TRƯỜNG ĐẠI HỌC ĐÀ LẠT

KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO MÔN HỌC CÁC PHƯƠNG PHÁP HỌC MÁY

HỌC MÁY PHÂN TÍCH CẢM XÚC BÌNH LUẬN TRÊN MẠNG XÃ HỘI

Sinh viên thực hiện

Nguyễn Việt Linh	**2115230**
Nguyễn Thọ Thành	**2115269**
Nguyễn Cao Nhất Duy	**2112973**
Nguyễn Dương Công Bảo	**2115188**

Giáo viên hướng dẫn: Tạ Hoàng Thắng

Đà Lạt, 17 tháng 11 năm 2024

MỤC LỤC

KHÁI NIỆM

- Precision, hay còn gọi là độ chính xác, là một trong những chỉ số quan trọng để đánh giá hiệu suất của một mô hình học máy, đặc biệt là trong các bài toán phân loại. Precision được định nghĩa là tỷ lệ giữa số lượng dự đoán đúng về các đối tượng thuộc một lớp cụ thể so với tổng số lượng dự đoán mà mô hình cho rằng thuộc lớp đó.

Công thức tính precision là:

Precision = (True Positives) / (True Positives + False Positives)

- Recall: hay còn gọi là độ nhạy, là một chỉ số quan trọng khác để đánh giá hiệu suất của một mô hình học máy trong các bài toán phân loại. Recall được định nghĩa là tỷ lệ giữa số lượng dự đoán đúng về các đối tượng thuộc một lớp cụ thể so với tổng số lượng đối tượng thực sự thuộc lớp đó.
- F1-Score: là một chỉ số dùng để đánh giá hiệu suất của một mô hình học máy, kết hợp cả hai chỉ số Precision và Recall. Chỉ số này đặc biệt hữu ích trong các bài toán phân loại với dữ liệu mất cân bằng, nơi mà tỷ lệ giữa các lớp không đồng đều.

Công thức tính F1-Score là:

F1-Score = 2 * (Precision * Recall) / (Precision + Recall)

- Support: chỉ số support trong học máy thể hiện số lượng mẫu thực sự thuộc về một lớp cụ thể trong tập dữ liệu. Nó không chỉ giúp đánh giá độ phổ biến của các lớp khác nhau mà còn ảnh hưởng đến các chỉ số khác như Precision và Recall. Support đóng vai trò quan trọng trong việc hiểu rõ tính chất của dữ liệu, đặc biệt là khi làm việc với các tập dữ liệu có sự phân bổ không đồng đều giữa các lớp.
- Accuracy: là một chỉ số quan trọng để đánh giá tổng thể hiệu suất của một mô hình học máy trong các bài toán phân loại. Accuracy được định nghĩa là tỷ lệ giữa số lượng dự đoán đúng (cả True Positives và True Negatives) so với tổng số lượng dự đoán. Accuracy thể hiện khả năng của mô hình trong việc phân loại chính xác các đối tượng thuộc các lớp khác nhau.
- Macro Average: là chỉ số tính toán trung bình của các chỉ số đánh giá hiệu suất (như Precision, Recall, F1-Score) cho từng lớp trong một bài toán phân loại và sau đó lấy trung bình cộng của các chỉ số này. Macro Average đặc biệt hữu ích khi bạn muốn xem hiệu suất của mô hình trên mỗi lớp mà không quá tập trung vào sự chênh lệch về số lượng mẫu giữa

các lớp. Nó giúp có cái nhìn cân bằng hơn về hiệu suất của mô hình trên các lớp khác nhau, đặc biệt khi làm việc với các tập dữ liệu không cân bằng.

- Confusion Matrix: là một công cụ quan trọng để đánh giá hiệu suất của một mô hình học máy trong các bài toán phân loại. Confusion Matrix là một bảng ma trận gồm bốn ô, biểu diễn số lượng dự đoán đúng và sai của mô hình qua các lớp. Các ô của Confusion Matrix bao gồm:
    - True Positives (TP): Số lượng mẫu thuộc lớp dương tính và được dự đoán đúng là dương tính.
    - False Positives (FP): Số lượng mẫu thuộc lớp âm tính nhưng được dự đoán sai là dương tính.
    - True Negatives (TN): Số lượng mẫu thuộc lớp âm tính và được dự đoán đúng là âm tính.
    - False Negatives (FN): Số lượng mẫu thuộc lớp dương tính nhưng được dự đoán sai là âm tính.
- Các nhãn dữ liệu dùng để training
    - Positive: dành cho bình luận mang tích chất tích cực
    - Neutral: dành cho bình luận mang tích chất trung tính
    - Negative: dành cho bình luận mang tính chất tiêu cực
    - Toxic: nhãn gán cho các bình luận có tính chất độc hại hoặc chứa những từ độc hại

MỞ ĐẦU

Kính thưa Quý Thầy Cô và các bạn,

Trong thời đại công nghệ số hiện nay, mạng xã hội đã trở thành một phần không thể thiếu trong cuộc sống hàng ngày của con người. Người dùng thường xuyên chia sẻ ý kiến, cảm xúc và quan điểm của mình thông qua các bình luận trên các nền tảng mạng xã hội. Việc phân tích cảm xúc từ các bình luận này không chỉ giúp hiểu rõ hơn về tâm lý và hành vi của người dùng mà còn cung cấp thông tin quý giá cho các doanh nghiệp và tổ chức trong việc cải thiện sản phẩm, dịch vụ và chiến lược kinh doanh. Vì vậy nhóm chúng em đã thực hiện mô hình học máy phân tích cảm xúc bình luận, đây là một công cụ mạnh mẽ giúp tự động hóa quá trình này. Bằng cách sử dụng các thuật toán học máy, mô hình có thể phân loại các bình luận thành các nhóm cảm xúc khác nhau như tích cực, tiêu cực, trung tính hoặc độc hại. Điều này không chỉ giúp tiết kiệm thời gian và công sức mà còn đảm bảo độ chính xác cao hơn so với phương pháp thủ công.

Trong suốt quá trình thực hiện dự án, chúng em đã trải qua nhiều thử thách và học hỏi được vô số kiến thức mới mẻ. Nhờ sự hướng dẫn tận tình và những bài giảng đầy tâm huyết của Thầy Tạ Hoàng Thắng, nhóm chúng em đã có thể hoàn thành bài nghiên cứu này một cách trọn vẹn và thành công. Chúng em xin gửi lời cảm ơn chân thành và sâu sắc đến Thầy vì những kiến thức, sự động viên và những lời khuyên quý báu mà Thầy đã dành cho chúng tôi trong suốt quá trình học tập và nghiên cứu.

Chúng tôi cũng xin cảm ơn các thành viên trong nhóm vì tinh thần làm việc chăm chỉ, sự đoàn kết và hỗ trợ lẫn nhau để cùng nhau vượt qua các khó khăn và hoàn thành tốt nhiệm vụ được giao.

Kính mong Quý Thầy Cô và các bạn dành thời gian quý báu để đọc và góp ý cho bài nghiên cứu của chúng em.

CHƯƠNG 1. GIỚI THIỆU

Trong thời đại số hóa và kết nối toàn cầu, các mạng xã hội đã trở thành nền tảng chính để con người giao tiếp, chia sẻ ý kiến và bày tỏ cảm xúc. Việc phân tích và hiểu được cảm xúc từ các bình luận trên mạng xã hội không chỉ giúp các doanh nghiệp, tổ chức nắm bắt được tâm lý khách hàng, mà còn đóng vai trò quan trọng trong nhiều lĩnh vực như marketing, chăm sóc khách hàng, và nghiên cứu xã hội.

Đề tài nghiên cứu của nhóm em tập trung vào việc xây dựng một hệ thống nhằm phân tích rõ cảm xúc của con người trong từng bình luận trên mạng xã hội, đồng thời phát hiện các ngôn từ mang tính độc hại. Cụ thể, nhóm em đã triển khai và so sánh hiệu suất của bốn mô hình học máy khác nhau:

- **Logistic Regression**: Một phương pháp đơn giản và hiệu quả trong phân loại tuyến tính.
- **Naive Bayes**: Một mô hình xác suất với giả định độc lập giữa các đặc trưng.
- **Support Vector Machine (SVM)**: Một phương pháp mạnh mẽ với khả năng phân tách phi tuyến tính.
- **Long Short-Term Memory (LSTM)**: Một kiến trúc mạng nơ-ron hồi quy sâu có khả năng xử lý dữ liệu chuỗi thời gian.

Mục tiêu chính của bài học này là nắm bắt và học tập cách thức huấn luyện và áp dụng các mô hình học máy tiên tiến để phân loại cảm xúc từ các bình luận trên mạng xã hội, và phát hiện các ngôn từ độc hại, góp phần cải thiện môi trường giao tiếp trên các nền tảng trực tuyến.

Quá trình học tập và triển khai được tiến hành qua các bước: chuẩn bị dữ liệu, gán nhãn, tiền xử lý, huấn luyện mô hình, và đánh giá hiệu suất. Kết quả thực nghiệm sẽ cung cấp cái nhìn sâu sắc về ưu và nhược điểm của từng mô hình, từ đó giúp nhóm em hiểu rõ hơn về ứng dụng thực tế của các phương pháp học máy.

1.1 Thành viên thực hiện mô hình

Mô Hình	Thành viên tìm hiểu và thực hiện
Naive Bayes	Nguyễn Cao Nhất Duy
Logistic Regression	Nguyễn Việt Linh
Support Vector Machine	Nguyễn Thọ Thành
Long Short-Term Memory	Nguyễn Dương Công Bảo

1.2 Quy trình huấn luyện mô hình

1.2.1 Chuẩn bị Dữ liệu

Thu thập Dữ liệu: Tập hợp các bình luận từ mạng xã hội, được lưu trữ trong các file demo3.csv và keyword.csv.

Làm sạch Dữ liệu: Xử lý các dữ liệu thiếu, loại bỏ các ký tự đặc biệt, và chuẩn hóa văn bản để đảm bảo tính nhất quán trong dữ liệu đầu vào.

1.2.2 Gán Nhãn Dữ liệu

Sử dụng Từ khóa: Áp dụng danh sách từ khóa từ file keyword.csv để gán nhãn cho từng bình luận, xác định cảm xúc (tích cực, trung lập, tiêu cực) và nhận diện ngôn từ độc hại.

1.2.3 Tiền Xử lý Dữ liệu

Tokenization: Phân chia văn bản thành các từ hoặc cụm từ nhỏ hơn (tokens).

Stemming/Lemmatization: Chuyển đổi các từ về dạng gốc để giảm thiểu số lượng từ khác nhau có cùng ý nghĩa.

Vector hóa Văn bản: Sử dụng các kỹ thuật như TF-IDF hoặc Word Embeddings để chuyển đổi văn bản thành các vector số, giúp mô hình hiểu và xử lý được.

1.2.4 Chia Dữ liệu

Tập Huấn Luyện và Tập Kiểm Tra: Chia dữ liệu thành hai phần: 80% cho huấn luyện mô hình và 20% cho kiểm tra hiệu suất mô hình.

1.2.5 Huấn Luyện Mô hình

Logistic Regression: Huấn luyện mô hình với thuật toán hồi quy logistic.

Naive Bayes: Sử dụng thuật toán Naive Bayes để huấn luyện mô hình.

SVM: Áp dụng thuật toán Support Vector Machine để huấn luyện mô hình.

LSTM: Huấn luyện mô hình với mạng nơ-ron hồi quy LSTM để xử lý dữ liệu chuỗi thời gian.

1.2.6 Đánh giá Hiệu suất Mô hình

Classification Report: Tạo báo cáo phân loại, bao gồm các chỉ số như độ chính xác (precision), khả năng hồi phục (recall), và F1-score cho từng nhãn cảm xúc.

Confusion Matrix: Tạo ma trận nhầm lẫn để xem xét số lượng dự đoán chính xác và nhầm lẫn của từng mô hình.

Biểu đồ So sánh: Vẽ các biểu đồ so sánh giữa số lượng tham số và độ chính xác của từng mô hình để đánh giá hiệu suất.

1.2.7 Tối Ưu và Hiệu Chỉnh Mô hình

Fine-tuning: Điều chỉnh các tham số của mô hình để cải thiện hiệu suất.

Cross-validation: Áp dụng kỹ thuật kiểm tra chéo để đảm bảo mô hình không bị overfitting và có khả năng tổng quát tốt.

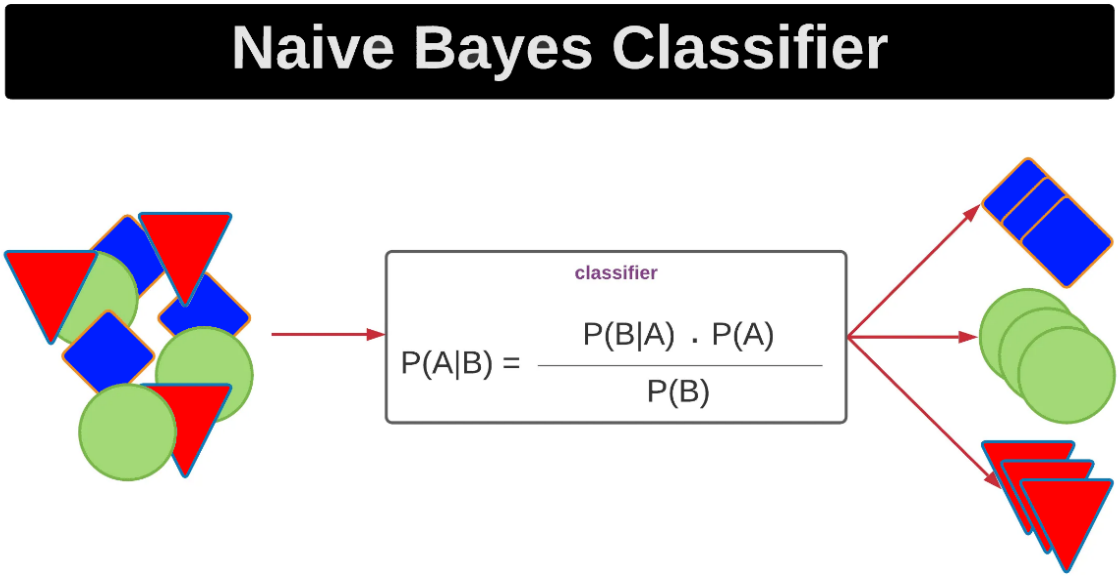
1.2.8 Triển khai và Ứng dụng

Triển khai Mô hình: Áp dụng mô hình đã huấn luyện vào hệ thống phân tích cảm xúc thực tế.

