

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN

NGÀNH KỸ THUẬT DỮ LIỆU



HỒNG TIẾN HÀO – 19133022

**Tìm hiểu bài toán phân tích biến động giá nông sản
từ dữ liệu tin tức**

**KHÓA LUẬN TỐT NGHIỆP
KỸ SƯ NGÀNH KỸ THUẬT DỮ LIỆU**

**GIẢNG VIÊN HƯỚNG DẪN
TS. TRẦN NHẬT QUANG**

KHÓA 2019 – 2023

ĐH SƯ PHẠM KỸ THUẬT TP. HCM
KHOA CNTT

XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh Phúc

PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

Họ và tên sinh viên: **Hồng Tiến Hào**

MSSV: **19133022**

Chuyên ngành: **Kỹ thuật dữ liệu**

Tên đề tài: **Tìm hiểu bài toán phân tích biến động giá nông sản từ dữ liệu tin tức**

Họ và tên giáo viên hướng dẫn: **TS. Trần Nhật Quang**

NHẬN XÉT

1. Về nội dung đề tài và khối lượng thực hiện:

2. Ưu điểm:

3. Khuyết điểm:

4. Đề nghị cho bảo vệ hay không?

5. Đánh giá loại:

6. Điểm:

Tp. Hồ Chí Minh, ngày....tháng....năm 202..

Giáo viên phản biện

(Ký & ghi rõ họ tên)

ĐH SƯ PHẠM KỸ THUẬT TP. HCM
KHOA CNTT

XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh Phúc

PHIẾU NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

Họ và tên sinh viên: **Hồng Tiến Hào**

MSSV: **19133022**

Chuyên ngành: **Kỹ thuật dữ liệu**

Tên đề tài: **Tìm hiểu bài toán phân tích biến động giá nông sản từ dữ liệu tin tức**

Họ và tên giáo viên phản biện: **TS. Nguyễn Thành Sơn**

NHẬN XÉT

1. Về nội dung đề tài và khối lượng thực hiện:

2. Ưu điểm:

3. Khuyết điểm:

4. Đề nghị cho bảo vệ hay không?

5. Đánh giá loại:

6. Điểm:

Tp. Hồ Chí Minh, ngày.... tháng....năm 202..

Giáo viên phản biện

(Ký & ghi rõ họ tên)

LỜI CẢM ƠN

Lời đầu tiên, tôi xin gửi lời cảm ơn chân thành và sâu sắc đến Khoa Công Nghệ Thông Tin - Trường Đại Học Sư Phạm Kỹ Thuật Thành Phố Hồ Chí Minh đã tạo điều kiện cho tôi phát triển nền tảng kiến thức sâu sắc và thực hiện đề tài này.

Tôi cũng muốn gửi lời cảm ơn sâu sắc nhất đến thầy Trần Nhật Quang, người đã giúp đỡ và hướng dẫn tôi trong suốt quá trình thực hiện luận văn tốt nghiệp của mình. Thầy đã tận tâm chỉ bảo nhiệt tình trong suốt quá trình từ lúc bắt đầu cũng như kết thúc đề tài này. Với những kinh nghiệm chuyên môn cũng như thực tiễn của các thầy cô, tôi đã học được rất nhiều kiến thức và kinh nghiệm khổng lồ về chuyên ngành và thực hiện dự án, giúp cho công việc và học vấn của tôi trong tương lai. Tôi rất biết ơn điều này đã giúp và thôi thúc tôi hoàn thành đề tài. Tôi sẽ luôn khắc ghi những kiến thức đó và sử dụng chúng như một hành trang vô cùng lớn trước khi bước vào cuộc sống mới. Tuy nhiên, kiến thức lúc nào cũng là vô tận và với khả năng và chuyên môn còn nhiều hạn chế của mình, tôi sẽ luôn cố gắng hết sức để hoàn thành một cách tốt nhất. Vì vậy, việc xảy ra những thiếu sót của tôi là điều khó tránh khỏi trong quá trình học hỏi và thực hiện luận văn.

Tôi hy vọng nhận được sự thông cảm và các góp ý tận tình và quý báu của các thầy cô. Thông qua đó, tôi có thể rút ra được bài học kinh nghiệm, hoàn thiện và nâng cấp sản phẩm của mình tốt hơn nữa. Cuối cùng, tôi xin gửi lời cảm ơn sâu sắc nhất đến thầy Trần Nhật Quang và tập thể các thầy cô Khoa Công Nghệ Thông Tin - Trường Đại Học Sư Phạm Kỹ Thuật Thành Phố Hồ Chí Minh vì tất cả những điều thầy cô đã gửi gắm và chỉ dạy cho tôi.

ĐỀ CƯƠNG KHÓA LUẬN TỐT NGHIỆP

Họ và tên sinh viên thực hiện: **Hồng Tiến Hào**

Mã số SV: **19133022**

Thời gian làm khóa luận: từ: 04/09/2023

Đến: 28/12/2023

Chuyên ngành: **Kỹ thuật dữ liệu**

Tên đề tài: **Tìm hiểu bài toán phân tích biến động giá nông sản từ dữ liệu tin tức**

Họ và tên Giáo viên hướng dẫn: TS. **Trần Nhật Quang**

Nhiệm Vụ Của Luận Văn:

1. Tìm hiểu về bài toán phân tích tin tức sử dụng NLP.
2. Tìm kiếm nguồn dữ liệu tin tức phù hợp.
3. Thu thập dữ liệu về tin tức tiếng Việt liên quan đến giá một số nông sản
4. Gắn nhãn tin tức
5. Tìm hiểu thuật toán mô hình và mô hình phù hợp để huấn luyện.
6. Áp dụng mô hình vào ứng dụng web

Đề cương viết luận văn:

1. Phần mở đầu

- 1.1. Tính cấp thiết của đề tài
- 1.2. Mục tiêu đề tài
- 1.3. Phương pháp thực hiện
- 1.4. Kết quả dự kiến đạt được

2. Phần nội dung

- 2.1. Chương 1. Dữ liệu
- 2.2. Chương 2. Xây dựng mô hình học sâu

2.3. Chương 3. Thực nghiệm và ứng dụng thực tế

3. Phần kết luận

3.1. Những kết quả đạt được

3.2. Thuận lợi

3.3. khó khăn

3.4. Hướng phát triển

KẾ HOẠCH THỰC HIỆN

Tuần	Thời gian	Nội dung công việc	Ghi chú
1	21/8 – 10/9	<ul style="list-style-type: none">- Liên hệ giáo viên hướng dẫn (GVHD)- Trao đổi với GVHD và chọn đề tài- Xác định các mục tiêu của đề tài- Lập kế hoạch thực hiện	
2			
3			
4	11/9 – 17/9	<ul style="list-style-type: none">- Tìm kiếm và tổng hợp các bài báo khoa học liên quan đến đề tài.- Báo cáo với GVHD	
5	18/9 – 24/9	<ul style="list-style-type: none">- Tìm kiếm nguồn dữ liệu tin tức nông sản- Thu thập dữ liệu bằng Selenium	
6	25/9 – 1/10	<ul style="list-style-type: none">- Xây dựng chính sách gắn nhãn- Gắn nhãn tin tức- Báo cáo với GVHD	
7	2/10 – 15/10	<ul style="list-style-type: none">- Tiếp tục gắn nhãn tin tức- Báo cáo với GVHD	
8			
9	16/10 – 22/10	<ul style="list-style-type: none">- Hoàn tất gắn nhãn- Trực quan hóa đánh giá dữ liệu hiện có- Tiền xử lý dữ liệu	
10	23/10 – 5/11	<ul style="list-style-type: none">- Tìm hiểu bài toán NLP- Tìm hiểu kiến trúc Transformer, BERT.- Tìm hiểu các giải pháp giải quyết dữ liệu bị mất cân bằng- Báo cáo với GVHD	
11			
12	6/11 – 12/11	<ul style="list-style-type: none">- Tìm hiểu các thư viện hỗ trợ và viết code hiện thực.- Huấn luyện mô hình- Đánh giá và ghi nhận kết quả thu được	

13	13/11 – 19/11	<ul style="list-style-type: none"> - Tinh chỉnh các tham số cải thiện độ chính xác cho mô hình - Thử các mô hình và các giải pháp khác để cải thiện độ chính xác - Báo cáo với GVHD 	
14	20/11 – 27/9	<ul style="list-style-type: none"> - Tìm hiểu Flask và các thư viện hỗ trợ viết ứng dụng - Xây dựng ứng dụng web - Triển khai mô hình học sâu trên ứng dụng - Kiểm thử và sửa lỗi ứng dụng 	
15			
16			
17	11/12 – 17/12	<ul style="list-style-type: none"> - Viết báo cáo 	
18	18/12 – 24/12	<ul style="list-style-type: none"> - Báo cáo với GVHD - Chỉnh sửa theo góp ý của giáo viên hướng dẫn 	
19	25/12 – 31/12	<ul style="list-style-type: none"> - Hoàn tất đề tài - Liên hệ Giáo viên phản biện để tiến hành phản biện 	
20	1/1 – 7/1/2024	<ul style="list-style-type: none"> - Tiến hành phản biện trước hội đồng - Chỉnh sửa báo cáo theo góp ý của hội đồng 	

TP.Hồ Chí Minh, ngày tháng năm 2024

Ý kiến của giáo viên hướng dẫn

(ký và ghi rõ họ tên)

Người viết đề cương

(ký và ghi rõ họ tên)

TS. Trần Nhật Quang

Hồng Tiến Hào

MỤC LỤC

DANH MỤC BẢNG	1
DANH MỤC HÌNH ẢNH	2
DANH MỤC TỪ VIẾT TẮT	4
PHẦN MỞ ĐẦU	5
1. TÍNH CẤP THIẾT CỦA ĐỀ TÀI.....	5
2. MỤC TIÊU ĐỀ TÀI	5
3. PHƯƠNG PHÁP THỰC HIỆN	5
4. KẾT QUẢ DỰ KIẾN ĐẠT ĐƯỢC	6
PHẦN NỘI DUNG	7
CHƯƠNG 1: DỮ LIỆU.....	7
1.1. XÁC ĐỊNH NGUỒN DỮ LIỆU.....	7
1.2. THU THẬP DỮ LIỆU	8
1.3. GẮN NHÃN.....	12
CHƯƠNG 2: XÂY DỰNG MÔ HÌNH HỌC SÂU	16
2.1. CƠ SỞ LÝ THUYẾT HỌC SÂU	16
2.2. XỬ LÝ NGÔN NGỮ TỰ NHIÊN	25
2.3. MÔ HÌNH TRANSFORMER.....	27
2.4. MÔ HÌNH BERT	29
2.5. TỐI ƯU HÓA	38
2.6. ĐÁNH GIÁ MÔ HÌNH.....	40
CHƯƠNG 3: THỰC NGHIỆM VÀ ỨNG DỤNG THỰC TẾ.....	43
4.1. TIỀN XỬ LÝ	43
4.2. HUẤN LUYỆN MÔ HÌNH	44
4.3. TRIỂN KHAI TRÊN ỨNG DỤNG	48
PHẦN KẾT LUẬN	54
1. KẾT QUẢ ĐẠT ĐƯỢC	54

1.1.	Kiến thức chuyên môn	54
1.2.	Kỹ năng	54
1.3.	Ưu nhược điểm.....	54
2.	ĐÓNG GÓP	54
3.	THUẬN LỢI VÀ KHÓ KHĂN	55
3.1.	Thuận lợi	55
3.2.	Khó khăn	55
4.	HƯỚNG PHÁT TRIỂN.....	55
4.1.	Tiếp tục phát triển mô hình hiện tại	55
4.2.	Là công cụ phát triển mô hình khác	55
TÀI LIỆU THAM KHẢO		57

DANH MỤC BẢNG

Bảng 1.1: Thông tin dữ liệu được thu thập.....	12
Bảng 1.2: Các nhân tố ảnh hưởng và loại nhãn ứng với các chủ đề lớn	13
Bảng 2.1 Các mô hình pre-trained của PhoBERT.....	37
Bảng 2.2 Confusion Matrix với bài toán 4 nhãn	41
Bảng 3.1 Cấu hình Colab notebook.....	44
Bảng 3.2 Các siêu tham số được chọn để huấn luyện mô hình.....	45
Bảng 3.3 Cấu hình máy tính dùng để chạy ứng dụng	48
Bảng 3.4 Thư viện được dùng để lập trình ứng dụng.....	48

DANH MỤC HÌNH ẢNH

Hình 1.1 Trang chủ website AGROINFO [3]	8
Hình 1.2 Giao diện website thu thập dữ liệu	10
Hình 1.3 Các trường dữ liệu lấy từ website.....	11
Hình 1.4 Biểu đồ phân bố nhãn của mỗi lớp	15
Hình 2.1 Minh họa mạng kết nối đầy đủ (trái) và cấu trúc một nơ-ron trong mạng (phải)	17
Hình 2.2 Mạng trước (trái) và sau khi áp dụng (phải) dropout 0.25	17
Hình 2.3 Thiết kế kiến trúc mạng nơ-ron có skip connection	18
Hình 2.4 Đồ thị biểu diễn hàm sigmoid	19
Hình 2.5 Đồ thị biểu diễn hàm ReLU.....	20
Hình 2.6 Sự đánh đổi giữa bias và variance	22
Hình 2.7 Khi không áp dụng (nét liền) và áp dụng học chuyển giao (nét đứt) ([7]).....	24
Hình 2.8 Minh họa word Embedding trong không gian 2 chiều.....	26
Hình 2.9 Kiến trúc Transformer gồm khối Encoder (trái) và Decoder (phải) ([10])	28
Hình 2.10 Chuyển từ văn bản sang véc-tơ số trong Transformer.....	30
Hình 2.11 Thuật toán Positional Encoding	31
Hình 2.12 Giá trị Query – Key – Value	31
Hình 2.13 Scale dot-product attention.....	32
Hình 2.14 Cơ chế Multi-head Attention ([10]).....	33
Hình 2.15 Cơ chế huấn luyện masked language modeling	35
Hình 2.16 Cơ chế huấn luyện next sentence prediction	36
Hình 2.17 Linearly learning rate decay	40
Hình 3.1 Các giai đoạn tiền xử lý dữ liệu.....	44
Hình 3.2 Kiến trúc mô hình giải quyết bài toán	46
Hình 3.3 Biểu đồ huấn luyện biểu diễn độ đo F1 (trái) và Loss (phải) qua mỗi epoch của mô hình PhoBERT _{base}	47
Hình 3.4 Biểu đồ huấn luyện biểu diễn độ đo F1 (trái) và Loss (phải) qua mỗi epoch của mô hình PhoBERT _{base} Version 2	47
Hình 3.5 Kiến trúc của ứng dụng web triển khai mô hình của đề tài.....	50
Hình 3.6 Giao diện trang chủ website	51

Hình 3.7 Giao diện danh sách tin website	51
Hình 3.8 Giao diện đăng nhập (trái) và đăng ký (phải).....	52
Hình 3.9 Giao diện phân tích tin tức của website.....	52

DANH MỤC TỪ VIẾT TẮT

BERT	Bidirectional Encoder Representation from Transformer
CNN	Convolutional Neural Network
FC	Fully Connected
MLM	Masked Language Model
NLP	Natural Language Processing
NSP	Next Sentence Prediction
seq2seq	Sequence to Sequence
SOTA	State-Of-The-Art

PHẦN MỞ ĐẦU

1. TÍNH CẤP THIẾT CỦA ĐỀ TÀI

Việt Nam là đất nước nông nghiệp, đi lên từ nông thôn. Do đó, việc quan tâm đến nền nông nghiệp có ý nghĩa để phát triển kinh tế, chống lại và xác định các mặt trái các yếu tố gây ảnh hưởng đến giá cả nông sản cũng mang ý nghĩa nhân văn để hỗ trợ người nông dân vượt qua. Nên tôi đã chọn đề tài: **“tìm hiểu bài toán biến động giá cả nông nghiệp từ dữ liệu tin tức”**.

Đoán trước được biến động giá có thể giúp quản lý rủi ro thực hiện kế hoạch kinh doanh và đầu tư hiệu quả hơn. Các quyết định về sản xuất nông sản cũng có thể được điều chỉnh dựa trên giá, từ đó tối ưu hóa sản xuất và cung ứng. Cùng với dữ liệu tin tức cung cấp thông tin về các sự kiện trong và ngoài nước, xu hướng ngắn và dài hạn, có thể ảnh hưởng đến giá nông sản. Nắm bắt được điều này, ta có thể thu thập dữ liệu và đào tạo một mô hình học sâu dự đoán và là tiền đề để áp dụng trí thông minh nhân tạo vào nền nông nghiệp của nước ta.

2. MỤC TIÊU ĐỀ TÀI

- Tìm hiểu về bài toán phân tích tin tức sử dụng NLP.
- Thu thập dữ liệu về tin tức tiếng Việt liên quan đến giá một số nông sản
- Gắn nhãn tin tức và huấn luyện mô hình.
- Xây dựng và áp dụng vào ứng dụng web đơn giản

3. PHƯƠNG PHÁP THỰC HIỆN

Tìm hiểu các kiến thức và công cụ để thực hiện đề tài:

- Tìm kiếm nguồn dữ liệu và thu thập chúng
- Xác định các khía cạnh tác động đến giá cả dựa trên cơ sở khoa học để gắn nhãn

- Tìm hiểu và sử dụng các thư viện để xây dựng đề tài
- Sử dụng phương pháp nghiên cứu: thống kê, tổng hợp và phân tích đánh giá, so sánh đối chiếu, khái quát hóa để hiện thực đề tài.
- Quát hóa xây dựng và áp dụng mô hình Transformer để giải quyết bài toán

4. KẾT QUẢ DỰ KIẾN ĐẠT ĐƯỢC

Sau khi thực hiện đề tài, nhóm mong muốn những điều sau:

- Hiểu các quy trình xoay quanh dữ liệu từ thu thập, gắn nhãn cho đến xử lý dữ liệu.
- Nắm được lý thuyết và phương pháp xây dựng mô hình học sâu dựa trên kiến trúc Transformer.
- Thiết kế, xây dựng được mô hình học sâu phân tích tin giá biến động dựa trên dữ liệu tin tức.
- Xây dựng được một ứng dụng web dùng để triển khai mô hình trong thực tế.

