

**BỘ GIÁO DỤC VÀ ĐÀO TẠO    NGÂN HÀNG NHÀ NƯỚC VIỆT NAM**  
**TRƯỜNG ĐẠI HỌC NGÂN HÀNG TP. HỒ CHÍ MINH**



## **ĐỀ TÀI NGHIÊN CỨU KHOA HỌC**

### **PHÂN TÍCH CẢM XÚC CỦA KHÁCH HÀNG DỰA TRÊN KHÍA CẠNH BÌNH LUẬN SỬ DỤNG MÔ HÌNH HỌC SÂU**

Bùi Mai Phương	030238220195	DH38TM01	HTTTQL
Nguyễn Ngọc Hồng Hòa	030238220067	DH38TM01	HTTTQL
Lê Thị Thùy Linh	030238220107	DH38CDS01	HTTTQL
Lê Anh Vũ	030238220300	DH38KH01	HTTTQL
Lê Thị Thảo Hiền	030238220059	DH38TM01	HTTTQL

**GVHD: ThS. NGUYỄN VĂN KIÊN**

**TP. HỒ CHÍ MINH, NĂM 2025**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO    NGÂN HÀNG NHÀ NƯỚC VIỆT NAM**  
**TRƯỜNG ĐẠI HỌC NGÂN HÀNG TP. HỒ CHÍ MINH**



## **ĐỀ TÀI NGHIÊN CỨU KHOA HỌC**

### **PHÂN TÍCH CẢM XÚC CỦA KHÁCH HÀNG DỰA TRÊN KHÓA CẠNH BÌNH LUẬN SỬ DỤNG MÔ HÌNH HỌC SÂU**

Bùi Mai Phương	030238220195	DH38TM01	HTTTQL
Nguyễn Ngọc Hồng Hòa	030238220067	DH38TM01	HTTTQL
Lê Thị Thùy Linh	030238220107	DH38CDS01	HTTTQL
Lê Anh Vũ	030238220300	DH38KH01	HTTTQL
Lê Thị Thảo Hiền	030238220059	DH38TM01	HTTTQL

**GVHD: ThS. NGUYỄN VĂN KIÊN**

**TP. HỒ CHÍ MINH, NĂM 2025**

## TÓM TẮT

Phân tích cảm xúc theo khía cạnh đóng vai trò quan trọng trong việc trích xuất thông tin chi tiết từ dữ liệu phản hồi của người tiêu dùng, góp phần nâng cao hiệu quả giám sát trải nghiệm khách hàng và hỗ trợ ra quyết định trong lĩnh vực thương mại điện tử. Nghiên cứu này đề xuất một phương pháp tiếp cận sử dụng mô hình học sâu dựa trên kiến trúc PhoBERT để biểu diễn văn bản thành các vector ngữ nghĩa và kết hợp với tầng phân lớp tuyến tính nhằm thực hiện đồng thời hai nhiệm vụ: (1) xác định mối liên hệ giữa bình luận và các khía cạnh cụ thể của sản phẩm và (2) phân loại cảm xúc tương ứng theo từng khía cạnh.

Tập dữ liệu thực nghiệm bao gồm 9.963 bình luận người dùng được thu thập tự động từ nền tảng thương mại điện tử tiki.vn với các thư viện hỗ trợ của ngôn ngữ lập trình Python, tập trung vào nhóm sản phẩm đồ dùng nhà bếp. Quá trình gán nhãn được thực hiện tự động với sự hỗ trợ của mô hình ngôn ngữ ChatGPT thông qua API, nhằm đảm bảo tính nhất quán và hiệu quả trong phân loại cảm xúc theo khía cạnh. Dữ liệu được xử lý và huấn luyện theo phương pháp phân chia k-fold stratified ( $k = 5$ ) để đảm bảo sự cân bằng giữa các lớp và khía cạnh.

Kết quả thực nghiệm cho thấy mô hình đạt hiệu quả với các chỉ số đánh giá như Accuracy, Precision, Recall và F1-score. Kết quả từ nghiên cứu cho thấy tiềm năng ứng dụng của mô hình trong các hệ thống phân tích phản hồi khách hàng theo thời gian thực, từ đó góp phần nâng cao chất lượng dịch vụ và năng lực cạnh tranh của doanh nghiệp trong môi trường kinh doanh số.

## ABSTRACT

Aspect-based sentiment analysis (ABSA) plays a crucial role in extracting detailed information from consumer feedback, thereby enhancing the effectiveness of customer experience monitoring and supporting decision-making processes in the field of e-commerce. This study proposes an approach that leverages a deep learning model based on the PhoBERT architecture to represent text as semantic vectors, combined with a linear classification layer to simultaneously perform two tasks: (1) identifying the relationship between a review and specific aspects of a product, and (2) classifying the corresponding sentiment for each aspect.

The experimental dataset consists of 9,963 user-generated reviews automatically collected from the e-commerce platform tiki.vn, using Python programming language with relevant web-scraping libraries, and focuses on kitchenware products. The annotation process is automated with the assistance of the ChatGPT language model via API, ensuring consistency and efficiency in aspect-level sentiment labeling. The data is preprocessed and trained using the k-fold stratified cross-validation method ( $k = 5$ ) to maintain a balanced distribution across sentiment classes and aspects.

Experimental results demonstrate the effectiveness of the proposed model through evaluation metrics such as Accuracy, Precision, Recall, and F1-score. The findings highlight the model's potential application in real-time customer feedback analysis systems, contributing to improved service quality and enhanced competitiveness of businesses in the digital economy.

### DANH MỤC CHỮ VIẾT TẮT TIẾNG ANH

Từ viết tắt	Cụm từ tiếng Anh	Nghĩa tiếng Việt
ABSA	Aspect-Based Sentiment Analysis	Phân tích cảm xúc theo khía cạnh
AD	Aspect Detection	Phát hiện khía cạnh
ANN	Artificial Neural Network	Mạng nơ-ron nhân tạo
AP	Aspect-based Polarity	Phân tích cảm xúc theo khía cạnh
BoW	Bag-of-Words	Mô hình túi từ
BERT	Bidirectional Encoder Representations from Transformers	Biểu diễn mã hóa hai chiều bằng Transformer
CBOW	Continuous Bag-of-Words	Mô hình túi từ liên tục
CNN	Convolutional Neural Networks	Mạng nơ-ron tích chập
Conv1D	1D Convolutional Layer	Lớp tích chập 1 chiều
ELMo	Embeddings from Language Models	Biểu diễn từ từ mô hình ngôn ngữ
FN	False Negative	Số lượng mẫu thuộc lớp dương bị dự đoán nhầm thành lớp âm
FP	False Positive	Số lượng mẫu thuộc lớp âm bị dự đoán nhầm thành lớp dương
GloVe	Global Vectors for Word Representation	Mô hình GloVe
GPT	Generative Pre-trained Transformer	Transformer tiền huấn luyện sinh văn bản
GRU	Gated Recurrent Unit	Đơn vị hồi tiếp có cổng
ID3	Information Gain	Độ lợi thông tin (thuật toán cây quyết định)
k-NN	k-Nearest Neighbors	Giải thuật phân lớp cơ bản dựa trên các điểm lân cận

LSTM	Long Short-Term Memory	Bộ nhớ ngắn hạn dài
MAE	Mean Absolute Error	Sai số tuyệt đối trung bình
MLM	Masked Language Modeling	Mô hình hóa ngôn ngữ với từ bị che
MTL	Multi-Task Learning	Mô hình đa tác vụ
NLP	Natural Language Processing	Xử lý ngôn ngữ tự nhiên
NSP	Next Sentence Prediction	Dự đoán câu tiếp theo
POS tagging	Part-of-Speech tagging	Gán nhãn từ loại
RNN	Recurrent Neural Networks	Mạng nơ-ron hồi quy
STL	Single-Task Learning	Mô hình đơn tác vụ
SV (SVM)	Support Vector Machine	Máy vector hỗ trợ
T5	Text-to-Text Transfer Transformer	Transformer chuyển đổi văn bản sang văn bản
TF-IDF	Term Frequency-Inverse Document Frequency	Tần suất từ - Tần suất đảo ngược văn bản
TN	True Negative	Số lượng mẫu thuộc lớp âm dự đoán đúng
TP	True Positive	Số lượng mẫu thuộc lớp dương dự đoán đúng

## **DANH MỤC BẢNG BIỂU**

Bảng 2.1. Ma trận nhằm lẫn .....	8
Bảng 5.1. Số lượng bình luận theo từng khía cạnh .....	34
Bảng 5.2. Kết quả đánh giá chung mô hình phân tích cảm xúc theo khía cạnh .....	35
Bảng 5.3. Kết quả đánh giá theo từng khía cạnh cụ thể.....	36

## **DANH MỤC HÌNH ẢNH**

Hình 4.1. Mô hình phân tích cảm xúc theo khía cạnh .....	30
Hình 5.1. Dữ liệu bình luận sau khi được tiền xử lý.....	34

## MỤC LỤC

CHƯƠNG 1. GIỚI THIỆU VẤN ĐỀ .....	1
1.1. Phát biểu vấn đề nghiên cứu .....	1
1.2. Mục tiêu nghiên cứu .....	2
1.3. Câu hỏi nghiên cứu .....	2
1.4. Đối tượng và phạm vi nghiên cứu .....	2
1.4.1. Đối tượng nghiên cứu .....	2
1.4.2. Phạm vi nghiên cứu .....	3
1.5. Ý nghĩa nghiên cứu .....	3
1.6. Bố cục nghiên cứu .....	3
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT .....	4
2.1. Phân lớp dữ liệu .....	4
2.1.1. Giới thiệu .....	4
2.1.2. Các giải thuật và mô hình phân lớp .....	4
2.1.3. Các độ đo đánh giá chất lượng phân lớp dữ liệu .....	8
2.1.4. Phân chia dữ liệu huấn luyện và kiểm tra .....	11
2.2. Phân tích cảm xúc .....	14
2.2.1. Giới thiệu .....	14
2.2.2. Các cấp độ phân tích cảm xúc .....	15
2.2.3. Mô hình học sâu .....	15
2.2.4. Các phương pháp biểu diễn văn bản thành vector .....	17
2.3. Xử lý ngôn ngữ tiếng Việt .....	21
2.3.1. Biểu diễn văn bản cho ngôn ngữ tiếng Việt .....	21
2.3.2. Mô hình PhoW2V .....	22
2.3.3. Mô hình mBERT .....	23
2.3.4. Mô hình PhoBERT .....	24
2.4. Phân tích cảm xúc dựa trên khía cạnh .....	25
CHƯƠNG 3. PHƯƠNG PHÁP NGHIÊN CỨU .....	28
3.1. Thu thập dữ liệu .....	28
3.2. Tiền xử lý dữ liệu .....	29
3.3. Xây dựng mô hình phân tích cảm xúc theo khía cạnh .....	30
CHƯƠNG 4. KẾT QUẢ THỰC NGHIỆM .....	33



4.1. Môi trường thực nghiệm .....	33
4.2. Dữ liệu thực nghiệm .....	33
4.3. Kết quả thực nghiệm.....	35
CHƯƠNG 5. KẾT LUẬN.....	38
5.1. Tổng kết .....	38
5.2. Đóng góp của đề tài .....	38
5.3. Hạn chế .....	39
5.4. Hướng phát triển .....	40
TÀI LIỆU THAM KHẢO.....	42
PHỤ LỤC.....	45
Phụ lục 1: Mã nguồn cho việc huấn luyện và đánh giá mô hình ABSA .....	45
Phụ lục 2: Kết quả đánh giá mô hình chạy theo từng Fold .....	50

## **CHƯƠNG 1. GIỚI THIỆU VẤN ĐỀ**

### **1.1. Phát biểu vấn đề nghiên cứu**

Với sự phát triển của internet và công nghệ, thương mại điện tử đang trở nên phổ biến hơn bao giờ hết. Trong vai trò người bán trên các nền tảng thương mại điện tử, việc phân tích cảm xúc của khách hàng là rất quan trọng để đảm bảo hiệu quả kinh doanh. Phân tích cảm xúc là quá trình thu thập và phân tích ý kiến, suy nghĩ của khách hàng về các chủ đề như sản phẩm và dịch vụ. Trong lĩnh vực thương mại điện tử, ý kiến của khách hàng thường được thể hiện qua các bình luận và đánh giá sản phẩm. Bằng cách phân tích những bình luận này, người bán có thể hiểu được cảm xúc của khách hàng và đưa ra những quyết định kinh doanh phù hợp. Tuy nhiên, với số lượng lớn bình luận trên các trang web thương mại điện tử, việc phân tích cảm xúc của khách hàng không thể thực hiện bằng cách thủ công mà cần sự tự động hóa. Đó là lý do tại sao tự động hóa phân tích cảm xúc đang trở thành một lĩnh vực nghiên cứu được quan tâm và áp dụng rộng rãi trong nhiều ngành công nghiệp. Trong những năm gần đây, đã có nhiều công trình nghiên cứu trong lĩnh vực xử lý ngôn ngữ tự nhiên liên quan đến phân tích cảm xúc dựa trên bình luận, nhưng chủ yếu được thực hiện trên các bộ dữ liệu bình luận sử dụng ngôn ngữ tiếng Anh (Chauhan & cộng sự, 2018; Hoang & cộng sự, 2019; Sivakumar & Reddy, 2017). Mặc dù vẫn có một số nghiên cứu phân tích cảm xúc trên các bình luận tiếng Việt (Q.-L. Tran & cộng sự, 2022; Van Nguyen & cộng sự, 2018; Van Thin & cộng sự, 2018), nhưng do sự phức tạp về ngữ nghĩa và tính phong phú câu từ của tiếng Việt cũng như chưa xem xét đến yếu tố ngữ cảnh trong quá trình xử lý nên vẫn còn nhiều hạn chế về kết quả khi phân tích cảm xúc theo khía cạnh. Một số công trình vẫn đưa ra các mô hình phân tích có kết quả tốt nhưng các mô hình này chỉ tập trung phân tích xu hướng cảm xúc chung (tiêu cực, tích cực) trong bình luận đánh giá của khách hàng mà chưa xem xét các khía cạnh trong bình luận khi phân tích (Dang & cộng sự, 2023).

Ví dụ về một bình luận của khách hàng khi đánh giá sản phẩm “nồi cơm điện” như sau: “Lòng nồi dày, chắc chắn, đẹp, nấu cơm chín đều, ngon. Tuy nhiên thời gian giao hàng hơi lâu”.

Trong ví dụ trên, bình luận của khách hàng liên quan đến hai khía cạnh. Đối với khía cạnh về chất lượng của sản phẩm, cảm xúc của khách hàng thể hiện theo hướng tích cực, còn về khía cạnh dịch vụ giao hàng, cảm xúc thể hiện theo hướng

tiêu cực. Qua đó cho thấy rằng, các khía cạnh này là rất quan trọng để hiểu được cảm xúc của khách hàng. Trong bối cảnh ngày nay, khi môi trường kinh doanh trực tuyến ngày càng cạnh tranh và khách hàng trở nên thông thái hơn, việc áp dụng phân tích cảm xúc trong lĩnh vực này trở thành một nhu cầu cấp bách, giúp doanh nghiệp dễ dàng nắm bắt nhu cầu và mong muốn của khách hàng, từ đó tối ưu hóa chiến lược kinh doanh và tăng cường sức cạnh tranh trên thị trường.

## **1.2. Mục tiêu nghiên cứu**

Mục tiêu tổng quát của đề tài là sử dụng mô hình học sâu để tự động phân tích cảm xúc và cải thiện chất lượng phân tích cho các bình luận theo từng khía cạnh của khách hàng, nhằm cung cấp cho doanh nghiệp thông tin giá trị để cải thiện dịch vụ, tối ưu hóa sản phẩm, cá nhân hóa trải nghiệm mua sắm và nâng cao sức cạnh tranh trên thị trường, với các nội dung nghiên cứu cụ thể gồm:

- Xây dựng được bộ dữ liệu huấn luyện chất lượng phục vụ cho bài toán phân tích cảm xúc theo khía cạnh, thông qua việc thu thập tự động và xử lý dữ liệu bình luận khách hàng trên các trang web thương mại điện tử.
- Xây dựng mô hình phân tích cảm xúc của khách hàng dựa trên các khía cạnh của nội dung bình luận bằng các kỹ thuật học sâu, nhằm nâng cao hiệu quả phân tích và hỗ trợ hiểu biết sâu sắc hơn về trải nghiệm khách hàng.
- Thực nghiệm và đánh giá mô hình trên bộ dữ liệu mẫu để đo lường hiệu quả qua các chỉ số cụ thể, từ đó rút ra các khuyến nghị cho ứng dụng thực tế.

## **1.3. Câu hỏi nghiên cứu**

Việc thực hiện đề tài nghiên cứu trên nhằm trả lời cho các câu hỏi sau:

- Làm thế nào để thu thập và xử lý tự động dữ liệu bình luận khách hàng trên các trang web thương mại điện tử?
- Mô hình học sâu nào có hiệu quả cao nhất trong việc phân tích cảm xúc dựa trên khía cạnh bình luận?
- Làm thế nào để đánh giá chính xác hiệu suất của mô hình phân tích cảm xúc dựa trên khía cạnh bình luận?

## **1.4. Đối tượng và phạm vi nghiên cứu**

### **1.4.1. Đối tượng nghiên cứu**

Đối tượng nghiên cứu của đề tài là cảm xúc của khách hàng được thể hiện trong các khía cạnh cụ thể của nội dung bình luận trên nền tảng thương mại điện tử.

#### **1.4.2. Phạm vi nghiên cứu**

Đề tài tập trung nghiên cứu một mô hình học sâu đã được huấn luyện để có thể tự động phân tích cảm xúc của khách hàng dựa trên các khía cạnh trong nội dung bình luận và mô hình có thể tích hợp được vào các website thương mại điện tử.

Với nền tảng thương mại điện tử, chúng tôi tập trung vào nền tảng thương mại điện tử hoặc trang web phổ biến hiện nay tại Việt Nam là Tiki để thực hiện việc thu thập dữ liệu. Việc này phụ thuộc vào mục tiêu cụ thể của nghiên cứu là thu thập tự động dữ liệu bình luận của khách hàng trên trang web thương mại điện tử và tiến hành xử lý để xây dựng bộ dữ liệu huấn luyện cho các mô hình học sâu.

Về ngành hàng, khi thực hiện thu thập dữ liệu tự động, chúng tôi hạn chế phạm vi nghiên cứu vào một số ngành hàng cụ thể không mang tính đặc thù để có thể đảm bảo tính đồng nhất dữ liệu và giảm độ phức tạp của mô hình. Chúng tôi lựa chọn ngành hàng: đồ dùng nhà bếp.

#### **1.5. Ý nghĩa nghiên cứu**

Kết quả dự kiến của đề tài là xây dựng thành công một mô hình học sâu đã được huấn luyện có khả năng phân tích chi tiết cảm xúc của khách hàng dựa trên các khía cạnh trong bình luận sản phẩm trên các nền tảng thương mại điện tử, từ đó mô hình có thể nhanh chóng cung cấp cho doanh nghiệp những thông tin giá trị để cải thiện dịch vụ khách hàng, tối ưu hóa sản phẩm và cá nhân hóa trải nghiệm mua sắm tiết kiệm rất nhiều thời gian, công sức trong việc phân loại các bình luận

#### **1.6. Bố cục nghiên cứu**

*Chương 1. Giới thiệu vấn đề*

*Chương 2. Cơ sở lý thuyết*

*Chương 3. Phương pháp nghiên cứu*

*Chương 4. Kết quả thực nghiệm*

*Chương 5. Kết luận*

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

### 2.1. Phân lớp dữ liệu

#### 2.1.1. Giới thiệu

Phân lớp là một dạng phân tích dữ liệu nhằm rút trích các mô hình mô tả đặc điểm các lớp dữ liệu (Han & cộng sự, 2012). Phân lớp là một quá trình học có giám sát gồm hai giai đoạn chính: huấn luyện (training) và phân lớp (classification).

