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

ĐỒ ÁN CUỐI KÌ

XỬ LÝ DỮ LIỆU LỚN

Project 6

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

Người thực hiện: **TRẦN PHÚC CHƯƠNG – 518H0144**

ĐINH HỒNG HÀ – 518H0171

Lớp **: 18H50205**

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 KÌ

XỬ LÝ DỮ LIỆU LỚN

Project 6

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

Người thực hiện: **TRẦN PHÚC CHƯƠNG – 518H0144**

ĐINH HỒNG HÀ – 518H0171

Lớp **: 18H50205**

Khoá **: 22**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021

LỜI CẢM ƠN

Nhóm em cảm ơn thầy Bùi Thanh Hùng rất nhiều vì đã giảng dạy tận tình, truyền đạt những kiến thức quý báu cho tụi em 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.

Nhóm em 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 / chúng tôi và được sự hướng dẫn của TS 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 23 tháng 12 năm 2021

Tác giả

(ký tên và ghi rõ họ tên)

Trần Phúc Chương

Đinh Hồng Hà

PHẦN ĐÁNH GIÁ CỦA GIẢNG VIÊN

_____

_____

_____

_____

_____

_____

_____

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

TÓM TẮT

Hiện nay với sự phát triển và bùng nổ ngành Công nghệ thông tin, Intenet chứa một lượng dữ liệu khổng lồ, cho nên vai trò của hệ thống khai thác thông tin trở nên rất quan trọng. Relation Extraction là một trong các nhiệm vụ của Information Extraction, nó tập trung vào việc phân loại các mối quan hệ giữa các cặp NE được đề cập trong văn bản.

Có rất nhiều phương pháp chiết xuất mới hiện nay, nó nhận được nhiều sự quan tâm từ các nhà nghiên cứu trong ngôn ngữ nói chung và tiếng Việt nói riêng. Theo thống kê thì các mô hình dựa tên BERT (Bidirectional Encoder Representations from Transformers) đã đạt được nhiều thành công trong và trở thành một xu hướng và sử dụng rộng rãi và được biệt là BERT đã được ứng dụng cho Tiếng Việt.

Trong bài báo cáo này, chúng tôi sẽ trình bày cách tiếp cận về cách áp dụng mô hình dựa trên BERT để trích xuất mối quan hệ nhiệm vụ chung của chiến dịch VLSP 2020. Về chi tiết, chúng tôi trình bày: (1) ý tưởng giải quyết nhiệm vụ này; (2) cách xử lý trước dữ liệu phù hợp sao cho có thể mang lại kết quả tốt nhất có thể; (3) cách sử dụng mô hình dựa trên BERT cho nhiệm vụ trích xuất quan hệ; và (4) kết quả thu được dựa trên dữ liệu của tổ chúc VLSP 2020.

MỤC LỤC

DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT

CÁC CHỮ VIẾT TẮT

NLP : Natural Language Processing

IE : Information Extraction

RE : Relation Extraction

NE : Named Entities

DANH MỤC CÁC HÌNH VẼ

DANH MỤC CÁC BẢNG

TRÍCH XUẤT MỐI QUAN HỆ TRONG TIẾNG VIỆT

1.1 Giới thiệu về bài toán

Ngày nay, xử lý ngôn ngữ tự nhiên (NLP) là một lĩnh vực nghiên cứu rất thú vị và cần thiết. Kết quả trong lĩnh vực xử lý ngôn ngữ tự nhiên có thể mang lại nhiều lợi ích cho con người, có thể giúp mọi người rất nhiều trong việc tự động hóa các tác vụ xử lý văn bản. Tuy nhiên, so với các ngôn ngữ phổ biến khác như tiếng Anh, tiếng Trung…, kết quả thu được đối với trích xuất trong hệ trong Tiếng Việt vẫn còn rất nhiều hạn chế.

Trong một cuộc hội thảo quốc tết về xử lí giọng nói, đặc biệt ngôn ngữ là Tiếng Việt, đây là lần đầu tiên có một nhiệm vụ chung về chiết tách quan hệ trong Tiếng Việt. Điều này thực sự tuyệt vời vì nó có nghĩa là Khai thác mối quan hệ bằng tiếng Việt đang được cộng đồng nghiên cứu và ngành công nghiệp quan tâm nhiều hơn.

Khai thác các mối quan hệ trong chiến dịch VLSP 2020, các nhà tổ chức sẽ phát hành Trainning, Development và Test dữ liệu.

- Dữ liệu Trainning and Development bao gồm là các tờ báo điện tử Việt Nam được gắn nhãn theo 3 loại NE (Locations, Organizations, and Persons) được đề cập trong các bài báo và các mối quan hệ được gắn nhãn trong các loại NE phải thuộc cùng một câu.
- Dữ liệu Test cũng chứa các thông tin tương tự như Training và Development. Dữ liệu Test này sẽ dự đoán nhãn mối quan hệ giữa các NE trong câu.

Tiếp theo, chúng tôi mô tả chi tiết dataset trong VLSP 2020 RE, cách xử lí dữ liệu và về cách áp dụng mô hình dựa trên BERT.

1.2 Phân tích yêu cầu của bài toán

1.2.1 Yêu cầu của bài toán và dữ liệu

Bài toán trích xuất mối quan hệ được đề xuất để làm nền tảng cho việc xử lý các tài liệu một cách thông minh bằng việc giải quyết một trong những bài toán cơ bản của trích xuất thông tin.

Trích xuất mối quan hệ. Bài toán này tập trung vào việc phân loại các cặp thực thể (NE) trong văn bản tin tức tiếng Việt thành bốn loại khác nhau không trùng lặp với các quan hệ ngữ nghĩa đã được xác định trước.

Bài toán này chỉ tập trung vào việc trích xuất quan hệ trong cùng một câu, tức là giới hạn quan hệ với các quan hệ được thể hiện duy nhất trong một câu đó. Mối quan hệ giữa những lần đề cập đến thực thể nó sẽ được chú thích khi mà mối quan hệ đó được tham chiếu một cách rõ ràng trong câu và phải chứa hai lượt đề cập. Ngay cả khi nó có mối quan hệ trên thực tế, thì bắt buộc vẫn phải có bằng chứng chứng minh cho mối quan hệ đó trong ngữ cảnh cục bộ nơi mà nó được gắn thẻ.

