization, and tokenization. The model was trained on a comprehensive Vietnamese dataset of 31,201 question pairs. The PhoBERT model was then trained and applied to the collected dataset to classify similar questions. The results show that the system achieved high accuracy, with a prediction success rate of 82.98%, outperforming traditional methods like TF-IDF. For practical application, the system also integrated FAISS to enable efficient, real-time similarity search. The system is not only effective in comparing questions but can also be applied to online Q&A platforms or customer support, opening up opportunities for applications in various fields.

SO SÁNH ĐỘ TƯƠNG ĐỒNG CÂU HỎI TRONG DIỄN ĐÀN

Võ Trần Tiến, Lương Trần Ngọc Khiết*, Nguyễn Phương Nam, Huỳnh Thị Tường Vi, Nguyễn Huỳnh Phúc Khang, Phan Thị Nam Anh, Lương Trần Hy Hiến
Trường Đại học Sư phạm Thành phố Hồ Chí Minh

THÔNG TIN BÀI BÁO

Ngày nhận bài: 09/4/2025
Ngày hoàn thiện: 26/6/2025
Ngày đăng: 28/6/2025

TỪ KHÓA

PhoBERT
Câu hỏi tương đồng
Phương pháp xử lý ngôn ngữ tự nhiên
Diễn đàn
Trích xuất văn bản

TÓM TẮT

Nghiên cứu này nhằm xây dựng hệ thống so sánh độ tương đồng giữa các câu hỏi trên các diễn đàn trực tuyến, sử dụng mô hình PhoBERT kết hợp với các phương pháp xử lý ngôn ngữ tự nhiên. Mục đích là cải thiện khả năng nhận diện câu hỏi tương đồng, từ đó tự động đề xuất câu trả lời thích hợp và nhanh chóng. Phương pháp nghiên cứu bao gồm thu thập dữ liệu từ các diễn đàn, confession của sinh viên và fanpage, xử lý dữ liệu bằng các bước như loại bỏ trùng lặp, chuẩn hóa văn bản và tách từ (tokenization). Mô hình được huấn luyện trên bộ dữ liệu tiếng Việt gồm 31.201 cặp câu hỏi. Sau đó, mô hình PhoBERT được áp dụng để phân loại các câu hỏi tương đồng. Kết quả cho thấy hệ thống đạt độ chính xác 82,98%, vượt trội hơn so với phương pháp truyền thống như TF-IDF. Để ứng dụng trong thực tế, hệ thống còn tích hợp thư viện FAISS nhằm cho phép tìm kiếm tương đồng hiệu quả và nhanh chóng. Hệ thống không chỉ hiệu quả trong việc so sánh câu hỏi mà còn có thể áp dụng trong các nền tảng hỏi đáp trực tuyến hoặc hỗ trợ khách hàng, mở ra cơ hội ứng dụng trong nhiều lĩnh vực khác.

DOI: https://doi.org/10.34238/tnu-jst.12516

* Corresponding author. *Email: khietltn@ hcmue.edu.vn*

1. Giới thiệu

Internet hiện nay cung cấp nguồn tài liệu vô cùng phong phú cho học tập và nghiên cứu, tuy nhiên sự khổng lồ của thông tin có thể gây khó khăn trong việc phân biệt thông tin chính xác và sai lệch. Các công cụ tìm kiếm đôi khi không thể hiểu đúng ngữ cảnh, dẫn đến kết quả không chính xác. Vì vậy, việc đặt câu hỏi trên các diễn đàn đặc biệt là các diễn đàn chuyên ngành trở thành một phương thức phổ biến, giúp người dùng nhận được câu trả lời chính xác từ cộng đồng có chuyên môn.

Các câu hỏi được đặt ra trên các diễn đàn có thể được diễn đạt khác nhau nhưng lại hướng đến cùng một nội dung trả lời, vì thế bài báo trình bày nghiên cứu về xây dựng hệ thống so sánh độ tương đồng của hai câu hỏi. Hệ thống được đề xuất nhằm mục đích tối ưu hóa trải nghiệm người dùng bằng cách phân tích và nhận diện câu hỏi tương đồng, từ đó tự động đề xuất câu trả lời phù hợp từ kho dữ liệu có sẵn thay vì mất nhiều thời gian chờ người khác trả lời.

Các bộ khung sườn có sẵn (framework) truy vấn sáng tạo như mô hình truy vấn hai bước (SQuID), sử dụng các kỹ thuật nhúng tiên tiến để xếp hạng các câu hỏi tương đồng [1]. Những phương pháp như vậy, sử dụng các mô hình tương tự BERT, tinh chỉnh thêm quá trình phân tích các câu hỏi và tận dụng sự tương đồng ngữ nghĩa để đạt được sự chính xác hơn. Các mô hình này ban đầu chọn các câu hỏi tương tự top-k và sau đó đánh giá thêm chúng để xác định câu trả lời phù hợp nhất, tối ưu hóa quy trình hỏi - đáp.

Một phương pháp kết hợp giữa BERT và các kiến trúc Siamese đã nâng cao thêm các tác vụ xác định sự tương đồng văn bản. Chúng tôi đề xuất kết hợp BERT Fine-tuned và mô hình Bi-LSTM Siamese để phân tích độ tương đồng văn bản từ bộ dữ liệu cặp câu hỏi Quora. Mô hình trích xuất đặc trưng câu hỏi, sau đó sử dụng Bi-LSTM để dự đoán độ tương đồng văn bản. Phương pháp này đạt độ chính xác 91%, vượt trội hơn so với các phương pháp hiện có trong việc phát hiện độ tương đồng văn bản [2]. Tương tự, mô hình Sentence-BERT (SBERT) đã chứng minh hiệu quả vượt trội khi sử dụng kiến trúc Siamese để tạo ra các vector biểu diễn câu có ý nghĩa ngữ nghĩa, cho phép so sánh tương đồng hiệu quả bằng khoảng cách cosine [3].

