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

NGUYỄN LỘC PHÚC - NGUYỄN ĐINH QUANG ĐỈNH

HỌC TƯƠNG PHẢN CHO HỆ THỐNG GỢI Ý NỘP BÀI BÁO

LUẬN VĂN CỬ NHÂN TOÁN HỌC

Tp. Hồ Chí Minh - 2024

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

NGUYỄN LỘC PHÚC - NGUYỄN ĐINH QUANG ĐỈNH

HỌC TƯƠNG PHẢN CHO HỆ THỐNG GỢI Ý NỘP BÀI BÁO

LUẬN VĂN CỬ NHÂN TOÁN HỌC

CHUYÊN NGÀNH KHOA HỌC DỮ LIỆU

NGƯỜI HƯỚNG DẪN
PGS.TS. NGUYỄN THANH BÌNH

Tp. Hồ Chí Minh - 2024

Lời cảm ơn

Lời đầu tiên, chúng tôi xin chân thành cảm ơn **trường Đại học Khoa học Tự nhiên, ĐHQG-TPHCM** và **Ban chủ nhiệm Khoa Toán - Tin học** đã tạo điều kiện cho chúng tôi hoàn thành chương trình học nói chung và khóa luận tốt nghiệp nói riêng. Nền tảng kiến thức được tích lũy qua 4 năm học cùng với kinh nghiệm tích lũy qua khoảng thời gian thực hiện khóa luận đã tạo tiền đề rất tốt cho con đường nghiên cứu khoa học về sau của chúng tôi.

Lời cảm ơn thứ hai, chúng tôi xin trân trọng gửi đến **PGS.TS. Nguyễn Thanh Bình** lời cảm ơn chân thành và sâu sắc nhất. Thầy không chỉ là người tạo cảm hứng cho chúng tôi đến với chuyên ngành Khoa học Dữ liệu, mà còn là người nhiệt tình hướng dẫn và cung cấp cho chúng tôi những kiến thức, tài liệu khoa học cần thiết để hoàn thành đề tài này.

Chúng tôi cảm thấy vô cùng biết ơn khi nhận được sự dạy dỗ giúp đỡ tận tình của các thầy, cô giảng viên ở khoa Toán - Tin học. Bên cạnh thầy hướng dẫn, thầy cô không chỉ trao cho chúng tôi kiến thức, kỹ năng mà còn là trách nhiệm và tinh thần làm việc chuyên nghiệp. Nhân dịp này, chúng tôi cũng xin gửi lời cảm ơn thứ ba đến quý thầy cô, kính chúc thầy cô luôn mạnh khỏe, thành công trong sự nghiệp giảng dạy .

Bên cạnh đó, chúng tôi xin gửi lời cảm ơn đến chị Đoàn Thị Trâm, người đã hỗ trợ chúng tôi rất nhiều từ giai đoạn chọn đề tài nghiên cứu đến việc đề xuất những phương pháp hữu ích cho đề tài này..

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

Trân trọng.

Tp.HCM, Ngày 6 tháng 8 năm 2024

Nhóm tác giả

Mục lục

Lời nói đầu

Trong bối cảnh học thuật hiện đại, việc công bố các bài báo khoa học chất lượng cao đóng vai trò quan trọng trong sự phát triển và lan tỏa tri thức. Tuy nhiên, quá trình nộp và đánh giá bài báo khoa học thường gặp nhiều khó khăn do số lượng bài báo ngày càng tăng, cùng với sự đa dạng về chủ đề và phương pháp nghiên cứu. Điều này đặt ra một thách thức lớn đối với các nhà khoa học trong việc lựa chọn tạp chí phù hợp để nộp bài báo của mình.

Hệ thống gợi ý nộp bài báo khoa học được phát triển nhằm hỗ trợ các nhà nghiên cứu trong quá trình này. Với sự trợ giúp của các thuật toán thông minh, hệ thống gợi ý có thể phân tích nội dung của bài báo và đề xuất các tạp chí hoặc hội nghị phù hợp. Điều này không chỉ giúp tiết kiệm thời gian và công sức cho các nhà nghiên cứu mà còn tăng khả năng bài báo được chấp nhận và công bố.

Trong luận văn tốt nghiệp này, chúng tôi đề xuất một mô hình dựa trên kiến trúc Transformer và phương pháp học tương phản cho hệ thống gợi ý nộp bài báo khoa học. Bằng cách kết hợp các thông tin từ bài báo như Title, Abstract và Keywords với Aims & Scope của tạp chí khoa học, mô hình có thể đề xuất các tạp chí hoặc hội nghị tối ưu cho các bài báo. Chúng tôi hy vọng nghiên cứu này có thể góp phần cải thiện các vấn đề liên quan đến nộp bài báo khoa học trong tương lai.

Nội dung luận văn bao gồm 5 chương:

1. Giới thiệu: Trình bày tổng quan về đề tài.

2. Tổng quan lý thuyết: Trình bày các kiến thức liên quan đến dề tài.

3. Phương pháp luận: Trình bày phương pháp tiếp cận đề tài.

4. Thí nghiệm: Trình bày kết quả thực nghiệm và đánh giá hiệu suất mô hình.

5. Kết luận: Trình bày kết luận và hướng đi trong tương lai.

Danh sách từ viết tắt

- **BERT:** Bidirectional Encoder Representation from Transformer
- **CNN:** Convolutional Neural Network.
- **FP8:** Float Point 8.
- **FP32:** Float Point 32.
- **GloVe:** Global Vectors for Word Representation.
- **LLM:** Large Language Model.
- **LLR:** Logistics Linear Regression.
- **LoRA:** Low-Rank Adaptation.
- **LSTM:** Long Short Term Memory.
- **LWAP:** Layer-Wise Attention Pooling.
- **MLP:** Multi-layer Perceptrons.
- **NF4:** Normal Float 4.
- **NLP:** Natural Language Processing.
- **QLoRA:** Efficient Finetuning of Quantized Large Language Models.
- **RoBERTa:** Robustly Optimized BERT Pretraining Approach.
- **SimCPSR:** Simple Contrastive Learning for Paper Submission Recommendation System.
- **SimCSE:** Simple contrastive learning of Sentence embedding.
- **STS-B:** Semantic Textual Similarity Benchmark.
- **TF-IDF:** term frequency - inverse document frequency.

Danh sách thuật ngữ

- **Abstract**: Tóm tắt của bài báo khoa học.
- **Adapter:** Một kỹ thuật để tinh chỉnh các mô hình ngôn ngữ lớn mà không cần phải huấn luyện lại toàn bộ mô hình từ đầu.
- **Aims & Scope:** Mục tiêu và phạm vi nghiên cứu của tạp chí khoa học.
- **Convolutional Neural Network:** Mạng nơ-ron tích chập nhân tạo.
- **[CLS]:** Một token đặc biệt được thêm vào đầu mỗi chuỗi văn bản đầu vào.
- **Cross-entropy:** Một hàm mất mát được sử dụng trong các bài toán phân loại.
- **Decoder:** Bộ giải mã trong kiến trúc sử dụng bộ mã hóa và bộ giải mã.
- **Downstream task:** Tác vụ học có giám sát được cải thiện dựa trên những mô hình được huấn luyện trước.
- **Dequantization:** Quá trình đảo ngược của quantization.
- **Encoder:** Bộ mã hóa trong kiến trúc sử dụng bộ mã hóa và bộ giải mã.
- **Embedding:** Một kỹ thuật đưa một vector có số chiều lớn, thường ở dạng thưa, về một vector có số chiều nhỏ, thường ở dạng dày đặc.
- **Fine-tuning:** Quá trình tinh chỉnh tham số của một mô hình được huấn luyện trước.
- **Framework:** Một thư viện thực hiện các chức năng cơ bản cho một nhiệm vụ lập trình cụ thể.
- **Keywords:** Từ khóa của bài báo khoa học.
- **Large language model:** Mô hình xác suất có khả năng hiểu và sinh ngôn ngữ tự nhiên dựa trên kiến thức được thu thập từ các tập dữ liệu cực lớn.
- **Layer normalization:** Một phương pháp chuẩn hóa được sử dụng trong mạng nơ-ron nhân tạo.
- **Long Short Term Memory:** Dạng đặc biệt của mạng thần kinh hồi quy nhân tạo.