Dữ liệu : Bộ dữ liệu ((training, development and test)) đã được tái sử dụng và phát triển từ nhiệm vụ VLSP-2018 (VNER 2018), được thu thập từ các báo điện tử đăng trên web. Nó được chú thích với ba loại thực thể (NE): Locations (LOC), Organizations (ORG) và Persons (PER), và bốn loại mối quan hệ giữa các NE. Các kiểu quan hệ này được mô tả trong bảng 1.2.1 bên dưới

No	Relation	Arguments	Directionality
1	LOCATED	PER – LOC, ORG – LOC	Directed
2	PART–WHOLE	LOC – LOC, ORG – ORG, ORG-LOC	Directed
3	PERSONAL–SOCIAL	PER – PER	Undirected
4	ORGANIZATION–AFFILIATION	PER–ORG, PER-LOC, ORG–ORG, LOC-ORG	Directed

Bảng 1.2. 1 Bảng mô tả dữ liệu

1.2.2 Các phương pháp giải quyết bài toán

1.2.2.1 Phương pháp PhoBert kết hợp với XML-RoBERTa

Các nghiên cứu của bài toán này tập trung vào các mô hình BERT-based. Các mô hình này đã đạt được những thành tựu cao trong các nghiên cứu NLP. Vì vậy nó trở thành xu hướng và được sử dụng rộng rãi cho rất nhiều nghiên cứu về NLP.

Về ý tưởng :

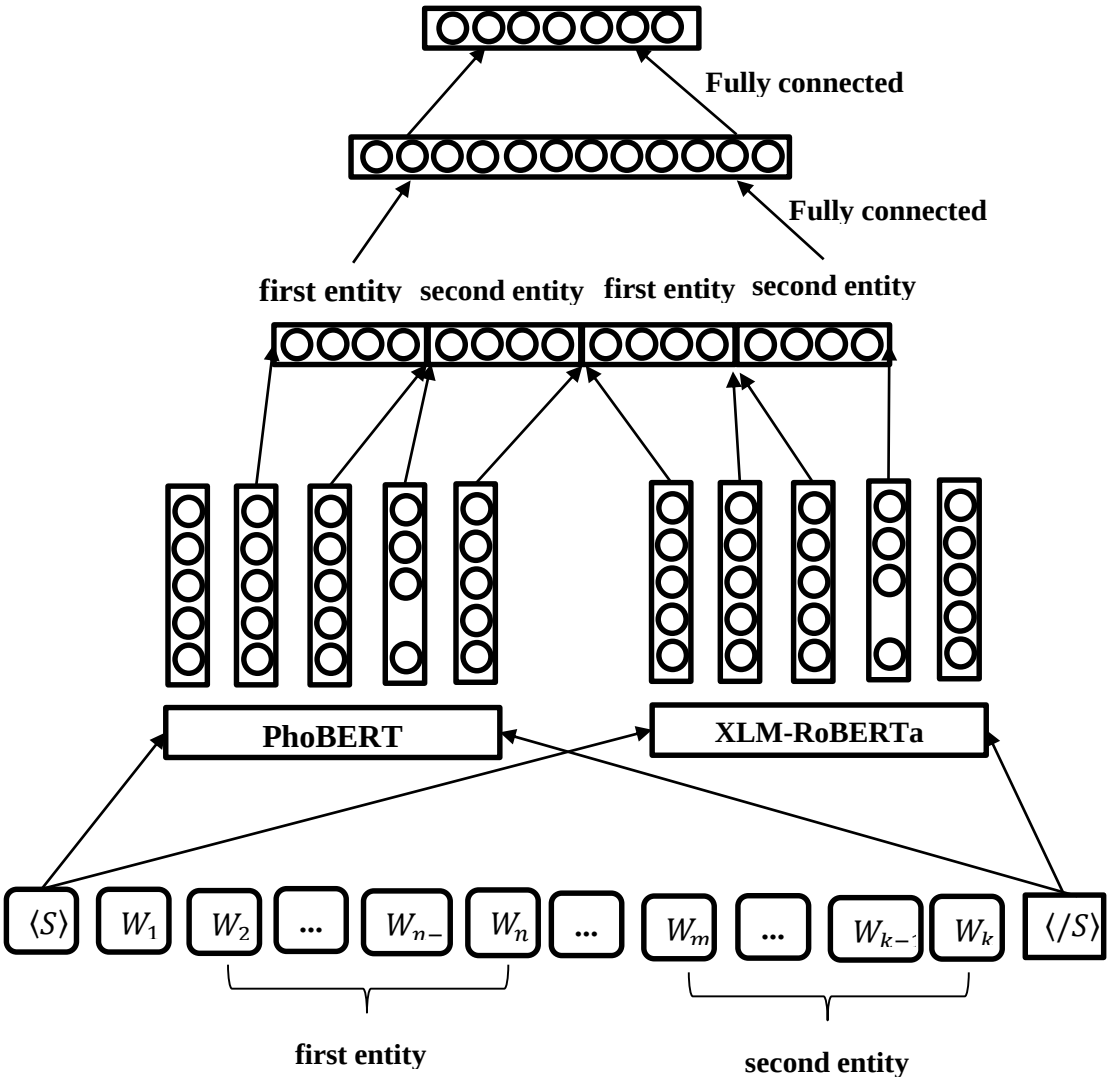
- Trước tiên cần chia tài liệu thô ban đầu theo từng câu vì tập dữ liệu chỉ chứa các mối quan hệ được gắn nhãn trước giữa các thực thể (NE) thuộc cùng một câu.
- Giả sử có tổng n các NE trong một câu, thì chúng ta tạo ra $\frac{n(n-1)}{2}$ câu tương ứng với $\frac{n(n-1)}{2}$ cặp NE. Mỗi câu này là một điểm dữ liệu được chuyển tới mô hình BERT-based sau này. Nhãn cho mỗi điểm dữ liệu là nhãn quan hệ giữa cặp NE trong câu đó.
- Có bốn loại quan hệ. Ba trong số chúng là có hướng (directed), vì vậy cần tạo hai quan hệ vô hướng (undirected) thì mới cho mỗi quan hệ có hướng, nó còn tùy thuộc vào việc nhãn quan hệ có hướng đứng trước hay sau của các cặp NE trong câu.
- Để nắm rõ hơn, dưới đây là một vài ví dụ. Các ví dụ này trích từ những bài báo "KINH TẾ" thuộc bộ dữ liệu VLSP2018.
    - VÍ DỤ 1: Trong câu: "Hà Nội là thủ đô của Việt Nam", dựa vào bảng 1.1 hai thực thể ("Hà Nội" và "Việt Nam") đều nhãn là Locations (LOC) suy ra mối quan hệ giữa cặp NE trên là "PART-WHOLE".
    - VÍ DỤ 2: Trong câu: "Đào Minh Tú – Phó Thống đốc Ngân hàng Nhà nước", dựa vào bảng 1 hai thực thể "Đào Minh Tú" có nhãn là Persons (PER) và "Ngân hàng Nhà nước" có nhãn là Organizations (ORG) suy ra mối quan hệ giữa cặp NE trên là "ORGANIZATION–AFFILIATION".