Sử dụng các kỹ thuật học máy được Hasmawati và Romadhony (đề xuất để cải thiện việc nhận diện các câu hỏi tương đồng, nâng cao hiệu quả bằng cách nhận diện các câu hỏi đã được gửi trước đó và hướng người dùng đến những câu trả lời đã có. Kết quả thử nghiệm cho thấy mô hình sử dụng đặc trưng tương đồng cosine trong thuật toán Support Vector Machine (SVM) mang lại hiệu suất tối ưu [4]. Tuy nhiên, khi sử dụng đặc trưng POS Tag hoặc kết hợp POS Tag với tương đồng cosine, mô hình gặp phải vấn đề quá khớp, khiến độ chính xác giảm. Để nâng cao hiệu suất trong các nghiên cứu sau, tác giả đề xuất thử nghiệm các phương pháp trích xuất đặc trưng khác như TF-IDF và đánh giá các đặc trưng ngữ nghĩa. Đồng thời, cần cải thiện bộ dữ liệu bằng cách mở rộng số lượng và sự đa dạng của từ ngữ sử dụng. Các nghiên cứu tổng quan gần đây cũng chỉ ra rằng việc lựa chọn phương pháp trích xuất đặc trưng phù hợp là yếu tố then chốt quyết định hiệu suất của các mô hình học máy truyền thống trong bài toán so sánh văn bản. Quá trình này nhấn mạnh tầm quan trọng của việc lựa chọn kỹ thuật tiền xử lý và trích xuất đặc trưng phù hợp, một lĩnh vực đã được phân tích và so sánh trong nhiều nghiên cứu tổng quan [5].

Mô hình huấn luyện Long Short-Term Memory (LSTM) đã cho kết quả đáng chú ý trong việc phát hiện sự tương đồng trong các câu hỏi bài tập, nhờ khả năng ghi nhớ các phụ thuộc dài hạn và nhận diện các mẫu, sự tương đồng và các yếu tố tinh tế trong dữ liệu văn bản [6]. Kết quả thí nghiệm cho thấy LSTM không chỉ phát hiện sự tương đồng rõ ràng mà còn nhận diện những sự tương đồng tinh tế, quan trọng trong các lĩnh vực như phát hiện đạo văn và phân tích ngữ nghĩa. Mô hình này cũng chứng tỏ khả năng mở rộng và thích ứng tốt, hứa hẹn ứng dụng trong các lĩnh vực khác như phân tích cảm xúc và tạo văn bản dự đoán.

Zhou và các cộng sự [7] đề xuất mô hình KEBERT-GCN để cải thiện độ tương đồng ngữ nghĩa và mô hình CPT-TK để đánh giá độ tương đồng cú pháp của văn bản ngắn. Kết hợp cả hai mô hình, tác giả tạo ra một mô hình đánh giá độ tương đồng văn bản ngắn hiệu quả hơn, đạt hiệu

suất tốt hơn các phương pháp hiện tại. Nghiên cứu cũng chỉ ra một số hướng phát triển trong tương lai như cải thiện ma trận tương đồng từ vựng và thử nghiệm với các biến thể của BERT.

Đã có không ít các công trình nghiên cứu về việc áp dụng các mô hình để so sánh câu hỏi tương đồng nhưng phần lớn đạt hiệu quả cao với ngôn ngữ là tiếng Anh. Đối với tiếng Việt, các thách thức về sự đa dạng trong biểu đạt và thiếu hụt bộ dữ liệu có gán nhãn vẫn là một rào cản lớn. Để giải quyết khoảng trống này, các mô hình chuyên biệt như PhoBERT đã được tạo ra [8]. Dựa trên nền tảng đó, các công trình trong nước đã bắt đầu ghi nhận những thành công đáng kể. Các hướng tiếp cận tiêu biểu bao gồm việc kết hợp Sentence-BERT và PhoBERT để nhận diện câu diễn giải tương đương [9], hay tích hợp thêm các nguồn tri thức ngữ nghĩa như WordNet để tăng cường độ chính xác [10]. Kế thừa các hướng tiếp cận đó và nhận thấy sự cần thiết của việc tích hợp các cơ sở tri thức ngoài [11], bài báo này trình bày một hệ thống toàn diện, không chỉ áp dụng PhoBERT để trích xuất ngữ nghĩa mà còn tích hợp thư viện FAISS để tối ưu hóa việc tìm kiếm và truy xuất câu trả lời từ kho dữ liệu lớn [12].

Một hướng tiếp cận đột phá khác trong việc tạo ra các biểu diễn câu chất lượng cao là sử dụng phương pháp học đối nghịch (contrastive learning). Thay vì chỉ dựa vào các cặp câu được gán nhãn, phương pháp này tự tạo ra các cặp câu "dương tính" (positive pairs) bằng cách áp dụng các kỹ thuật nhiễu nhẹ (ví dụ: dropout) trên cùng một câu đầu vào. Mô hình sau đó được huấn luyện để kéo các biểu diễn của cặp dương tính lại gần nhau và đẩy các biểu diễn của các câu khác (cặp âm tính) ra xa trong không gian vector. Công trình tiêu biểu cho hướng đi này là SimCSE, đã chứng minh rằng phương pháp đơn giản này có thể tạo ra các vector biểu diễn câu vượt trội, đạt hiệu suất hàng đầu trên nhiều tác vụ so sánh độ tương đồng ngữ nghĩa mà không cần dữ liệu gán nhãn phức tạp [13].

