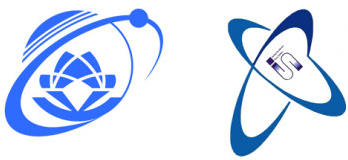
**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**--------------ÐÑ¶ÐÑ-------------**

****

**BÁO CÁO** **ĐỒ ÁN MÔN HỌC**

**KHAI THÁC DỮ LIỆU**

**ĐỀ TÀI**

**Ứng dụng của PhoBERT-base trong bài toán phân loại dành cho Tiếng Việt**

Sinh viên thực hiện**: Nhóm 3**

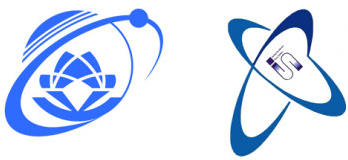
1. **Nguyễn Thành Phát MSSV: 20520270**
2. **Nguyễn Anh Kiệt MSSV: 20521498**
3. **Trần Thị Mỹ Nhung MSSV: 205202****67**
4. **Nguyễn Hoài Linh MSSV: 20521534**

**TP. HỒ CHÍ MINH 06/2023**

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**--------------ÐÑ¶Ð-------------**

****

**BÁO CÁO ĐỒ ÁN MÔN HỌC**

**KHAI THÁC DỮ LIỆU**

**ĐỀ TÀI**

**Ứng dụng của PhoBERT-base trong bài toán phân loại dành cho Tiếng Việt**

Giảng viên hướng dẫn**: Ths. Dương Phi Long**

Sinh viên thực hiện**: Nhóm 3**

1. **Nguyễn Thành Phát MSSV: 20520270**
2. **Nguyễn Anh Kiệt MSSV: 20521498**
3. **Trần Thị Mỹ Nhung MSSV: 20520267**
4. **Nguyễn Hoài Linh MSSV: 20521534**

**TP. HỒ CHÍ MINH 06/2023**

# MỤC LỤC

[MỤC LỤC 1](#_Toc137577671)

[LỜI CẢM ƠN 4](#_Toc137577672)

[LỜI NHẬN XÉT CỦA GIẢNG VIÊN 5](#_Toc137577673)

[DANH MỤC HÌNH ẢNH 6](#_Toc137577674)

[DANH MỤC BẢNG 7](#_Toc137577675)

[DANH MỤC TỪ VIẾT TẮT VÀ THUẬT NGỮ 8](#_Toc137577676)

[MỞ ĐẦU 10](#_Toc137577677)

[Chương I. TỔNG QUAN ĐỀ TÀI 11](#_Toc137577678)

[1.1. Giới thiệu bài báo 11](#_Toc137577679)

[1.1.1. Tên bài báo, tác giả, Năm XB 11](#_Toc137577680)

[1.1.2. Abstract 12](#_Toc137577681)

[1.2. Bài toán cần giải quyết 12](#_Toc137577682)

[1.3. Hướng phát triển bài toán 13](#_Toc137577683)

[Chương II. CÁC NGHIÊN CỨU LIÊN QUAN 14](#_Toc137577684)

[2.1. Hướng xử lý bài toán 14](#_Toc137577685)

[2.2. Các nghiên cứu và phương pháp liên quan 15](#_Toc137577686)

[2.2.1. Lựa chọn dữ liệu trong huấn luyện mô hình 15](#_Toc137577687)

[2.2.2. Mô hình tiền huấn luyện – BERT 16](#_Toc137577688)

[2.2.3. Mô hình XLM-R 16](#_Toc137577689)

[2.3. Cải tiến của PhoBERT 17](#_Toc137577690)

[Chương III. PHƯƠNG PHÁP 18](#_Toc137577691)

[3.1. Cơ sở lý thuyết 18](#_Toc137577692)

[3.1.1. Pre-trained 18](#_Toc137577693)

[3.1.2. Transformer 19](#_Toc137577694)

[3.1.3. NLP Task 21](#_Toc137577695)

[3.1.4. Fine-tuning 22](#_Toc137577696)

[3.1.5. BPE 23](#_Toc137577697)

[3.2. Công cụ thực hiện 24](#_Toc137577698)

[3.2.1. fairseq 24](#_Toc137577699)

[3.2.3. Transformer 25](#_Toc137577700)

[3.2.3. fastBPE 26](#_Toc137577701)

[3.2.4. RDRSegmenter 26](#_Toc137577702)

[3.2.5. VnCoreNLP 27](#_Toc137577703)

[3.3. Kiến trúc 28](#_Toc137577704)

[3.3.1. BERT 28](#_Toc137577705)

[Giới thiệu 28](#_Toc137577706)

[Vai trò của BERT 28](#_Toc137577707)

[Đổi mới của BERT 30](#_Toc137577708)

[3.3.2. RoBERTa 31](#_Toc137577709)

[Giới thiệu 31](#_Toc137577710)

[Kỹ thuật cải tiến 32](#_Toc137577711)

[Ứng dụng 33](#_Toc137577712)

[Chương IV. GIỚI THIỆU DATASET 34](#_Toc137577713)

[4.1. Mô tả về tập dữ liệu 34](#_Toc137577714)

[4.1.1. VNTC 34](#_Toc137577715)

[4.1.2. Emotion 34](#_Toc137577716)

[4.2. Tập dữ liệu sẽ sử dụng 34](#_Toc137577717)

[4.2.1. VNTC 10Topics 34](#_Toc137577718)

[4.2.1.1. Tập Train 34](#_Toc137577719)

[4.2.1.2. Tập Test 35](#_Toc137577720)

[4.2.2. Emotion 36](#_Toc137577721)

[4.2.2.1. Tập Train 36](#_Toc137577722)

[4.2.2.2. Tập Test 37](#_Toc137577723)

[4.2.2.3. Tập Validate 38](#_Toc137577724)

[Chương V. THỰC NGHIỆM 39](#_Toc137577725)

[5.1. Giới thiệu độ đo 40](#_Toc137577726)

[5.1.1. Confusion matrix 40](#_Toc137577727)

[5.1.2. Accuracy (Độ chính xác) 40](#_Toc137577728)

[5.1.3. F1-score 40](#_Toc137577729)

[5.2. Thực nghiệm trên các bài toán 41](#_Toc137577730)

[5.2.1. VNTC 41](#_Toc137577731)

[5.2.2. Emotion 42](#_Toc137577732)

[5.3. Đánh giá kết quả thực nghiệm 44](#_Toc137577733)

[Chương VI. KẾT LUẬN 44](#_Toc137577734)

[6.1. Kết quả đạt được 44](#_Toc137577735)

[6.2. Hạn chế 45](#_Toc137577736)

[6.3. Hướng phát triển 45](#_Toc137577737)

[PHÂN CÔNG CÔNG VIỆC 46](#_Toc137577738)

[TÀI LIỆU THAM KHẢO 47](#_Toc137577739)

# LỜI CẢM ƠN

Chúng em xin gửi lời cám ơn chân thành đến thầy Dương Phi Long đã tận tình hướng dẫn, giải đáp các thắc mắc của nhóm chúng em trong suốt quá trình thực hiện đồ án môn học. Cám ơn thầy vì những kinh nghiệm quý báu mà thầy đã truyền đạt cho chúng em về các phương pháp làm việc, nghiên cứu hiệu quả, cũng như các kinh nghiệm để nhóm chúng em có thể áp dụng vào trong suốt quá trình làm việc nhóm. Thầy đã luôn quan tâm đến quá trình thực hiện đồ án của cả lớp và luôn nhắc nhở, quan tâm đến tiến độ thực hiện đồ án, điều đó tạo động lực cho nhóm chúng em rất nhiều để có thể hoàn thành đồ án một cách tốt nhất.

Ngoài ra không thể không kể đến sự giúp sức và hỗ trợ của các bạn đồng trang lứa, đặc biệt là những người bạn thân thiết, vì đã luôn đồng hành cũng như đóng góp ý kiến, thời gian để hỗ trợ nhóm trong lần thực hiện đồ án này.

Dù đã cố gắng và nỗ lực hoàn thành đồ án một cách tốt nhất có thể, nhưng nhóm chúng em cũng không tránh khỏi những thiếu sót, rất mong thầy và các bạn thông cảm. Chúng em luôn biết ơn sự góp ý và chỉ bảo từ thầy và các bạn để nhóm chúng em có thể cải thiện bản thân và thực hiện tốt hơn trong tương lai.

# LỜI NHẬN XÉT CỦA GIẢNG VIÊN

# DANH MỤC HÌNH ẢNH

[Hình 1.1: Hình ảnh về bài báo 11](#_Toc137318981)

[Hình 3.1: Kiến trúc của Transformer 20](#_Toc137318907)

[Hình 3.2: Mô hình Pre-training và Fine-Tuning cho BERT 29](#_Toc137318908)

[Hình 3.3: Ví dụ cho NSP 31](#_Toc137318909)

[Hình 3.4: Giới thiệu RoBERTa 31](#_Toc137318910)

[Hình 4.1: Số lượng mẫu tin của mỗi thể loại trong tập Train\_Full 34](#_Toc137318601)

[Hình 4.2: Số lượng mẩu tin của mỗi thể loại trong tập Test\_Full 35](#_Toc137318602)

[Hình 4.3: Biểu đồ trực quan hóa dữ liệu tập train UIT - VSMEC 36](#_Toc137318603)

[Hình 4.4: Biểu đồ trực quan hóa dữ liệu tập test UIT - VSMEC 37](#_Toc137318604)

[Hình 4.5: Biểu đồ trực quan hóa dữ liệu tập validate UIT - VSMEC 38](#_Toc137318605)

# DANH MỤC BẢNG

[Bảng I: Danh mục từ viết tắt 8](#_Toc137318657)

[Bảng II: Giải thích thuật ngữ 9](#_Toc137318658)

[Bảng 3.1: Điểm khác biệt của Transformer với các mô hình trước 21](#_Toc137318667)

[Bảng 5.1: Ma trận nhầm lẫn 39](#_Toc137318675)

[Bảng 6.1: Bảng phân công công việc 45](#_Toc137318684)

# DANH MỤC TỪ VIẾT TẮT VÀ THUẬT NGỮ

**DANH MỤC TỪ VIẾT TẮT**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Từ viết tắt** | **Tên đầy đủ** | **Giải thích** |
| 1 | NLP | Natural Language Processing | Xử lý ngôn ngữ tự nhiên |
| 2 | XLM-R | Cross-Lingual Language Model | Mô hình tự giám sát |
| 3 | BERT | Bidirectional Encoder Representations from Transformers | Mô hình học sẵn học ra các vector đại diện |
| 4 | BPE | Byte Pair Encoding | Kỹ thuật nén dữ liệu |

Bảng I: Danh mục từ viết tắt

**GIẢI THÍCH THUẬT NGỮ**

|  |  |  |
| --- | --- | --- |
| **STT** | **Thuật ngữ** | **Giải thích** |
| 1 | PhoBERT | Phở BERT |
| 2 | Transformer | Một kiến trúc mô hình máy học sử dụng mạng nơ-ron hồi quy (RNN) dựa trên Attention |
| 3 | Confusion matrix | Công cụ phân tích kết quả phân loại trong học máy |
| 4 | Fairseq | Thư viện mã nguồn mở cung cấp các công cụ và mô hình tiên tiến cho dịch máy |
| 5 | FastBPE | Công cụ mã nguồn mở để thực hiện mã hóa và giải mã các chuỗi văn bản sử dụng phương pháp mã hóa dựa trên ngữ cảnh |
| 6 | RDRSegmenter | Công cụ phân đoạn từ tiếng Việt dựa trên quy tắc |
| 7 | VnCoreNLP | Thư viện xử lý ngôn ngữ tự nhiên cho tiếng Việt. VnCoreNLP cung cấp các chức năng như phân đoạn câu, gán nhãn từ loại, … |

Bảng II: Giải thích thuật ngữ

# MỞ ĐẦU

Ngày nay, việc xây dựng, phát triển các ứng dụng và webiste để áp dụng vào đời sống xã hội đang được nhiều doanh nghiệp, tổ chức tập trung thực hiện. Với mục đích tương tác trực tiếp với người dùng, nhiều ứng dụng tập trung vào việc nhận dạng ngôn ngữ của người dùng để xử lý, đưa ra các giải pháp cũng như hỗ trợ người dùng trong quá trình sử dụng. Nổi bật nhất trong việc tương tác giữa máy tính và người dùng là NLP\_một dạng AI giúp xử lý ngôn ngữ tự nhiên của người dùng và tạo sự tương tác tốt giữa ngôn ngữ tự nhiên với ngôn ngữ máy tính.