PHẦN NỘI DUNG

CHƯƠNG 1: DỮ LIỆU

1.1. XÁC ĐỊNH NGUỒN DỮ LIỆU

Dự đoán giá nói chung và giá nông sản nói riêng không phải là đề tài mới, nhiều công trình nghiên cứu được liệt kê trong khảo sát [1] đa phần là các bài toán sử dụng dữ liệu lịch sử giá nông sản (time-series). Nhưng các yếu tố khác tác động ảnh hưởng đến giá làm giảm khả năng dự đoán cho nếu mô hình chỉ dựa vào duy nhất lịch sử giá cũng là một hạn chế. Công trình nghiên cứu [2] cũng cho thấy khi sử dụng thêm tiêu đề tin tức kết hợp dữ liệu lịch sử giá để huấn luyện mô hình cũng cải thiện được khả năng dự đoán giá của mô hình. Dù vậy mô hình chỉ dừng lại ở dự đoán giá và không giải thích các yếu tố ảnh hưởng đến giá, điều tôi cho là quan trọng để có thể giúp cho những người hay tổ chức liên quan để điều chỉnh phù hợp với thị trường. Trong đề tài này sẽ cố gắng đưa ra nhưng khía cạnh phần nào giải thích cho sự biến động giá của nông sản.

Dữ liệu tin tức tuy nhiều nhưng phải có tính đúng đắn, khoa học, và phù hợp với đề tài sẽ cần phải đảm bảo các tiêu chí sau:

- Tin tức nông sản Việt Nam
- Dữ liệu là ngôn ngữ Việt
- Tin tức phải có tính xác thực
- Số mẫu dữ liệu đủ cho việc huấn luyện mô hình.

Do dữ liệu tiếng Việt còn hạn chế nên hiện nay vẫn chưa có tập dữ liệu có sẵn nào, nên đã tiến hành tự thu thập dữ liệu. Sau khi tìm kiếm nguồn dữ liệu thì trang thông tin thuộc Trung tâm Thông tin Phát triển Nông nghiệp Nông thôn (AGROINFO) được xác định là một trang báo uy tín phù hợp dùng để phân tích và thu thập dữ liệu.

KẾT NỐI NGHIÊN CỨU VỚI THỰC TIỄN
 CHO MỘT NỀN NÔNG NGHIỆP TĂNG TRƯỞNG TOÀN DIỆN

Tìm kiếm

TRANG CHỦ

GIỚI THIỆU

TIN TỨC

HOẠT ĐỘNG

NHÂN SỰ

SẢN PHẨM

29 | 11 | 2023

THÁI LAN TĂNG CƯỜNG XUẤT KHẨU NÔNG SẢN BẰNG ĐƯỜNG SẮT
 Từ tháng 12, Thái Lan sẽ tăng cường xuất khẩu nông sản sang Trung Quốc thông qua hệ thống đường sắt.

XEM TIẾP

THÁI LAN TĂNG CƯỜNG XUẤT KHẨU NÔNG SẢN BẰNG ĐƯỜNG SẮT

THỊ TRƯỜNG HÀNG HÓA NÔNG SẢN DỰ KIẾN "HẠ NHIỆT" TRONG NĂM 2024

8.000 HỘ DÂN HƯỞNG LỢI KHI THAM GIA SẢN XUẤT HỒ TIÊU BẾN VŨNG

DOANH NGHIỆP FDI CHIẾM GẦN 50% KIM NGẠCH XUẤT KHẨU NGÀNH GỖ

GIÁ CẢ THỊ TRƯỜNG

Tên mặt hàng	Thị trường	Ngày	Giá	Loại tiền	Loại giá	Nguồn	Đơn vị tính
Gạo thơm thái hạt dài	An Giang	28-11-2023	20,000	VND	Bán lẻ	AGROINFO	kg

BÁO CÁO PHÂN TÍCH THỊ TRƯỜNG

Số tháng 10/2023

BÁO CÁO TÌNH HÌNH XUẤT NHẬP KHẨU NÔNG LÂM THỦY SẢN

Báo cáo

Hình 1.1 Trang chủ website AGROINFO [3]

AGROINFO thành lập vào năm 2006 là Trung tâm tự chủ trực thuộc Viện Chính sách và Chiến lược Phát triển Nông nghiệp Nông thôn (IPSARD), với nhiệm vụ cung cấp thông tin nông nghiệp và nông thôn cũng như tiến hành phân tích các chính sách của Chính phủ và các quy định trong nông nghiệp cho Bộ Nông Nghiệp và Phát triển Nông thôn. Với nguồn dữ liệu tin tức tiếng việt dồi dào từ năm 2006 và hơn hết là có thể tin cậy là một nơi phù hợp để thu thập dữ liệu cho đề tài.

1.2. THU THẬP DỮ LIỆU

1.2.1. Cách tiếp cận

Dữ liệu tin tức được hiển thị trên trình duyệt, nên bằng cách tiếp cận bằng Selenium tương tác với trình duyệt Web Driver chúng ta có thể tự động điều khiển trình duyệt web và thu thập dữ liệu từ các trang AGROINFO. Đây là một phương pháp phổ biến để thu thập và trích xuất thông tin từ các trang web động hoặc tĩnh.

1.2.1.1. *Selenium*

Selenium là một bộ công cụ phần mềm mã nguồn mở được sử dụng để tự động hóa việc kiểm thử và điều khiển trình duyệt web. Selenium còn cho phép viết các kịch bản tự động hóa bằng các ngôn ngữ lập trình như Java, Python, C#, Ruby, và nhiều ngôn ngữ khác. Nó hỗ trợ các trình duyệt phổ biến như Google Chrome, Mozilla Firefox, Microsoft Edge, và Safari. Từ đó Selenium có thể được sử dụng để thực hiện các nhiệm vụ như điều hướng qua các trang web, nhập liệu, kiểm tra sự hiển thị của các phần tử trên trang, kiểm tra các chức năng và tính năng của trang web, và thu thập dữ liệu tự động. Để có thể sử dụng công cụ Selenium để thu thập dữ liệu từ website:

- **Bước 1:** Cài đặt thư viện bằng lệnh:

`pip install selenium`

- **Bước 2:** Tải Web Driver: Web Driver là là một automation framework của trình duyệt cho phép thực hiện các kiểm thử trên nhiều trình duyệt: Firefox, Chrome, Edge... Nó là một thành phần không thể thiếu trong bộ kiểm thử tự động Selenium. Mỗi trình duyệt sẽ cần cài đặt Selenium WebDriver phù hợp với trình duyệt đó.

1.2.1.2. Thực hiện thu thập

Ta có giao diện website như Hình 1.2 sau:

GIÁ GẠO THẾ GIỚI TĂNG TRỞ LẠI, GẠO VIỆT VẪN DẪN ĐẦU	18 10 2023
Thị trường gạo thế giới đang nóng trở lại do nhu cầu tiêu thụ vẫn cao trong khi nước xuất khẩu lớn nhất là Ấn Độ vẫn chưa nới lỏng các chính sách hạn chế xuất khẩu.	
VIỆT NAM SẼ LÀ NGUỒN CUNG GẠO CHÍNH CHO INDONESIA	13 10 2023
Trước thông tin Indonesia sẽ nhập khẩu 1,5 triệu tấn gạo từ Việt Nam và Thái Lan trong thời gian tới, Thương vụ Việt Nam tại Indonesia nhấn mạnh: Việc Indonesia chọn Việt Nam là nguồn cung chính cho các đợt thu mua lúa gạo của Indonesia đã khẳng định thêm vị thế, uy tín chất lượng của hạt gạo Việt Nam.	
KỶ LỤC MỚI CỦA NGÀNH LÚA GẠO	06 10 2023
Kết thúc 9 tháng, cả nước đã xuất khẩu được 6,6 triệu tấn gạo, đạt kim ngạch gần 3,7 tỉ USD, con số cao nhất từ trước tới nay. Với đà này, dự kiến cả năm 2023, xuất khẩu sẽ lập kỷ lục trên 8 triệu tấn, với kim ngạch khoảng 4,5 tỉ USD. . .	
THỊ TRƯỜNG TRẦM LẶNG, GIÁ GẠO GIẢM NHIỆT	27 09 2023
Những ngày cuối tháng 9, giá gạo VN và thế giới về quanh mốc 600 USD/tấn. Nếu so với đỉnh điểm cách đây một tháng, giá gạo đã giảm khoảng 40 USD/tấn. Điều gì làm giá gạo giảm nhanh và liệu sẽ còn kéo dài?	
THỊ TRƯỜNG XUẤT KHẨU GẠO CÒN NHIỀU BIẾN ĐỘNG	19 09 2023
Theo Hiệp hội Lương thực Việt Nam, tuần qua giá gạo xuất khẩu của Việt Nam và các nước trên thế giới đều ghi nhận sự sụt giảm nhẹ. Từ đó kéo theo giá lúa trong nước cũng giảm nhẹ do thị trường giao dịch chậm.	

1 2 3 ...359 360 361

Hình 1.2 Giao diện website thu thập dữ liệu

Dựa vào giao diện trên website Hình 1.2, ta có thể lập một quy trình thu thập dữ liệu như sau:

Bước 1. Mở trình duyệt truy cập trang bằng url

Bước 2. Thu thập dữ liệu (các trường thông tin tiêu đề, tóm tắt, ngày, nội dung...)

Bước 3. Trường hợp đã thu thập hết dữ liệu của trang thực hiện:

- Nếu còn trang tiếp theo chuyển trang và quay lại **Bước 1**
- Nếu hết trang → **Bước 4**

Bước 4. Đóng trình duyệt và lưu dữ liệu

Quá trình trên có thể mất một khoảng thời gian, để giảm thiểu thời gian chúng ta áp dụng kỹ thuật xử lý đa luồng ở **Bước 1**. Bằng cách mở cùng lúc nhiều trình duyệt, mỗi trình duyệt sẽ truy cập vào nhiều trang khác nhau để cùng lúc thu thập

1.2.2. Dữ liệu thu được

Số lượng tin tức thu về được khoảng 3544 mẫu dữ liệu, bao gồm các trường thông tin như Hình 1.3:



Hình 1.3 Các trường dữ liệu lấy từ website

Các trường thông tin được thu thập ứng với Bảng 1.1 sau:

Bảng 1.1: Thông tin dữ liệu được thu thập

Trường dữ liệu	Mô tả	Kiểu
Tiêu đề	Tiêu đề tin tức	Văn bản
Tóm tắt	Tóm tắt nội dung tin tức	Văn bản
Nội dung	Nội dung toàn văn của tin tức	Văn bản
Ngày	Ngày bài báo được công bố trên website	Ngày
Nguồn	Liên kết đến nơi bài báo được thu thập	Văn bản

1.3. GẮN NHÃN

1.3.1. Phương pháp gắn nhãn

Dữ liệu đóng vai trò rất lớn đến hiệu suất của mô hình AI. Tập trung nhiều hơn vào dữ liệu khi các thuật toán, mô hình dùng tới giới hạn. Trong một dự án học sâu, thời gian dành nhiều nhất là cho công việc gắn dữ liệu, chúng phụ thuộc vào:

- Nguồn lực con người
- Kiến thức về lĩnh vực đó
- Độ lớn của dữ liệu
- Chất lượng dữ liệu
- Tính bảo mật

Vì vậy cần xây dựng một chính sách phù hợp và khoa học để mô hình có thể đạt được hiệu suất cao nhất. Theo công trình nghiên cứu [4] biến động giá cả của nông sản nói chung dựa trên nhiều yếu tố như: quan hệ cung cầu, chi phí sản xuất, đầu cơ, dịch bệnh, thiên tai, khủng hoảng, sản lượng, chất lượng chính sách... Nhưng để thuận tiện cho việc gắn nhãn, ta sẽ quy về một số chủ đề lớn như: biến động giá, thị trường, chính sách, nội tại và ngoại vi.

Mỗi tin tức sẽ được gắn nhãn theo Bảng 1.2. Quá trình gắn nhãn sẽ dựa vào các thông tin đặc trưng có trong bài báo. Các thông tin và cách gắn nhãn sẽ được dựa theo hướng dẫn sau:

Bảng 1.2: Các nhân tố ảnh hưởng và loại nhãn ứng với các chủ đề lớn

		<u>NHÃN ĐƯỢC GẮN</u>			
		0	1	2	3
<u>NHÂN TỐ TÁC ĐỘNG</u>	Biến động giá	Không có thông tin	Giảm	Ổn định	Tăng
	Thị trường		- Cung tăng - Cầu giảm - Cung vượt cầu	Cung cầu cân bằng	- Cầu tăng - Cung giảm - Cầu vượt cung
	Chính sách		Đối nội	Đối ngoại	Khác
	Nội tại		Sản lượng	Chất lượng	Chi phí sản xuất
	Ngoại vi		Dịch bệnh	Thiên tai	Khủng hoảng

- **Biến động giá:** Bài báo có đề cập tường minh đến giá tăng/giảm/ổn định một cách rõ ràng ta sẽ gắn nhãn tương ứng.
- **Nhân tố thị trường:** Bài báo có đề cập tường minh đến cung cầu tăng/giảm/ổn định một cách rõ ràng ta sẽ gắn nhãn tương ứng. Nếu không đề cập đến cung cầu thị trường thì cũng có thể dựa vào số lượng và giá trị xuất/nhập khẩu của nông sản để xác định cung cầu của thị trường, thị trường cạnh tranh nhau.
- **Nhân tố chính sách/quy định:** được chia thành 3 kiểu tác động
 - Chính sách/quy định liên quan đến một nước cụ thể, ví dụ tiền tệ, lạm phát, thuế, chính sách kinh tế... của mỗi nước sẽ được gắn nhãn “1”.
 - Chính sách/quy định mà các nước phải tuân thủ theo: ví dụ hiệp định thương mại, hợp tác quốc tế, liên doanh liên kết doanh nghiệp... cần tuân thủ theo quy định chung được sẽ gắn nhãn “2”.
 - Chính sách/quy định có tính thời điểm cụ thể, ví dụ: tác mua nông sản dự trữ, giải cứu nông sản, điều chỉnh giá trần/sàn, ... để giải quyết tình huống nhất thời sẽ được gắn nhãn “3”.
- **Nhân tố nội tại:** các yếu tố bắt nguồn từ chính sản phẩm nông sản:
 - Liên quan đến sản lượng, năng suất, bội thu... sẽ được gắn nhãn “1”

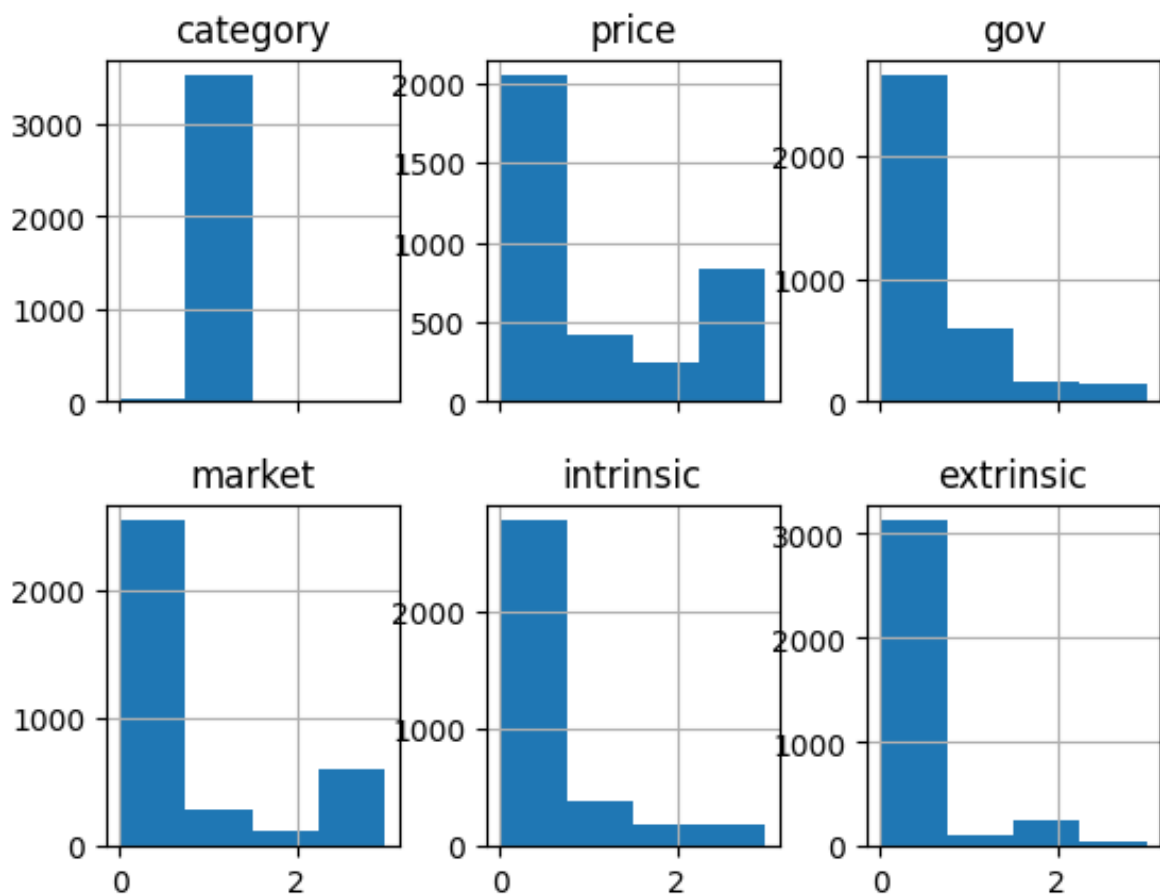
- Liên quan đến chất lượng sản phẩm nông sản sẽ được gắn nhãn “2”
- Các chi phí sản xuất liên quan đến sản xuất nông sản như phân bón, công nghệ kỹ thuật, diện tích đất hay cơ sở vật chất sẽ được gắn nhãn “3”
- **Nhân tố ngoại vi:**
 - Các loại dịch bệnh ảnh hưởng đến người mua hoặc đến nông sản như bệnh vàng lúa, Covid-19... sẽ được gắn nhãn “1”
 - Các loại thiên tai, thời tiết ảnh hưởng đến nông sản: Bão, ngập mặn... hoặc côn trùng sẽ được gắn nhãn “2”
 - Các xung đột giữa các nước: chiến tranh vũ trang, chiến tranh thương mại, cấm vận... hoặc an ninh lương thực, dầu cơ sẽ được gắn nhãn “3”

Nếu bài báo không đề cập đến thông tin của chủ đề cần gắn nhãn, ta gắn nhãn là “0” ứng với không có thông tin/không xác định.

1.3.2. Kết quả đánh nhãn

Kết quả đánh nhãn chỉ đối với nông sản lúa gạo, phân bố nhãn của các chủ đề lớn về nông sản như Hình 1.4.

Ngoài mục “Category” chỉ có một loại nông sản là lúa gạo, dễ dàng nhận ra rằng nhãn dữ liệu đa phần tập trung ở nhãn “0” (không có thông tin về chủ đề đó). Sự chênh lệch này cũng dễ hiểu bởi vì với mỗi bài báo được gắn nhãn sẽ chủ đề cụ thể, không phải bài báo nào cũng có đa chủ đề nên các hạng mục còn lại sẽ được gắn nhãn “0” nên gây ra sự chênh lệch.