- **[MASK]:** Một token đặc biệt được sử dụng để che dấu một từ hoặc một phần của văn bản trong quá trình huấn luyện mô hình ngôn ngữ
- **Mini-batch:** Một tập hợp con của tập dữ liệu huấn luyện.
- **Neural network:** Mạng nơ-ron nhân tạo.
- **Natural Language Processing:** Một lĩnh vực trong khoa học máy tính và trí tuệ nhân tạo liên quan đến việc tạo ra và hiểu các ngôn ngữ tự nhiên mà con người sử dụng để giao tiếp.
- **Out of Memory:** Một lỗi xảy ra khi một chương trình hoặc hệ thống không còn đủ bộ nhớ để tiếp tục hoạt động.
- **Pooling:** Một kỹ thuật trong mạng nơ-ron tích chập nhân tạo, dùng để giảm kích thước của dữ liệu đầu vào và giữ lại các đặc trưng quan trọng.
- **Pre-trained model:** Mô hình được huấn luyện trước với một bộ dữ liệu lớn để giảm công sức huấn luyện mô hình từ đầu.
- **Quantization:** Quá trình ánh xạ tập hợp chứa các giá trị liên tục vô hạn thành tập hợp nhỏ hơn chứa các giá trị rời rạc hữu hạn.
- **Residual connection:** Một liên kết bổ sung để kết nối một số lớp không liền kề trực tiếp trong mạng nơ-ron nhân tạo.
- **Spearsman's correlation score:** Thước đo phi tham số về tương quan thứ hạng (sự phụ thuộc thống kê giữa thứ hạng của hai biến).
- **Sub-layer:** Lớp con trong mạng nơ-ron nhân tạo.
- **STS-B:** một bộ dữ liệu và bài kiểm tra được sử dụng rộng rãi trong lĩnh vực xử lý ngôn ngữ tự nhiên để đánh giá mức độ tương đồng về ngữ nghĩa giữa các cặp câu.
- **Title:** Tiêu đề của bài báo khoa học.
- **Token:** Một đơn vị cơ bản của văn bản sau khi được tách từ chuỗi văn bản ban đầu.
- **Tokenizer:** Một công cụ trong xử lý ngôn ngữ tự nhiên có nhiệm vụ tách một đoạn văn bản thành các đơn vị nhỏ hơn, thường là các từ hoặc các token.

Danh sách bảng

Danh sách hình vẽ

Chương 1

Giới thiệu

1.1 Lý do chọn đề tài

Hệ thống gợi ý ngày càng trở nên phổ biến và được sử dụng trong nhiều lĩnh vực khác nhau như bán lẻ, truyền thông, tin tức, dịch vụ phát trực tuyến và thương mại điện tử. Thấy được lợi ích của hệ thống gợi ý đối với sự phát triển của nền kinh tế, nhiều công ty đã xây dựng hệ thống gợi ý dựa trên lịch sử tương tác của khách hàng để cung cấp cho họ những đề xuất có liên quan, nhằm đáp ứng nhu cầu của khách hàng và cải thiện chất lượng sản phẩm. Các hệ thống gợi ý nổi tiếng hiện nay gồm có Spotify và Netflix trong lĩnh vực phát trực tuyến, Amazon trong lĩnh vực thương mại điện tử, Google, Facebook và Youtube trong lĩnh vực truyền thông. Bên cạnh đó, hệ thống gợi ý trong lĩnh vực học thuật bắt đầu trở nên quan trọng trong những năm gần đây, chẳng hạn như hệ thống gợi ý nộp bài báo khoa học, hệ thống gợi ý cơ sở tri thức và hệ thống đề xuất bài báo khoa học. Trong đó việc lựa chọn một tạp chí hoặc hội nghị khoa học uy tín và phù hợp không phải là điều dễ dàng đối với hầu hết các nhà nghiên cứu, đặc biệt là với những người trẻ và không có kinh nghiệm. Dựa trên ý tưởng này, chúng tôi quyết định chọn đề tài **học tương phản cho hệ thống gợi ý nộp bài báo**. Đề tài tập trung vào việc nghiên cứu và đề xuất hướng tiếp cận mới cho bài toán hệ thống gợi ý nộp bài báo dựa trên việc kết hợp kiến trúc transformer và phương pháp học tương phản.

1.2 Trình bày bài toán

Ý tưởng về hệ thống gợi ý nộp bài báo khoa học được phát triển bởi nhóm tác giả trong bài báo A content-based recommender system for computer science publications [1]. Nhóm tác giả sử dụng thống kê Chi bình phương và chỉ số term frequency-inverse document frequency (TF-IDF) cho việc lực chọn đặc trưng từ Abstract của mỗi bài báo, kết hợp với mô hình hồi quy logistic để phân loại các tạp chí hoặc hội nghị khoa học có liên quan. Họ áp dụng hai mô hình học máy là Logistics Linear Regression (LLR) và Multi-layer Perceptrons (MLP) cho nhiều tổ hợp các đặc trưng khác nhau. Phương pháp do nhóm tác giả đề xuất đạt được kết

quả vượt trội cho Top 3 - Accuracy, đặc biệt là 88, 60% đối với mô hình LLR và 89, 07% đối với mô hình MLP khi sử dụng các đặc trưng bao gồm Title, Abstract và Keywords.

Cũng nghiên cứu về vấn đề này, một nhóm tác giả khác đã đề xuất một hướng tiếp cận mới trong bài báo A new approach for paper submission recommendation [2]. Nhóm tác giả này áp dụng hai phương pháp nhúng Global Vectors for Word Representation (GloVe) [3] và FastText [4] kết hợp với mô hình Convolutional Neural Network (CNN) và Long Short Term Memory (LSTM) để trích xuất đặc trưng. Họ xem xét bảy tổ hợp đặc trưng khác nhau bao gồm: Title, Abstract, Keywords, Title + Keywords, Title + Abstract, Keywords + Abstract, and Title + Keywords + Abstract cho quá trình huấn luyện. Kết quả thực nghiệm cho thấy mô hình được đề xuất có kết quả tốt nhất với Top 1 - Accuracy là 68, 11% khi sử dụng nhóm đặc trưng Title + Abstract + Keyword. Top 3, 4, 5 - Accuracy lần lượt là 90, 8%, 96, 25%, và 99, 21%. Sau đó, nhóm tác giả này tiếp tục đề xuất một hướng tiếp cận mới trong bài báo A fusion approach for paper submission recommendation system [5]. Trong bài báo này, nhóm tác giả bổ sung thêm đặc trưng Aims & Scope của tạp chí khoa học vào tập dữ liệu đầu vào. Họ thu thập tập bộ dữ liệu chứa 414512 bài báo và các tạp chí tương ứng. Họ xây dựng một kiến trúc mới vẫn sử dụng FastText làm phương pháp nhúng và dữ liệu đầu vào là thông tin từ bài báo. Họ tạo ra một đặc trưng mới bằng cách đo lường độ tương đồng giữa thông tin bài báo và tạp chí khoa học. Phương pháp đề xuất của nhóm tác giả này là một hướng tiếp cận khả quan để giải quyết bài toán gợi ý nộp bài báo.

Trong luận văn này, chúng tôi nghiên cứu việc xây dựng hệ thống gợi ý bằng phương pháp tiếp cận sử dụng thông tin từ bài báo và tạp chí khoa học. Mục tiêu của chúng tôi là trích xuất mối quan hệ ngữ nghĩa giữa bài báo và tạp chí khoa học thông qua các tổ hợp đặc trưng một cách tốt nhất có thể. Chúng tôi giải quyết vấn đề này bằng cách áp dụng kiến trúc transformer để trích xuất đặc trưng đầu vào một cách hiệu quả và sử dụng phương pháp học tương phản để tăng hiệu suất của mô hình trong downstream task.

Đóng góp của chúng tôi có thể được tóm tắt như sau:

- Chúng tôi đề xuất một framework mới cho vấn đề gợi ý nộp bài báo bằng cách sử dụng kiến trúc transformer. Kết quả thực nghiệm ở chương 4 cho thấy phương pháp của chúng tôi có hiệu suất cạnh tranh so với các công trình trước đây.

- Bằng cách tận dụng học tương phản kết hợp với phương pháp chọn layer hiệu quả, chúng tôi tăng hiệu suất của framework trong việc học các mối quan hệ ngữ nghĩa giữa các tài liệu hoặc câu.

- Phương pháp của chúng tôi cung cấp một framework cơ bản có thể được mở rộng thêm bằng cách áp dụng các mô hình transformer khác để đạt được hiệu suất tốt hơn trong vấn đề gợi ý nộp bài báo.

Chương 2

Tổng quan lý thuyết

2.1 Hệ thống gợi ý dựa trên neural network

Hệ thống gợi ý thường được chia thành hai dạng là hệ thống gợi ý dựa trên lọc cộng tác và hệ thống gợi ý dựa trên nội dung. Ý tưởng chung của hai dạng này là tìm cách ước lượng xếp hạng của những cặp người dùng - vật phẩm chưa được xếp hạng bởi người dùng sao cho sai số so với giá trị thực tế là nhỏ nhất. Tuy nhiên những phương pháp này đều có nhưng hạn chế: Nếu một người dùng hoặc vật phẩm chưa từng xuất hiện trong cơ sở dữ liệu thì sẽ không thể xây dựng được phương trình hồi qui để dự báo xếp hạng cho vật phẩm, do đó không thể gợi ý được cho khách hàng mới; những vật phẩm được sử dụng phổ biến và vật phẩm có xếp hạng cao thì xuất hiện trong hầu hết các gợi ý, chứng tỏ tính cá nhân hóa của các phương pháp này chưa cao.