Trong giai đoạn huấn luyện, một mô hình phân lớp được xây dựng thông qua việc phân tích một tập dữ liệu huấn luyện đã được gán nhãn. Mỗi đối tượng dữ liệu được biểu diễn dưới dạng một vector thuộc tính:  $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)$ , trong đó  $n$  là số lượng thuộc tính, và mỗi vector tương ứng với một nhãn lớp  $y$ . Quá trình huấn luyện có thể được hiểu là quá trình ước lượng một hàm ánh xạ  $f: \mathbf{X} \rightarrow y$ , với hàm ánh xạ  $f$  đóng vai trò dự đoán nhãn lớp đầu ra dựa trên đặc trưng đầu vào. Hàm này có thể được biểu diễn thông qua các luật phân lớp (classification rules), cây quyết định (decision trees), mạng nơron hoặc các biểu thức toán học khác.

Khi bước huấn luyện hoàn tất, mô hình được sử dụng trong giai đoạn phân lớp nhằm gán nhãn cho các đối tượng trong tập dữ liệu kiểm thử (test set) với các mẫu dữ liệu chưa biết nhãn lớp. Nếu bộ phân lớp đạt được chất lượng đủ tốt trên dữ liệu kiểm thử thì bộ phân lớp có thể được triển khai để dự đoán cho các mẫu dữ liệu mới trong thực tế.

Phân lớp là một trong những kỹ thuật nền tảng và được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau như tài chính, thương mại điện tử, y tế, bảo mật, và phân tích hành vi người dùng. Vai trò của phân lớp ngày càng quan trọng trong bối cảnh dữ liệu lớn và nhu cầu tự động hóa ngày càng cao trong các hệ thống thông minh.

#### 2.1.2. Các giải thuật và mô hình phân lớp

Mục tiêu chính của phân lớp là xây dựng một mô hình học từ dữ liệu huấn luyện có nhãn, nhằm dự đoán chính xác nhãn lớp cho các đối tượng dữ liệu mới. Có nhiều phương pháp phân lớp khác nhau được phát triển, mỗi phương pháp có những đặc điểm, ưu điểm và hạn chế riêng, phù hợp với từng loại dữ liệu và mục tiêu ứng dụng cụ thể.

### 2.1.2.1. Giải thuật k-NN

k-NN (k-Nearest Neighbors) (Cover & Hart, 1967) là một trong những giải thuật phân lớp cơ bản dựa trên các điểm lân cận gần nhất trong tập huấn luyện. Khi cần gán nhãn để phân loại cho một đối tượng dữ liệu mới, k-NN sẽ tìm ra k điểm dữ liệu gần nhất và xác định lớp của đối tượng đó dựa theo đa số lớp trong k điểm dữ liệu lân cận.

Giải thuật k-NN thuộc nhóm phương pháp "lazy learning", do đó k-NN không tiến hành huấn luyện mô hình trước như các thuật toán học máy truyền thống. Thay vì trích xuất tri thức khái quát từ dữ liệu huấn luyện, k-NN giữ nguyên toàn bộ dữ liệu và chỉ bắt đầu thực hiện các phép tính khi có yêu cầu phân lớp. Mỗi lần thực hiện phân lớp, k-NN cần tính khoảng cách từ điểm dữ liệu mới đến toàn bộ các điểm trong tập huấn luyện, sau đó chọn ra k điểm dữ liệu gần nhất để xác định nhãn lớp. Vì vậy, khi áp dụng trên các tập dữ liệu lớn, k-NN thường đòi hỏi chi phí tính toán cao, ảnh hưởng đến hiệu năng xử lý.

Nhờ tính đơn giản và hiệu quả trong nhiều trường hợp, k-NN được ứng dụng rộng rãi trong các bài toán như phân loại ảnh, nhận dạng mẫu, hệ thống gợi ý và phát hiện bất thường. Tuy nhiên, hiệu quả của giải thuật k-NN phụ thuộc vào việc lựa chọn tham số k và độ đo khoảng cách sử dụng trong quá trình phân lớp. Việc chọn giá trị k không phù hợp có thể dẫn đến mô hình quá nhạy với nhiễu hoặc làm mất đi tính phân biệt giữa các lớp. Do đó, nhiều nghiên cứu đã tập trung vào tối ưu hóa các tham số này nhằm nâng cao độ chính xác và tính ổn định của thuật toán. Trong đó, nghiên cứu của Sahu & cộng sự (2018) đã đề xuất một phiên bản cải tiến của k-NN, giúp cải thiện độ chính xác so với mô hình truyền thống. Ngoài ra, một nghiên cứu khác tận dụng các thuật toán thống kê để tự động tối ưu hóa giá trị k nhằm áp dụng hiệu quả k-NN vào từng trường hợp cụ thể (Lubis & cộng sự, 2020).

### 2.1.2.2. Cây quyết định

Cây quyết định (Decision Tree) (Quinlan, 1986) là một trong những phương pháp phân lớp phổ biến, trực quan và dễ triển khai trong lĩnh vực học máy và khai phá dữ liệu. Mô hình cây quyết định mô phỏng quá trình ra quyết định bằng một cấu trúc dạng cây, trong đó mỗi nút biểu diễn một phép kiểm tra điều kiện trên thuộc tính dữ liệu, các nhánh tương ứng với các kết quả có thể xảy ra của phép kiểm tra đó và

các nút lá đại diện cho nhãn lớp đầu ra. Khi phân loại một điểm dữ liệu mới, thuật toán bắt đầu từ nút gốc của cây, thực hiện lần lượt các điều kiện kiểm tra tại từng nút và đi theo nhánh phù hợp cho đến khi đến một nút lá, tại đó kết quả phân loại được xác định. Chính nhờ cấu trúc phân cấp này, cây quyết định có khả năng biểu diễn các quyết định dưới dạng các quy tắc luận lý, giúp người dùng dễ dàng giải thích và theo dõi dòng suy luận của mô hình.

Trong quá trình xây dựng cây quyết định, việc lựa chọn thuộc tính tại mỗi nút phân nhánh đóng vai trò quan trọng trong việc đảm bảo hiệu quả và độ chính xác của mô hình. Các độ đo như ID3, Gain Ratio và Gini Index (Han & cộng sự, 2012) thường được sử dụng nhằm đánh giá mức độ phù hợp của từng thuộc tính khi dùng để phân chia dữ liệu. Các độ đo này đều nhằm mục tiêu là tìm ra cách phân tách tập dữ liệu sao cho các tập con tạo thành sau mỗi lần phân chia có mức độ thuần nhất cao nhất có thể với mỗi tập con chứa các đối tượng thuộc cùng một lớp càng nhiều càng tốt. Việc áp dụng các độ đo này không chỉ giúp chọn được thuộc tính tốt nhất để tách dữ liệu tại mỗi bước mà còn góp phần xây dựng một cây quyết định ngắn gọn, dễ hiểu và có khả năng khái quát hóa tốt hơn trên dữ liệu mới. Bằng cách tối ưu hóa độ thuần nhất của các tập con, cây quyết định có thể đạt được độ chính xác cao hơn trong các bài toán phân lớp và khai phá tri thức từ dữ liệu.

Một ưu điểm của cây quyết định là khả năng làm việc tốt với cả dữ liệu rời rạc lẫn liên tục, đồng thời không yêu cầu chuẩn hóa dữ liệu đầu vào. Ngoài ra, nhờ tính minh bạch và khả năng giải thích cao, mô hình cây quyết định được ưa chuộng trong nhiều lĩnh vực đòi hỏi sự rõ ràng trong giải thích quyết định, chẳng hạn như chẩn đoán y khoa, kiểm định tín dụng, phân tích rủi ro, hay hệ thống hỗ trợ ra quyết định. Tuy nhiên, bên cạnh những ưu điểm đó, cây quyết định cũng tồn tại những hạn chế nhất định. Mô hình dễ bị ảnh hưởng bởi dữ liệu nhiễu hoặc các thuộc tính không quan trọng, dẫn đến cấu trúc cây phức tạp không cần thiết. Hơn nữa, cây quyết định đơn thường thiếu tính ổn định khi một sự thay đổi nhỏ trong dữ liệu huấn luyện cũng có thể dẫn đến một cấu trúc cây hoàn toàn khác. Để khắc phục nhược điểm này, các phương pháp học tổ hợp như Random Forest, được Leo Breiman giới thiệu vào năm 2001 (Breiman, 2001), ra đời nhằm kết hợp nhiều cây quyết định nhằm tạo ra mô hình phân lớp mạnh mẽ và ổn định hơn.

### 2.1.2.3. Mạng nơ ron nhân tạo

Mạng nơ ron nhân tạo (Artificial Neural Network - ANN) (Zhang, 2002) là một trong những mô hình học máy được ứng dụng rộng rãi trong các bài toán phân lớp nhờ khả năng xử lý linh hoạt và hiệu quả trên nhiều loại dữ liệu khác nhau. Mạng nơ-ron được thiết kế dựa trên cách thức hoạt động của các nơ ron sinh học trong bộ não con người với mục tiêu mô phỏng khả năng học hỏi và suy luận của con người. Điểm nổi bật của mạng nơ ron là khả năng học và mô hình hóa các mối quan hệ phi tuyến phức tạp giữa dữ liệu đầu vào và nhãn đầu ra, điều mà nhiều phương pháp học máy tuyến tính truyền thống không thể thực hiện hiệu quả. Khả năng xử lý phi tuyến mạnh mẽ này đã được chứng minh trong nhiều ứng dụng như dự báo tài ngắn hạn, nơi ANN vượt trội so với các mô hình tuyến tính truyền thống. Do đó, mạng nơ ron đặc biệt phù hợp để xử lý các tập dữ liệu có cấu trúc phức tạp, đa chiều, hoặc không thể phân lớp bằng các phương pháp tuyến tính đơn giản.

Trong tác vụ phân lớp, mạng nơ ron hoạt động bằng cách tiếp nhận một vector đặc trưng đầu vào và truyền tín hiệu này qua một chuỗi các lớp, bao gồm lớp đầu vào, một hoặc nhiều lớp ẩn và cuối cùng là lớp đầu ra. Mỗi lớp ẩn bao gồm nhiều nơ ron, trong đó mỗi nơ ron thực hiện phép biến đổi tuyến tính thông qua việc nhân đầu vào với trọng số đã học, cộng thêm một hệ số điều chỉnh (bias), sau đó áp dụng một hàm kích hoạt phi tuyến để tạo ra đầu ra. Các hàm kích hoạt phổ biến như ReLU, sigmoid và softmax thường được sử dụng ở lớp đầu ra để chuẩn hóa đầu ra thành xác suất trong các bài toán phân lớp. Kết quả cuối cùng của mạng nơ ron là một giá trị xác suất hoặc nhãn lớp được dự đoán, cho phép mạng có thể giải quyết cả bài toán phân lớp nhị phân và phân lớp đa lớp một cách hiệu quả.

Một trong những lợi thế quan trọng nhất của mạng nơ ron là khả năng tự động học và trích xuất đặc trưng từ dữ liệu mà không cần phải thiết kế thủ công các đặc trưng đầu vào. Vì vậy mạng nơ-ron đạt được hiệu suất cao trong nhiều ứng dụng phân lớp như thị giác máy tính, xử lý ngôn ngữ tự nhiên, phân tích hành vi người dùng và nhiều lĩnh vực khác trong khoa học dữ liệu và trí tuệ nhân tạo. Khi được mở rộng với nhiều lớp ẩn và số lượng lớn nơ ron, mạng nơ ron trở thành một mô hình học sâu có khả năng học các biểu diễn trừu tượng và phức tạp hơn từ dữ liệu.



### 2.1.3. Các độ đo đánh giá chất lượng phân lớp dữ liệu

Trong bài toán phân lớp, để đánh giá hiệu quả hoạt động của mô hình, một công cụ cơ bản và trực quan thường được sử dụng là ma trận nhầm lẫn (*confusion matrix*). Ma trận này cho phép phân tích chi tiết các trường hợp mô hình dự đoán đúng hoặc sai, từ đó tính toán các độ đo đánh giá khác nhau.

Đối với bài toán phân lớp nhị phân (chỉ có hai lớp: dương và âm), ma trận nhầm lẫn là một bảng 2x2 gồm bốn ô:

Bảng 2.1. Ma trận nhầm lẫn

	Dự đoán: Dương (Positive)	Dự đoán: Âm (Negative)
Thực tế: Dương	TP (True Positive)	FN (False Negative)
Thực tế: Âm	FP (False Positive)	TN (True Negative)

Trong đó:

- TP: Số lượng mẫu thuộc lớp dương dự đoán đúng
- TN: Số lượng mẫu thuộc lớp âm dự đoán đúng
- FP: Số lượng mẫu thuộc lớp âm bị dự đoán nhầm thành lớp dương
- FN: Số lượng mẫu thuộc lớp dương bị dự đoán nhầm thành lớp âm

Thuật ngữ lớp dương (positive) và lớp âm (negative) dùng để chỉ các lớp trong bài toán phân lớp dữ liệu, trong đó lớp dương đại diện cho lớp mà mô hình cần phát hiện còn lớp âm tính các lớp còn lại.

Từ ma trận nhầm lẫn, các độ đo đánh giá chất lượng phân lớp (Han & cộng sự, 2012) được tính toán cụ thể như sau:

#### ▪ Accuracy

Độ đo Accuracy là một trong những chỉ số quan trọng trong đánh giá hiệu suất của các mô hình học máy và thống kê. Accuracy được định nghĩa là tỉ lệ phần trăm của các dự đoán đúng trên tổng số các dự đoán.

Công thức tính độ đo Accuracy :

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Độ đo Accuracy là một chỉ số đơn giản và dễ hiểu, dễ dàng tính toán và được sử dụng phổ biến để đánh giá hiệu suất mô hình phân loại. Tuy nhiên, Accuracy lại nhạy cảm với sự mất cân bằng lớp, trong các tập dữ liệu không cân bằng khi một lớp chiếm tỷ lệ vượt trội so với lớp còn lại, chỉ số Accuracy có thể gây hiểu lầm. Ví dụ,

nếu 95% dữ liệu thuộc một lớp và mô hình luôn dự đoán lớp đó, thì accuracy sẽ đạt 95%, dù mô hình thực tế không hề phân biệt được giữa các lớp. Vì vậy, trong nhiều tình huống, Accuracy không phản ánh đầy đủ hiệu suất mô hình. Khi đó, các độ đo khác như Precision, Recall và F1-score thường cung cấp cái nhìn toàn diện và chính xác hơn về khả năng phân loại thực sự của mô hình.

#### ▪ Precision

Độ đo Precision là một chỉ số quan trọng khác trong đánh giá hiệu suất của các mô hình phân lớp, đặc biệt là trong các bài toán mà sự cân bằng giữa các lớp không đồng đều. Precision được định nghĩa là tỷ lệ giữa số lượng dự đoán đúng cho lớp dương và tổng số dự đoán là dương. Công thức tính precision như sau:

$$Precision = \frac{TP}{TP + FP}$$

Precision đo lường tỷ lệ các mẫu thực sự thuộc lớp dương trong số các mẫu mà mô hình dự đoán là dương. Khi Precision cao, điều đó có nghĩa là mô hình ít mắc sai lầm trong việc dự đoán lớp dương bị sai (False Positives) hay nói cách khác, các dự đoán dương mà mô hình đưa ra là đáng tin cậy. Tuy nhiên, cần lưu ý rằng Precision cao không có nghĩa là mô hình luôn đúng khi gặp một mẫu thuộc lớp dương, mà khi mô hình đưa ra dự đoán là dương, thì phần lớn các dự đoán đó là chính xác. Điều này đặc biệt quan trọng trong những bài toán có rủi ro cao hoặc yêu cầu độ chính xác cao trong kết quả dương, chẳng hạn như phát hiện gian lận và chẩn đoán y tế. Trong phát hiện gian lận, ví dụ như trong lĩnh vực tài chính, Precision cao đồng nghĩa với việc nếu hệ thống cảnh báo một giao dịch là gian lận, thì khả năng cao đó thực sự là gian lận, giúp tránh làm phiền người dùng hợp pháp. Tương tự, trong chẩn đoán y tế, Precision cao giúp đảm bảo rằng khi hệ thống xác định một bệnh nhân mắc bệnh, thì khả năng chẩn đoán đúng là rất lớn, từ đó tránh gây lo lắng không cần thiết cho bệnh nhân khỏe mạnh. Như vậy, Precision đóng vai trò then chốt trong việc đánh giá độ tin cậy của các dự đoán dương, đặc biệt trong các bài toán đòi hỏi sự cẩn trọng cao.

#### ▪ Recall

Recall còn được gọi là Sensitivity là một độ đo trong các bài toán mà việc bỏ sót các mẫu thuộc lớp dương có thể dẫn đến những ảnh hưởng nghiêm trọng. Recall cho biết trong số các mẫu, mẫu nào thực sự thuộc lớp dương, có bao nhiêu mẫu được dự đoán đúng. Công thức tính độ đo Recall:

$$Recall = \frac{TP}{TP + FN}$$

Recall tập trung vào việc giảm thiểu các trường hợp lớp dương bị bỏ sót (False Negatives). Điều này đặc biệt hữu ích trong các bài toán có dữ liệu không cân bằng, nơi mà lớp dương (lớp quan tâm) thường xuất hiện với tần suất thấp, như trong chẩn đoán bệnh hoặc phát hiện gian lận tài chính. Trong những trường hợp như vậy, Recall cao có thể mang ý nghĩa quyết định, giúp kịp thời phát hiện bệnh tật hoặc ngăn chặn hành vi gian lận. Tuy nhiên, Recall cũng rất dễ bị ảnh hưởng bởi số lượng False Negatives, nếu mô hình bỏ sót quá nhiều mẫu dương, chỉ số này sẽ giảm mạnh, từ đó làm giảm hiệu quả ứng dụng của mô hình trong thực tế.

Recall và Precision là hai độ đo có mối quan hệ bổ sung nhưng thường đối lập nhau. Trong khi Recall chú trọng vào việc phát hiện tất cả các trường hợp dương, ngay cả khi điều đó dẫn đến việc mô hình có thể đưa ra nhiều dự đoán sai lớp dương (False Positives), thì Precision lại tập trung vào việc đảm bảo rằng các dự đoán dương là chính xác, tức là giảm thiểu số lượng False Positives. Việc lựa chọn ưu tiên Recall hay Precision phụ thuộc vào bối cảnh và mục tiêu cụ thể của bài toán. Trong một số trường hợp như phát hiện ung thư trong chẩn đoán bệnh, việc bỏ sót một trường hợp dương là không thể chấp nhận và Recall cần được ưu tiên. Ngược lại, trong các hệ thống khuyến nghị hoặc cảnh báo giao dịch, việc đưa ra cảnh báo sai có thể gây phiền hà cho người dùng, nên Precision sẽ được chú trọng hơn. Do đó, để cân bằng giữa hai chỉ số này, người ta thường sử dụng F1-score, một chỉ số tổng hợp hài hòa giữa Precision và Recall, giúp đánh giá mô hình một cách toàn diện hơn khi tồn tại sự đánh đổi giữa hai mục tiêu.

#### ▪ F1- score

F1-score là một độ đo quan trọng trong các bài toán khi có sự không cân bằng giữa hai độ đo Precision và Recall. F1-score giúp cân bằng giữa Precision và Recall, đặc biệt khi dữ liệu có sự lệch lớp (imbalanced data), tức là khi một lớp chiếm tỷ lệ lớn hơn đáng kể so với lớp còn lại. Công thức tính độ đo F1 - score :

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Nếu giữa Precision và Recall có sự chênh lệch lớn, F1-score sẽ phản ánh sự mất cân bằng đó bằng cách cho ra một giá trị thấp hơn, giúp tránh việc đánh giá không

chính xác mô hình khi một trong hai độ đo vượt trội. F1-score đặc biệt hữu ích trong các tình huống có sự đánh đổi giữa Precision và Recall và giúp xác định điểm cân bằng tối ưu giữa hai yếu tố này, từ đó cung cấp một cái nhìn khách quan hơn về hiệu suất mô hình. Điều này càng trở nên quan trọng trong các bài toán có dữ liệu lệch lớp, nơi mà chỉ số Accuracy có thể gây hiểu nhầm, khiến mô hình tưởng như hoạt động hiệu quả trong khi thực chất lại bỏ sót nhiều trường hợp quan trọng hoặc đưa ra nhiều dự đoán sai lệch.