Hình 1.4 Biểu đồ phân bố nhãn của mỗi lớp

Kết quả này có thể gây ra khó khăn trong huấn luyện, bởi sự chênh lệch quá lớn có thể làm cho mô hình mất đi tính tổng quát do tập trung vào số dữ liệu có nhãn nhiều hơn.

CHƯƠNG 2: XÂY DỰNG MÔ HÌNH HỌC SÂU

2.1. CƠ SỞ LÝ THUYẾT HỌC SÂU

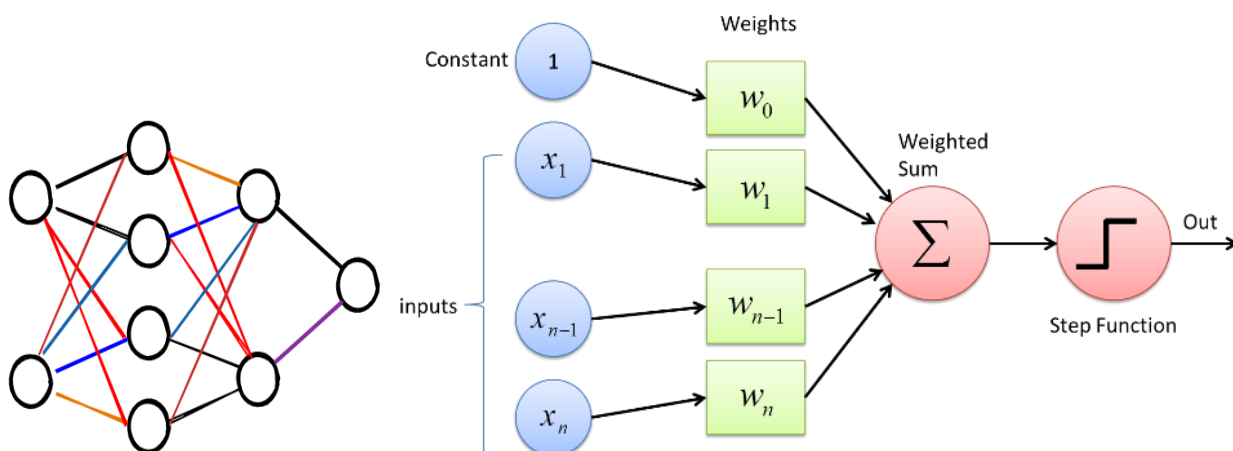
Học sâu mô phỏng cách tiếp cận nhiều lớp của mạng lưới thần kinh nơ-ron. Với cách tiếp cận học máy truyền thống, các đặc trưng được xác định và trích xuất thủ công hoặc bằng phương pháp lựa chọn đặc trưng (feature selection). Tuy nhiên, trong các mô hình học sâu, các đặc trưng được học hoặc trích xuất một cách tự động, đạt được độ chính xác và hiệu năng tốt hơn. Học sâu có lịch sử phát triển khá lâu nhưng chưa thể thay thế được học máy bởi:

- Phần cứng (Ram, GPU, CPU, etc.) thời điểm đó chưa cho phép triển một mạng sâu
- Số lượng dữ liệu còn khá ít để phát huy ưu điểm của học sâu

Theo thời gian 2 yếu tố trên ngày càng phát triển, học sâu bắt đầu trở thành giải pháp tối ưu để giải quyết các bài toán phức tạp. Nhìn chung, các tham số của mô hình phân loại của học sâu cũng được tự động cập nhật từ dữ liệu như học máy. Học sâu hiện tại đang cung cấp các giải pháp tốt nhất cho những vấn đề trong nhiều lĩnh vực, đặc biệt trong ngôn ngữ tự nhiên.

2.1.1. Mạng Kết Nối Đầy Đủ

Tầng kết nối đầy đủ (FC) nhận đầu vào là các dữ liệu đã được trích xuất đặc trưng trước đó. Mỗi đầu vào (hay nơ-ron) trước đó được kết nối đến tất cả nơ-ron phía sau như Hình 2.1 (trái). Tùy vào thiết kế mà số lượng nơ-ron sẽ được thay đổi để phù hợp với bài toán, số lượng nơ-ron trong mạng lớn tỉ lệ thuận với số tham số mô hình. Nếu mạng có nhiều tham số có thể học được những tác vụ khó hơn nhưng ngược lại sẽ đối mặt với một số vấn đề như thời gian huấn luyện, triệt tiêu (hoặc bùng nổ) gradient, quá khớp...

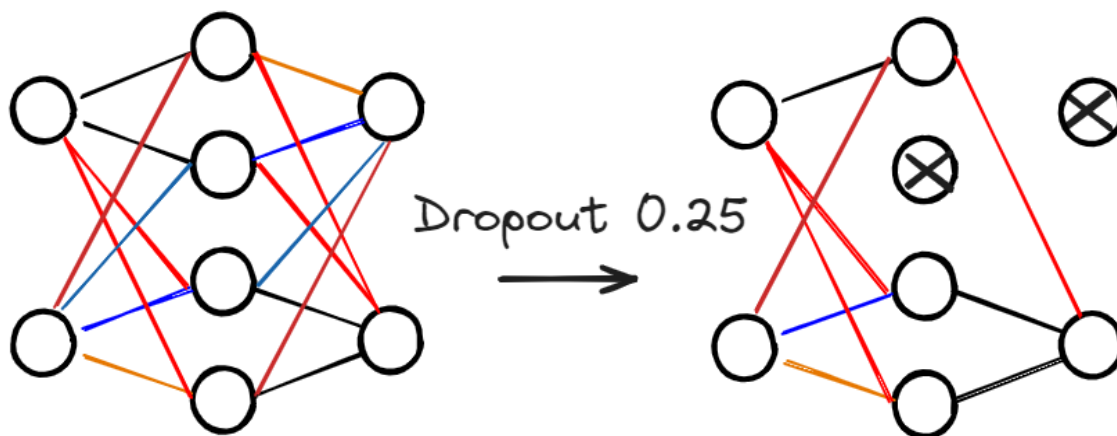


Hình 2.1 Minh họa mạng kết nối đầy đủ (trái) và cấu trúc một nơ-ron trong mạng (phải)

Xét trên một nơ-ron Hình 2.1 (phải) Output của nơ-ron đó sẽ bằng tổng các Input (nơ-ron trước đó) sẽ nhân cho trọng số trước khi qua một hàm kích hoạt.

2.1.1.1. Dropout

Dropout là một trong nhiều kỹ thuật Regularization trong học máy và cũng là một trong những kỹ thuật đơn giản để triển khai nhất và được dùng phổ biến. Về cơ bản, Dropout sẽ ngẫu nhiên bỏ đi một tỉ lệ các nơ-ron trong mạng nhằm giảm độ phức tạp của mô hình (tránh hiện tượng quá khớp). Tổng lượng thông tin ban đầu so với lúc sử dụng Dropout cũng giảm đi bằng với tỉ lệ Dropout.



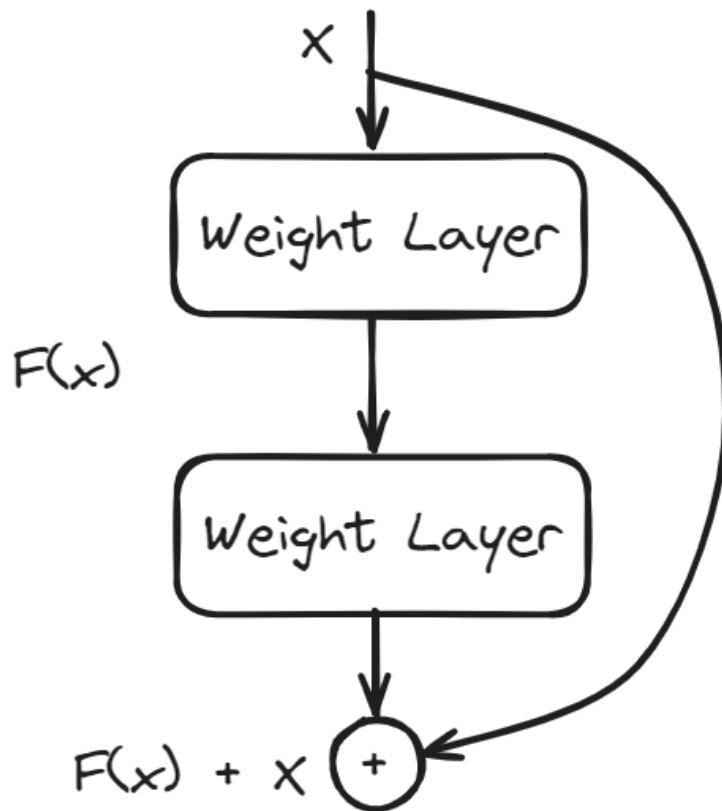
Hình 2.2 Mạng trước (trái) và sau khi áp dụng (phải) dropout 0.25

Để cân bằng mỗi nơ-ron sẽ nhân với một lượng tương ứng:

$$\text{scale} = \frac{1}{1 - \text{Dropout rate}} \quad (1)$$

2.1.1.2. Skip Connection

Skip connection là một kỹ thuật được giới thiệu bởi Kaiming He [5] bằng cách tạo một kết nối trực tiếp để bỏ qua một số lớp huấn luyện không cần thiết Hình 2.3.



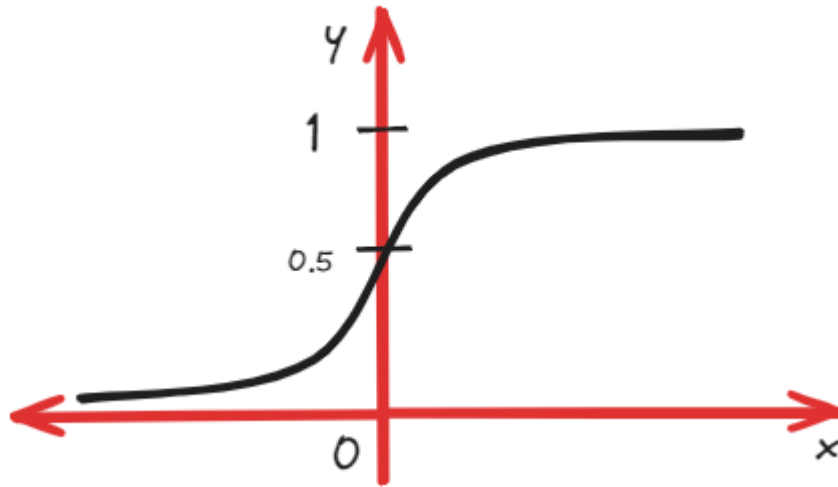
Hình 2.3 Thiết kế kiến trúc mạng nơ-ron có skip connection

Với lối tắc này có thể giúp các lớp cao hơn học được đặc trưng quan trọng hơn, các tham số dư thừa sẽ được bỏ qua để tránh bùng nổ (hoặc mất) gradient.

2.1.2. Hàm Kích Hoạt

Sức mạnh của mạng nơ-ron đến từ các hàm kích hoạt, chúng cho phép mô hình học được các mối quan hệ phức tạp phi tuyến giữa Input và Output.

2.1.2.1. Sigmoid



Hình 2.4 Đồ thị biểu diễn hàm sigmoid

Hàm kích hoạt sigmoid có dạng:

$$\sigma(x) = \frac{1}{1+e^{-x}} \quad (2)$$

Tính chất:

- $\sigma(x) \in [0,1]$
- Giá trị đạo hàm tối đa $= \frac{\max(\sigma')}{\max(\sigma)} = \frac{1}{4}$

2.1.2.2. Softmax

Hàm softmax dùng để tính xác suất tương tự như sigmoid nhưng trong trường hợp có n tham số (so với 1 của sigmoid), thường được ứng dụng trong phân loại nhiều lớp

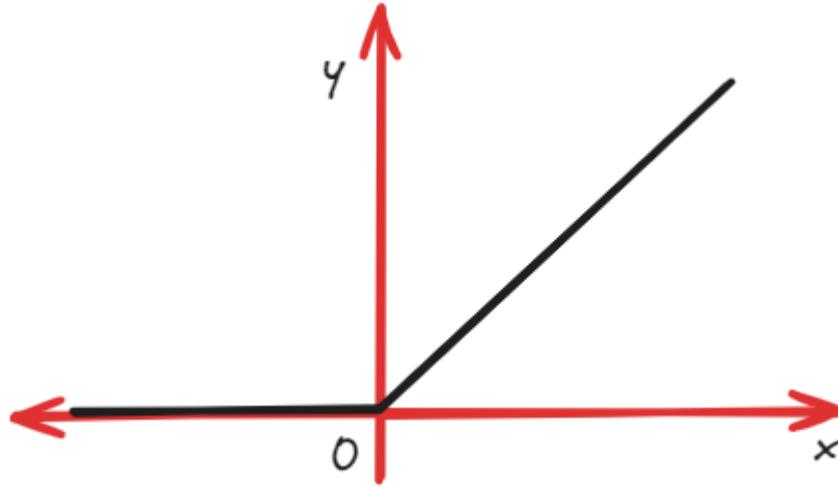
$$\text{softmax}(z)_i = \frac{e^{z_j}}{\sum_{i=1}^n e^{z_i}} \quad (3)$$

Tính chất:

- $\text{softmax}(z)_i \in [0,1]$

- $\sum_{i=1}^n \text{softmax}(z)_i = 1$

2.1.2.3. *ReLU*



Hình 2.5 Đồ thị biểu diễn hàm **ReLU**

ReLU có công thức như sau:

$$g(x) = \max\{0, x\} \quad (4)$$

ReLU dễ tối ưu vì chúng gần giống đơn vị tuyến tính, ReLU sẽ cho kết quả bằng 0 trên một nửa miền xác định của hàm. Điều này khiến đạo hàm lan truyền qua đơn vị này giữ nguyên độ lớn nếu đơn vị này còn hoạt động. Đây là hàm kích hoạt mặc định được khuyến cáo sử dụng trong hầu hết các mạng nơ-ron lan truyền thuật.

2.1.3. Học Dựa Trên Gradient

Việc thiết kế và huấn luyện mạng nơ-ron không khác nhiều so với huấn luyện mô hình học máy bằng Gradient Descent. Khác biệt lớn nhất giữa mô hình tuyến tính khác và mạng nơ-ron là tính phi tuyến của mạng nơ-ron thường được huấn luyện bằng các bộ tối ưu lặp dựa trên gradient để hướng đến tối ưu hàm mục tiêu, thay vì giải phương trình tuyến tính thông thường (hoặc sử dụng các thuật toán tối ưu lồi) để đảm bảo tính hội tụ.

Mạng nơ-ron thường được thiết kế nhiều tầng nên sẽ sử dụng một kỹ thuật lan truyền ngược để cập nhật lại trọng số. Là giải thuật cốt lõi của mạng nơ-ron để thực thi tính toán ngược và tối ưu với đạo hàm. Nhưng kiến trúc mạng ngày nay càng lúc được thiết kế càng sâu nên nảy sẽ nảy sinh vấn đề như: triệt tiêu gradient hoặc bùng nổ gradient. Trong quá trình tối ưu bằng cách đi ngược hướng đạo hàm:

$$w := w - \eta \times L' \quad (5)$$

Giá trị đạo hàm sẽ trở nên nhỏ hơn sau mỗi lần đạo hàm qua các lớp nơ-ron khác nhau (quá trình lan truyền ngược). Sau n lớp như vậy làm cho $\eta \times L' \approx 0$ (hoặc rất lớn tùy vào hàm mất mát L) dẫn đến quá trình cập nhật trọng số (5) không thay đổi (hoặc rất lớn) làm mất thông tin đạo hàm.

2.1.4. Các vấn đề gặp phải khi thiết kế mô hình

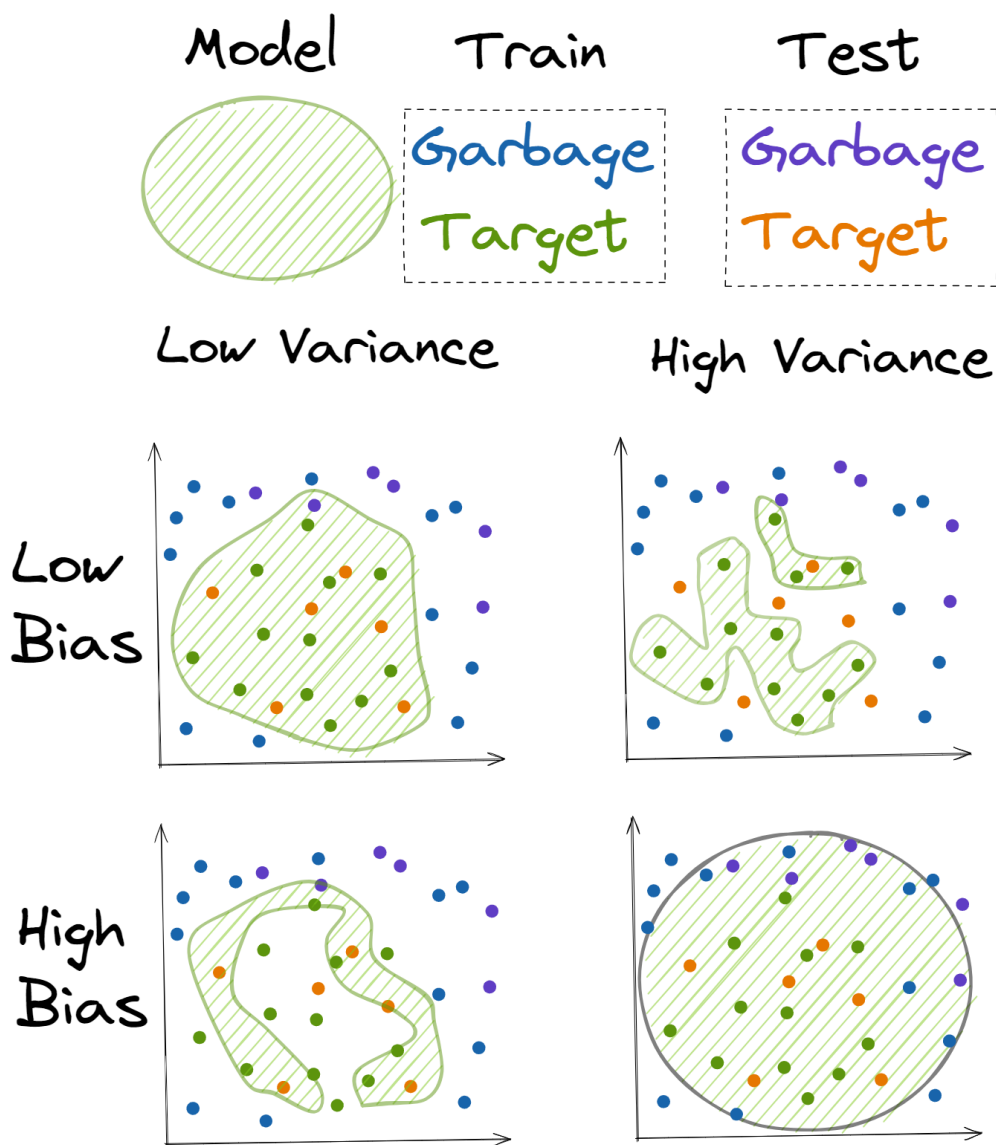
Một yếu tố then chốt khác trong mạng nơ-ron làm xác định kiến trúc của mạng. Kiến trúc mà ta thiết kế đề cập đến cấu trúc tổng thể của mạng: mạng cần bao nhiêu đơn vị và các đơn vị này sẽ kết nối với nhau như thế nào. Mạng nơ-ron từ lâu được xem như hộp đen, tức không thể giải thích tại sao nó lại đưa ra kết luận như vậy so với học máy có hàm mục tiêu được thiết kế rõ ràng có thể giải thích kết quả đầu ra. Như vậy chỉ có thể tìm thấy kiến trúc mạng lý tưởng thông qua thực nghiệm bằng cách quan sát sai số trên tập kiểm định.