Giám sát và Bảo trì: Giám sát hiệu suất của mô hình khi triển khai thực tế và thực hiện bảo trì, cập nhật mô hình khi cần thiết.

CHƯƠNG 2. CÁC PHƯƠNG PHÁP HUẤN LUYỆN

2.1 Naive Bayes



2.1.1 Khái niệm

Naive Bayes là một nhóm các thuật toán phân loại dựa trên Định lý Bayes với giả định đơn giản rằng các đặc trưng (features) của dữ liệu là độc lập với nhau. Mô hình này rất phổ biến trong việc phân loại văn bản, chẳng hạn như lọc thư rác, phân loại tài liệu và phát hiện cảm xúc.

2.1.2 Cách thức hoạt động

Mô hình Naive Bayes dự đoán nhãn cho dữ liệu mới dựa trên xác suất có điều kiện của từng nhãn đối với các đặc trưng của dữ liệu. Xác suất này được tính dựa trên dữ liệu huấn luyện và sử dụng Định lý Bayes để cập nhật xác suất khi có thông tin mới.

- **Huấn luyện:**
  - Thu thập một tập dữ liệu bình luận đã được gán nhãn cảm xúc (tích cực, tiêu cực, trung tính).
  - Tính toán xác suất tiên nghiệm của mỗi loại cảm xúc (ví dụ: xác suất một bình luận ngẫu nhiên là tích cực).

- Tính toán xác suất có điều kiện của mỗi từ xuất hiện trong mỗi loại cảm xúc (ví dụ: xác suất từ "tuyệt vời" xuất hiện trong bình luận tích cực).

- **Phân loại:**

- Với một bình luận mới, tính toán xác suất bình luận đó thuộc về mỗi loại cảm xúc dựa trên các xác suất đã tính ở bước huấn luyện.
- Phân loại bình luận vào loại cảm xúc có xác suất cao nhất.

2.1.3 Công thức

Định lý Bayes được biểu diễn như sau:

$$P(y \mid X) = \frac{P(X \mid y) \cdot P(y)}{P(X)}$$

Trong đó:

- $P(y \mid X)$là xác suất của nhãn y cho một dữ liệu đầu vào X.
- $P(X \mid y)$ là xác suất của dữ liệu X cho một nhãn y nhất định.
- $P(y)$ là xác suất tiên nghiệm của nhãn y.
- $P(X)$ là xác suất của dữ liệu X (hằng số đối với mọi nhãn).

2.1.4 Ví dụ

Giả sử chúng ta có một bình luận như sau: "Dịch vụ rất tệ và nhân viên không thân thiện".

Mô hình Naive Bayes sẽ phân tích bình luận này và tính toán xác suất cho từng nhãn cảm xúc (chẳng hạn: "Positive", "Negative", "Neutral") dựa trên các từ trong bình luận và dữ liệu huấn luyện trước đó.

Ví dụ, mô hình có thể tính toán các xác suất như sau:

- P(Negative|X)=0.75P(Negative|X) = 0.75
- P(Neutral|X)=0.15P(Neutral|X) = 0.15
- P(Positive|X)=0.10P(Positive|X) = 0.10

Kết quả cuối cùng, mô hình sẽ gán nhãn "Negative" cho bình luận này vì xác suất P(Negative|X)P(Negative|X) là cao nhất.

2.1.5 Ưu điểm

- **Đơn giản và nhanh chóng**: Dễ triển khai và thực thi nhanh chóng.

- **Hiệu quả với dữ liệu nhỏ**: Hoạt động tốt ngay cả khi tập dữ liệu nhỏ và không cần nhiều tài nguyên tính toán.

- **Giả định độc lập**: Giả định độc lập giữa các đặc trưng giúp đơn giản hóa việc tính toán xác suất.

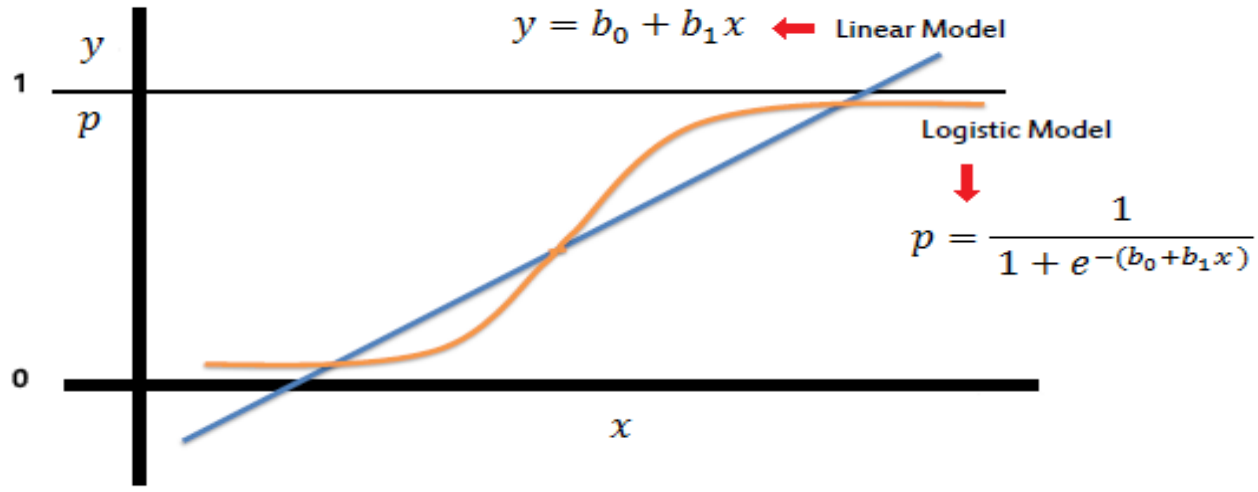
2.1.6 Nhược Điểm

- **Giả định độc lập**: Giả định này không luôn luôn đúng trong thực tế, có thể ảnh hưởng đến hiệu suất của mô hình.

- **Hiệu suất thấp với dữ liệu phức tạp**: Khi các đặc trưng không độc lập hoặc có mối quan hệ phức tạp, mô hình có thể không hoạt động tốt.

- **Không cập nhật được**: Naive Bayes không dễ dàng cập nhật với dữ liệu mới mà không phải huấn luyện lại toàn bộ mô hình.

2.2 Logistic Regression



2.2.1 Khái niệm

Logistic Regression là một thuật toán học máy dùng cho các bài toán phân loại nhị phân (binary classification). Mặc dù tên gọi "regression" (hồi quy), Logistic Regression là một phương pháp phân loại, chứ không phải hồi quy. Nó dùng hàm logistic (hay còn gọi là sigmoid) để dự đoán xác suất của nhãn đầu ra.

2.2.2 Cách thức hoạt động

Mô hình Logistic Regression dự đoán nhãn cho dữ liệu mới bằng cách sử dụng hàm logistic để chuyển đổi đầu ra của mô hình tuyến tính thành xác suất. Nhãn đầu ra được xác định dựa trên ngưỡng xác suất (thông thường là 0.5).

- **Biểu diễn dữ liệu:** mỗi bình luận được biểu diễn thành một điểm dữ liệu trong không gian vector, với mỗi chiều tương ứng với một đặc trưng.
- **Hàm sigmoid:** Logistic Regression sử dụng hàm sigmoid để ánh xạ giá trị đầu vào (tổng trọng số của các đặc trưng) thành một giá trị xác suất.
- **Huấn luyện:** Quá trình huấn luyện nhằm tìm ra bộ trọng số tối ưu cho các đặc trưng sao cho mô hình dự đoán xác suất chính xác nhất trên tập huấn luyện.
- **Phân loại:** Với một bình luận mới, mô hình tính toán xác suất của mỗi loại cảm xúc. Bình luận sẽ được phân loại vào loại cảm xúc có xác suất cao nhất.

2.2.3 Công thức tính

Hàm logistic (hay sigmoid) được biểu diễn như sau:

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

Trong đó:

$x = w^T \cdot X + b$

w là vector trọng số (weights)

x là vector đặc trưng (features)

b là hệ số điều chỉnh (bias)

Xác suất dự đoán P(y|X) được tính như sau:

$$P(y \mid X) = \sigma(w^T \cdot X + b)$$

2.2.4 Ví dụ

Giả sử chúng ta có một bình luận như sau: "Dịch vụ tốt nhưng giá hơi cao".

Mô hình Logistic Regression sẽ phân tích bình luận này và tính toán xác suất cho từng nhãn cảm xúc (chẳng hạn: "Positive", "Negative", "Neutral") dựa trên các từ trong bình luận và dữ liệu huấn luyện trước đó.

Ví dụ, mô hình có thể tính toán các xác suất như sau:

- P(Negative|X)=0.25P(Negative|X) = 0.25
- P(Neutral|X)=0.20P(Neutral|X) = 0.20
- P(Positive|X)=0.55P(Positive|X) = 0.55

Kết quả cuối cùng, mô hình sẽ gán nhãn "Positive" cho bình luận này vì xác suất P(Positive|X)P(Positive|X) là cao nhất.

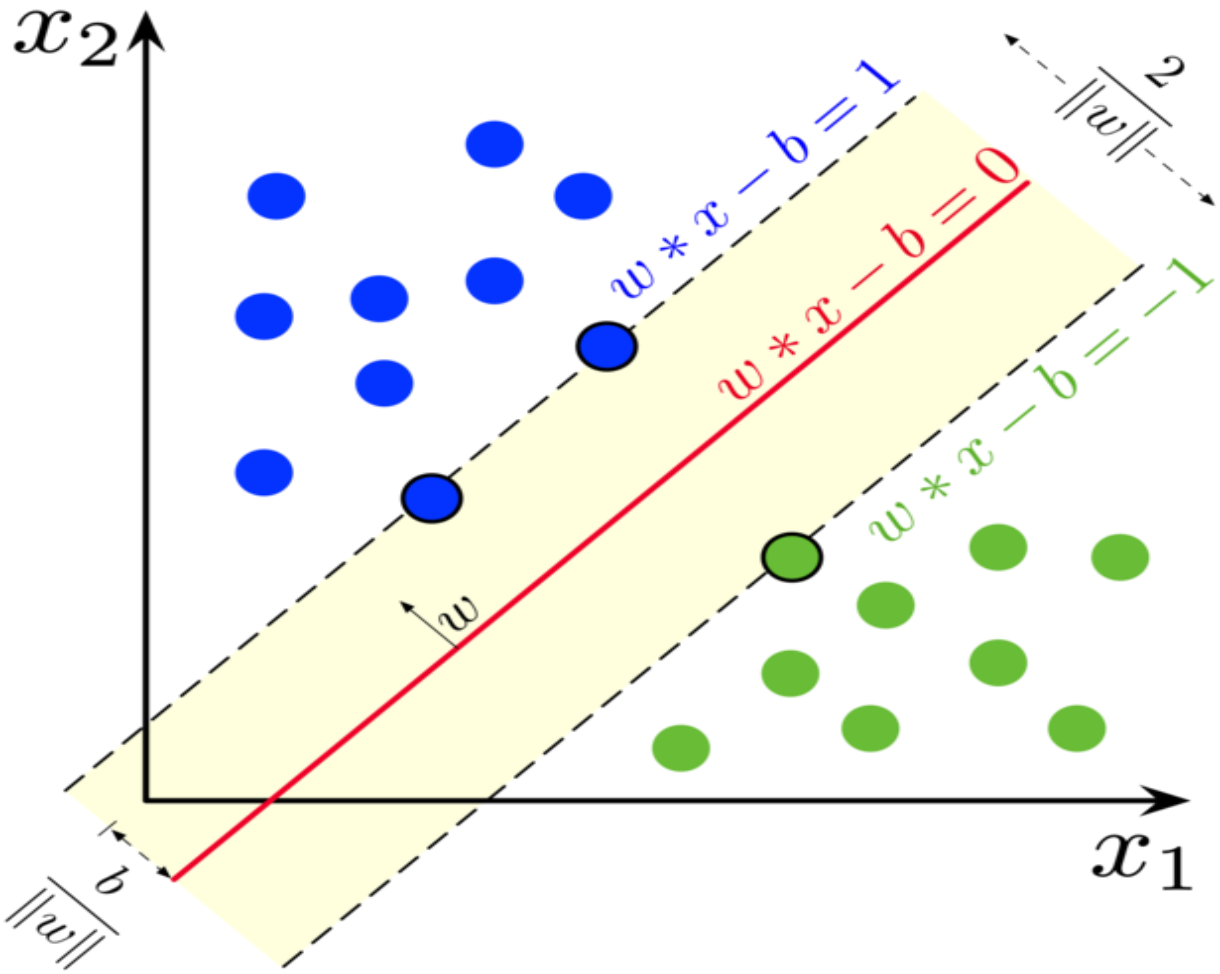
2.2.5 Ưu Điểm

- **Đơn giản và hiệu quả**: Dễ hiểu, dễ triển khai và thực thi nhanh chóng.
- **Hiệu quả với dữ liệu nhỏ**: Hoạt động tốt với các tập dữ liệu nhỏ và vừa.
- **Không yêu cầu điều chỉnh nhiều**: Thường hoạt động tốt với các tham số mặc định mà không cần tối ưu nhiều.