Mục tiếp theo của bài báo sẽ đề xuất phương pháp nghiên cứu bao gồm kế hoạch triển khai, thu thập, xử lý dữ liệu và xây dựng hệ thống. Mục 3 trình bày kết quả hệ thống và cuối cùng là kết luận về hiệu suất, ứng dụng và hướng phát triển.

2. Phương pháp nghiên cứu

2.1. Kế hoạch triển khai

2.1.1. Định nghĩa bộ dữ liệu

Bộ dữ liệu được xây dựng bao gồm các cặp câu hỏi có hình thức biểu đạt gần giống nhau, nhưng có thể tương đồng hoặc không tương đồng về mặt ý nghĩa. Mỗi cặp câu hỏi được gán nhãn nhị phân theo tiêu chí sau:

- **Nhãn 1**: Hai câu hỏi có ý nghĩa tương đương, thể hiện cùng một mục đích hoặc nội dung, chỉ khác nhau về cách diễn đạt.
- **Nhãn 0**: Hai câu hỏi khác nhau về mục đích, nội dung hoặc ngữ nghĩa, mặc dù có thể có một số từ ngữ hoặc cấu trúc ngôn ngữ tương tự.

2.1.2. Kế hoạch

Dữ liệu được thu thập từ nhiều nguồn khác nhau nhằm đảm bảo tính đa dạng về cách diễn đạt và chủ đề, bao gồm:

- Các diễn đàn trực tuyến, fanpage, confession liên quan đến sinh viên, sổ tay sinh viên, v.v.
- Các câu hỏi được tạo mới bằng cách biến đổi thủ công các câu hỏi có sẵn như thay đổi trật tự từ, thay từ đồng nghĩa, rút gọn câu, v.v.
- Một phần dữ liệu được chọn lọc từ tập *Quora Question Pairs* trên Kaggle.
- Sau khi thu thập, dữ liệu được xử lý thủ công để đảm bảo chất lượng, cụ thể gồm:

- Dịch các cặp câu hỏi từ tiếng Anh sang tiếng Việt (áp dụng đối với dữ liệu từ Quora) để đảm bảo sự nhất quán trong toàn bộ tập dữ liệu.
- Loại bỏ phần lớn các cặp câu hỏi trùng lặp hoàn toàn hoặc gần giống nhau về mặt hình thức, chỉ giữ lại một số ít để phục vụ mục đích kiểm tra và đánh giá độ nhạy của hệ thống.

2.2. Thu thập, xử lý dữ liệu và xây dựng hệ thống

2.2.1. Thu thập và xử lý dữ liệu

Nguồn gốc bộ dữ liệu:

- Nguồn dữ liệu thu thập thủ công: Nhóm nghiên cứu thu thập các câu hỏi từ fanpage Facebook chính thức của các trường đại học tại Việt Nam, nơi sinh viên thường đăng thắc mắc về tuyển sinh, đào tạo, học phí, học bổng, lịch học,... Ngoài ra, một phần câu hỏi được trích từ sổ tay sinh viên và được diễn đạt lại thông qua các thao tác thủ công như thay đổi từ ngữ, cấu trúc câu, rút gọn hoặc mở rộng nội dung. Tất cả dữ liệu đều được thu thập và xử lý thủ công nhằm tạo ra các cặp câu hỏi có hình thức khác nhau phục vụ mục tiêu phân loại mức độ tương đồng.
- Nguồn dữ liệu từ tập "Quora Question Pairs" trên Kaggle: Tập dữ liệu gồm hơn 400.000 cặp câu hỏi có nhãn phân biệt câu hỏi trùng lặp hoặc không. Nhóm nghiên cứu chọn lọc, dịch sang tiếng Việt và tích hợp một phần vào bộ dữ liệu chính nhằm tăng tính đa dạng và khả năng tổng quát của dữ liệu.

Quy mô và đặc điểm: Bộ dữ liệu sau khi xử lý và tổng hợp bao gồm 31.201 cặp câu hỏi, gồm:

- **12.189 cặp** nhãn **1**: Hai câu hỏi có ý nghĩa tương đồng
- **19.012 cặp** nhãn **0**: Hai câu hỏi khác nhau về mặt ý nghĩa, dù có thể giống về từ ngữ

Mỗi dòng dữ liệu đại diện cho một cặp câu hỏi, được lưu trữ dưới định dạng .csv, bao gồm các trường:

- id: Mã định danh của cặp câu hỏi
- qid1, qid2: Mã định danh của từng câu trong cặp
- question1, question2: Văn bản gốc của hai câu hỏi
- is_duplicate: Nhãn nhị phân xác định mức độ tương đồng về ý nghĩa (1: giống, 0: khác)
- answer: Câu trả lời đi kèm cho các cặp mang nhãn 1, do nhóm nghiên cứu trực tiếp biên soạn.

Quy trình thu thập và xử lý dữ liệu:

- **Bước 1**: Truy cập các fanpage Facebook dành cho sinh viên của một số trường đại học trên cả nước, chọn lọc các bài đăng và bình luận có dạng câu hỏi.
- **Bước 2**: Lưu trữ thủ công các câu hỏi.
- **Bước 3**: Chọn lọc và dịch các cặp câu hỏi từ tập Quora Question Pairs sang tiếng Việt để đảm bảo đồng nhất về ngôn ngữ.
- **Bước 4** : Tiền xử lý dữ liệu
- **Bước 5**: Kết hợp và xây dựng các cặp câu hỏi có tiềm năng tương đồng hoặc khác biệt về ý nghĩa, chuẩn bị cho quá trình gán nhãn.