Các bước Tiền xử lí dữ liệu :

- Trước tiên xóa các ký tự không phải là chữ và số ở đầu hoặc cuối thực thể (NE).
- Tiếp đến sử dụng thư viện Underthesea để chia tài liệu thô thành các câu và tạo phân đoạn từ cho câu
- Đôi khi, Undethesea không chia tài liệu thô bằng một số ký tự ở cuối câu như dấu chấm, dấu ba chấm, ... Vì vậy, chúng tôi tìm thấy những câu bị lỗi này và sửa lại bằng cách sử dụng một số quy tắc.
- Khắc phục sự cố với phân đoạn từ bị lỗi của Underthesea để khớp với các NE…
- Bên cạnh đó, có thể thực hiện một số bước tiền xử lý khác như: Kiểm tra và sửa nếu có mối liên hệ giữa các thực thể thuộc các câu khác nhau,… để đảm bảo dữ liệu trích xuất từ dữ liệu thô là chính xác

Model	**Micro-averaged F-score**
Model 1	0.9323
Model 2	0.9310
Model 3	0.9309

Bảng 1.2. 2 Hiệu suất của các mô hình trên tập phát triển



Hình 1.2. 1 Sơ đồ tổng quát PhoBERT và XLM-RoBERTa

Như hình 1.2. 1 trên, chúng tôi sử dụng hai mô hình BERT-based hỗ trợ tiếng Việt: PhoBERT (PB) và XLM-RoBERTa (XLMR).

Về chi tiết, chúng tôi làm theo các bước sau để xử lý câu:

- Chuyển các câu vào các mô hình BERT-based để tạo ra các vectơ nhúng cho từng cặp NE của mỗi câu. Sử dụng cả hai mẫu BERT-base PB và XLMR; và chỉ sử dụng PB hoặc chỉ XLMR.

- Đặc biệt, mỗi NE có thể có nhiều mảnh ghép từ. Vì vậy, sử dụng và kết hợp các phép nhúng của nó từ các lớp BERT khác nhau thành một vectơ nhúng duy nhất cho đoạn từ đó.
- Sau đó, với mỗi NE, thực hiện quy trình tương tự như vậy để tạo một vectơ nhúng duy nhất từ các vectơ nhúng các mảnh từ của nó.
- Mỗi câu có hai thực thể, vì vậy có hai vectơ nhúng. Đặt vectơ đầu tiên là h1; vectơ nhúng thứ hai là h2. Từ hai vectơ này, tạo ra một vectơ nhúng duy nhất cho câu hiện tại: [h1, h2].

Kết quả đạt được :

- Khi sử dụng cả hai models (PB và XLMR) và chỉ một trong hai models này (PB hoặc XLMR) và có thể nhận thấy rằng kết quả khi việc sử dụng cả hai models thì tốt hơn nhiều. Chi tiết về kết quả được trình bày trong Bảng 1.2. 2

1.2.2.2 Phương pháp R-BERT và BERT-ES

Phương pháp nghiên cứu này sử dụng pre-trained BERT-models để thực hiện việc trích xuất các mối quan hệ trong chiến dịch VLSP 2020.

Áp dụng hai mô hình BERT-models hiện đại đó là R-BERT và BERT-ES . So sánh hai mô hình pre-trained BERT-models: FPTAI/vibert and NlpHUST/vibert4news. Và nhận thấy rằng mô hình NlpHUST/vibert news tốt hơn đáng kể so với FPTAI/vibert về trích xuất quan hệ trong Tiếng Việt

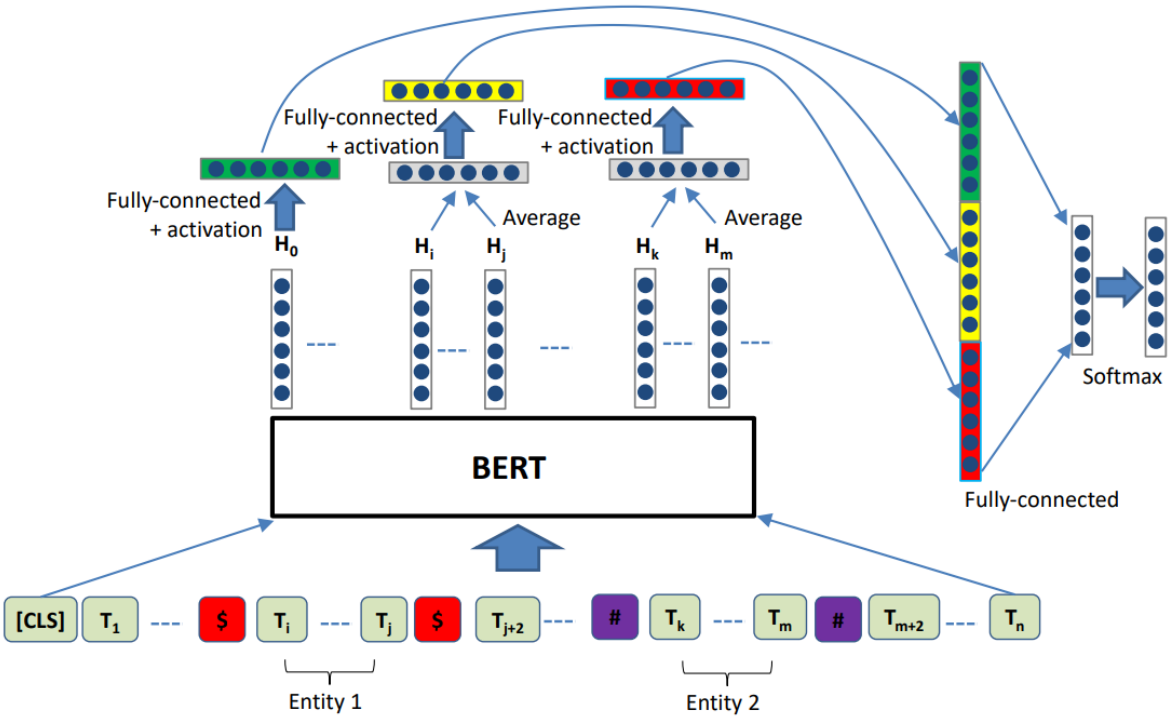
Vì vậy trong phương pháp này chúng tôi sẽ kết hợp R BERT và BERT-ES thành một mô hình đơn giản.