2.2.6 Nhược điểm

- **Giới hạn trong phân loại tuyến tính**: Chỉ hoạt động tốt với các bài toán phân loại tuyến tính.
- **Không mạnh mẽ với dữ liệu phức tạp**: Hiệu suất có thể giảm khi dữ liệu không tuyến tính hoặc có nhiều nhiễu.
- **Không hỗ trợ tốt cho phân loại nhiều lớp**: Cần mở rộng để xử lý các bài toán phân loại đa lớp (multiclass classification).

2.3 SVM



2.3.1 Khái niệm

Support Vector Machine (SVM) là một thuật toán học máy mạnh mẽ được sử dụng cho các bài toán phân loại và hồi quy. Mô hình SVM tìm cách xác định một siêu phẳng (hyperplane) tốt nhất để phân tách dữ liệu thuộc các nhãn khác nhau. Đối với dữ liệu không tuyến tính, SVM sử dụng các hạt nhân (kernels) để ánh xạ dữ liệu sang không gian có chiều cao hơn, nơi mà dữ liệu có thể phân tách tuyến tính.

2.3.2 Cách thức hoạt động

Mô hình SVM hoạt động bằng cách tìm kiếm siêu phẳng tối ưu trong không gian đặc trưng để phân tách các nhãn khác nhau với khoảng cách lớn nhất (margin) giữa các lớp. Các điểm dữ liệu gần siêu phẳng nhất gọi là "support vectors" và có ảnh hưởng trực tiếp đến vị trí của siêu phẳng.

- **Biểu diễn dữ liệu:** Mỗi bình luận được biểu diễn thành một điểm dữ liệu trong không gian vector, với mỗi chiều tương ứng với một đặc trưng (ví dụ: tần suất xuất hiện của một từ, giá trị TF-IDF).
- **Tìm kiếm siêu phẳng tối ưu:** SVM tìm kiếm một siêu phẳng để phân chia các điểm dữ liệu sao cho margin là lớn nhất.
- **Phân loại:** Các bình luận mới sẽ được phân loại dựa trên vị trí của chúng so với siêu phẳng này.

2.3.3 Công thức

Siêu phẳng trong không gian d chiều được biểu diễn như sau:

$$w \cdot x - b = 0$$

Trong đó:

w là vector trọng số

x là vector đặc trưng

b là hệ số điều chỉnh

Mục tiêu của SVM là tối đa hóa khoảng cách giữa các support vectors và siêu phẳng, được biểu diễn bởi hàm mất mát (loss function):

$$\boldsymbol{L(w,b) = \frac{1}{2}\|w\|2 + C\sum_{i=1}^{n} max\ (0, 1 - y_i(w \cdot x_i - b))}$$

2.3.4 Ví dụ

Giả sử chúng ta có một bình luận như sau: "Sản phẩm không tốt, tôi rất thất vọng".

Mô hình SVM sẽ phân tích bình luận này và xác định vị trí của nó trong không gian đặc trưng, sau đó sử dụng siêu phẳng để dự đoán nhãn cảm xúc (chẳng hạn: "Positive", "Negative", "Neutral").

Ví dụ, mô hình có thể xác định các khoảng cách như sau:

- Khoảng cách đến siêu phẳng "Negative" = 0.8
- Khoảng cách đến siêu phẳng "Neutral" = 0.3
- Khoảng cách đến siêu phẳng "Positive" = 0.1

Kết quả cuối cùng, mô hình sẽ gán nhãn "Negative" cho bình luận này vì khoảng cách đến siêu phẳng "Negative" là lớn nhất.

2.3.5 Ưu điểm

- **Mạnh mẽ với dữ liệu phức tạp**: SVM hoạt động tốt với dữ liệu có đặc trưng phi tuyến tính và có thể sử dụng các hàm kernel để xử lý dữ liệu này.

- **Giảm thiểu overfitting**: SVM tập trung vào các điểm dữ liệu gần siêu phẳng, giúp giảm thiểu nguy cơ overfitting.

- **Hiệu quả cao**: Thích hợp cho các bài toán có nhiều chiều dữ liệu và nhiều nhãn.

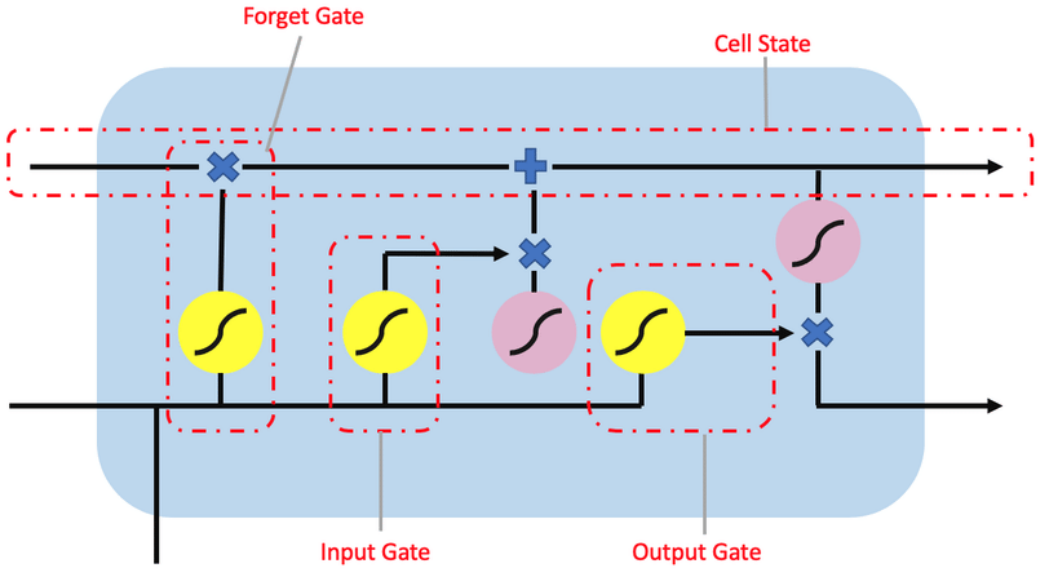
2.3.6 Nhược điểm

- **Yêu cầu tài nguyên tính toán cao**: SVM có thể yêu cầu nhiều tài nguyên tính toán, đặc biệt là với các tập dữ liệu lớn.

- **Khó khăn trong việc điều chỉnh tham số**: Việc lựa chọn tham số CC và kernel phù hợp đòi hỏi phải thử nghiệm và tối ưu hóa cẩn thận.

- **Hiệu suất thấp với dữ liệu nhiễu**: SVM có thể bị ảnh hưởng bởi các điểm nhiễu (noise) trong dữ liệu huấn luyện.

2.4 Long Short-Term Memory (LSTM)



2.4.1 Khái niệm

Long Short-Term Memory (LSTM) là một loại mạng nơ-ron hồi quy sâu (RNN) được thiết kế để ghi nhớ và học các mẫu trong dữ liệu chuỗi thời gian dài. LSTM vượt trội hơn các RNN thông thường nhờ khả năng giảm thiểu vấn đề tiêu tan và bùng nổ gradient, giúp nó học các phụ thuộc dài hạn trong dữ liệu.

2.4.2 Cách thức hoạt động

Mô hình LSTM sử dụng các "ô nhớ" (memory cells) để lưu giữ thông tin và ba cổng chính (input gate, forget gate, output gate) để kiểm soát luồng thông tin vào và ra khỏi ô nhớ. Các cổng này giúp LSTM quyết định thông tin nào cần lưu lại, thông tin nào cần quên, và thông tin nào cần sử dụng cho dự đoán tiếp theo.

- **Cấu trúc tế bào LSTM:** Mỗi tế bào LSTM bao gồm 3 cổng:
  - **Cổng quên (forget gate):** Quyết định thông tin nào cần được loại bỏ khỏi trạng thái tế bào.
  - **Cổng đầu vào (input gate):** Quyết định thông tin mới nào cần được thêm vào trạng thái tế bào.
  - **Cổng đầu ra (output gate):** Quyết định thông tin nào từ trạng thái tế bào sẽ được sử dụng làm đầu ra.

- **Xử lý tuần tự:** LSTM xử lý văn bản theo từng từ. Tại mỗi bước thời gian, tế bào LSTM nhận đầu vào là từ hiện tại và trạng thái ẩn từ bước thời gian trước đó. Sau đó, tế bào LSTM cập nhật trạng thái ẩn và tạo ra đầu ra dựa trên cơ chế cổng.
- **Phân loại cảm xúc:** Đầu ra của LSTM tại bước thời gian cuối cùng (sau khi xử lý hết câu) được sử dụng để phân loại cảm xúc của bình luận.

2.4.3 Công thức

LSTM được điều hành bởi ba cổng chính:

- **Cổng vào (input gate)**: Quyết định thông tin nào sẽ được thêm vào trạng thái ô nhớ.
- **Cổng quên (forget gate)**: Quyết định thông tin nào sẽ bị quên trong trạng thái ô nhớ.
- **Cổng ra (output gate)**: Quyết định phần nào của trạng thái ô nhớ sẽ được dùng làm đầu ra.

Các công thức của LSTM như sau:

- **Cổng quên**: $\mathbf{f_t} = \sigma(W_f \mathbf{x_t} + U_f h_{t-1} + b_f)$
- **Cổng vào**: $\mathbf{i_t} = \sigma(W_i \mathbf{x_t} + U_i h_{t-1} + b_i)$
- **Cổng ra**: $\mathbf{o_t} = \sigma(W_o \mathbf{x_t} + U_o h_{t-1} + b_o)$

2.4.4 Ví dụ

Giả sử chúng ta có một bình luận như sau: "Không hài lòng với sản phẩm, dịch vụ kém chất lượng".

Mô hình LSTM sẽ phân tích bình luận này, lưu giữ thông tin quan trọng qua các ô nhớ và các cổng, sau đó sử dụng thông tin này để dự đoán nhãn cảm xúc (chẳng hạn: "Positive", "Negative", "Neutral").

Ví dụ, mô hình có thể dự đoán các xác suất như sau:

- $P(\text{Negative}|X) = 0.80$
- $P(\text{Neutral}|X) = 0.10$
- $P(\text{Positive}|X) = 0.10$

Kết quả cuối cùng, mô hình sẽ gán nhãn "Negative" cho bình luận này vì xác suất $P(\text{Negative}|X)$ là cao nhất.

2.4.5 Ưu điểm

- **Khả năng học ngữ cảnh dài hạn**: LSTM có khả năng ghi nhớ thông tin từ lâu dài, giúp nó học các phụ thuộc dài hạn trong dữ liệu chuỗi thời gian.

- **Hiệu quả trong các bài toán tuần tự**: LSTM hoạt động tốt trong các bài toán như dịch ngôn ngữ, nhận diện giọng nói và phân tích cảm xúc.

- **Giảm thiểu vấn đề tiêu tan gradient**: Nhờ cơ chế các cổng, LSTM giảm thiểu vấn đề tiêu tan gradient thường gặp trong các RNN truyền thống.

2.4.6 Nhược điểm

- **Yêu cầu tài nguyên tính toán cao**: LSTM đòi hỏi nhiều tài nguyên tính toán và thời gian huấn luyện lâu hơn so với các mô hình đơn giản.

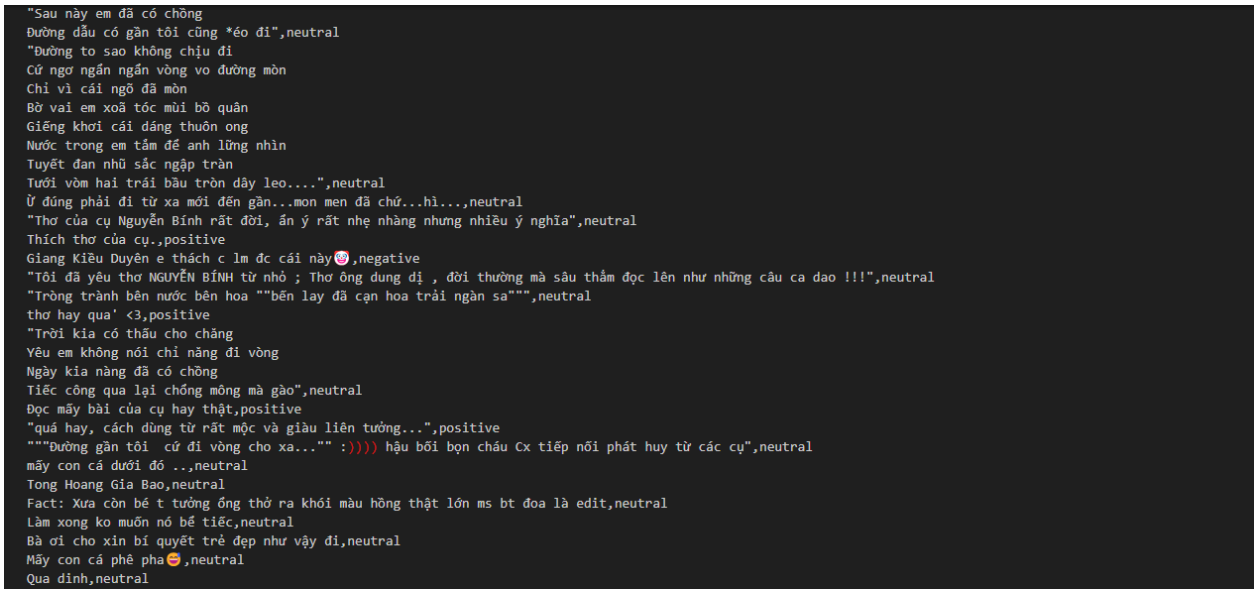
- **Phức tạp**: Cấu trúc phức tạp của LSTM có thể gây khó khăn trong việc tối ưu hóa và triển khai.

