

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM  
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG  
KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CUỐI KỲ  
XỬ LÝ DỮ LIỆU LỚN**

**PROJECT 3**

*Người hướng dẫn:* **TS. BÙI THANH HÙNG**

*Người thực hiện:* **NGUYỄN QUỐC CƯỜNG – 518H0003**

**THÂN DUY TÙNG – 518H0309**

**Lớp : 18H50203 – 18H50201**

**Khoá : 22**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021**

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM  
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG  
KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CUỐI KỲ  
XỬ LÝ DỮ LIỆU LỚN**

**PROJECT 3**

*Người hướng dẫn:* **TS. BÙI THANH HÙNG**

*Người thực hiện:* **NGUYỄN QUỐC CƯỜNG – 518H0003**

**THÂN DUY TÙNG – 518H0309**

**Lớp : 18H50203 – 18H50201**

**Khoá : 22**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021**

## LỜI CẢM ƠN

Chúng tôi cảm ơn thầy rất nhiều vì đã giảng dạy tận tình, truyền đạt những kiến thức quý báu cho chúng tôi trong suốt thời gian học tập vừa qua. Do chưa có nhiều kinh nghiệm cũng như những hạn chế về kiến thức, trong bài báo cáo sẽ không tránh khỏi những thiếu sót. Rất mong nhận được lời nhận xét, đóng góp ý kiến, phê bình từ thầy để bài báo cáo được hoàn thiện hơn.

Chúng tôi kính chúc thầy nhiều sức khỏe, hạnh phúc và thành công trong công việc cũng như trong cuộc sống.

## **ĐỒ ÁN ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

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

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

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

*TP. Hồ Chí Minh, ngày tháng năm 2021*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Nguyễn Quốc Cường*

*Thân Duy Tùng*

## PHẦN ĐÁNH GIÁ CỦA GIẢNG VIÊN

---

---

---

---

---

---

Tp. Hồ Chí Minh, ngày    tháng    năm  
(kí và ghi họ tên)

## TÓM TẮT

Bài toán phân lớp văn bản là bài toán cơ bản trong khai phá dữ liệu văn bản. Phân lớp văn bản là công việc được sử dụng để hỗ trợ trong quá trình tìm kiếm thông tin, chiết lọc thông tin, lọc văn bản hoặc tự động dẫn đường cho các văn bản tới những chủ đề xác định trước. Hiện nay, bài toán này đã và đang trở thành lĩnh vực nhận được nhiều sự quan tâm, nghiên cứu của nhiều nhà khoa học trên thế giới.

Sử dụng phương pháp 1D-CNN kết hợp model PhoBERT để giải quyết bài toán phân lớp văn bản. Kết quả đạt được nhận thấy các kết quả đạt độ chính xác khá cao. Về phần kết quả của thuật toán học sâu (Deep Learning) là One-dimensional Convolutional Neural Network (1D-CNN) nhìn chung cho kết quả tốt hơn so với 2 thuật toán học máy (Machine Learning) là: Support Vector Machine (SVM) và K-nearest Neighbors (KNN).

# MỤC LỤC

LỜI CẢM ƠN .....	i
PHẦN ĐÁNH GIÁ CỦA GIẢNG VIÊN .....	iii
MỤC LỤC.....	1
DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT .....	3
DANH MỤC HÌNH ẢNH.....	4
DANH MỤC CÁC BẢNG.....	5
1 GIỚI THIỆU BÀI TOÁN .....	6
2 PHÂN TÍCH YÊU CẦU CỦA BÀI TOÁN .....	7
2.1 Yêu cầu của bài toán .....	7
2.2 Các phương pháp giải quyết bài toán.....	7
2.2.1 Phương pháp học máy Support Vector Machine (SVM).....	7
2.2.2 Phương pháp học máy K Nearest Neighbor (KNN) .....	7
2.2.3 Phương pháp học sâu 1D Convolutional Neural Network (1D-CNN) .....	8
2.2.4 Kết hợp mô hình PhoBERT cùng các phương pháp trên.....	9
2.2.4.1 BERT là gì? .....	9
2.2.4.2 RoBERTa là gì?.....	11
2.2.4.3 PhoBERT là gì? .....	11
2.3 Phương pháp đề xuất giải quyết bài toán .....	12
3 PHƯƠNG PHÁP GIẢI QUYẾT BÀI TOÁN.....	13
3.1 Mô hình tổng quát .....	13
3.2 Đặc trưng của mô hình đề xuất .....	15
3.2.1 Thu thập và xử lý dữ liệu. ....	15
3.2.2 Ứng dụng model PhoBERT. ....	16
3.2.3 Tokenize và trích xuất đặc trưng dữ liệu. ....	17
3.2.4 Huấn luyện mô hình bằng các phương pháp: SVM, KNN, 1D-CNN.....	17
4 THỰC NGHIỆM.....	18
4.1 Dữ liệu.....	18
4.2 Xử lý dữ liệu .....	19
4.3 Công nghệ sử dụng.....	19
4.4 Cách đánh giá.....	19
4.4.1 Accuracy .....	19
4.4.2 Mean Squared Error (MSE) .....	20

4.4.3	Root Mean Squared Error (RMSE).....	20
5	KẾT QUẢ ĐẠT ĐƯỢC .....	21
5.1	Tham số thực nghiệm.....	21
5.2	So sánh kết quả đạt được.....	22
5.2.1	Phương pháp đo Accuracy .....	22
5.2.2	Phương pháp đo MSE .....	22
5.2.3	Phương pháp đo RMSE.....	23
6	KẾT LUẬN.....	24
6.1	Kết quả đạt được .....	24
6.2	Hạn chế .....	24
6.1	Hướng phát triển .....	24
	TÀI LIỆU THAM KHẢO.....	25
	TỰ ĐÁNH GIÁ.....	26



## **DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT**

### **CÁC CHỮ VIẾT TẮT**

NLP	Natural Language Processing
CNN	Convolutional Neural Network
MSE	Mean Squared Error
RMSE	Root Mean Squared Error
SVM	Support Vector Machine
KNN	K-Nearest Neighbor

## DANH MỤC HÌNH ẢNH

<i>Hình 2.1 Minh họa SVM (1)</i> .....	7
<i>Hình 2.2 1D-NN classification (Wikipedia)</i> .....	8
<i>Hình 2.3 Mô hình Transformer (5)</i> .....	10
<i>Hình 2.4 Tiến trình pre-training và fine-tuning của BERT (6)</i> .....	11
<i>Hình 3.1 Mô hình của phương pháp SVM (7)</i> .....	13
<i>Hình 3.2 Mô hình của phương pháp KNN (8)</i> .....	14
<i>Hình 3.3 Mô hình của phương pháp 1D-CNN (9)</i> .....	15
<i>Hình 3.4 Scrapy framework</i> .....	15
<i>Hình 3.5 Pretrain model base từ package fairseq (PhoBERT_fairseq)</i> .....	16
<i>Hình 3.6 Nội dung folder PhoBERT_base_fairseq</i> .....	17
<i>Hình 4.1 Dữ liệu được cào</i> .....	18
<i>Hình 4.2 Nội dung từng folder</i> .....	18
<i>Hình 4.3 Nội dung của file text</i> .....	19
<i>Hình 5.1 Chỉ số accuracy trong quá trình huấn luyện</i> .....	21
<i>Hình 5.2 Chỉ số loss trong quá trình huấn luyện</i> .....	21
<i>Hình 5.3 Kết quả phương pháp đo Accuray</i> .....	22
<i>Hình 5.4 Kết quả phương pháp đo MSE</i> .....	22
<i>Hình 5.5 Kết quả phương pháp đo RMSE</i> .....	23

## DANH MỤC CÁC BẢNG

<i>Bảng 5.1 Kết quả chi tiết phương pháp đo Accuracy .....</i>	<i>22</i>
<i>Bảng 5.2 Kết quả chi tiết phương pháp đo MSE .....</i>	<i>23</i>
<i>Bảng 5.3 Kết quả chi tiết phương pháp đo RMSE .....</i>	<i>23</i>

# 1 GIỚI THIỆU BÀI TOÁN

Đề tài: Phân lớp văn bản tiếng Việt bằng mô hình PHO-BERT và các phương pháp học sâu.