#### **2.1.4. Phân chia dữ liệu huấn luyện và kiểm tra**

Việc lựa chọn phương pháp phân chia phù hợp không chỉ ảnh hưởng đến độ chính xác của mô hình mà còn phản ánh khả năng mô hình hóa đối với dữ liệu thực tế. Có nhiều kỹ thuật phân chia dữ liệu đã được đề xuất và áp dụng tùy thuộc vào đặc điểm của tập dữ liệu và mục tiêu nghiên cứu. Dưới đây là một số phương pháp phân chia dữ liệu phổ biến được trình bày trong tài liệu chuyên ngành của Han & cộng sự (2012).

##### **2.1.4.1. Phương pháp Holdout và Random Subsampling**

Phương pháp giữ Holdout là một trong những kỹ thuật đơn giản và phổ biến nhất để đánh giá hiệu suất của các mô hình học máy. Trong phương pháp này, toàn bộ tập dữ liệu ban đầu được chia ngẫu nhiên thành hai tập con: tập huấn luyện (training set) và tập kiểm tra (test set). Thông thường, khoảng hai phần ba dữ liệu được sử dụng để huấn luyện mô hình, trong khi phần còn lại được dùng để kiểm tra độ chính xác của mô hình đã được xây dựng. Tập huấn luyện là cơ sở để mô hình học và phát hiện ra các mẫu hoặc quy luật từ dữ liệu, còn tập kiểm tra giúp đánh giá khả năng tổng quát hóa của mô hình đối với các dữ liệu mới chưa từng thấy trước đó.

Mặc dù phương pháp holdout đơn giản và dễ triển khai, nhưng mô hình được đánh giá theo cách này có thể cho ra ước lượng hiệu suất thấp hơn thực tế, do không tận dụng toàn bộ tập dữ liệu trong quá trình huấn luyện. Việc không tận dụng toàn bộ dữ liệu khiến mô hình khó học được đầy đủ các đặc điểm tiềm ẩn trong tập dữ liệu, dẫn đến việc hiệu suất thu được trên tập kiểm tra không phản ánh đúng năng lực tổng quát hóa của mô hình trong thực tế, đặc biệt khi kích thước tập dữ liệu nhỏ hoặc khi dữ liệu có sự đa dạng cao, khiến kết quả đánh giá phụ thuộc nhiều vào cách chia dữ liệu ban đầu.

Để khắc phục hạn chế này, một biến thể của Holdout là Random Subsampling đã được đề xuất. Trong kỹ thuật này, quá trình phân chia dữ liệu theo phương pháp Holdout được lặp lại  $k$  lần, mỗi lần sử dụng một cách chia ngẫu nhiên khác nhau. Mỗi vòng lặp sẽ cho ra một mô hình riêng biệt và một kết quả đánh giá hiệu suất tương ứng. Cuối cùng, giá trị trung bình của các lần đánh giá sẽ được sử dụng làm ước lượng tổng thể cho hiệu suất của mô hình. Phương pháp này không chỉ cải thiện tính ổn định và độ tin cậy trong quá trình đánh giá, mà còn giúp giảm thiểu tác động của sự ngẫu nhiên từ việc phân chia dữ liệu ban đầu.

#### **2.1.4.2. Kiểm định chéo (cross validation)**

Kiểm định chéo là một kỹ thuật đánh giá hiệu suất mô hình phổ biến trong học máy và khai phá dữ liệu, giúp kiểm tra khả năng tổng quát hóa của mô hình đối với dữ liệu chưa thấy trước đó. Phương pháp này hoạt động bằng cách chia tập dữ liệu ban đầu thành nhiều phần nhỏ (folds), sau đó lặp lại quá trình huấn luyện và kiểm tra mô hình trên các phần khác nhau. Trong các kỹ thuật kiểm định chéo, kiểm định chéo  $k$ -fold là phương pháp phổ biến và được sử dụng rộng rãi nhất do khả năng cân bằng tốt giữa hiệu quả đánh giá và chi phí tính toán. Phương pháp này không chỉ giúp tận dụng toàn bộ dữ liệu cho cả hai mục đích huấn luyện và kiểm tra, mà còn hạn chế được sự thiên lệch trong quá trình đánh giá mô hình do cách chia dữ liệu ngẫu nhiên.  $k$ -fold chia ngẫu nhiên tập dữ liệu ban đầu thành  $k$  tập con không giao nhau, gọi là các "fold" (ký hiệu  $D_1, D_2, \dots, D_k$ ), với kích thước xấp xỉ bằng nhau. Quá trình huấn luyện và kiểm tra mô hình sẽ được lặp lại  $k$  lần, mỗi lần tương ứng với một fold khác nhau đóng vai trò là tập kiểm tra, trong khi  $k - 1$  fold còn lại được dùng làm tập huấn luyện. Cụ thể, tại lần huấn luyện thứ  $i$ , fold  $D_i$  sẽ được giữ lại làm tập kiểm tra, còn các fold còn lại ( $D_1, \dots, D_{i-1}, D_{i+1}, \dots, D_k$ ) được kết hợp để huấn luyện mô hình. Ví dụ, trong lần huấn luyện đầu tiên, mô hình được huấn luyện trên  $D_2$  đến  $D_k$  và kiểm tra trên  $D_1$ ; ở lần huấn luyện thứ hai, huấn luyện trên  $D_1$  và  $D_3$  đến  $D_k$  và thực hiện kiểm tra trên dữ liệu  $D_2$ ; quá trình tiếp tục như vậy cho đến khi mỗi fold đã được dùng làm tập kiểm tra đúng một lần.

Khác với các phương pháp như holdout hay random subsampling, trong  $k$ -fold, mỗi mẫu dữ liệu đều được sử dụng đúng một lần để kiểm tra và  $(k - 1)$  lần để huấn luyện, giúp đảm bảo tính công bằng và ổn định trong quá trình đánh giá. Đối với bài

toán phân lớp, độ chính xác tổng thể của mô hình được tính bằng cách lấy kết quả trung bình của  $k$  lần kiểm tra.

$k$ -fold thường được đánh giá cao vì khả năng tận dụng tối đa dữ liệu để huấn luyện mà vẫn giữ được tính khách quan trong kiểm tra, từ đó cung cấp một ước lượng chính xác và đáng tin cậy hơn về hiệu suất thực sự của mô hình.

Trong các trường hợp làm việc với tập dữ liệu không cân bằng, một biến thể thường được sử dụng thay cho phương pháp  $k$ -fold truyền thống là  $k$ -fold stratified. Phương pháp này không chỉ chia dữ liệu thành  $k$  phần bằng nhau, mà còn đảm bảo rằng mỗi phần đều duy trì được tỷ lệ phân bố giữa các lớp tương tự như trong toàn bộ tập dữ liệu gốc. Cách tiếp cận này giúp tránh tình trạng một số phần ngẫu nhiên có quá ít hoặc không có mẫu thuộc một lớp nhất định dẫn đến đánh giá sai lệch về hiệu suất mô hình. Nhờ khả năng duy trì cấu trúc phân lớp ổn định trong từng phần,  $k$ -fold stratified không những tối ưu hóa việc sử dụng dữ liệu huấn luyện mà còn đảm bảo tính khách quan và độ tin cậy cao hơn trong quá trình đánh giá, đặc biệt hiệu quả trong các bài toán phân loại có sự mất cân bằng về phân bố nhãn.

#### 2.1.4.3. Phương pháp Bootstrap

Bootstrap là một kỹ thuật đánh giá hiệu suất mô hình học máy khác biệt so với các phương pháp như holdout hay kiểm định chéo ở cách thức chọn mẫu. Cụ thể, bootstrap tiến hành lấy mẫu ngẫu nhiên có hoàn lại từ tập dữ liệu ban đầu với một mẫu dữ liệu sau khi được chọn có thể được chọn lại nhiều lần trong quá trình tạo tập huấn luyện. Ví dụ khi chọn một mẫu để đưa vào tập huấn luyện thì mẫu đó vẫn có khả năng được chọn lại và đưa vào tập huấn luyện thêm một hoặc nhiều lần nữa.

Một trong những biến thể phổ biến của phương pháp này là bootstrap .632 với cách hoạt động như sau: Giả sử tập dữ liệu ban đầu gồm  $d$  mẫu, thuật toán sẽ thực hiện lấy mẫu ngẫu nhiên có hoàn lại  $d$  lần để tạo ra một tập huấn luyện mới. Do quá trình lấy mẫu có hoàn lại, nên một số mẫu dữ liệu ban đầu có thể xuất hiện nhiều lần, trong khi một số khác có thể không xuất hiện. Những mẫu không được chọn vào tập huấn luyện sẽ tạo thành tập kiểm tra. Trung bình, khi số lượng mẫu đủ lớn, khoảng 63.2% dữ liệu ban đầu sẽ xuất hiện trong tập huấn luyện, còn 36.8% còn lại sẽ nằm trong tập kiểm tra. Con số 63.2% được xác định theo xác suất: mỗi mẫu có xác suất  $\frac{1}{d}$  được chọn trong một lần, nên xác suất không được chọn là  $1 - \frac{1}{d}$ . Sau  $d$  lần lặp, xác

suất một mẫu không bao giờ được chọn là  $\left(1 - \frac{1}{d}\right)^d$ , khi  $d$  lớn, biểu thức này tiệm cận  $e^{-1} \approx 0.368$ , tức là khoảng 36.8% mẫu không được chọn và các mẫu đó sẽ thuộc tập kiểm tra.

Để tăng độ chính xác của phép ước lượng, quá trình lấy mẫu và đánh giá có thể được lặp lại  $k$  lần. Mỗi lần lặp, mô hình được huấn luyện từ bộ dữ liệu huấn luyện được lựa chọn bằng phương pháp bootstrap và đánh giá trên tập kiểm tra. Độ chính xác trung bình của một mô hình  $M$  sẽ được tính bằng công thức:

$$Acc(M) = \frac{1}{k} \sum_{i=1}^k (0.632 \cdot ACC(M_i)_{test} + 0.368 \cdot ACC(M_i)_{train})$$

Trong đó,  $ACC(M_i)_{test}$  là độ chính xác của mô hình  $M_i$  (thu được từ lần lấy mẫu thứ  $i$ ) khi áp dụng trên tập kiểm tra, còn  $ACC(M_i)_{train}$  là độ chính xác của mô hình đó khi áp dụng trên toàn bộ tập dữ liệu ban đầu. Phương pháp bootstrap có xu hướng lạc quan trong việc đánh giá hiệu suất mô hình, nhưng lại đặc biệt hiệu quả đối với các tập dữ liệu có kích thước nhỏ.

## 2.2. Phân tích cảm xúc

### 2.2.1. Giới thiệu

Phân tích cảm xúc (*Sentiment Analysis*), còn được gọi là *phân tích ý kiến* (*opinion analysis*) là một lĩnh vực nghiên cứu trong xử lý ngôn ngữ tự nhiên tập trung vào việc trích xuất, xác định và phân loại cảm xúc, thái độ hoặc quan điểm thể hiện trong các đoạn văn bản một cách tự động (B. Liu & Zhang, 2012).

Phân tích cảm xúc được xem là một dạng đặc biệt của bài toán phân lớp dữ liệu trong lĩnh vực học máy và xử lý ngôn ngữ tự nhiên. Mục tiêu của bài toán này là gán một nhãn cảm xúc cho một đơn vị văn bản, có thể là một tài liệu, một câu, hoặc một khía cạnh cụ thể được đề cập trong văn bản. Quá trình này tương tự như các bài toán phân lớp truyền thống, trong đó một mô hình học máy được huấn luyện trên tập dữ liệu có gán nhãn để học cách phân biệt giữa các loại cảm xúc dựa trên các đặc trưng trích xuất từ văn bản như từ vựng, ngữ cảnh, cú pháp. Tuy nhiên, để xây dựng một mô hình phân lớp hiệu quả cho bài toán phân tích cảm xúc, cần phải giải quyết nhiều vấn đề đặc thù liên quan đến hiện tượng từ đa nghĩa, sự đồng thời tồn tại của các thái cực cảm xúc trong cùng một đoạn văn và sự thay đổi ý nghĩa phụ thuộc vào

ngữ cảnh sử dụng. Những yếu tố này đòi hỏi mô hình không chỉ nhận diện đúng các tín hiệu ngôn ngữ bề mặt mà còn phải hiểu sâu sắc ngữ nghĩa và sắc thái biểu đạt trong văn bản.

### 2.2.2. Các cấp độ phân tích cảm xúc

Phân tích cảm xúc chủ yếu bao gồm nhiệm vụ phân loại sắc thái cảm xúc (polarity) của một đoạn văn bản ở cấp độ tài liệu, câu hoặc thuộc tính/khía cạnh, với mục tiêu xác định cảm xúc được thể hiện là tích cực, tiêu cực hay trung tính. Tùy theo mức độ chi tiết và mục tiêu phân tích, phân tích cảm xúc có thể được thực hiện ở ba cấp độ sau:

- **Cấp độ tài liệu:** Ở cấp độ này, mục tiêu chính là xác định cảm xúc tổng thể của toàn bộ văn bản. Khó khăn chính trong việc phân tích cảm xúc cấp độ tài liệu là việc có thể tồn tại đồng thời các phát biểu khách quan và chủ quan trong một văn bản có thể gây nhiễu cho mô hình học máy khi phân tích để xác định cảm xúc chung của bình luận, đặc biệt là khi các cảm xúc trái ngược cùng xuất hiện, khiến cho quá trình phân lớp trở nên phức tạp hơn (Jagtap & Pawar, 2013).

- **Cấp độ câu:** Đây là mức độ phân tích chi tiết hơn so với cấp độ tài liệu. Mỗi câu được đánh giá độc lập để xác định xem nó thể hiện cảm xúc tích cực, tiêu cực hay trung tính với mục đích chính là xác định xem câu đó có liên quan đến chủ đề được đề cập hay không (Jagtap & Pawar, 2013). Tuy nhiên, việc đánh giá cảm xúc ở cấp độ câu cũng gặp phải một số thách thức, đặc biệt là trong các trường hợp câu mang ý nghĩa mơ hồ hoặc chứa nhiều lớp nghĩa.

- **Cấp độ khía cạnh:** Tại cấp độ này, việc phân tích được thực hiện chi tiết hơn, trong đó các kỹ thuật chủ yếu tập trung vào việc mô hình hóa mối quan hệ giữa thuật ngữ khía cạnh (aspect term), danh mục khía cạnh (aspect category), thuật ngữ cảm xúc (opinion term) và chiều hướng cảm xúc (sentiment polarity) (Mao & cộng sự, 2024). Phân tích cảm xúc cấp độ khía cạnh được xem là mô hình phân tích chi tiết nhất trong số các cấp độ phân tích cảm xúc (Zhai & cộng sự, 2011).

### 2.2.3. Mô hình học sâu

Deep learning là một phương pháp học máy sử dụng các mạng nơ ron nhân tạo nhiều lớp xử lý để học biểu diễn dữ liệu ở nhiều mức độ trừu tượng khác nhau. Nhờ thuật toán lan truyền ngược (backpropagation), các hệ thống này tự động điều chỉnh



tham số để cải thiện hiệu suất, giúp đạt được những tiến bộ đáng kể trong nhiều lĩnh vực. Khác với các phương pháp truyền thống yêu cầu thiết kế thủ công các bộ rút xuất đặc trưng phức tạp, deep learning có khả năng tự động học các đặc trưng từ dữ liệu thô thông qua nhiều lớp biến đổi phi tuyến tính. Sự phát triển mạnh mẽ của deep learning hiện nay phần lớn nhờ vào sự gia tăng của dữ liệu và năng lực tính toán, và hứa hẹn mở ra nhiều ứng dụng tiềm năng khi kết hợp với các phương pháp suy luận phức tạp (LeCun & cộng sự, 2015).

Trong giai đoạn đầu, mạng nơ ron hồi quy (Recurrent Neural Networks - RNN) và các biến thể như LSTM (Long Short-Term Memory) hoặc GRU (Gated Recurrent Unit) được sử dụng phổ biến để xử lý dữ liệu tuần tự trong ngôn ngữ. Các mô hình này có khả năng ghi nhớ thông tin trong chuỗi và mô hình hóa ngữ cảnh của từ dựa trên lịch sử trước đó. Tuy nhiên, các mô hình này gặp khó khăn khi xử lý các văn bản dài, cũng như khả năng mô hình hóa ngữ cảnh dài hạn còn hạn chế.

Sự xuất hiện của kiến trúc Transformer đã tạo ra sự thay đổi lớn trong xử lý ngôn ngữ tự nhiên. Transformer sử dụng cơ chế tự chú ý (self-attention) để học mối quan hệ giữa các từ trong toàn bộ câu hoặc đoạn văn mà không cần xử lý theo thứ tự tuần tự như RNN, từ đó cải thiện đáng kể hiệu quả tính toán và khả năng biểu diễn ngữ nghĩa. Trên nền tảng Transformer, một loạt mô hình ngôn ngữ tiên huấn luyện quy mô lớn đã ra đời như BERT, GPT, RoBERTa, XLNet và T5. Những mô hình này được huấn luyện trên các bộ dữ liệu văn bản rất lớn và có khả năng học được biểu diễn ngữ nghĩa đa chiều, giàu thông tin về ngữ cảnh.

Một trong những đặc điểm nổi bật của các mô hình học sâu hiện đại trong xử lý ngôn ngữ tự nhiên là khả năng tinh chỉnh (fine-tuning) để phù hợp cho các nhiệm vụ cụ thể mà không cần huấn luyện lại từ đầu. Điều này giúp tiết kiệm thời gian và tài nguyên tính toán, đồng thời nâng cao độ chính xác và hiệu quả triển khai trên thực tế.

Với sự kết hợp giữa kiến trúc tiên tiến, dữ liệu huấn luyện lớn và kỹ thuật điều chỉnh linh hoạt, các mô hình học sâu không chỉ nâng cao năng lực xử lý ngôn ngữ của máy tính mà còn mở ra nhiều triển vọng cho việc phát triển các hệ thống giao tiếp người - máy thông minh hơn trong tương lai.

## 2.2.4. Các phương pháp biểu diễn văn bản thành vector

### 2.2.4.1. Bag-of-word (BoW)

Bag-of-Words (BoW) (Harris, 1954) là một trong những phương pháp đơn giản và phổ biến nhất để biểu diễn văn bản dưới dạng vector trong xử lý ngôn ngữ tự nhiên. Phương pháp này hoạt động dựa trên nguyên lý là mỗi văn bản sẽ được biểu diễn bằng một vector số, trong đó độ dài của vector tương ứng với kích thước của bộ từ vựng. Mỗi vị trí trong vector đại diện cho một từ cụ thể trong từ điển và giá trị tại vị trí đó phản ánh số lần từ này xuất hiện trong văn bản. Cách biểu diễn này không quan tâm đến ngữ pháp, trật tự các từ trong câu, mà chỉ tập trung vào tần suất xuất hiện của từ ngữ, giúp mô hình học máy có thể xử lý và phân tích văn bản dưới dạng số liệu.

Ví dụ, xét hai câu sau:

*Câu 1: "Tôi thích học máy."*

*Câu 2: "Máy học rất thú vị."*

Bộ từ vựng sẽ là: ["Tôi", "thích", "học", "máy", "rất", "thú", "vị"] (7 từ).