Trong phương pháp này tập trung vào nhiệm vụ phân loại quan hệ. Dữ liệu Training là một chuỗi các ví dụ.

- Mẫu quan hệ: $\boldsymbol{r} = (\boldsymbol{x},\ \boldsymbol{s_1},\ \boldsymbol{s_2},\ \boldsymbol{y})$ **.** Trong đó
  - $\boldsymbol{x} = [\boldsymbol{x_0}, \ldots, \boldsymbol{x_n}]$ là một chuỗi của mã thông báo, $\boldsymbol{x_0} = [\boldsymbol{CLS}]$ là một điểm đánh dấu bắt đầu đặc biệt.
  - $\boldsymbol{s_1} = (\boldsymbol{i}, \boldsymbol{j})$, $\boldsymbol{s_2} = (\boldsymbol{k}, \boldsymbol{l})$ là chỉ số của hai NE mục tiêu
  - $\boldsymbol{y}$ biểu thị nhãn quan hệ của hai NE được đề cập trong chuỗi $\boldsymbol{x}$

- Sử dụng một nhãn đặc biệt OTHER cho các NE không có mối quan hệ nào.

- Ví dụ (trích từ những bài báo "THẾ GIỚI" thuộc bộ dữ liệu VLSP2018.):
  - x = Triều Tiên hôm nay còn tuyên bố có thể thử bom nhiệt hạch trên Thái Bình Dương , sau khi Tổng thống Mỹ Donald Trump doạ " huỷ diệt hoàn toàn " nước này .
  - $s_1 = (23, 24)$ (Donald Trump), $s_2 = (22, 22)$ (Mỹ)
  - y = AFFILIATION
- Từ một dữ liệu training của mối quan hệ mẫu, train thành một mô hình classification.

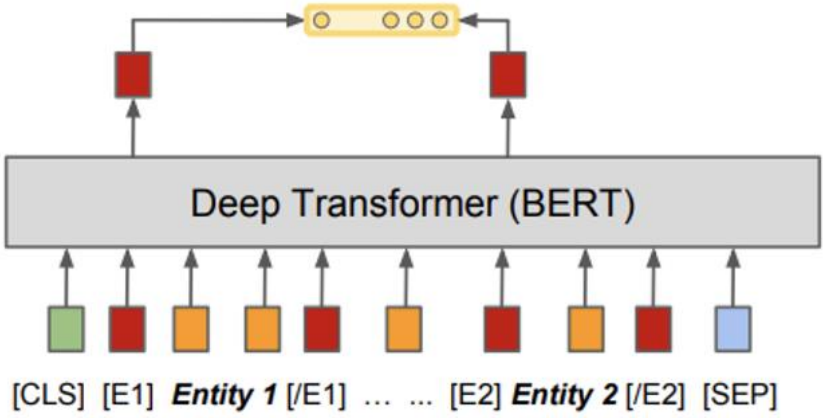


Hình 1.2. 2 Sơ đồ tổng quát R-BERT

Trong R-BERT, đối với một chuỗi x và hai thực thể (NE) mục tiêu e_1 và e_1 được chỉ định bởi các chỉ mục của s_1 và s_2, để làm cho mô-đun BERT nắm bắt thông tin vị trí của

hai NE, chúng tôi thêm kí tự đặc biệt **'$'** ở phần đầu và phần cuối của NE đầu tiên, kí tự đặc biệt **'#'** ở đầu và phần cuối của NE thứ hai. [CLS] cũng được thêm vào đầu chuỗi

- Ví dụ (trích từ những bài báo "THẾ GIỚI" thuộc bộ dữ liệu VLSP2018.):
    - x = [CLS] Triều Tiên hôm nay còntuyên bố có thể thử bom nhiệt hạch trên Thái Bình Dương , sau khi Tổng thống # Mỹ # $ Donald Trump $ doạ " huỷ diệt hoàn toàn " nước này ..
    - $s_1 = (26, 29)$, $s_2 = (23, 25)$
    - y = AFFILIATION



Hình 1.2. 3 Sơ đồ tổng quát BERT-ES

Mô hình BERT-ES tương tự như R-BERT, các kí tự đặc biệt được thêm vào đầu và cuối của hai thực thể. Trong các thử nghiệm của BERT-ES để phân loại quan hệ tiếng Việt, sử dụng các kí tự '**$**' và '**#**' thay vì '**[E1]**', '**[/ E1]**', '**[E1]**' và '**[/ E2]** '. Và không thêm [SEP] vào cuối chuỗi.

Trong phương pháp này, chúng tôi đã áp dụng R-BERT và BERT-ES, và đề xuất một mô hình tổng hợp của RBERT và BERT-ES.

Dưới đây là cách tiền xử lí dữ liệu để trainning các mô hình BERT-based và cách kết hợp R-BERT và BERT-ES.

- Các bước xử lí dữ liệu :
    - Trước tiên sử dụng thư viện VnCoreNLP để phân đoạn câu và mã hóa
    - Tiếp theo sử dụng âm tiết là đơn vị cơ bản trong câu
    - Tuy nhiên một số quan hệ có thể bị bỏ sót do lỗi phân đoạn câu

- Cách kết hợp hai mô hình :
    - Kết hợp R-BERT và BERT-ES để tạo thành một mô hình đồng bộ. Bằng cách tính toán các xác suất trung bình có trọng số được trả về bởi R –BERT và BERT -ES.
    - Trong các thử nghiệm, BERT-ES hoạt động tốt hơn một chút so với R-BERT, cho nên sử dụng trọng số lần lượt là 0,4 và 0,6 cho R-BERT và BERT-ES.