- **Hiệu suất thấp với dữ liệu ngắn hạn**: LSTM không phải là lựa chọn tốt nhất cho các bài toán chỉ yêu cầu phân tích dữ liệu ngắn hạn.

CHƯƠNG 3. KẾT QUẢ THỰC NGHIỆM

3.1 Tham số mô hình huấn luyện và dữ liệu training

Tất cả các mô hình đều chạy cùng 1 dữ liệu gồm khoản 400 bình luận facebook (demo) đã được gán nhãn



```
"Sau này em đã có chồng
Đường dẫu có gần tôi cũng *éo đi",neutral
"Đường to sao không chịu đi
Cứ ngơ ngẩn ngẩn vòng vo đường mòn
Chỉ vì cái ngõ đã mòn
Bờ vai em xoã tóc mùi bồ quân
Giếng khơi cái dáng thuôn ong
Nước trong em tắm để anh lừng nhìn
Tuyết đan nhũ sắc ngập tràn
Tưới vòm hai trái bầu tròn dây leo....",neutral
Ừ đúng phải đi từ xa mới đến gần...mon men đã chứ...hì...,neutral
"Thơ của cụ Nguyễn Bính rất đời, ẩn ý rất nhẹ nhàng nhưng nhiều ý nghĩa",neutral
Thích thơ của cụ.,positive
Giang Kiều Duyên e thách c lm đc cái này🤡,negative
"Tôi đã yêu thơ NGUYỄN BÍNH từ nhỏ ; Thơ ông dung dị , đời thường mà sâu thẳm đọc lên như những câu ca dao !!!",neutral
"Tròng trành bến nước bên hoa ""bến lay đã cạn hoa trải ngàn sa""",neutral
thơ hay qua' <3,positive
"Trời kia có thấu cho chăng
Yêu em không nói chỉ nằng đi vòng
Ngày kia nàng đã có chồng
Tiếc công qua lại chống mông mà gào",neutral
Đọc mấy bài của cụ hay thật,positive
"quá hay, cách dùng từ rất mộc và giàu liên tưởng...",positive
"""Đường gần tôi  cứ đi vòng cho xa..."" :)))) hậu bối bọn cháu Cx tiếp nối phát huy từ các cụ",neutral
mấy con cá dưới đó ..,neutral
Tong Hoang Gia Bao,neutral
Fact: Xưa còn bé t tưởng ống thở ra khói màu hồng thật lớn ms bt đoa là edit,neutral
Làm xong ko muốn nó bể tiếc,neutral
Bà ơi cho xin bí quyết trẻ đẹp như vậy đi,neutral
Mấy con cá phê pha😇,neutral
Qua dinh,neutral
```

3.1.1 Logistic Regression

Tham số mô hình:

- solver: 'liblinear' để tối ưu hóa mô hình
- C: 1.0 (hệ số điều chỉnh regularization)
- max_iter: 100 (số vòng lặp tối đa)

Dữ liệu huấn luyện:

- Vector hóa: Sử dụng TfidfVectorizer với 5000 từ đặc trưng
- Kích thước tập huấn luyện và kiểm tra: Chia theo tỷ lệ 80:20
- Tập huấn luyện: 80% bình luận đã được gán nhãn
- Tập kiểm tra: 20% bình luận đã được gán nhãn để đánh giá mô hình

3.1.2 Naive Bayes

Tham số mô hình:

- alpha: 1.0 (hệ số điều chỉnh Laplace smoothing)
- fit_prior: True (ước lượng xác suất tiên nghiệm)

Dữ liệu huấn luyện:

- Vector hóa: Sử dụng CountVectorizer
- Kích thước tập huấn luyện và kiểm tra: Chia theo tỷ lệ 80:20
- Tập huấn luyện: 80% bình luận đã được gán nhãn
- Tập kiểm tra: 20% bình luận đã được gán nhãn để đánh giá mô hình

3.1.3 Support Vector Machine (SVM)

Tham số mô hình:

- kernel: 'linear' (sử dụng hàm kernel tuyến tính)
- C: 1.0 (hệ số điều chỉnh regularization)

Dữ liệu huấn luyện:

- Vector hóa: Sử dụng CountVectorizer
- Kích thước tập huấn luyện và kiểm tra: Chia theo tỷ lệ 80:20
- Tập huấn luyện: 80% bình luận đã được gán nhãn
- Tập kiểm tra: 20% bình luận đã được gán nhãn để đánh giá mô hình

3.1.4 Long Short-Term Memory (LSTM)

Tham số mô hình:

- Embedding Layer: input_dim=5000, output_dim=64, input_length=100
- LSTM Layer 1: 64 units, return_sequences=True
- Dropout Layer 1: rate=0.2 (giảm thiểu overfitting)
- LSTM Layer 2: 64 units
- Dropout Layer 2: rate=0.2
- Dense Layer: 32 units, activation='relu'
- Output Layer: 4 units, activation='softmax'

Dữ liệu huấn luyện:

- Tokenize: Sử dụng Tokenizer với 5000 từ đặc trưng
- Padding: Pad sequences to maxlen=100
- Kích thước tập huấn luyện và kiểm tra: Chia theo tỷ lệ 80:20
- Tập huấn luyện: 80% bình luận đã được gán nhãn
- Tập kiểm tra: 20% bình luận đã được gán nhãn để đánh giá mô hình
- Epochs: 10 (số vòng huấn luyện)

- Batch size: 32 hoặc 64 (số mẫu huấn luyện trong mỗi bước cập nhật trọng số)

3.2 Kết Quả

3.2.2 Naive bayes

3.2.2.1 Classification Report

Kết quả báo cáo phân loại sau khi training với dữ liệu demo3

	Precision	Recall	F1-score	Support
Negative	0.68	0.78	0.73	46
Neutral	0.73	0.82	0.77	60
Positive	0.62	0.44	0.52	36
Toxic	0.33	0.14	0.20	7
Accuracy			0.68	149
Macro avg	0.59	0.55	0.55	149
Weighted avg	0.67	0.68	0.67	149

Precision: Độ chính xác - tỷ lệ dự đoán chính xác trong tổng số dự đoán của mô hình.

- negative: 0.68
- neutral: 0.73
- positive: 0.62
- toxic: 0.33
- **macro avg**: 0.59 (trung bình cộng của các giá trị precision)
- **weighted avg**: 0.67 (trung bình gia quyền của các giá trị precision)

Recall: Khả năng hồi phục - tỷ lệ dự đoán chính xác trong tổng số mẫu của từng nhãn.

- negative: 0.78
- neutral: 0.82
- positive: 0.44
- toxic: 0.14
- **macro avg**: 0.55 (trung bình cộng của các giá trị recall)
- **weighted avg**: 0.68 (trung bình gia quyền của các giá trị recall)

F1-score: Trung bình điều hòa của precision và recall.

- negative: 0.73
- neutral: 0.77
- positive: 0.52
- toxic: 0.20
- **macro avg**: 0.55 (trung bình cộng của các giá trị f1-score)
- **weighted avg**: 0.67 (trung bình gia quyền của các giá trị f1-score)

Support: Số lượng mẫu của từng nhãn trong tập kiểm tra.

- negative: 46
- neutral: 60
- positive: 36
- toxic: 7

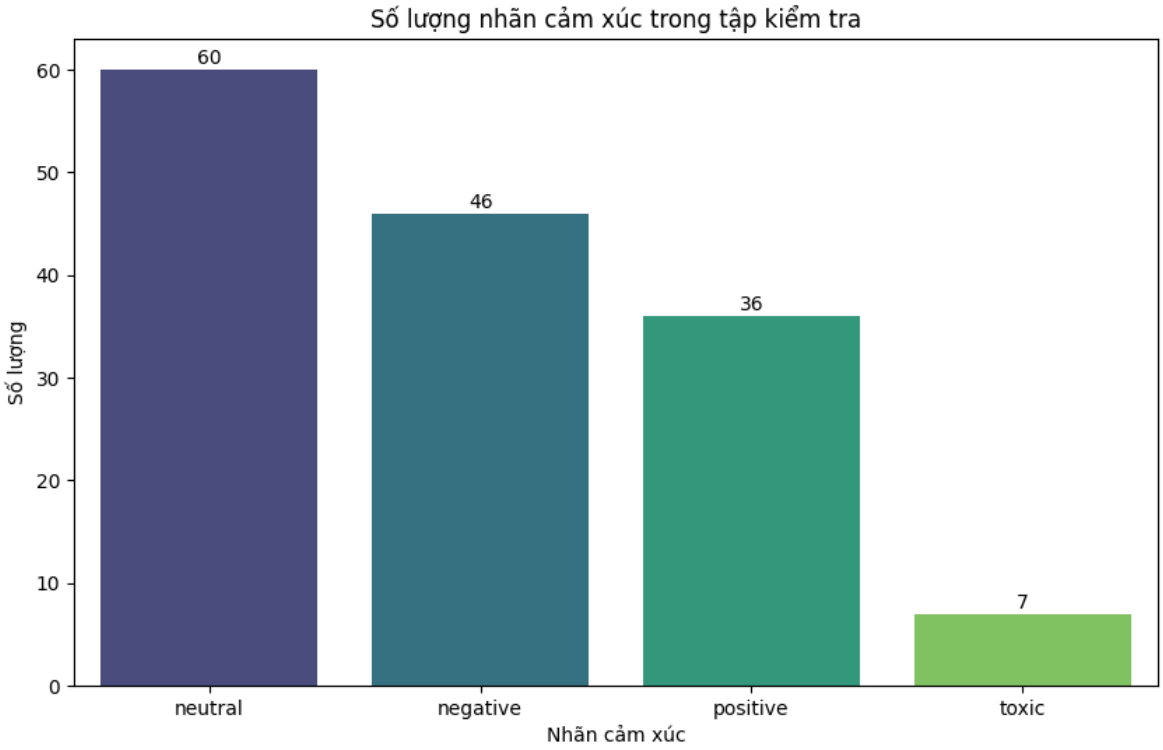
Accuracy: Độ chính xác tổng thể của mô hình trên tập kiểm tra.

- **Accuracy**: 0.68 (mô hình đạt độ chính xác 68%)

Nhận xét:

- Mô hình Naive Bayes đạt độ chính xác 68%.
- Precision và recall của nhãn neutral và negative cao, cho thấy mô hình dự đoán tốt hai nhãn này.
- Precision và recall của nhãn positive và toxic thấp, đặc biệt là nhãn toxic. Nguyên nhân có thể do số lượng mẫu của nhãn toxic ít hơn các nhãn khác.