Về mặt tổng quan, chúng ta vẫn thể kiểm soát kích thước mô hình, mô hình càng lớn (nhiều tham số) thì khả năng học được các tác vụ khó và ngược lại. Từ cách thiết kế mô hình này, ta có hai khái niệm được sinh ra là bias và variance.

2.1.4.1. Sự đánh đổi giữa bias và variance

khi mô hình hóa một mối quan hệ giữa các biến số, mô hình có thể sai khác với giá trị thực (giá trị cần mô tả nhưng không biết), sai khác này gọi là bias. Nếu muốn giảm

bias chúng ta có thể xây dựng mô hình phức tạp hơn, nhưng điều này lại chứa đựng rủi ro do mô hình được thiết lập dựa trên dữ liệu hiện có. Nếu dữ liệu hiện có không phản ánh hết thực tế thì mô hình phức tạp dựa trên dữ liệu này lại dẫn đến phương sai lớn, gọi là Variance. Bias và Variance luôn đi ngược nhau và nhiệm vụ là phải tìm được điểm cân bằng giữa chúng. Nếu Bias cao ta sẽ bị vị khớp – Underfitting, ngược lại nếu Variance cao sẽ Overfitting. Trong đó Chữ Under và Over mô tả mức độ phức tạp của mô hình, chữ Fit mô tả việc mô hình được xây dựng từ dữ liệu.



Hình 2.6 Sự đánh đổi giữa bias và variance

Từ Hình 2.6, có 4 trường hợp:

- Khi mô hình có Bias và Variance đều thấp thì Model được huấn luyện rất tốt hay Fitting
- Khi Bias cao và Variance thấp lúc này mô hình đang quá khớp với dữ liệu
- Ngược lại, với Bias thấp và Variance cao là hiện tượng vị khớp.
- Bias và Variance đều cao có thể có nhiều lý do ngoài mô hình (ví dụ như dữ liệu không tốt).

Để có thể xác định được khi nào mô hình đang Bias hoặc Variance, ta có thể dựa vào

- Tập huấn luyện để xác định Bias của mô hình
- Tập đánh giá (hoặc kiểm định) để xác định mô hình có Variance hay không.

2.1.4.2. *Vị khớp – Underfitting*

Dấu hiệu nhận biết vị khớp là khi sai số huấn luyện cao nhưng sai số trên tập đánh giá thấp. Có nhiều nguyên nhân gây ra hiện tượng này như:

- Mô hình chưa đủ độ phức tạp cần thiết để học được trên dữ liệu.
- Không đủ dữ liệu.

Có thể khắc phục bằng cách:

- Thu thập thêm dữ liệu.
- Sử dụng hoặc thiết kế mô hình có độ phức tạp (nhiều tham số) hơn.

2.1.4.3. *Quá khớp – Overfitting*

Khác với vị khớp, quá khớp xảy ra khi mô hình học quá tốt và cho kết quả rất tốt trên tập huấn luyện, nhưng sai số trên tập đánh giá cao. Một số nguyên nhân gây ra như:

- Dữ liệu ít hoặc chưa tổng quát
- Mô hình quá phức tạp

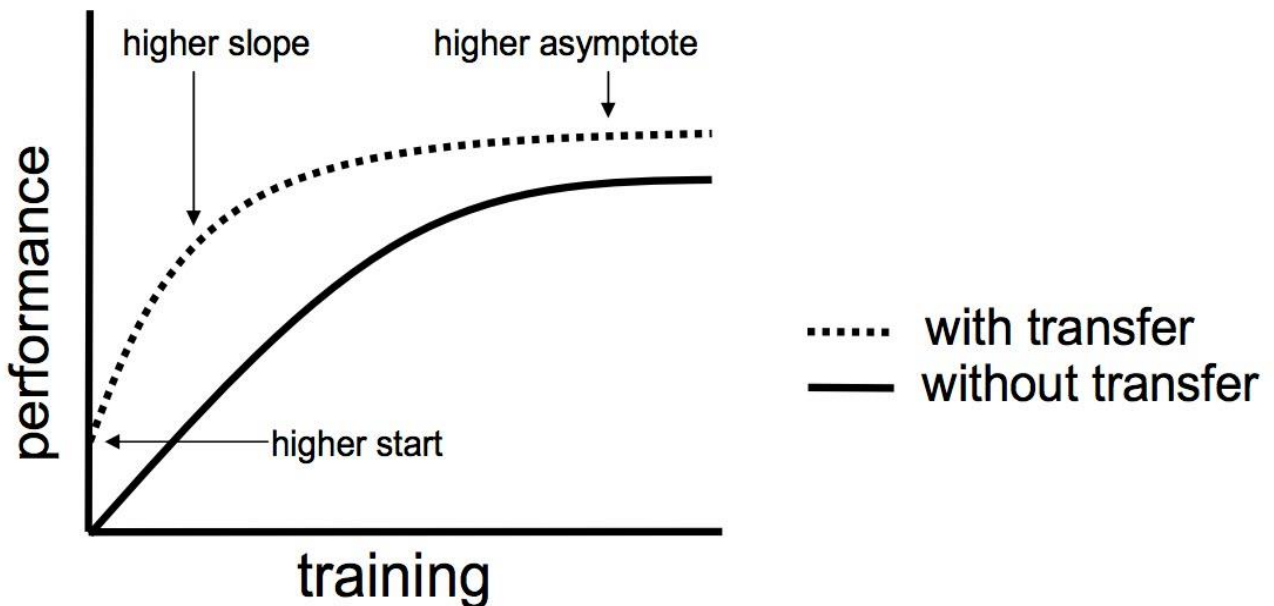
Có thể khắc phục bằng cách:

- Sử dụng kỹ thuật Regularization (Dropout, l_1 , l_2 ...)
- Thêm nhiễu vào dữ liệu bằng các kỹ thuật làm giàu dữ liệu
- Thu thập thêm dữ liệu

Quá khớp và vị khớp đều khiến mô hình có độ chính xác kém. Nhưng hiện nay, vấn đề phổ biến nhất xuất hiện là quá khớp khi mô hình càng lúc càng trở nên mạnh mẽ.

2.1.5. Học Chuyển Giao

Học chuyển giao được Lorien Pratt thực nghiệm và giới thiệu vào năm 1998 [6] đã hiện thực hóa ý tưởng về chuyển giao tri thức giữa các mô hình như giữa con người với nhau. Một mô hình đã có khả năng tận dụng lại các tri thức đã huấn luyện trước đó và cải thiện lại trên tác vụ phân loại của nó. Nhưng phải đến những năm gần đây nhờ có sự bùng nổ về thông tin dữ liệu, học chuyển giao trở thành xu hướng. Cùng với đó, ngày càng có nhiều các mô hình pre-trained có chất lượng tốt và độ chính xác cao hỗ trợ rất nhiều trong quá trình huấn luyện.



Hình 2.7 Khi không áp dụng (nét liền) và áp dụng học chuyển giao (nét đứt) ([7])

Hình 2.7 cho thấy một số lợi thế của học chuyển giao:

- Mô hình sẽ có điểm khởi đầu của cho độ chính xác tốt hơn so với huấn luyện từ đầu và mất ít số epoch.
- Hiệu năng mô hình cũng tăng nhanh hơn và cao hơn so với không học chuyển giao

Ngoài ra khi tập dữ liệu có kích thước quá nhỏ và khó có thể tìm kiếm và mở rộng dẫn đến mô hình được huấn luyện không có hiệu suất cao. Bằng cách tận dụng lại tri thức từ cả 2 nguồn dữ liệu cũ và mới hiệu suất mô hình sẽ tăng.

2.2. XỬ LÝ NGÔN NGỮ TỰ NHIÊN

2.2.1. Dữ Liệu Văn Bản Tiếng Việt

Bản chất của ngôn ngữ viết (văn bản) là dòng sự kiện các ký tự tiếp nối nhau và bổ sung cho nhau để diễn giải dòng suy nghĩ của con người. Các từ thông thường không đứng độc lập mà chúng sẽ đi kèm với các từ khác để liên kết mạch lạc thành một câu. Hiệu quả biểu thị nội dung và truyền đạt ý nghĩa sẽ lớn hơn so với từng từ đứng độc lập. Tùy vào ngữ cảnh, lĩnh vực khác nhau có thể ảnh hưởng rất lớn trong việc giải thích ý nghĩa của từ có thể khác nhau.

2.2.2. Embedding Véc-tơ

Các đặc trưng hoặc đầu vào của bất kỳ mô hình học máy hay học sâu nào thường có hai dạng là liên tục hoặc phân loại. Các đặc trưng là số liên tục có thể là: điểm số, mức lương, hay giá tiền... Đối với đặc trưng phân loại có thể là xếp loại học lực, bậc lương, loại sản phẩm... Một embedding sẽ là một loại biểu diễn số cho đặc trưng phân loại, nghĩa là véc-tơ số được học trong quá trình đào tạo mô hình. Số lượng các giá trị này, còn được gọi là chiều embedding, có thể khác nhau tùy theo mô hình.

Mô hình học sâu không thể tiếp nhận thông tin như cách con người tiếp nhận thông tin là văn bản, vì mô hình chỉ xử lý được trên dữ liệu số. Do đó cần phải biểu diễn văn bản bằng giá trị số để mô hình ngôn ngữ có thể học được, ngoài ra chúng cũng phải thể hiện

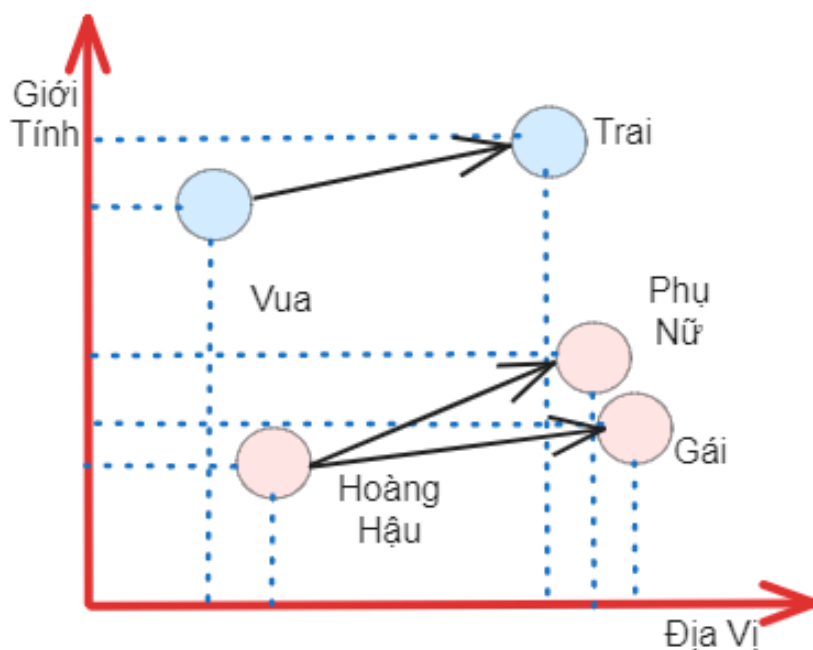
toàn bộ ý nghĩa của văn bản đó. Theo [8] Có nhiều cách biểu diễn thành các véc-tơ từ khác nhau có thể kể đến như: Count-based và Word Embedding.

2.2.2.1. *Count-based*

Count-based: dựa trên số lượng từ, hoặc số liệu thống kê dựa trên tần suất xuất hiện đồng thời giữa các từ để biểu diễn từ. Phương pháp này có hạn chế khả năng diễn giải ý nghĩa từ cũng như mối quan hệ giữa các từ xung quanh. Có thể kể đến như: One-hot véc-tơ, Bag-of-Word, hay TF-IDF.

2.2.2.2. *Word Embedding*

Word Embedding: là cách biểu diễn các từ thành các véc-tơ số thực trong không gian nhiều chiều nhưng vẫn giữ lại mối quan hệ ngữ nghĩa dựa trên giá trị của véc-tơ trong không gian. Nguyên tắc cơ bản đằng sau dựa trên giả thuyết phân bố được học trong quá trình huấn luyện là các từ xuất hiện trong các ngữ cảnh tương tự có xu hướng có ngữ nghĩa tương tự.



Hình 2.8 Minh họa word Embedding trong không gian 2 chiều.

Giả sử một Embedding véc-tơ biểu diễn ngữ nghĩa trong không gian 2 chiều Hình 2.8. Tại trục thứ nhất, sẽ biểu diễn mối quan hệ địa vị giữa các từ (Từ King và Queen sẽ gần nhau theo trục ngang). Trục thứ hai, sẽ biểu diễn mối quan hệ giới tính (các từ Man và Woman, Girl sẽ có xu hướng gần nhau hơn).

Có thể kể đến như: Word2Vec, Glove, BERT. Đặc biệt là BERT embedding được sử dụng rộng rãi trong nhiều tác vụ NLP và đã cho thấy được sự hiệu quả trong quá trình thực nghiệm [9].

2.3. MÔ HÌNH TRANSFORMER

Transformer xuất phát từ bài báo [10] được giới thiệu bởi Vaswani vào năm 2017 trở thành bài báo có sức ảnh hưởng lớn đến toàn bộ ngành khoa học máy tính. Kể từ đó nhiều mô hình mạnh mẽ ra đời như T5, Vision Transformer, hay GPT.

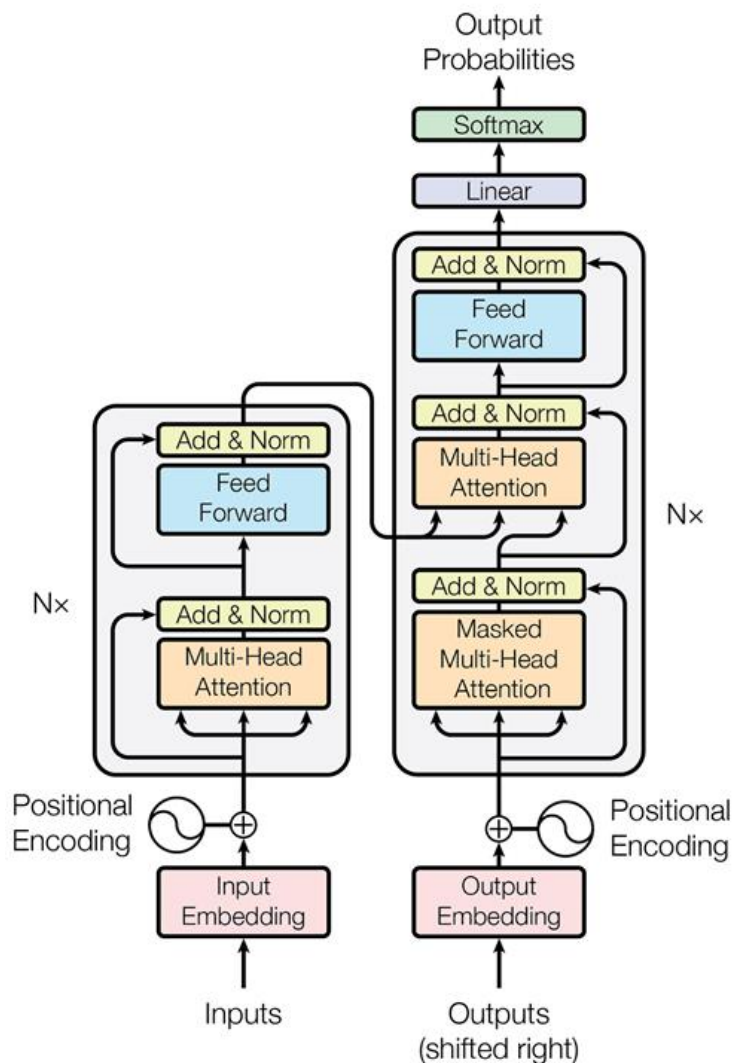
Ngoài những thành phần cơ bản của kiến trúc seq2seq Transformer giới thiệu 3 kỹ thuật mới:

- Scale dot-product attention: kỹ thuật cho phép Transformer mô tả ngữ cảnh văn bản tốt hơn từ cả 2 chiều.
- Multi-head attention: áp dụng cơ chế Scale dot-product attention nhiều lần giúp trích xuất đặc trưng tốt hơn.
- Positional encoding: mỗi từ trong câu giờ đây sẽ được biểu diễn thêm bởi thông tin vị trí của từ.

Kiến trúc Transformer được chia thành hai pha gồm:

- **Khối Encoder:** 6 lớp xếp chồng lên nhau, mỗi lớp được cấu tạo từ 2 lớp con:
 - **Multi-head self-attention** giúp học các ngữ nghĩa các từ xung quanh vào các véc-tơ biểu diễn từ.
 - **Fully connected feedforward network** sẽ học những đặc trưng từ giúp rút trích thông tin.
- **Khối Decoder:** 6 lớp xếp chồng lên nhau, gồm 3 lớp con:
 - 2 lớp tương tự như khối Encoder

- **Masked Multi-head Self-Attention** thực hiện attention với đầu ra của từng từ sau khi qua encoder, đóng vai trò lấy véc-tơ ngữ cảnh và căn chỉnh giữa chuỗi nguồn và chuỗi đích. Ngoài ra, masked (che) thông tin để mô phỏng hành vi dự đoán của mô hình ngôn ngữ để tránh lấy thông tin từ tương lai khi huấn luyện.



Hình 2.9 Kiến trúc Transformer gồm khối Encoder (trái) và Decoder (phải) ([10])

Như vậy, khối Encoder có nhiệm vụ học ngữ nghĩa ngôn ngữ và ngữ cảnh của các từ và từ xung quanh hay nói cách khác là hiểu ngôn ngữ. Khối Decoder được huấn luyện bằng cách các từ và cố gắng dự đoán những từ này để học cách tạo sinh ra một chuỗi các từ phù hợp cho các tác vụ như dịch hoặc sinh từ có kích thước lớn. Tùy thuộc vào loại bài

toán bài toán ta áp dụng một trong hai khối hoặc thậm trí cả hai như trong bài báo cho tác vụ dịch máy.