- Thực nghiệm và kết quả của phương pháp này

  Thử nghiệm được thiết lập:
    - Train mô hình bằng cách sử dụng dữ liệu training được cung cấp
    - Đánh giá trên tập dữ liệu development

Hyper-Parameters	**Value**
Max sequence length	384
Training epochs	10
Train batch size	16
Learning rate	2e-5

Bảng 1.2. 3 Hyper-parameters được sử dụng training models

- Kết quả thu được :

Model	Pre-trained BERT Model	MACRO F1	MICRO F1
R-BERT	NlpHUST/vibert4news	0.6392	0.7092
R-BERT	FPTAI/vibert	0.596	0.6736
BERT-ES	NlpHUST/vibert4news	**0.6439**	0.7101
BERT-ES	FPTAI/vibert	0.5976	0.6822
Ensemble Model	NlpHUST/vibert4news	0.6412	**0.7108**
Ensemble Model	FPTAI/vibert	0.6029	0.6851

Bảng 1.2. 4 Kết quả đánh giá trên tập dữ liệu dev

Model	MACRO F1	MICRO F1
R-BERT	0.6294	0.6645
BERT-ES	0.6276	0.6696
Ensemble Model	**0.6342**	**0.6756**

Bảng 1.2. 5 Kết quả đánh giá trên tập dữ liệu test

- Bảng 1.2. 4 cho thấy các kết quả đánh giá thu được trên tập dữ liệu dev. Chúng ta có thể thấy rằng việc sử dụng NlpHUST/vibert news vượt trội hơn đáng kể so với FPTAI / vibert về cả điểm số MICRO F1 và MACRO F1. BERT-ES hoạt động tốt hơn một chút so với R-BERT. Mô hình tập hợp được đề xuất được cải thiện một chút so với R-BERT và BERT-ES về điểm số MICRO F1.
- Bảng 1.2. 5 cho thấy kết quả đánh giá thu được trên tập dữ liệu test. Chúng tôi đã sử dụng NlpHUST/vibert news để tạo kết quả test. Mô hình tổng hợp đã thu được điểm MACRO F1 tốt nhất và điểm MICRO F1 tốt nhất trên dữ liệu test trong số ba mô hình.

	Precision	Recall	F1
AFFILIATION	0.7615	0.744	0.7528
LOCATED	0.7053	0.7007	0.7030
PART – WHOLE	0.65	0.8085	0.7206
PERSONAL - SOCIAL	0.6136	0.2842	0.3885

Bảng 1.2. 6 Precision, Recall, F1 cho từng loại quan hệ trên tập dữ liệu dev

	FPTAI/vibert	vibert4news
Data size	10GB	20GB
Data domain	News	News
Tokenization	Subword	Syllable
Vocab size	38168	62000

Bảng 1.2. 7 So sánh giữa FPTAI / vibert và NlpHUST / vibert4news

1.2.3 Phương pháp đề xuất giải quyết bài toán

Trước tiên chúng ta sẽ xây dựng bộ dữ liệu, bộ dữ liệu gồm tập train, dev và test. Mỗi dữ liệu sẽ bao gồm: một đoạn văn, vị trí của 2 thực thể và loại quan hệ của 2 thực thế đó.

Sử dụng mô hình PhoBERT hỗ trợ tiếng Việt để tạo các vectơ nhúng.

Sử dụng các mô hình dự đoán như: Multil-layers neural network để dự đoán mối quan hệ của từng cặp thực thể.

Nhóm lựa chọn phương pháp này vì:

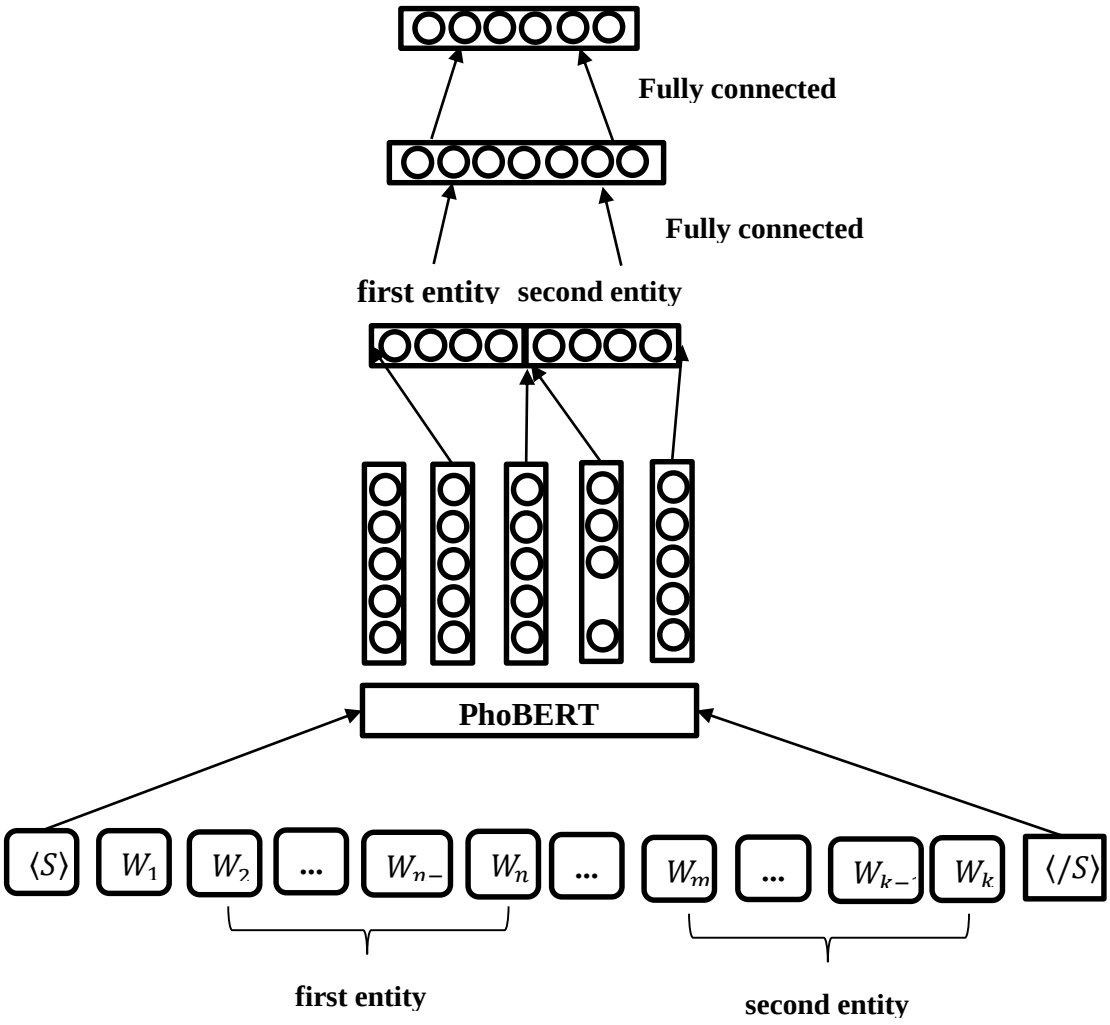
- BERT được coi là bước đột phá thực sự của Google trong lĩnh vực xử lý ngôn ngữ tự nhiên. Và PhoBERT – một pre-trained được huấn luyện sẵn dành cho tiếng Việt. PhoBERT đã được train sẵn trên khoảng 20GB dữ liệu.
- Hiện tại PhoBERT đang là mô hình hỗ trợ tiếng Việt được đánh giá đem lại kết quả tốt nhất.