Quy trình gán nhãn: Việc gán nhãn được thực hiện thủ công bởi nhóm nghiên cứu. Mỗi cặp câu hỏi được đánh giá dựa trên tiêu chí sau:

- Gán **nhãn 1** nếu hai câu hỏi có cùng nội dung và mục đích, chỉ khác nhau về cách diễn đạt.
- Gán **nhãn 0** nếu hai câu hỏi có ý nghĩa hoặc mục tiêu khác nhau, kể cả khi cấu trúc ngôn ngữ tương tự.

Ngoài ra, đối với các cặp mang nhãn 1, nhóm nghiên cứu còn xây dựng một câu trả lời mẫu tương ứng, nhằm phục vụ cho các hướng nghiên cứu mở rộng liên quan đến hệ thống hỏi – đáp (Question Answering).

2.2.2. Các kiến trúc mô hình

Nghiên cứu áp dụng kết hợp các mô hình học sâu (deep learning) và các phương pháp xử lý ngôn ngữ tự nhiên (NLP) để giải quyết bài toán so sánh độ tương đồng giữa các câu hỏi trong diễn đàn. Kiến trúc mô hình hoàn chỉnh được xây dựng với ba thành phần chính:

- **NLP (Natural Language Processing):** Giai đoạn này đóng vai trò nền tảng nhằm chuẩn hóa và xử lý dữ liệu đầu vào trước khi đưa vào mô hình học sâu. Quá trình xử lý dữ liệu văn bản bao gồm:

- Làm sạch dữ liệu (Data Cleaning): loại bỏ ký tự không cần thiết, dấu câu, khoảng trắng thừa, và các ký tự đặc biệt nhằm hạn chế nhiễu dữ liệu.
- Chuẩn hóa văn bản (Text Normalization): thực hiện chuyển đổi toàn bộ văn bản về dạng viết thường, chuẩn hóa định dạng ký tự Unicode, loại bỏ từ dừng (stopwords).
- Tách từ và mã hóa: văn bản được phân tách thành các token và mã hóa thành dạng vector đầu vào phù hợp với mô hình học sâu (dạng mã hóa các từ hoặc câu thành một dãy số (embedding)).

Giai đoạn này đảm bảo rằng dữ liệu đưa vào mô hình có tính đồng nhất, ổn định và phản ánh đầy đủ ngữ nghĩa gốc của văn bản [9].

- **PhoBERT**: PhoBERT là một mô hình ngôn ngữ tiền huấn luyện dựa trên kiến trúc Transformer, được phát triển riêng cho tiếng Việt bởi nhóm tác giả tại VinAI Research. PhoBERT sử dụng cơ chế attention đa đầu (multi-head self-attention) để học biểu diễn ngữ nghĩa của văn bản, từ đó mô hình hóa mối quan hệ giữa các từ trong câu dựa trên ngữ cảnh cục bộ lẫn toàn cục [8]. Cấu trúc mô hình PhoBERT trong nghiên cứu bao gồm:

- Encoder Layer: Chuỗi các lớp Transformer được huấn luyện trên dữ liệu tiếng Việt giúp mô hình có khả năng hiểu sâu về cấu trúc câu và ngữ cảnh ngôn ngữ.
- Classification Head: Sau khi văn bản được biểu diễn thành vector ngữ nghĩa, lớp phân loại sẽ thực hiện việc đánh giá độ tương đồng giữa hai câu hỏi. Lớp này thường sử dụng một hàm fully connected kết hợp với hàm softmax hoặc sigmoid nhằm xuất ra xác suất hoặc mức độ tương đồng.

PhoBERT đóng vai trò là bộ mã hóa ngôn ngữ, giúp trích xuất vector đặc trưng có khả năng phản ánh ý nghĩa ngữ cảnh của câu hỏi.

- **FAISS (Facebook AI Similarity Search):** Sau khi vector ngữ nghĩa của câu hỏi được sinh ra từ PhoBERT, thư viện FAISS được triển khai để giải quyết bài toán tìm kiếm các câu hỏi có mức độ tương đồng cao nhất trong không gian vector. Đây là một trong những bước quan trọng giúp hệ thống có thể truy xuất câu trả lời chính xác và nhanh chóng trong kho dữ liệu lớn.

Trong nghiên cứu, quá trình xây dựng FAISS index được thực hiện theo các bước sau:

- **Xây dựng chỉ mục FAISS (FAISS Index):** Bộ vector embedding sinh ra từ PhoBERT cho toàn bộ câu hỏi trong cơ sở dữ liệu sẽ được lưu trữ trong một cấu trúc chỉ mục (index) chuyên biệt của FAISS. Để đảm bảo cân bằng giữa tốc độ truy vấn và độ chính xác, nghiên cứu sử dụng loại index IndexFlatIP hoặc IndexIVFFlat.