Các ngôn ngữ loài người không chỉ đơn giản là các kí tự chữ Latinh thông dụng, mà còn phức tạp đối với từng quốc gia, từng loại ngôn ngữ khác nhau. Nhiều quốc gia sử dụng bảng chữ cái tượng hình riêng như tiếng Nhật, tiếng Hàn, tiếng Trung, tiếng Thái,… làm cho nhiều người ngoại quốc không thể đọc hiểu được ngôn ngữ của quốc gia đó. Đối với tiếng Việt, tuy được xây dựng trên nền tảng là chữ Latinh nhưng còn kèm theo các thanh điệu (thanh sắc, huyền, hỏi, ngã, nặng, thanh ngang). Kèm với đó, mỗi vùng miền có thể sẽ hiểu một từ theo các ý nghĩa khác nhau, tạo nên sự phong phú về nội dung và ý nghĩa của tiếng Việt. Thế nhưng, sự phong phú này đồng thời cũng đặt ra một thử thách cho các nhiệm vụ phân tích và xử lý ngôn ngữ tự nhiên.

Nhờ phân tích được những thử thách đặt ra, nhóm đồ án đặt mục tiêu tìm hiểu và áp dụng những nghiên cứu về xử lý ngôn ngữ tự nhiên nổi bật nhất hiện nay, đặc biệt là mô hình xử lý ngôn ngữ tiếng Việt PhoBERT để ứng dụng vào việc phân tích, dự đoán và đưa ra các gợi ý cho các ứng dụng có tích hợp ngôn ngữ tiếng Việt.

Vì những lí do trên, nhóm lựa chọn đề tài ***“Ứng dụng của PhoBER-base trong bài toán phân loại dành cho Tiếng Việt”*** với ý tưởng từ bài báo PhoBERT: Pre-trained language models for Vietnamese làm đề tài thực hiện cho môn Khai thác dữ liệu.

# TỔNG QUAN ĐỀ TÀI

Nội dung Chương I sẽ trình bày các thông tin tổng quan về bài báo mà nhóm tham khảo, tìm hiểu cũng như giới thiệu về bài toán mà nhóm đặt ra. Bên cạnh đó, nhóm đồ án cũng phân tích những gì đã ứng dụng được cũng như những hạn chế và hướng phát triển của bài toán.

## 1.1. Giới thiệu bài báo

### 1.1.1. Tên bài báo, tác giả, Năm XB

Đồ án tập trung tìm hiểu và nghiên cứu bài báo *“PhoBERT: Pre-trained language models for Vietnamese”* được công bố bởi hai tác giả người Việt Nam là Dat Quoc Nguyen và Anh Tuan Nguyen.

Bài báo được phát hành vào năm 2020 tại hội nghị IJCAI (International Joint Conference on Artificial Intelligence). Đây là một hội nghị hàng đầu trong lĩnh vực trí tuệ nhân tạo và có quy mô lớn cùng với tính chọn lọc cao.

A picture containing text, font, screenshot, paper

Description automatically generated

Hình 1.1: Hình ảnh về bài báo

### 1.1.2. Giới thiệu bài báo

Nhóm tác giả đã công bố PhoBERT với hai phiên bản là PhoBERTbase và PhoBERTlarge là mô hình đơn ngữ có quy mô lớn đầu tiên được huấn luyện cho tiếng Việt. Kết quả thử nghiệm cho thấy PhoBERT luôn vượt trội hơn so với mô hình đa ngôn ngữ tiền huấn luyện tốt nhất hiện tại XLM-R (Conneau et al., 2020) và cải thiện state-of-the-art trong nhiều tác vụ xử lý ngôn ngữ tự nhiên dành riêng cho tiếng Việt, bao gồm gán nhãn từ loại; phân tích sự phụ thuộc cú pháp; xác định và phân loại các thực thể tên riêng; và đánh giá mối quan hệ logic và ngữ nghĩa giữa hai câu. Nhóm tác giả phát hành PhoBERT để tạo điều kiện cho việc nghiên cứu và ứng dụng tiếp theo trong lĩnh vực xử lý ngôn ngữ tự nhiên cho tiếng Việt. Các mô hình PhoBERT của nhóm tác giả được công bố tại: <https://github.com/VinAIResearch/PhoBERT>.

## 1.2. Bài toán cần giải quyết

PhoBERTbase là một mô hình ngôn ngữ dựa trên kiến trúc BERT và được huấn luyện dựa trên tập dữ liệu tiếng Việt rộng lớn đên từ nhiều nguồn khác nhau như báo chí, trang web, diễn đàn, …. PhoBERTbase có kiến trúc bao gồm 12 lớp mã hóa và khác biệt so với PhoBERTlarge là có tới 24 lớp mã hóa. Khi thực hiện huấn luyện mô hình PhoBERTbase, nhóm tác giả đã thiết lập batch size là 1024 và tốc độ học là 0.0004.

PhoBERTbase được ứng dụng trong các bài toán như phân loại văn bản (phân loại tin tức, phân loại cảm xúc, phân loại theo thể loại văn bản,…), nhận diện các thực thể trong văn bản (tên riêng, tên đường, địa điểm, ngày tháng,…), dịch máy, tóm tắt văn bản,…

Khi thực hiện lại mô hình này, nhóm đồ án đã ứng dụng PhoBERTbase vào bài toán gán nhãn đơn nhãn. Trong đó, có hai bài toán chính mà nhóm đã thực hiện thành công là phân loại văn bản theo chủ đề và phân loại cảm xúc của văn bản.

Phân loại văn bản đơn nhãn có thể được hiểu là việc gán một nhãn duy nhất cho một mẫu dữ liệu đầu vào. Trong đó, mô hình sẽ học các đặc trưng của từng nhãn được cung cấp và phân tích thông tin đầu vào theo các tác vụ NLP để có thể suy đoán ra nhãn phù hợp cho thông tin đầu vào đó.

Với bài toán phân loại văn bản, mô hình sẽ được huấn luyện từ tập dữ liệu bao gồm các chủ đề của văn bản như đời sống, chính trị xã hội, thể thao, văn hóa, kinh doanh, khoa học,… Sau quá trình học, mô hình có thể dự đoán được nhãn chủ đề của một đoạn văn bản đầu vào. Với mỗi đoạn văn bản đầu vào bất kì, mô hình sẽ cho ra kết quả là một nhãn chủ đề duy nhất và thuộc vào các chủ đề mà mô hình đã được huấn luyện. Việc giải quyết được bài toán này sẽ giúp ích rất nhiều trong việc ứng dụng và phát triển các tính năng như tóm tắt thông tin, phân loại tin tức, lọc các thư rác và dự đoán hành vi của người dùng.

Đối với bài toán phân loại cảm xúc, mô hình sẽ được học dữ liệu là các đoạn văn bản đã được gán các nhãn cảm xúc như vui, buồn, hạnh phúc, bối rối,… Nhờ đó, với dữ liệu đầu vào là một câu hoặc một đoạn văn bản ngắn, mô hình sẽ đưa ra dự đoán một sắc thái cảm xúc duy nhất của dữ liệu đầu vào đó. Bài toán phân loại cảm xúc này sẽ giúp ích rất nhiều trong việc nhận diện cảm xúc của người tiêu dùng, đánh giá mức độ hài lòng thông qua bình luận và lĩnh vực chăm sóc khách hàng.

## 1.3. Hướng phát triển bài toán

Mô hình PhoBERTbase mà nhóm đồ án đã thực hiện tuy đã đạt được những kết quả khả quan trong bài toán gán nhãn đơn nhãn (bao gồm hai bài toán nhỏ là phân loại văn bản và phân loại cảm xúc) nhưng vẫn còn tồn tại những mặt hạn chế nhất định.

Đầu tiên, do chỉ dừng lại ở bài toán gán nhãn đơn nhãn nên đối với các đoạn văn bản đầu vào mang nội dung thuộc về hai hay nhiều phân loại, nhiều trạng thái cảm xúc khác thì kết quả dự đoán lại không thể hiện được tất cả các nhãn tương ứng với thông tin đầu vào. Mô hình chỉ xét những câu, những kí tự cuối cùng của văn bản đầu vào để dự đoán nhãn duy nhất của văn bản. Điều này làm cho kết quả dự đoán không đầy đủ so với thực tế.

Thứ hai, đối với dữ liệu đầu vào thuộc vào những nhãn khác mà mô hình chưa được học, mô hình không đưa ra được dự đoán là văn bản đầu vào thuộc một nhãn mới, thay vào đó là nhãn bất kì trong những nhãn được học. Điều này làm cho kết quả dự đoán không chính xác và không phù hợp với thực tế.

Để khắc phục được các hạn chế kể trên, nhóm đồ án đề ra hướng phát triển là huấn luyện mô hình trên tập dữ liệu với đa dạng các thể loại văn bản cũng như nhiều thể loại cảm xúc hơn, để kết quả dự đoán được phong phú, trải rộng hết được các chủ đề và các tình huống trong cuộc sống hàng ngày. Kế đến là việc phát triển mô hình thành mô hình phân loại đa nhãn để mô hình đưa ra được dự đoán chính xác và đầy đủ hơn.

# CÁC NGHIÊN CỨU LIÊN QUAN

Nội dung Chương II sẽ trình bày về kiến trúc và các phương pháp để giải quyết bài toán xử lý ngôn ngữ tự nhiên cho tiếng Việt. Đồng thời, nội dung chương 2 cũng nêu lên những nghiên cứu về các phương pháp, kỹ thuật có liên quan đến đề tài và những cải tiến của PhoBERT so với các kỹ thuật được nghiên cứu và công bố trước đó.

## 2.1. Hướng xử lý bài toán

*Sử dụng các mô hình đã được tiền huấn luyện đa ngôn ngữ dựa trên BERT hiện có*

Việc xây dựng mô hình cho tiếng Việt sử dụng các mô hình đã được tiền huấn luyện đa ngôn ngữ dựa trên BERT hiện có là một phương pháp phổ biến để giải quyết vấn đề hiệu quả trong việc xử lý ngôn ngữ tự nhiên cho các ngôn ngữ khác nhau.

Các mô hình đã được tiền huấn luyện đa ngôn ngữ dựa trên BERT hiện có không được đào tạo trên dữ liệu tiếng Việt cụ thể, do đó có thể không hiệu quả như mô hình được đào tạo trực tiếp trên dữ liệu tiếng Việt.

Tuy nhiên, các nghiên cứu gần đây đã cho thấy rằng việc sử dụng các mô hình đa ngôn ngữ có thể cải thiện hiệu suất cho các tác vụ xử lý ngôn ngữ tự nhiên tiếng Việt. Ví dụ, mô hình mBERT (multilingual BERT) đã được đào tạo trên 104 ngôn ngữ, bao gồm cả tiếng Việt, và đã được sử dụng thành công trong nhiều nhiệm vụ, bao gồm phân loại văn bản và dịch máy.

Nhiều nghiên cứu đã chỉ ra rằng mBERT và các mô hình đa ngôn ngữ khác có thể cải thiện hiệu suất so với các mô hình được đào tạo riêng cho một ngôn ngữ cụ thể, đặc biệt là trong những trường hợp có ít dữ liệu đào tạo. Tuy nhiên, hiệu suất của các mô hình đa ngôn ngữ phụ thuộc vào sự giống nhau giữa các ngôn ngữ trong mô hình, do đó vẫn cần có các nghiên cứu cụ thể để đánh giá hiệu suất của các mô hình đa ngôn ngữ trong từng ngôn ngữ cụ thể.