Khi đó, vector BoW của hai câu sẽ là:

Câu 1: [1, 1, 1, 1, 0, 0, 0]

Câu 2: [0, 0, 1, 1, 1, 1, 1]

BoW đơn giản và dễ triển khai vì không yêu cầu xây dựng mô hình phức tạp, chỉ cần đếm tần suất xuất hiện của từ trong văn bản. Phương pháp này đặc biệt hiệu quả đối với các bài toán xử lý ngôn ngữ tự nhiên cơ bản có dữ liệu ít nhiễu như phát hiện thư rác, phân loại chủ đề hay phân tích cảm xúc đơn giản. Ngoài ra, do chỉ sử dụng phép đếm nên BoW có thể xử lý một lượng lớn văn bản với tốc độ cao, tiết kiệm thời gian tính toán.

Tuy nhiên, BoW cũng tồn tại nhiều hạn chế, một trong những nhược điểm lớn nhất là việc bỏ qua thứ tự và ngữ cảnh của từ trong câu. Phương pháp không quan tâm đến vị trí hay quan hệ giữa các từ nên những câu có thứ tự từ khác nhau nhưng mang ý nghĩa hoàn toàn trái ngược có thể được biểu diễn bởi cùng một vector. Thêm vào đó, BoW không thể nắm bắt được ngữ nghĩa của từ: các từ đồng nghĩa như "máy tính" và "computer" được xem là hai từ hoàn toàn khác nhau, còn các từ đa nghĩa cũng không được phân biệt. Cuối cùng, BoW tạo ra các vector có kích thước lớn và

rất thưa (đa số phần tử bằng 0) khi bộ từ vựng mở rộng, dẫn đến lãng phí bộ nhớ và giảm hiệu quả khi áp dụng cho các mô hình học sâu, vốn cần thông tin ngữ nghĩa phong phú hơn để hoạt động hiệu quả.

#### 2.2.4.2. Term Frequency-Inverse Document Frequency (TF-IDF)

Để khắc phục một phần hạn chế của BoW, TF-IDF được phát triển nhằm tính đến mức độ quan trọng của một từ trong một văn bản cụ thể so với toàn bộ tập dữ liệu (Salton & cộng sự, 1975). Phương pháp này dựa trên hai thành phần chính là Tần suất từ trong văn bản (Term Frequency) và Tần suất đảo ngược trong tập văn bản (Inverse Document Frequency). Tần suất từ trong văn bản đo lường số lần một từ xuất hiện trong một văn bản cụ thể, thường được chuẩn hóa bằng cách chia cho tổng số từ trong văn bản để tránh thiên lệch do độ dài văn bản khác nhau với công thức:

$$TF(t, d) = \frac{\text{Số lần từ } t \text{ xuất hiện trong văn bản } d}{\text{Tổng số từ trong văn bản } d}$$

Trong khi đó, tần suất đảo ngược trong tập văn bản đánh giá mức độ hiếm của từ trong toàn bộ tập dữ liệu, giúp giảm trọng số của các từ phổ biến vì chúng ít mang thông tin phân biệt theo công thức sau:

$$IDF(t, D) = \log \left( \frac{\text{Tổng số văn bản trong tập dữ liệu } D}{\text{Số văn bản chứa từ } t} \right)$$

Giá trị TF-IDF của một từ trong văn bản được tính bằng:

$$TF - IDF(t, d, D) = TF(t, d) \cdot IDF(t, D)$$

TF-IDF có một số ưu điểm nổi bật so với phương pháp BoW trong việc biểu diễn văn bản thành vector. Một trong những ưu điểm của TF-IDF là khả năng giảm thiểu ảnh hưởng của các từ phổ biến trong tập dữ liệu, những từ xuất hiện nhiều ở hầu hết các văn bản như "và", "là", "của" sẽ được gán trọng số thấp. Ngược lại, các từ hiếm nhưng mang tính đặc trưng cao sẽ được tăng cường trọng số, từ đó giúp mô hình tập trung vào những yếu tố mang tính phân biệt. Chính vì vậy, TF-IDF đặc biệt phù hợp cho các bài toán tìm kiếm và trích xuất thông tin như trong hệ thống máy tìm kiếm, TF-IDF giúp xếp hạng kết quả dựa trên độ liên quan của từ khóa truy vấn, hay trong trích xuất từ khóa, nó hỗ trợ xác định các thuật ngữ quan trọng nhất trong văn bản. Ngoài ra, khi được kết hợp với các mô hình học máy truyền thống như Naive Bayes hoặc SVM, TF-IDF thường giúp cải thiện đáng kể hiệu suất phân loại văn bản so với BoW.

Tuy nhiên, TF-IDF cũng tồn tại một số nhược điểm giống BoW như không nắm bắt được ngữ nghĩa hay thứ tự từ trong câu. Do đó, không thể hiện được mối quan hệ giữa các từ có nghĩa gần nhau, cũng như không phân biệt được các ngữ cảnh khác nhau của từ đa nghĩa. Bên cạnh đó, phương pháp này cũng tạo ra các vector có độ thưa cao và kích thước lớn khi bộ từ vựng mở rộng, dẫn đến vấn đề về không gian lưu trữ và tính toán. Hiệu quả của TF-IDF cũng phụ thuộc đáng kể vào quá trình tiền xử lý văn bản để đảm bảo các trọng số phản ánh đúng mức độ quan trọng của từ trong ngữ cảnh.

#### **2.2.4.3. Word Embedding**

Một trong những bước tiến quan trọng nhất trong lĩnh vực xử lý ngôn ngữ tự nhiên là sự ra đời của các kỹ thuật Word Embedding như Word2Vec (Mikolov & cộng sự, 2013), GloVe (Pennington & cộng sự, 2014) và FastText (Joulin & cộng sự, 2016). Khác với các phương pháp truyền thống như BoW hay TF-IDF chỉ xem xét tần suất xuất hiện của từ mà bỏ qua ngữ nghĩa, Word Embedding đã mang đến một cách tiếp cận hoàn toàn mới bằng cách ánh xạ các từ vào một không gian vector liên tục nhiều chiều. Trong không gian này, những từ có ngữ nghĩa tương đồng hoặc xuất hiện trong ngữ cảnh tương tự sẽ được đặt gần nhau, cho phép máy tính không chỉ nhận biết từ mà còn hiểu được mối quan hệ giữa chúng.

Word2Vec là một trong những mô hình biểu diễn từ đầu tiên và có ảnh hưởng sâu rộng. Mô hình này hoạt động dựa trên hai kiến trúc chính: Skip-gram và CBOW. Skip-gram có mục tiêu dự đoán các từ xung quanh dựa trên từ mục tiêu. Trong khi đó, CBOW sử dụng các từ xung quanh để dự đoán từ mục tiêu. Ưu điểm nổi bật của Word2Vec là khả năng nắm bắt các mối quan hệ ngữ nghĩa và cú pháp phức tạp thông qua các phép toán vector. Tuy nhiên, Word2Vec chỉ học từ ngữ cảnh cục bộ nên bỏ qua các mẫu xuất hiện toàn cục của từ trong tập dữ liệu.

Khắc phục một số hạn chế của Word2Vec, mô hình GloVe được phát triển bởi Pennington & cộng sự (2014) kết hợp giữa thống kê toàn cục và học ngữ cảnh cục bộ. Thay vì chỉ sử dụng thông tin từ cửa sổ ngữ cảnh nhỏ, GloVe học các vector từ dựa trên ma trận đồng xuất hiện (co-occurrence matrix) của toàn bộ tập dữ liệu. Nhờ đó, GloVe hoạt động hiệu quả trên các tập dữ liệu lớn và có khả năng bảo toàn các quan hệ tỷ lệ giữa các từ.

Một cải tiến khác đến từ FastText mở rộng mô hình Word2Vec bằng cách đưa vào thông tin về các đơn vị từ con (subword). Thay vì chỉ học vector cho từng từ riêng lẻ, FastText biểu diễn mỗi từ bằng tổng các vector của các n-gram ký tự. Nhờ cách tiếp cận này, FastText có thể xử lý tốt các từ hiếm hoặc từ mới bằng cách tổng hợp thông tin từ các subword đã học, đồng thời đặc biệt hiệu quả đối với các ngôn ngữ có cấu trúc hình thái phức tạp như tiếng Việt.

Mặc dù có nhiều ưu điểm hơn so với BoW và TF-IDF trong việc biểu diễn ngữ nghĩa của từ, các phương pháp Word Embedding như Word2Vec hay GloVe vẫn tồn tại một số hạn chế nhất định. Các phương pháp này không xử lý được hiện tượng đa nghĩa, do mỗi từ chỉ được gán một vector duy nhất, bất kể ngữ cảnh sử dụng. Bên cạnh đó, chất lượng của vector ngữ nghĩa phụ thuộc nhiều vào tập dữ liệu huấn luyện. Ngoài ra, Word Embedding chỉ học biểu diễn cho từng từ riêng lẻ, nên không thể nắm bắt được ngữ nghĩa đầy đủ ở cấp độ câu hoặc đoạn văn, vốn quan trọng trong nhiều tác vụ xử lý ngôn ngữ tự nhiên như tóm tắt, dịch máy hay trả lời câu hỏi.

#### **2.2.4.4. Các mô hình ngữ cảnh**

Khác với các phương pháp Word Embedding truyền thống như Word2Vec hoặc GloVe, chỉ tạo ra một vector cố định cho mỗi từ bất kể ngữ cảnh, các mô hình biểu diễn vector dựa theo ngữ cảnh như BERT (Bidirectional Encoder Representations from Transformers) (Devlin & cộng sự, 2019), ELMo (Embeddings from Language Models) (Peters & cộng sự, 2018) và GPT (Generative Pre-trained Transformer) (Radford & cộng sự, 2018), có thể tạo ra vector biểu diễn động tùy theo ngữ cảnh cụ thể của văn bản, qua đó giúp máy tính hiểu được sắc thái ngữ nghĩa phức tạp của ngôn ngữ và cải thiện suất phân tích cho các tác vụ xử lý ngôn ngữ tự nhiên.

Các mô hình biểu diễn văn bản theo ngữ cảnh đều được xây dựng dựa trên kiến trúc học sâu hiện đại, tuy nhiên mỗi mô hình lại có cách tiếp cận riêng để nắm bắt ngữ cảnh ngôn ngữ. Mô hình ELMo sử dụng mạng LSTM hai chiều (Bi-LSTM) để học biểu diễn từ, đồng thời kết hợp các tầng khác nhau của LSTM nhằm tạo ra các vector có khả năng phân biệt giữa các lớp ngữ nghĩa khác nhau như cú pháp, ngữ nghĩa bề mặt và ngữ nghĩa sâu. Nhờ đó, ELMo có thể xử lý hiện tượng đa nghĩa từ tốt hơn so với các phương pháp như Word2Vec hay GloVe. Tuy nhiên, do tính tuần

tự của mạng LSTM, ELMo thường có hiệu suất thấp hơn các mô hình dựa trên kiến trúc Transformer.

Mô hình GPT dựa trên kiến trúc Transformer một chiều, chỉ sử dụng ngữ cảnh phía trước của từ. GPT được huấn luyện theo cơ chế dự đoán từ tiếp theo trong câu, giúp nó đặc biệt mạnh trong các tác vụ sinh văn bản tự nhiên như viết nội dung, hội thoại. Ngoài ra, GPT cũng có thể sử dụng cho tác vụ biểu diễn văn bản thành vector. Tuy nhiên, do chỉ khai thác ngữ cảnh một chiều, GPT không thể nắm bắt đầy đủ mối quan hệ hai chiều giữa các từ như một số mô hình khác.

Khác với GPT, mô hình BERT sử dụng Transformer hai chiều cho phép đồng thời xem xét cả ngữ cảnh trước và sau của từ. Quá trình huấn luyện BERT bao gồm hai nhiệm vụ chính: Masked Language Modeling (MLM) nhằm che giấu một số từ trong câu và yêu cầu mô hình dự đoán chúng, và Next Sentence Prediction (NSP), dùng để xác định xem hai câu có liên tiếp nhau trong văn bản hay không. Nhờ kiến trúc này, BERT có khả năng biểu diễn ngữ nghĩa một cách sâu sắc và chính xác hơn so với ELMo hay GPT, tuy nhiên cũng đòi hỏi tài nguyên tính toán lớn để huấn luyện và triển khai.

Nhờ cơ chế ngữ cảnh hóa, các mô hình biểu diễn văn bản hiện đại như BERT và GPT đã khắc phục hiệu quả những hạn chế của các phương pháp Word Embedding truyền thống. Các mô hình này có khả năng xử lý hiện tượng đa nghĩa từ dựa vào ngữ cảnh, phản ánh đúng ý nghĩa cụ thể trong từng trường hợp. Bên cạnh đó, nhờ kiến trúc Transformer, những mô hình như BERT có thể xử lý tốt các quan hệ ngữ nghĩa giữa các từ nằm cách xa nhau trong câu, giúp hiểu được ngữ cảnh dài và cấu trúc ngữ pháp phức tạp. Đặc biệt, các mô hình này cũng thể hiện khả năng nắm bắt ý nghĩa trong các biểu thức phức tạp như thành ngữ hoặc ẩn dụ.

## **2.3. Xử lý ngôn ngữ tiếng Việt**

### **2.3.1. Biểu diễn văn bản cho ngôn ngữ tiếng Việt**

Đối với tiếng Việt, việc biểu diễn văn bản đặt ra nhiều thách thức đặc thù do những đặc điểm ngôn ngữ học phức tạp. Tiếng Việt là ngôn ngữ đơn âm tiết, có hệ thống thanh điệu phong phú và mang tính phân tách từ không rõ ràng, khiến việc xác định ranh giới từ trở thành một bài toán quan trọng. Không giống như tiếng Anh, nơi khoảng trắng phân tách từ rõ ràng, tiếng Việt yêu cầu quá trình tách từ (tokenization)

chính xác để đảm bảo việc biểu diễn ngữ nghĩa là đúng với văn bản gốc. Ví dụ, cụm từ "sản phẩm tốt" khi thực hiện tách từ thì kết quả sau khi tách sẽ là "sản\_phẩm" và "tốt". Trong tiếng Việt có những từ phải đi cùng nhau thì mới thể hiện được ý nghĩa của câu, do đó việc thực hiện tách từ sẽ có ảnh hưởng quan trọng đến quá trình xử lý biến đổi văn bản thành vector.

Các phương pháp truyền thống như BoW hay TF-IDF có thể được sử dụng để chuyển văn bản tiếng Việt thành vector ngữ nghĩa cho các bài toán xử lý ngôn ngữ tự nhiên. Tuy nhiên, những phương pháp này chủ yếu dựa trên tần suất xuất hiện của từ và bỏ qua hoàn toàn yếu tố ngữ nghĩa và ngữ cảnh. Do đó không thể phân biệt được các từ đồng nghĩa, trái nghĩa hay hiểu được sự đa nghĩa tùy theo ngữ cảnh của từ. Điều này đặc biệt là một thách thức đối với tiếng Việt một ngôn ngữ giàu ngữ nghĩa, ngữ cảnh, và nhiều từ đa nghĩa.

Để khắc phục những hạn chế đó, các phương pháp biểu diễn văn bản hiện đại hơn như Word Embedding và mô hình biểu diễn ngữ cảnh đã được nghiên cứu và ứng dụng. Các phương pháp này cho phép mã hóa thông tin ngữ nghĩa phong phú hơn vào vector từ, đồng thời xét đến ngữ cảnh của từ trong câu. Tuy nhiên, do đặc thù ngôn ngữ tiếng Việt và sự hạn chế về tài nguyên, việc xây dựng từ đầu các mô hình này là rất tốn kém và phức tạp. Do đó, các mô hình đã được huấn luyện sẵn cho tiếng Việt như PhoBERT hoặc mô hình theo kiến trúc Transformer, đóng vai trò quan trọng trong việc biểu diễn văn bản tiếng Việt một cách hiệu quả và có tính ngữ cảnh hóa cao.

### 2.3.2. Mô hình PhoW2V

PhoW2V (A. T. Nguyen & cộng sự, 2020) là một mô hình Word Embedding được huấn luyện dành riêng cho tiếng Việt, dựa trên kiến trúc Word2Vec với mục tiêu tạo ra các biểu diễn vector phản ánh ngữ nghĩa và ngữ cảnh sử dụng của từ ngữ tiếng Việt một cách hiệu quả. Mô hình này được xây dựng từ tập dữ liệu văn bản lớn bao gồm các nguồn dữ liệu tiếng Việt với đặc điểm nổi bật là khả năng phản ánh tốt đặc trưng ngôn ngữ tiếng Việt.

Việc sử dụng PhoW2V cho thấy hiệu quả đáng kể trong nhiều tác vụ xử lý ngôn ngữ tự nhiên tiếng Việt (K. Q. Tran & cộng sự, 2021; Van Huynh & cộng sự, 2022). Các vector từ được huấn luyện từ PhoW2V có khả năng thể hiện tốt mối quan

hệ ngữ nghĩa và ngữ cảnh giữa các từ, nhờ đó giúp mô hình học máy phân biệt rõ hơn giữa các từ có ý nghĩa tương đồng hoặc đồng xuất hiện trong những ngữ cảnh cụ thể. Khác với các phương pháp biểu diễn văn bản truyền thống như Bag of Words (BoW) hay TF-IDF vốn chỉ dựa trên tần suất xuất hiện của từ mà không tính đến ngữ cảnh, PhoW2V tạo ra các vector có tính liên kết ngữ nghĩa cao, từ đó hỗ trợ hiệu quả hơn trong việc trích xuất đặc trưng đầu vào cho các thuật toán học sâu hoặc học máy.

Hơn nữa, so với mô hình fastText do Facebook AI phát triển, PhoW2V tuy không khai thác đặc trưng subword nhưng lại thể hiện sự phù hợp cao với văn bản tiếng Việt do được huấn luyện từ dữ liệu thuần Việt, thay vì dữ liệu đa ngữ. Điều này giúp PhoW2V tránh được hiện tượng sai lệch ngữ nghĩa do khác biệt văn hoá và cấu trúc ngôn ngữ. Tuy nhiên, PhoW2V không xem xét ngữ cảnh trong quá trình huấn luyện nên các từ trong văn bản ở các ngữ cảnh khác nhau đều biểu diễn thành các vector giống nhau dẫn đến khó xử lý các trường hợp từ đa nghĩa hoặc ngữ nghĩa của từ phụ thuộc vào ngữ cảnh.

Mặc dù vẫn còn những hạn chế và hiện nay các mô hình biểu diễn văn bản theo ngữ cảnh như BERT hoặc PhoBERT ngày càng phổ biến, nhưng các mô hình Embedding tĩnh như PhoW2V vẫn giữ vai trò quan trọng trong các hệ thống xử lý ngôn ngữ đòi hỏi tốc độ xử lý cao, ít tài nguyên tính toán hoặc cần tích hợp vào các pipeline truyền thống.

### 2.3.3. Mô hình mBERT

mBERT (Multilingual BERT) (Pires & cộng sự, 2019) là một biến thể đa ngữ của mô hình BERT do Google phát triển, được tiền huấn luyện trên tập dữ liệu bao gồm 104 ngôn ngữ khác nhau thông qua kỹ thuật học có giám sát một phần gọi là "masked language modeling" (MLM). Khác với các mô hình đơn ngữ được huấn luyện riêng biệt cho từng ngôn ngữ, mBERT sử dụng một bộ tham số và từ vựng chung cho toàn bộ các ngôn ngữ trong tập huấn luyện. Cách tiếp cận này cho phép mô hình học được các đặc trưng ngữ nghĩa và ngữ cảnh có tính khái quát cao từ dữ liệu đa ngôn ngữ, từ đó hỗ trợ tốt các tác vụ xử lý ngôn ngữ tự nhiên đa ngôn ngữ (cross-lingual NLP) như phân loại văn bản, gán nhãn từ loại, phân tích cảm xúc và dịch máy. Tuy nhiên, do không được tối ưu hóa riêng cho từng ngôn ngữ, mBERT có



thể gặp hạn chế khi xử lý các hiện tượng ngôn ngữ đặc thù hoặc cấu trúc cú pháp phức tạp của từng ngôn ngữ cụ thể.