Để khắc phục hạn chế trên chúng ta dựa vào một phương pháp dựa trên neural network giúp xây dựng một embedding vector nhằm trích suất các đặc trưng của người dùng và vật phẩm. Từ đó có thể dễ dàng áp dụng để tìm ra các vật phẩm hoặc tập khách hàng tương dựa trên thước đo như hệ số cosine similarity. Trên thực tế, số lượng dữ liệu đối với khách hàng hoặc đối với vật phẩm ở một số hệ thống là rất lớn. Nếu sử dụng những mô hình truyền thống sẽ gặp hạn chế lớn là không tận dụng được toàn bộ các thông tin này. Ngoài ra mô hình neural network có ưu điểm đó là tiếp nhận dữ liệu đầu vào rất đa dạng. Chính vì ưu điểm này mà hệ thống gợi ý dựa trên neural network được ưa chuộng hơn các phương pháp cũ.

Kiến trúc của một hệ thống gợi ý dựa trên neural network đơn giản bao gồm: Input layer là thông tin của khách hàng, thông tin vật phẩm, lịch sử giao dịch hoặc truy vấn; hidden layer để giảm chiều dữ liệu và chuyển đổi tính phi tuyến; output layer một véc tơ phân phối xác suất về khả năng ưa thích hoặc một embedding vector điểm xếp hạng của khách hàng.

2.2 Kiến trúc Transformer

Kiến trúc transformer [6] là một kiến trúc neural network sử dụng cho NLP, nổi bật với việc sử dụng cơ chế attention. Cơ chế attention cho phép neural network

tập trung vào các phần quan trọng của dữ liệu đầu vào mà không cần sử dụng các phép tích chập hoặc mạng LSTM. Sự có mặt của cơ chế attention trong kiến trúc transformer mang lại nhiều ảnh hưởng tích cực lên các tác vụ NLP.

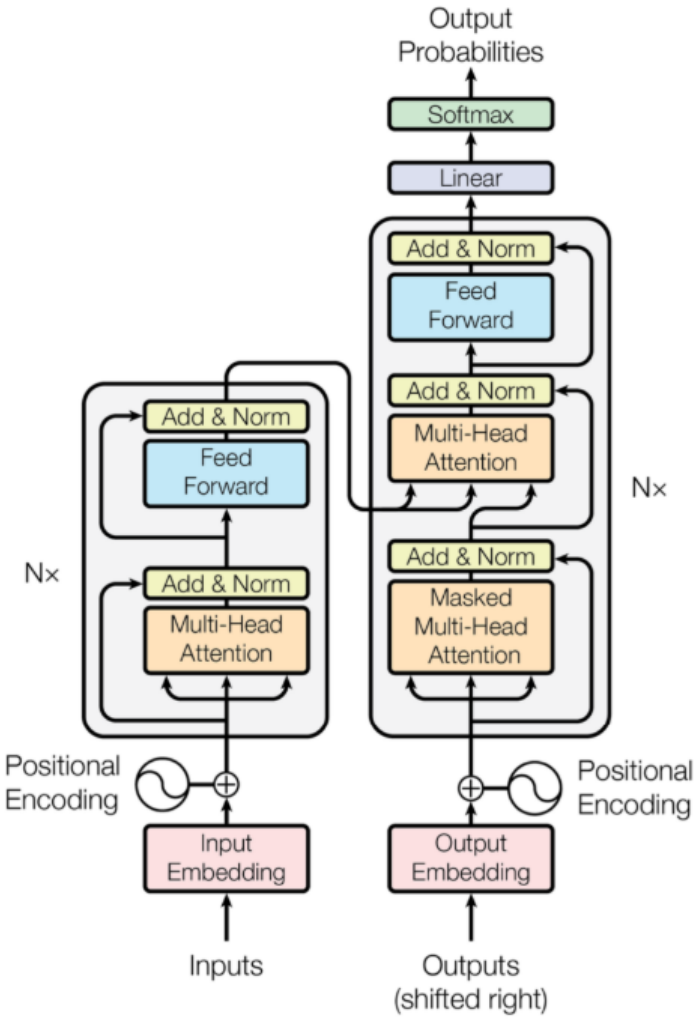
Kiến trúc transfomer bao gồm các thành phần như sau: Encoder là tổng hợp xếp chồng lên nhau của sáu layer xác định. Mỗi layer bao gồm hai sub-layer. Sub-layer đầu tiên là multi-head self-attention. Sub-layer thứ hai là fully-connected feed-forward network. Một lưu ý là chúng ta sẽ sử dụng một residual connection ở mỗi sub-layer ngay sau layer normalization. Đầu ra của mỗi sub-layer là có số chiều là 512; Decoder cũng là tổng hợp xếp chồng của sáu layer. Kiến trúc tương tự như các sub-layer ở encoder ngoại trừ thêm một sub-layer thể hiện phân phối attention ở vị trí đầu tiên. Layer này không gì khác so với multi-head self-attention layer ngoại trừ được điều chỉnh để không đưa các từ trong tương lai vào attention. Tại bước thứ i của decoder chúng ta chỉ biết được các từ ở vị trí nhỏ hơn i nên việc điều chỉnh đảm bảo attention chỉ áp dụng cho những từ nhỏ hơn vị trí thứ i. Cơ chế residual cũng được áp dụng tương tự như trong encoder.

Cơ chế attention có thể được hiểu là một ánh xạ chiếu một truy vấn và một tập hợp các cặp khóa - giá trị tới một đầu ra, trong đó truy vấn, khóa, giá trị và đầu ra đều là các vector. Đầu ra được tính toán dưới dạng tổng trọng số của các giá trị, trong đó trọng số được gán cho mỗi giá trị được tính bằng hàm tương thích của truy vấn với khóa tương ứng. Trong kiến trúc transformer chúng ta áp dụng hai dạng attention khác nhau: Scaled dot-product attention là một cơ chế self-attention khi mỗi từ có thể điều chỉnh trọng số của nó cho các từ khác trong câu sao cho từ ở vị trí càng gần nó nhất thì trọng số càng lớn và càng xa thì càng nhỏ dần; Multi-head attention là quá trình lặp đi lặp lại scaled dot-product attention. Multi-head attention cho phép mô hình hiểu được sự liên quan giữa các từ trong một câu.

2.3 Bidirectional Encoder Representation from Transformer

Bidirectional Encoder Representation from Transformer (BERT) [7] là mô hình biểu diễn từ theo hai chiều ứng dụng kỹ thuật transformer. BERT được thiết kế để huấn luyện trước các biểu diễn từ (pre-train word embedding). Điểm đặc biệt ở BERT đó là nó có thể điều hòa cân bằng bối cảnh theo cả hai chiều trái và phải. Điểm nổi bật của BERT là sử dụng Masked Language Model (MLM) và Next Sentence Prediction (NSP).

MLM là một tác vụ cho phép chúng ta fine-tuning lại các biểu diễn từ trên các bộ dữ liệu văn bản không có nhãn bất kỳ. Trong mô hình BERT, khoảng 15% các token của câu đầu vào được thay thế bởi [MASK] token trước khi truyền vào mô hình đại diện cho những từ bị che dấu (masked). Mô hình sẽ dựa trên các từ không được che (non-masked) dấu xung quanh [MASK] và đồng thời là bối cảnh của [MASK] để dự báo giá trị gốc của từ được che dấu. Số lượng từ được che dấu được lựa chọn là một số ít (15%) để tỷ lệ bối cảnh chiếm nhiều hơn (85%). Để tính toán phân phối xác suất cho từ đầu ra, BERT thêm một fully-connected layer



Hình 2.1: Kiến trúc transformer bao gồm encoder bên trái và decoder bên phải [6].

ngay sau transformer encoder. Hàm loss function của BERT sẽ bỏ qua mất mát từ những từ không bị che dấu và chỉ đưa vào mất mát của những từ bị che dấu. Việc lựa chọn ngẫu nhiên 15% số lượng các từ bị che dấu cũng tạo ra vô số các kịch bản đầu ra cho mô hình huấn luyện nên mô hình sẽ cần phải huấn luyện rất lâu mới học được toàn diện các khả năng.

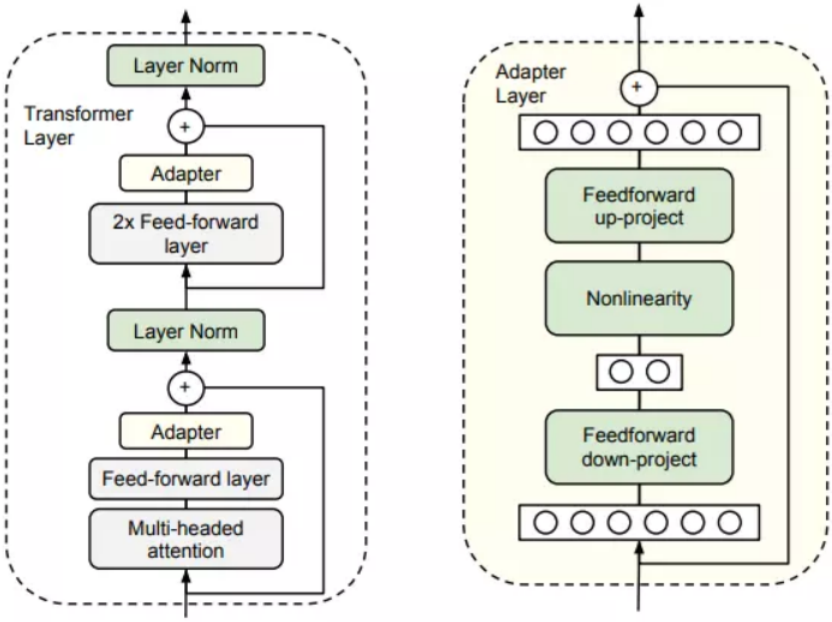
NSP là một tác vụ phân loại học có giám sát với hai nhãn (hay còn gọi là phân loại nhị phân). Đầu vào của mô hình là một cặp câu sao cho xác suất 50% câu thứ hai được lựa chọn là câu tiếp theo của câu thứ nhất và 50% được lựa chọn một cách ngẫu nhiên từ bộ văn bản mà không có mối liên hệ gì với câu thứ nhất. Nhãn của mô hình sẽ tương ứng với isnext khi cặp câu là liên tiếp hoặc notnext nếu cặp câu không liên tiếp.