*Đào tạo lại một mô hình dành riêng cho ngôn ngữ bằng cách sử dụng kiến trúc BERT*

Trong bài báo *TNLP: A Visual-Aided Systematic Approach to Select Pre-Trained Embeddings for a Downstream Task,* tác giả đã đào tạo lại mô hình BERT trên một tập dữ liệu tiếng Việt đa dạng, bao gồm các loại văn bản khác nhau như báo cáo kinh tế, văn bản y khoa, trang web và báo chí. Tập dữ liệu này được tác giả thu thập bằng cách sử dụng các công cụ tự động để thu thập dữ liệu từ web và các nguồn khác.

Cách thức đào tạo lại mô hình BERT trên dữ liệu tiếng Việt tương tự như cách thức đào tạo mô hình BERT trên tiếng Anh. Tác giả đã sử dụng hàm mất mát cross-entropy để đào tạo lại mô hình BERT, cùng với một số kỹ thuật như dropout và early stopping để tránh tình trạng overfitting.

Sau khi đào tạo lại mô hình BERT trên dữ liệu tiếng Việt, tác giả đã tiến hành thực nghiệm trên các nhiệm vụ xử lý ngôn ngữ tự nhiên tiếng Việt, bao gồm phân loại văn bản và dự đoán từ khóa. Kết quả thực nghiệm cho thấy mô hình BERT được đào tạo lại trên dữ liệu tiếng Việt của tác giả đã đạt được kết quả tốt trên các nhiệm vụ này.

Ngoài ra, tác giả cũng so sánh mô hình BERT được đào tạo lại trên dữ liệu tiếng Việt của họ với mô hình BERT đa ngôn ngữ được đào tạo trên nhiều ngôn ngữ khác nhau. Kết quả cho thấy mô hình BERT đào tạo lại trên dữ liệu tiếng Việt có hiệu quả hơn đáng kể so với mô hình BERT đa ngôn ngữ trong các nhiệm vụ xử lý ngôn ngữ tự nhiên tiếng Việt.

## 2.2. Các nghiên cứu và phương pháp liên quan

### 2.2.1. Lựa chọn dữ liệu trong huấn luyện mô hình

Bộ dữ liệu Wikipedia là bộ dữ liệu duy nhất được sử dụng để huấn luyện các mô hình đơn ngữ và đây cũng là bộ dữ liệu tiếng việt duy nhất được sử dụng bởi tất cả mô hình đa ngôn ngữ ngoại trừ XLM-R. Tuy nhiên dữ liệu từ Wikipedia không đại diện cho việc sử dụng ngôn ngữ thông thường và dữ liệu Wikipedia Tiếng Việt có kích thước tương đối nhỏ (1GB sau khi giải nén), trong khi các mô hình ngôn ngữ tiền huấn luyện có thể được cải thiện đáng kể bằng cách sử dụng nhiều dữ liệu tiền huấn luyện hơn.

### 2.2.2. Mô hình tiền huấn luyện – BERT

BERT đã đạt được rất nhiều thành công trong xử lý ngôn ngữ tiếng Anh. Các mô hình BERT tiền huấn luyện cho các tác vụ như phân loại văn bản, dự đoán từ tiếp theo và sinh văn bản tự động đã đạt được kết quả tốt hơn so với các phương pháp truyền thống trước đó. BERT đã trở thành một trong những mô hình tiên tiến nhất trong lĩnh vực xử lý ngôn ngữ tự nhiên và đã được sử dụng rộng rãi trong các ứng dụng như xử lý ngôn ngữ tự nhiên, dịch thuật tự động, tóm tắt văn bản và trả lời câu hỏi tự động.

Các mô hình BERT cũng đã thể hiện được khả năng xử lý tiếng Anh từ vựng phức tạp, cấu trúc câu phức hợp và sự liên kết giữa các từ. Các mô hình BERT đã học được các quan hệ ngữ nghĩa và cú pháp giữa các từ trong ngữ cảnh khác nhau, giúp cải thiện đáng kể khả năng hiểu và xử lý ngôn ngữ tự nhiên.

Tuy nhiên, nhược điểm của việc sử dụng BERT và các mô hình ngôn ngữ dựa trên BERT cho tiếng Việt là chúng không chú trọng đến sự phân biệt giữa các âm trong tiếng Việt. Điều này là do trong tiếng Việt có nhiều từ có cùng cách viết nhưng khác nhau về âm tiết và nghĩa, gọi là từ đồng âm. Ví dụ như từ “cốc” và “cốc” đều có cách phát âm giống nhau, nhưng có nghĩa khác nhau. Việc không phân biệt được những từ đồng âm này có thể dẫn đến sai sót trong việc hiểu và phân tích ngôn ngữ. Bên cạnh đó, một nhược điểm khác là không phân biệt được các cụm từ có khoảng trắng. Ví dụ như “nghiên cứu viên” sẽ không được hiểu là một cụm từ mà là từng từ riêng lẻ “nghiên”, “cứu” và “viên”.

### 2.2.3. Mô hình XLM-R

Mô hình XLM-RoBERTa (viết tắt của “Cross-lingual Language Model - RoBERTa”) là một mô hình ngôn ngữ đa ngôn ngữ được huấn luyện dựa trên kiến trúc của RoBERTa và được tinh chỉnh để có thể học được các đặc trưng ngôn ngữ chung của nhiều ngôn ngữ khác nhau. Nó được huấn luyện trên 2,5TB dữ liệu CommonCrawl được lọc. Mục tiêu của nó là cải thiện hiệu suất cho nhiều tác vụ chuyển đổi ngôn ngữ không giám sát.

XLM-RoBERTa được tạo ra nhằm giải quyết vấn đề thiếu dữ liệu trong việc huấn luyện các mô hình ngôn ngữ cho các ngôn ngữ hiếm hoặc thiếu dữ liệu.

XLM-RoBERTa có các ưu điểm như khả năng mã hóa các thông tin ngôn ngữ của nhiều ngôn ngữ khác nhau, giúp cho các nhiệm vụ xử lý ngôn ngữ đa ngôn ngữ được thực hiện hiệu quả hơn.

XML-RoBERTa là một phiên bản cải tiến của RoBERTa, một biến thể của BERT được huấn luyện trên một tập dữ liệu lớn hơn và được xử lý với phương pháp mã hóa ngôn ngữ tài nguyên mở (XML). Mục tiêu của XML-RoBERTa là nâng cao độ chính xác của RoBERTa trên các tác vụ NLP bằng cách sử dụng thông tin liên quan đến cấu trúc của tài liệu XML.

Một trong những nhược điểm của XLM-RoBERTa là nó có thể không đạt được hiệu suất tốt khi được sử dụng cho các tác vụ xử lý ngôn ngữ đặc thù của một ngôn ngữ cụ thể. Mặc dù mô hình đã được huấn luyện trên nhiều ngôn ngữ khác nhau, nhưng sự đa dạng của dữ liệu huấn luyện không đảm bảo rằng nó sẽ có khả năng phân loại và xử lý ngôn ngữ tốt trong tất cả các trường hợp.

Ngoài ra, XLM-RoBERTa cũng có một số hạn chế trong việc xử lý văn bản dài và phức tạp. Do kích thước của mô hình rất lớn, nó có thể gặp khó khăn trong việc xử lý các văn bản dài và phức tạp, và yêu cầu nhiều tài nguyên tính toán để đưa ra dự đoán chính xác.

## 2.3. Cải tiến của PhoBERT

*Dataset*

Sử dụng một bộ dữ liệu tiền huấn luyện gồm khoảng 20GB văn bản không nén. Bộ dữ liệu này là sự kết hợp của hai tập dữ liệu: tập dữ liệu Wikipedia tiếng Việt (khoảng 1GB), và tập dữ liệu thứ hai (khoảng 19GB) được tạo ra bằng cách loại bỏ các bài viết tương tự và trùng lặp từ một tập dữ liệu tin tức tiếng Việt khoảng 50GB.

*Đánh giá dựa trên bốn tác vụ xử lý ngôn ngữ tự nhiên trong Tiếng Việt*

Bốn tác vụ bao gồm:

*Part-of-speech (POS) Tagging*: Tác vụ này nhằm gán nhãn cho từng từ trong câu với nhãn phân loại từ loại, như danh từ, động từ, tính từ, trạng từ, ... POS tagging giúp hiểu và phân tích cú pháp của câu, đồng thời hỗ trợ các tác vụ xử lý ngôn ngữ khác như phân tích ngữ nghĩa và dịch máy.

*Dependency Parsing*: Tác vụ này nhằm xác định mối quan hệ phụ thuộc cú pháp giữa các từ trong câu. Dependency parsing giúp hiểu cấu trúc câu, xác định các quan hệ như quan hệ chủ - tân, quan hệ đồng nghĩa, quan hệ phụ thuộc từ vị trí và chức năng của từ.

*Named-Entity Recognition (NER)*: Tác vụ này liên quan đến việc xác định và phân loại các thực thể tên riêng trong văn bản như tên người, địa điểm, tổ chức, thời gian, ... NER giúp rút trích thông tin quan trọng từ văn bản và hỗ trợ các tác vụ như truy vấn thông tin và tổ chức dữ liệu.

*Natural Language Inference (NLI)*: Tác vụ này nhằm đánh giá mối quan hệ logic và ngữ nghĩa giữa hai câu. NLI đo lường khả năng suy luận và phân loại mối quan hệ giữa câu gốc (premise) và câu kết luận (hypothesis), ví dụ như xác định liệu câu kết luận có đúng dựa trên thông tin từ câu gốc hay không.

Kết quả thực nghiệm cho thấy các mô hình đạt được kết quả tốt nhất hiện nay trên tất cả các tác vụ này (dựa trên SOTA: State-of-the-art).

# PHƯƠNG PHÁP

Nội dung Chương III sẽ bao gồm các cơ sở lý thuyết mà nhóm sử dụng để huấn luyện mô hình. Chương này sẽ giới thiệu chi tiết về các công cụ được nhóm sử dụng và cung cấp một cái nhìn tổng quan về kiến trúc của PhoBERT, dựa trên các mô hình BERT và RoBERTa.

## 3.1. Cơ sở lý thuyết

### 3.1.1. Pre-trained

**Định nghĩa**

Pre-trained - Tiền huấn luyện là một phương pháp trong học sâu, trong đó một mô hình được huấn luyện trên một tập dữ liệu lớn để học các đặc trưng ngôn ngữ tổng quát trước khi được fine-tune trên các tác vụ cụ thể. Mô hình được huấn luyện sẽ học được các đặc trưng ngôn ngữ. Việc tiền huấn luyện này đã giúp cho các mô hình ngôn ngữ đạt được những thành tựu đáng kể trong nhiều tác vụ xử lý ngôn ngữ tự nhiên.

**Vì sao cần sử dụng Pre-trained**

Transfer learning: Pre-trained models cho phép chuyển giao học tập, tức là sử dụng kiến thức đã học từ một nhiệm vụ để cải thiện hiệu suất của mô hình trong nhiệm vụ khác. Thay vì bắt đầu huấn luyện một mô hình từ đầu, chúng ta có thể sử dụng mô hình đã được huấn luyện trước đó và chỉ cần điều chỉnh lại một số phần nhỏ để phù hợp với nhiệm vụ cụ thể.

Hiệu suất tốt: Pre-trained models thường đã được huấn luyện trên một lượng lớn dữ liệu và đi qua quá trình tối ưu hóa. Nhờ đó, chúng có khả năng phân loại, nhận dạng hoặc sinh ra thông tin chất lượng cao.