Phân lớp văn bản là bài toán cơ bản trong khai phá dữ liệu văn bản. Bài toán phân lớp văn bản là việc gán tên các chủ đề (tên lớp/nhãn lớp) đã được xác định trước, vào các văn bản dựa trên nội dung của chúng.

Phân lớp văn bản là công việc được sử dụng để hỗ trợ trong quá trình tìm kiếm thông tin, chiết lọc thông tin, lọc văn bản hoặc tự động dẫn đường cho các văn bản tới những chủ đề xác định trước. Phân lớp văn bản có thể thực hiện thủ công hoặc tự động sử dụng các kỹ thuật học máy có giám sát.

Các hệ thống phân lớp có thể ứng dụng trong việc phân loại tài liệu của các thư viện điện tử, phân loại văn bản báo chí trên các trang tin điện tử, ... những hệ thống tốt, cho ra kết quả khả quan, giúp ích nhiều cho con người.

Mặt khác, phân lớp văn bản là một trong những thành phần cơ bản nhưng quan trọng nhất trong kiến trúc tổng thể của hầu hết các máy tìm kiếm. Hiện nay, bài toán này đã và đang trở thành lĩnh vực nhận được nhiều sự quan tâm, nghiên cứu của nhiều nhà khoa học trên thế giới.

## 2 PHÂN TÍCH YÊU CẦU CỦA BÀI TOÁN

### 2.1 Yêu cầu của bài toán

Tự cào dữ liệu từ trang web vnexpress.net hay vietnamnet dựa theo các chủ đề. Ví dụ: Thể thao, văn hóa, ....

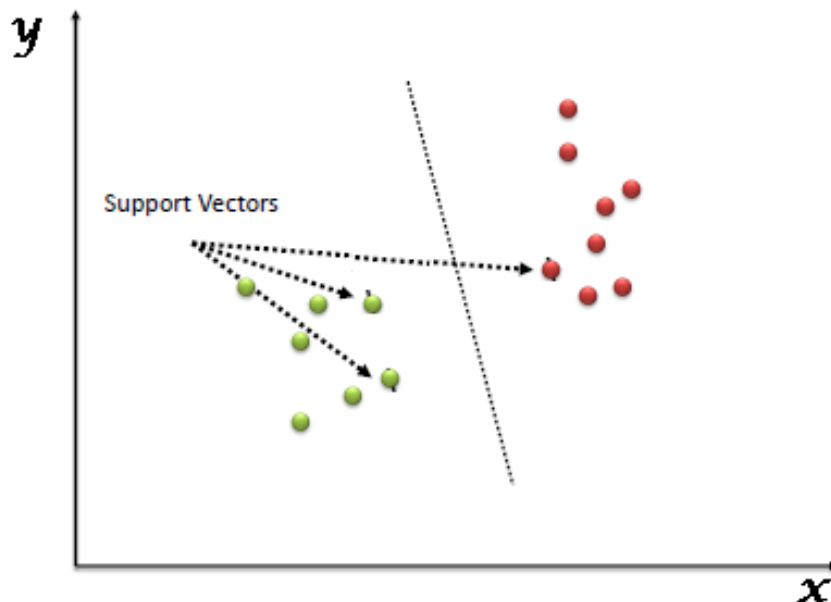
Sử dụng phương pháp học sâu và mô hình PhoBERT để phân lớp tập dữ liệu văn bản cào được.

### 2.2 Các phương pháp giải quyết bài toán

#### 2.2.1 Phương pháp học máy Support Vector Machine (SVM)

SVM là một thuật toán giám sát (supervised-learning), nó có thể sử dụng cho cả việc phân loại (classification) hoặc đệ quy (regression). Tuy nhiên nó được sử dụng chủ yếu cho việc phân loại.

Trong thuật toán này, chúng ta vẽ đồ thị mỗi dữ liệu là một điểm trong  $n$  chiều (ở đây  $n$  là số lượng đặc điểm (feature)) với giá trị của mỗi đặc điểm sẽ là giá trị của một tọa độ (coordinate) cụ thể. Sau đó chúng ta thực hiện tìm "đường bay" (hyper-plane) phân chia các lớp. Hyper-plane nó chỉ hiểu đơn giản là 1 đường thẳng có thể phân chia các lớp ra thành hai phần riêng biệt. (1)



Hình 2.1 Minh họa SVM (1)

#### 2.2.2 Phương pháp học máy K Nearest Neighbor (KNN)

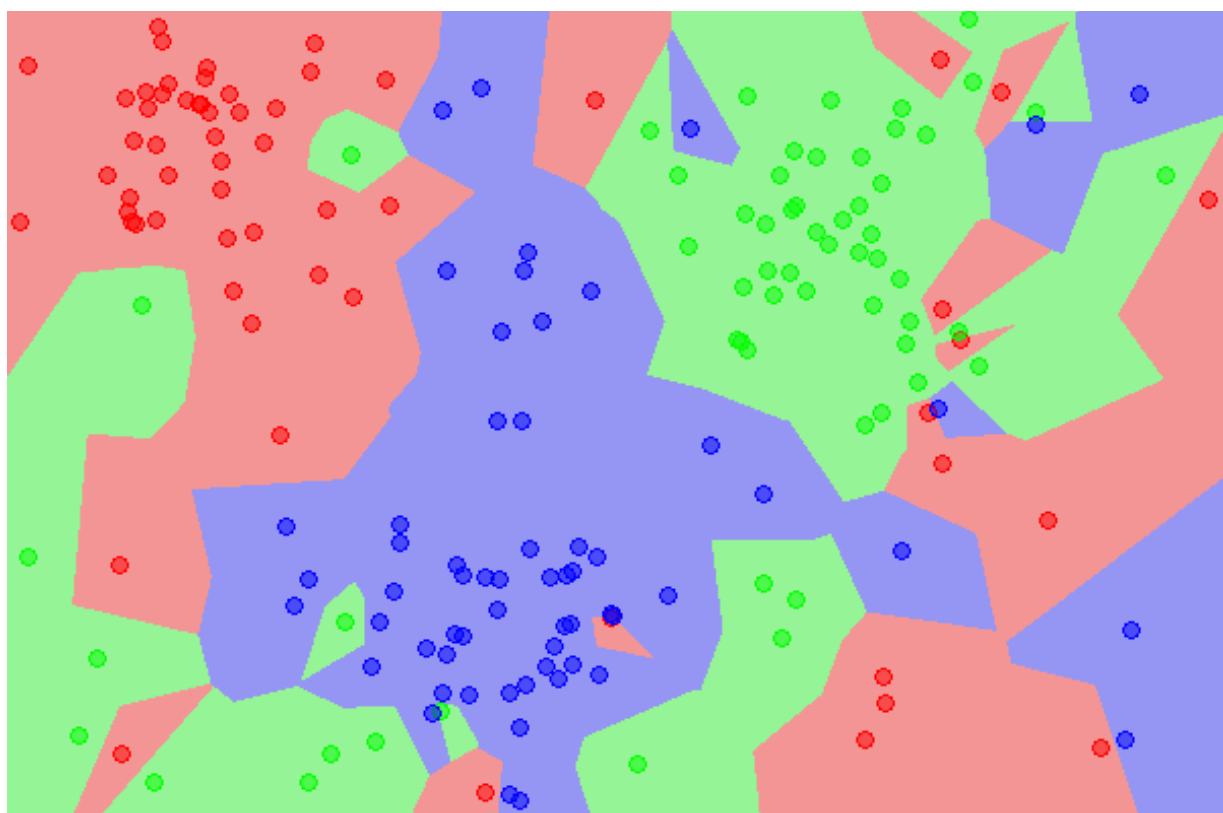
K-nearest neighbor là một trong những thuật toán giám sát (supervised-learning) đơn giản nhất (mà hiệu quả trong một vài trường hợp) trong Machine Learning. Khi training, thuật

toán này không học một điều gì từ dữ liệu training (lazy learning), mọi tính toán được thực hiện khi nó cần dự đoán kết quả của dữ liệu mới.