1.3 Phương pháp giải quyết bài toán

1.3.1 Mô hình tổng quát

Mô hình tổng quát của phương pháp được trình bày theo sơ đồ dưới đây. Trong mô hình này gồm 3 phần chính:

- Phần 1: Tạo đầu vào cho mô hình PhoBERT bằng việc tạo word tokenize bằng việc sử dụng Underthesea.
- Phần 2: Sử dụng PhoBERT để tạo các vectơ nhúng.
- Phần 3: Huấn luyện bằng các mô hình multi-layer neural network



Hình 1.3. 1 Sơ đồ tổng quát PhoBERT đề xuất

1.3.2 Đặc trưng của mô hình đề xuất

1.3.2.1 Tạo word tokenize bằng Underthesea

Đầu vào của mô hình PhoBERT cần dữ liệu đã được word tokenize sẵn.

Trong tiếng việt, chúng ta cần phải tiến hành phân tách từ, vì một số từ được cấu thành bởi 2 hoặc nhiều từ trở lên. Ví dụ "đất nước" chúng ta phải tạo word tokenize có thể phân tách được các từ như vậy.

Vì vậy chúng ta sử dụng Underthesea để tạo các word tokenize.

1.3.2.2 Tạo vectơ nhúng bằng mô hình PhoBERT

Bước thứ 2 đó là tạo các vectơ nhúng hay còn gọi là xây dựng các ma trận thông tin đặc trưng. Bước này rất quan trọng để có thể huấn luyện được mô hình.

Tạo các vectơ nhúng gồm các bước chính:

- Ánh xạ các word tokenize vào bộ từ điển của PhoBERT để encode.
- Tạo embedding
  - Dựa vào kết quả của BertForSequenceClassification gồm 25 tầng layer.
  - Chúng ta sẽ lấy các embedding các word piece của từng thực thể sau đó cộng chúng lại với nhau để tạo thành một embedding hoàn chỉnh cho một thực thể.

1.3.2.3 Phương pháp huấn luyện

Sử dụng mô hình Multi-layer Neural Network gồm 2 tầng Linear Classification và 2 tầng Dropout..

Đầu ra của tầng Linear Classification thứ nhất là một ma trận 1024 cột.

Đầu ra của tầng Linear Classification thứ hai là một ma trận gồm 5 cột.

1.4 Thực nghiệm

1.4.1 Dữ liệu

Dữ liệu được lấy từ VLSP-2018 bao gồm ba bộ dữ liệu: Trainning, Development và Test, mỗi file chỉ chứa một dữ liệu thô (các bài báo điện tử) đã được xử lí và tách

thành câu. Với mỗi câu sẽ bao gồm 1 cặp thực thể (NE), căp thực thể đó phải dựa trên ba thực thể đã cho từ bộ dữ liệu lấy từ VLSP-2018: Locations , Organizations , và Persons. Sau khi dựa vào cặp thực thể đó chúng ta có thể suy ra mối quan hệ giữa Chúng trong câu, mối quan hệ bao gồm: LOCATED, PART-WHOLE, PERSONAL SOCIAL và ORGANIZATION–AFFILIATION. Và nếu giữa cặp thực thể đó không có mối quan hệ nào thì kiểu quan hệ của nó sẽ là OTHERS

1.4.2 Xử lí dữ liệu

Vì đầu vào của mô hình PhoBERT cần dữ liệu đã được word tokenize sẵn do đó chúng ta sử dụng Underthsea cho bước tiền xử lý dữ liệu.

Underthesea là thư viện hỗ trợ tiếng Việt giúp phân tách từ. Một số từ bao gồm nhiều từ trở lên mới tạo thành nghĩa hoàn chỉnh.

Underthesea cung cấp hàm word_tokenize hỗ trợ chúng ta làm điều này.

Sau khi tạo word tokenize cho đoạn văn bản, ta phải tiến hành xác định lại vị trí của các thực thể vì đoạn văn bản gốc có khoảng cách còn word tokenize thì không.

1.4.3 Công nghệ sử dụng

Ngôn ngữ	**Python 3.6**
Thư viện	Pytorch, Underthesea, tranformers, PhoBERT, Sklearn, numpy.
Môi trường	Google Colab với GPU

Bảng 1.4. 1 Thư viện và môi trường

1.4.4 Các đánh giá

Nhóm sử dụng độ đo Accuracy và MSE

- Accuracy
    - Độ đo này đơn giản dựa trên tỉ lệ số điểm dự đoán đúng trên tổng số điểm trong tập kiểm.
    - Công thức: n_correct / total_data
- F1-score macro