Việc sử dụng một bộ từ vựng chung cho tất cả các ngôn ngữ trong mBERT dẫn đến những hạn chế đáng kể khi áp dụng cho các ngôn ngữ có đặc điểm hình thái đặc thù như tiếng Việt. Do không được thiết kế riêng cho tiếng Việt, mBERT có xu hướng phân tách các cụm từ đa âm tiết thành các đơn vị từ rời rạc, gây suy giảm tính toàn vẹn ngữ nghĩa trong quá trình biểu diễn văn bản. Ví dụ, cụm từ "sản phẩm" có thể bị tách thành hai đơn vị riêng biệt, làm mất đi ý nghĩa kết hợp vốn có. Hạn chế này ảnh hưởng đáng kể đến hiệu quả của mô hình trong các tác vụ xử lý ngôn ngữ tự nhiên đòi hỏi độ chính xác cao về ngữ nghĩa và ngữ cảnh, đặc biệt là trong việc nhận diện cụm từ cố định hoặc thành ngữ trong tiếng Việt.

Ngoài ra, mBERT không được huấn luyện chuyên biệt trên dữ liệu tiếng Việt mà được đào tạo từ tập ngữ liệu đa ngôn ngữ tổng hợp, dẫn đến khả năng biểu diễn ngữ nghĩa đối với tiếng Việt có thể kém hiệu quả hơn so với các mô hình chuyên biệt như PhoBERT. Tuy nhiên, mBERT vẫn là một lựa chọn phổ biến trong các tác vụ xử lý đa ngôn ngữ, nhờ vào khả năng khái quát hóa trên nhiều ngôn ngữ khác nhau và đặc biệt hữu ích trong các bối cảnh mà mô hình đơn ngữ chuyên biệt chưa sẵn có.

#### 2.3.4. Mô hình PhoBERT

PhoBERT (D. Q. Nguyen & Nguyen, 2020) là một mô hình ngôn ngữ tiền huấn luyện (*pre-trained language model*) được thiết kế cho tiếng Việt, phát triển bởi nhóm nghiên cứu tại VinAI Research. PhoBERT được xây dựng dựa trên kiến trúc RoBERTa (Y. Liu & cộng sự, 2019), một biến thể cải tiến của BERT với khả năng huấn luyện sâu hơn và sử dụng tập dữ liệu huấn luyện lớn hơn. PhoBERT được huấn luyện trên một tập dữ liệu thuần tiếng Việt có quy mô lớn khoảng 20GB văn bản, giúp nắm bắt tốt ngữ nghĩa và các đặc trưng cú pháp đặc thù của tiếng Việt.

PhoBERT được huấn luyện trên văn bản đã được tách từ theo đơn vị từ vựng hoàn chỉnh thay vì theo âm tiết như nhiều mô hình trước đó. Do đó, để đảm bảo đầu vào phù hợp với cấu trúc của mô hình, văn bản tiếng Việt cần được tiền xử lý bằng cách tách từ trước khi đưa vào PhoBERT để tránh dẫn đến sai lệch ngữ nghĩa, vì tiếng Việt là ngôn ngữ đơn lập, trong đó nhiều từ được hình thành từ các cụm âm tiết nên có thể gây hiểu nhầm nếu tách từ không chính xác. PhoBERT có khả năng biểu diễn

văn bản theo ngữ cảnh một cách linh hoạt và chính xác, nhờ vào kiến trúc Transformer cho phép mô hình học được mối quan hệ phụ thuộc giữa các từ trong câu. Thay vì gán một vector cố định cho mỗi từ như các mô hình Word Embedding tĩnh, PhoBERT tạo ra biểu diễn phụ thuộc vào ngữ cảnh cụ thể mà từ đó xuất hiện. Điều này đặc biệt hiệu quả đối với các cụm từ có nghĩa cố định, thành ngữ, hoặc các cấu trúc ngữ pháp phức tạp trong tiếng Việt với sự đa dạng về ngữ cảnh. Vì vậy, PhoBERT giúp nâng cao độ chính xác trong các tác vụ xử lý ngôn ngữ tự nhiên như phân tích cảm xúc, phân loại văn bản hay trích xuất thông tin. Kết quả thực nghiệm cho thấy PhoBERT đạt độ chính xác cao trong một số tác vụ (Dang & cộng sự, 2023; L. T. Nguyen & cộng sự, 2022; Pham Duy & Le Thanh, 2023), cho thấy được hiệu quả của việc huấn luyện mô hình trên dữ liệu tiếng Việt với kiến trúc tối ưu hóa hiện đại. Ngoài ra, PhoBERT còn đóng vai trò làm nền tảng cho các mô hình tinh chỉnh cho các tác vụ đặc thù trong các hệ thống xử lý ngôn ngữ tiếng Việt.

#### 2.4. Phân tích cảm xúc dựa trên khía cạnh

Phân tích cảm xúc theo khía cạnh (ABSA - *Aspect-Based Sentiment Analysis*) là một kỹ thuật phân tích cảm xúc nâng cao, trong đó mục tiêu không chỉ là xác định thái độ tích cực, tiêu cực hay trung lập của người dùng, mà còn phân tích cụ thể cảm xúc hướng đến từng khía cạnh cụ thể của đối tượng. Ví dụ, thay vì chỉ nói "người dùng thích sản phẩm", ABSA sẽ xác định rằng "người dùng thích thiết kế nhưng không hài lòng với hiệu năng của sản phẩm". Đây là điều đặc biệt quan trọng trong các lĩnh vực như thương mại điện tử hoặc đánh giá sản phẩm, nơi người tiêu dùng thường thể hiện nhiều cảm xúc khác nhau đối với các đặc điểm khác nhau của sản phẩm.

ABSA thường bao gồm hai nhiệm vụ chính: (1) trích xuất các khía cạnh và (2) xác định cảm xúc tương ứng với từng khía cạnh (Pontiki & cộng sự, 2016). Ví dụ, trong câu "*Pin điện thoại rất tốt nhưng màn hình thì dễ xước*", mô hình ABSA cần nhận diện "pin" và "màn hình" là các khía cạnh, rồi gán cảm xúc tương ứng là "tốt" (tích cực) cho pin và "dễ xước" (tiêu cực) cho màn hình.

Trong lĩnh vực phân tích cảm xúc dựa trên khía cạnh dựa trên các mô hình học sâu, nhiều nghiên cứu gần đây đã tập trung vào phát triển phương pháp và bộ dữ liệu cho lĩnh vực cụ thể nhằm tối ưu hóa độ chính xác của việc nhận diện khía cạnh và

phân loại cảm xúc, đặc biệt trong lĩnh vực thương mại điện tử. Trong đó, công trình của Q.-L. Tran & cộng sự (2022) đã xây dựng thành công bộ dữ liệu gồm 16,227 đánh giá tiếng Việt về son môi từ nền tảng thương mại điện tử Shopee, được chú thích chi tiết cho 7 khía cạnh (màu sắc, giá cả, độ bám, mùi hương, đóng gói, vận chuyển, kết cấu) và 3 mức độ cảm xúc (tích cực, trung lập, tiêu cực). Kết quả thực nghiệm cho thấy, phương pháp Single-Task Learning (STL) đạt hiệu suất vượt trội so với Multi-Task Learning (MTL), với F1-score lần lượt là 98.09% cho phát hiện khía cạnh và 91.01% đối với phân tích cảm xúc. Tuy nhiên, STL đòi hỏi xây dựng 8 mô hình riêng biệt với 1 mô hình cho phát hiện khía cạnh và 7 mô hình cho phân tích cảm xúc, dẫn đến chi phí tính toán cao. Về kiến trúc, mô hình kết hợp tuần tự lớp BiGRU (Bidirectional Gated Recurrent Unit) và Conv1D (1D Convolutional Layer) cho kết quả tốt nhất, nhờ khả năng nắm bắt thông tin chuỗi dài hạn (BiGRU) và trích xuất đặc trưng cục bộ (Conv1D). Hạn chế của nghiên cứu nằm ở sự mất cân bằng dữ liệu, đặc biệt với các đánh giá trung lập/tiêu cực chỉ chiếm tỷ lệ nhỏ trên Shopee. Để khắc phục, nghiên cứu đề xuất hướng phát triển ứng dụng mô hình Transformer và xây dựng hệ thống tự động thu thập - phân tích - trực quan hóa dữ liệu, nhằm hỗ trợ doanh nghiệp và người dùng đưa ra quyết định dựa trên phân tích đa chiều.

Nghiên cứu của (Ho & cộng sự, 2023) đã đề xuất một mô hình lai kết hợp giữa khai phá khía cạnh và phân loại bình luận để thực hiện phân tích cảm xúc dựa trên khía cạnh trong lĩnh vực thương mại điện tử di động tại Việt Nam. Dữ liệu gồm hơn 1,2 triệu đánh giá từ bốn ứng dụng phổ biến: Shopee, Lazada, Sendo và Tiki, được thu thập từ năm 2013 đến 2022. Nhóm tác giả tiến hành tiền xử lý tiếng Việt chi tiết (xử lý teencode, từ mượn, thiếu dấu, emoji...), xây dựng từ điển cảm xúc và sử dụng kỹ thuật POS tagging để gắn nhãn cảm xúc cho từng khía cạnh. Mô hình Bi-LSTM được sử dụng để phân loại cảm xúc và các thông số đánh giá độ chính xác bao gồm Accuracy, Precision, Recall và F1-score. Kết quả thử nghiệm cho thấy mô hình Bi-LSTM sau tiền xử lý đạt độ chính xác 92.01% và F1-score 94.61%, vượt trội hơn các mô hình học máy truyền thống. Tuy nhiên, nghiên cứu này cũng gặp một số thách thức đáng kể, bao gồm: độ phức tạp của tiếng Việt với các đặc điểm như dấu, từ mượn, viết tắt và từ lóng; sự xuất hiện đồng thời nhiều khía cạnh và cảm xúc trái chiều trong cùng một bình luận và khó khăn trong việc xây dựng bộ từ điển cảm xúc

cũng như ánh xạ chính xác giữa khía cạnh và cảm xúc. Bên cạnh việc sử dụng lớp Embedding trong quá trình huấn luyện thay vì sử dụng các mô hình huấn luyện sẵn có thể không biểu diễn tốt ngữ nghĩa cho các từ trong bình luận.

Một nghiên cứu gần đây của (Quí, 2024) đã sử dụng mô hình T5 (Text-to-Text Transfer Transformer) với phiên bản ViT5 dành cho tiếng Việt, nhằm giải quyết bài toán phân tích cảm xúc dựa trên khía cạnh (ABSA) trong ngôn ngữ tiếng Việt. Nghiên cứu này đề xuất chuyển bài toán Aspect-Based Sentiment Analysis (ABSA) thành tác vụ sinh văn bản, trong đó đầu vào là câu văn bản và đầu ra là các cặp "khía cạnh - cảm xúc" tương ứng. Mô hình ViT5-base đã được huấn luyện trước (pre-trained) được tinh chỉnh các tham số phù hợp với dữ liệu VLSP 2018. Kết quả thử nghiệm trên bộ dữ liệu VLSP 2018 cho thấy mô hình ViT5 đạt F1-scores 84.04% cho nhiệm vụ phát hiện khía cạnh (AD) và 72.29% cho phân tích cảm xúc theo khía cạnh (AP) trong lĩnh vực nhà hàng, cùng 62.68% (AD) và 66.34% (AP) trong lĩnh vực khách sạn, cho thấy sự cải thiện so với một số phương pháp khác. Tuy nhiên, tác giả cũng chỉ ra rằng hiệu suất của mô hình phụ thuộc đáng kể vào cách chuẩn hóa dữ liệu đầu ra, đồng thời đòi hỏi chi phí tính toán cao do độ phức tạp của kiến trúc Transformer. Điều này không chỉ khẳng định tiềm năng của T5 trong xử lý ngôn ngữ tiếng Việt mà còn mở ra hướng nghiên cứu mới về tối ưu hóa mô hình và ứng dụng các kỹ thuật Few-shot/Zero-shot với mô hình ngôn ngữ lớn để giải quyết các thách thức về hiệu quả tính toán.

Nghiên cứu của (Wankhade & cộng sự, 2024) đã thực hiện bài khảo sát toàn diện về các phương pháp phân tích cảm xúc dựa trên khía cạnh. Các mô hình học sâu được sử dụng như CNN, RNN, LSTM, Transformer và BERT được đánh giá hiệu quả thông qua nhiều bộ dữ liệu chuẩn với các tác vụ dựa trên các độ đo như Accuracy, F1-score và MAE. Kết quả tổng hợp từ nhiều mô hình cho thấy rằng các phương pháp dựa trên Transformer như BERT có độ chính xác cao nhất trong việc phân tích cảm xúc theo khía cạnh.

## CHƯƠNG 3. PHƯƠNG PHÁP NGHIÊN CỨU

### 3.1. Thu thập dữ liệu

Dữ liệu đầu vào cho nghiên cứu được thu thập từ website thương mại điện tử tiki.vn, một trong những nền tảng mua sắm trực tuyến phổ biến tại Việt Nam. Cụ thể, dữ liệu bao gồm các bình luận và đánh giá của người dùng liên quan đến các sản phẩm đã mua, thể hiện quan điểm, mức độ hài lòng cũng như cảm xúc của khách hàng về chất lượng sản phẩm và dịch vụ. Việc thu thập dữ liệu được thực hiện một cách tự động nhằm đảm bảo hiệu quả và độ bao phủ cao trong việc khai thác thông tin.

Để thực hiện thu thập dữ liệu, nhóm nghiên cứu đã xây dựng một công cụ crawler sử dụng ngôn ngữ lập trình Python kết hợp với hai thư viện là Selenium và BeautifulSoup. Trong đó, Selenium đóng vai trò như một trình điều khiển trình duyệt tự động, giúp mô phỏng quá trình người dùng truy cập và tương tác với giao diện website để tiếp cận đầy đủ các nội dung cần thiết. Sau khi tải xong nội dung trang web, thư viện BeautifulSoup được sử dụng để phân tích và trích xuất dữ liệu HTML, giúp xác định và tách lọc các thành phần chứa bình luận của người dùng một cách chính xác. Nhờ sự kết hợp linh hoạt giữa hai công cụ này, quá trình thu thập dữ liệu trở nên hiệu quả, có khả năng xử lý các tình huống như tải nội dung tự động.

Quá trình thu thập dữ liệu được khởi đầu bằng việc lựa chọn các danh mục sản phẩm cụ thể trên nền tảng thương mại điện tử tiki.vn nhằm tập trung khai thác bình luận trong một phạm vi chủ đề nhất định. Dựa trên các danh mục này, hệ thống sẽ tự động truy xuất và thu thập đường dẫn (URL) của từng sản phẩm, sau đó lưu trữ toàn bộ danh sách liên kết này dưới định dạng .csv để phục vụ cho các bước truy xuất dữ liệu tiếp theo. Từ danh sách đường dẫn đã được tổng hợp, thư viện Selenium trong Python được sử dụng như một công cụ điều khiển trình duyệt tự động, giúp truy cập lần lượt vào từng trang sản phẩm. Selenium không chỉ mô phỏng thao tác duyệt web của người dùng mà còn hỗ trợ tương tác với các thành phần động nhằm hiển thị đầy đủ phần nội dung bình luận như mô phỏng thao tác lăn chuột, nhấn chuột. Bên cạnh đó, tính năng lọc bình luận của website được tận dụng để lọc các bình luận theo mức độ đánh giá để thu thập dữ liệu theo từng nhóm điểm số, giúp tăng cường tính đa dạng và cân bằng trong tập dữ liệu thu thập được. Sau khi nội dung HTML của các trang

sản phẩm được lấy về, thư viện BeautifulSoup được tích hợp để phân tích cấu trúc trang và trích xuất chính xác các đoạn văn bản chứa bình luận của người dùng. Các dữ liệu này sau đó được chuyển đổi và lưu trữ dưới định dạng Excel (.xlsx), để chuẩn bị cho bước tiền xử lý.

### 3.2. Tiền xử lý dữ liệu

Sau khi hoàn tất quá trình thu thập, các bình luận người dùng sẽ được đưa vào giai đoạn tiền xử lý dữ liệu nhằm chuẩn hóa văn bản. Đầu tiên, toàn bộ nội dung bình luận sẽ được chuyển đổi về chữ thường nhằm giảm thiểu sự phân biệt không cần thiết giữa các từ có cùng ngữ nghĩa nhưng khác hình thức viết. Sau đó, hệ thống tiến hành loại bỏ các thành phần không mang ý nghĩa ngữ nghĩa như các ký tự đặc biệt như !, @, #, các thẻ HTML còn sót lại do ảnh hưởng từ cấu trúc website, cũng như các biểu tượng cảm xúc (emoji) thường xuất hiện trong bình luận trực tuyến nhưng gây nhiễu cho quá trình phân tích ngữ nghĩa.

Tiếp theo, mỗi bình luận sẽ được tách thành các đơn vị từ vựng riêng biệt thông qua kỹ thuật tách từ tiếng Việt. Trong nghiên cứu này, thư viện *underthesea* được sử dụng để thực hiện quá trình tách từ để đảm bảo đầu ra là chuỗi các từ có thể xử lý tiếp bởi các mô hình học máy hoặc học sâu.

Sau khi dữ liệu đã được làm sạch và chuẩn hóa, các bình luận sẽ được tiến hành gán nhãn cảm xúc theo từng khía cạnh liên quan. Trong quá trình phân tích, các khía cạnh liên quan đến bình luận không nhất thiết phải được trích xuất trực tiếp từ các từ hoặc cụm từ cụ thể xuất hiện trong văn bản. Trên thực tế, trong nhiều trường hợp, khía cạnh không được đề cập một cách tường minh mà lại được thể hiện một cách gián tiếp thông qua ngữ nghĩa của toàn bộ câu. Trong phạm vi nghiên cứu này, thay vì trích xuất tự động các khía cạnh từ nội dung bình luận, một tập hợp các khía cạnh phổ biến và có tính đại diện cao trong lĩnh vực thương mại điện tử đã được xác định trước. Các khía cạnh này phản ánh những yếu tố mà người tiêu dùng thường xuyên quan tâm khi đánh giá sản phẩm bao gồm: chất lượng, giao hàng, hình thức, đóng gói, giá tiền, đánh giá chung. Bên cạnh đó, các nội dung bình luận đề cập đến những yếu tố khác không thuộc các nhóm chính nêu trên sẽ được phân loại vào một nhóm tổng hợp có tên là “khía cạnh khác”. Việc xác định trước tập hợp khía cạnh không chỉ giúp đơn giản hóa quá trình phân tích mà còn tạo điều kiện thuận lợi cho việc gán

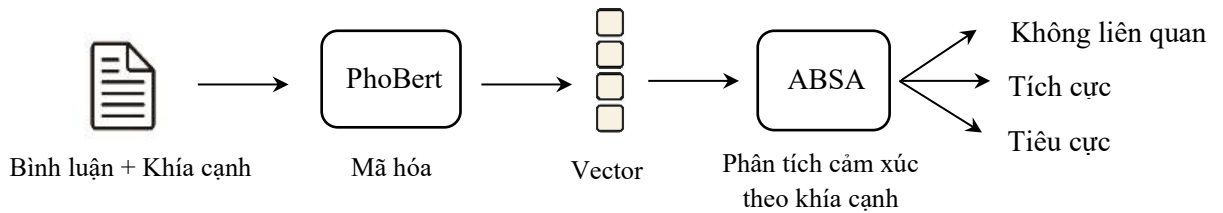
nhân và huấn luyện mô hình phân tích cảm xúc dựa trên khía cạnh. Cách tiếp cận này đặc biệt phù hợp trong dữ liệu bình luận tiếng Việt, nơi mà việc trích xuất khía cạnh ngầm định từ ngữ cảnh thường gặp nhiều khó khăn do đặc thù ngôn ngữ và cấu trúc diễn đạt linh hoạt.