K-nearest neighbor có thể áp dụng được vào cả hai loại của bài toán Supervised learning là Classification và Regression.

KNN là thuật toán đi tìm đầu ra của một điểm dữ liệu mới bằng cách chỉ dựa trên thông tin của K điểm dữ liệu trong training set gần nó nhất (K-lân cận), không quan tâm đến việc có một vài điểm dữ liệu trong những điểm gần nhất này là nhiều.

Hình dưới đây là một ví dụ về KNN trong classification với  $K = 1$ . (2)



Hình 2.2 1D-NN classification ([Wikipedia](#))

### 2.2.3 Phương pháp học sâu 1D Convolutional Neural Network (1D-CNN)

Tích chập được ứng dụng phổ biến trong lĩnh vực thị giác máy tính (computer vision). Thông qua các phép tích chập, các đặc trưng chính từ ảnh được trích xuất và truyền vào các tầng tích chập (layer convolution). Mỗi một tầng tích chập sẽ bao gồm nhiều đơn vị mà kết quả ở mỗi đơn vị là một phép biến đổi tích chập từ layer trước đó thông qua phép nhân tích chập với bộ lọc.

Để thực hiện phân loại hình ảnh, CNN đi qua mọi góc, vector và kích thước của ma trận pixel. Các lớp tích chập bao gồm nhiều tính năng như phát hiện các cạnh (edge), góc (corner) và nhiều loại kết cấu (multiple textures), làm cho nó trở thành một công cụ đặc biệt để CNN thực hiện mô hình hóa. Lớp này trượt (slide) trên ma trận hình ảnh và có thể phát hiện tất cả

các đặc trưng của nó. Điều này có nghĩa là mỗi lớp tích chập trong mạng có thể phát hiện các đặc trưng phức tạp hơn. Khi đặc trưng này mở rộng, chúng ta cần mở rộng kích thước của lớp tích chập. (3)

Chúng ta có thể coi dữ liệu văn bản là dữ liệu tuần tự (sequential data) như dữ liệu theo chuỗi thời gian (time series), ma trận một chiều. Ý tưởng của mô hình này cũng tương tự như vậy, chỉ thay đổi kiểu dữ liệu và lớp tích chập. Để có thể phân lớp văn bản ta cần một tầng nhúng từ (word embedding layer) và 1D-CNN. (4)

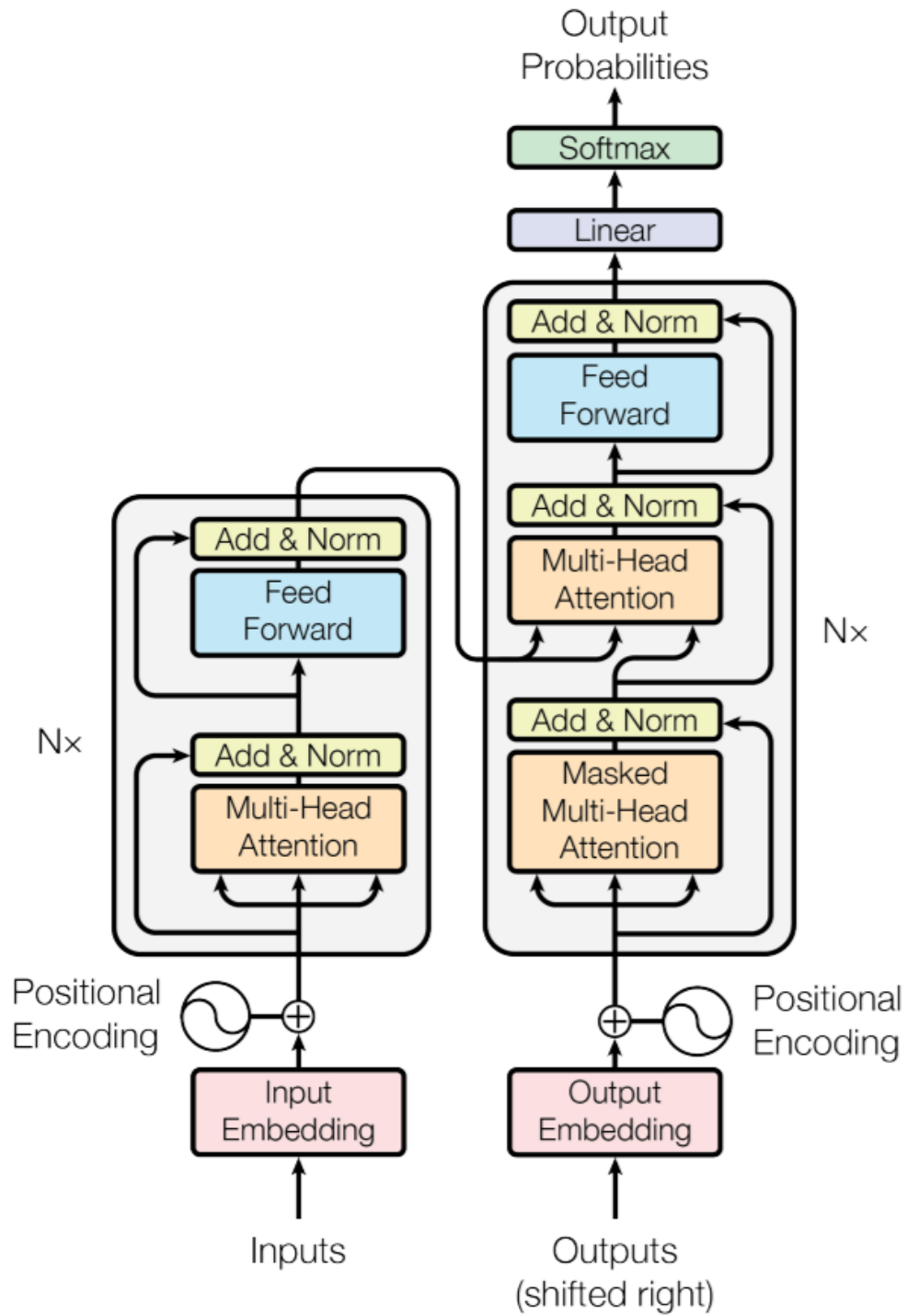
## 2.2.4 Kết hợp mô hình PhoBERT cùng các phương pháp trên

Trước khi đi sâu vào mô hình PhoBERT chúng tôi sẽ giới thiệu một chút về BERT và RoBERTa.

### 2.2.4.1 BERT là gì?

BERT là viết tắt của cụm từ Bidirectional Encoder Representation from Transformer có nghĩa là mô hình biểu diễn từ theo 2 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 2 chiều trái và phải.