Trong đó:

- **IndexFlatIP (Inner Product)**: thích hợp cho việc tính độ tương đồng cosine, có khả năng trả về kết quả chính xác nhất nhưng tốn bộ nhớ do lưu toàn bộ vector.
- **IndexIVFFlat (Inverted File with Flat quantization)**: chia nhỏ dữ liệu thành các cụm centroid, chỉ tìm kiếm trong một phần tập con của dữ liệu, giúp tăng tốc độ truy vấn nhưng có thể hy sinh một phần độ chính xác.
- **Quy mô dữ liệu**: Bộ index FAISS có thể lưu trữ hàng chục nghìn đến hàng triệu vector embedding, mỗi vector thường có kích thước cố định (ví dụ: 768 chiều đối với PhoBERT base). Trong nghiên cứu này, FAISS được xây dựng dựa trên một tập dữ liệu cỡ trung (từ vài nghìn đến vài chục nghìn cặp câu hỏi), cho phép truy vấn và so khớp nhanh chóng các câu hỏi có nội dung tương tự.
- **Cơ chế truy vấn**: Khi hệ thống tiếp nhận một câu hỏi mới, câu hỏi sẽ được mã hóa thành vector embedding thông qua PhoBERT, sau đó FAISS sẽ sử dụng phương pháp tìm kiếm vector gần nhất (**Nearest Neighbor Search**) trong chỉ mục. Khoảng cách cosine hoặc Euclidean giữa vector của câu hỏi mới và các vector trong chỉ mục sẽ được tính toán, từ đó xác định câu hỏi có mức độ tương đồng cao nhất và hỗ trợ truy xuất câu trả lời thích hợp.

Quá trình xây dựng chỉ mục và tìm kiếm với FAISS giúp hệ thống đạt được tốc độ xử lý cao trong môi trường dữ liệu có quy mô lớn, đồng thời duy trì độ chính xác cần thiết cho bài toán nhận diện câu hỏi tương đồng [11].

2.2.3. Quy trình xây dựng hệ thống so sánh độ tương đồng của 2 câu hỏi với mô hình PhoBERT

Giai đoạn 1: Xây dựng bộ dữ liệu

Trong giai đoạn này, nhóm nghiên cứu sử dụng bộ dữ liệu được xây dựng theo quy trình như đã mô tả, bao gồm dữ liệu thu thập thủ công từ mạng xã hội và dữ liệu từ tập *Quora Question Pairs* đã dịch sang tiếng Việt. Bộ dữ liệu đã được làm sạch, chuẩn hóa và gán nhãn nhị phân, phân thành hai nhóm: cặp câu hỏi tương đồng và không tương đồng về mặt ý nghĩa.

Sau khi hoàn tất tiền xử lý và cân bằng dữ liệu, tập dữ liệu được lưu trữ ở định dạng .csv, sẵn sàng để nhập vào mô hình huấn luyện.

Giai đoạn 2: Xây dựng mô hình PhoBERT

Bước 2.1: Chuẩn hóa văn bản:

- Chuyển văn bản về chữ thường.
- Loại bỏ các ký tự đặc biệt, các ký tự khoảng trắng thừa
- Chuyển đổi các chữ viết tắt thành chữ đầy đủ (VD: đh: đại học, ktx: kí túc xá)
- Tách từ (Tokenization): PhoBERT không dùng phương pháp tách từ truyền thống mà sử dụng Byte Pair Encoding (BPE), nên cần tokenizer chuyên biệt.

Bước 2.2: Xây dựng mô hình PhoBERT

Mô hình PhoBERT được huấn luyện trong **5 vòng lặp (epoch)**. Trong mỗi epoch, mô hình được chuyển sang chế độ huấn luyện bằng model.train(). Tập dữ liệu huấn luyện được chia thành các batch nhỏ thông qua train_loader. Với mỗi batch:

- Các thành phần đầu vào gồm input_ids, attention_mask, và labels được chuyển sang thiết bị (device) để xử lý bằng GPU hoặc CPU.
- Bộ tối ưu optimizer.zero_grad() được gọi để reset gradient.
- Mô hình tính toán đầu ra và hàm mất mát (loss) từ các đầu vào.
- Hàm mất mát được lan truyền ngược (loss.backward()), và các tham số của mô hình được cập nhật bằng optimizer.step().
- Lưu lại mô hình vào một Folder để có thể sử dụng lại.

Bước 2.3: Chạy thử chương trình:

- Sử dụng giao diện web streamlit hoặc tkinter để hiển thị.
- Tải lại mô hình PhoBERT sau khi lưu mô hình thành công.
- Chạy mô hình và nhập dữ liệu từ người dùng là hai câu hỏi khác nhau, sau đó dự đoán độ tương đồng của hai câu hỏi.

Giai đoạn 3: Xây dựng hệ thống tìm kiếm câu hỏi với FAISS

Bước 3.1: Tập dữ liệu đầu vào:

- Sử dụng cặp câu hỏi từ tập train.csv đã được xây dựng từ trước.
- Áp dụng tiền xử lý cùng quy trình như Bước 2.1 (chuẩn hoá, loại bỏ các ký tự đặc biệt và thay thế các từ viết tắt).

Bước 3.2: Mã hóa câu hỏi thành vector - Mỗi câu hỏi được chuyển đổi thành vector 768 chiều bằng mô hình PhoBERT. Vector này là trung bình của các token embedding, đảm bảo biểu diễn ngữ nghĩa toàn câu.

Bước 3.3: Xây dựng chỉ mục FAISS:

- Phân cụm (Inverted File): Chia dữ liệu thành 256 cụm (cluster) để giảm không gian tìm kiếm.
- Lượng từ hóa (PQ): Nén mỗi vector 768 chiều thành 8 subvector 8-bits, giảm dung lượng lưu trữ 10 lần.
- Huấn luyện chỉ mục: FAISS được huấn luyện trên 10.000 câu hỏi ngẫu nhiên để học cách phân phối dữ liệu.
- Dung lượng: Chỉ mục cho 1.2 triệu câu hỏi chiếm ~1,5 GB, cho phép triển khai trên máy chủ thông thường.