Đối với đề tài, ta hoàn toàn có thể không cần đến khối Decoder bởi vì mục tiêu đề tài là phân tích biến động giá dựa trên dữ liệu tin tức nên thiết yếu nhất là hiểu ngôn ngữ không phải dịch, sinh từ. Nên chỉ dùng khối Encoder sẽ đơn giản hơn, hiệu quả hơn. Một trường hợp cũng cần đáng lưu tâm là dữ liệu thu thập được vẫn còn khá hạn chế nếu huấn luyện lại từ đầu thì với dữ liệu này thì mô hình nhiều khả năng cũng sẽ không cho độ chính xác cao. Cùng với những vấn đề trên thì giải pháp sử dụng một mô hình pre-trained chỉ gồm kiến trúc Encoder để hiểu ngôn ngữ khắc phục vấn đề thiếu dữ liệu. Từ mô hình này ta sẽ tinh chỉnh lại bằng dữ liệu của đề tài để chúng học được những tác vụ mà này. Với những vấn đề được đặt ra mô hình BERT hoàn toàn đáp ứng được kiến trúc mà chúng ta mong muốn.

2.4. MÔ HÌNH BERT

2.4.1. Giới thiệu

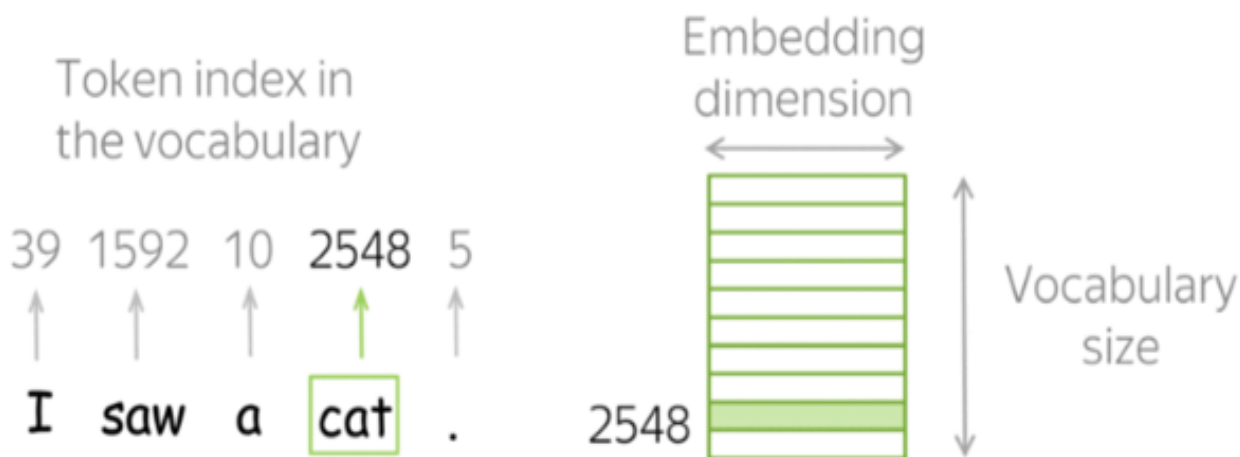
Năm 2018, Devlin xuất bản bài báo [9] đạt kết quả cao trong nhiều tác vụ khác nhau trong lĩnh vực NLP. BERT viết tắt của Bidirectional Encoder Representation from Transformer được huấn luyện trên lượng dữ liệu lớn và có thể xem là một pre-trained model thường được fine-tune cùng với một số lớp nơ-ron đơn giản để tạo nên những mô hình SOTA giải quyết nhiều bài toán khác nhau trong lĩnh NLP.

BERT về cơ bản là mô hình được huấn luyện trên khối Encoder của kiến trúc Transformer, nhưng bao gồm nhiều khối xếp chồng liên tiếp nhau (12 hoặc 24 khối tùy vào cấu hình).

2.4.2. Kiến trúc mô hình BERT

2.4.2.1. *Input Embedding*

Trong Transformer, phương pháp được dùng để embedding văn bản là lookup table như Hình 2.10.



Hình 2.10 Chuyển từ văn bản sang véc-tơ số trong Transformer.

BERT tokenizer được dùng để chuyển một từ sang véc-tơ bằng cách ánh xạ theo token id tương ứng trong từ điển. Ứng với id sẽ được chuyển thành véc-tơ thông qua một ma trận trọng số. Ma trận trọng số này sẽ được cập nhật trong quá trình training. Số lượng véc-tơ tương ứng bằng với số lượng từ trong từ điển. Chiều của mỗi véc-tơ sẽ được tùy chỉnh phù hợp với kiến trúc của mô hình ví dụ 512 với kiến trúc gốc [10], hoặc 768, 1024 lần lượt ứng với BERT_{base} và BERT_{large}.

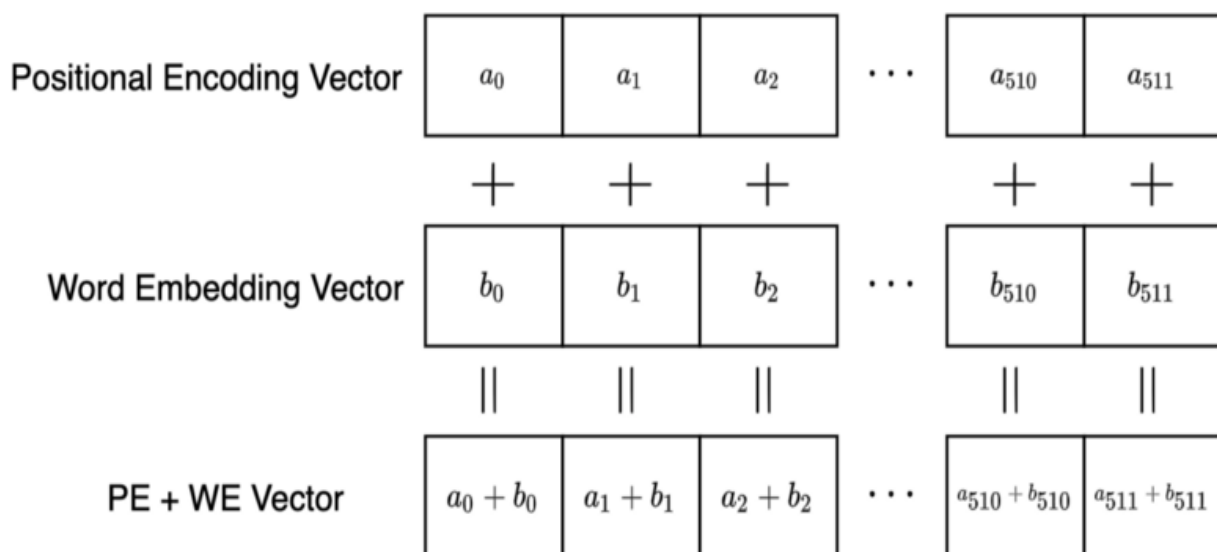
2.4.2.2. *Positional Encoding*

Một vấn đề đặt ra là các từ giống nhau sẽ có Embedding véc-tơ giống nhau. Trong trường hợp một câu có chứa nhiều Token giống nhau thì điều cần quan tâm đến vị trí của từ đó trong câu. Ví dụ:

“Con ngựa đá con ngựa đá”

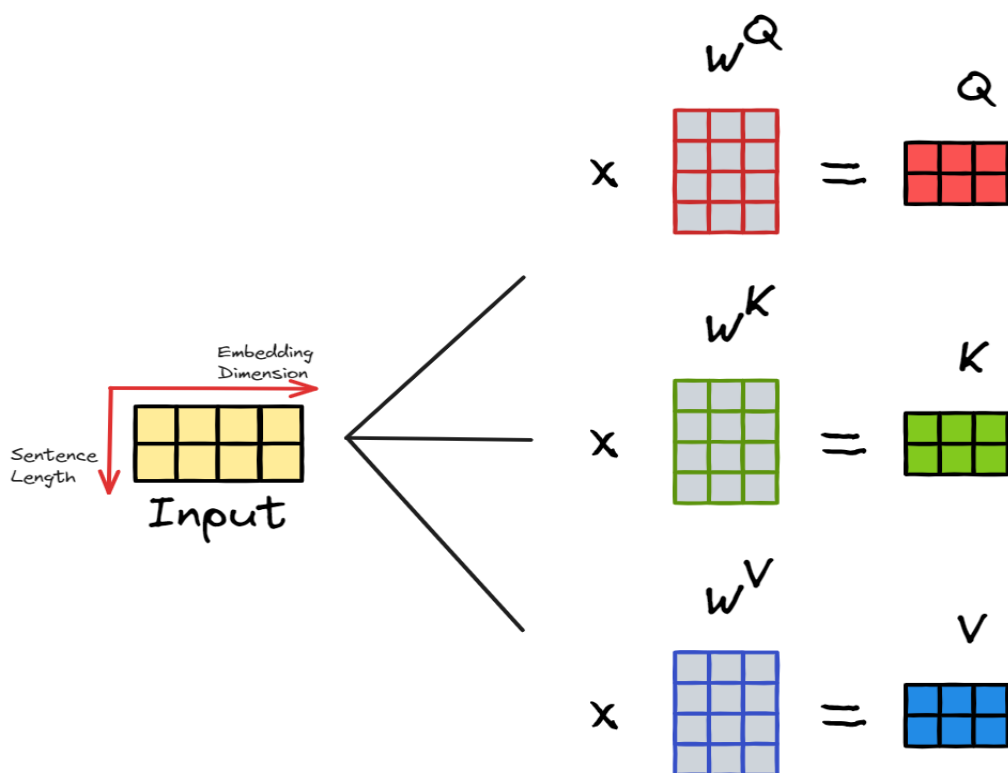
Để giải quyết vấn đề đó mô hình Transformer sử dụng thuật toán Positional Encoding chuyển vị trí tương ứng thành Véc-tơ sau đó cộng chập với Embedding Véc-tơ

của từ đó Hình 2.11. Ta sẽ thu được một Véc-tơ tổng hợp chứa thông tin về nghĩa của từ và thông tin về vị trí của từ.



Hình 2.11 Thuật toán Positional Encoding

2.4.2.3. Scaled Dot-Product Attention



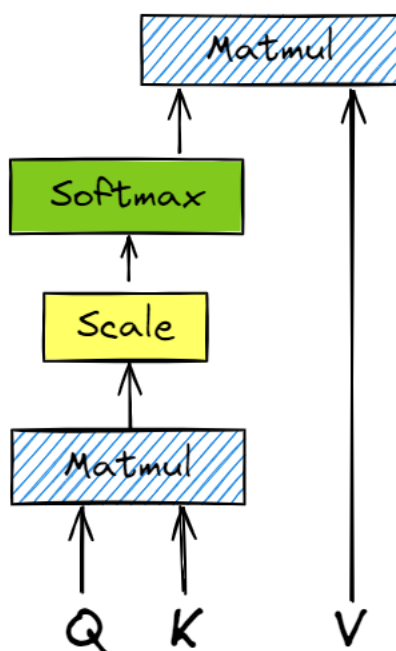
Hình 2.12 Giá trị Query – Key – Value

Scaled Dot-Product Attention là cơ chế tính toán nổi quan hệ mạnh yếu giữa từ hiện tại và xung quanh. Mượn khái niệm từ hệ thống truy vấn gồm Query Q (đại diện cho một từ) sẽ truy vấn đến Key K (là những từ được so sánh) từ đó kết hợp với Value V (giá trị đối chiếu) để trả về một giá trị mà quan tâm.

Các thành phần Q, K, V được tạo ra bằng cách ma trận đầu vào (kích thước bằng chiều embedding và số từ trong câu) nhân với một ma trận trọng số được khởi tạo ban đầu ngẫu nhiên (ma trận trọng số này sẽ cập nhật cho phù hợp trong huấn luyện). Ba thành phần trên được sử dụng để tính Scaled Dot-Product Attention như sau:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}}\right) \cdot V \quad (6)$$

Lần lượt thực hiện:



Hình 2.13 Scale dot-product attention

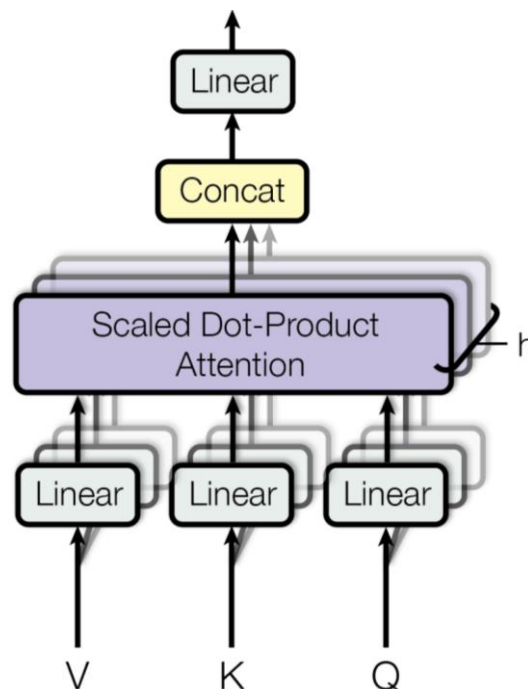
- **Bước 1:** Kết quả tích chấm $Q \cdot K^T$ (phép chiếu) thể hiện mối quan hệ giữa 2 véc-tơ giá trị càng lớn thể hiện các véc-tơ càng xa nhau (về ngữ nghĩa), kết quả phép tính sẽ được chuẩn hóa cho số chiều d của embedding (tránh số quá lớn bởi vì Softmax có đạo hàm nhỏ nhỏ gây khó cho việc huấn luyện)

- **Bước 2:** Véc-tơ sẽ qua hàm Softmax tính phần trăm (xác suất) thể hiện phần trăm đóng góp của từ đó với các từ xung quanh.
- **Bước 3:** Xác suất này sẽ nhân với V để lấy ra các giá trị tương ứng với phần trăm tương ứng.

2.4.2.4. *Multi-head Attention*

Scaled Dot-Product Attention là một phần của Multi-head Attention thay vì chỉ tính (4) một lần với mỗi từ (hoặc câu). Multi-head Attention sẽ chia 3 thành phần Q , K và V thành nhiều head (số head phải chia hết cho số chiều h của embedding)

- $Q \rightarrow Q_1, Q_2 \dots Q_d$
- $K \rightarrow K_1, K_2 \dots K_d$
- $V \rightarrow V_1, V_2 \dots V_d$.



Hình 2.14 Cơ chế Multi-head Attention ([10])

Sau đó các head tương ứng (ví dụ: Q_1 , K_1 và V_1) sẽ lần lượt thực hiện Scaled Dot-Product Attention tạo các kết quả tương ứng. Cuối cùng kết quả của mỗi head sẽ được nối lại có kích thước như cũ.

Bản chất của việc chia thành nhiều head để thực hiện nhằm mục đích để có thể trích xuất đặc trưng từ dữ liệu ở mức sâu hơn (tương tự CNN dùng nhiều lớp tích chập). Khi chia các head theo chiều embedding, cơ chế attention sẽ có thể đào sâu vào từng chiều) của embedding (mỗi chiều embedding sẽ đại diện cho một khía cạnh ý nghĩa của từ) từ đó mô hình có thể trích xuất đặc trưng tốt hơn.

2.4.2.5. *Skip connection và layer normalization*

Hai cơ chế Skip Connection và Layer normalization nhằm làm giảm số lượng tham số trong quá trình huấn luyện, cũng như chuẩn hóa để cải thiện sự ổn định của dữ liệu mỗi khi thực hiện một phép biến đổi khi qua nhiều lớp như một mạng sâu như kiến trúc Transformer (nhiều khối xếp chồng). Layer normalization còn có nhiệm vụ chuẩn hóa tương tự batch normalization nhưng layer normalization sẽ chuẩn hóa theo chiều của embedding. Layer normalization mang lại lợi thế về tốc độ, độ ổn định và hiệu suất đào tạo. Nó đã trở thành một thành phần quan trọng trong nhiều kiến trúc deep learning, đặc biệt là những kiến trúc liên quan đến dữ liệu tuần tự như văn bản [11].

2.4.2.6. *Feed Forward và các thành phần khác*

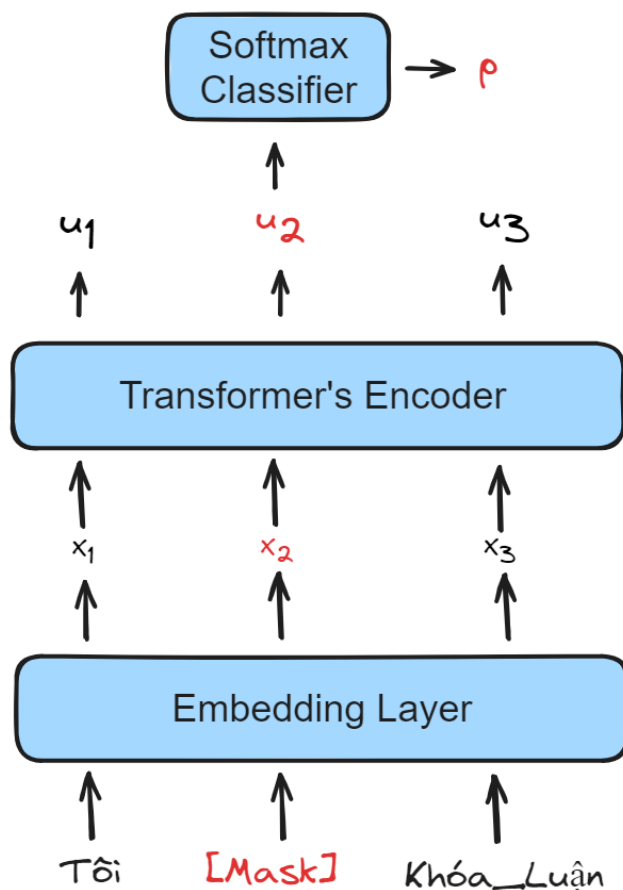
Feed forward là một mạng kết nối đầy đủ (Fully connected layer) các nơ-ron phía trước được kết nối đầy đủ với các nơ-ron ở phía sau để huấn luyện.

2.4.3. Chiến lược huấn luyện BERT

BERT dùng chung một kiến trúc với khối encoder của Transformer, và được huấn luyện trên hai tác vụ: masked language modeling (MLM) và next sentence prediction (NSP). Quá trình huấn luyện BERT diễn ra trên tập dữ liệu rất lớn và không cần nhãn.

2.4.3.1. Masked language modeling

Với mục tiêu Masked language modeling (MLM), BERT sẽ ngẫu nhiên che (mask) 15% token trong chuỗi đầu vào. Trong số này sẽ được mask (với 80% trong số đó bị thay thế, 10% được thay thế ngẫu nhiên bởi một token khác và 10% còn lại giữ nguyên), sau đó BERT được huấn luyện để dự đoán những token bị mask dựa vào ngữ cảnh của những token xung quanh.