Sau khi xác định tập hợp khía cạnh cụ thể, quá trình gán nhãn cảm xúc theo từng khía cạnh được thực hiện với sự hỗ trợ của mô hình ngôn ngữ ChatGPT thông qua API. Mỗi bình luận được cung cấp cho mô hình dưới dạng prompt có cấu trúc rõ ràng với nội dung như sau:

"Hãy phân tích các bình luận dưới đây và xác định các khía cạnh sau: 'chất lượng', 'giao hàng', 'hình thức', 'đóng gói', 'giá tiền', 'đánh giá chung', 'khía cạnh khác'. Với mỗi khía cạnh, trả về giá trị: 1 nếu tích cực, -1 nếu tiêu cực và loại bỏ khía cạnh khỏi kết quả trả về nếu không liên quan. Mỗi bình luận trả về một kết quả, kết quả trả về dưới dạng json, không có nội dung gì thêm"

Dữ liệu phản hồi từ API của ChatGPT, sau khi được xử lý, sẽ được tự động chuyển đổi và lưu trữ dưới dạng bảng trong một tập tin Excel nhằm phục vụ cho các bước xử lý và huấn luyện mô hình phân tích cảm xúc.

### 3.3. Xây dựng mô hình phân tích cảm xúc theo khía cạnh



Hình 3.1. Mô hình phân tích cảm xúc theo khía cạnh

Hình 4.1 minh họa quy trình tổng thể của hệ thống phân tích cảm xúc dựa trên khía cạnh sử dụng mô hình PhoBERT để chuyển đổi văn bản thành vector. Trong bước đầu tiên, mỗi bình luận được kết hợp với lần lượt từng khía cạnh đã xác định trước để tạo thành các cặp đầu vào dưới dạng [bình luận, khía cạnh]. Cách kết hợp này giúp mô hình có thể đánh giá mức độ liên quan và thái độ cảm xúc của người dùng đối với từng khía cạnh cụ thể, ngay cả trong các bình luận đề cập đến nhiều khía cạnh cùng lúc.

Sau khi chuẩn hóa, các cặp [bình luận, khía cạnh] được đưa vào mô hình PhoBERT để tạo ra các vector đặc trưng cho văn bản. PhoBERT có khả năng mã hóa

chuỗi văn bản đầu vào thành các vector đặc trưng phản ánh mối quan hệ ngữ nghĩa giữa nội dung bình luận và khía cạnh được xét. Khác với các phương pháp biểu diễn truyền thống, PhoBERT cho phép nắm bắt thông tin ngữ cảnh, từ đó tăng cường khả năng phân biệt các xu hướng cảm xúc liên quan đến từng khía cạnh trong các ngữ cảnh phức tạp. Để tối ưu hóa hiệu quả cho bài toán phân tích cảm xúc theo khía cạnh, mô hình PhoBERT được tinh chỉnh lại thông qua quá trình huấn luyện trên tập dữ liệu đã được gán nhãn. Việc tinh chỉnh giúp điều chỉnh các trọng số bên trong mô hình sao cho phù hợp hơn với dữ liệu bình luận thực tế của người dùng nhằm giúp mô hình đạt được độ chính xác cao hơn trong việc nhận diện cảm xúc theo ngữ cảnh cụ thể và cải thiện đáng kể hiệu suất phân tích cảm xúc ở cấp độ khía cạnh.

Tiếp theo, các vector ngữ nghĩa có được từ mô hình PhoBERT sẽ được đưa vào tầng phân lớp (classification layer) để thực hiện nhiệm vụ dự đoán cảm xúc của bình luận đối với từng khía cạnh cụ thể. Tầng phân lớp này thường bao gồm một hoặc nhiều lớp tuyến tính trong mạng nơ-ron truyền thẳng, có chức năng tính toán xác suất trên các nhãn cảm xúc cụ thể, từ đó đưa ra dự đoán cuối cùng cho từng cặp bình luận và khía cạnh. Mỗi cặp [bình luận, khía cạnh] sẽ được mô hình phân loại vào một trong ba nhãn cảm xúc tương ứng gồm các nhãn:

- **Không liên quan:** bình luận không đề cập đến khía cạnh được xét hoặc không chứa cảm xúc có liên quan.
- **Tích cực:** bình luận đề cập đến khía cạnh và thể hiện cảm xúc tích cực
- **Tiêu cực:** bình luận đề cập đến khía cạnh và thể hiện cảm xúc tiêu cực

Để đánh giá mô hình một cách chính xác và đảm bảo dữ liệu được phân bổ đều giữa các tập huấn luyện và kiểm tra, nghiên cứu sử dụng phương pháp kiểm định chéo k-fold stratified. Phương pháp này chia dữ liệu thành k phần sao cho mỗi phần đều có tỷ lệ tương đối đồng đều về các khía cạnh và cảm xúc. Nhờ đó, mô hình có thể học và kiểm tra trên những tập dữ liệu đại diện, hạn chế tình trạng mất cân bằng dữ liệu giữa các nhãn.

Sau khi hoàn thiện mô hình phân tích cảm xúc theo khía cạnh, nghiên cứu sử dụng các chỉ số đánh giá phổ biến trong bài toán phân loại để đo lường hiệu quả của mô hình, bao gồm: Accuracy, Precision, Recall và F1-score. Các độ đo này giúp đánh



giá toàn diện hiệu suất phân loại của mô hình trên cả phương diện độ chính xác lẫn khả năng nhận diện đầy đủ các nhãn cảm xúc.

Cách tiếp cận được đề xuất trong nghiên cứu cho phép thực hiện đồng thời hai nhiệm vụ: xác định mối liên hệ giữa bình luận và khía cạnh, đồng thời xác định cảm xúc tương ứng trong một kiến trúc mô hình thống nhất. Vì vậy, mô hình có thể khai thác được thông tin ngữ cảnh và học được mối quan hệ chặt chẽ giữa nội dung bình luận và các khía cạnh trong một không gian biểu diễn thống nhất.

Cách tiếp cận này không chỉ góp phần giảm thiểu chi phí triển khai của mô hình mà còn mang lại khả năng tổng quát hóa tốt hơn, đặc biệt trong những trường hợp bình luận đề cập đồng thời nhiều khía cạnh với các cảm xúc khác nhau. Ngoài ra, việc tích hợp toàn bộ quy trình phân tích vào một mô hình duy nhất cũng tạo điều kiện thuận lợi cho việc mở rộng hệ thống, cập nhật dữ liệu mới và triển khai trong các ứng dụng theo dõi phản hồi người dùng theo thời gian thực trong môi trường thương mại điện tử.

## CHƯƠNG 4. KẾT QUẢ THỰC NGHIỆM

### 4.1. Môi trường thực nghiệm

Toàn bộ quá trình thực nghiệm được triển khai trên nền tảng Google Colaboratory (Colab), môi trường lập trình dựa trên nền tảng điện toán đám mây do Google cung cấp, cho phép người dùng thực hiện các tác vụ tính toán chuyên sâu trực tiếp từ trình duyệt mà không cần cài đặt phần mềm cục bộ. Google Colab hỗ trợ tích hợp sẵn các thư viện phổ biến trong lĩnh vực học máy và xử lý ngôn ngữ tự nhiên như PyTorch, TensorFlow, Transformers và các công cụ phân tích dữ liệu như Pandas hoặc NumPy.

Một ưu điểm nổi bật của Google Colab là khả năng truy cập miễn phí vào phần cứng tăng tốc hiệu năng cao, đặc biệt là GPU và TPU, giúp rút ngắn thời gian huấn luyện và thử nghiệm mô hình. Trong quá trình thực nghiệm, Google Colab hỗ trợ cấu hình phần cứng gồm: GPU NVIDIA T4 với dung lượng bộ nhớ GPU khoảng 15 GB, CPU 2 vCPU (Intel Xeon) và RAM xấp xỉ 12.7 GB.

Việc sử dụng Google Colab không chỉ giúp tối ưu hóa tài nguyên phần cứng mà còn tạo điều kiện thuận lợi cho việc phát triển, kiểm thử và lặp lại mô hình trong các chu trình nghiên cứu, đặc biệt hữu ích đối với các mô hình yêu cầu tính toán lớn như các mô hình ngôn ngữ dựa trên Transformer. Ngoài ra, khả năng tích hợp với Google Drive cũng hỗ trợ lưu trữ và quản lý dữ liệu huấn luyện hiệu quả trong suốt quá trình thực nghiệm.

### 4.2. Dữ liệu thực nghiệm

Bộ dữ liệu thực nghiệm được thu thập từ website thương mại điện tử tiki.vn gồm các bình luận đánh giá sản phẩm do người dùng đăng tải trong quá trình mua sắm liên quan đến chủ đề đồ dùng nhà bếp. Sau khi thu thập, dữ liệu được xử lý sơ bộ thông qua các bước tiền xử lý như loại bỏ ký tự đặc biệt, biểu tượng cảm xúc, các mã HTML và thực hiện tách từ nhằm chuẩn hóa dữ liệu cho quá trình xây dựng mô hình phân tích cảm xúc.

Tập dữ liệu sau khi làm sạch bao gồm 9.963 mẫu bình luận, trong đó mỗi bình luận được kết hợp với một khía cạnh cụ thể đã được xác định trước để hình thành cặp [bình luận, khía cạnh]. Việc gán nhãn cảm xúc cho các cặp này được thực hiện với sự hỗ trợ của chatGPT API, theo một cấu trúc prompt được thiết kế có hệ thống. Mỗi

cặp được phân loại vào một trong ba nhãn cảm xúc: không liên quan, tích cực và tiêu cực.

Các mẫu được gán nhãn tích cực và tiêu cực là những phản hồi có giá trị đặc biệt trong việc phân tích mức độ hài lòng của người tiêu dùng theo từng khía cạnh cụ thể, từ đó cung cấp thông tin quan trọng phục vụ cho bài toán phân tích cảm xúc theo khía cạnh. Cụ thể, số lượng bình luận thuộc hai nhãn này được phân bổ theo từng khía cạnh như sau:

Bảng 4.1. Số lượng bình luận theo từng khía cạnh

Khía cạnh	Số lượng mẫu	
	Tích cực	Tiêu cực
CHẤT LƯỢNG	3659	3106
GIAO HÀNG	2241	1137
HÌNH THỨC	1989	471
ĐÓNG GÓI	1302	431
GIÁ TIỀN	1076	254
ĐÁNH GIÁ CHUNG	2186	1065
KHÓA CẠNH KHÁC	607	2194

Dữ liệu bình luận sau khi được tiền xử lý sẽ đưa về dạng phù hợp cho quá trình xây dựng mô hình phân tích cảm xúc như hình 5.1

comment	aspect	label	input_text
nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao	CHẤT LƯỢNG	1	nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao [SEP] CHẤT LƯỢNG
nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao	GIAO HÀNG	0	nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao [SEP] GIAO HÀNG
nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao	HÌNH THỨC	0	nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao [SEP] HÌNH THỨC
nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao	ĐÓNG GÓI	0	nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao [SEP] ĐÓNG GÓI
nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao	GIÁ TIỀN	0	nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao [SEP] GIÁ TIỀN
nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao	ĐÁNH GIÁ CHUNG	0	nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao [SEP] ĐÁNH GIÁ CHUNG
nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao	KHÓA CẠNH KHÁC	0	nồi bị gỉ sét khi chưa sử dụng sản phẩm chính hãng sao [SEP] KHÓA CẠNH KHÁC
giao thiếu quà tặng , mình liên hệ sdr nào để khiếu nại ?	CHẤT LƯỢNG	0	giao thiếu quà tặng , mình liên hệ sdr nào để khiếu nại ? [SEP] CHẤT LƯỢNG
giao thiếu quà tặng , mình liên hệ sdr nào để khiếu nại ?	GIAO HÀNG	1	giao thiếu quà tặng , mình liên hệ sdr nào để khiếu nại ? [SEP] GIAO HÀNG
giao thiếu quà tặng , mình liên hệ sdr nào để khiếu nại ?	HÌNH THỨC	0	giao thiếu quà tặng , mình liên hệ sdr nào để khiếu nại ? [SEP] HÌNH THỨC

Hình 4.1. Dữ liệu bình luận sau khi được tiền xử lý

### 4.3. Kết quả thực nghiệm

Sau khi hoàn thiện quy trình xây dựng mô hình và chuẩn bị dữ liệu huấn luyện, bước tiếp theo là tiến hành thực nghiệm nhằm đánh giá hiệu quả của mô hình phân tích cảm xúc theo khía cạnh trên tập dữ liệu đã được gán nhãn. Các thực nghiệm được thiết kế nhằm kiểm tra khả năng phân lớp cảm xúc của mô hình đối với từng khía cạnh cụ thể trong các tình huống đánh giá thực tế.

Để đảm bảo đánh giá mô hình một cách toàn diện và khách quan, nghiên cứu sử dụng phương pháp kiểm định chéo k-fold stratified với  $k = 5$  trong quá trình phân chia dữ liệu huấn luyện và kiểm thử. Với thiết lập  $k = 5$ , toàn bộ tập dữ liệu được chia thành 5 phần; ở mỗi lần huấn luyện, 4 phần được sử dụng để huấn luyện mô hình và phần còn lại dùng để kiểm tra. Quá trình được lặp lại tổng cộng 5 lần, đảm bảo rằng mỗi mẫu dữ liệu đều được sử dụng trong vai trò kiểm thử đúng một lần. Kết quả đánh giá cuối cùng được tổng hợp bằng cách tính trung bình các độ đo

Việc sử dụng k-fold giúp hạn chế tình trạng quá khớp (overfitting) và phụ thuộc vào một tập kiểm thử cố định, đồng thời giúp đánh giá ổn định và tin cậy hơn về hiệu suất tổng thể của mô hình trên toàn bộ tập dữ liệu. Ngoài ra, việc giữ cân bằng các nhãn trong từng fold còn giúp mô hình học được đặc trưng của cả ba lớp một cách nhất quán, ngay cả trong điều kiện dữ liệu không đồng đều.

Kết quả thực nghiệm đánh giá chung về mô hình phân tích cảm xúc được trình bày chi tiết thông qua các độ đo đánh giá gồm: Accuracy, Precision, Recall và F1-score được trình bày trong bảng 5.2

Bảng 4.2. Kết quả đánh giá chung mô hình phân tích cảm xúc theo khía cạnh

Độ đo	Accuracy	Precision	Recall	F1-score
Nhãn lớp				
Không liên quan	$0.8973 \pm 0.0024$	$0.9258 \pm 0.0037$	$0.9296 \pm 0.0047$	$0.9277 \pm 0.0021$
Tích cực		$0.8606 \pm 0.0040$	$0.8741 \pm 0.0113$	$0.8673 \pm 0.0048$
Tiêu cực		$0.7942 \pm 0.0176$	$0.7575 \pm 0.0094$	$0.7752 \pm 0.0055$

Kết quả trình bày trong Bảng 5.2 cho thấy mô hình phân tích cảm xúc theo khía cạnh đạt hiệu suất tổng thể khá cao, với độ chính xác trung bình là 0.8973, phản ánh khả năng phân lớp ổn định trên tập dữ liệu thực nghiệm. Mô hình thể hiện hiệu suất cao ở lớp "Không liên quan" với F1-score đạt 0.9277, cho thấy khả năng xác định các khía cạnh liên quan và không liên quan đến bình luận. Đối với lớp "Tích

cực", mô hình cũng đạt kết quả tốt với F1-score là 0.8673, cho thấy khả năng học tốt các biểu hiện cảm xúc tích cực trong ngôn ngữ tự nhiên tiếng Việt. Mặc dù lớp "Tiêu cực" có độ đo thấp hơn so với hai lớp còn lại, mô hình vẫn đạt F1-score là 0.7752. Đồng thời, mức độ dao động ở các độ đo vẫn được duy trì ở mức thấp, phản ánh tính ổn định và khả năng tổng quát hóa tốt của mô hình qua nhiều lần huấn luyện với dữ liệu phân chia khác nhau.

Để thấy rõ hiệu suất đạt được của mô hình trên từng khía cạnh cụ thể, nghiên cứu cũng tiến hành đánh giá từng khía cạnh cụ thể với các độ đo được trình bày trong bảng 5.3.

Bảng 4.3. Kết quả đánh giá theo từng khía cạnh cụ thể

Khía cạnh	Nhãn lớp	Accuracy	Precision	Recall	F1-score
CHẤT LƯỢNG	Không liên quan	$0.8837 \pm 0.0068$	$0.8111 \pm 0.0198$	$0.8197 \pm 0.0142$	$0.8152 \pm 0.0084$
	Tích cực		$0.9150 \pm 0.0082$	$0.9235 \pm 0.0124$	$0.9192 \pm 0.0078$
	Tiêu cực		$0.9126 \pm 0.0115$	$0.8934 \pm 0.0136$	$0.9029 \pm 0.0098$
GIAO HÀNG	Không liên quan	$0.9472 \pm 0.0033$	$0.9624 \pm 0.0035$	$0.9634 \pm 0.0046$	$0.9629 \pm 0.0028$
	Tích cực		$0.9582 \pm 0.0058$	$0.9607 \pm 0.0099$	$0.9594 \pm 0.0031$
	Tiêu cực		$0.8377 \pm 0.0227$	$0.8276 \pm 0.0174$	$0.8324 \pm 0.0141$
HÌNH THỨC	Không liên quan	$0.8904 \pm 0.0080$	$0.9336 \pm 0.0057$	$0.9249 \pm 0.0085$	$0.9292 \pm 0.0054$
	Tích cực		$0.7974 \pm 0.0126$	$0.8527 \pm 0.0170$	$0.8241 \pm 0.0142$
	Tiêu cực		$0.5973 \pm 0.0771$	$0.5032 \pm 0.0148$	$0.5436 \pm 0.0241$
ĐÓNG GÓI	Không liên quan	$0.9625 \pm 0.0013$	$0.9815 \pm 0.0021$	$0.9749 \pm 0.0015$	$0.9782 \pm 0.0005$
	Tích cực		$0.9346 \pm 0.0173$	$0.9616 \pm 0.0105$	$0.9478 \pm 0.0105$
	Tiêu cực		$0.7029 \pm 0.0447$	$0.7286 \pm 0.0366$	$0.7147 \pm 0.0305$
GIÁ TIỀN	Không liên quan	$0.9784 \pm 0.0069$	$0.9918 \pm 0.0030$	$0.9866 \pm 0.0050$	$0.9892 \pm 0.0039$
	Tích cực		$0.9253 \pm 0.0271$	$0.9526 \pm 0.0246$	$0.9387 \pm 0.0252$
	Tiêu cực		$0.7727 \pm 0.0625$	$0.8108 \pm 0.0575$	$0.7908 \pm 0.0549$

ĐÁNH GIÁ CHUNG	Không liên quan	$0.7843 \pm 0.0078$	$0.8353 \pm 0.0077$	$0.8449 \pm 0.0180$	$0.8399 \pm 0.0074$
	Tích cực		$0.7193 \pm 0.0266$	$0.7356 \pm 0.0378$	$0.7266 \pm 0.0202$
	Tiêu cực		$0.5837 \pm 0.0166$	$0.5136 \pm 0.0520$	$0.5457 \pm 0.0357$
KHÍA CẠNH KHÁC	Không liên quan	$0.8298 \pm 0.0075$	$0.8696 \pm 0.0101$	$0.9012 \pm 0.0121$	$0.8850 \pm 0.0065$
	Tích cực		$0.5928 \pm 0.0116$	$0.4993 \pm 0.0473$	$0.5412 \pm 0.0300$
	Tiêu cực		$0.7541 \pm 0.0246$	$0.7010 \pm 0.0345$	$0.7258 \pm 0.0149$

Kết quả đánh giá theo từng khía cạnh cụ thể trong Bảng 5.3 cho thấy mô hình đề xuất đạt hiệu suất tốt và ổn định đối với hầu hết các khía cạnh được khảo sát. Đặc biệt, ở các khía cạnh như giao hàng, giá tiền và đóng gói, mô hình cho thấy hiệu quả cao với F1-score của lớp “Tích cực” đạt trên 0.94.