3.2.2.2 Biểu Đồ kết quả với tập kiểm tra



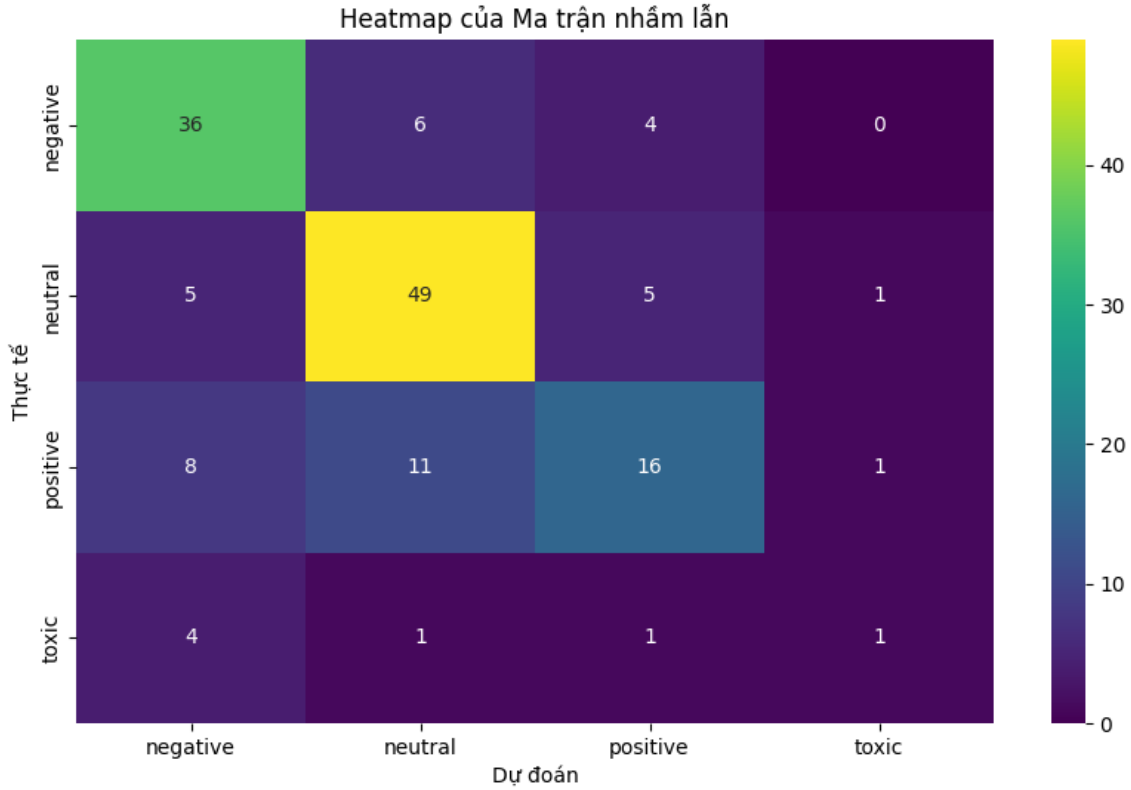
Neutral: 60 nhãn

Negative: 46 nhãn

Positive: 36 nhãn

Toxic: 7 nhãn

3.2.2.3 Confusion matrix

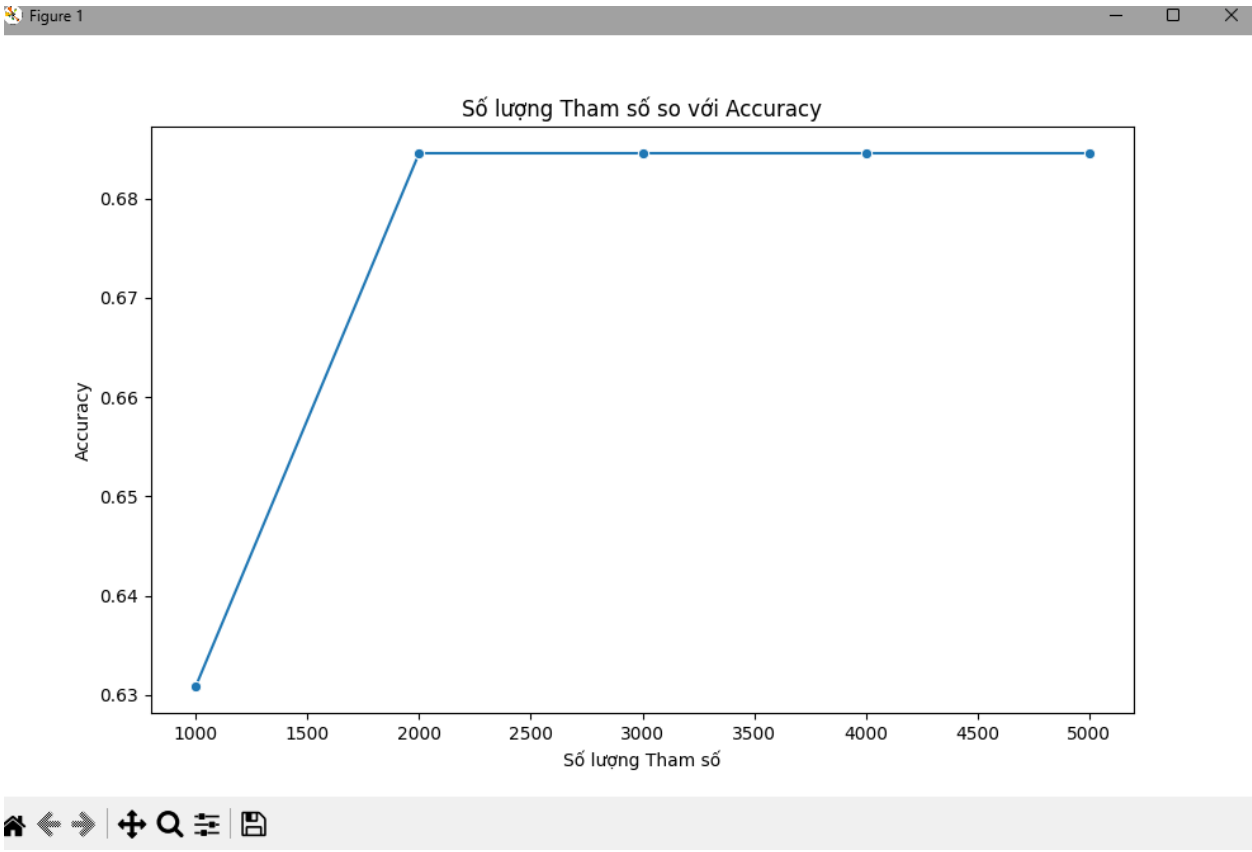


Độ chính xác cao nhất: Các giá trị trên đường chéo chính của ma trận biểu thị dự đoán chính xác. Nhãn "Neutral" có độ chính xác cao nhất với 49 dự đoán đúng.

Các nhầm lẫn phổ biến:

- Nhãn "Positive" thường bị nhầm với "Negative" (8 lần) và "Neutral" (11 lần).
- Nhãn "Toxic" có ít mẫu nhưng nhiều nhầm lẫn với "Negative" (4 lần).

3.2.2.4 Số lượng tham số so với Accuracy



- **Các điểm chính:**

Trục hoành (x-axis): Số lượng tham số

Trục tung (y-axis): Độ chính xác

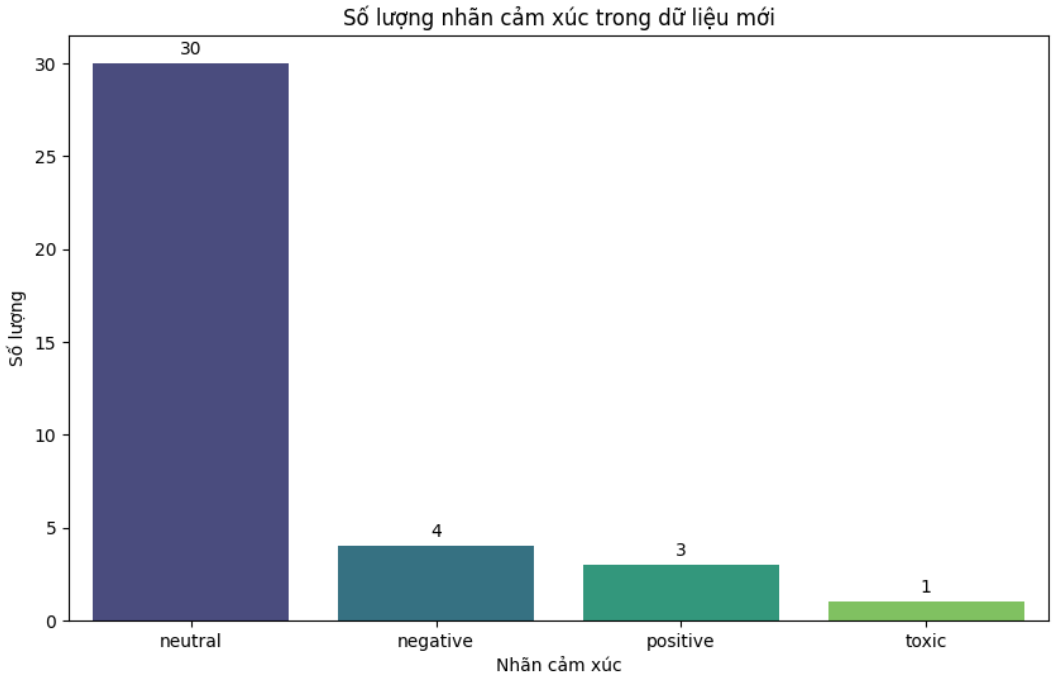
Giá trị và xu hướng:

- **1000 Tham số**: Độ chính xác ~0.63
- **2000 Tham số**: Độ chính xác ~0.68
- **3000 Tham số**: Độ chính xác duy trì ~0.68
- **4000 Tham số**: Độ chính xác duy trì ~0.68
- **5000 Tham số**: Độ chính xác duy trì ~0.68

Nhận xét:

- **Cải thiện ban đầu**: Tăng đáng kể từ 0.63 lên 0.68 khi tham số tăng từ 1000 lên 2000.
- **Bão hòa**: Độ chính xác dừng tăng ở mức 0.68 sau 2000 tham số, dù số lượng tham số tiếp tục tăng.

3.2.2.5 Kết quả kiểm tra dữ liệu mới



1 số input và output sau khi kiểm tra dữ liệu mới

Input	Output
Tôi rất thích dịch vụ này, Tuyệt vời	Neutral
Không tệ nhưng cũng không quá tốt	Neutral
Dịch vụ quá tồi tệ tôi không hài lòng chút nào	Negative
Thật không ngờ sản phẩm tuyệt vời đến vậy	Positive
Chất lượng kém không đáng tiền	Positive
Không có gì đặc biệt	Neutral
Rất tệ	Neutral
G2 bắn như cc	Negative

3.2.2.6 Nhận xét

Mô hình Naive Bayes đạt độ chính xác tổng thể 68%, hoạt động tốt đặc biệt với các nhãn "Neutral" và "Negative". Tuy nhiên, mô hình gặp khó khăn trong việc dự đoán nhãn "Positive" và "Toxic" do số lượng mẫu ít và nhầm lẫn giữa các nhãn. Việc lựa chọn mô hình Naive Bayes là hợp lý cho các bài toán cần triển khai nhanh và yêu cầu tài nguyên tính toán thấp.

3.2.3 Logistic regression

3.2.3.1 Classification Report

	Precision	Recall	F1-score	Support
Negative	0.86	0.51	0.64	37
Neutral	0.64	1.00	0.78	75
Positive	1.00	0.32	0.49	28
Toxic	1.00	0.11	0.20	9
Accuracy			0.7	149
Macro avg	0.88	0.49	0.53	149
Weighted avg	0.79	0.7	0.66	149

Precision: Độ chính xác - tỷ lệ dự đoán chính xác trong tổng số dự đoán của mô hình.

- negative: 0.86
- neutral: 0.64
- positive: 1.00
- toxic: 1.00
- **macro avg**: 0.88 (trung bình cộng của các giá trị precision)
- **weighted avg**: 0.79 (trung bình gia quyền của các giá trị precision)

Recall: Khả năng hồi phục - tỷ lệ dự đoán chính xác trong tổng số mẫu của từng nhãn.

  - negative: 0.51
  - neutral: 1.00
  - positive: 0.32
  - toxic: 0.11
  - **macro avg**: 0.49 (trung bình cộng của các giá trị recall)
  - **weighted avg**: 0.70 (trung bình gia quyền của các giá trị recall)

F1-score: Trung bình điều hòa của precision và recall.

  - negative: 0.64
  - neutral: 0.78
  - positive: 0.49
  - toxic: 0.20

- **macro avg**: 0.53 (trung bình cộng của các giá trị f1-score)
- **weighted avg**: 0.66 (trung bình gia quyền của các giá trị f1-score)

Support: Số lượng mẫu của từng nhãn trong tập kiểm tra.

- negative: 37
- neutral: 75
- positive: 28
- toxic: 9

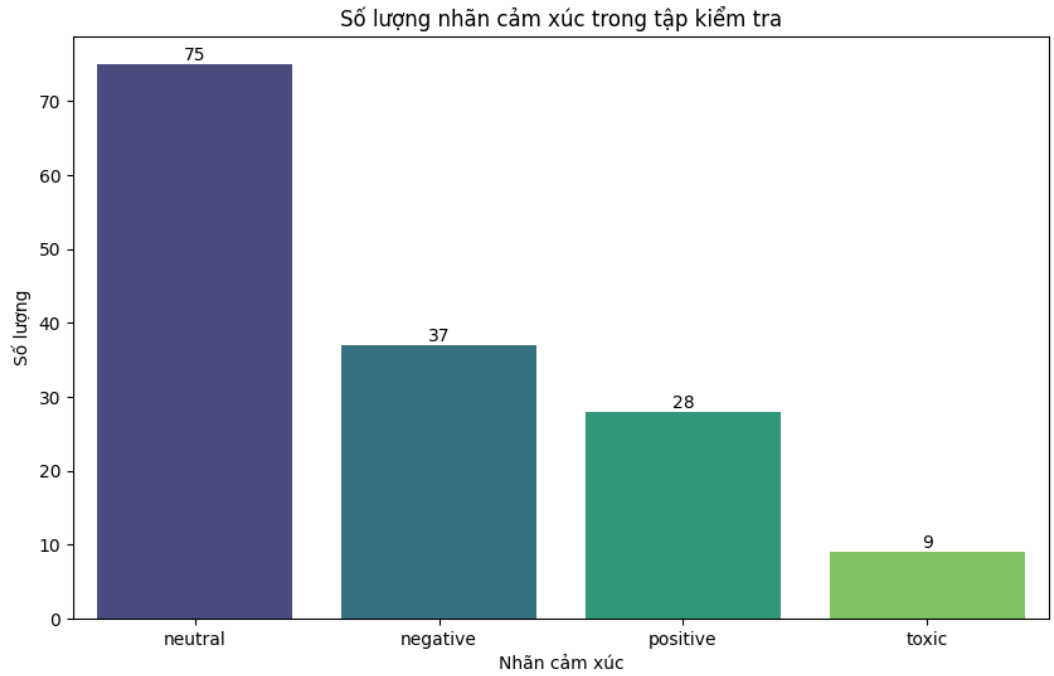
Accuracy: Độ chính xác tổng thể của mô hình trên tập kiểm tra.