Robustly Optimized BERT Pretraining Approach (RoBERTa) [8] là một biến thể của mô hình BERT, được phát triển bởi nhóm nghiên cứu của Facebook AI vào năm 2019. Mô hình này được tối ưu hóa mạnh mẽ với một loạt các biến đổi và cải tiến nhằm cải thiện hiệu suất trên các tác vụ NLP. RoBERTa giữ lại kiến trúc cơ bản của kiến trúc transformer như BERT, nhưng áp dụng một số cải tiến trong quá trình huấn luyện và điều chỉnh siêu tham số. Cụ thể, RoBERTa loại bỏ NSP

và thay đổi phương pháp tạo mini-batch để tăng cường hiệu suất. Song song với hiệu suất vượt trội, mô hình phải đối mặt với một thách thức là số lượng tham số rất lớn dẫn tới việc huấn luyện mô hình tốn kém về mặt thời gian và tài nguyên.

2.4 Low-Rank Adaptation

Trong bài báo Parameter-efficient Fine-tuning [9], nhóm tác giả đã nghĩ ra một cách fine-tuning hiệu quả và áp dụng vào BERT-large: Ở mỗi khối transformer trong mô hình, ta chèn thêm hai lớp adapter trước khi thực hiện fine-tuning. Việc làm này giúp ta chỉ thực hiện huấn luyện với các lớp adapter được chèn thêm vào mô hình, và đóng băng toàn bộ pre-trained model trong quá trình fine-tuning. Phương pháp này làm cho số lượng tham số giảm đi đáng kể, do đó quá trình huấn luyện và lưu trữ sẽ tiêu tốn tài nguyên ít hơn rất nhiều. Tuy nhiên, việc chèn vào mô hình các lớp adapter sẽ làm độ nặng tính toán tăng lên, và không có cách trực tiếp nào để loại bỏ quá trình tính toán thêm này của adapter. Để giải quyết vấn đề này, nhóm tác giả trong bài báo Low-Rank Adaptation of Large Language Models (LoRA) [10] đã đưa ra một phương pháp phân rã ma trận. Mục đích của phương pháp này là tìm cách biểu diễn ma trận trọng số thành hai ma trận có hạng thấp hơn rất nhiều so với ma trận ban đầu. Điểm hạn chế của LoRA là chỉ áp dụng với linear layer.



Hình 2.2: Hai lớp adapter được chèn vào mỗi khối transformer [9].

Chương 3

Phương pháp luận

Trong chương này chúng tôi trình bày phương pháp tiếp cận đối với bài toán hệ thống gợi ý nộp bài báo và độ đo đánh giá hiệu suất mô hình.

3.1 Các phương pháp sử dụng

3.1.1 Học tương phản

Học tương phản gần đây đã trở thành một trong những phương pháp phổ biến được áp dụng cho các bài toán học có giám sát và không giám sát. Ý tưởng chính của học tương phản là tìm ra các cặp dữ liệu có tính tương đồng - không tương đồng trong tập dữ liệu bằng cách kéo những cặp dữ liệu tương đồng lại gần nhau, và đẩy những cặp không tương đồng học ra xa trong không gian embedding. Điểm mạnh của phương pháp này là tận dụng tối đa được các pre-trained model thông qua quá trình fine-tuning, từ đó có thể nâng cao hiệu suất của nhiều downstream task.

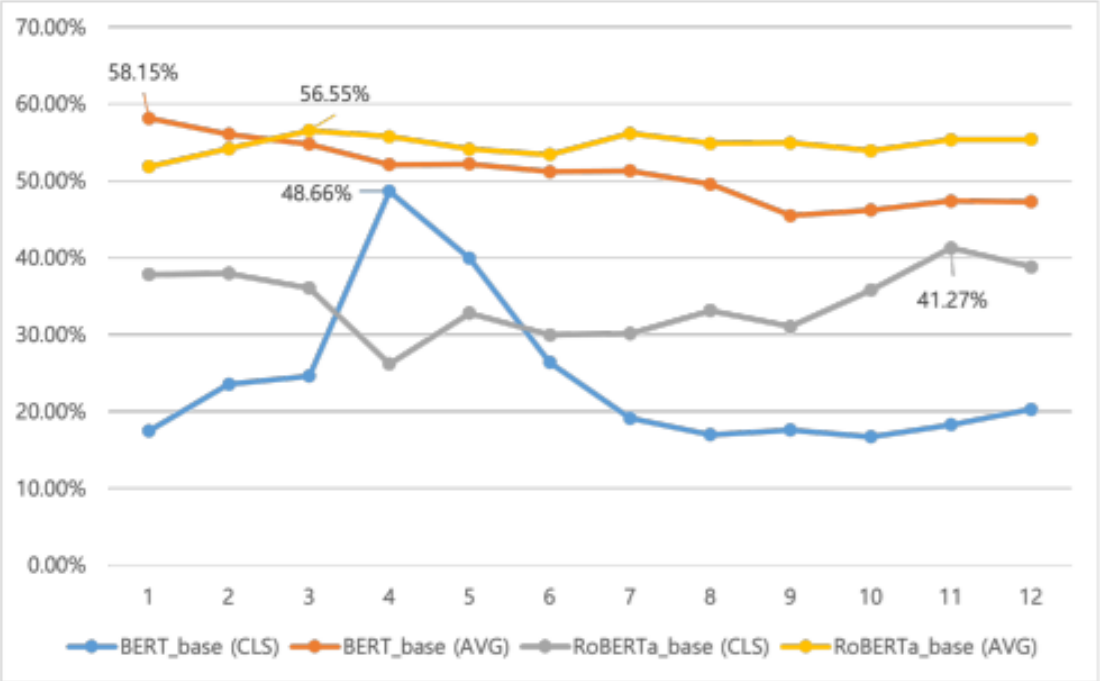
Đầu tiên, chúng tôi sử dụng tập dữ liệu huấn luyện có dán nhãn (được trình bày trong mục 4.2) để xây dựng tập hợp các cặp dữ liệu tương đồng $\mathcal{D} = \left\{\left(x_i, x_i^+\right)\right\}_{i=1}^n$ cho quá trình fine-tuning, trong đó x bao gồm Title, Abstract và Keywords của bài báo và x_i^+ bao gồm Aims & Scope của tạp chí khoa học. Tiếp theo, chúng tôi sẽ dựa vào framework được mô tả trong bài báo Simple contrastive learning of sentence embeddings (SimCSE) [11]. Với mỗi mini-batch gồm N cặp từ tập dữ liệu $\mathcal{D}$, đặt $\mathbf{h}_i$ và $\mathbf{h}_i^+$ lần lượt là các embedding vector của x_i và x_i^+; hàm mục tiêu của học tương phản được định nghĩa như sau

$$\mathcal{L} = -\sum_{i=1}^{N} \log \frac{e^{\text{sim}(\mathbf{h}_i, \mathbf{h}_i^+)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}(\mathbf{h}_i, \mathbf{h}_j^+)/\tau}}$$

trong đó τ là một siêu tham số và $\text{sim}(\mathbf{h}_1, \mathbf{h}_2)$ là độ tương đồng cosine được xác định theo công thức

$$\text{sim}(\mathbf{h}_1, \mathbf{h}_2) = \frac{\mathbf{h}_1^T \mathbf{h}_2}{||\mathbf{h}_1||, ||\mathbf{h}_2||}$$

3.1.2 Layer-Wise Attention Pooling



Hình 3.1: Hệ số tương quan Spearman của mỗi layer được đánh giá trên tập dữ liệu STS-B [13].

Thông thường, để trích xuất biểu diễn câu một cách hiệu quả, các pre-trained model sử dụng [CLS] token ở layer cuối cùng, hoặc AVG token hay còn gọi là biểu diễn trung bình của các token ở layer cuối cùng, hoặc First-Last AVG token hay còn gọi là biểu diễn trung bình của các token ở layer đầu tiên và cuối cùng. Tuy nhiên, các tác giả trong bài báo Contrastive Learning with Layer-Wise Attention Pooling (LWAP) [12] đã chỉ ra rằng không chỉ mỗi layer cuối cùng, mà các layer khác cũng chứa một lượng thông tin đáng kể cho việc biểu diễn câu. Từ đó, họ đề xuất phương pháp LWAP để gán trọng số cho mỗi layer, mục đích để tìm được layer thích hợp giúp mô hình học được việc biểu diễn câu hiệu quả đối với mỗi tác vụ được giao.