Ở những khía cạnh như chất lượng và hình thức, mặc dù độ chính xác tổng thể thấp hơn đôi chút, mô hình vẫn có khả năng nhận diện khá tốt các phản hồi tích cực từ người dùng. Tuy nhiên, hiệu suất ở lớp “Tiêu cực” tại các khía cạnh này vẫn chưa được cao nhưng kết quả vẫn có thể chấp nhận được. Đối với hai khía cạnh đánh giá chung và khía cạnh khác, kết quả đạt được ở mức tương đối vì các bình luận thuộc hai nhóm này thường có nội dung tổng hợp, không đi sâu vào một đặc điểm cụ thể nào hoặc đề cập đến các yếu tố ít phổ biến, dẫn đến việc biểu diễn đặc trưng không rõ ràng. Bên cạnh đó, số lượng mẫu không đồng đều giữa các nhãn hoặc giữa các khía cạnh cũng có thể ảnh hưởng đến hiệu suất huấn luyện và dự đoán.

Tổng quan kết quả thực nghiệm cho thấy mô hình phân tích cảm xúc theo khía cạnh được đề xuất đạt hiệu suất tổng thể khá tốt. Các khía cạnh phổ biến như giao hàng, giá tiền và đóng gói đạt hiệu quả nhận diện cao, phản ánh rõ rệt sự tương thích giữa cấu trúc ngôn ngữ người dùng và biểu diễn của mô hình. Mặc dù hiệu suất ở một số khía cạnh như đánh giá chung hay khía cạnh khác còn hạn chế do sự phân bố không đều của dữ liệu và đặc điểm ngữ nghĩa phức tạp trong các bình luận thuộc các khía cạnh này. Dù vậy, mô hình vẫn cho thấy khả năng tổng quát hóa tốt và tiềm năng triển khai trong các hệ thống giám sát trải nghiệm khách hàng theo thời gian thực, hỗ trợ doanh nghiệp đưa ra quyết định nhanh chóng và hiệu quả hơn.

## CHƯƠNG 5. KẾT LUẬN

### 5.1. Tổng kết

Đề tài đã xây dựng và triển khai thành công một mô hình phân tích cảm xúc theo khía cạnh sử dụng mô hình học sâu, với mục tiêu khai thác thông tin cảm xúc từ các bình luận người dùng trong bối cảnh thương mại điện tử tại Việt Nam. Mô hình được thiết kế theo hướng tiếp cận tích hợp, kết hợp hiệu quả giữa các bước xử lý ngôn ngữ tự nhiên, gán nhãn tự động bằng mô hình ngôn ngữ ChatGPT và huấn luyện mô hình học sâu dựa trên kiến trúc PhoBERT được tinh chỉnh. Cách tiếp cận này cho phép thực hiện đồng thời hai nhiệm vụ: xác định mối liên hệ giữa bình luận và khía cạnh, cũng như phân loại cảm xúc tương ứng, từ đó tối ưu hóa quy trình xử lý và giảm thiểu sự phụ thuộc vào các thành phần trích xuất thủ công.

Kết quả thực nghiệm trên tập dữ liệu gồm hơn 9.000 mẫu bình luận đánh giá sản phẩm cho thấy mô hình đạt hiệu suất phân lớp khả quan, đặc biệt đối với các khía cạnh phổ biến như giao hàng, giá tiền và đóng gói. Độ chính xác và F1-score cao ở những khía cạnh này phản ánh khả năng học của mô hình khi xử lý ngữ liệu tiếng Việt trong bối cảnh thực tế. Những kết quả tích cực này cho thấy tiềm năng ứng dụng của mô hình trong các hệ thống giám sát tự động cảm xúc khách hàng, góp phần hỗ trợ doanh nghiệp theo dõi trải nghiệm người dùng theo thời gian thực, phát hiện kịp thời các vấn đề phát sinh và đưa ra quyết định chiến lược hiệu quả hơn.

### 5.2. Đóng góp của đề tài

Đề tài mang lại nhiều đóng góp đáng kể, cả về mặt phương pháp luận lẫn ứng dụng thực tiễn, đặc biệt trong lĩnh vực phân tích cảm xúc tiếng Việt.

Thứ nhất, về đóng góp phương pháp luận, cách tiếp cận tích hợp giữa xử lý ngôn ngữ tự nhiên truyền thống, gán nhãn tự động bằng mô hình ngôn ngữ tiên tiến như ChatGPT và huấn luyện mô hình học sâu dựa trên kiến trúc PhoBERT được tinh chỉnh là một điểm đột phá. Trong bối cảnh các tài nguyên ngôn ngữ cho tiếng Việt còn hạn chế so với tiếng Anh, việc sử dụng ChatGPT để tự động hóa quy trình gán nhãn là một giải pháp sáng tạo, giúp giảm thiểu đáng kể chi phí và thời gian cho việc chuẩn bị dữ liệu. Điều này không chỉ đẩy nhanh quá trình nghiên cứu mà còn mở ra hướng đi mới cho việc xây dựng các bộ dữ liệu chất lượng cao trong tương lai. Hơn nữa, việc tinh chỉnh PhoBERT cho nhiệm vụ cụ thể của phân tích cảm xúc theo khía cạnh đã tối ưu hóa khả năng hiểu ngữ nghĩa và ngữ cảnh phức tạp của bình luận tiếng

Việt, điều mà các mô hình tổng quát khác có thể gặp khó khăn. Sự kết hợp này tạo nên một khung kiến trúc mạnh mẽ, linh hoạt, có thể thích nghi với nhiều loại dữ liệu văn bản khác nhau, không chỉ giới hạn trong lĩnh vực thương mại điện tử.

Thứ hai, về hiệu quả xử lý và tự động hóa, mô hình đã tối ưu hóa toàn bộ quy trình phân tích cảm xúc. Bằng cách thực hiện đồng thời hai nhiệm vụ – nhận diện khía cạnh và phân loại cảm xúc – mô hình đã loại bỏ nhu cầu về các bước trích xuất đặc trưng thủ công riêng biệt. Điều này không chỉ nâng cao tốc độ xử lý mà còn giảm thiểu lỗi do yếu tố con người, tăng tính nhất quán và độ tin cậy của kết quả. Khả năng tự động hóa cao của mô hình là một bước tiến quan trọng, giúp các doanh nghiệp dễ dàng tích hợp vào hệ thống hiện có, giảm chi phí vận hành và tăng cường khả năng phản ứng nhanh với biến động thị trường.

Thứ ba, về tiềm năng ứng dụng thực tiễn và tác động kinh doanh, đề tài đã chứng minh khả năng của mô hình trong việc hỗ trợ doanh nghiệp giám sát trải nghiệm khách hàng theo thời gian thực. Trong môi trường kinh doanh cạnh tranh khốc liệt như hiện nay, việc nắm bắt nhanh chóng và chính xác cảm xúc của khách hàng là yếu tố then chốt. Mô hình có thể tự động quét và phân tích hàng ngàn bình luận mỗi ngày, giúp doanh nghiệp phát hiện kịp thời các vấn đề phát sinh, từ đó đưa ra các biện pháp khắc phục nhanh chóng, trước khi các vấn đề này trở nên nghiêm trọng và ảnh hưởng đến uy tín thương hiệu. Ngoài ra, việc phân tích cảm xúc theo từng khía cạnh cụ thể cung cấp cái nhìn chi tiết hơn, giúp doanh nghiệp hiểu rõ điểm mạnh cần phát huy và điểm yếu cần cải thiện của từng sản phẩm hoặc dịch vụ. Điều này hỗ trợ đắc lực cho các quyết định chiến lược về phát triển sản phẩm, chiến dịch marketing, chính sách giá, và quản lý chuỗi cung ứng, từ đó nâng cao sự hài lòng của khách hàng và tối đa hóa lợi nhuận.

### **5.3. Hạn chế**

Mặc dù đã đạt được những kết quả khả quan và mang lại nhiều đóng góp ý nghĩa, đề tài vẫn còn tồn tại một số hạn chế nhất định, chỉ ra các hướng cải thiện và phát triển cho các nghiên cứu tiếp theo.

Hạn chế đầu tiên và rõ rệt nhất là hiệu suất phân lớp còn chưa đồng đều và chưa cao đối với các khía cạnh ít phổ biến. Điều này chủ yếu xuất phát từ sự mất cân bằng nghiêm trọng về số lượng mẫu dữ liệu giữa các lớp khía cạnh. Khi dữ liệu của



một khía cạnh quá ít, mô hình không đủ thông tin để xây dựng một "biểu diễn" mạnh mẽ cho khía cạnh đó, dẫn đến hiệu suất phân loại thấp hơn đáng kể. Đặc biệt, đặc điểm biểu đạt phức tạp của ngôn ngữ người dùng trong các bình luận cũng góp phần vào thách thức này. Những khía cạnh ít phổ biến thường được diễn đạt một cách ẩn ý, sử dụng từ ngữ đa nghĩa, hoặc lồng ghép trong các câu văn dài, khó phân tách, khiến mô hình gặp khó khăn hơn trong việc liên kết chính xác bình luận với khía cạnh cụ thể.

Hạn chế thứ hai là khả năng của mô hình trong việc phân biệt giữa các trạng thái cảm xúc không rõ ràng hoặc mang tính mơ hồ còn gặp nhiều thách thức. Ngôn ngữ con người rất phong phú và phức tạp, đôi khi một bình luận không chỉ thể hiện một cảm xúc duy nhất mà có thể chứa đựng nhiều sắc thái đan xen (ví dụ: vừa hài lòng về một khía cạnh nhưng không hài lòng về khía cạnh khác), hoặc thậm chí là cảm xúc trung tính nhưng lại ngụ ý sự không hài lòng. Các bình luận mang tính chất mỉa mai, châm biếm, hay các câu hỏi không trực tiếp thể hiện cảm xúc cũng là những thách thức lớn. Hiện tại, mô hình vẫn còn gặp khó khăn trong việc giải quyết những tình huống này, dẫn đến việc phân loại cảm xúc có thể chưa đạt được độ chính xác tuyệt đối trong các trường hợp ngữ nghĩa phức tạp. Điều này đòi hỏi mô hình phải có khả năng hiểu sâu sắc hơn về ngữ cảnh, ý định của người nói, và thậm chí là các yếu tố văn hóa trong giao tiếp để đưa ra phán đoán chính xác hơn.

Ngoài ra, một hạn chế tiềm tàng khác có thể phát sinh từ sự phụ thuộc vào mô hình ngôn ngữ lớn như ChatGPT trong giai đoạn gán nhãn tự động. Mặc dù ChatGPT mang lại hiệu quả cao trong việc tạo nhãn, nhưng nó có thể vẫn tiềm ẩn những sai lệch (bias) nhất định trong quá trình gán nhãn hoặc gặp khó khăn với các thuật ngữ, tiếng lóng, hay ngữ cảnh đặc thù của tiếng Việt mà nó chưa được huấn luyện đủ sâu. Điều này có thể ảnh hưởng đến chất lượng của tập dữ liệu huấn luyện và gián tiếp ảnh hưởng đến hiệu suất tổng thể của mô hình học sâu.

#### **5.4. Hướng phát triển**

Trên cơ sở những kết quả đạt được, đề tài đề xuất một số hướng phát triển tiềm năng trong tương lai.

Thứ nhất, việc mở rộng quy mô và nâng cao tính đa dạng của tập dữ liệu huấn luyện là cần thiết, đặc biệt là tăng cường số lượng mẫu ở những khía cạnh có tần suất

xuất hiện thấp nhằm đảm bảo tính cân bằng dữ liệu, từ đó nâng cao hiệu suất tổng thể của mô hình.

Thứ hai, nghiên cứu có thể cải tiến bằng cách tích hợp cơ chế trích xuất khía cạnh tự động từ các bình luận thay vì phụ thuộc vào tập khía cạnh định sẵn. Phương pháp này không chỉ tăng tính linh hoạt của mô hình mà còn mở rộng khả năng ứng dụng trong nhiều bối cảnh khác nhau.

Cuối cùng, để đánh giá khả năng tổng quát hóa của mô hình, nghiên cứu có thể triển khai ứng dụng thử nghiệm trong nhiều lĩnh vực khác như giáo dục, du lịch, y tế và dịch vụ công. Việc mở rộng phạm vi ứng dụng không chỉ kiểm chứng tính hiệu quả của mô hình mà còn góp phần gia tăng giá trị thực tiễn của nghiên cứu.

Những hướng phát triển nêu trên không chỉ mở rộng giá trị học thuật của nghiên cứu mà còn gia tăng khả năng ứng dụng thực tiễn trong nhiều lĩnh vực khác nhau. Việc tiếp tục phát triển theo các hướng này sẽ góp phần hoàn thiện mô hình, đồng thời mang lại những đóng góp có ý nghĩa cho cả cộng đồng khoa học và các ngành công nghiệp liên quan.

## TÀI LIỆU THAM KHẢO

1. Breiman, L. (2001). Random Forests. *Machine learning*, 45(1), 5-32.
2. Chauhan, G. S., Agrawal, P., & Meena, Y. K. (2018). Aspect-based Sentiment Analysis of Students' Feedback to Improve Teaching–Learning Process. In *Information and Communication Technology for Intelligent Systems: Proceedings of ICTIS 2018, Volume 2* (pp. 259-266): Springer.
3. Cover, T., & Hart, P. (1967). Nearest Neighbor Pattern Classification. *IEEE transactions on information theory*, 13(1), 21-27.
4. Dang, C. N., Moreno-García, M. N., De la Prieta, F., et al. (2023). Sentiment Analysis for Vietnamese–Based Hybrid Deep Learning Models. *International Conference on Hybrid Artificial Intelligence Systems*.
5. Devlin, J., Chang, M.-W., Lee, K., et al. (2019). Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minnesota, USA.
6. Han, J., Pei, J., & Tong, H. (2012). *Data Mining: Concepts and Techniques*: Morgan Kaufmann.
7. Harris, Z. S. (1954). Distributional Structure. *Word*, 10(2-3), 146-162.
8. Ho, T., Bui, H. M., & Phung, T. K. (2023). A Hybrid Model for Aspect-Based Sentiment Analysis on Customer Feedback: Research on the Mobile Commerce Sector in Vietnam. *International Journal of Advances in Intelligent Informatics*, 9(2).
9. Hoang, M., Bihorac, O. A., & Rouces, J. (2019). Aspect-Based Sentiment Analysis Using Bert. *Proceedings of the 22nd Nordic Conference on Computational Linguistics*.
10. Jagtap, V., & Pawar, K. (2013). Analysis of Different Approaches to Sentence-level Sentiment Classification. *International Journal of Scientific Engineering and Technology*, 2(3), 164-170.
11. Joulin, A., Grave, E., Bojanowski, P., et al. (2016). Bag of Tricks for Efficient Text Classification. *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, Valencia, Spain.
12. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature*, 521(7553), 436-444.
13. Liu, B., & Zhang, L. (2012). A Survey of Opinion Mining and Sentiment Analysis. In *Mining Text Data* (pp. 415-463): Springer.
14. Liu, Y., Ott, M., Goyal, N., et al. (2019). Roberta: A Robustly Optimized BERT Pretraining Approach. *arXiv, abs/1907.11692*.
15. Lubis, Z., Sihombing, P., & Mawengkang, H. (2020). Optimization of K Value at the K-NN Algorithm in Clustering Using the Expectation Maximization Algorithm. *IOP Conference Series: Materials Science and Engineering*.
16. Mao, Y., Liu, Q., Zhang, Y., et al. (2024). Sentiment Analysis Methods, Applications, and Challenges: A Systematic Literature Review. *Journal of King Saud University-Computer*, 36(4), 102048.

17. Mikolov, T., Chen, K., Corrado, G., et al. (2013). Efficient Estimation of Word Representations in Vector Space. *1st International Conference on Learning Representations*, Arizona, USA.
18. Nguyen, A. T., Dao, M. H., & Nguyen, D. Q. (2020). A Pilot Study of Text-to-SQL Semantic Parsing for Vietnamese. *Findings of the Association for Computational Linguistics: EMNLP 2020*.
19. Nguyen, D. Q., & Nguyen, A. T. (2020). PhoBERT: Pre-trained Language Models for Vietnamese. *Findings of the Association for Computational Linguistics: EMNLP 2020*.
20. Nguyen, L. T., Van Nguyen, K., & Nguyen, N. L.-T. (2022). SMTCE: A Social Media Text Classification Evaluation Benchmark and BERTology Models for Vietnamese. *Proceedings of the 36th Pacific Asia Conference on Language, Information and Computation*, Manila, Philippines.
21. Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global Vectors for Word Representation. *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*.
22. Peters, M. E., Neumann, M., Iyyer, M., et al. (2018). Deep Contextualized Word Representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Louisiana, USA.
23. Pham Duy, A., & Le Thanh, H. (2023). A Question-Answering System for Vietnamese Public Administrative Services. *Proceedings of the 12th International symposium on information and communication technology*.
24. Pires, T., Schlinger, E., & Garrette, D. (2019). How Multilingual is Multilingual BERT? *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy.
25. Pontiki, M., Galanis, D., Papageorgiou, H., et al. (2016). SemEval-2016 Task 5: Aspect Based Sentiment Analysis. *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, San Diego, California.
26. Quí, Đ. B. (2024). Mô hình T5 cho bài toán trích xuất cảm xúc dựa trên khía cạnh trên dữ liệu tiếng Việt. *Tạp chí khoa học và công nghệ*, 24(4).
27. Quinlan, J. R. (1986). Induction of Decision Trees. *Machine learning*, 1(1), 81-106.
28. Radford, A., Narasimhan, K., Salimans, T., et al. (2018). Improving Language Understanding by Generative Pre-Training.
29. Sahu, S. K., Kumar, P., & Singh, A. P. (2018). Modified K-NN Algorithm for Classification Problems with Improved Accuracy. *International Journal of Information Technology*, 10(1), 65-70.
30. Salton, G., Wong, A., & Yang, C.-S. (1975). A Vector Space Model for Automatic Indexing. *Communications of the ACM*, 18(11), 613-620.
31. Sivakumar, M., & Reddy, U. S. (2017). Aspect Based Sentiment Analysis of Students Opinion Using Machine Learning Techniques. *2017 International Conference on Inventive Computing and Informatics (ICICI)*.

32. Tran, K. Q., Nguyen, A. T., Le, A. T.-H., et al. (2021). ViVQA: Vietnamese Visual Question Answering. *Proceedings of the 35th Pacific Asia Conference on Language, Information and Computation*, Shanghai, China.
33. Tran, Q.-L., Le, P. T. D., & Do, T.-H. (2022). Aspect-Based Sentiment Analysis for Vietnamese Reviews About Beauty Product on E-commerce Websites. *Proceedings of the 36th Pacific Asia conference on Language, Information and Computation*.
34. Van Huynh, T., Van Nguyen, K., & Nguyen, N. L.-T. (2022). ViNLI: A Vietnamese Corpus for Studies on Open-Domain Natural Language Inference. *Proceedings of the 29th International Conference on Computational Linguistics*.
35. Van Nguyen, K., Nguyen, V. D., Nguyen, P. X., et al. (2018). UIT-VSFC: Vietnamese Students' Feedback Corpus for Sentiment Analysis. *2018 10th International Conference on Knowledge and Systems Engineering (KSE)*.
36. Van Thin, D., Nguye, V. D., Van Nguyen, K., et al. (2018). Deep Learning for Aspect Detection on Vietnamese Reviews. *2018 5th NAFOSTED Conference on Information and Computer Science (NICS)*.
37. Wankhade, M., Kulkarni, C., & Rao, A. C. S. (2024). A Survey on Aspect Base Sentiment Analysis Methods and Challenges. *Applied Soft Computing*, 167, 112249.
38. Zhai, Z., Liu, B., Xu, H., et al. (2011). Clustering Product Features for Opinion Mining. *Proceedings of the fourth ACM International Conference on Web Search and Data Mining*.
39. Zhang, G. P. (2002). Neural Networks for Cassification: A survey. *IEEE Transactions on Systems, Man, Cybernetics, Part C*, 30(4), 451-462.