- **Accuracy**: 0.70 (mô hình đạt độ chính xác 70%)

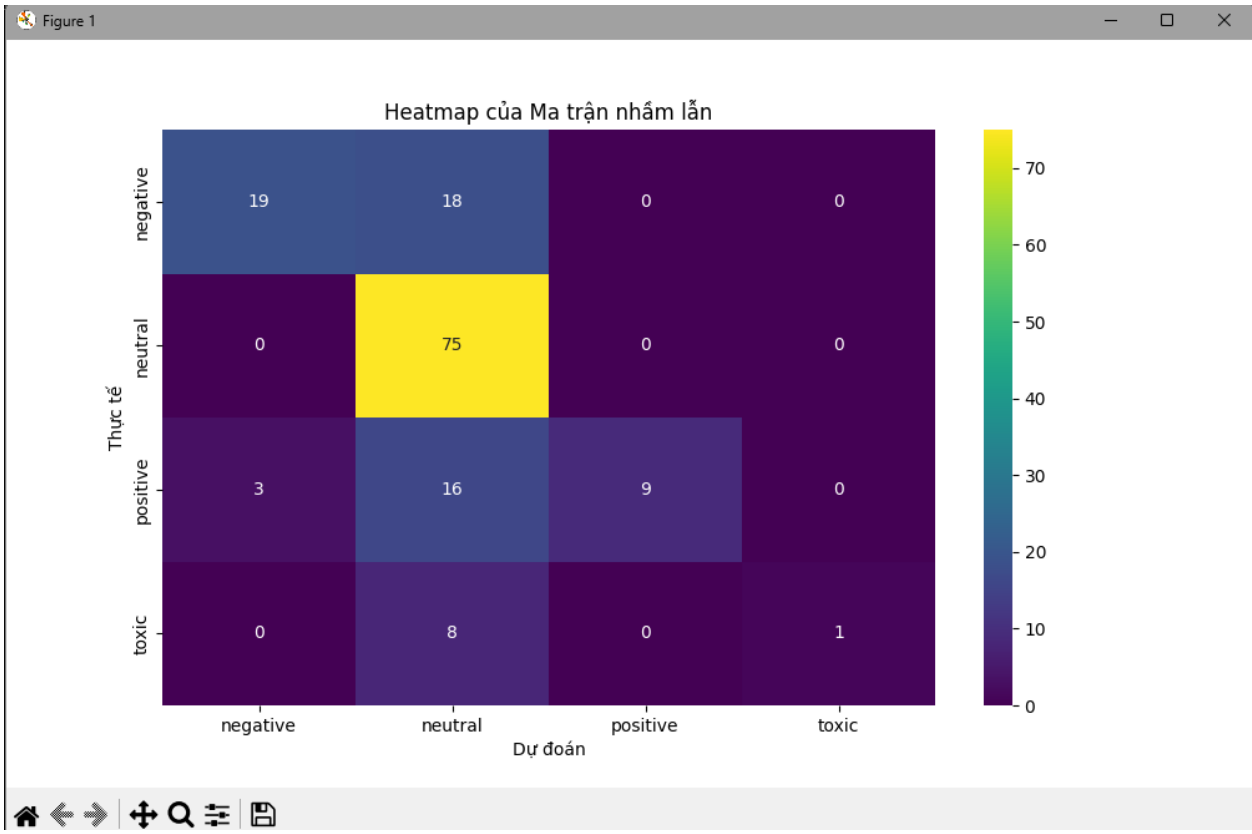
Nhận xét:

- Mô hình Logistic Regression đạt được độ chính xác tổng thể là 70%,
- Precision và recall của nhãn neutral là cao nhất với precision là 0.64 và recall là 1.00, cho thấy mô hình dự đoán tốt nhất cho nhãn này.
- Precision của nhãn positive và toxic là 1.00, cho thấy mô hình ít nhầm lẫn khi dự đoán nhãn này, nhưng recall thấp (0.32 cho positive và 0.11 cho toxic), cho thấy mô hình bỏ sót nhiều mẫu thực tế thuộc các nhãn này.
- F1-score của các nhãn positive và toxic thấp hơn, cho thấy cần cải thiện để dự đoán tốt hơn các nhãn này.

3.2.3.2 Biểu Đồ kết quả với tập kiểm tra



3.2.3.3 Confusion matrix

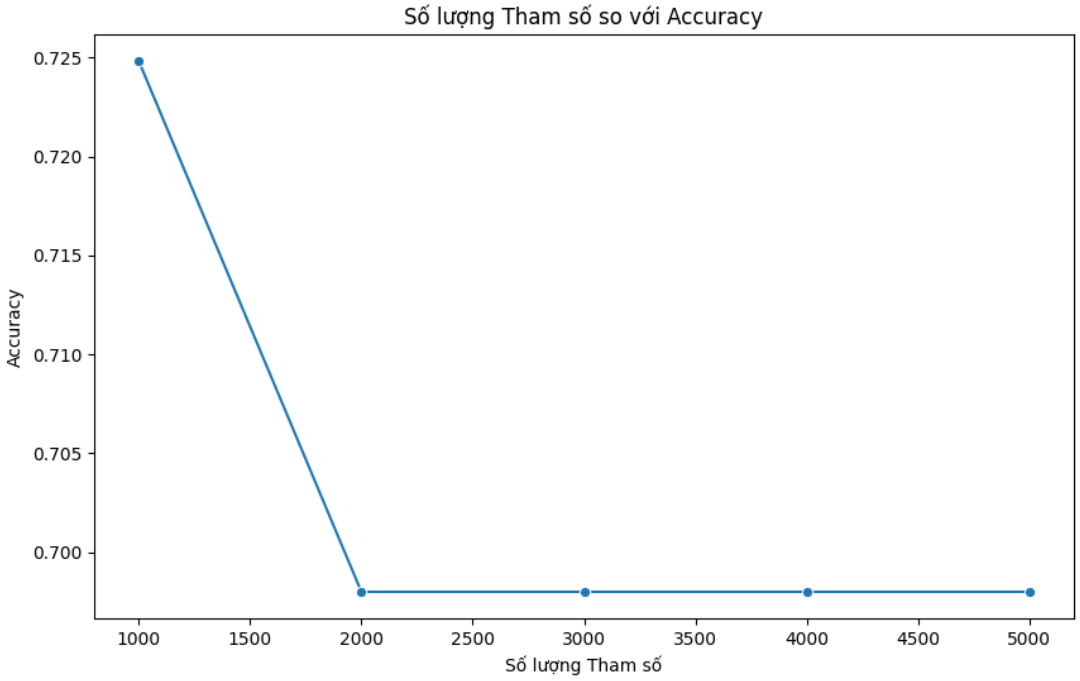


Độ chính xác cao nhất: Các giá trị trên đường chéo chính của ma trận đại diện cho các dự đoán chính xác. Nhãn "Neutral" có độ chính xác cao nhất với 75 dự đoán đúng, cho thấy Logistic Regression dự đoán tốt nhất cho nhãn này.

Các nhầm lẫn phổ biến:

- Nhãn "Negative" nhầm lẫn với "Neutral" (18 lần).
- Nhãn "Positive" nhầm lẫn với "Neutral" (16 lần).
- Nhãn "Toxic" nhầm lẫn với "Neutral" (8 lần).

3.2.3.4 Số Lượng tham số so với Accuracy



- **Các điểm chính:**

  Trục hoành (x-axis): Số lượng tham số (Số lượng Tham số)

  Trục tung (y-axis): Độ chính xác (Accuracy)
- **Các giá trị và xu hướng:**

  1000 Tham số: Độ chính xác khoảng 0.725.

  2000 Tham số: Độ chính xác giảm xuống còn 0.700.

  3000 Tham số: Độ chính xác duy trì ở mức 0.700.

  4000 Tham số: Độ chính xác duy trì ở mức 0.700.

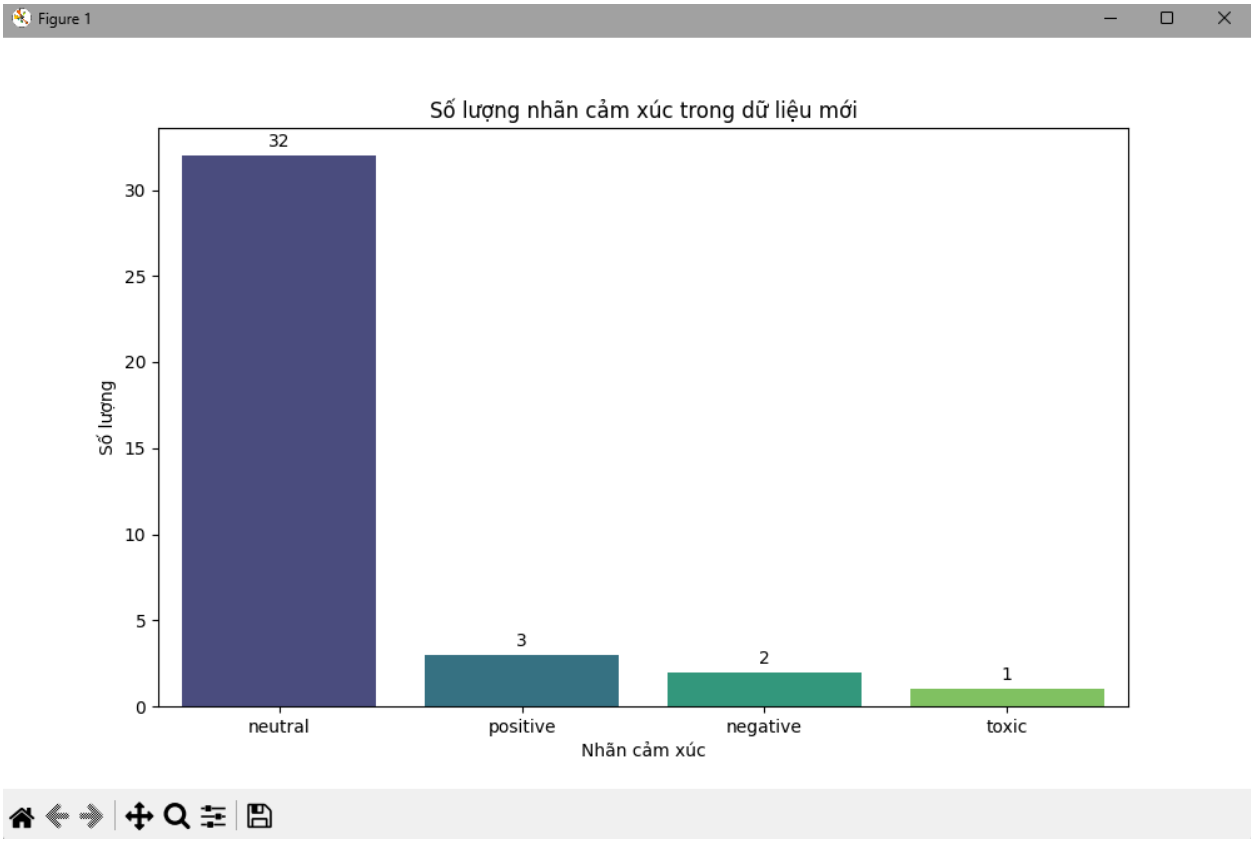
  5000 Tham số: Độ chính xác duy trì ở mức 0.700.

Nhận xét:

Giảm độ chính xác ban đầu: Độ chính xác giảm từ 0.725 xuống 0.700 khi tham số tăng từ 1000 lên 2000.

Trạng thái bão hòa: Độ chính xác giữ nguyên ở mức 0.700 sau 2000 tham số, cho thấy thêm tham số không cải thiện mô hình.

3.2.3.5 Kết quả kiểm tra dữ liệu mới



1 số input và output sau khi kiểm tra dữ liệu mới

Input	Output
Tôi rất thích dịch vụ này, Tuyệt vời	Positive
Không tệ nhưng cũng không quá tốt	Neutral
Dịch vụ quá tồi tệ tôi không hài lòng chút nào	Neutral
Thật không ngờ sản phẩm tuyệt vời đến vậy	Positive
Chất lượng kém không đáng tiền	Neutral
Không có gì đặc biệt	Neutral
Rất tệ	Neutral
G2 bắn như cc	Neutral

3.2.3.6 Nhận xét

Mô hình Logistic Regression đạt được kết quả tốt với độ chính xác tổng thể 70%, đặc biệt tốt với các nhãn "Neutral" và "Negative". Tuy nhiên, mô hình gặp khó khăn trong việc dự đoán nhãn "Positive" và "Toxic" do số lượng mẫu ít và nhầm lẫn giữa các nhãn. Logistic Regression là lựa chọn tốt cho các bài toán cần triển khai nhanh và yêu cầu tài nguyên tính toán thấp.

3.2.4 SVM

3.2.4.1 Classification Report

	Precision	Recall	F1-score	Support
Negative	0.88	0.5	0.64	46
Neutral	0.57	0.92	0.7	60
Positive	0.75	0.5	0.6	36
Toxic	1.00	0.29	0.44	7
Accuracy			0.66	149
Macro avg	0.8	0.55	0.6	149
Weighted avg	0.73	0.66	0.65	149

1. **Precision**: Độ chính xác - tỷ lệ dự đoán chính xác trong tổng số dự đoán của mô hình.
    - negative: 0.88
    - neutral: 0.57
    - positive: 0.75
    - toxic: 1.00
    - **macro avg**: 0.80 (trung bình cộng của các giá trị precision)
    - **weighted avg**: 0.73 (trung bình gia quyền của các giá trị precision)
2. **Recall**: Khả năng hồi phục - tỷ lệ dự đoán chính xác trong tổng số mẫu của từng nhãn.
    - negative: 0.50
    - neutral: 0.92
    - positive: 0.50

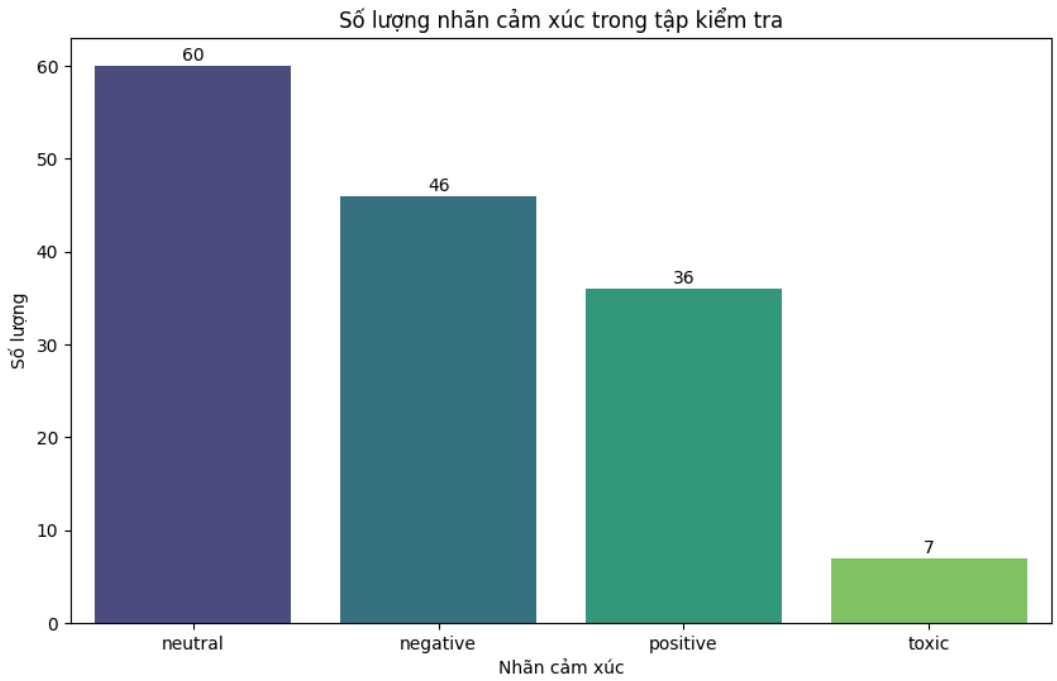
- toxic: 0.29
- **macro avg**: 0.55 (trung bình cộng của các giá trị recall)
- **weighted avg**: 0.66 (trung bình gia quyền của các giá trị recall)