Xử lý ngôn ngữ tự nhiên: Trong lĩnh vực xử lý ngôn ngữ tự nhiên, pre-trained models như GPT-3 hoặc BERT đã được huấn luyện trên rất nhiều văn bản từ internet. Chúng có khả năng hiểu và phân tích ngôn ngữ tự nhiên phức tạp, giúp trong các tác vụ như dịch máy, tóm tắt văn bản, phân loại văn bản, và tạo ra nội dung tự động.

Tiết kiệm tài nguyên: Huấn luyện một mô hình từ đầu có thể đòi hỏi nhiều tài nguyên tính toán và thời gian kéo dài. Sử dụng pre-trained models giúp tiết kiệm tài nguyên này, vì chúng đã được huấn luyện trước và có thể được sử dụng lại trong nhiều ứng dụng khác nhau.

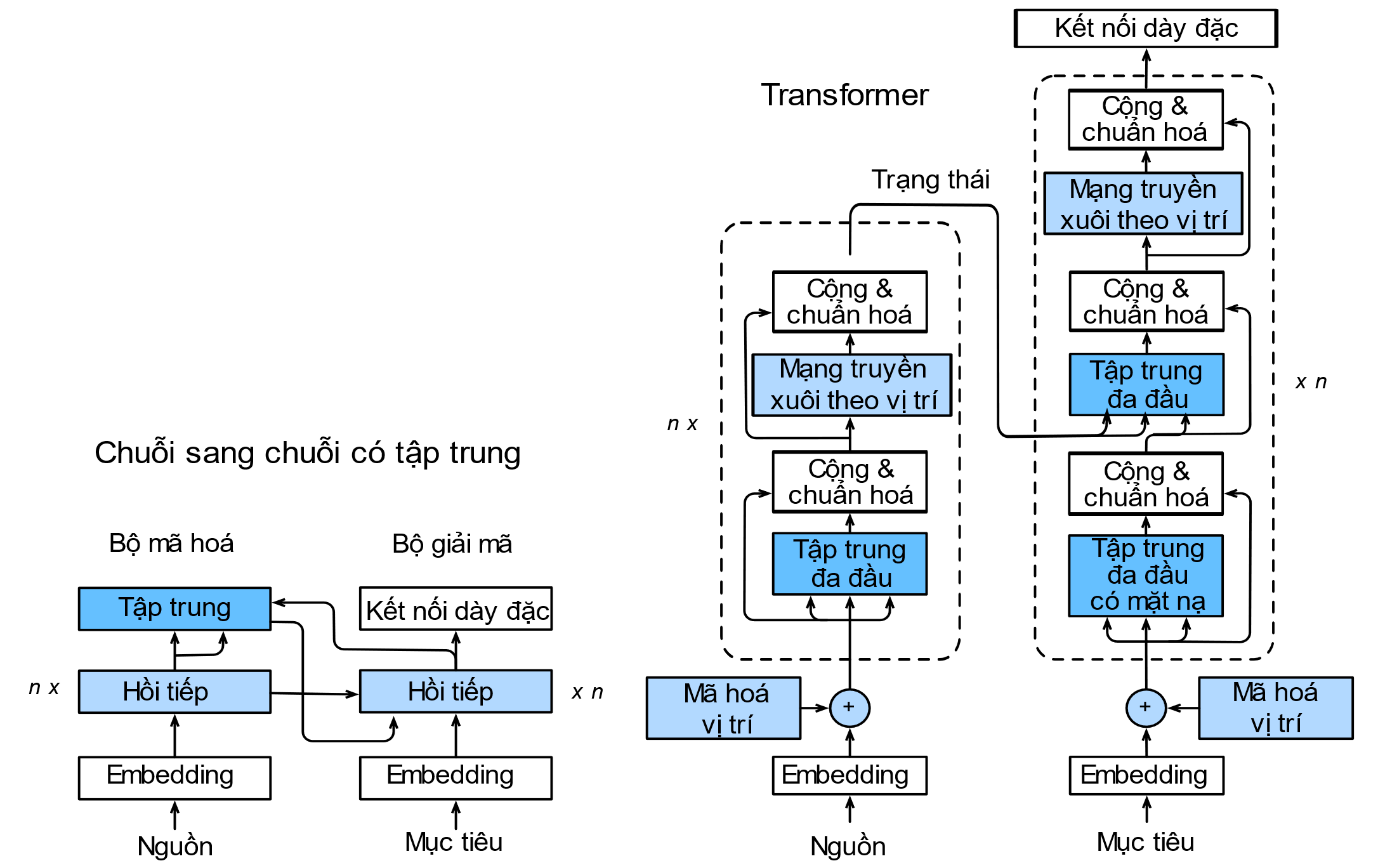
Sự đa dụng: Pre-trained models có thể được sử dụng trong nhiều lĩnh vực và nhiệm vụ khác nhau. Chúng có thể được áp dụng trong xử lý ảnh, xử lý ngôn ngữ tự nhiên, dự báo, gợi ý, và nhiều lĩnh vực khác. Điều này giúp giảm đáng kể thời gian và công sức để phát triển các ứng dụng mới.

### 3.1.2. Transformer

**Định nghĩa**

Là một mô hình học sâu được phát triển bởi các nhà nghiên cứu tại Google và Đại học Toronto vào năm 2017. Giống như các mạng thần kinh hồi quy (recurrent neural network - RNN), các Transformer được thiết kế để xử lý dữ liệu tuần tự, chẳng hạn như ngôn ngữ tự nhiên, cho các tác vụ như dịch máy thống kê hay tóm tắt tự động. Tuy nhiên, khác với RNN, các Transformer không yêu cầu dữ liệu tuần tự được xử lý theo thứ tự.

**Kiến trúc**



Hình 3.1: Kiến trúc của Transformer

Từ bên phải qua ta có:

* **Cộng và chuẩn hóa**: Đầu vào và đầu ra của cả tầng tập trung đa đầu hoặc mạng truyền xuôi theo vị trí được xử lý bởi hai tầng “cộng và chuẩn hóa” bao gồm cấu trúc phần dư và tầng chuẩn hóa theo tầng (layer normalization).
* **Khối Transformer**: Một tầng hồi tiếp trong seq2seq được thay bằng một Khối Transformer. Với bộ mã hóa, khối này chứa một tầng tập trung đa đầu và một mạng truyền xuôi theo vị trí (position-wise feed-forward network) gồm hai tầng dày đặc. Đối với bộ giải mã, khối này có thêm một tầng tập trung đa đầu khác để nhận vào trạng thái bộ mã hóa.
* **Biễu diễn vị trí**: Do tầng tự tập trung không phân biệt thứ tự phần tử trong một chuỗi, nên tầng biễu diễn vị trí được sử dụng để thêm thông tin vị trí vào từng phần tử trong chuỗi.

**Điểm khác biệt của Transformer so với các mô hình trước đó**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **seq2seq** | **seq2seq với attention** | **Transformer** |
| Encoder | RNN/CNN | RNN/CNN | attention |
| Decoder | RNN/CNN | RNN/CNN | attention |
| Tương tác giữa Encoder và Decoder | vector | attention | attention |

Bảng 3.1: Điểm khác biệt của Transformer với các mô hình trước

* Tất cả mọi hoạt động đều dựa trên cơ chế **Attention.**

### 3.1.3. NLP Task

**Định nghĩa**

NLP Task (Natural Language Processing Task) là các nhiệm vụ hoặc công việc trong lĩnh vực xử lý ngôn ngữ tự nhiên (NLP). NLP Task tập trung vào việc phân tích, hiểu và xử lý các ngôn ngữ tự nhiên được sử dụng bởi con người.

**Phân loại NLP Task**

Xử lý cú pháp và ngữ pháp

* Tokenization: Phân tách văn bản thành các từ hoặc đơn vị nhỏ hơn.
* Part-of-speech tagging: Gán nhãn từng từ trong câu với loại từ (danh từ, động từ, tính từ).
* Parsing: Phân tích cú pháp câu để hiểu cấu trúc ngữ pháp của nó.
* Dependency parsing: Xác định mối quan hệ phụ thuộc giữa các từ trong câu.

Trích xuất thông tin:

* Named entity recognition (NER): Xác định và phân loại các thực thể tên riêng (tên người, địa điểm, tổ chức, v.v.) trong văn bản.

Phân loại và phân tích văn bản:

* Text classification: Phân loại văn bản vào các nhãn hay danh mục xác định.
* Sentiment analysis: Xác định tính chất tích cực, tiêu cực hoặc trung lập của ý kiến trong văn bản.
* Topic modeling: Xác định các chủ đề chính trong một tập hợp các văn bản.
* Text summarization: Tạo bản tóm tắt ngắn gọn của văn bản dài.

**Ứng dụng**

Trợ giúp tìm kiếm thông tin: NLP Task trích xuất thông tin và phân loại văn bản có thể được sử dụng để tìm kiếm, trích xuất và tổ chức thông tin từ các nguồn dữ liệu lớn, như cơ sở dữ liệu, trang web, bài viết, tài liệu.

Dịch máy và sinh văn bản: NLP Task có thể được sử dụng để có dịch đoạn văn bản từ ngôn ngữ ban đầu sang ngôn ngữ mà ta cần. NLP Task hỗ trợ dịch máy và sinh văn bản trên đa ngôn ngữ.

Xử lý ngôn ngữ tự nhiên trong diễn giải dữ liệu: NLP Task có thể giúp phân tích, hiểu và rút trích thông tin từ các nguồn dữ liệu tự nhiên như email, bài viết, bình luận xã hội

### 3.1.4. Fine-tuning

**Định nghĩa**

Fine-tuning là một kỹ thuật trong deep learning giúp cải thiện hiệu suất của mô hình bằng cách sử dụng lại một mô hình đã được huấn luyện trước đó và tiếp tục huấn luyện trên một tập dữ liệu mới.

Kỹ thuật này được sử dụng để giải quyết các bài toán có liên quan đến các tập dữ liệu nhỏ hoặc không đủ đại diện cho tập dữ liệu thực tế.s

**Vì sao cần Fine-tunning**

Dễ dàng tùy chỉnh cho các tác vụ cụ thể: Trong một số trường hợp, mô hình học máy được huấn luyện để có thể xử lý với tác vụ là dịch văn bản. Khi áp dụng mô hình với một tác vụ cho ra kết quả vô cùng xấu vì vậy, ta cần phải tinh chỉnh lại mô hình đã được học đó để dễ dàng sử dụng hơn.

Hiệu suất tốt: Giống với Pre-trained bằng cách Fine-tunning một mô hình đã được huấn luyện sẵn, ta có thể hiệu suất của các mô hình đó với các tác vụ khác nhau mà không cần phải train lại từ một mô hình ban đầu với dữ liệu ít.

Tiết kiệm thời gian và tài nguyên: Việc sử dụng một mô hình có sẵn và Fine-tune theo tác vụ mà ta cần để thực hiện giảm thiểu tối đa thời gian để huấn luyện mô hình từ ban đầu.

### 3.1.5. BPE

**Định nghĩa**

BPE (Byte Pair Encoding) ban đầu là một thuật toán nén dữ liệu được sử dụng để tìm cách biểu diễn dữ liệu tốt nhất bằng cách xác định các cặp byte phổ biến.

Hiện nay, chúng ta sử dụng nó trong NLP để tìm cách đại diện văn bản tốt nhất bằng số lượng token nhỏ nhất bằng việc nén và mã hóa các từ ngữ thành các đơn vị nhỏ hơn, thường là các cặp byte (byte pairs).

**Cách hoạt động**

1. Thêm một định danh (</w>) vào cuối mỗi từ để xác định kết thúc một từ và sau đó tính tần số của từ đó trong văn bản.
2. Tách từ thành các ký tự và tính tần số của ký tự đó.
3. Từ các ký tự này, trong một số lần lặp được định trước, đếm tần số của các cặp byte liên tiếp và hợp nhất các cặp byte xuất hiện nhiều nhất.
4. Tiếp tục lặp lại cho đến khi bạn đạt đến giới hạn lặp (được đặt bởi bạn) hoặc khi bạn đạt đến giới hạn mã thông báo.