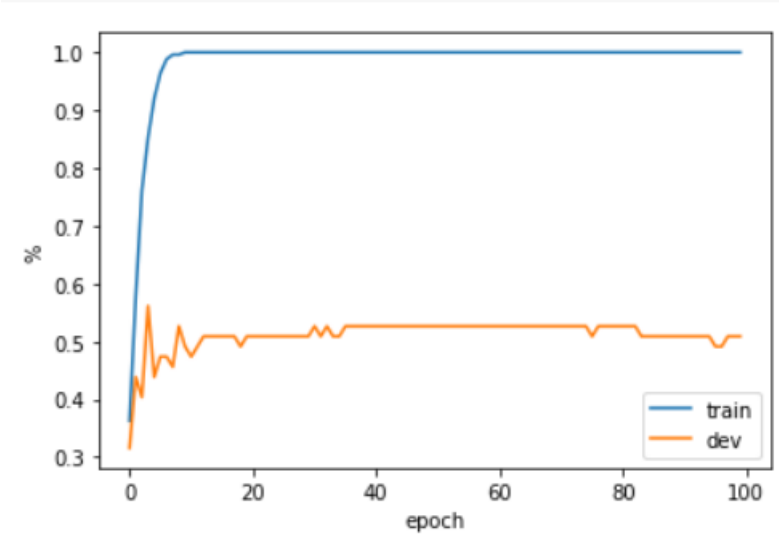
- F1 Score là trung bình điều hòa của precision và recall. Với tham số trung bình macro sẽ tính toán các chỉ số cho từng nhãn và tìm giá trị trung bình không trọng số của chúng.
- Công thức: F1= 2(precision * recall) / (precision + recall)
- Với:
  - precision = TP / (TP + FP)
  - recall = TP / (TP + FN)

1.5 Kết quả đạt được

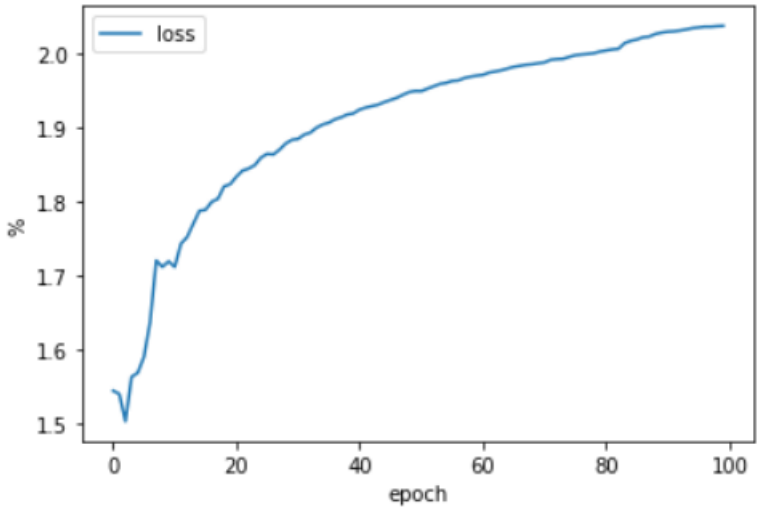
1.5.1 Tham số thực nghiệm

Tiểu luận sẽ tiến hành thực nghiệm trên tập dữ liệu tự xây dựng dựa theo tập dữ liệu vlsp 2020. Tập dữ liệu được chia làm 3 phần: train và dev dùng để huấn luyện và đánh giá mồ hình, tập test dùng để kiểm thử kết quả của mô hình.

Tiểu luận sẽ sử dụng mô hình multi-layer neural network với 2 layers Linear classifications và 2 lớp dropout. Mô hình sẽ thực hiện huấn luyện với batch size là 16, tổng số epoch là 100, đầu ra của tầng đầu tiên sẽ là ma trận 1024 cột và đầu ra của tầng cuối cùng là ma trận 5 cột tương ứng với 5 loại mối quan hệ.

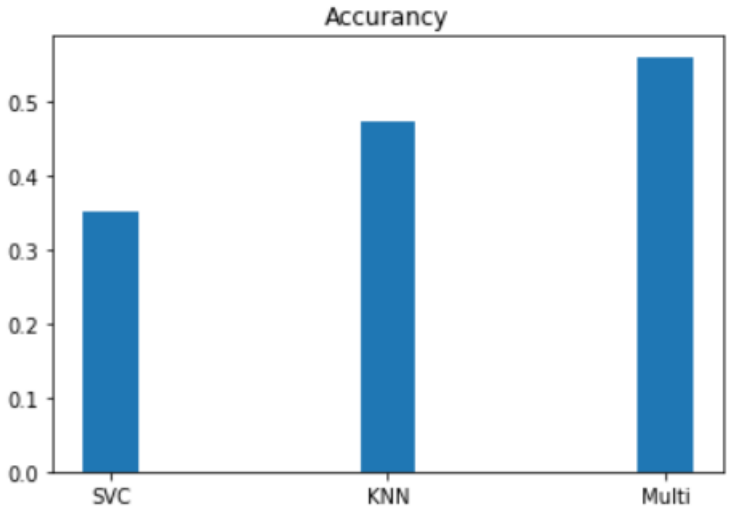


Hình 1.5. 1 Chỉ số accuracy của tập train và dev trong quá trình huấn luyện



Hình 1.5. 2 Chỉ số loss trong quá trình huấn luyện

1.5.2 Kết quả đạt được

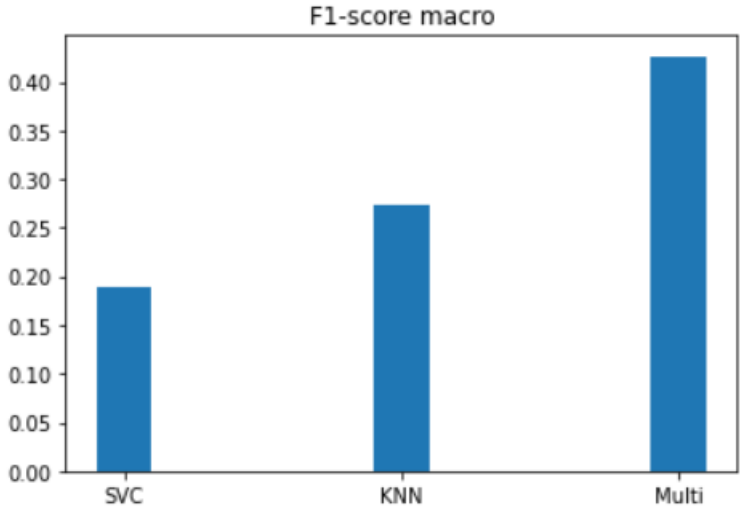


Hình 1.5. 3 Độ chính xác của 3 model dựa theo độ đo accuracy trên tập dev

- Cụ thể

Model	SVC	KNN	Multi
Accuaracy	0.35	0.47	0.57

Bảng 1.5. 1 Độ chính xác của 3 model dựa theo độ đo accuracy trên tập dev



Hình 1.5. 4 Độ chính xác của 3 model dựa theo độ đo F1-score với average macro trên tập dev