Bước 3.3: Tích hợp vào hệ thống, tìm kiếm thời gian thực (real-time):

Khi nhận câu hỏi mới, hệ thống thực hiện:

- Mã hoá câu hỏi thành vector bằng PhoBERT.
- Tìm câu hỏi gần nhất trong FAISS với ngưỡng tương đồng cao nhất.

2.2.4. Chiến lược đánh giá mô hình

Sau khi huấn luyện, mô hình được chuyển sang chế độ đánh giá với model.eval() để vô hiệu hoá dropout và các bước cập nhật. Đồng thời, sử dụng torch.no_grad() để tắt việc tính gradient nhằm tiết kiệm bộ nhớ và tăng tốc.

Dữ liệu từ test_loader được đưa vào mô hình, và dự đoán (logits) được tính cho từng batch. Kết quả phân lớp được lấy bằng cách chọn chỉ số có xác suất cao nhất từ logits (argmax). Các nhãn dự đoán và nhãn thực tế được lưu lại để tính toán hiệu suất. Mô hình được đánh giá bằng thang đo Accuracy (độ chính xác) để tính tỷ lệ dự đoán đúng trên toàn bộ tập kiểm thử.

3. Kết quả hệ thống

Sau khi đã xây dựng hệ thống so sánh câu hỏi bằng PhoBERT, hệ thống cần được kiểm tra xem hoạt động tốt không.

3.1. Chuẩn bị tập dữ liệu kiểm tra

Để đảm bảo hệ thống có thể nhận diện chính xác các cặp câu hỏi trùng lặp, việc xây dựng một tập dữ liệu huấn luyện chất lượng là vô cùng quan trọng. Tập dữ liệu kiểm tra được lựa chọn từ bộ dữ liệu đã xây dựng trước đó, bao gồm các cặp câu hỏi đã được gán nhãn rõ ràng theo hai nhóm là tương đồng và không tương đồng về mặt ý nghĩa. Dữ liệu kiểm tra được chọn lọc nhằm đảm bảo tính đa dạng về chủ đề, cách diễn đạt và độ phức tạp, giúp đánh giá chính xác năng lực phân biệt của hệ thống trong các tình huống thực tế. Việc chuẩn bị kỹ lưỡng này giúp đảm bảo đầu vào phù hợp và kết quả đánh giá phản ánh đúng hiệu quả của mô hình.

3.2. Chạy thử nghiệm trên hệ thống

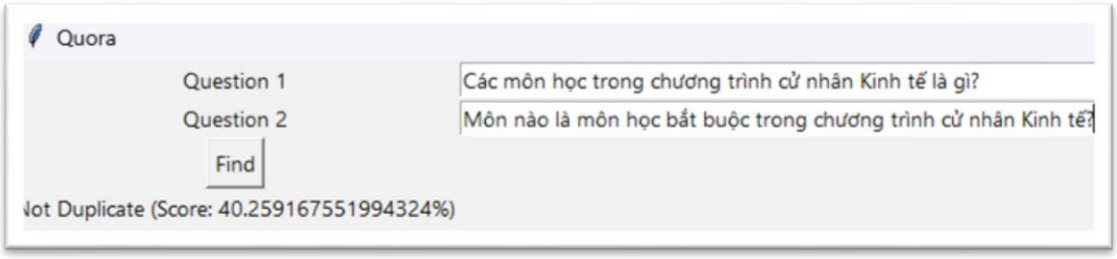
Hệ thống sẽ thực hiện các bước sau:

- Load mô hình PhoBERT đã được fine-tuned cho bài toán phân loại câu hỏi trùng lặp.
- Load tokenizer tương ứng với mô hình PhoBERT.
- Hệ thống sẽ nhận 2 câu hỏi do người dùng nhập vào.
- Sau đó hệ thống sẽ xử lý văn bản như là đổi sang hết chữ thường, lọc và loại bỏ các kí tự đặc biệt, chuyển các từ viết tắt thành từ đầy đủ.
- Tách từ (Tokenization) với PhoBERT.
- Mã hoá câu hỏi thành vector đặc trưng.
- Tính toán độ tương đồng giữa các vector.
- So sánh với giá trị nhãn thực tế để đánh giá hiệu suất.

Nếu xác suất thuộc lớp tương đồng lớn hơn 0,6, dự đoán hai câu hỏi giống nhau.

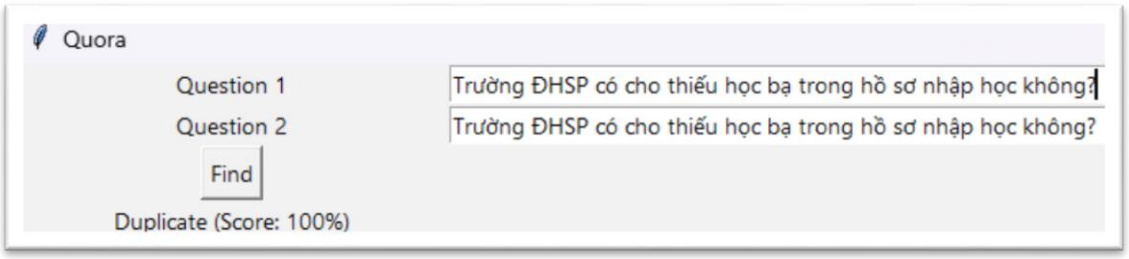
Hình 1. *Dự đoán hai câu hỏi tương đồng*

Kết quả trong Hình 1 minh hoạ dự đoán hai câu hỏi khác nhau về từ ngữ, cách bố trí khác nhau và độ dài khác nhau nhưng đều chung một ý nghĩa, Khi thực hiện so sánh, mô hình dự đoán với kết quả trùng khớp lên tới hơn 0,99 tương đương với hơn 99%, điều này có nghĩa là 2 câu hỏi này là một cặp câu hỏi tương đồng nhau.