**Ứng dụng**

Mã hóa từ vựng: BPE được sử dụng để nén và mã hóa từ vựng trong các mô hình ngôn ngữ. Giúp giảm được kích thước của từ vựng cũng như dễ dàng giúp các mô hình học được các đặc trưng hơn.

Dịch máy: BPE được sử dụng trong các mô hình dịch máy để mã hóa và xử lý ngôn ngữ. Nó giúp mô hình hiểu được các từ ngữ không xuất hiện trong từ điển và tạo ra các đơn vị nhỏ hơn cho việc tìm kiếm tương ứng trong dịch ngữ liệu.

Tách từ: Trong các ngôn ngữ không có dấu cách giữa các từ, BPE có thể được sử dụng để tách các từ ra khỏi nhau. Điều này hữu ích trong các bước tiền xử lý và xử lý ngôn ngữ tự nhiên. Dễ dàng nhận thấy được ứng dụng này nhất là trong phân tích ngôn ngữ tự nhiên bằng tiếng việt tuy nhiên hạn chế là có một số từ dù có dấu cách nhưng gộp chúng lại mới ra 1 một cụm từ có nghĩa

Phân loại văn bản: BPE cũng được sử dụng trong các nhiệm vụ phân loại văn bản để xử lý và biểu diễn văn bản dưới dạng các đơn vị nhỏ hơn. Điều này giúp giảm kích thước dữ liệu và tăng tính đặc trưng của từ ngữ dễ dàng giúp mô hình học các đặc trưng đó và phân loại**.**

## 3.2. Công cụ thực hiện

### 3.2.1. fairseq

**Định nghĩa**

Được phát triển bởi Facebook AI Research (FAIR) và được viết bằng ngôn ngữ Python và PyTorch. Thêm vào đó, fairseq còn hỗ trợ cho nhiều ngôn ngữ khác như Lua, C++.

Là một một bộ công cụ mô hình hóa chuỗi mã nguồn mở, cho phép các nhà nghiên cứu và nhà phát triển huấn luyện các mô hình tùy chỉnh cho dịch máy, tóm tắt, mô hình ngôn ngữ và các nhiệm vụ tạo văn bản khác.

**Ưu điểm**

* Cung cấp các mô hình mạnh mẽ để giải quyết các tác vụ xử lý ngôn ngữ tự nhiên.
* Có thể xử lý được với các mô hình và phức tạp.
* Được phát triển bởi một cộng đồng người dùng lớn là Facebook AI Research nên sẽ được tinh chỉnh liên tục và có nhiều tài liệu hỗ trợ.

**Nhược điểm**

* Khó sử dụng với người mới vì có nhiều tùy chọn và cấu hình phức tạp.
* Đối với các mô hình lớn, yêu cầu phải có tài nguyên phần cứng mạnh.

### 3.2.3. Transformer

Là một mô hình neural network tiên tiến và được ứng dụng nhiều trong các tác vụ NLP quan trọng như dịch máy, tạo phụ đề, xử lý ngôn ngữ tự nhiên.

Transformer đã giúp giải quyết được vấn đề xử lý song song các phần dữ liệu đầu vào để tránh làm giảm hiệu suất khi xử lý các câu dài hoặc quan hệ xa giữa các từ. Transformer đã giải quyết được vấn đề này bằng cách sử dụng cơ chế self-attension (tự chú ý) để việc xử lý phụ thuộc xa giữa các từ được hiệu quả hơn. Cơ chế này giúp cho mô hình chú ý vào những từ quan trọng trong câu và xử lý chúng một cách song song.

Mô hình Transformer thông thường gồm 2 thành phần là **bộ mã hóa** (encoder) dùng để chuyển đổi dữ liệu đầu vào thành dạng biểu diễn nhiều chiều và **bộ giải mã** (decoder) sử dụng dữ liệu đã được mã hóa và thông tin ngữ cảnh để tạo ra dữ liệu đầu ra.

Được ứng dụng nhiều trong lĩnh vực dịch máy, phân loại văn bản, tự động tạo phụ đề, xác định ngữ cảnh câu và tạo các câu hỏi và câu trả lời.

**Ưu điểm**

* Hiệu suất mạnh mẽ do việc tìm hiểu sự phụ thuộc xa giữa các từ và xử lý chúng một cách song song.
* Xử lý được ngữ cảnh phức tạp nhờ vào cơ chế tự chú ý, đáp ứng được yêu cầu hiểu và tạo ra ngữ cảnh rõ ràng.

**Nhược điểm**

* Đòi hỏi dữ liệu lớn để huấn luyện mô hình nhằm xử lý đạt kết quả tốt trên bộ ngữ cảnh.
* Thuật toán tính toán phức tạp và đòi hỏi tính toán lớn.
* Không xử lý tuần tự như thuật toán RNN, do đó không phù hợp với các bài toán dự đoán từ tiếp theo trong một chuỗi.
* Việc đào tạo mô hình rất tốn kém thời gian và tài nguyên.

### 3.2.3. fastBPE

Là một công cụ được dùng phổ biến trong lĩnh vực xử lý ngôn ngữ tự nhiên và được sử dụng để mã hóa và giải mã dữ liệu dựa trên phương pháp mã hóa byte pair (Byte Pair Encoding – BPE).

Trong đó, phương pháp BPE thực hiện việc nén dữ liệu thông qua việc thay thế các cặp byte phổ biến bằng một ký tự đặc biệt. Điều này giúp cho kích thước của dữ liệu giảm đi nhưng không làm mất đi những thông tin quan trọng.

Thường được sử dụng trong các ứng dụng dịch máy, dự đoán từ tiếp theo, gán nhãn ngôn ngữ và các tác vụ xử lý ngôn ngữ tự nhiên khác.

**Ưu điểm**

* Được thiết kế để đạt được tốc độ xử lý nhanh và hiệu suất cao trong việc mã hóa và giải mã dữ liệu.
* Cho phép tùy chỉnh các tham số, kích thước từ điển và các quy tắc mã hóa để tối ưu hóa quá trình mã hóa và tăng tính linh hoạt.
* Xử lý được với các tập dữ liệu có độ phức tạp cao.

**Nhược điểm**

* Cần thêm bước tiền xử lý và các bước giải mã, điều này làm tăng độ phức tạp của mô hình và yêu cầu nhiều tài nguyên tính toán.
* Nếu tập dữ liệu đầu vào quá nhỏ có thể làm giảm hiệu suất đáng kể.

### 3.2.4. RDRSegmenter

RDRSegmenter là một công cụ phân đoạn từ tiếng Việt được phát triển bởi Đặng Quốc Nguyên và được viết bằng Java1. Công cụ này có tốc độ xử lý nhanh và độ chính xác cao.

Là một công cụ trong lĩnh vực xử lý ngôn ngữ tự nhiên được dùng để phân đoạn ngữ pháp, phân đoạn các từ trong văn bản dựa trên quy tắc (rule-based) và được phát triển dựa trên phân đoạn ngữ pháp RDR (Recursive Deep Model).

**Ưu điểm**

* Do dựa vào các quy tắc ngữ pháp để phân đoạn từ ngữ nên sẽ cho ra kết quả chính xác tương đối cao và đáng tin cậy.
* RDRSegmenter được áp dụng cho nhiều ngôn ngữ khác nhau, tạo nên sự hữu ích và linh hoạt khi áp dụng cho dữ liệu đa ngôn ngữ.
* Do có giao diện và API rõ ràng nên có thể tích hợp vào các dự án NLP khác nhau.

**Nhược điểm**

* Không xử lý được các từ không đồng nhất.
* Đòi hỏi phải có nguồn dữ liệu lớn, đa dạng về các quy tắc, cấu trúc ngữ pháp và văn bản mẫu.

### 3.2.5. VnCoreNLP

Là một thư viện NLP được phát triển bởi nhóm nghiên cứu ngôn ngữ tự nhiên tại Đại học Khoa học và Công nghệ Hà Nội.

Được xây dựng trên nền tảng CoreNLP nhằm cung cấp các chức năng phân tích ngôn ngữ tự nhiên cho tiếng Việt như phân đoạn, gán nhãn từ loại, phân tích cú pháp, xác định tên riêng, phân giải đồng tham chiếu và phân tích ngữ nghĩa.

Thư viện còn cung cấp một giao diện dễ sử dụng để tích hợp vào các ứng dụng và dự án NLP.

**Ưu điểm**

* Là công cụ hàng đầu được tối ưu hóa cho tiếng Việt với độ chính xác và đạt hiệu quả cao.
* Xử lý được các từ ghép các từ không phổ biến
* Được tích hợp sẵn các tác vụ NLP nên việc phát triển các ứng dụng NLP cho tiếng Việt sẽ dễ dàng hơn.
* Đạt hiệu suất cao, cho phép xử lý các văn bản lớn và phức tạp một cách nhanh chóng.

**Nhược điểm**

* Còn hạn chế một vài chức năng phức tạp trong việc xử lý ngôn ngữ tự nhiên cho tiếng Việt.
* Không xử lý được các từ không đồng nhất.
* Chưa cung cấp các chức năng kiểm tra chính tả cho tiếng Việt.

## 3.3. Kiến trúc

### 3.3.1. BERT

#### Giới thiệu

BERT (Bidirectional Encoder Representations from Transformers) là một mô hình ngôn ngữ (Language Model) được tạo ra bởi Google AI và được giới thiệu vào năm 2018. BERT được coi như là một bước đột phá lớn trong Machine Learning bởi vì khả năng ứng dụng vô cùng lớn của nó trong nhiều bài toán về xử lý ngôn ngữ tự nhiên (Natural Language Processing) như: Question Answering, …

Ưu điểm:

* Hoàn toàn miễn phí, có thể sử dụng BERT cho bài toán của riêng mình.

Nhược điểm:

* Phần lớn biến thể của BERT chỉ giới hạn ở ngôn ngữ tiếng Anh 🡪 Các ngôn ngữ khác cần đào tạo lại mô hình dành riêng cho ngôn ngữ bằng kiến trúc BERT hoặc sử dụng các mô hình dựa trên BERT đa ngôn ngữ đã được đào tạo trước đó.

#### Vai trò của BERT

Hiện nay các mô hình Deep Learning cần đến một lượng dữ liệu rất lớn để cho ra kết quả tốt. Nếu bạn muốn tận dụng được nguồn dữ liệu vô cùng lớn để giải quyết bài toán của mình thì Transfer Learning là lựa chọn thích hợp nhất. Đặc biệt là BERT. BERT sẽ đồng nhất các tập dữ liệu khổng lồ trên internet (pre-training). Điều này sẽ tinh chỉnh fine-tune để người dùng giải được các bài toán cụ thể. Kỹ thuật của BERT không những phù hợp với NLP mà còn phù hợp với Computer Vision.

Nhờ có kỹ thuật này mà kết quả cho các bài toán được cải thiện rõ rệt, không chỉ trong NLP mà còn trong các lĩnh vực khác như Computer Vision.

Graphical user interface, diagram, application

Description automatically generated

Hình 3.2: Mô hình Pre-training và Fine-Tuning cho BERT

* Pre-training:
  + Là quá trình huấn luyện mô hình BERT trên một tập dữ liệu lớn và đa dạng để học được đặc trưng ngôn ngữ tổng quát.
  + Quá trình Pre-training bao gồm 2 bước chính:
    - Pre-training trên tập dữ liệu lớn
    - Fine-tuning trên các tác vụ cụ thể
* Fine-tuning:
  + Là quá trình điều chỉnh lại các trọng số của mô hình BERT được huấn luyện trước đó trên 1 tập dữ liệu mới và giải quyết 1 tác vụ cụ thể. Mục đích chính là cải thiện độ chính xác của BERT trên 1 tác vụ cụ thể.
  + Trong quá trình này BERT được sử dụng như một mô hình cơ sở và được tinh chỉnh lại bằng cách đưa vào tập dữ liệu mới cùng với 1 tác vụ cụ thể để huấn luyện lại. VD: giải quyết tác vụ phân loại văn bản thì ta dùng BERT được huấn luyện trước đó tiến hành huấn luyện lại trên các tập dữ liệu chứa các văn bản được phân loại.
  + Quá trình Fine-tunning gồm 3 bước chính sau:
    - Tạo tập dữ liệu huấn luyện mới cho tác vụ cụ thể.
    - Fine-tunning BERT trên tập dữ liệu mới.
    - Kiểm tra độ chính xác của BERT.