- Cụ thể

Model	SVC	KNN	Multi
F1-score macro	0.19	0.27	0.43

Bảng 1.5. 2 Độ chính xác của 3 model dựa theo độ đo F1-score với average macro trên tập dev

Kết quả độ chính xác của 3 model trên gần như rất thấp so. Kết quả dự đoán có thể ảnh hưởng bởi tập dữ liệu. Vì nhóm tự xây dựng dữ liệu do đó có thể sai sót trong quá trình xác định label cho từng dữ liệu dẫn đến ảnh hưởng tới kết quả dự đoán. Và tập dữ liệu để train còn thực sự ít, chỉ khoảng 250 dòng.

Kết quả thu được ở 3 model training có sự khác nhau khá lớn. Chúng ta có thể thấy rằng, ở model multi-layers neral network cho kết quả tốt nhất kế đến là KNN và cuối cùng là SVC. Multi-layers cho kết quả tốt hơn nhiều 2 model còn lại ngoài tính chất của model ra còn bởi: do nhóm thực hiện trên môi trường colab với bộ xử lý GPU của google. Trong quá trình embedding, khi chúng ta embedding cùng lúc toàn bộ tập input

thì sẽ bị hết RAM, điều này bởi vì đầu vào input rất lớn. Do đó ở 2 model SVC và KNN nhóm không thực hiện embedding.

1.6 Kết luận

1.6.1 Kết quả đạt được

Về mặt lý thuyết, tiểu luận đã tìm hiểu về các phương pháp giải quyết bài toán rút trích mối quan hệ trong tiếng việt, đồng thời tiểu luận cũng đề xuất phương pháp sử dụng PhoBERT.

Về mặt thực nghiệm, tiểu luận đã tự xây dựng tập dữ liệu dựa trên format của bộ dữ liệu VLSP 2020 cho mồ hình đề xuất cùng với 2 mô hình SVC và KNN để so sánh kết quả. Kết quả cho thấy mô hình đề xuất mang lại kết quả tốt hơn so với các mô hình còn lại.

1.6.2 Hạn chế

Tiểu luận hạn chế bởi tập dữ liệu, vì không có tập dữ liệu đủ lớn và chính xác và việc tự xây dựng bộ dữ liệu tốn rất nhiều thời gian và dễ nhầm lẫn. Vì các mô hình xử lý Tiếng Việt rất ảnh hưởng bởi dữ liệu nhiễu, trong tiếng việt chỉ cần thay đổi dấu hay dấu phẩy cũng đã ảnh hưởng tới ý nghĩa của câu.

1.6.3 Hướng phát triển

Tiểu luận thực hiện dựa trên format của bộ VLSP 2020, trong đó đã cung cấp label cho 1 cặp thực thể. Trong tương lai, nhóm sẽ tiến hành thực hiện bước NER (Named Entity Recognition) để xác định được tag của từng thực thể, từ đó xác định được label cho mối quan hệ trước khi tiến hành huấn luyện dữ liệu để dự đoán.

Do đó có thể tạo được một ứng dụng hoàn chỉnh cho việc rút trích mối quan hệ tiếng việt bằng việc chỉ cần nhập vào một đoạn văn tiếng Việt và chọn cặp thực thể cần lấy mối quan hệ.

TÀI LIỆU THAM KHẢO

1. https://github.com/undertheseanlp/NLP-Vietnamese-progress
2. https://vlsp.org.vn/vlsp2020/eval/re
3. https://drive.google.com/file/d/1uzTIuxjsCw3TmNAdE7T6v6XB2zurkpTF/view
4. https://drive.google.com/file/d/1ckKBS8T55oCWS3JnwWsNdzKPBsclDDHb/view
5. https://drive.google.com/file/d/1To2pL-UAEiKWNFYDJzHgO7ls20lh4_J0/view
6. Linear Regression - Hồi quy tuyến tính trong Machine Learning NguyenDuong https://viblo.asia/p/linear-regression-hoi-quy-tuyen-tinh-trong-machine-learning-4P856akRlY3
7. sklearn.metrics.accuracy_score và mean_squared_error, https://scikitlearn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html
8. Machine learning: Trích xuất đặc trưng văn bản - Part 1 Thuan https://viblo.asia/p/machine-learning-trich-xuat-dac-trung-van-ban-part-1-oOVlYqzzl8W
9. Deep Neural Network Based Relation Extraction: An Overview Wang et al. https://arxiv.org/abs/2101.01907
10. NLP: Deep learning for relation extraction Handmark https://towardsdatascience.com/nlp-deep-learning-for-relation-extraction-9c5d13110afa
11. BERT, RoBERTa, PhoBERT, BERTweet: Ứng dụng state-of-the-art pre-trained model cho bài toán phân loại văn bản Quang https://viblo.asia/p/bert-roberta-phobert-bertweet-ung-dung-state-of-the-art-pre-trained-model-cho-bai-toan-phan-loai-van-ban-4P856PEWZY3
12. Giới thiệu BERT và ứng dụng vào bài toán phân loại văn bản N.v.t https://blog.vietnamlab.vn/gioi-thieu-bert-va-ung-dung-vao-bai-toan-phan-loai-van-ban/

TỰ ĐÁNH GIÁ

Câu	Nội dung	Điểm chuẩn	Tự chấm	Ghi chú
1 (8.5)	1.1 Giới thiệu về bài toán	0.5đ	0.5đ	
	1.2 Phân tích yêu cầu của bài toán	1.0đ	1.0đ	
	1.3 Phương pháp giải quyết bài toán	1.5đ	1.0đ	
	1.4 Thực nghiệm	4.0đ	3.0đ	
	1.5 Kết quả đạt được	1.0đ	0.5đ	
	1.6 Kết luận	0.5đ	0.5đ	
2	Điểm nhóm	0.5đ	0.5đ	
3	Báo cáo (chú ý các chú ý 2,3,4,6 ở trang trước, nếu sai sẽ bị trừ điểm nặng)	1.0đ	1.0đ	
Tổng điểm			8.0đ	