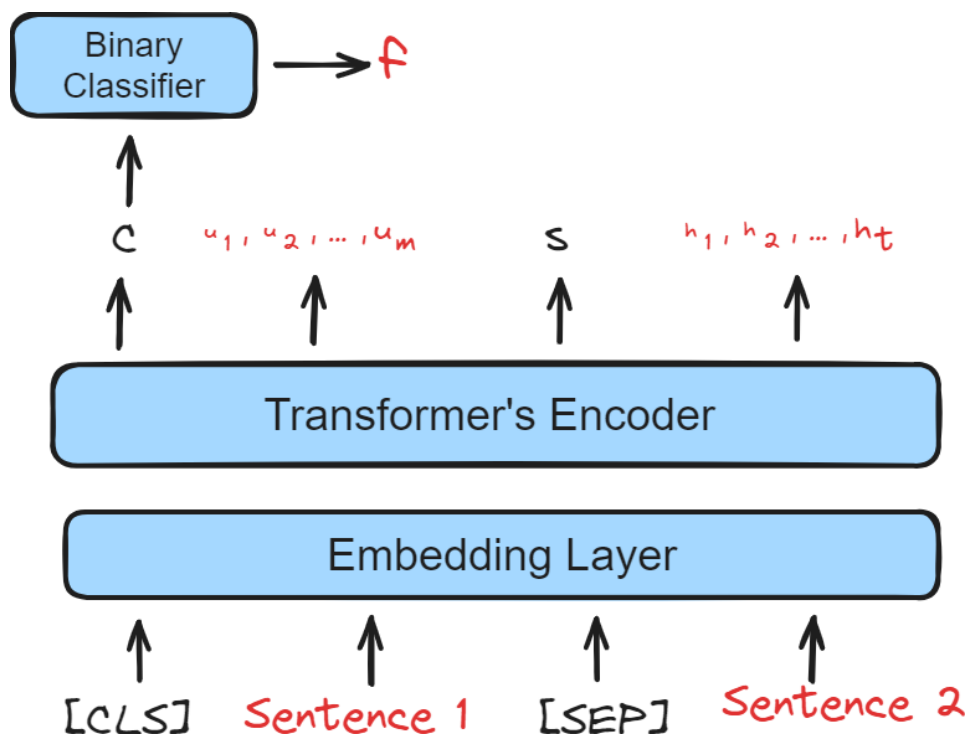
Hình 2.15 Cơ chế huấn luyện masked language modeling

Vì vậy đầu ra của BERT có thể được dùng như một lớp embedding đã được nhúng vào thông tin ngữ cảnh từ cả hai hướng nhờ vào cơ chế attention trong mỗi lớp Transformer encoder. Sau khi dữ liệu đầu vào đi qua các khối Encoder sẽ thu được các véc-tơ embedding sau cùng. Các véc-tơ embedding này tiếp tục đi qua một tầng FC để tạo ra các véc-tơ phân bố xác suất của bộ từ vựng tại vị trí tương ứng trong câu. Đây là dự đoán của BERT cho xác suất xuất hiện của từng từ tại vị trí đầu ra của mô hình. BERT sẽ

so sánh tiên đoán xác suất từ xuất hiện ở vị trí MASK và so sánh với từ tương ứng ở ground truth để xác định giá trị loss cho lần huấn luyện này và cập nhật trọng số tương ứng. Như vậy, chúng ta có thể hiểu tầng FC sau cùng chính là decoder của BERT. Sau khi đã được huấn luyện với một tập corpus tương ứng, tầng FC này sẽ được gỡ ra, phần Encoder còn lại của BERT sẽ được sử dụng như một tầng embedding để tạo ra các véc-tơ BERT-embedding cho các tài liệu khác. Đây cũng là cách BERT thường được sử dụng nhiều nhất. BERT là một mô hình ngôn ngữ lớn dạng AutoEncoding, được sử dụng để tạo ra các dạng biểu diễn cho văn bản để tiếp tục sử dụng cho tác vụ khác.

2.4.3.2. Next sentence prediction

Với mục tiêu Next sentence prediction (NSP), mô hình nhận đầu vào các cặp chuỗi ngắn cách nhau bởi token <SEP> và được huấn luyện để dự đoán đó có phải là hai chuỗi liên tiếp hay không. Mô hình sẽ chọn một số cặp câu liên tiếp nhau và một số cặp câu ngẫu nhiên với giá trị ground truth được quy định bởi token <CLS> như Hình 2.16.



Hình 2.16 Cơ chế huấn luyện next sentence prediction

Bằng cách thay đổi ý nghĩa của token <CLS> với các input tương ứng, chúng ta có thể huấn luyện BERT cho các bài toán học có giám sát khác.

2.4.4. Các mô hình pre-trained

Các mô hình pre-trained hiện nay phần lớn được huấn luyện trên dữ liệu tiếng Anh hay các ngôn ngữ gốc La-tinh, Nguyên bản của mô hình pre-trained BERT của Google được huấn luyện trên dữ liệu thuần túy tiếng anh. Đối với ngôn ngữ Việt do dữ liệu ít và đặc trưng riêng như:

- Các ký tự utf-8 để biểu diễn các dấu câu
- Mã hóa phần lớn cần nhiều byte hơn để biểu diễn (ví dụ: thesis → khóa luận)
- Ngữ pháp khác biệt có thể kể đến như thứ tự danh từ và tính từ khác nhau (ví dụ: good thesis ⇔ khóa luận tốt)
- Cách dùng từ (ví dụ: wear a hat and wear glasses ⇔ tôi đội nón và đeo kính)

Nên nếu sử dụng những mô hình pre-trained này sẽ không phù hợp (hoặc kém hiệu quả) đối với bài toán yêu cầu hiểu được ngữ nghĩa cũng như bối cảnh của Việt Nam. Nên ta sẽ sử dụng PhoBERT [12] là một dự án của VinAI, mô hình sử dụng kiến trúc của BERT nhưng mô hình sẽ được huấn luyện trên toàn bộ dữ liệu là tiếng việt.

Bảng 2.1 Các mô hình pre-trained của PhoBERT

Kiến trúc	Tham số	Dữ liệu
PhoBERT _{base}	135 triệu	20Gb văn bản từ Wikipedia và tin tức
PhoBERT _{base} (version 2)	135 triệu	20Gb văn bản từ Wikipedia và tin tức và 120Gb văn bản từ tập dữ liệu OSCAR-2301
PhoBERT _{large}	370 triệu	20Gb văn bản từ Wikipedia và tin tức

Dữ liệu pre-trained cũng bao gồm dữ liệu là tin tức, điều này mang lại nhiều lợi thế cho các tác vụ liên quan đến phân tích tin tức. kích thước của mô hình cũng bao gồm hai loại base và large tương ứng với số hidden unit lần lượt là 768 và 1024.

Ngoài ra, bộ từ điển từ vựng tiếng Việt của PhoBERT cũng vượt trội so với tiếng anh gồm 64001 token.

2.5. TỐI ƯU HÓA

Để mô hình được chọn khớp được với dữ liệu huấn luyện thì mô hình cần diễn ra quá trình huấn luyện để có thể tối ưu trọng số mô hình nhằm giảm sai số dự đoán của mô hình xuống thấp nhất có thể.

2.5.1. Hàm mất mát

Hàm mất mát (hay hàm mục tiêu) là một hàm số có được sau khi lan truyền thuận để dự đoán kết quả đầu ra. Từ đó, hàm này sẽ được lan truyền ngược để tính đạo hàm để tìm ra trọng số tối ưu cho sai số giữa dự đoán và nhãn thật.

2.5.1.1. *Cross Entropy*

Cross Entropy là một hàm mất mát được dùng phổ biến trong các bài toán phân loại. Cross Entropy đo lường sự khác biệt giữa 2 phân bố thực P và phân bố hiện tại Q để đo lường trung bình thông tin khi dùng mã hóa thông tin của phân bố Q thay cho mã hóa thông tin của phân bố P .

$$\text{Cross entropy} = - \sum_{i=1}^{\text{Class}} p_i (\log q_i) \quad (7)$$

Trong đó, q và p là 2 phân bố xác suất.

2.5.1.2. *Sigmoid Focal Loss*

Biểu đồ phân bố của nhãn Hình 1.4, cho thấy ta có số lượng nhãn bị mất cân bằng khá nhiều. Bằng cách huấn luyện thông thường dùng Cross Entropy, chúng sẽ xử lý các nhãn như nhau làm cho hàm mất mát trở nên kém hiệu quả đối với các nhãn ít hơn.

Focal Loss Cũng là một hàm mất mát, “Focal” có nghĩa là trọng tâm nói cách khác chúng sẽ đặt trọng tâm hơn cho dữ liệu mà thuật toán còn học chưa tốt (dữ liệu có ít nhãn), so với các dữ liệu đã học tốt (dữ liệu nhiều nhãn hơn) [13].

$$\text{Sigmoid Focal Loss} = -\alpha(1 - p_t)^\gamma \log(p_t) \quad (8)$$

Trong đó:

- α và γ là 2 siêu tham số
- p_t là xác suất

2.5.2. Trình tối ưu hóa Adam – Adam Optimizer

Adam là thuật toán tối ưu kết hợp [14] của 2 thuật toán khác và tận dụng ưu điểm của 2 thuật toán đó là:

- Momentum: Giúp vượt qua các cực tiểu cục bộ để đến các cực tiểu cục bộ khác tốt hơn.
- RMSProp: Giúp hội tụ nhanh hơn thay vì phải mất một khoảng thời gian để dao động xung quanh điểm hội tụ như Momentum

$$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{s_t + \epsilon}} m_t \quad (9)$$

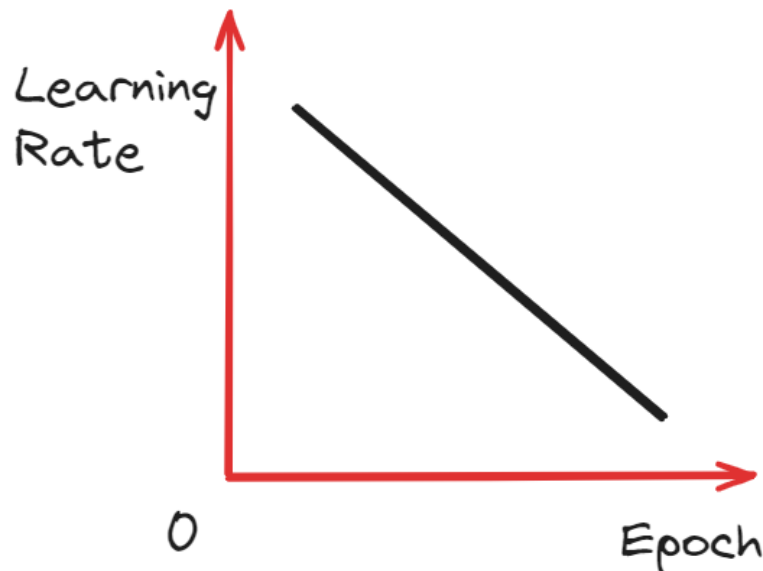
Trong đó:

- θ_t là trọng số tại thời điểm t
- η là tốc độ học (hay learning rate)
- m_t và s_t lần lượt là momentum và RMSProp tại thời điểm t

2.5.3. Linearly learning rate decay

Tốc độ học hay learning rate là một siêu tham số phản ánh mức độ cập nhật trọng số sau mỗi lần lan truyền ngược nhằm tối ưu mô hình. Giá trị của learning rate thường là một số dương nhỏ để đảm bảo khi nhân với đạo hàm giá trị cập nhật sẽ không quá lớn.

Nếu learning rate quá lớn quá trình tối ưu hóa (tìm cực tiểu) sẽ khó (hoặc không) hội tụ, ngược lại nếu quá nhỏ quá trình huấn luyện sẽ mất rất nhiều thời gian và chi phí. Một learning rate tốt sẽ đủ nhỏ để mô hình có thể hội và đủ lớn để không mất nhiều chi phí để đi đến cực tiểu. Thực tế chọn một learning rate tốt rất khó, nhưng bằng cách kiểm soát learning rate tự động giảm một cách tuyến tính về sau mỗi epoch như Hình 2.17:



Hình 2.17 Linearly learning rate decay

Bằng cách sử dụng cơ chế huấn luyện này có thể:

- Learning rate ở những epoch đầu cao có thể cải thiện tốc độ huấn luyện
- Learning rate ở những epoch cuối giảm giúp mô hình hội tụ tốt hơn

2.6. ĐÁNH GIÁ MÔ HÌNH

2.6.1. Confusion Matrix

Đối với bài toán phân loại các Output mà mô hình dự đoán đúng hoặc sai được sẽ được thống kê trên Confusion Matrix. Ma trận này sẽ được dùng để tính toán các độ đo đánh giá mô hình đã được huấn luyện có tốt hay không. Ma trận có 4 loại thông số sau:

- True Positive (TP): Số lượng dự đoán đúng
- True Negative (TN): Số lượng dự đoán đúng một cách gián tiếp

- False Positive (FP): Số lượng dự đoán sai
- False Negative (FN): Số lượng dự đoán sai một cách gián tiếp

Như chương 1 đã trình bày, bài toán của đề tài sẽ gồm 5 chủ đề với mỗi chủ đề sẽ gồm 4 nhãn. Như Confusion Matrix sẽ có dạng như sau:

Bảng 2.2 Confusion Matrix với bài toán 4 nhãn

	Nhãn	Nhãn Thực Tế				N=TP+FP
		0	1	2	3	
Mô Hình Dự Đoán	0	TN	FN	TN	TN	
	1	FP	TP	FP	FP	N ₁
	2	TN	FN	TN	TN	
	3	TN	FN	TN	TN	
M =TP + FN			M ₁			

Trường hợp trong Bảng 2.2 hiện đang xét nhãn 1, đối với các nhãn khác có thể xử lý tương tự bằng cách chọn trục dọc và ngang tương ứng

2.6.2. F-score

F-score là một độ đo cân bằng giữa độ chính xác và độ phủ của dữ liệu đo lường hiệu suất tổng thể của mô hình trong việc phân loại dữ liệu. Do tính cân bằng giữa độ chính xác và độ phủ của dữ liệu nên F-score thường được sử dụng trong các bài toán mất cân bằng dữ liệu. Vì vậy một mô hình có F-score cao cho thấy thuật toán phân loại tốt cả hai trường hợp Positive và Negative. F-score được tính bằng trung bình điều hoà của Precision và Recall:

$$F_{\beta} = (1 + \beta^2) \frac{\text{precision} \times \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}} \quad (10)$$

Trong đó:

- Precision là tỉ lệ số TP trong số những điểm được phân loại là Positive (TP + FP):

$$\text{Precision} = \frac{TP}{TP+FP} \quad (11)$$

- Recall là số điểm TP trong số những điểm thực sự là Positive (TP + FN):

$$\text{Recall} = \frac{TP}{TP+FN} \quad (12)$$

Từ công thức (6) và (7) và Confusion Matrix như Bảng 2.2, ta có thể tính được F-score cho mỗi nhãn tương ứng. Sau khi thu được F-score của mỗi nhãn, ta bắt đầu tính Macro F-score:

$$\text{F-score}_{macro} = \frac{\sum_i^N \text{F-score}_i}{N} \quad (13)$$

Với độ đo Macro F-score được tính trên trung bình số nhãn có thể phản ánh hiệu suất của mô hình, đặc biệt trong trường hợp dữ liệu các lớp bị lệch như trường hợp giống với dữ liệu của đề tài.

CHƯƠNG 3: THỰC NGHIỆM VÀ ỨNG DỤNG THỰC TẾ

3.1. TIỀN XỬ LÝ

Mô hình Transformer có thể trích xuất đặc trưng rất tốt nhờ Encoder và cơ chế Attention của mình, tuy nhiên trong thực tế dữ liệu thường ở dạng thô, đến từ nhiều nguồn khác nhau cụ thể là văn bản. Các vấn đề có thể gặp phải như:

- **Chữ thường/hoa:** chữ thường và chữ hoa cho dù 2 từ có giống nhau thì khi mã hóa thông tin sẽ trở thành 2 thông tin riêng biệt. Nếu xét trên toàn bộ văn bản sẽ làm nhân lên số lượng từ vựng không đáng có mặc dù các từ đó chỉ mang cùng một ngữ nghĩa.
- **Những từ không liên quan, ký tự đặc biệt và lỗi phong:** các từ không liên quan, ký tự đặc biệt và lỗi chèn chèn sẽ không đóng góp cho quá trình huấn luyện mô hình. Đây được xem là nhiễu cho mô hình.
- **Số liệu:** số liệu thực sự rất có ích cho việc phân tích biến động giá cả nhưng thực tế thì mô hình BERT vẫn chưa thể làm tác vụ phân tích và suy luận nên số liệu cho dù rất có ích nhưng trường hợp này chỉ làm dữ liệu thêm phức tạp.
- **Từ viết tắt:** từ viết tắt có thể là thành phần rất quan trọng trong đề tài (như đồng bằng sông cửu long → đbscl) nhưng chúng nằm ngoài từ vựng mã hóa, nên khả năng bị mã hóa sai, vô tình gây mất mát thông tin quan trọng.

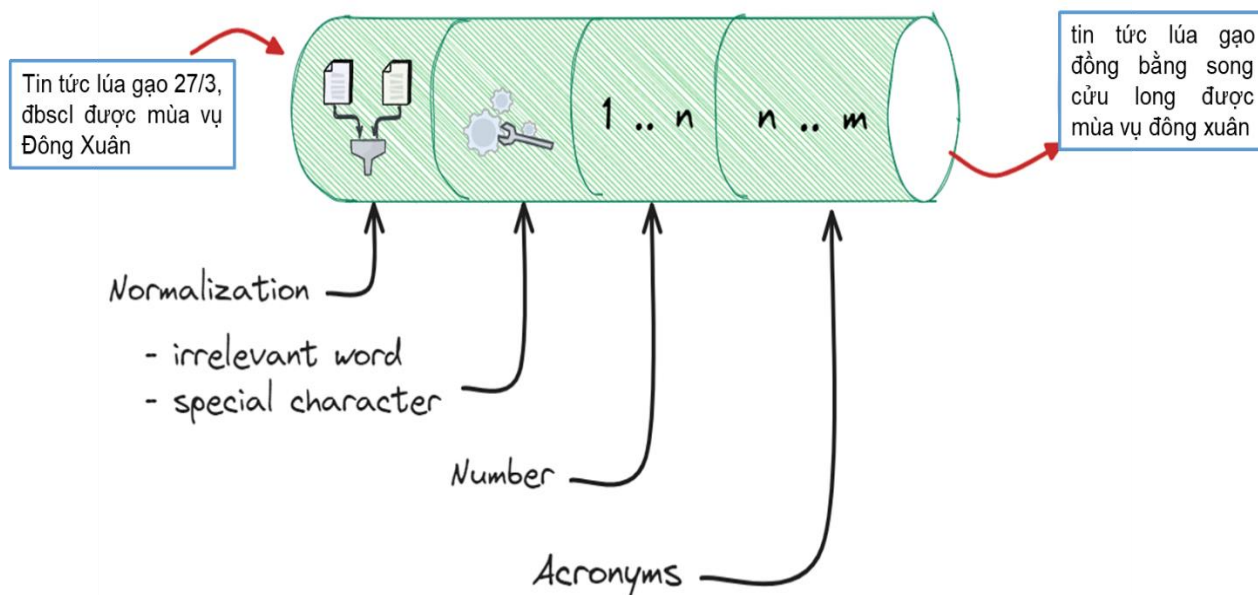
Nên cho dù khả năng Encoder của mô hình có tốt đến đâu cũng nếu dữ liệu không đủ tốt sẽ gây ra nhiễu nhiễu làm quá trình huấn luyện gặp khó khăn. Quá trình tiền xử lý dữ liệu sẽ được diễn ra để giảm thiểu vấn đề trên. Do vậy, dữ liệu văn bản thô sau đó sẽ đi qua một chuỗi các xử lý theo thứ tự sau:

Bước 1. Văn bản sẽ được quy về chữ thường.