3. **F1-score**: Trung bình điều hòa của precision và recall.
   - negative: 0.64
   - neutral: 0.70
   - positive: 0.60
   - toxic: 0.44
   - **macro avg**: 0.60 (trung bình cộng của các giá trị f1-score)
   - **weighted avg**: 0.65 (trung bình gia quyền của các giá trị f1-score)
4. **Support**: Số lượng mẫu của từng nhãn trong tập kiểm tra.
   - negative: 46
   - neutral: 60
   - positive: 36
   - toxic: 7
5. **Accuracy**: Độ chính xác tổng thể của mô hình trên tập kiểm tra.
   - **Accuracy**: 0.66 (mô hình đạt độ chính xác 66%)

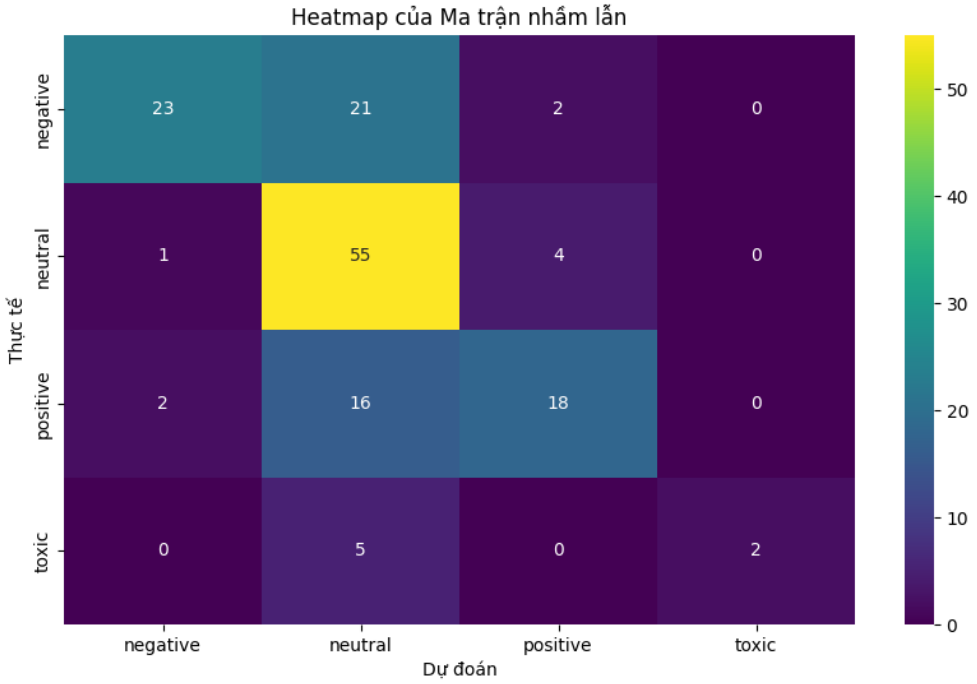
6. Nhận xét:

- Mô hình SVM đạt được độ chính xác tổng thể là 66%, tương đương với Naive Bayes và thấp hơn một chút so với Logistic Regression.
- Precision cao nhất cho nhãn toxic (1.00) nhưng recall rất thấp (0.29), cho thấy mô hình dự đoán chính xác nhãn này khi nó nhận diện được, nhưng bỏ sót nhiều bình luận thuộc nhãn này.
- Nhãn neutral có recall rất cao (0.92), cho thấy mô hình có khả năng nhận diện tốt các bình luận trung lập, nhưng precision thấp hơn (0.57).
- F1-score cao nhất cho nhãn neutral (0.70), cho thấy mô hình hoạt động tốt nhất với nhãn này.

3.2.4.2 Biểu đồ kết quả



3.2.4.3 Confusion matrix



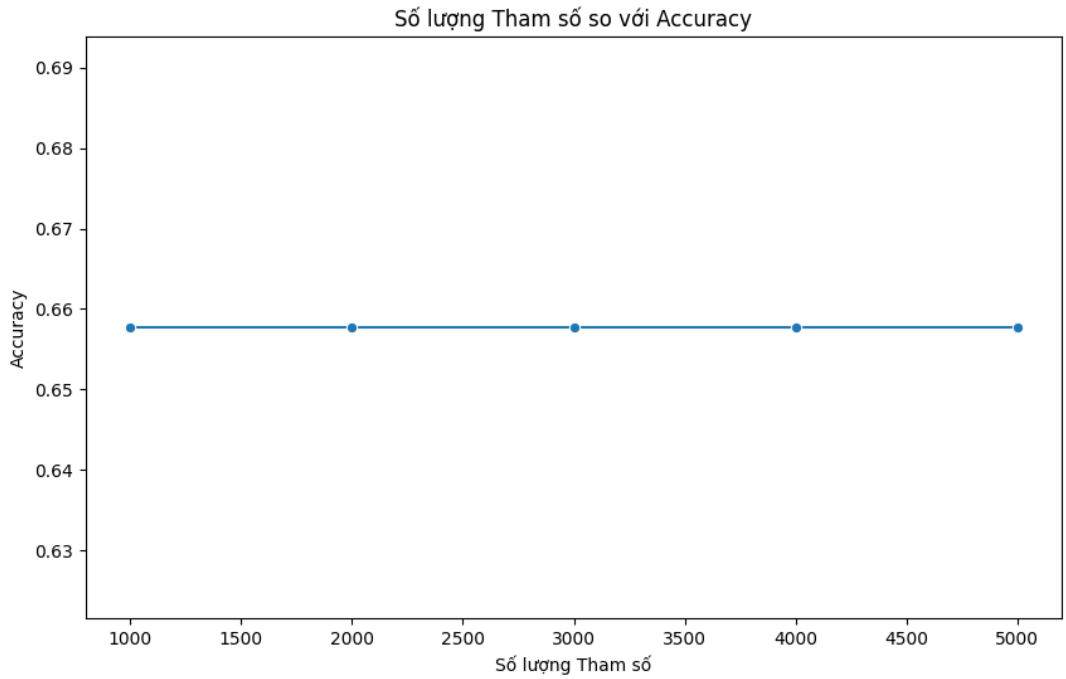
Độ chính xác cao nhất: Các giá trị dọc theo đường chéo chính của ma trận (từ trên trái xuống dưới phải) đại diện cho các dự đoán chính xác. Giá trị

cao nhất là cho nhãn "Neutral" với 55 dự đoán chính xác, cho thấy mô hình SVM dự đoán tốt nhất cho nhãn này.

Các nhầm lẫn phổ biến:

- Nhãn "Negative" có nhiều mẫu bị nhầm lẫn với "Neutral" (21).
- Nhãn "Positive" có số lượng mẫu nhầm lẫn đáng kể với "Neutral" (16).
- Nhãn "Toxic" có 5 mẫu bị nhầm lẫn với "Neutral".

3.2.4.4 Số lượng tham số so với Accuracy



Các điểm chính:

Trục hoành (x-axis): Số lượng tham số (Số lượng Tham số), từ 1000 đến 5000.

Trục tung (y-axis): Độ chính xác (Accuracy), từ 0.63 đến 0.69.

Các giá trị và xu hướng:

Độ chính xác của mô hình duy trì ở mức khoảng 0.66 cho tất cả các số lượng tham số (1000, 2000, 3000, 4000, và 5000).

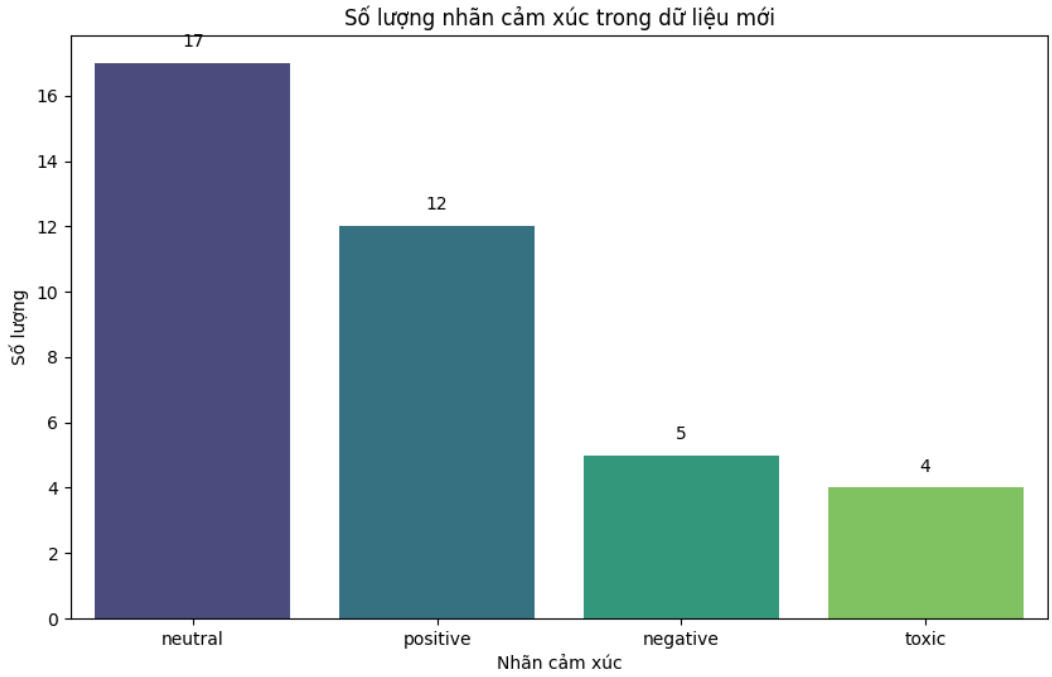
Nhận xét:

Trạng thái bão hòa: Biểu đồ cho thấy rằng độ chính xác của mô hình SVM không thay đổi nhiều khi số lượng tham số tăng từ 1000 lên 5000. Điều này cho

thấy rằng việc tăng thêm số lượng tham số không giúp cải thiện độ chính xác của mô hình SVM trong khoảng giá trị này.

Hiệu quả ổn định: Mô hình SVM duy trì một hiệu suất ổn định với độ chính xác ở mức khoảng 0.66 bất kể số lượng tham số.

3.2.4.5 Kết Quả kiểm tra



1 số input và output sau khi kiểm tra dữ liệu mới

Input	Output
Tôi rất thích dịch vụ này, Tuyệt vời	Positive
Không tệ nhưng cũng không quá tốt	Negative
Dịch vụ quá tồi tệ tôi không hài lòng chút nào	Negative
Thật không ngờ sản phẩm tuyệt vời đến vậy	Positive
Chất lượng kém không đáng tiền	Negative
Không có gì đặc biệt	Neutral
Rất tệ	Neutral
G2 bắn như cc	Toxic

3.2.4.6 Nhận xét

Mô hình SVM đạt được kết quả tốt với độ chính xác tổng thể 66%, đặc biệt tốt với nhãn "Neutral". Tuy nhiên, mô hình gặp khó khăn trong việc dự đoán nhãn "Positive" và "Toxic" do số lượng mẫu ít và nhầm lẫn giữa các nhãn. SVM là lựa chọn tốt cho các bài toán có dữ liệu phức tạp và yêu cầu phân tách phi tuyến tính.