Hình *2. Dự đoán hai câu hỏi không tương đồng nhau*

Kết quả trong Hình 2 minh hoạ dự đoán hai câu hỏi khác nhau về từ ngữ, cách bố trí khác nhau và độ dài khác nhau và cũng khác nhau về mặt ngữ nghĩa. Khi thực hiện so sánh, mô hình dự đoán với kết quả trùng khớp khoảng 0,4 tương đương với hơn 40%, điều này có nghĩa là 2 câu hỏi này là một cặp câu hỏi mang ý nghĩa khác nhau và không tương đồng nhau.



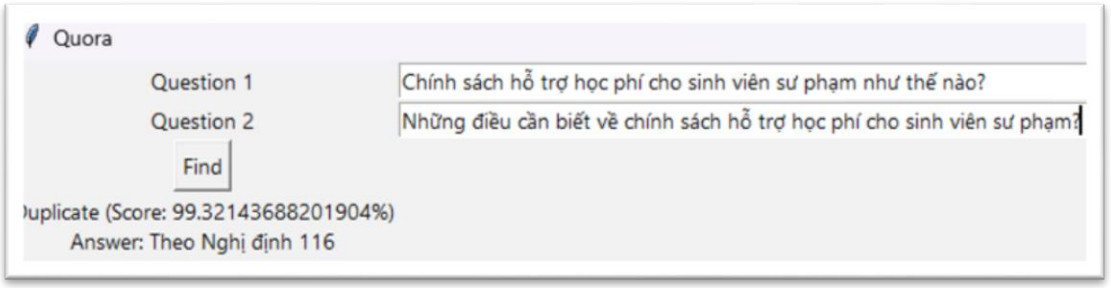
Hình 3. *Dự đoán hai câu hỏi giống nhau*

Kết quả trong Hình 3 minh hoạ dự đoán hai câu hỏi giống nhau, độ tương đồng lên đến 100% vì 2 câu hỏi đều giống nhau về ngữ nghĩa.

3.3. Dự đoán câu trả lời cho câu hỏi

Hệ thống sẽ thực hiện các bước sau nếu kết quả là 2 câu hỏi tương đồng nhau:

- Load mô hình PhoBERT để tạo embeddings (vector đặc trưng) của câu hỏi.
- Load chỉ mục FAISS đã được xây dựng trước đó để tìm kiếm câu trả lời.
- Đọc dữ liệu từ file CSV và encode các câu hỏi thành vector.
- Sử dụng PhoBERT để chuyển câu hỏi thành vector nhúng.



Hình 4. *Dự đoán trả lời câu hỏi*

Kết quả trong Hình 4 minh hoạ dự đoán nếu cặp câu hỏi là tương đồng thì sẽ lấy câu hỏi 1 làm gốc, sử dụng mô hình PhoBERT chuyển câu hỏi thành vector nhúng và sử dụng mô hình FAISS để tìm kiếm câu hỏi có vector tương tự trùng khớp cao nhất trong bộ dữ liệu, sau đó đưa ra câu trả lời của câu hỏi vừa tìm được.

3.4. Kết quả thực nghiệm và phân tích

Bảng 1 trình bày kết quả thực nghiệm mô hình PhoBERT và phương pháp baseline (TF-IDF kết hợp Cosine Similarity) trên tập dữ liệu kiểm tra độc lập.

Bảng 1. *Bảng so sánh định lượng hiệu năng của PhoBERT với TF-IDF + Cosine Similarity*

Phương pháp	Accuracy	Precision	Recall	F1-score
TF-IDF + Cosine Similarity	0,7562	0,7431	0,7685	0,7556
PhoBERT (fine-tuned)	**0,8298**	**0,8270**	**0,8462**	**0,8365**

Accuracy = 0,8298 cho thấy mô hình có khả năng phân loại chính xác cao đối với bài toán nhận diện cặp câu hỏi tương đồng.

Phân tích kết quả:

- Mô hình PhoBERT fine-tuned vượt trội hơn rõ rệt so với phương pháp TF-IDF truyền thống trên tất cả các chỉ số. Đặc biệt, độ chính xác (accuracy) tăng từ 75,62% lên 82,98%, cho thấy khả năng hiểu ngữ nghĩa ngữ cảnh của PhoBERT là vượt trội so với phương pháp dựa trên thống kê từ đơn giản.
- Precision và Recall của PhoBERT cũng cao hơn, chứng tỏ mô hình có khả năng phát hiện tốt hơn các cặp câu hỏi thực sự giống nhau mà vẫn giữ được mức sai lệch thấp. Điều này đặc biệt quan trọng trong các ứng dụng thực tế như hệ thống hỏi đáp, diễn đàn trực tuyến, v.v.