## PHỤ LỤC

## Phụ lục 1: Mã nguồn cho việc huấn luyện và đánh giá mô hình ABSA

1. Cài đặt và import thư viện
<pre> !pip install underthesea  import pandas as pd import numpy as np from underthesea import word_tokenize from transformers import (     AutoTokenizer,     AutoModelForSequenceClassification,     Trainer,     TrainingArguments,     EarlyStoppingCallback ) from datasets import Dataset from sklearn.model_selection import KFold from sklearn.metrics import classification_report import torch from torch.utils.data import DataLoader from tqdm import tqdm from sklearn.model_selection import StratifiedKFold import warnings warnings.filterwarnings('ignore') from google.colab import drive drive.mount('/content/drive') </pre>
2. Tải và tiền xử lý dữ liệu
<pre> # Load and preprocess data df = pd.read_excel("drive/My Drive/2025_ABSA/processed_result_full.xlsx", sheet_name="Sheet 1") aspects = ["CHẤT LƯỢNG", "GIAO HÀNG", "HÌNH THỨC", "ĐÓNG GÓI", "GIÁ TIỀN", "ĐÁNH GIÁ CHUNG", "KHÓA CẠNH KHÁC"]  # Tokenize comments df['comment'] = df['comment'].apply(lambda x: word_tokenize(x, format="text")) </pre>
3. Chuẩn bị dataset đầu vào cho mô hình
<pre> # Prepare dataset rows = [] for _, row in df.iterrows():     comment = row["comment"]     for aspect in aspects:         if pd.isna(row[aspect]):             label = 0 # Không đề cập         elif int(row[aspect]) == -1:             label = 1 # Tiêu cực </pre>

```

elif int(row[aspect]) == 1:
    label = 2 # Tích cực
else:
    continue
rows.append({
    "comment": comment,
    "aspect": aspect,
    "label": label
})
df_joint = pd.DataFrame(rows)
df_joint["input_text"] = df_joint.apply(lambda r: f"{r['comment']}"
[SEP] {r['aspect']}", axis=1)
df_joint["stratify_label"] = df_joint["aspect"] + "_" +
df_joint["label"].astype(str)
print(f"Tổng số mẫu: {len(df_joint)}")

```

#### 4. Thiết lập Tokenizer và K-fold

```

# Initialize tokenizer
tokenizer = AutoTokenizer.from_pretrained("vinai/phobert-base",
use_fast=False)
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")

# K-Fold Cross Validation setup
num_folds = 5
skf = StratifiedKFold(n_splits=num_folds, shuffle=True,
random_state=42)

# Store results for each fold
fold_results = []

```

#### 5. Vòng lặp huấn luyện – đánh giá mô hình với K-fold

```

for fold, (train_idx, val_idx) in enumerate(skf.split(df_joint,
df_joint["stratify_label"])):
    print(f"\n{'='*40}")
    print(f"Fold {fold + 1}/{num_folds}")
    print(f"{'='*40}")

    # Split data
    train_df = df_joint.iloc[train_idx]
    val_df = df_joint.iloc[val_idx]

    # Tokenize data
    train_encodings = tokenizer(
        train_df["input_text"].tolist(),
        truncation=True,
        padding=True,
        max_length=128
    )
    val_encodings = tokenizer(

```

```

        val_df["input_text"].tolist(),
        truncation=True,
        padding=True,
        max_length=128
    )

    # Create datasets
    train_dataset = Dataset.from_dict({
        **train_encodings,
        "labels": train_df["label"].tolist()
    })
    val_dataset = Dataset.from_dict({
        **val_encodings,
        "labels": val_df["label"].tolist()
    })

    # Initialize model for each fold
    model = AutoModelForSequenceClassification.from_pretrained(
        "vinai/phobert-base",
        num_labels=3
    )
    model.to(device)

    # Training arguments with early stopping
    training_args = TrainingArguments(
        output_dir=f"./joint_absa_model_fold_{fold}",
        per_device_train_batch_size=16,
        per_device_eval_batch_size=16,
        num_train_epochs=5, # Increased epochs for better fine-tuning
        save_strategy="epoch",
        logging_dir=f"./logs_joint_fold_{fold}",
        fp16=True,
        #load_best_model_at_end=True,
        #metric_for_best_model="eval_loss",
        greater_is_better=False,
        save_total_limit=2,
        report_to="none",
        learning_rate=2e-5, # Lower learning rate for fine-tuning
        weight_decay=0.01
    )

    # Custom compute_metrics function
    def compute_metrics(p):
        predictions, labels = p
        predictions = np.argmax(predictions, axis=1)

        # Calculate metrics for each class
        report = classification_report(

```



```

        labels,
        predictions,
        output_dict=True,
        target_names=["KHÔNG", "TIÊU CỰC", "TÍCH CỰC"],
        zero_division=0
    )

    return {
        "accuracy": report["accuracy"],
        "macro_avg_precision": report["macroavg"]["precision"],
        "macro_avg_recall": report["macroavg"]["recall"],
        "macro_avg_f1": report["macro avg"]["f1-score"],
        "weighted_avg_f1": report["weightedavg"]["f1score"]
    }

# Create Trainer with early stopping
trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=train_dataset,
    eval_dataset=val_dataset,
    compute_metrics=compute_metrics
)

# Train the model
trainer.train()

# Evaluate on validation set
val_results = trainer.predict(val_dataset)
y_pred = np.argmax(val_results.predictions, axis=1)
y_true = val_results.label_ids

# Store predictions for detailed analysis
val_df = val_df.copy()
val_df["y_true"] = y_true
val_df["y_pred"] = y_pred

# Print classification report for each aspect
print("\nValidation Results:")
for aspect in aspects:
    print(f"\n Aspect: {aspect}")
    df_subset = val_df[val_df["aspect"] == aspect]
    print(classification_report(
        df_subset["y_true"],
        df_subset["y_pred"],
        digits=4,
        target_names=["KHÔNG", "TIÊU CỰC", "TÍCH CỰC"],
        zero_division=0
    ))

```

```
    ))

    # Save fold results
    fold_results.append({
        "fold": fold + 1,
        "model": model,
        "val_df": val_df,
        "metrics": val_results.metrics
    })
```

## 6. Lưu kết quả

```
import pickle

# Lưu fold_results vào file
with open('drive/My Drive/2025_ABSA/fold_results.pkl', 'wb') as f:
    pickle.dump(fold_results, f)
```

## Phụ lục 2: Kết quả đánh giá mô hình chạy theo từng Fold

### Fold 1/5:

Validation Results:

Aspect: CHẤT LƯỢNG					Aspect: GIAO HÀNG				
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.8112	0.8215	0.8163	549	KHÔNG	0.9633	0.9640	0.9636	1306
TIÊU CỰC	0.9034	0.8875	0.8954	622	TIÊU CỰC	0.8213	0.8465	0.8337	228
TÍCH CỰC	0.9280	0.9331	0.9305	732	TÍCH CỰC	0.9659	0.9487	0.9572	448
accuracy			0.8860	1903	accuracy			0.9470	1982
macro avg	0.8809	0.8807	0.8807	1903	macro avg	0.9168	0.9197	0.9182	1982
weighted avg	0.8863	0.8860	0.8861	1903	weighted avg	0.9475	0.9470	0.9472	1982
Aspect: HÌNH THỨC					Aspect: ĐÓNG GỐI				
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.9320	0.9170	0.9244	1494	KHÔNG	0.9816	0.9750	0.9783	1642
TIÊU CỰC	0.5465	0.5000	0.5222	94	TIÊU CỰC	0.7711	0.7442	0.7574	86
TÍCH CỰC	0.7860	0.8492	0.8164	398	TÍCH CỰC	0.9200	0.9693	0.9440	261
accuracy			0.8837	1986	accuracy			0.9643	1989
macro avg	0.7548	0.7554	0.7544	1986	macro avg	0.8909	0.8962	0.8932	1989
weighted avg	0.8845	0.8837	0.8837	1986	weighted avg	0.9644	0.9643	0.9643	1989
Aspect: GIÁ TIỀN					Aspect: ĐÁNH GIÁ CHUNG				
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.9930	0.9912	0.9921	1714	KHÔNG	0.8447	0.8169	0.8305	1305
TIÊU CỰC	0.8571	0.8235	0.8400	51	TIÊU CỰC	0.5833	0.5258	0.5531	213
TÍCH CỰC	0.9591	0.9814	0.9701	215	TÍCH CỰC	0.6806	0.7803	0.7271	437
accuracy			0.9859	1980	accuracy			0.7770	1955
macro avg	0.9364	0.9321	0.9341	1980	macro avg	0.7029	0.7077	0.7036	1955
weighted avg	0.9858	0.9859	0.9858	1980	weighted avg	0.7795	0.7770	0.7772	1955
Aspect: KHẢ CẢNH KHÁC									
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.8604	0.9024	0.8809	1352					
TIÊU CỰC	0.7520	0.6560	0.7007	439					
TÍCH CỰC	0.5856	0.5372	0.5603	121					
accuracy			0.8227	1912					
macro avg	0.7326	0.6985	0.7140	1912					
weighted avg	0.8181	0.8227	0.8192	1912					

### Fold 2/5:

Validation Results:

Aspect: CHẤT LƯỢNG					Aspect: GIAO HÀNG				
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.8092	0.8342	0.8215	549	KHÔNG	0.9628	0.9709	0.9669	1307
TIÊU CỰC	0.9160	0.8953	0.9055	621	TIÊU CỰC	0.8756	0.8333	0.8539	228
TÍCH CỰC	0.9177	0.9139	0.9158	732	TÍCH CỰC	0.9577	0.9577	0.9577	449
accuracy			0.8849	1902	accuracy			0.9521	1984
macro avg	0.8810	0.8812	0.8810	1902	macro avg	0.9320	0.9206	0.9262	1984
weighted avg	0.8858	0.8849	0.8852	1902	weighted avg	0.9516	0.9521	0.9518	1984
Aspect: HÌNH THỨC					Aspect: ĐÓNG GỐI				
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.9378	0.9378	0.9378	1494	KHÔNG	0.9828	0.9750	0.9789	1642
TIÊU CỰC	0.6866	0.4894	0.5714	94	TIÊU CỰC	0.7011	0.7011	0.7011	87
TÍCH CỰC	0.8160	0.8715	0.8429	397	TÍCH CỰC	0.9124	0.9579	0.9346	261
accuracy			0.9033	1985	accuracy			0.9608	1990
macro avg	0.8135	0.7662	0.7840	1985	macro avg	0.8655	0.8780	0.8715	1990
weighted avg	0.9015	0.9033	0.9014	1985	weighted avg	0.9613	0.9608	0.9609	1990

Aspect: GIÁ TIỀN				
	precision	recall	f1-score	support
KHÔNG	0.9865	0.9784	0.9824	1714
TIÊU CỰC	0.7037	0.7600	0.7308	50
TÍCH CỰC	0.8850	0.9259	0.9050	216

accuracy			0.9672	1980
macro avg	0.8584	0.8881	0.8727	1980
weighted avg	0.9683	0.9672	0.9676	1980

Aspect: KHÍA CẠNH KHÁC				
	precision	recall	f1-score	support
KHÔNG	0.8778	0.9083	0.8928	1352
TIÊU CỰC	0.7750	0.7078	0.7399	438
TÍCH CỰC	0.5982	0.5537	0.5751	121
accuracy			0.8399	1911
macro avg	0.7503	0.7233	0.7359	1911
weighted avg	0.8365	0.8399	0.8376	1911

Aspect: ĐÁNH GIÁ CHUNG				
	precision	recall	f1-score	support
KHÔNG	0.8416	0.8590	0.8502	1305
TIÊU CỰC	0.5943	0.4883	0.5361	213
TÍCH CỰC	0.7433	0.7620	0.7525	437

accuracy			0.7969	1955
macro avg	0.7264	0.7031	0.7129	1955
weighted avg	0.7927	0.7969	0.7942	1955

## Fold 3/5:

Validation Results:

Aspect: CHẤT LƯỢNG				
	precision	recall	f1-score	support
KHÔNG	0.8435	0.8051	0.8239	549
TIÊU CỰC	0.9177	0.9163	0.9170	621
TÍCH CỰC	0.9075	0.9398	0.9234	731

accuracy			0.8932	1901
macro avg	0.8896	0.8871	0.8881	1901
weighted avg	0.8924	0.8932	0.8926	1901

Aspect: GIAO HÀNG				
	precision	recall	f1-score	support
KHÔNG	0.9676	0.9594	0.9635	1307
TIÊU CỰC	0.8333	0.8370	0.8352	227
TÍCH CỰC	0.9498	0.9710	0.9603	448

accuracy			0.9480	1982
macro avg	0.9169	0.9225	0.9196	1982
weighted avg	0.9482	0.9480	0.9481	1982

Aspect: HÌNH THỨC				
	precision	recall	f1-score	support
KHÔNG	0.9333	0.9183	0.9257	1493
TIÊU CỰC	0.5326	0.5158	0.5241	95
TÍCH CỰC	0.7929	0.8467	0.8190	398

accuracy			0.8847	1986
macro avg	0.7529	0.7603	0.7562	1986
weighted avg	0.8860	0.8847	0.8851	1986

Aspect: ĐỒNG GÓI				
	precision	recall	f1-score	support
KHÔNG	0.9804	0.9750	0.9777	1643
TIÊU CỰC	0.6629	0.6860	0.6743	86
TÍCH CỰC	0.9511	0.9731	0.9620	260

accuracy			0.9623	1989
macro avg	0.8648	0.8781	0.8713	1989
weighted avg	0.9629	0.9623	0.9625	1989

Aspect: GIÁ TIỀN				
	precision	recall	f1-score	support
KHÔNG	0.9930	0.9877	0.9903	1714
TIÊU CỰC	0.8070	0.9020	0.8519	51
TÍCH CỰC	0.9174	0.9302	0.9238	215

accuracy			0.9793	1980
macro avg	0.9058	0.9400	0.9220	1980
weighted avg	0.9800	0.9793	0.9796	1980

Aspect: ĐÁNH GIÁ CHUNG				
	precision	recall	f1-score	support
KHÔNG	0.8273	0.8597	0.8432	1304
TIÊU CỰC	0.5569	0.4366	0.4895	213
TÍCH CỰC	0.7436	0.7352	0.7394	438

accuracy			0.7857	1955
macro avg	0.7093	0.6771	0.6907	1955
weighted avg	0.7791	0.7857	0.7814	1955

Aspect: KHÍA CẠNH KHÁC				
	precision	recall	f1-score	support
KHÔNG	0.8721	0.8817	0.8769	1353
TIÊU CỰC	0.7222	0.7403	0.7312	439
TÍCH CỰC	0.5895	0.4628	0.5185	121

accuracy			0.8228	1913
macro avg	0.7279	0.6950	0.7089	1913
weighted avg	0.8198	0.8228	0.8208	1913

## Fold 4/5:

Validation Results:

Aspect: CHẤT LƯỢNG					Aspect: GIAO HÀNG				
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.8007	0.8051	0.8029	549	KHÔNG	0.9587	0.9602	0.9595	1307
TIÊU CỰC	0.8987	0.8857	0.8921	621	TIÊU CỰC	0.8194	0.8194	0.8194	227
TÍCH CỰC	0.9119	0.9194	0.9156	732	TÍCH CỰC	0.9596	0.9554	0.9575	448
accuracy			0.8754	1902	accuracy			0.9430	1982
macro avg	0.8704	0.8701	0.8702	1902	macro avg	0.9126	0.9117	0.9121	1982
weighted avg	0.8755	0.8754	0.8754	1902	weighted avg	0.9430	0.9430	0.9430	1982
Aspect: HÌNH THỨC					Aspect: ĐÓNG GÓI				
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.9400	0.9230	0.9314	1493	KHÔNG	0.9840	0.9726	0.9783	1643
TIÊU CỰC	0.5444	0.5213	0.5326	94	TIÊU CỰC	0.6634	0.7791	0.7166	86
TÍCH CỰC	0.8042	0.8668	0.8343	398	TÍCH CỰC	0.9470	0.9615	0.9542	260
accuracy			0.8927	1985	accuracy			0.9628	1989
macro avg	0.7629	0.7704	0.7661	1985	macro avg	0.8648	0.9044	0.8830	1989
weighted avg	0.8940	0.8927	0.8931	1985	weighted avg	0.9653	0.9628	0.9638	1989
Aspect: GIÁ TIỀN					Aspect: ĐÁNH GIÁ CHUNG				
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.9929	0.9860	0.9895	1714	KHÔNG	0.8342	0.8374	0.8358	1304
TIÊU CỰC	0.7222	0.7647	0.7429	51	TIÊU CỰC	0.5842	0.5540	0.5687	213
TÍCH CỰC	0.9330	0.9721	0.9522	215	TÍCH CỰC	0.7065	0.7162	0.7114	437
accuracy			0.9788	1980	accuracy			0.7794	1954
macro avg	0.8827	0.9076	0.8948	1980	macro avg	0.7083	0.7026	0.7053	1954
weighted avg	0.9795	0.9788	0.9791	1980	weighted avg	0.7784	0.7794	0.7789	1954
Aspect: KHÍA CẠNH KHÁC									
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.8801	0.9002	0.8900	1353					
TIÊU CỰC	0.7395	0.7244	0.7319	439					
TÍCH CỰC	0.6100	0.5000	0.5495	122					
accuracy			0.8344	1914					
macro avg	0.7432	0.7082	0.7238	1914					
weighted avg	0.8306	0.8344	0.8320	1914					

## Fold 5/5:

Validation Results:

Aspect: CHẤT LƯỢNG					Aspect: GIAO HÀNG				
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.7910	0.8327	0.8113	550	KHÔNG	0.9596	0.9625	0.9610	1307
TIÊU CỰC	0.9272	0.8824	0.9043	621	TIÊU CỰC	0.8387	0.8018	0.8198	227
TÍCH CỰC	0.9100	0.9112	0.9106	732	TÍCH CỰC	0.9581	0.9710	0.9645	448
accuracy			0.8791	1903	accuracy			0.9460	1982
macro avg	0.8761	0.8755	0.8754	1903	macro avg	0.9188	0.9118	0.9151	1982
weighted avg	0.8812	0.8791	0.8798	1903	weighted avg	0.9454	0.9460	0.9457	1982
Aspect: HÌNH THỨC					Aspect: ĐÓNG GÓI				
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.9252	0.9283	0.9268	1493	KHÔNG	0.9786	0.9769	0.9778	1642
TIÊU CỰC	0.6765	0.4894	0.5679	94	TIÊU CỰC	0.7159	0.7326	0.7241	86
TÍCH CỰC	0.7876	0.8291	0.8078	398	TÍCH CỰC	0.9425	0.9462	0.9443	260
accuracy			0.8877	1985	accuracy			0.9623	1988
macro avg	0.7964	0.7489	0.7675	1985	macro avg	0.8790	0.8852	0.8821	1988
weighted avg	0.8859	0.8877	0.8859	1985	weighted avg	0.9626	0.9623	0.9624	1988

Aspect: GIÁ TIỀN					Aspect: ĐÁNH GIÁ CHUNG				
	precision	recall	f1-score	support		precision	recall	f1-score	support
KHÔNG	0.9936	0.9895	0.9915	1714	KHÔNG	0.8285	0.8513	0.8398	1305
TIÊU CỰC	0.7736	0.8039	0.7885	51	TIÊU CỰC	0.6000	0.5634	0.5811	213
TÍCH CỰC	0.9318	0.9535	0.9425	215	TÍCH CỰC	0.7222	0.6842	0.7027	437
accuracy			0.9808	1980	accuracy			0.7826	1955
macro avg	0.8997	0.9156	0.9075	1980	macro avg	0.7169	0.6996	0.7079	1955
weighted avg	0.9812	0.9808	0.9810	1980	weighted avg	0.7798	0.7826	0.7809	1955

Aspect: KHÓA CẠNH KHÁC				
	precision	recall	f1-score	support
KHÔNG	0.8576	0.9135	0.8847	1352
TIÊU CỰC	0.7816	0.6765	0.7253	439
TÍCH CỰC	0.5806	0.4426	0.5023	122
accuracy			0.8291	1913
macro avg	0.7400	0.6775	0.7041	1913
weighted avg	0.8225	0.8291	0.8237	1913