Bước 2. Loại bỏ đi những từ không liên quan, ký tự đặc biệt, lỗi phong.

Bước 3. Loại bỏ số liệu.

Bước 4. Xác định từ viết tắt và viết lại đầy đủ.



Hình 3.1 Các giai đoạn tiền xử lý dữ liệu

Sau khi thực hiện tiền xử lý, dữ liệu sau cùng sẽ được chia theo tỉ lệ 9:1 (tập huấn luyện và tập đánh giá) và dùng để huấn luyện mô hình.

3.2. HUẤN LUYỆN MÔ HÌNH

Mô hình sẽ được huấn luyện trên Google Colab với cấu hình như sau:

Bảng 3.1 Cấu hình Colab notebook

VRAM	16Gb
GPU	NVIDIA's Turing T4
RAM	12.7Gb

3.2.1. Siêu Tham số

Khác với tham số (hay parameter) sẽ thay đổi trong quá trình huấn luyện, siêu tham số (hay Hyperparameter) là tham số cần xác định trước để mô hình có thể bắt đầu huấn luyện. Xác định bộ siêu tham số phù hợp và tốt nhất thường phải thử nhiều bộ siêu tham số (mất rất nhiều thời gian). Với mô hình của đề tài ta sẽ sử dụng các siêu tham số sau:

- Learning rate là tỉ lệ học của mô hình.
- Batch size là số mẫu dữ liệu trong một lần huấn luyện
- Epoch là số vòng mà mô hình sử dụng tập dữ liệu
- gamma và alpha là 2 siêu tham số của sigmoid focal loss

Bảng 3.2 Các siêu tham số được chọn để huấn luyện mô hình

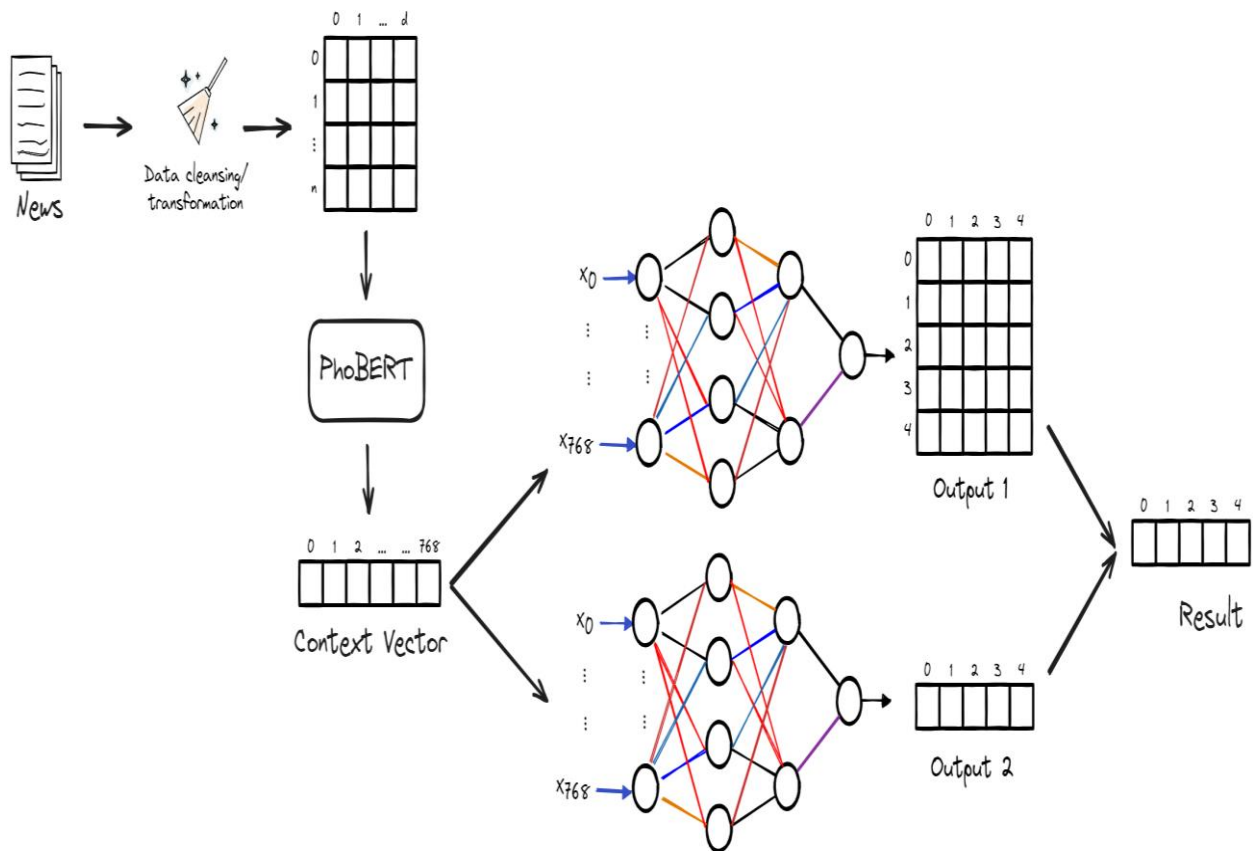
Siêu tham số	Learning rate	Batch size	Epoch	γ	α
Giá trị	5×10^{-5}	32	30	1	-1

3.2.2. Kiến Trúc Mô Hình Giải Quyết Bài Toán

Với mô hình pre-trained PhoBERT hiện có, bây giờ ta có thể sử dụng mô hình đã được huấn luyện với các kiến trúc trước đó. Bằng kỹ thuật học chuyển giao rất phù hợp với tình huống chúng ta có rất ít dữ liệu. Bằng cách freeze các tham số của khối Encoder đã được pre-trained trước đó và chỉ huấn luyện tầng phân loại để mô hình có thể học tác vụ mới từ dữ liệu của đề tài. Quá trình huấn luyện sẽ được tiến hành như sau:

- Preprocssing: dữ liệu sẽ được tiền xử lý như ở Mục 3.1.
- Embedding: Sau khi tiền xử lý dữ liệu sẽ được biểu diễn dữ liệu từ thành một ma trận số thực có chiều d là chiều embedding và n số từ số trong câu để diễn giải ý nghĩa dữ liệu, và được đưa vào mô hình PhoBERT như Hình 3.2.
- Dữ liệu sẽ đi qua 12 khối Encoder để học được các mối quan hệ giữa các từ trong văn bản, từ đó tạo ra một biểu diễn ngữ nghĩa đầy đủ và chính xác. Cuối cùng tạo ra một véc-tơ biểu diễn ngữ nghĩa của văn bản đầu vào có kích thước 768 đại diện cho ý nghĩa toàn bộ văn bản.
- Dữ liệu tiếp tục đi qua 2 mạng kết nối đầy đủ để tổng hợp, phân tích véc-tơ đầu ra của PhoBERT từ tạo ra:
 - Output 1: Là một ma trận vuông có kích thước là 5 (ứng với 5 hạng mục cần dự báo). Trong mỗi hạng mục sẽ chọn index của giá trị lớn nhất làm kết quả dự báo.

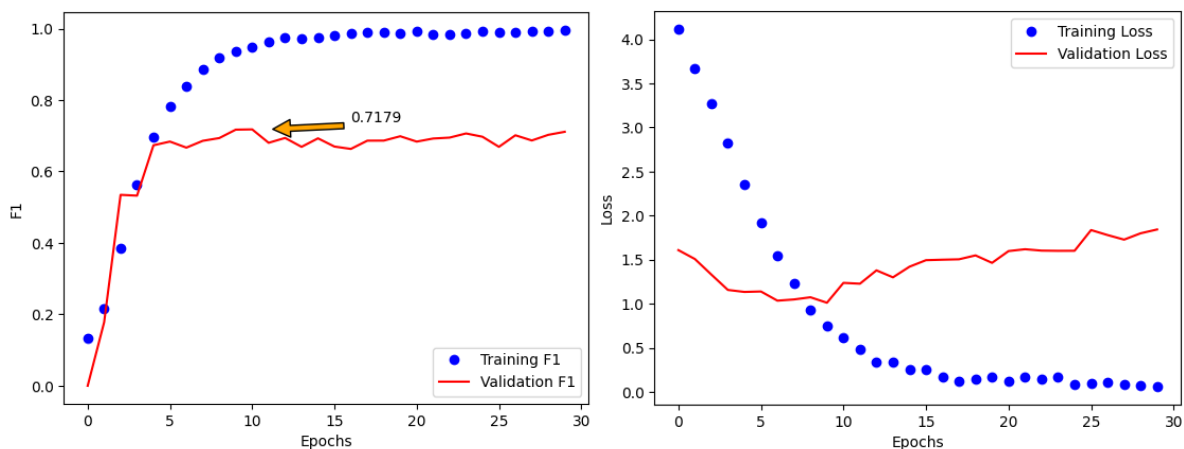
- Output 2: Sau khi qua mạng kết nối đầy đủ sẽ qua hàm kích hoạt sigmoid tạo ra một véc-tơ xác suất có chiều là 5 ứng 5 xác suất/độ tin cậy của với Output 1.
- Kết quả cuối cùng sẽ được đưa ra từ Output 1 nếu xác suất từ kết quả Output 2 đạt độ tin cậy (lớn hơn 50%).



Hình 3.2 Kiến trúc mô hình giải quyết bài toán

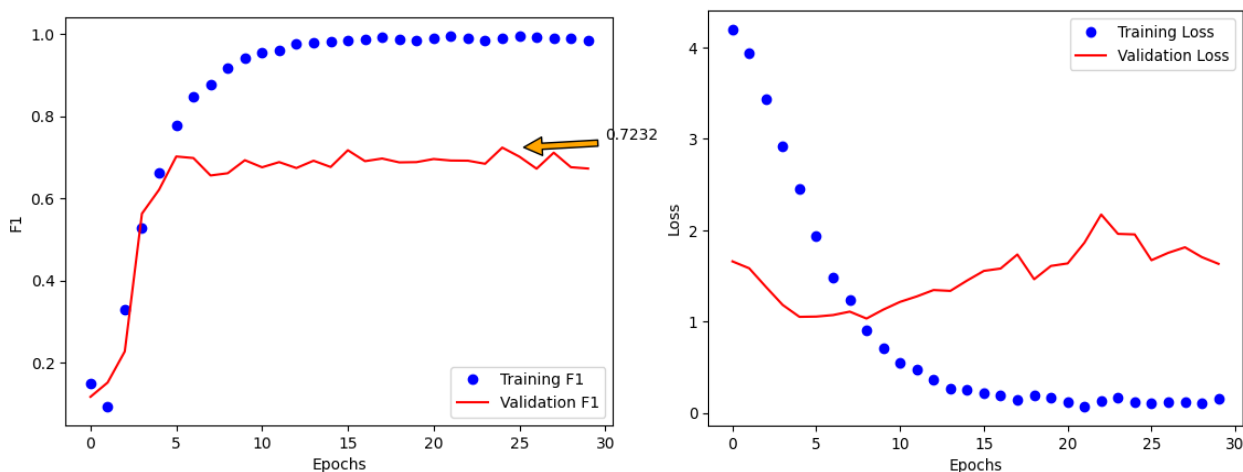
Giải pháp trên xuất phát từ Challenge 2 thuộc Quy Nhon AI Hackathon 2022 [15] đứng nhất thử thách này giải quyết vấn đề đánh giá số sao dựa trên các khía cạnh phục vụ. Với đầu ra gần như tương đương, ta sẽ áp dụng cho bài toán của đề tài bằng cách chỉnh sửa đầu ra mô hình.

3.2.3. Kết quả thực nghiệm



Hình 3.3 Biểu đồ huấn luyện biểu diễn độ đo F1 (trái) và Loss (phải) qua mỗi epoch của mô hình PhoBERT_{base}

Mô hình đã đạt được cao nhất 0.7179 F1-score cho tập đánh giá sau 30 epoch, kết quả không cải thiện nhiều càng về cuối mỗi epoch. Trong khi đó ở tập huấn luyện cho kết quả rất cao, tương như hàm mất mát. Càng huấn luyện mô hình có thể gặp tình trạng quá khớp. Nguyên nhân có thể do dữ liệu quá ít và mất cân bằng mặc dù có dùng biện pháp khắc phục (như Focal loss). Nhưng về bản chất dữ liệu vẫn đóng vai trò quan trọng trong huấn luyện mô hình, các giải pháp đưa ra chỉ hạn chế phần nào.



Hình 3.4 Biểu đồ huấn luyện biểu diễn độ đo F1 (trái) và Loss (phải) qua mỗi epoch của mô hình PhoBERT_{base} Version 2

Tiếp tục thực hiện với mô hình PhoBERT_{base} Version 2 có cải thiện nhưng không đáng kể cao nhất chỉ 0.72 F1-score. Đối với mô hình PhoBERT_{large} hiện tại chưa đủ phần cứng để thử nghiệm.

3.3. TRIỂN KHAI TRÊN ỨNG DỤNG

3.3.1. Môi trường và hệ thống

Bảng 3.3 Cấu hình máy tính dùng để chạy ứng dụng

Laptop Dell Latitude 7490	
OS	Windows 10 64-bit
Processor	i5-8350U
RAM	16 GB
Language	Python 3.11.4

Bảng 3.4 Thư viện được dùng để lập trình ứng dụng

Thư viện	Phiên bản	Mô tả
Numpy	1.26	Thư viện hỗ trợ tính toán dữ liệu số trên 1 hoặc nhiều chiều
Torch	2.1.0	Thư viện dùng để cấu hình kiến trúc của mô hình mạng học sâu/học máy
Transformers	4.34.1	Thư viện hỗ trợ cung cấp các mô hình được huấn luyện sẵn (pre-trained)
Openpyxl	3.1.2	Thư viện dùng để đọc/ghi các định dạng tệp như xlsx/xlsm/xltx/xltm
Datasets	2.14.6	Thư viện dùng để tải bộ dữ liệu từ máy cục bộ
Vncorenlp	1.0.3	Thư viện dùng để
Flask	3.0.0	Thư viện sử dụng triển khai flask framework
Flask-sqlalchemy	3.1.1	Mở rộng của thư viện flask dùng hỗ trợ SQLAlchemy
SQLAlchemy	2.0.23	Một công cụ python SQL sử dụng trình ORM
pymysql	1.1.0	Thư viện mysql cho python

3.3.2. Giới thiệu công cụ

3.3.2.1. Flask

Python Flask là một micro web framework được xây dựng bằng Python để phát triển ứng dụng web. Nó được thiết kế để đơn giản và linh hoạt, cho phép người phát triển xây dựng các ứng dụng web nhanh chóng và dễ dàng. Flask không yêu cầu cấu hình phức tạp và có ít yêu cầu về cài đặt ban đầu, giúp người dùng tập trung vào việc xây dựng ứng dụng mà không phải lo lắng về các vấn đề kỹ thuật phức tạp. Flask cung cấp các công cụ cần thiết để xử lý yêu cầu HTTP, quản lý tương tác với cơ sở dữ liệu, xử lý mẫu, và tạo ra các trang web động. Nó cung cấp một cấu trúc cơ bản để xây dựng ứng dụng web, nhưng cũng cho phép người dùng tùy chỉnh và mở rộng theo nhu cầu. Một trong những đặc điểm nổi bật của Flask là việc sử dụng đơn vị xử lý (micro framework), cho phép lựa chọn các thành phần cụ thể muốn sử dụng, thay vì bị ràng buộc bởi một cấu trúc toàn diện và phức tạp. Điều này giúp Flask trở nên nhẹ nhàng, nhỏ gọn, và dễ dàng học hơn so với các framework web khác. Cũng chính vì lý do này, Flask phù hợp việc triển khai của mô hình học sâu quy mô nhỏ.

3.3.2.2. SQLAlchemy

SQLAlchemy là bộ công cụ mã nguồn mở đa nền tảng xử lý SQL cho ngôn ngữ Python dựa trên kỹ thuật ORM (Object Relational Mapper) nhằm cung cấp nhà hát triển một công cụ mạnh mẽ và linh hoạt. Bằng với kỹ thuật ORM cho phép ánh xạ cơ sở dữ liệu đến các đối tượng thuộc ngôn ngữ lập trình hướng đối tượng thay vì phải viết trực tiếp bằng các câu lệnh truy vấn.