4. Kết luận

Hệ thống so sánh độ tương đồng giữa hai câu hỏi sử dụng mô hình PhoBERT đã được xây dựng và thử nghiệm. Qua quá trình kiểm tra, chúng ta có thể rút ra những điểm quan trọng sau:

- Hiệu suất vượt trội của PhoBERT: Kết quả thử nghiệm cho thấy mô hình PhoBERT có độ chính xác 82,98% trong việc xác định câu hỏi tương đồng. Mô hình này có khả năng hiểu ngữ nghĩa sâu hơn, không chỉ dựa vào sự trùng lặp từ khóa mà còn xem xét bối cảnh của câu hỏi.
- Ứng dụng thực tế: Hệ thống có thể được áp dụng trong nhiều lĩnh vực khác nhau như nền tảng hỏi đáp trực tuyến, chatbot hỗ trợ khách hàng, hoặc hệ thống kiểm tra câu hỏi trùng lặp trong ngân hàng đề thi. Điều này giúp tối ưu hóa quy trình tìm kiếm thông tin và nâng cao trải nghiệm người dùng.
- Những thách thức và hướng phát triển: Mặc dù đạt được kết quả khả quan, hệ thống vẫn còn một số hạn chế cần khắc phục. Việc tinh chỉnh (fine-tuning) PhoBERT trên tập dữ liệu lớn hơn có thể giúp cải thiện hiệu suất. Ngoài ra, các thuật toán đo độ tương đồng có thể được tối ưu hóa hơn nữa, chẳng hạn như kết hợp thêm mạng nơ-ron hoặc các mô hình tiên tiến hơn như transformer đa tầng để tăng độ chính xác.

Nhìn chung, nghiên cứu này đã cung cấp một cách tiếp cận hiệu quả trong việc so sánh câu hỏi tương đồng trên các diễn đàn trực tuyến, mở ra nhiều tiềm năng cho các ứng dụng thông minh hơn trong tương lai.

Lời cảm ơn

Nghiên cứu này được tài trợ bởi nguồn ngân sách khoa học và công nghệ Trường Đại học Sư phạm Thành phố Hồ Chí Minh trong đề tài nghiên cứu khoa học của sinh viên năm học 2024 – 2025.

REFERENCES

[1] Y. Seonwoo, J. Son, J. Jin, S.-W. Lee, J.-H. Kim, J.-W. Ha, and A. Oh, “Two-Step Question Retrieval for Open-Domain QA,” 19 May 2022. [Online]. Available: https://arxiv.org/abs/2205.09393. [Accessed December 24, 2024].

[2] D. Viji and S. Revathy, "A novel approach for paraphrase detection in Tamil language using deep learning models," *Multimedia Tools and Applications*, vol. 81, pp. 18881–18901, 2022, doi: 10.1007/s11042-021-11771-6. [Online]. Available: https://link.springer.com/article/10.1007/s11042-021-11771-6. [Accessed April 21, 2025].

[3] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019, pp. 3982–3992, doi: 10.18653/v1/D19-1410. [Accessed April 23, 2025].

[4] A. Romadhony and A. Hasmawati, "Question Similarity Detection to Handle Similar User Questions using Support Vector Machine," *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, vol. 11, no. 3, pp. 248-257, 2023, doi: 10.23887/janapati.v11i3.52582. [Online]. Available: https://ejournal.undiksha.ac.id/index.php/janapati/article/view/52582. [Accessed March 24, 2025].

[5] S. Gochhait, "Comparative Analysis of Machine and Deep Learning Techniques for Text Classification with Emphasis on Data Preprocessing," *Multimedia Tools and Applications*, vol. 81, no. 5, pp. 7455–7471, 2022, doi: 10.1007/s1042-021-11786-z. [Accessed March 25, 2025].

[6] I. Bandara and F. Ioras, "A deep learning similarity-checking method that can identify patterns of resemblance in duplicated questions can be used to combat the problem of plagiarism," in *ICERI2023 Proceedings*, 2023, pp. 5876-5884, doi: 10.21125/iceri.2023.1464. [Online]. Available: https://library.iated.org/view/BANDARA2023ADE. [Accessed Jan. 30, 2025].

[7] Y. Zhou, C. Li, G. Huang, Q. Guo, H. Li, and X. Wei, "A Short Text Similarity Evaluation Method Combining Syntax and Semantics," *Electronics*, vol. 12, no. 14, 2023, Art. no. 3126, doi: 10.3390/electronics12143126. [Online]. Available: https://www.mdpi.com/2079-9292/12/14/3126. [Accessed Jan. 18, 2025].

[8] D. Q. Nguyen and A. T. Nguyen, "PhoBERT: Pre-trained language models for Vietnamese," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020, pp. 1037-1042, doi: 10.48550/arXiv.2003.00744. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.92. [Accessed March 21, 2025].

[9] P. Q. Long, T. H. P. Doan, L. H. Ngoc, and D. Tran, "Vietnamese Sentence Paraphrase Identification Using Sentence-BERT and PhoBERT," in *Intelligent Things and Technologies for an Green and Smart Environment (ITT-GSE 2022)*, 2022, pp. 416–423, doi: 10.1007/978-3-031-15063-0_40. [Accessed March 30, 2025].

[10] S. Cao, H. Vo, L. T. T. Hang, and D. Dinh, "Hybrid approach for text similarity detection in Vietnamese based on Sentence-BERT and WordNet," in *Proceedings of the 4th International Conference on Information Technology and Computer Communications (ITCC '22)*, 2022, pp. 59–63, doi: 10.1145/3548636.3548645. [Accessed March 25, 2025].

[11] V. D. T. Doan *et al.,* "Dive into Deep Learning," 2025. [Online]. Available: https://d2l.aivivn.com/chapter_recurrent-neural-networks/text-preprocessing_vn.html. [Accessed Feb. 24, 2025].

[12] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with GPUs," *IEEE Transactions on Big Data*, vol. 7, no. 3, pp. 535-547, 2019, doi: 10.1109/TBDATA.2019.2921276.

[13] T. Gao, X. Yao, and D. Chen, "SimCSE: Simple Contrastive Learning of Sentence Embeddings," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 6894–6910, doi: 10.18653/v1/2021.emnlp-main.552.