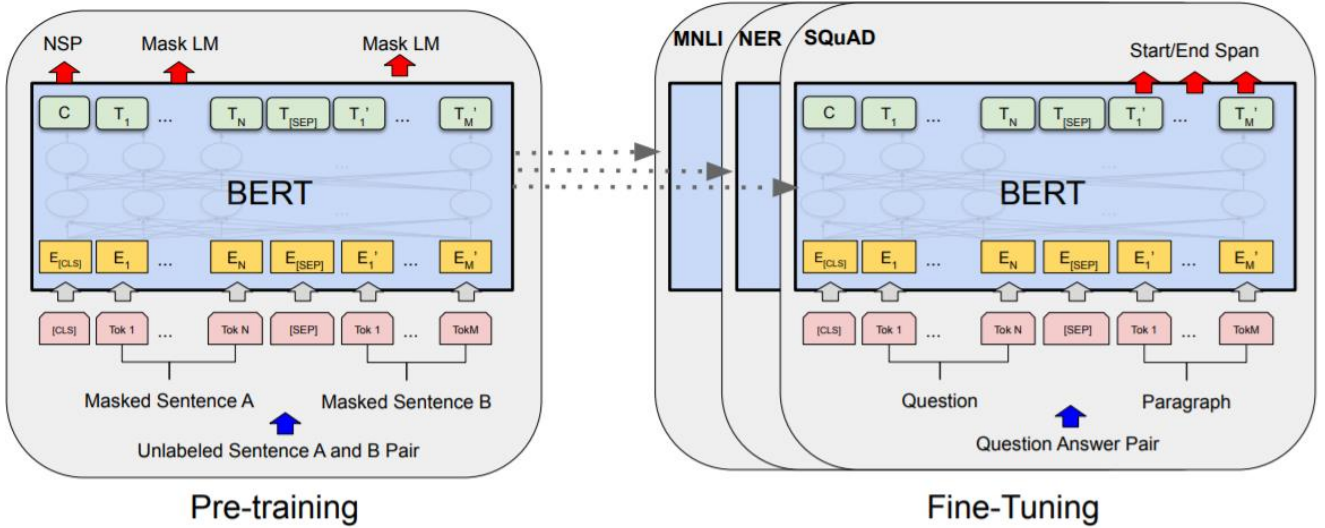
Transformer là một lớp mô hình seq2seq gồm 2 phrase encoder và decoder. Mô hình hoàn toàn không sử dụng các kiến trúc Recurrent Neural Network của RNN mà chỉ sử dụng các layers attention để embedding các từ trong câu.



Hình 2.3 Mô hình Transformer (5)

Một điểm đặc biệt ở BERT mà các model embedding trước đây chưa từng có đó là kết quả huấn luyện có thể fine-tuning được. Chúng ta sẽ thêm vào kiến trúc model một output layer để tùy biến theo tác vụ huấn luyện.





Hình 2.4 Tiến trình pre-training và fine-tuning của BERT (6)

Hiện tại có nhiều phiên bản khác nhau của model BERT. Các phiên bản đều dựa trên việc thay đổi kiến trúc của Transformer tập trung ở 3 tham số:  $L$ : số lượng các block sub-layers trong transformer,  $H$ : kích thước của embedding véc tơ (hay còn gọi là hidden size),  $A$ : Số lượng head trong multi-head layer, mỗi một head sẽ thực hiện một self-attention. Tên gọi của 2 kiến trúc chính bao gồm:

**BERT<sub>BASE</sub>** ( $L = 12, H = 768, A = 12$ ). Tổng tham số 110 triệu.

**BERT<sub>LARGE</sub>** ( $L = 24, H = 1024, A = 16$ ). Tổng tham số 340 triệu.

Như vậy ở kiến trúc BERT Large chúng ta tăng gấp đôi số layer, tăng kích thước hidden size của embedding véc tơ gấp 1.33 lần và tăng số lượng head trong multi-head layer gấp 1.33 lần. (6)

#### 2.2.4.2 RoBERTa là gì?

RoBERTa là một project của facebook kế thừa lại các kiến trúc và thuật toán của model BERT trên framework pytorch. Đây là một project hỗ trợ việc huấn luyện lại các model BERT trên những bộ dữ liệu mới cho các ngôn ngữ khác ngoài một số ngôn ngữ phổ biến.

Ở bài báo nghiên cứu cho biết mặc dù RoBERTa lặp lại các thủ tục huấn luyện từ model BERT, nhưng có một thay đổi đó là huấn luyện mô hình lâu hơn, với batch size lớn hơn và trên nhiều dữ liệu hơn. 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. (7)

#### 2.2.4.3 PhoBERT là gì?

Mô hình pre-trained PhoBERT là mô hình ngôn ngữ hiện đại nhất (SOTA: state-of-the-art) dành cho tiếng Việt. PhoBERT được xây dựng dựa trên mô hình RoBERTa và tối ưu hóa quy trình đào tạo (training procedure) để có hiệu suất mạnh mẽ hơn.

Trong Tiếng Việt thì chúng ta có thể ứng dụng BERT trong một số tác vụ như:

- Tìm từ đồng nghĩa, trái nghĩa, cùng nhóm dựa trên khoảng cách của từ trong không gian biểu diễn đa chiều.
- Xây dựng các véc tơ embedding cho các tác vụ NLP như sentiment analysis, phân loại văn bản, NER, POS, huấn luyện chatbot.
- Gợi ý từ khóa tìm kiếm trong các hệ thống search.
- Xây dựng các ứng dụng seq2seq như robot viết báo, tóm tắt văn bản, sinh câu ngẫu nhiên với ý nghĩa tương đồng, ....

### 2.3 Phương pháp đề xuất giải quyết bài toán

Đầu tiên lấy dữ liệu từ trang báo vnexpress.net là nội dung chính của bài báo. Sau đó tiến đến xử lý dữ liệu.

Tiếp theo, load mô hình PhoBERT để sử dụng.

Tiếp theo, tokenize dữ liệu bằng thuật toán Byte Pair Encoding (BPE) và tiến hành trích xuất đặc trưng (feature extraction).

Cuối cùng tiến hành huấn luyện model kết hợp với các phương pháp ở trên.

Chúng tôi lựa chọn mô hình 1D CNN vì một số lí do như sau:

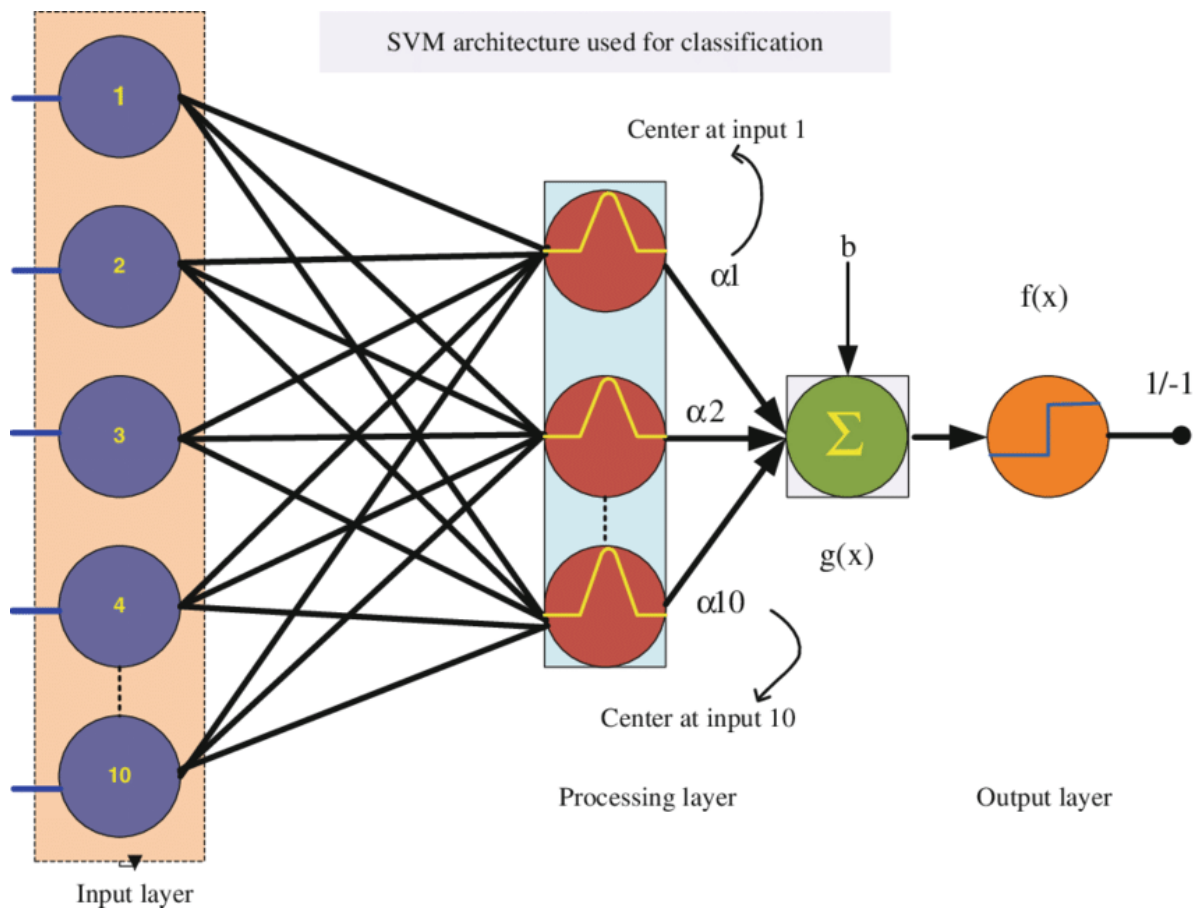
- Chúng tôi tin rằng mô hình 1D CNN sẽ đem lại khả năng phân loại lớp tốt như mô hình 2D CNN trong thị giác máy tính.
- Nó sẽ tốt hơn các mô hình học máy như KNN hay SVM.