🡺 Với Pre-training và Fine-tuning, BERT đã chứng tỏ được khả năng mạnh mẽ của mình trong việc giải quyết vấn đề ngôn ngữ tự nhiên và trở thành một trong những mô hình ngôn ngữ được ứng dụng rộng rãi hiện nay.

#### Đổi mới của BERT

Trong khi huấn luyện mô hình, một thách thức gặp phải là các mô hình directional truyền thống gặp giới hạn khi học ngữ cảnh của từ. Để khắc phục nhược điểm của các mô hình cũ, BERT sử dụng 2 chiến lược training như sau:

* Masked LM (MLM): Trước khi đưa vào BERT, thì 15% số từ trong chuỗi được thay thế bởi token [MASK], khi đó mô hình sẽ dự đoán từ được thay thế bởi [MASK] với context là các từ không bị thay thế bởi [MASK]. Masked LM gồm các bước xử lý sau:
  + Thêm một classification layer với input là output của Encoder.
  + Nhân các vector đầu ra với ma trận embedding để đưa chúng về không gian từ vựng (vocabulary dimensional).
  + Tính toán xác suất của mỗi từ trong tập từ vựng sử dụng hàm softmax.
* Trên thực tế con số 15% không cố định mà có thể thay đổi theo mục đích bài toán.
* Next Sentece Prediction (NSP): Mô hình sử dụng một cặp câu là dữ liệu đầu vào và dự đoán câu thứ 2 là câu tiếp theo của câu thứ 1 hay không. Trong quá trình huấn luyện, 50% lượng dữ liệu đầu vào là cặp câu trong đó câu thứ 2 thực sự là câu tiếp theo của câu thứ 1, 50% còn lại thì câu thứ 2 được chọn ngẫu nhiên từ tập dữ liệu. Nguyên tắc làm:
  + Chèn token [CLS] vào trước câu đầu tiên và [SEP] vào cuối mỗi câu.
  + Các token trong từng câu được đánh dấu là A hoặc B.
  + Chèn thêm vector embedding biểu diễn vị trí của token trong câu (chi tiết về vector embedding này có thể tìm thấy trong bài báo về Transformer).

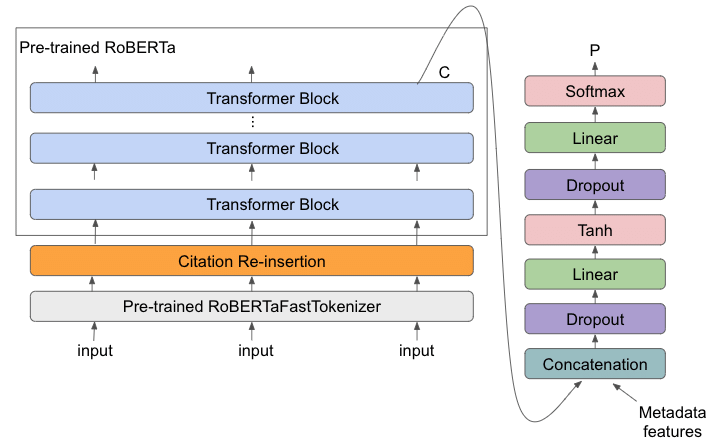
Diagram

Description automatically generated

Hình 3.3: Ví dụ cho NSP

### 3.3.2. RoBERTa

#### Giới thiệu



Hình 3.4: Giới thiệu RoBERTa

RoBERTa là một mô hình ngôn ngữ được phát triển bởi Facebook AI Research (FAIR) vào năm 2019. Nó là một phiên bản cải tiến của mô hình BERT và được đào tạo trên một lượng lớn dữ liệu hơn so với BERT.

RoBERTa sử dụng một loạt các kỹ thuật tinh chỉnh để cải thiện quá trình huấn luyện và nâng cao hiệu suất của mô hình, bao gồm điều chỉnh các siêu tham số, tăng kích thước mô hình, mở rộng tập dữ liệu huấn luyện và thực hiện các quá trình tiền huấn khác. Nhờ những cải tiến này, RoBERTa đạt được kết quả tốt hơn so với BERT trên nhiều tác vụ ngôn ngữ tự nhiên.

#### Kỹ thuật cải tiến

RoBERTa sử dụng cùng một kiến trúc mạng nơ-ron như BERT, nhưng có một số cải tiến trong quá trình huấn luyện. Cụ thể, RoBERTa áp dụng các kỹ thuật cải tiến như:

* **Không sử dụng mã hóa vị trí tương đối** **(Relative Positional Encoding)**: Thay vì sử dụng mã hóa vị trí tương đối để định vị vị trí của từ, RoBERTa sử dụng mã hóa vị trí tuyệt đối (Absolute Positional Encoding) để chỉ định vị trí của từ trong câu.
* **Không sử dụng token** **[MASK]**: Thay vì sử dụng token [MASK] như trong BERT để đưa ra câu hỏi cho mô hình, RoBERTa sử dụng một phương pháp khác để tạo ra dữ liệu huấn luyện, đó là sử dụng các phương pháp xáo trộn (data augmentation) để tạo ra các câu mới từ các câu đã có trong bộ dữ liệu. Cụ thể:
* Trong quá trình tiền xử lý dữ liệu, BERT sử dụng một ký hiệu đặc biệt là [MASK] để thay thế ngẫu nhiên một số từ trong câu để cho mô hình học được khả năng dự đoán từ đó. Tuy nhiên, phương pháp này có thể gây ra sự hiện diện của [MASK] quá nhiều trong dữ liệu, dẫn đến sự thiếu đa dạng và overfitting.
* Với RoBERTa, các tác giả đã thay đổi phương pháp tiền xử lý dữ liệu bằng cách sử dụng một phương pháp mới để xáo trộn chuỗi dữ liệu một cách hiệu quả hơn. Thay vì sử dụng [MASK], RoBERTa sử dụng phương pháp đưa ra một danh sách các câu khác nhau, sau đó xáo trộn các câu đó và lấy ra một số từ ngẫu nhiên để tạo thành câu mới. Phương pháp này giúp tăng tính đa dạng của dữ liệu và giảm thiểu sự xuất hiện quá nhiều của [MASK], giúp cải thiện khả năng mô hình học và tránh hiện tượng overfitting.
* **Huấn luyện với thêm dữ liệu**: RoBERTa sử dụng một bộ dữ liệu huấn luyện lớn hơn nhiều so với BERT và các mô hình tiền đề khác. Bộ dữ liệu huấn luyện của RoBERTa bao gồm các nguồn khác nhau như Wikipedia, Common Crawl, BooksCorpus và một số nguồn báo chí điện tử. Điều này giúp tăng độ đa dạng của dữ liệu và cải thiện hiệu suất của mô hình.
* **Tối ưu hóa tham số huấn luyện**: RoBERTa sử dụng các kỹ thuật tối ưu hóa tham số mới để cải thiện hiệu suất của mô hình. Cụ thể, RoBERTa sử dụng kỹ thuật "dynamic masking" để ngăn chặn mô hình quá tập trung vào một phần nhất định của dữ liệu huấn luyện.

Ngoài ra để nâng cao độ chuẩn xác trong biểu diễn từ thì RoBERTa đã loại bỏ tác vụ dự đoán câu tiếp theo và huấn luyện trên các câu dài hơn.

#### Ứng dụng

* Một số ứng dụng của RoBERTa bao gồm:
* **Phân loại văn bản**: RoBERTa được sử dụng để phân loại các văn bản trong nhiều lĩnh vực khác nhau, chẳng hạn như phân loại cảm xúc, phân loại tin tức, phân loại định hướng chính trị, phân loại ngôn ngữ, …
* **Trích xuất thông tin**: RoBERTa được sử dụng để trích xuất thông tin từ các văn bản, chẳng hạn như trích xuất tên, địa chỉ, ngày tháng, … từ các tài liệu hợp đồng hoặc bản tin báo chí.
* **Sinh câu**: RoBERTa được sử dụng để sinh câu tự động, giúp cho việc viết văn bản nhanh chóng và hiệu quả hơn.
* **Dịch máy**: RoBERTa được sử dụng trong các hệ thống dịch máy để cải thiện chất lượng và độ chính xác của kết quả dịch.
* **Xử lý ngôn ngữ tự nhiên**: RoBERTa được sử dụng để giải quyết các vấn đề trong xử lý ngôn ngữ tự nhiên, chẳng hạn như phân tích cú pháp, phân tích tình cảm, nhận dạng thực thể tên, …

Một trong những ứng dụng thực tế của RoBERTa là trong việc xử lý ngôn ngữ tự nhiên và phân tích cảm xúc. Ví dụ, các công ty có thể sử dụng RoBERTa để phân tích cảm xúc của khách hàng trên các kênh truyền thông xã hội như Twitter hoặc Facebook. RoBERTa có thể phân tích và đánh giá nội dung từ những bài viết trên các trang mạng xã hội, từ đó giúp các công ty có thể hiểu được cảm nhận của khách hàng đối với sản phẩm hoặc dịch vụ của mình và đưa ra các cải tiến hoặc điều chỉnh phù hợp. Ngoài ra, RoBERTa cũng có thể được sử dụng trong các ứng dụng như dịch thuật tự động, nhận diện giọng nói, tóm tắt văn bản và phân tích tâm lý khách hàng.

# GIỚI THIỆU DATASET

Chương IV của đề tài sẽ giới thiệu bộ dữ liệu nhóm chọn để giải quyết bài toán ứng dụng PhoBERTbase trong bài toán phân loại Tiếng Việt. Nội dung chương sẽ trình bày về nguồn gốc, thống kê và dữ liệu train, test, validate của từng bộ dữ liệu.

## 4.1. Mô tả về tập dữ liệu

### 4.1.1. VNTC

Bộ dữ liệu VNTC của các tác giả Cong Duy Vu Hoang, Dien Dinh, Le Nguyen Nguyen, Quoc Hung Ngo.

Được công bố trong bài báo “A Comparative Study on Vietnamese Text Classification Methods” trong hội nghị IEEE International Conference on Research, Innovation and Vision for the Future (RIVF 2007).

Bộ dữ liệu được thu thập từ các trang báo Việt Nam như VnExpress, Tuổi trẻ, Thanh niên, …

### 4.1.2. Emotion

Bài toán phân loại cảm xúc Tiếng Việt sử dụng bộ dữ liệu được công bố trong bài báo Emotion Recognition for Vietnamese Social Media Text của tác giả Vong Anh Ho, Duong Huynh, Cong Nguyen, Danh Hoang Nguyen, Linh Thi Van Pham, Duc Vu Nguyen, Kiet Van Nguyen và Ngan Luu Thuy Nguyen.

Được công bố trong Hội nghị quốc tế thứ 16 của Hiệp hội Ngôn ngữ học tính toán Thái Bình Dương (PACLING 2019) là một hội nghị tập trung vào ngôn ngữ học tính toán, diễn ra vào năm 2019.

## 4.2. Tập dữ liệu sẽ sử dụng

### 4.2.1. VNTC 10Topics

#### 4.2.1.1. Tập Train

Bộ dữ liệu được sử dụng cho quá trình huấn luyện bao gồm 10 chủ đề khác nhau, gồm: Vi tính, Văn hóa, Thể thao, Thế giới, Sức khỏe, Pháp luật, Kinh doanh, Khoa học, Đời sống, Chính trị xã hội.

Tổng số lượng mẫu tin có trong bộ dữ liệu này là 33,759 tệp. Hình dưới đây thể hiện chi tiết số lượng mẫu tin có trong từng chủ đề trong tập huấn luyện.