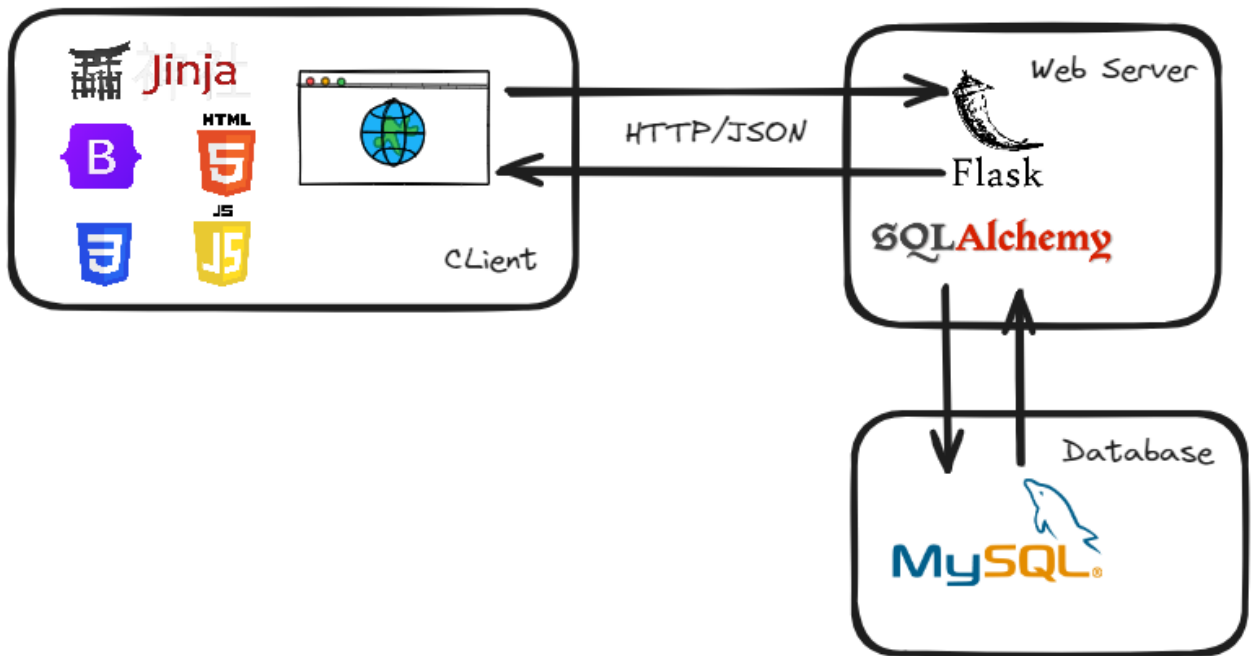
3.3.3. Kiến trúc hệ thống

3.3.3.1. Kiến trúc

Trang website tin tức được xây dựng khá đơn giản bao gồm ba phần chính:

- Cơ sở dữ liệu dùng lưu trữ tất cả nội dung trang web

- Máy chủ web dùng để chạy flask
- Máy khách sẽ hiển thị nội dung được trả về từ máy chủ web.



Hình 3.5 Kiến trúc của ứng dụng web triển khai mô hình của đề tài

HTML, CSS, Javascript và Bootstrap được sử dụng để thiết kế giao diện cho Client. Jinja là một template engine cho python tạo các template và cũng được sử dụng để hiển các thông tin được xử lý từ web server.

Trường hợp trình duyệt *request* đến web server, lúc này web server sẽ chạy mã Python cần thiết hoặc tương tác với cơ sở dữ liệu thông qua SQLAlchemy để trả về cho Client. Dữ liệu được *request* sẽ xử lý thành mẫu HTML để gửi đến và hiển thị bằng jinja cho Client.

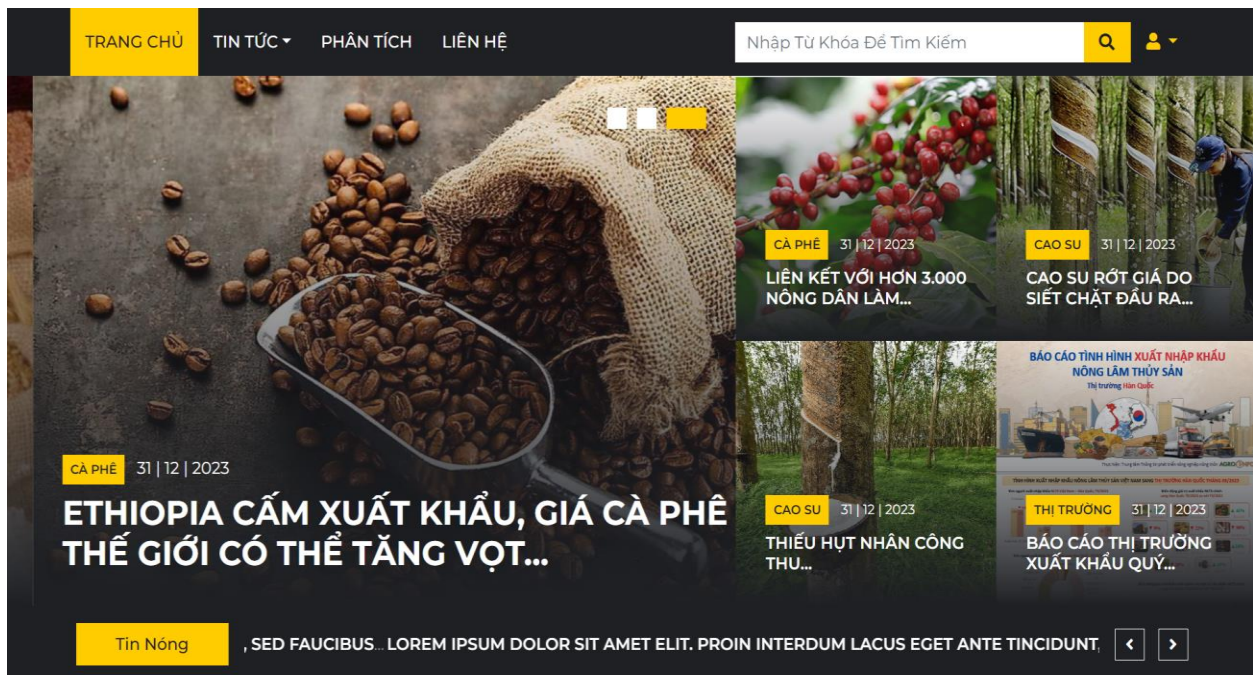
3.3.3.2. Chức năng của trang web

- Tìm kiếm
- Xem tin tức
- Đăng nhập, đăng ký tài khoản
- Phân tích tin tức

3.3.4. Kết quả triển khai

3.3.4.1. Giao diện website

- Trang chủ của ứng dụng web:



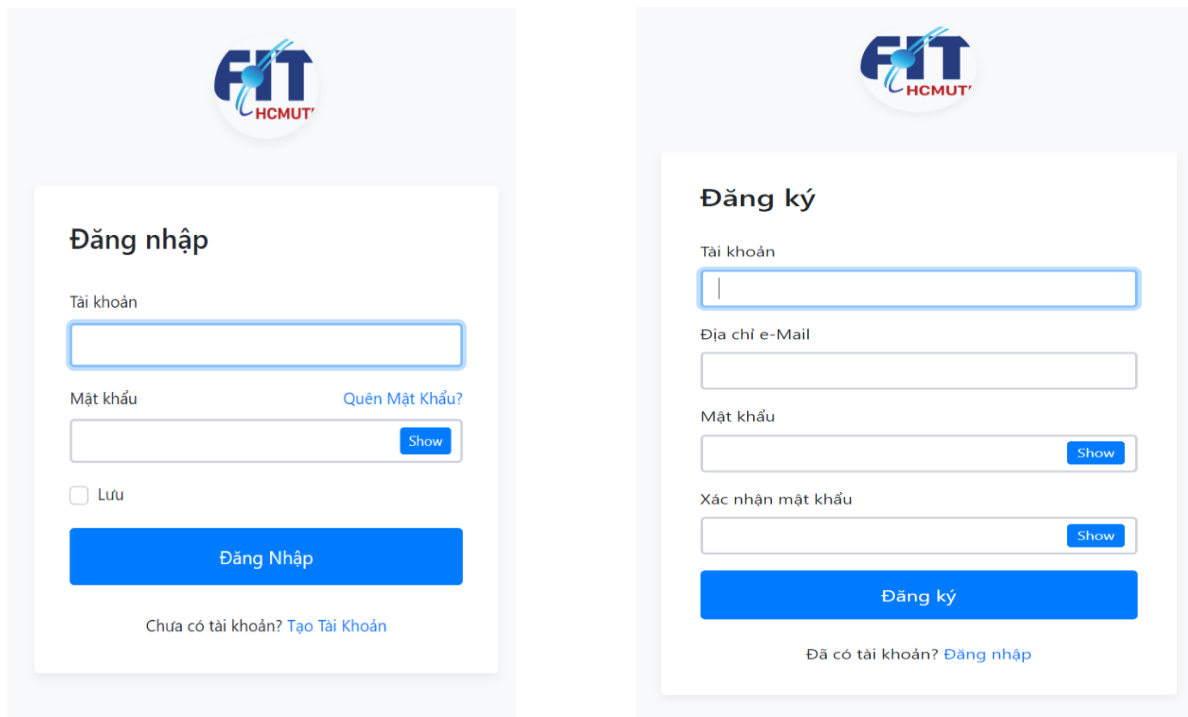
Hình 3.6 Giao diện trang chủ website

- Trang tổng hợp tin theo danh mục nông sản (hoặc hiển thị kết quả tìm kiếm):



Hình 3.7 Giao diện danh sách tin website

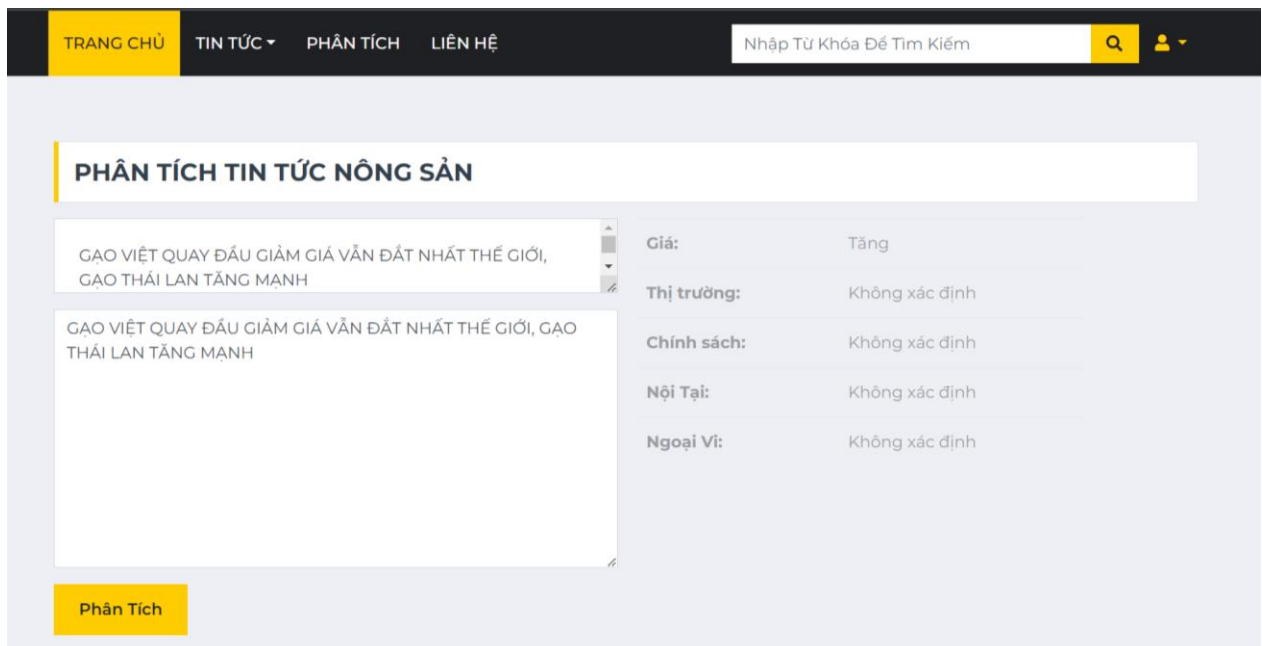
- Giao diện chức năng đăng nhập và đăng ký:



The image displays two side-by-side screenshots of the FIT HCMUT website's user interface. The left screenshot shows the 'Đăng nhập' (Login) form, which includes fields for 'Tài khoản' (Username) and 'Mật khẩu' (Password), a 'Quên Mật Khẩu?' (Forgot Password?) link, a 'Show' button for the password, a 'Lưu' (Remember) checkbox, and a blue 'Đăng Nhập' (Login) button. Below the button is a link for 'Chưa có tài khoản? Tạo Tài Khoản' (Don't have an account? Create Account). The right screenshot shows the 'Đăng ký' (Registration) form, which includes fields for 'Tài khoản' (Username), 'Địa chỉ e-Mail' (Email address), 'Mật khẩu' (Password), and 'Xác nhận mật khẩu' (Confirm password), each with a 'Show' button. It features a blue 'Đăng ký' (Register) button and a link for 'Đã có tài khoản? Đăng nhập' (Already have an account? Login).

Hình 3.8 Giao diện đăng nhập (trái) và đăng ký (phải)

- Giao diện chức năng phân tích tin tức:



The image shows a screenshot of the website's news analysis interface. At the top is a navigation bar with links for 'TRANG CHỦ', 'TIN TỨC', 'PHÂN TÍCH', and 'LIÊN HỆ', along with a search bar labeled 'Nhập Từ Khóa Để Tìm Kiếm'. The main content area is titled 'PHÂN TÍCH TIN TỨC NÔNG SẢN'. It features a large text box on the left containing the text 'GAO VIỆT QUAY ĐẦU GIẢM GIÁ VẮN ĐẤT NHẤT THẾ GIỚI, GAO THÁI LAN TĂNG MẠNH'. To the right of this text box is a table with the following data:

Giá:	Tăng
Thị trường:	Không xác định
Chính sách:	Không xác định
Nội Tại:	Không xác định
Ngoại Vĩ:	Không xác định

At the bottom left of the main content area is a yellow button labeled 'Phân Tích'.

Hình 3.9 Giao diện phân tích tin tức của website

3.3.4.2. Hiệu quả thực thi

- Thời gian tải mô hình lần đầu: 3 – 8 giây
- Thời gian thực thi dự đoán: 0.1 – 1.5 giây (tùy vào độ dài văn bản)

PHẦN KẾT LUẬN

1. KẾT QUẢ ĐẠT ĐƯỢC

1.1. Kiến thức chuyên môn

- Cách thu thập dữ liệu
- Cách xử lý dữ liệu văn bản
- Kiến trúc mô hình Transformer
- Học hỏi được các debug để tìm lỗi và giải quyết các lỗi xảy ra trong quá trình
- phát triển đề tài.
- Học và tiếp thu được nhiều kiến thức mới.

1.2. Kỹ năng

- Kỹ năng đọc tài liệu tiếng anh, tự học, tự nghiên cứu để giải quyết khó khăn.
- Kỹ năng quản lý thời gian để thực hiện đúng tiến độ mà không bị ảnh hưởng bởi nhiều yếu tố khác nhau

1.3. Ưu nhược điểm

1.3.1. Ưu điểm

- Tốc độ phản hồi của mô hình nhanh có thể đáp ứng trong thực tế

1.3.2. Nhược điểm

- Tập dữ liệu chưa thực sự đủ lớn
- Tập dữ liệu bị mất cân bằng.
- Mô hình cho độ đo F1 khoảng 0.72

2. ĐÓNG GÓP

- Bộ dữ liệu tin tức nông sản đã được gắn nhãn thủ công.
- Một mô hình giải quyết bài toán có tốc độ phản hồi nhanh
- Mô hình hỗ trợ ra quyết định các vấn đề liên quan đến nông sản

3. THUẬN LỢI VÀ KHÓ KHĂN

3.1. Thuận lợi

- Dễ dàng tiếp cận và phát triển phần mềm do đã có kinh nghiệm lập trình.
- Nhờ có sự hướng dẫn nhiệt tình của thầy Trần Nhật Quang cũng như các thầy cô trong Khoa Công Nghệ Thông Tin trường Đại học Sư Phạm Kỹ Thuật giúp đỡ nhóm trong thời gian thực hiện đề tài.

3.2. Khó khăn

- Quá trình gắn nhãn thủ công mất nhiều thời gian và nhiều công sức
- Dữ liệu mất cân bằng
- Tài nguyên phần cứng hạn chế nên chưa thể huấn luyện trên các mô hình lớn hơn
- Chưa thể tích hợp và phân tích dữ liệu có thời gian để đưa ra dự đoán một cách chính xác

4. HƯỚNG PHÁT TRIỂN

Trong quá trình thực hiện đề tài, do thời gian và kinh nghiệm còn hạn chế nên gặp không ít khó khăn trong việc triển khai đề tài, vì vậy tôi đã đề ra 2 hướng phát triển cho đề tài như sau:

4.1. Tiếp tục phát triển mô hình hiện tại

- Thu thập thêm dữ liệu hoặc sử dụng các kỹ thuật tăng cường dữ liệu.
- Sử dụng kết hợp thêm các đặc trưng để huấn luyện khác như: nội dung toàn văn hoặc thời gian của bài báo.
- Tiền xử lý dữ liệu sâu hơn.
- Áp dụng các mô hình SOTA và các phương pháp mới hơn
- Tinh chỉnh lại các siêu tham số phù hợp hơn

4.2. Là công cụ phát triển mô hình khác

- Kết hợp dữ liệu time-series giá nông sản (từ bên ngoài) và dữ liệu của đề tài có thể làm tăng độ chính xác dự đoán.

- Là mô hình hỗ trợ cho các mô hình dự đoán giá liên quan đến nông sản

TÀI LIỆU THAM KHẢO

- [1]. Wang, L., Feng, J., Sui, X., Chu, X., & Mu, W. (2020). Agricultural product price forecasting methods: research advances and trend. *British Food Journal*, 122(7), 2121-2138.
- [2]. Li, J., Li, G., Liu, M., Zhu, X., & Wei, L. (2022). A novel text-based framework for forecasting agricultural futures using massive online news headlines. *International Journal of Forecasting*, 38(1), 35-50.
- [3]. Viện chính sách và chiến lược phát triển nông nghiệp nông thôn (n.d.). Trung tâm thông tin phát triển nông nghiệp nông thôn. AGROINFO. Retrieved December 22, 2023, from <https://agro.gov.vn>
- [4]. Tong, A. H. (2012). Factors influencing price of agricultural products and stability countermeasures. *Asian agricultural research*, 4(1812-2016-143301), 17-43.
- [5]. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Identity mappings in deep residual networks. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14* (pp. 630-645). Springer International Publishing.
- [6]. Thrun, S., & Pratt, L. (1998). Learning to learn: Introduction and overview. In *Learning to learn* (pp. 3-17). Boston, MA: Springer US.
- [7]. Olivas, E. S., Guerrero, J. D. M., Martinez-Sober, M., Magdalena-Benedito, J. R., & Serrano, L. (Eds.). (2009). *Handbook of research on machine learning applications and trends: Algorithms, methods, and techniques: Algorithms, methods, and techniques*. IGI global.
- [8]. Jiao, Q., & Zhang, S. (2021, March). A brief survey of word embedding and its recent development. In *2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)* (Vol. 5, pp. 1697-1701). IEEE.
- [9]. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- [10]. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- [11]. Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.
- [12]. Nguyen, D. Q., & Nguyen, A. T. (2020). PhoBERT: Pre-trained language models for Vietnamese. *arXiv preprint arXiv:2003.00744*.
- [13]. Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* (pp. 2980-2988).

- [14]. Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
- [15]. CTA Matrix (n.d.). *QN-Hackathon-CTA-Matrix*. GitHub. Retrieved December 26, 2023, from <https://github.com/véc-tonguyen76/QN-Hackathon-CTA-Matrix>