### 3 PHƯƠNG PHÁP GIẢI QUYẾT BÀI TOÁN

#### 3.1 Mô hình tổng quát

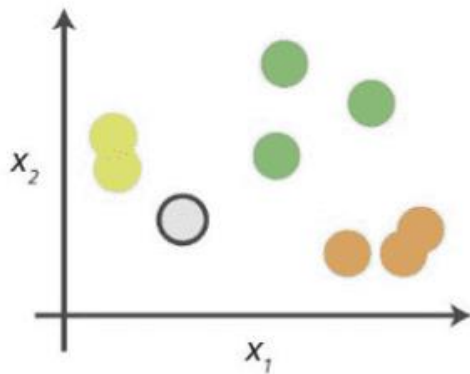
Mô hình tổng quát để giải quyết bài toán gồm các phần sau:

- Phần 1: Thu thập và xử lý dữ liệu.
- Phần 2: Ứng dụng model PhoBERT.
- Phần 3: Tokenize và trích xuất đặc trưng dữ liệu.
- Phần 4: Huấn luyện mô hình bằng các phương pháp: SVM, KNN, 1D-CNN.



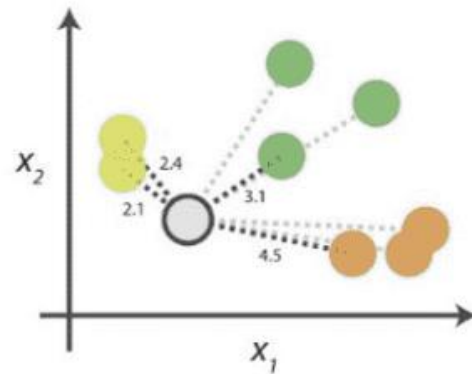
Hình 3.1 Mô hình của phương pháp SVM (7)

## 0. Look at the data



Say you want to classify the grey point into a class. Here, there are three potential classes - lime green, green and orange.

## 1. Calculate distances



Start by calculating the distances between the grey point and all other points.

## 2. Find neighbours

Point Distance			
	...		2.1 → 1st NN
	...		2.4 → 2nd NN
	...		3.1 → 3rd NN
	...		4.5 → 4th NN

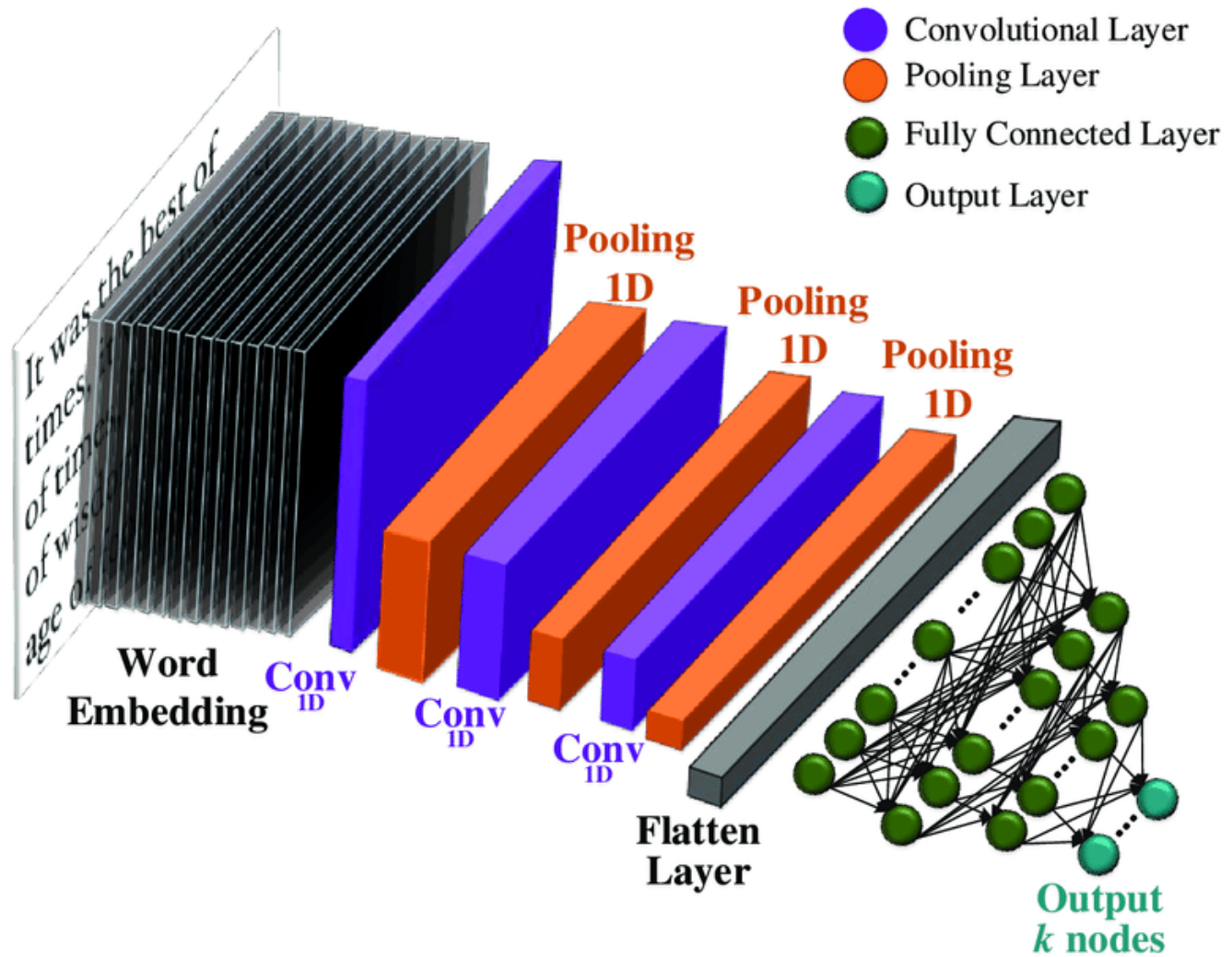
Next, find the nearest neighbours by ranking points by increasing distance. The nearest neighbours (NNs) of the grey point are the ones closest in dataspace.

## 3. Vote on labels

Class	# of votes	
	2	Class  wins the vote! Point  is therefore predicted to be of class .
	1	
	1	

Vote on the predicted class labels based on the classes of the  $k$  nearest neighbours. Here, the labels were predicted based on the  $k=3$  nearest neighbours.

Hình 3.2 Mô hình của phương pháp KNN (8)



Hình 3.3 Mô hình của phương pháp 1D-CNN (9)

## 3.2 Đặc trưng của mô hình đề xuất

### 3.2.1 Thu thập và xử lý dữ liệu.

Sử dụng framework Scrapy để cào dữ liệu từ trang báo điện tử vnexpress.vn.

Chi tiết của dữ liệu và cách xử lý dữ liệu trước khi sử dụng sẽ được trình bày ở phần 4 của đồ án.



Hình 3.4 Scrapy framework

### 3.2.2 Ứng dụng model PhoBERT.

Để áp dụng được model BERT thì trước tiên chúng ta cần phải load được model.