A picture containing text, screenshot, number, line

Description automatically generated

Hình 4.1: Số lượng mẫu tin của mỗi thể loại trong tập Train\_Full

#### 4.2.1.2. Tập Test

Bộ dữ liệu được sử dụng để kiểm thử có các chủ đề tương đương với bộ dữ liệu đã được sử dụng trong quá trình huấn luyện, với tổng số mẫu tin là 50,373. Biểu đồ dưới đây cung cấp thông tin chi tiết về số lượng mẫu tin trong từng chủ đề trong tập kiểm tra.

A picture containing text, screenshot, number, line

Description automatically generated

Hình 4.2: Số lượng mẩu tin của mỗi thể loại trong tập Test\_Full

Có thể thấy bộ dữ liệu này nhiều hơn bộ dữ liệu huấn luyện xấp xỉ 1,5 lần. Điều này giúp đảm bảo rằng việc kiểm tra mô hình trên các chủ đề tương tự sẽ được thực hiện một cách đầy đủ. Cụ thể:

* Chủ đề có số lượng mẫu tin nhiều nhất trong bộ dữ liệu huấn luyện là “Chính trị xã hội” với 5,219 mẫu tin, trong khi chủ đề có số lượng mẫu tin nhiều nhất trong bộ dữ liệu kiểm thử là “Chính trị xã hội” với 7,567 mẫu tin. Tương tự, chủ đề có số lượng mẫu tin ít nhất trong bộ dữ liệu huấn luyện là “Khoa học” với 1,820 mẫu tin, trong khi chủ đề có số lượng mẫu tin ít nhất trong bộ dữ liệu kiểm thử là “Đời sống” với 2,036 mẫu tin.
* Một số chủ đề có sự chênh lệch đáng kể về số lượng mẫu tin giữa hai bộ dữ liệu. Ví dụ, chủ đề “Văn hóa” có 3,080 mẫu tin trong bộ dữ liệu huấn luyện, trong khi trong bộ dữ liệu kiểm thử có 6,250 mẫu tin, tức là hơn gấp đôi.
* Các chủ đề như “Thể thao”, “Thế giới”, “Sức khỏe”, và “Kinh doanh” cũng có sự chênh lệch đáng kể về số lượng mẫu tin giữa hai bộ dữ liệu.

### 4.2.2. Emotion

Tập dữ liệu giới thiệu về cảm xúc trên mạng xã hội tiếng Việt với khoảng 6.927 câu được gán nhãn bởi con người với sáu nhãn cảm xúc: Other, Disgust, Enjoyment, Anger, Surprise, Sadness, Fear.

Các tập train, test, validate được chia theo tỉ lệ 8:1:1. Cụ thể như sau:

#### 4.2.2.1. Tập Train

Tập dữ liệu được sử dụng để huấn luyện bao gồm 5548 câu. Biểu đồ dưới đây cung cấp thông tin chi tiết về số lượng mẫu tin trong từng chủ đề trong tập huấn luyện.

A picture containing text, screenshot, diagram, rectangle

Description automatically generated

Hình 4.3: Biểu đồ trực quan hóa dữ liệu tập train UIT – VSMEC

#### 4.2.2.2. Tập Test

Bộ dữ liệu được sử dụng để kiểm thử có tổng số câu là 693. Biểu đồ dưới đây cung cấp thông tin chi tiết về số lượng mẫu tin trong từng chủ đề trong tập kiểm tra.

A picture containing text, screenshot, diagram, rectangle

Description automatically generated

Hình 4.4: Biểu đồ trực quan hóa dữ liệu tập test UIT – VSMEC

#### 4.2.2.3. Tập Validate

Bộ dữ liệu được sử dụng để validate có tổng số câu là 686. Biểu đồ dưới đây cung cấp thông tin chi tiết về số lượng mẫu tin trong từng chủ đề trong tập validate.

A picture containing text, screenshot, diagram, rectangle

Description automatically generated

Hình 4.5: Biểu đồ trực quan hóa dữ liệu tập validate UIT - VSMEC

Có thể thấy không có sự cân bằng giữa số lượng câu từng các nhãn cảm xúc ở cả ba bộ dữ liệu. Việc phân bổ số lượng câu ở các nhãn sẽ hỗ trợ đánh giá độ hiệu quả của mô hình khi áp dụng trên bộ dataset UIT – VSMEC. Cụ thể: Nhãn có số lượng câu cao nhất là Enjoyment có số câu gấp 5-7 lần nhãn Fear.

Tuy nhiên, số lượng câu theo từng nhãn lại được phân bố theo tỉ lệ khá tương đồng ở cả ba bộ dữ liệu. Cả ba bộ dữ liệu đều có số lượng câu ở nhãn Enjoyment, Other, Sadness, Disgust là cao nhất, nhãn có số lượng câu ít nhất là Fear và Surprise.

# THỰC NGHIỆM

Chương V của đề tài sẽ trình bày thực nghiệm lại bài toán trên bộ dữ liệu được giới thiệu ở chương IV. Ngoài ra nội dung về độ đo cũng như kết quả thực nghiệm sẽ được nhóm trình bày trong chương này.

## 5.1. Giới thiệu độ đo

### 5.1.1. Confusion matrix

|  |  |  |
| --- | --- | --- |
| Lớp thực tế  Lớp dự đoán | C | C |
| C | True Positives  (TP) | False Negatives  (FN) |
| C | False Positives  (FP) | True Negatives  (TN) |

Bảng 5.1: Ma trận nhầm lẫn

Trong đó:

* True positive (TP): Số lượng mẫu được dự đoán đúng thuộc vào lớp positive.
* True negative (TN): Số lượng mẫu được dự đoán đúng là thuộc vào lớp negative.
* False positive (FP): Số lượng mẫu được dự đoán sai là thuộc vào lớp positive, trong khi thực tế lại thuộc vào lớp negative.
* False negative (FN): Số lượng mẫu được dự đoán sai là thuộc vào lớp negative, trong khi thực tế lại thuộc vào lớp positive.

### 5.1.2. Accuracy (Độ chính xác)

Accuracy là tỷ lệ phần trăm của số lượng dự đoán đúng trên tổng số lượng dữ liệu được đánh giá.

Công thức:

### 5.1.3. F1-score

F1-score là một phép đo kết hợp giữa Precision (độ chính xác dương tính) và Recall (độ phủ dương tính). Nó là một con số trong khoảng từ 0 đến 1, thể hiện trung bình điều hòa giữa Precision và Recall.

Công thức:

F1-score =

Trong đó:

* Precision =
* Recall =

## 5.2. Thực nghiệm trên các bài toán

### 5.2.1. VNTC

Quy trình và các bước chi tiết trong việc chuẩn bị dữ liệu, huấn luyện, đánh giá và áp dụng mô hình PhoBERT cho nhiệm vụ phân loại văn bản đơn:

1. Chuẩn bị dữ liệu đầu vào:

* Tiền xử lý: Bước này thực hiện tách từ (tokenization) để chia câu hoặc đoạn văn thành các từ riêng biệt. Sau đó, chúng ta sử dụng mã hóa BPE (Byte-Pair Encoding) để chuyển đổi từ văn bản thành các token. Mô hình PhoBERT sử dụng BPE để giải quyết vấn đề khó khăn về việc mã hóa các từ ngữ trong ngôn ngữ Việt.
* Định dạng đầu vào: Mô hình PhoBERT yêu cầu đầu vào là một tensor 2D có kích thước cố định. Mỗi hàng của tensor đại diện cho một câu hoặc đoạn văn bản. Để đạt được kích thước cố định, chúng ta có thể sử dụng các phương pháp như padding (điền thêm giá trị 0 vào các câu ngắn hơn) hoặc cắt bỏ các phần dư để điều chỉnh độ dài của các câu.

1. Huấn luyện mô hình:

* Chuẩn bị dữ liệu huấn luyện: Cần có một tập dữ liệu huấn luyện được gắn nhãn đúng để phân loại. Mỗi mẫu văn bản trong tập dữ liệu huấn luyện được kết hợp với một nhãn tương ứng.
* Xây dựng mô hình: Xây dựng một mô hình sử dụng PhoBERT đã được huấn luyện trước đó. Mô hình này thường bao gồm các lớp biến đổi và lớp phân loại.
* Tạo optimizer và loss function: Sử dụng thuật toán tối ưu AdamW và hàm mất mát (loss function) như cross-entropy để huấn luyện mô hình.
* Huấn luyện mô hình: Từ tập dữ liệu huấn luyện, chúng ta sử dụng một vòng lặp để truyền qua mô hình các mẫu văn bản, tính toán gradient và cập nhật trọng số của mô hình để tối thiểu hóa hàm mất mát. Quá trình này được thực hiện qua nhiều epoch cho đến khi mô hình đạt được hiệu suất mong muốn.

1. Đánh giá và tinh chỉnh:

* Đánh giá kết quả: Sau khi huấn luyện, chúng ta đánh giá hiệu suất của mô hình trên tập dữ liệu kiểm tra. Các phép đo như độ chính xác (accuracy) và F1-score được sử dụng để đánh giá khả năng phân loại của mô hình. Độ chính xác tính tỷ lệ giữa số lượng dự đoán đúng và tổng số mẫu, trong khi F1-score tính toán sự kết hợp giữa độ chính xác và độ phủ (recall) của mô hình.
* Tinh chỉnh mô hình: Dựa trên kết quả đánh giá, chúng ta có thể tinh chỉnh mô hình bằng cách điều chỉnh các siêu tham số như learning rate, số lượng epoch, hoặc thay đổi kiến trúc mạng để cải thiện hiệu suất phân loại. Một số kỹ thuật khác như tăng cường dữ liệu (data augmentation) cũng có thể được áp dụng để tăng tính đa dạng và khả năng tổng quát hóa của mô hình.

1. Áp dụng mô hình cho dữ liệu mới:

* Tiền xử lý dữ liệu mới: Trước khi đưa dữ liệu mới vào mô hình, chúng ta cần thực hiện các bước tiền xử lý như tách từ và chuyển đổi văn bản thành dạng số hóa.
* Đưa dữ liệu mới vào mô hình: Sau khi tiền xử lý, chúng ta chạy các đoạn văn bản mới qua mô hình PhoBERT để nhận được dự đoán nhãn tương ứng cho chúng.

### 5.2.2. Emotion

Quy trình và các bước tiến hành thực nghiệm áp dụng mô hình PhoBERTbase cho bài toán phân loại cảm xúc bao gồm các bước chính sau:

1. Chuẩn bị:

* Sử dụng các packages hỗ trợ: Sử dụng openpyxl để thực hiện đọc, ghi và chỉnh sửa dữ liệu trong các tệp Excel, cho phép thao tác với các bảng tính, các ô và các công thức trong tệp Excel. Sử dụng thư viện transformers áp dụng các mô hình để thực hiện các tác vụ NLP.
* Chuẩn bị dữ liệu: Sử dụng bộ dữ liệu với các tỉ lệ chia train:test:validate đã được cung cấp sẵn.

1. Huấn luyện mô hình

* Tiền xử lý dữ liệu đầu vào: Thực hiện tách từ (tokenization) để chia câu hoặc đoạn văn thành các từ riêng biệt.
* Xây dựng mô hình:
  + Tạo lớp để định nghĩa và xử lý dữ liệu cho tập dữ liệu văn bản (sentiment dataset) trong quá trình huấn luyện mô hình.
  + Định nghĩa mô hình cho phân loại cảm xúc trong văn bản sử dụng kiến trúc RoBERTa.
* Huấn luyện mô hình:
  + Huấn luyện mô hình bằng cách thực hiện các vòng lặp qua dữ liệu huấn luyện, tính toán loss, cập nhật gradient và cập nhật các tham số của mô hình để tối ưu hóa loss.
  + Thực hiện đánh giá mô hình bằng cách thực hiện các vòng lặp qua dữ liệu kiểm tra hoặc validation, tính toán loss và độ chính xác của mô hình.
  + K-fold cross-validation: Thực hiện huấn luyện và đánh giá mô hình trên mỗi fold trong quá trình cross-validation với mục đích đánh giá khả năng tổng quát hóa của mô hình trên các dữ liệu mới mà chưa được sử dụng trong quá trình huấn luyện.