Tiếp theo, chúng tôi xây dựng LWAP dựa theo mô tả trong bài báo Contrastive Learning with LWAP. Trong phương trình (1), h^a là vector biểu diễn AVG token, h^c là vector biểu diễn [CLS] token, α_i là hệ số biểu diễn mức độ quan trọng của layer thứ i. Trong phương trình (2), h_i^l là vector biểu diễn mức độ quan trọng của mỗi layer. Trong phương trình (3), h^L là vector trung bình của các h_i^l, đồng thời nó cũng biểu diễn độ tương quan của tất cả các layer ($\mathcal{N}$ là số lượng layer). W_k, W_q và W_v là các tham số cần phải học của mô hình.

$$\alpha_i = \frac{W_q h_i^c W_k h_i^a}{\sum_{j \in \mathcal{N}} W_q h_i^c W_k h_j^a} \quad (1)$$

$$h_i^l = \sum_{j \in \mathcal{N}} \alpha W_v h_j^a \quad (2)$$

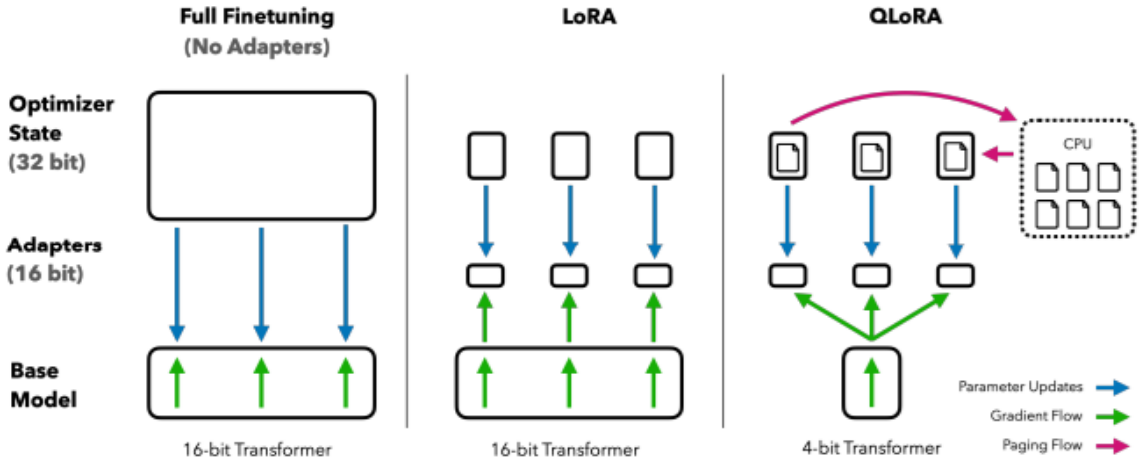
$$h^L = \frac{1}{\mathcal{N}} \sum^{\mathcal{N}_i} h_i^l \quad (3)$$

Chúng tôi thêm một MLP được khởi tạo ngẫu nhiên sau bước pooling, tương tự như phương pháp được mô tả trong bài báo SimCSE, và giữ nguyên việc khởi tạo ngẫu nhiên này. Trong phương trình (4), h_{last}^c là vector biểu diễn [CLS] token của layer cuối cùng. Chúng tôi kết hợp h_{last}^c với h^L làm đầu vào cho MLP. Trong phương trình (5), h có cùng số chiều với số chiều biểu diễn câu của pre-trained model ban đầu thông qua MLP.

$$h^{CL} = \left[h_{last}^c; h^L\right] \quad (4)$$

$$h = MLP\left(h^{CL}\right) \quad (5)$$

3.1.3 Quantized Low-Rank Adaptation



Hình 3.2: Các phương pháp fine-tuning khác nhau và yêu cầu bộ nhớ của chúng. QLoRA cải tiến so với LoRA bằng cách thực hiện quantization mô hình transformer xuống độ chính xác 4-bit và sử dụng page optimizers để xử lý các bộ nhớ.[13].

Quá trình fine-tuning pre-trained large language model (LLM) thường sẽ có số lượng tham số lên đến hàng tỷ, do đó quá trình này yêu cầu bộ nhớ có dung lượng từ hàng chục đến hàng trăm gigabyte. Mặc dù gần đây có rất nhiều phương pháp quantization giúp giảm dung lượng bộ nhớ của LLM, hầu hết các phương pháp này gặp những hạn chế nhất định trong quá trình huấn luyện. Xuất phát từ thách thức trên, các tác giả trong bài báo Efficient Finetuning of Quantized LLMs (QLoRA) [13] đã đề xuất một hướng tiếp cận mới. Mô hình QLoRA là sự kết hợp giữa kỹ thuật quantization và mô hình LoRA. QLoRA giới thiệu ba điểm mới: Normal Float 4, double quantization và page optimizers.

Normal Float 4 (NF4) là một dtype sử dụng 4 bit biểu diễn; có giá trị nằm trong khoảng $[-1, 1]$; tập giá trị phân bố bất đối xứng, có sự biểu diễn cho giá trị 0; được tạo ra để áp dụng cho vector tuân theo phân phối chuẩn $\mathcal{N}(0, 1)$.

Double quantization là kỹ thuật thực hiện quantization hai lần. Thay vì thực hiện quantization với một vector, ta chia vector đó thành nhiều đoạn nhỏ rồi thực hiện thực hiện quantization với mỗi đoạn. Khi đó, mỗi đoạn sẽ có hằng số quantization riêng và bộ nhớ để lưu trữ hằng số quantization cũng từ đó mà tăng lên. Vì vậy, QLoRA thực hiện quantization cả hằng số quantization của mỗi đoạn.

Page optimizers là phương pháp để khác phục hiện tượng Out of Memorry khi huấn luyện mô hình với GPU.

Tiếp theo chúng tôi xây dựng phương pháp quantization cho mỗi linear layer thuộc pre-trained model dựa theo bài báo QLoRA. Mỗi layer bao gồm hai thành phần: thành phần pre-trained và thành phần LoRA. Quá trình tính toán đầu ra Y từ đầu vào X dạng Float Point 32 (FP32) của linear layer với QLoRA như sau

$$\mathrm{Y}^{\mathrm{FP32}} = \mathrm{X}^{\mathrm{FP32}}\mathrm{doubleDequant}\left(c_1^{\mathrm{FP32}}, c_2^{\mathrm{FP8}}, \mathrm{W}^{\mathrm{NF4}}\right) + \mathrm{X}^{\mathrm{FP32}}\mathrm{L}_1^{\mathrm{FP32}}\mathrm{L}_2^{\mathrm{FP32}}$$

trong đó W là ma trận trọng số tương ứng với X; L_1, L_2 là hai ma trận phân rã của ma trận W; c_1, c_2 là các hằng số quantizaton; hàm doubleDequant và hàm dequant được xác định theo công thức

$$\mathrm{doubleDequant}\left(c_1^{\mathrm{FP32}}, c_2^{\mathrm{FP8}}, \mathrm{W}^{\mathrm{NF4}}\right) = \mathrm{dequant}\left(\mathrm{dequant}\left(c_1^{\mathrm{FP32}}, c_2^{\mathrm{FP8}}\right), \mathrm{W}^{\mathrm{NF4}}\right)$$

$$\mathrm{dequant}\left(c_1^{\mathrm{FP32}}, c_2^{\mathrm{FP8}}\right) = \frac{c_2^{\mathrm{FP8}}}{c_1^{\mathrm{FP32}}} = c_3^{\mathrm{FP32}}$$

$$\mathrm{dequant}\left(c_3^{\mathrm{FP32}}, \mathrm{W}^{\mathrm{NF4}}\right) = \frac{\mathrm{W}^{\mathrm{NF4}}}{c_3^{\mathrm{FP32}}} = \mathrm{W}^{\mathrm{FP32}}$$