Các dependency packages:

- fairseq: Là project của facebook chuyên hỗ trợ các nghiên cứu và dự án liên quan đến model seq2seq.
- fastBPE: Là package hỗ trợ tokenize từ (word) thành các từ phụ (subwords) theo phương pháp mới nhất được áp dụng cho các pretrain model NLP hiện đại như BERT và các biến thể của BERT.

Tiếp theo là download các model pretrain từ list các pretrain models được liệt kê trong PhoBERT.

Sử dụng pretrain model BERT base được huấn luyện từ package fairseq. Download và giải nén.

#### Pre-trained models

Model	#params	size	Download
PhoBERT-base	135M	1.2GB	<a href="#">PhoBERT_base_fairseq.tar.gz</a>
PhoBERT-large	370M	3.2GB	<a href="#">PhoBERT_large_fairseq.tar.gz</a>

PhoBERT-base :



- `wget https://public.vinai.io/PhoBERT_base_fairseq.tar.gz`
- `tar -xzvf PhoBERT_base_fairseq.tar.gz`

Hình 3.5 Pretrain model base từ package fairseq ([PhoBERT\\_fairseq](#))

Sau khi giải nén pretrain model sẽ kiểm tra thấy bên trong folder sẽ bao gồm 3 files đó là bpe.codes, dict.txt, model.pt có tác dụng như sau:

- bpe.codes: Là BPE token mà mô hình đã áp dụng để mã hóa văn bản sang index.
- dict.txt: Từ điển subword của bộ dữ liệu huấn luyện.
- model.pt: File lưu trữ của mô hình trên pytorch.



 bpe.codes	3/2/2020 11:50 PM	CODES File	1,109 KB
 dict.txt	3/2/2020 11:42 PM	Text Document	875 KB
 model.pt	1/20/2020 10:57 AM	PT File	1,319,016 ...

*Hình 3.6 Nội dung folder PhoBERT\_base\_fairseq*

### 3.2.3 Tokenize và trích xuất đặc trưng dữ liệu.

Tokenize là quá trình mã hóa các văn bản thành các index dạng số mang thông tin của văn bản để cho máy tính có thể huấn luyện được. Khi đó mỗi một từ hoặc ký tự sẽ được đại diện bởi một index.

Trong bài sử dụng thuật toán Byte Pair Encoding (BPE) để tokenize dữ liệu.

BPE là một kỹ thuật nén từ cơ bản giúp chúng ta index được toàn bộ các từ kể cả trường hợp từ mở (không xuất hiện trong từ điển) nhờ mã hóa các từ bằng chuỗi các từ phụ (subwords). Nguyên lý hoạt động của BPE dựa trên phân tích trực quan rằng hầu hết các từ đều có thể phân tích thành các thành phần con.

Phương pháp BPE sẽ thống kê tần suất xuất hiện của các từ phụ cùng nhau và tìm cách gộp chúng lại nếu tần suất xuất hiện của chúng là lớn nhất. Cứ tiếp tục quá trình gộp từ phụ cho tới khi không tồn tại các subword để gộp nữa, ta sẽ thu được tập subwords cho toàn bộ văn bản mà mọi từ đều có thể biểu diễn được thông qua subwords. (10)

Sau đó trích xuất đặc trưng của layer cuối cùng.

### 3.2.4 Huấn luyện mô hình bằng các phương pháp: SVM, KNN, 1D-CNN.

Quá trình huấn luyện một model classification trên pytorch sẽ bao gồm những bước chính sau đây:

- Đưa dữ liệu vào huấn luyện và thẩm định.
- Thiết lập kiến trúc mô hình.
- Optimization giúp tối ưu loss function.
- Huấn luyện mô hình qua các epochs.

## 4 THỰC NGHIỆM

### 4.1 Dữ liệu

Dữ liệu được lấy từ trang báo điện tử vnexpress.net.

Dữ liệu bao gồm 10000 file text được sắp xếp thành 10 folder (10 topics). Mỗi folder tương ứng với mỗi chủ đề (topic) trong đó có 1000 file text.

Name	Date modified	Type	Size
doi-song	12/12/2021 7:52 PM	File folder	
du-lich	12/12/2021 7:45 PM	File folder	
giao-duc	12/12/2021 7:43 PM	File folder	
khoa-hoc	12/12/2021 7:41 PM	File folder	
kinh-doanh	12/12/2021 7:47 PM	File folder	
oto-xe-may	12/12/2021 7:50 PM	File folder	
phap-luat	12/12/2021 7:48 PM	File folder	
so-hoa	12/12/2021 7:37 PM	File folder	
suc-khoe	12/12/2021 7:53 PM	File folder	
the-thao	12/12/2021 7:40 PM	File folder	

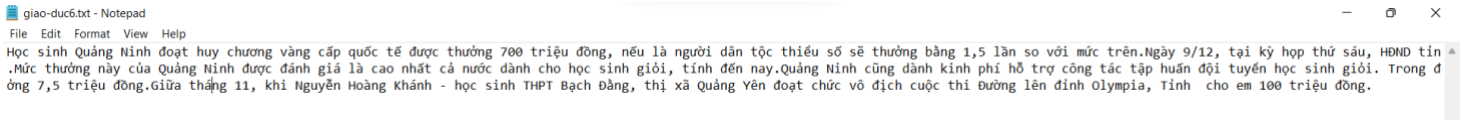
Hình 4.1 Dữ liệu được cào

Name	Date modified	Type	Size
doi-song1.txt	12/12/2021 7:50 PM	Text Document	1 KB
doi-song2.txt	12/12/2021 7:50 PM	Text Document	1 KB
doi-song3.txt	12/12/2021 7:50 PM	Text Document	7 KB
doi-song4.txt	12/12/2021 7:50 PM	Text Document	7 KB
doi-song5.txt	12/12/2021 7:50 PM	Text Document	3 KB
doi-song6.txt	12/12/2021 7:50 PM	Text Document	3 KB
doi-song7.txt	12/12/2021 7:50 PM	Text Document	3 KB
doi-song8.txt	12/12/2021 7:50 PM	Text Document	8 KB
doi-song9.txt	12/12/2021 7:50 PM	Text Document	5 KB
doi-song10.txt	12/12/2021 7:50 PM	Text Document	1 KB

Hình 4.2 Nội dung từng folder



Nội dung của từng file text tương đương với nội dung của từng bài báo trên vnexpress.



Hình 4.3 Nội dung của file text

## 4.2 Xử lý dữ liệu

Dữ liệu cần phải được tiền xử lý. Tại vì giúp chuẩn hóa dữ liệu, loại bỏ đi các thành phần không có ý nghĩa vừa để đảm bảo ý nghĩa của từ được toàn vẹn, vừa để không ảnh hưởng tới quá trình embedding, cũng như những tính chất đặc trưng của dữ liệu.

Dữ liệu được xử lý như sau:

- Chuẩn hóa bảng mã Unicode.
- Thay thế các dấu ngoặc kép (""") thành những khoảng trắng (whitespace).
- Xóa các ký tự đặc biệt: ".", ",", ";", ")", ....
- Chuyển tất cả dữ liệu về dạng chữ thường (lowercase).
- Tách từ (words segmentation) sử dụng thư viện underthesea.
- Loại bỏ các stopwords.

## 4.3 Công nghệ sử dụng

Ngôn ngữ lập trình	Python 3
Thư viện	Torch, Sklearn, Fairseq, Tensorflow, Underthesea, Numpy, Tqdm, Re, Pickle, Matplotlib, Seaborn, Glob2
Môi trường	Google Colab

## 4.4 Cách đánh giá

Trong bài báo cáo này chúng tôi sử dụng 3 độ đo chính: Accuracy, Mean Squared Error (MSE), Root Mean Squared Error (RMSE).

### 4.4.1 Accuracy

Phép đo Accuracy tính mức độ giống nhau của phần dự đoán (predictions) và phần nhãn (labels). Có giá trị trả về từ 0 đến 1.

Công thức tính: Tổng số dự đoán chính xác chia cho số lượng bản ghi (records).