1. Đánh giá:

* Đánh giá hiệu suất mô hình: Đánh giá độ chính xác và giá trị trung bình loss của mô hình trên tập dữ liệu kiểm thử. Bằng cách tính toán mất mát và đếm số lượng dự đoán đúng, ta có thể đánh giá hiệu suất của mô hình và điều chỉnh các tham số để cải thiện kết quả.

1. Áp dụng mô hình cho dữ liệu thực tế:

* Đưa dữ liệu mới vào mô hình: Dự đoán cảm xúc của các câu văn mới bằng cách gọi hàm và truyền vào câu văn cần dự đoán.

## 5.3. Đánh giá kết quả thực nghiệm

Đối với bài toán phân loại văn bản, dựa trên kết quả đạt được của mô hình trên tập dữ liệu kiểm tra, ta có thể rút ra những nhận xét sau:

* Mô hình phân loại văn bản đã đạt được hiệu suất tốt với F1-score và accuracy đạt khoảng 0.9029 đối với bài toán phân loại văn bản. Điều này cho thấy mô hình có khả năng phân loại văn bản với độ chính xác cao và độ phủ tốt.
* Kết quả F1-score và accuracy gần như bằng nhau cho thấy mô hình không có sự chênh lệch lớn giữa các lớp phân loại. Điều này chứng tỏ mô hình có khả năng đồng nhất trong việc phân loại các lớp văn bản, không có lớp nào bị ưu ái hoặc bị bỏ sót nhiều.
* Hiệu suất cao trên tập dữ liệu kiểm tra cho thấy mô hình có khả năng tổng quát hóa tốt. Điều này có ý nghĩa rằng mô hình có khả năng áp dụng cho các tài liệu văn bản mới mà chưa được mô hình quen thuộc trước đó.

Kết quả thực nghiệm trên cả hai bài toán phân loại văn bản và phân loại cảm xúc cho thấy việc ứng dụng PhoBERTbase đối với các bài toán phân loại đơn nhãn mang tính hiệu quả. Bên cạnh đó, mô hình PhoBERTbase cho thấy khả năng ứng dụng cao đối với các bài toán xử lý ngôn ngữ tự nhiên cho tiếng Việt.

# KẾT LUẬN

Nội dung chương VI sẽ đề cập những kết quả được của đồ án cũng như những hạn chế mà nhóm gặp phải. Bên cạnh đó sẽ là hướng phát triển tiếp theo của đề tài mà nhóm đã chọn trong thời gian sắp tới.

## 6.1. Kết quả đạt được

Trong lần thực hiện đề tài này nhóm đã học và tiếp thu được các kiến thức vô cùng quan trọng về cách một mô hình được xây dựng, sử dụng, tác dụng cũng như ưu điểm và nhược điểm của một mô hình.

Nhóm cũng đã thực hiện hóa được việc ứng dụng một mô hình PhoBERT-base để tạo train ra một mô hình phân loại văn bản Tiếng Việt. Nhóm đã thử nghiệm thành công và kiểm tra được trên các câu với các dữ liệu khác nhau với sự chính xác lên đến gần 90%.

Bên cạnh nhóm cũng đã nghiên cứu thêm về các ứng dụng của PhoBERT-base để cho ra thêm một ứng dụng về phân loại cảm xác cũng dựa vào tiếng việt với độ chính xác tương tự.

## 6.2. Hạn chế

Mặc dù đã làm được rất nhiều nhưng bên cạnh đó nhóm vẫn mắc phải các vấn đề sau:

* Vẫn chưa thể xây dựng được một mô hình: Nhóm đã tiếp cận theo hướng là sử dụng một mô hình PhoBERT-base đã được pre-Trained sẵn và tải về tiến hành train lại để có thể sử dụng được cho việc phân loại văn bản.
* Phụ thuộc gần như rất nhiều vào câu dữ liệu cuối cùng mà ta đưa vào: Việc phân loại văn bản của mô hình còn dựa khá nhiều vào dữ liệu cuối cùng mà chúng ta sẽ đưa vào mô hình hơn là việc phân tích một đoạn văn dài để tìm xem đoạn văn hướng về nội dung nào nhiều hơn (thường xảy ra khi dữ liệu thuộc vào loại mang nhiều chủ đề)
* Chỉ mới phân loại được đơn nhãn: Như đã có nói ở trên thì đây là vấn chính của mô hình khi thể phân biệt được đa nhãn mà sẽ chỉ dựa vào dữ liệu cuối cùng được đưa vào.

## 6.3. Hướng phát triển

Để khắc phục được các hạn chế nêu trên cũng như phát triển được mô hình một cách tốt hơn thì nhóm đặt ra các hướng phát triển sau:

* Tìm hiểu thêm về cách có thể xây dựng được một mô hình ngay từ đầu mà không sử dụng một mô hình có sẵn.
* Tiến hành train lại mô hình với một bộ dữ liệu lớn hơn, đa dạng hơn, giúp mô hình dễ dàng phân biệt được rõ hơn các chủ đề.
* Tiến hành việc nghiên cứu để có thể ứng dụng được mô hình trong phân loại đa nhãn để góp phần làm đa dạng ứng dụng của mô hình.

# PHÂN CÔNG CÔNG VIỆC

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tên thành viên  Nội dung công việc | | | Nguyễn Thành Phát | Nguyễn Anh Kiệt | Trần Thị  Mỹ Nhung | Nguyễn Hoài Linh |
| Chương 1 | 1.1 | | x | x | x | x |
| 1.2 | |  |  |  | x |
| 1.3 | | x |  |  | x |
| Chương 2 | 2.1 | | x | x |  | x |
| 2.2 | | x | x |  | x |
| 2.3 | |  | x |  | x |
| Choung 3 | 3.1 | | x |  | x |  |
| 3.2 | |  |  | x | x |
| 3.3 | | x | x |  |  |
| Chương 4 | 4.1 | VNTC |  | x |  |  |
| Emotion |  |  | x |  |
| 4.2 | VNTC 10 |  | x |  |  |
| Emotion |  |  | x |  |
| Chương 5 | 5.1 | |  | x |  |  |
| 5.2 | VNTC |  | x |  |  |
| Emotion |  |  | x |  |
| 5.3 | |  |  | x |  |
| Chương 6 | | | x |  | x |  |
| Tài liệu tham khảo | | | x | x | x | x |
| Chỉnh sửa tài liệu | | | x | x | x | x |
| Đánh giá mức độ  hoàn thành | | | 100% | 100% | 100% | 100% |

Bảng 6.1: Bảng phân công công việc

# TÀI LIỆU THAM KHẢO

[1] BERT base, BERT large. Truy cập từ: [BERT base vs BERT large (opengenus.org)](https://iq.opengenus.org/bert-base-vs-bert-large/?fbclid=IwAR2oCkwbiX0X0BWyyCG2tjlZS-OR6gjNxBbYclv2GBK7LB4_5lGzGu_KRis#:~:text=BERT%20base%20vs%20BERT%20large%20BERT%20is%20based,of%20encoders%20stacked%20on%20top%20of%20each%20other)

[2] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in Neural Information Processing Systems (pp. 5998-6008).

[3] Khanh, P. D. (2020). Cách thức cài đặt PhoBERT: Một mô hình ngôn ngữ tiếng Việt mã nguồn mở dựa trên BERT. Truy cập từ: [https://phamdinhkhanh.github.io/2020/06/04/PhoBERT\_Fairseq.htmlhttps://miai.vn/2020/12/14/bert-series-c](https://phamdinhkhanh.github.io/2020/06/04/PhoBERT_Fairseq.htmlhttps:/miai.vn/2020/12/14/bert-series-c)

[[4]](https://miai.vn/2020/12/14/bert-series-chuong-1-bert-la-cai-chi-chi/) [Lai, V., Tran, T., & Pham, T. T. (2020). BERT Series - Chương 1: BERT là cái gì? Truy cập từ: https://miai.vn/2020/12/14/bert-series-chuong-1-bert-la-cai-chi-chi/](https://uithcm-my.sharepoint.com/personal/20520270_ms_uit_edu_vn/Documents/Lai,%20V.,%20Tran,%20T.,%20&%20Pham,%20T.%20T.%20(2020).%20BERT%20Series%20-%20Chương%201:%20BERT%20là%20cái%20gì?%20Truy%20cập%20từ:%20https://miai.vn/2020/12/14/bert-series-chuong-1-bert-la-cai-chi-chi/)[huong-1-bert-la-cai-chi-chi/](https://miai.vn/2020/12/14/bert-series-chuong-1-bert-la-cai-chi-chi/)

[[5]](https://miai.vn/2020/12/14/bert-series-chuong-1-bert-la-cai-chi-chi/) 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. Truy cập từ: [1907.11692.pdf (arxiv.org)](https://arxiv.org/pdf/1907.11692.pdf)

[[6]](https://miai.vn/2020/12/14/bert-series-chuong-1-bert-la-cai-chi-chi/)  XNLI: Evaluating Cross-lingual Sentence Representations. Truy cập từ: [arxiv.org/pdf/1809.05053.pdf](https://arxiv.org/pdf/1809.05053.pdf)

[[7]](https://miai.vn/2020/12/14/bert-series-chuong-1-bert-la-cai-chi-chi/)  Unsupervised Cross-lingual Representation Learning at Scale. Truy cập từ: [1911.02116.pdf (arxiv.org)](https://arxiv.org/pdf/1911.02116.pdf)

[[8]](https://miai.vn/2020/12/14/bert-series-chuong-1-bert-la-cai-chi-chi/) Hieu, N. V. (n.d.). Fine-tuning pre-trained model trong PyTorch và áp dụng vào visual saliency prediction. Truy cập từ: [Fine tuning pre-trained model trong pytorch và áp dụng vào Visual Saliency Prediction (viblo.asia)](https://viblo.asia/p/fine-tuning-pre-trained-model-trong-pytorch-va-ap-dung-vao-visual-saliency-prediction-4P856LY1ZY3)

[[9]](https://miai.vn/2020/12/14/bert-series-chuong-1-bert-la-cai-chi-chi/) PyTorch Tutorials. (n.d.). Transfer Learning for Computer Vision Tutorial. Truy cập từ: [Transfer Learning for Computer Vision Tutorial — PyTorch Tutorials 2.0.1+cu117 documentation](https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html)

[10] Toward Data Science. (2020). A Gentle Introduction to Transfer Learning in NLP. Truy cập từ: [A Gentle Introduction to Transfer Learning in NLP | by Neil Sinclair | Towards Data Science](https://towardsdatascience.com/a-gentle-introduction-to-transfer-learning-in-nlp-b71e87241d66)

[11] IBM. (n.d.). Natural Language Processing. Truy cập từ: <https://www.ibm.com/topics/natural-language-processing>

[12] Hugging Face. (n.d.). Tokenizer Summary. Truy cập từ: [Summary of the tokenizers (huggingface.co)](https://huggingface.co/docs/transformers/tokenizer_summary)

[13] VnCoreNLP: A Vietnamese Natural Language Processing Toolkit. Truy cập từ: [1801.01331.pdf (arxiv.org)](https://arxiv.org/pdf/1801.01331.pdf)

[14] Baeldung. (n.d.). Neural Network Pre-training. Truy cập từ: [What Does Pre-training a Neural Network Mean? | Baeldung on Computer Science](https://www.baeldung.com/cs/neural-network-pre-training)

[15] [Vietnamese Emotion Classification using PhoBERT | Kaggle](https://www.kaggle.com/code/hmthanh/vietnamese-emotion-classification-using-phobert)

[16] [sLeeNguyen/nlp-text-classification: This project performs fine-tuning PhoBERT for text classification using VNTC dataset (github.com)](https://github.com/sLeeNguyen/nlp-text-classification)