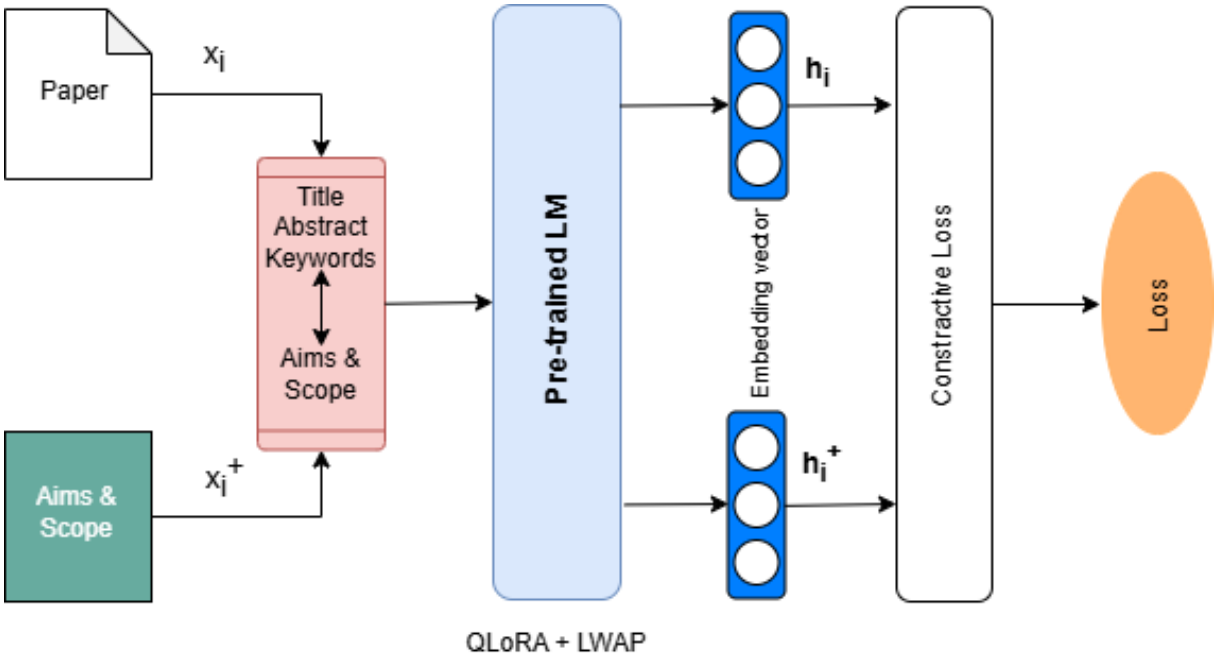
Trong quá trình tính toán đầu ra, thành phần pretrained sẽ được dequantization từ NF4 về FP32, rồi kết hợp tính toán với thành phần LoRA ở dạng FP32. Khi tính xong thì thành phần pre-trained lại được quantization lại về NF4. Vì vậy, QLoRA giúp giảm dung lượng bộ nhớ của mô hình trong quá trình huấn luyện, nhưng không tăng tốc độ của mô hình.

3.2 Xây dựng mô hình

Mô hình của chúng tôi được tham khảo dựa trên bài báo SimCPSR [14]. Mô hình bao gồm hai giai đoạn. Trong giai đoạn đầu, chúng tôi thực hiện fine-tuning một pre-trained model, có sử dụng hàm mục tiêu của học tương phản, nhằm mục đích mã hóa dữ liệu đầu vào thành các embedding vector một cách hiệu quả. Điểm khác biệt giữa mô hình của chúng tôi và SimCPSR là việc bổ sung QLoRA và LWAP vào pre-trained model; ngoài ra chúng tôi sử dụng pre-trained model RoBERTa thay vì Distil-RoBERTa (phiên bản nhỏ gọn hơn của RoBERTa). Trong giai đoạn hai, chúng tôi áp dụng pre-trained model đã thực hiện fine-tuning với tổ hợp các đặc trưng khác nhau từ tập dữ liệu cho downstream task, nhằm mục đích phân loại được Top-K Accuracy.

3.2.1 Fine-tuning

Đầu tiên, chúng tôi xem Aims & Scopes của tạp chí khoa học tương ứng với Title, Abstract và Keywords của bài báo là cặp tương đồng Sau đó chúng tôi thực hiện fine-tuning pre-trained model RoBERTa kết hợp với hàm mục tiêu của học tương phản trên toàn bộ các cặp tương đồng trong tập dữ liệu. Chúng tôi sử dụng mô hình QLoRA để giảm dung lượng bộ nhớ của RoBERTa trong quá trình fine-tuning. Chúng tôi cũng thêm LWAP vào mỗi layer trong mô hình RoBERTa, để tìm ra được layer có biểu diễn câu hiệu quả nhất.



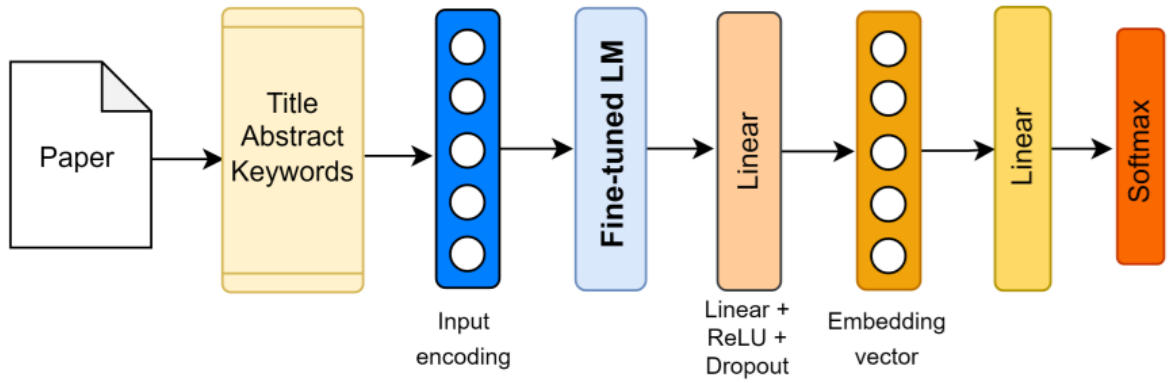
Hình 3.3: Quá trình fine-tuning pre-trained model sử dụng hàm mục tiêu của học tương phản, trong đó x_i, x_i^+ là các cặp tương đồng trong bộ dữ liệu và h_i, h_i^+ là các cặp embedding vector tương ứng.

3.2.2 Downstream task

Chúng tôi xem pre-trained model đã thực hiện fine-tuning là khuôn mẫu cho tác vụ nhận dạng. Do đó, chúng tôi sẽ huấn luyện nó với nhiều tổ hợp các đặc trưng khác nhau, bao gồm sử dụng các đặc trưng từ bài báo hoặc kết hợp các đặc trưng từ bài báo và tạp chí khoa học. Trong chương này, chúng tôi sẽ trình bày hai mô hình khác nhau cho downstream task.

Mô hình sử dụng thông tin từ bài báo

Chúng tôi sử dụng ba đặc trưng từ bài báo bao gồm Title (T), Abstract(A), và Keywords (K). Các đặc trưng này tổ hợp với nhau tạo thành: Title(T), Abstract



Hình 3.4: Kiến trúc của mô hình sử dụng thông tin từ bài báo

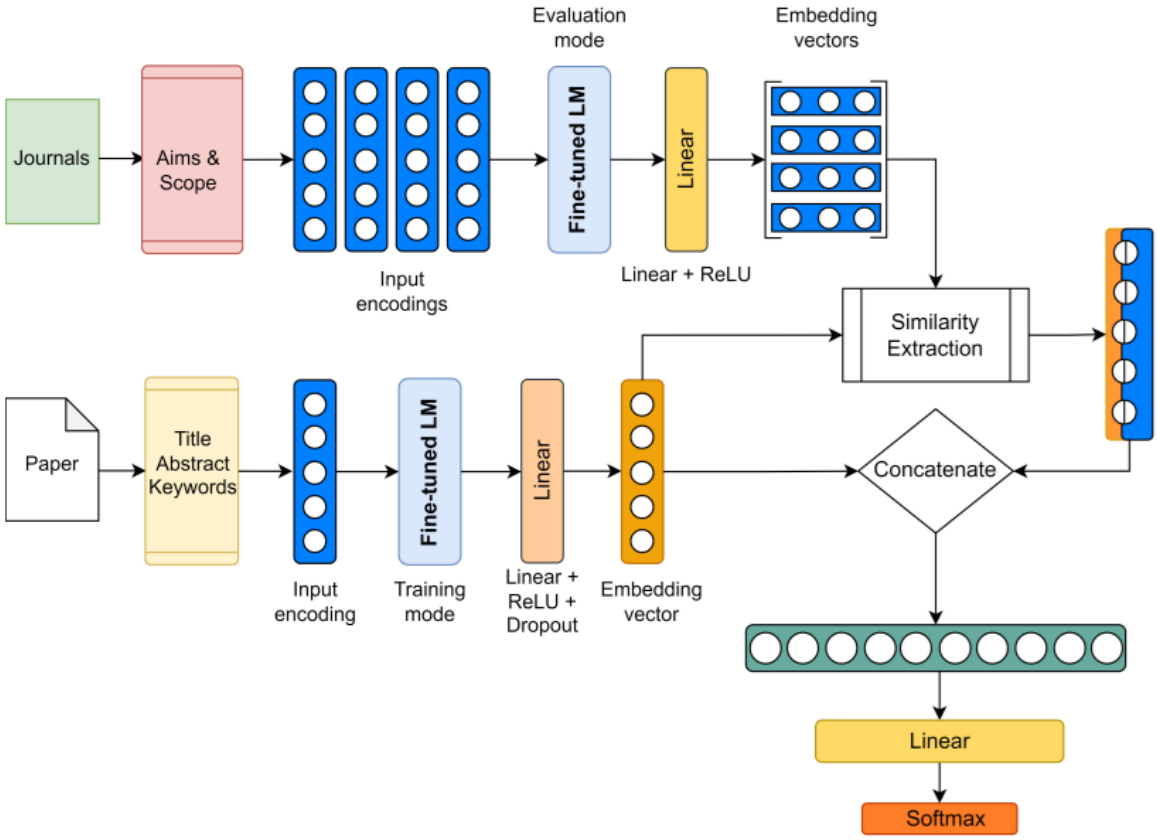
(A), Keywords (K), Title + Abstract (TA), Title + Keywords (TK), Abstract + Keywords (AK), Title + Abstract + Keywords (TAK).