#### 4.4.2 Mean Squared Error (MSE)

Mean Squared Error (MSE) hay tiếng Việt được gọi là sai số toàn phương trung bình là một phép tính tính trung bình của bình phương các sai số, tức là sự khác biệt giữa các ước lượng và những gì được đánh giá.

Công thức tính:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (4.1)$$

#### 4.4.3 Root Mean Squared Error (RMSE)

Root Mean Squared Error (RMSE) như tên gọi của nó chỉ đơn giản là căn bậc hai của phép đo MSE. RMSE cho biết các dự đoán sẽ giảm bao xa so với các giá trị thực đo được bằng cách sử dụng khoảng cách Euclide. Là quy tắc tính điểm bậc hai đo mức độ lỗi trung bình.

Công thức tính:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (4.2)$$

## 5 KẾT QUẢ ĐẠT ĐƯỢC

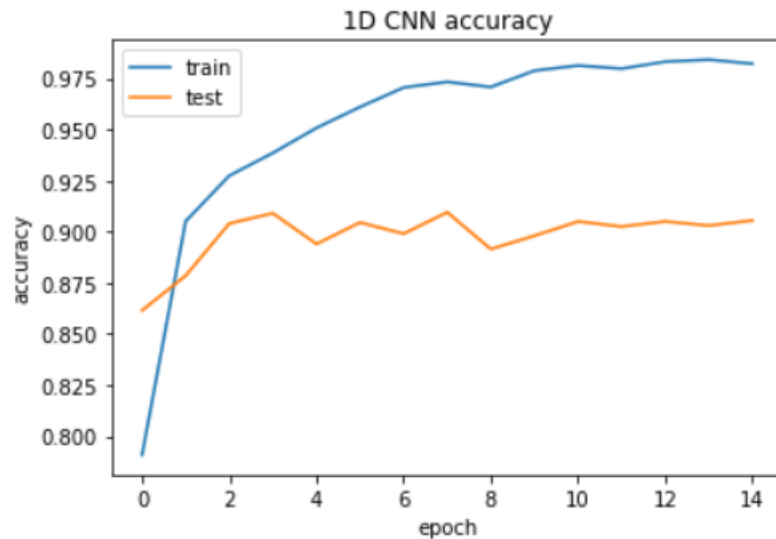
### 5.1 Tham số thực nghiệm

Tiểu luận sẽ tiến hành thực nghiệm trên tập dữ liệu các bài báo như đã trình bày ở phần trước. Tập dữ liệu sẽ được chia ra làm 8 phần cho huấn luyện và 2 phần để kiểm thử kết quả của mô hình.

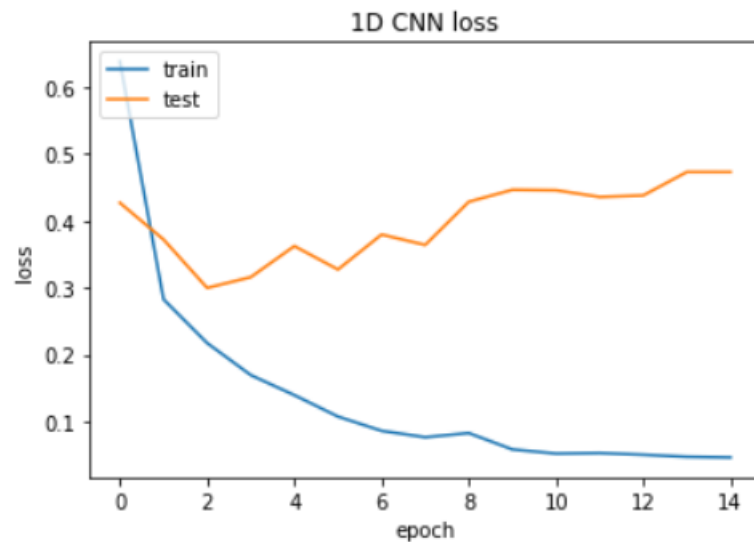
Tiểu luận sẽ sử dụng mô hình 1D-CNN với 2 layers Conv1D, 1 layer Dropout, 1 layer MaxPooling1D và cuối cùng là layer FullyConnected sử dụng hàm Sigmoid.

Hàm Loss tiểu luận sẽ sử dụng hàm Sparse Categorical Cross Entropy. Thuật toán tối ưu tiểu luận sẽ sử dụng thuật toán Adam.

Tiểu luận sẽ sử dụng số Epochs là 15.



Hình 5.1 Chỉ số accuracy trong quá trình huấn luyện

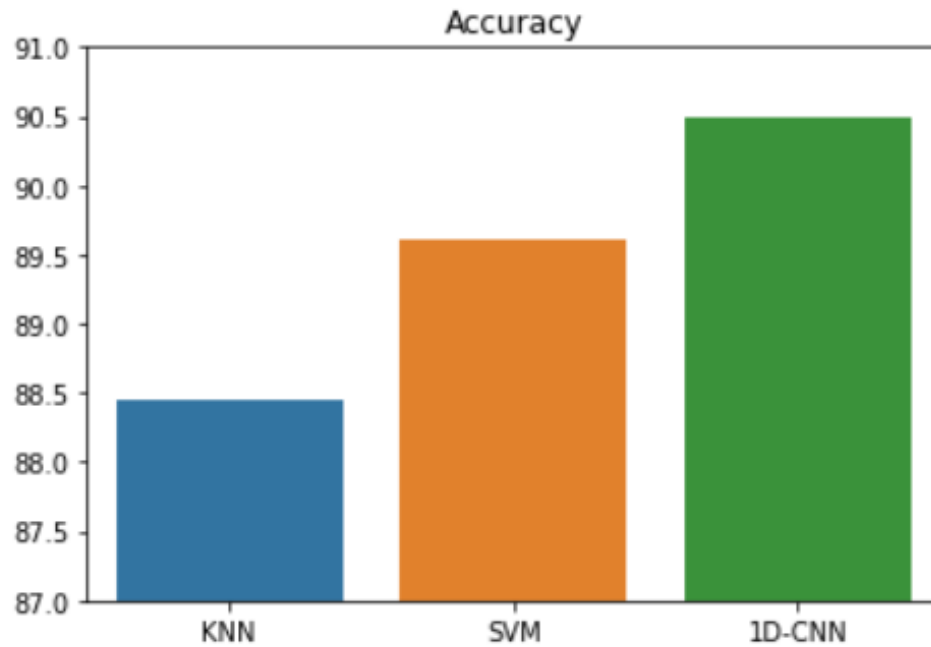


Hình 5.2 Chỉ số loss trong quá trình huấn luyện

## 5.2 So sánh kết quả đạt được

So sánh kết quả đạt được dựa trên 3 thuật toán: SVM, KNN và 1D-CNN.

### 5.2.1 Phương pháp đo Accuracy



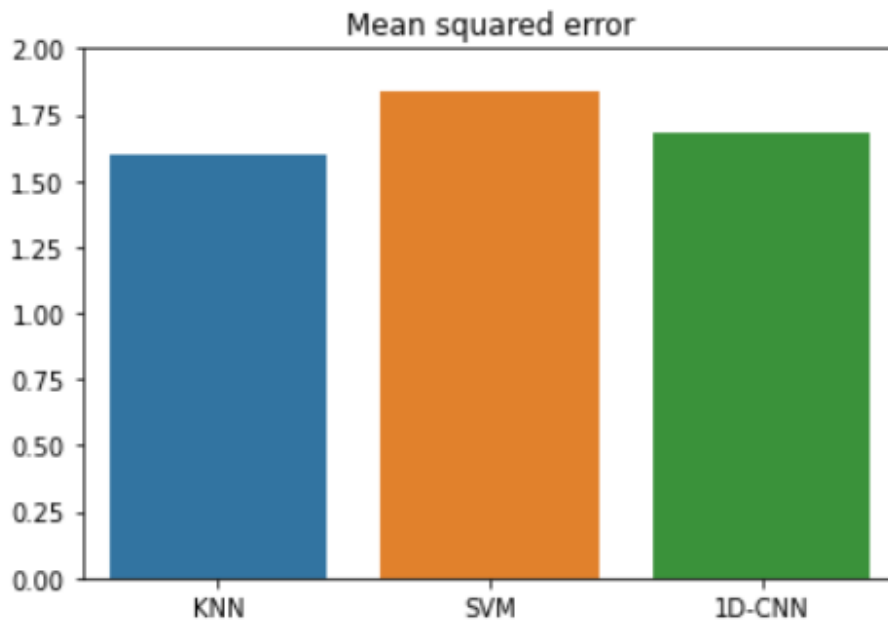
Hình 5.3 Kết quả phương pháp đo Accuracy

Chi tiết:

Phương pháp đo	SVM	KNN	1D-CNN
Accuracy	0.896	0.884	0.949

Bảng 5.1 Kết quả chi tiết phương pháp đo Accuracy

### 5.2.2 Phương pháp đo MSE



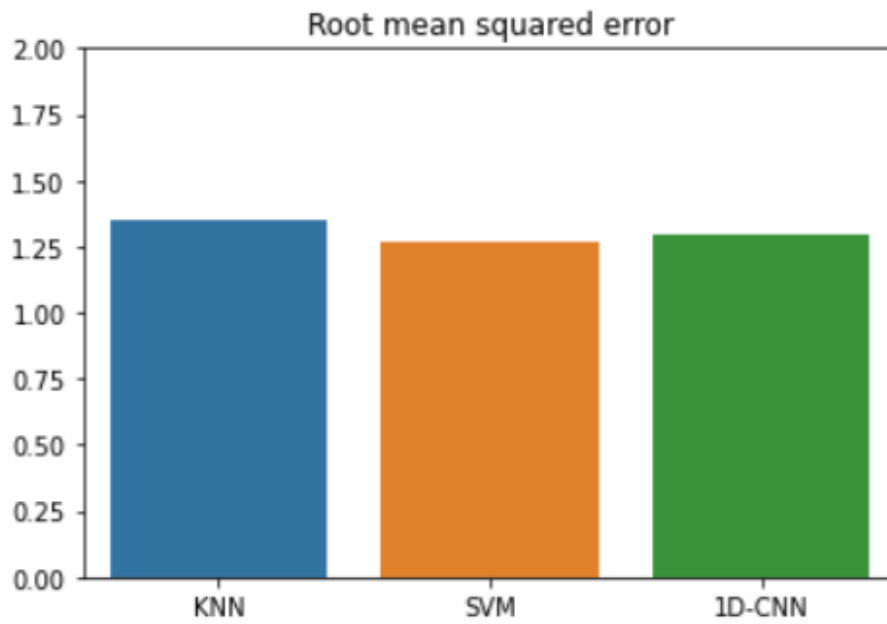
Hình 5.4 Kết quả phương pháp đo MSE

Chi tiết:

Phương pháp đo	SVM	KNN	1D-CNN
MSE	1.602	1.834	1.6795

*Bảng 5.2 Kết quả chi tiết phương pháp đo MSE*

### 5.2.3 Phương pháp đo RMSE



*Hình 5.5 Kết quả phương pháp đo RMSE*

Chi tiết:

Phương pháp đo	SVM	KNN	1D-CNN
RMSE	1.265	1.354	1.295

*Bảng 5.3 Kết quả chi tiết phương pháp đo RMSE*

## 6 KẾT LUẬN

### 6.1 Kết quả đạt được

Về mặt lý thuyết, tiểu luận đã tìm hiểu về các phương pháp giải quyết bài toán phân lớp văn bản trong tiếng Việt, đồng thời tiểu luận sử dụng phương pháp học sâu 1D-CNN kết hợp với PhoBERT để giải quyết bài toán.

Về mặt thực nghiệm, tiểu luận đã sử dụng tập dữ liệu là các bài báo trên vnexpress.net cho mô hình cùng với 3 thuật toán SVM, KNN và 1D-CNN để so sánh. Kết quả thực nghiệm cho thấy thuật toán 1D-CNN mang lại kết quả chung tốt hơn so với các 2 thuật toán còn lại.

### 6.2 Hạn chế

Sau khi thực hiện mô hình 1D-CNN để giải quyết bài toán trên thì chúng tôi đưa ra một số hạn chế của mô hình này:

- Mô hình 1D-CNN chưa hề đưa ra được kết quả nổi trội hơn so với các mô hình Machine Learning cơ bản như SVM.
- Bị ảnh hưởng bởi phép tích chập, bởi sau mỗi lần thực hiện phép nhân tích chập, kích thước của dữ liệu sẽ bị giảm xuống và chỉ có thể thực hiện được vài lần trước khi kích thước của dữ liệu bị giảm xuống quá nhỏ.
- Ngoài ra, ở giữa chuỗi dữ liệu 1D, sẽ được bao phủ bởi rất nhiều kernel, trong khi ở 2 đầu của chuỗi dữ liệu chỉ được kernel quét qua 1 hoặc 2 lần và điều này có thể dẫn đến việc thất thoát thông tin (quan trọng) trong quá trình huấn luyện.
- Tốc độ huấn luyện của mô hình 1D CNN khá chậm so với những mô hình học sâu khác do các lớp như maxpool.

### 6.1 Hướng phát triển

Hiện nay mảng NLP trên thế giới đã rất phát triển đặc biệt là ở tiếng Anh và tiếng Trung Quốc, nhưng vẫn còn chưa được phổ biến rộng rãi ở Việt Nam, chính vì vậy chúng tôi sẽ tiếp tục nghiên cứu và phát triển về vấn đề này trong tương lai:

- Thực hiện trên một số mô hình học sâu tốt hơn như LTSM, RCNN.
- Xây dựng trợ lý ảo dành cho tiếng Việt tương tự như Siri hay “Ok Google”, điều này có thể ứng dụng trong nhiều lĩnh vực khác nhau như tư vấn sản phẩm, trả lời câu hỏi tự động, IOT, .... Điều này làm giảm bớt khối lượng công việc con người cần phải làm trong tương lai.

## TÀI LIỆU THAM KHẢO

1. **Ray, Sunil.** *Analytics Vidhya*. [Online] 9 2017.  
<https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/>.
2. **Tiếp, Vũ Hữu.** *machinelearningcoban.com*. [Online] 1 2017.  
<https://machinelearningcoban.com/2017/01/08/knn/>.
3. **Pham, Khanh Dinh.** *phamdinhkhanh.github.io*. [Online] 8 2019.  
<https://phamdinhkhanh.github.io/2019/08/22/convolutional-neural-network.html>.
4. **Verma, Yugesh.** *analyticsindiamag.com*. [Online] 7 2021. <https://analyticsindiamag.com/guide-to-text-classification-using-textcnn/>.
5. *Attention Is All You Need*. **Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin.** 2017, arXiv preprint arXiv: 1706.03762.
6. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. **Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova.** 2019, arXiv preprint arXiv: 1810.04805.
7. *Enhancing SVM performance in intrusion detection using optimal feature*. **Iftikhar Ahmad, Muhammad Hussain, Abdullah Alghamdi, Abdulhameed Alelaiwi.** s.l. : ResearchGate, 2013.
8. **Christopher, Antony.** *medium.com*. [Online] <https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4>.
9. *Text Classification Algorithms: A Survey*. **Kowsari, Kamran, et al.** 2019, MDPI - Multidisciplinary Digital Publishing Institute, p. 38.
10. *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. **Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov.** 2019, arXiv preprint arXiv: 1907.11692.

## TỰ ĐÁNH GIÁ

Câu	Nội dung	Điểm chuẩn	Tự chấm	Ghi chú
1 (8.5)	1.1 Giới thiệu về bài toán	0.5	0.5	
	1.2 Phân tích yêu cầu của bài toán	1.0	1.0	
	1.3 Phương pháp giải quyết bài toán	1.5	1.25	
	1.4 Thực nghiệm	4	3.5	
	1.5 Kết quả đạt được	1	1	
	1.6 Kết luận	0.5	0.25	
2	Điểm nhóm	0.5	0.5	
3	Báo cáo	1.0	1.0	
<b>Tổng điểm</b>			9	