Đầu tiên, chúng tôi sử dụng pre-trained model đã thực hiện fine-tuning để mã hóa dữ liệu đầu vào thành các embedding vector, chúng tôi cũng bổ sung một linear layer với ReLU làm hàm kích hoạt và dropout để tránh bị overfitting. Sau đó, chúng tôi sử dụng một linear layer với softmax làm hàm kích hoạt nhằm đưa đầu ra về khoảng giá trị $(0, 1]$, tương ứng với xác suất để bài báo khoa học thuộc về tạp chí tương ứng. Để xác định được K - tạp chí khoa học liên quan mà xác suất được chấp nhận cao nhất, chúng tôi chọn Top K - giá trị cao nhất.

Mô hình sử dụng thông tin từ bài báo và tạp chí khoa học

Chúng tôi bổ sung thêm đặc trưng Aims & Scope từ các tạp chí khoa học. Các đặc trưng này tổ hợp với nhau tạo thành: Title + Aims & Scope (TS), Abstract + Aims & Scope (AS), Keywords + Aims & Scope (KS), Title + Abstract + Aims & Scope (TAS), Title + Keywords + Aims & Scope (TKS), Abstract + Keywords + Aims & Scope (AKS), Title + Abstract + Keywords + Aims & Scope (TAKS).

Mô hình bao gồm hai nhánh phụ và một nhánh chính. Ở nhánh phụ thứ nhất, chúng tôi sử dụng pre-trained model RoBERTa để mã hóa dữ liệu đầu vào từ tạp chí thành các embedding vector, chúng tôi cũng bổ sung một linear layer với ReLU làm hàm kích hoạt để giảm chiều dữ liệu. Ở nhánh phụ thứ hai, dữ liệu đầu vào từ bài báo được mã hóa tương tự như mô hình sử dụng thông tin từ bài báo đã được đề cập ở trên. Bước tiếp theo, chúng tôi trích xuất độ tương đồng giữa đầu ra của hai nhánh phụ bằng các sử dụng độ tương đồng cosine, sau đó ghép chúng lại với nhau tạo thành các embedding vector mới thông qua độ tương đồng này. Ở nhánh chính, chúng tôi sử dụng một linear layer với softmax làm hàm kích hoạt nhằm tính xác suất để bài báo thuộc về tạp chí khoa học tương ứng và sắp xếp chúng theo thứ tự giảm dần, mục đích là để đưa ra được danh sách top các item được gợi ý.



Hình 3.5: Kiến trúc mô hình sử dụng thông tin từ bài báo và tạp chí khoa học.

3.3 Độ đo đánh giá

Để đánh hiệu năng của mô hình được đề xuất, chúng tôi sử dụng độ đo Top K - Accuracy, trong đó chúng tối lựa chọn K = 1, 3, 5, 10. Độ đo Top K - Accuracy là tỷ số giữa số lượng các vật phẩm được dự đoán đúng trong mỗi cụm K và số lượng các K cụm vật phẩm

$$P_{Top-K} = \frac{\text{Số lượng các vật phẩm liên quan}}{\text{Số lượng các vật phẩm hiển thị}}$$

Chương 4

Thí nghiệm

Trong chương này, chúng tôi sẽ trình bày các thí nghiệm cho phương pháp đề xuất và so sánh kết quả của nó với phương pháp trước đó.

4.1 Chuẩn bị thí nghiệm

Các thí nghiệm của chúng tôi được thực hiện trên nền tảng Google Colab Pro, sử dụng Tesla P100-PCIE 16GB VRAM GPU, và triển khai bằng framework PyTorch. Ngoài ra, chúng tôi còn sử dụng thư viện HuggingFace, là một trong những framework NLP phổ biến nhất cung cấp các API cho phép tải và sử dụng các pre-trained model một cách dễ dàng.

4.2 Tập dữ liệu

Tập dữ liệu bao gồm 447659 bài báo và 351 tạp chí khoa học, trong đó 331464 bài báo được sử dụng cho quá trình huấn luyện, 33147 bài báo được sử dụng để đánh giá độ chính xác của mô hình trong quá trình huấn luyện và 83048 bài báo được sử dụng cho quá trình kiểm tra. Tỷ lệ giữa tập huấn luyện và tập kiểm tra là 80:20. Các đặc trưng được sử dụng bao gồm Title, Abstract, Keywords, nhãn của các bài báo và Aims & Scope của các tạp chí khoa học. Tất cả bài báo khoa học đều thuộc nhà xuất bản Springer.

Bên cạnh kích thước của tập dữ liệu, chất lượng của tập dữ liệu cũng ảnh hưởng rất nhiều đến hiệu suất của mô hình. Do đó, tiền xử lý dữ liệu là một bước quan trọng trong hầu hết các tác vụ học máy, quy trình này càng quan trọng hơn trong các vấn đề NLP. Quá trình tiền xử lý dữ liệu bao gồm các bước như sau: (1) chuyển sang chữ thường, (2) loại bỏ dấu câu, (3) loại bỏ URLs, (4) loại bỏ stop words từ Natural Language Toolkit (NLTK6), (5) loại bỏ khoảng trống không cần thiết, (6) loại bỏ văn bản không phải chữ cái. Chúng tôi áp dụng quá trình tiền xử lý dữ liệu cho hai loại dữ liệu, bao gồm Title, Abstract, Keywords của bài báo và Aims & Scope của tạp chí khoa học.

Đặc trưng	Trước khi tiền xử lý	Sau khi tiền xử lý
Title	Effects of environmental conditions and aboveground biomass on CO	effects environmental conditions aboveground biomass
Abstract	Estuarial saline wetlands have been recognized as a vital role in CO 2 cycling. However, insufficient attention has been paid to estimating CO 2 fluxes from estuarial saline wetlands. In this study, the static chamber-gas chromatography (GC) method was used to quantify CO 2 budget of an estuarial saline ...	estuarial saline wetlands recognized vital role cycling insufficient attention paid estimating fluxes estuarial saline wetlands study static chamber gas chromatography gc method used quantify budget estuarial saline ...
Keywords	net ecosystem CO, 2, exchange, ecosystem respiration, gross primary production, influencing factor, estuarial saline reed wetland, static chamber-GC method	net ecosystem exchange ecosystem respiration gross primary production influencing factor estuarial saline reed wetland static chamber gc method
Aims & Scope	Chinese Geographical Science is an international journal, sponsored by Northeast Institute of Geography and Agroecology, Chinese Academy of Sciences, and published by Science Press, Beijing, China. Chinese Geographical Science is devoted to leading scientific and technological innovation ...	chinese geographical science international journal sponsored northeast institute geography agroecology chinese academy sciences published science press beijing china chinese geographical science devoted leading scientific technological innovation ...

Bảng 4.1: Các đặc trưng trước và sau khi tiền xử lý dữ liệu.

4.3 Chi tiết quá trình huấn luyện

Đầu tiên, chúng tôi sử dụng pre-trained model RoBERTa và bắt đầu thực hiện quá trình fine-tuning với tập hợp các cặp dữ liệu được tạo thành từ việc ghép Title, Abstract, và Keyword của bài báo với Aims & Scope của tạp chí khoa học tương ứng. Tokenizer của pre-trained model có nhiệm vụ mã hóa mỗi thành phần trong các đặc trưng trên thành biểu diễn đầu vào. Để trích xuất được sentence representation một cách hiệu quả, thay vì chọn token [CLS] của layer cuối cùng, chúng tôi thêm WLAP vào mỗi layer như đã trình bày ở mục 3.1.3, và tính toán để xác định được layer hiệu quả nhất. Bên cạnh đó, chúng tôi cũng sử dụng mô hình QLoRA được trình bày ở mục 3.1.2 để giảm dung lượng bộ nhớ trong quá trình fine-tuning.

Tiếp theo, chúng tôi sử dụng hàm mục tiêu của học tương phản để điều chỉnh các tham số của pre-trained model, nhằm mục đích nâng cao khả năng trích xuất mối quan hệ ngữ nghĩa giữa các câu. Trong quá trình thí nghiệm, chúng tôi nhận thấy rằng phương pháp tối ưu AdamW [15] (một biến thể của phương pháp Adam [16]), sử dụng kỹ thuật phân rã trọng số để tránh overfitting, là lựa chọn hiệu quả để tối ưu mô hình.

Cuối cùng, chúng tôi huấn luyện pre-trained model RoBERTa đã được fine-tuning trên các tổ hợp đặc trưng khác nhau và sử dụng AdamW để tối ưu hóa hàm cross-entropy. Chúng tôi đặt lớp softmax để đạt các giá trị Top K-Accuracy, tương ứng với xác suất của các đầu vào thuộc về các nhãn tương ứng và tính toán kết quả tại mỗi giá trị K như mô tả trong mục 3.3.

Để có được kết quả mong muốn, chúng tôi thực hiện tổng cộng 10 vòng lặp cho quá trình huấn luyện; kích thước mini-batch của tập huấn luyện là 16, của tập đánh giá và tập kiểm tra là 8; đối với hàm mục tiêu của học tương phản, chúng tôi chọn tham số τ là 0.1; đối với hàm tối ưu AdamW, chúng tôi chọn tham số lr là 5×10^{-5} và tham số gamma là 0.96; số phân lớp được sử dụng trong bài toán phân loại là 2, trong đó giá trị 1 tương ứng với việc mô hình dự đoán chính xác tạp chí khoa học của bài báo và ngược lại là 0.

4.4 Kết quả

Thí nghiệm của chúng tôi được thực hiện dựa trên hai mô hình đã được định nghĩa trước đó: mô hình sử dụng thông tin từ bài báo, và mô hình sử dụng thông tin từ bài báo và tạp chí khoa học. Tuy nhiên, do chi phí có giới hạn, chúng tôi chỉ thực hiện thí nghiệm đối với hai tổ hợp đặc trưng TAK và TAKS. Đây cũng là hai tổ hợp đặc trưng cho kết quả tốt hơn so với các tổ hợp còn lại trong bài báo SimCPSR. Chúng tôi gọi mô hình thí nghiệm là SimCPSR (QLoRA-LWAP) và so sánh kết quả của nó với mô hình SimCPSR.

Thứ nhất, đối với trường hợp sử dụng thông tin từ bài báo, mô hình SimCPSR (QLoRA-LWAP) đạt kết quả lần lượt là 0.8154, 0.8929, 0.9553 khi K = 3, 5, 10. Kết quả này tốt hơn so với 0.8097, 0.8862, 0.9496 của mô hình SimCPSR. Tuy nhiên, khi K = 1, mô hình SimCPSR đạt kết quả 0.5173 tốt hơn so với 0.5116 của

Phương pháp	Tổ hợp đặc trưng	Top 1	Top 3	Top 5	Top 10
SimCPSR	TAK	**0.5173**	0.8097	0.8862	0.9496
	TAKS	**0.5194**	0.8112	0.8866	0.9496
SimCPSR (QLoRA-LWAP)	TAK	0.5116	**0.8154**	**0.8929**	**0.9553**
	TAKS	0.5143	**0.8161**	**0.8926**	**0.9558**

Bảng 4.2: So sánh kết quả giữa mô hình SimCPSR (QLoRA-LWAP) và SimCPSR.

SimPSCR (QLoRA-LWAP). Điều này cũng tương tự với trường hợp sử dụng thông tin từ bài báo và tạp chí khoa học.

Thứ hai, cả hai mô hình SimCPSR (QLoRA-LWAP) và SimCPSR đều cải thiện hiệu suất khi sử dụng thông tin từ tạp chí khoa học kết hợp với thông tin từ bài báo. Cụ thể, mô hình SimCPSR sử dụng TAKS đạt kết quả tốt hơn TAK khi K = 1, 3, 5; mô hình SimCPSR (QLoRA-LWAP) sử dụng TAKS đạt kết quả tốt hơn TAK khi K = 1, 5, 10 . Điều này cho thấy mô hình sử dụng thông tin từ bài báo và tạp chí khoa học hiệu quả hơn mô hình chỉ sử dụng thông tin từ bài báo.

Cuối cùng, với kết quả 0.8161, 0.8926, 0.9558 khi K = 3, 5, 10, mô hình SimCPSR (QLoRA-LWAP) đã thu được kết quả tốt nhất đối với bài toán hệ thống gợi ý nộp bài báo so với các phương pháp trước đây.

Tổng kết, mô hình đề xuất của chúng tôi, SimCPSR (QLoRA-LWAP), đã đạt một số cải thiện so với mô hình SimCPSR. Việc sử dụng QLoRa kết hợp với một pre-trained LLM giúp quá trình fine-tuning với hàng trăm triệu tham số chỉ cần dung lượng bộ nhớ chưa tới vài chục gigabyte. Cụ thể, mô hình RoBERTa có số lượng tham số phải học là 125985024; trong khi mô hình RoBERTa kết hợp QLoRA có số lượng tham số phải học là 1339392, tương đương 1.0631% số lượng tham số ban đầu. Mặt khác, việc thêm LWAP vào mỗi layer trong pre-trained model nhằm tính toán layer có biểu diễn câu tốt nhất, mô hình SimCPSR (QLoRA-LWAP) đã thu được kết quả cạnh tranh hơn so với SimCPSR. Ngoài ra, mô hình của chúng tôi cũng cho thấy phương pháp tiếp cận sử dụng thông tin từ bài báo và tạp chí khoa học là một hướng đi hiệu quả và tiềm năng đối với bài toán hệ thống gợi ý nộp bài báo.

Chương 5

Kết luận

Chúng tôi đã trình bày một phương pháp tiếp cận dựa trên kiến trúc transformer kết hợp với học tương phản cho bài toán hệ thống gợi ý nộp bài báo. Trong đó, chúng tôi đề xuất hai hướng tiếp cận mới: Sử dụng mô hình QLoRA để giảm dung lượng bộ nhớ trong quá trình huấn luyện; sử dụng phương pháp LWAP cho mỗi layer trong pre-trained model để tăng khả năng trích xuất độ tương đồng ngữ nghĩa. Các kết quả thực nghiệm cho thấy các đề xuất của chúng tôi đã đem lại kết quả tương đối cạnh tranh. Tuy nhiên, do nguồn kinh phí có hạn, chúng tôi chỉ thực hiện thí nghiệm trên hai tổ hợp đặc trưng TAK, TAKS và chỉ sử dụng một pre-trained model cho quá trình fine-tuning. Ngoài ra, mô hình của chúng tôi hiện tại chỉ được huấn luyện từ tập dữ liệu thuộc nhà xuất bản Springer. Do đó, chúng tôi dự định sẽ tiếp tục thực hiện các nghiên cứu trong tương lai như sau:

- Chúng tôi sẽ thí nghiệm trên toàn bộ các tổ hợp đặc trưng từ bài báo và tạp chí khoa học để có được kết quả tổng quan hơn.
- Chúng tôi sẽ thử nghiệm các pre-trained model khác như Distil-RoBERTa, ChatGPT để nâng cao hiệu suất của mô hình.
- Chúng tôi sẽ cải tiến mô hình để nó có thể áp dụng đối với tập dữ liệu từ nhiều nhà xuất bản khác như Elsevier, Nature, Wiley.

Tài liệu tham khảo

[1] Wang, D., Liang, Y., Xu, D., Feng, X., & Guan, R. (2018). A content-based recommender system for computer science publications. Knowledge-based systems, 157, 1-9.

[2] Nguyen, D., Huynh, S., Huynh, P., Dinh, C. V., & Nguyen, B. T. (2021). S2CFT: A new approach for paper submission recommendation. In SOFSEM 2021: Theory and Practice of Computer Science: 47th International Conference on Current Trends in Theory and Practice of Computer Science, SOFSEM 2021, Bolzano-Bozen, Italy, January 25–29, 2021, Proceedings 47 (pp. 563-573). Springer International Publishing.

[3] Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).

[4] Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. Transactions of the association for computational linguistics, 5, 135-146.

[5] Huynh, S. T., Dang, N., Huynh, P. T., Nguyen, D. H., & Nguyen, B. T. (2021). A fusion approach for paper submission recommendation system. In Advances and Trends in Artificial Intelligence. From Theory to Practice: 34th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2021, Kuala Lumpur, Malaysia, July 26–29, 2021, Proceedings, Part II 34 (pp. 72-83). Springer International Publishing.

[6] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

[7] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[8] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

[9] Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., ... & Gelly, S. (2019, May). Parameter-efficient transfer learning for NLP. In International conference on machine learning (pp. 2790-2799). PMLR.

[10] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., ... & Chen, W. (2021). Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685.

[11] Gao, T., Yao, X., & Chen, D. (2021). Simcse: Simple contrastive learning of sentence embeddings. arXiv preprint arXiv:2104.08821.

[12] Oh, D., Kim, Y., Lee, H., Huang, H. H., & Lim, H. (2022). Don't Judge a Language Model by Its Last Layer: Contrastive Learning with Layer-Wise Attention Pooling. arXiv preprint arXiv:2209.05972.

[13] Dettmers, T., Pagnoni, A., Holtzman, A., & Zettlemoyer, L. (2024). Qlora: Efficient finetuning of quantized llms. Advances in Neural Information Processing Systems, 36.

[14] Le, D. H., Doan, T. T., Huynh, S. T., & Nguyen, B. T. (2022, November). SimCPSR: Simple Contrastive Learning for Paper Submission Recommendation System. In Asian Conference on Intelligent Information and Database Systems (pp. 51-63). Cham: Springer International Publishing.

[15] Loshchilov, I., & Hutter, F. (2017). Fixing weight decay regularization in adam. arXiv preprint arXiv:1711.05101, 5.

[16] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.