3.2.5 LSTM

3.2.5.1 Classification Report

	Precision	Recall	F1-score	Support
Negative	0.35	0.68	0.47	25
Neutral	1	0.31	0.48	32
Positive	0.59	0.67	0.62	36
Toxic	0	0	0	6
Accuracy			0.52	99
Macro avg	0.48	0.41	0.39	99
Weighted avg	0.63	0.52	0.5	99

1. **Precision**: Độ chính xác - tỷ lệ dự đoán chính xác trong tổng số dự đoán của mô hình.
    - positive: 0.35
    - neutral: 1.00
    - negative: 0.59
    - toxic: 0.00
    - **macro avg**: 0.48 (trung bình cộng của các giá trị precision)
    - **weighted avg**: 0.63 (trung bình gia quyền của các giá trị precision)
2. **Recall**: Khả năng hồi phục - tỷ lệ dự đoán chính xác trong tổng số mẫu của từng nhãn.
    - positive: 0.68
    - neutral: 0.31
    - negative: 0.67
    - toxic: 0.00

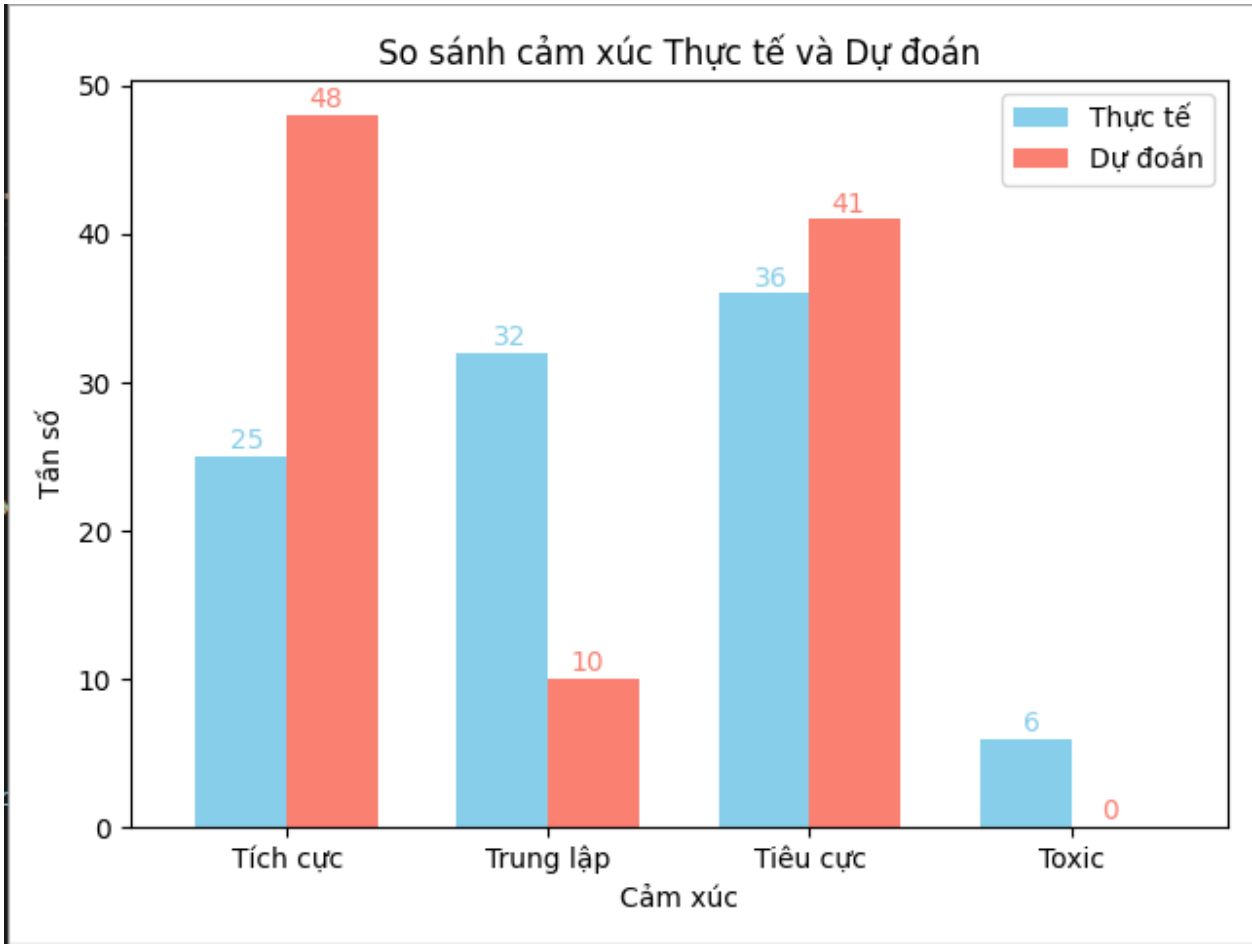
- **macro avg**: 0.41 (trung bình cộng của các giá trị recall)
- **weighted avg**: 0.52 (trung bình gia quyền của các giá trị recall)

3. **F1-score**: Trung bình điều hòa của precision và recall.
    - positive: 0.47
    - neutral: 0.48
    - negative: 0.62
    - toxic: 0.00
    - **macro avg**: 0.39 (trung bình cộng của các giá trị f1-score)
    - **weighted avg**: 0.50 (trung bình gia quyền của các giá trị f1-score)
4. **Support**: Số lượng mẫu của từng nhãn trong tập kiểm tra.
    - positive: 25
    - neutral: 32
    - negative: 36
    - toxic: 6
5. **Accuracy**: Độ chính xác tổng thể của mô hình trên tập kiểm tra.
    - **Accuracy**: 0.52 (mô hình đạt độ chính xác 52%)

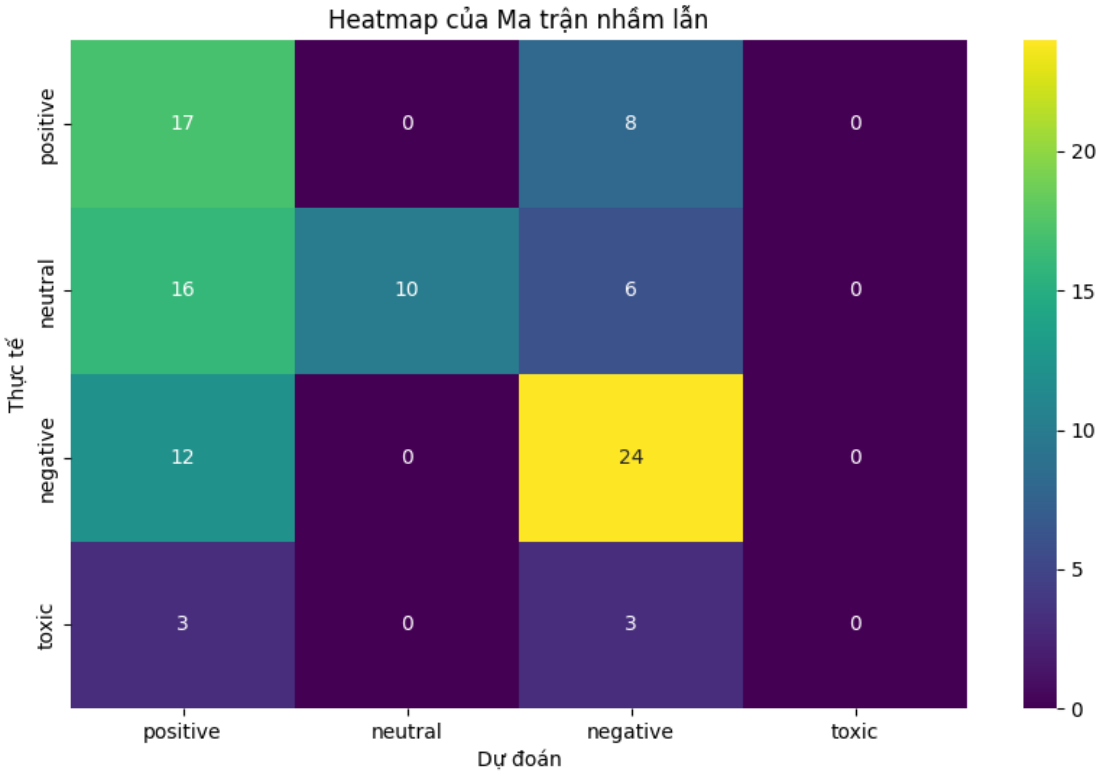
6. Nhận xét:

- Mô hình LSTM đạt được độ chính xác tổng thể là 52%,
- Precision và recall của nhãn neutral là cao nhất với precision là 1.00 nhưng recall chỉ là 0.31, cho thấy mô hình dự đoán chính xác khi nhận diện nhãn này nhưng bỏ sót nhiều bình luận trung lập.
- Nhãn toxic có precision và recall là 0.00, cho thấy mô hình không thể nhận diện chính xác nhãn này.
- F1-score cao nhất cho nhãn negative (0.62), cho thấy mô hình hoạt động tốt nhất với nhãn này.

3.2.5.2 Biểu đồ kết quả với tập kiểm tra



3.2.5.3 Confusion matrix

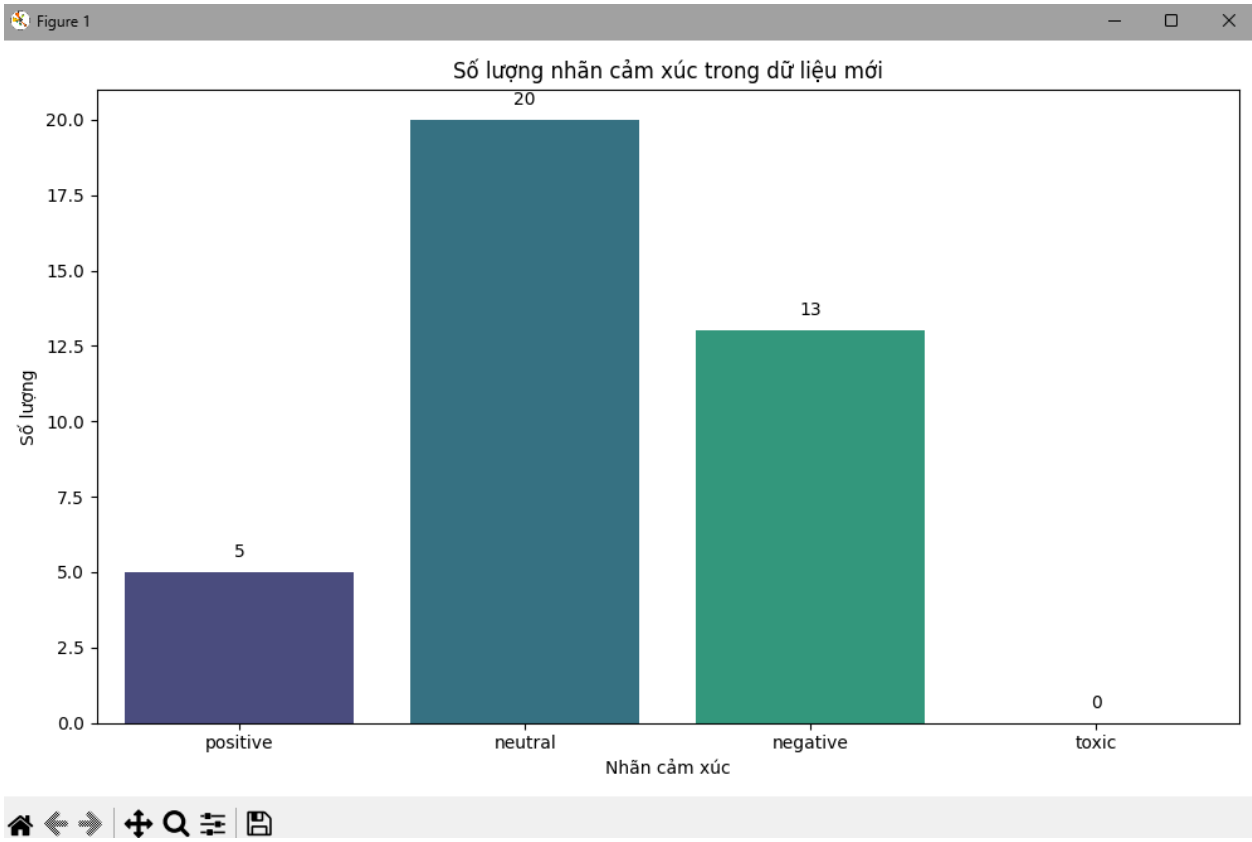


Độ chính xác cao nhất: Các giá trị dọc theo đường chéo chính của ma trận (từ trên trái xuống dưới phải) đại diện cho các dự đoán chính xác. Giá trị cao nhất là cho nhãn "Negative" với 24 dự đoán chính xác, cho thấy mô hình LSTM dự đoán tốt nhất cho nhãn này.

Các nhầm lẫn phổ biến:

- Nhãn "Positive" có nhiều mẫu bị nhầm lẫn với "Negative" (8).
- Nhãn "Neutral" có nhiều mẫu nhầm lẫn với "Positive" (16) và "Negative" (6).
- Nhãn "Toxic" có số lượng mẫu rất ít và bị nhầm lẫn nhiều với "Positive" (3) và "Negative" (3).

3.2.5.4 Kết quả kiểm tra



1 số input và output sau khi kiểm tra dữ liệu mới

Input	Output
Tôi rất thích dịch vụ này, Tuyệt vời	Positive
Không tệ nhưng cũng không quá tốt	Neutral
Dịch vụ quá tồi tệ tôi không hài lòng chút nào	Negative
Thật không ngờ sản phẩm tuyệt vời đến vậy	Positive
Chất lượng kém không đáng tiền	Positive
Không có gì đặc biệt	Neutral
Rất tệ	Neutral
G2 bắn như cc	Negative

3.2.5.5 Nhận xét

Mô hình LSTM đạt được kết quả thấp với độ chính xác tổng thể 52%, đặc biệt khó khăn trong việc dự đoán nhãn "Toxic". Mô hình hoạt động tốt nhất với nhãn "Negative" nhưng gặp nhiều nhầm lẫn giữa các nhãn. LSTM là lựa chọn tốt cho các bài toán yêu cầu xử lý dữ liệu chuỗi và học ngữ cảnh dài hạn, nhưng yêu cầu tài nguyên tính toán cao và không phải là lựa chọn tốt nhất cho bài toán phân loại cảm xúc bình luận với dữ liệu hiện tại.

3.3 Kết Luận

Trong quá trình thực nghiệm, nhóm đã đánh giá hiệu suất của các mô hình Logistic Regression, Naive Bayes, SVM và LSTM trong việc phân loại cảm xúc từ các bình luận trên mạng xã hội. Kết quả cho thấy mỗi mô hình có ưu và nhược điểm riêng, phù hợp với các đặc điểm và yêu cầu khác nhau của bài toán.

Logistic Regression và **SVM** cho thấy độ chính xác tổng thể cao hơn so với các mô hình khác, cụ thể là với nhãn "Neutral" và "Negative". Logistic Regression đạt độ chính xác 70%, thể hiện khả năng phân loại ổn định và hiệu quả, trong khi SVM đạt độ chính xác 66%, mạnh mẽ trong việc xử lý dữ liệu phức tạp nhưng có xu hướng nhầm lẫn giữa các nhãn.

Naive Bayes cũng đạt được kết quả đáng kể với độ chính xác 68%, dễ triển khai và nhanh chóng trong thực thi. Tuy nhiên, mô hình này gặp khó khăn với các nhãn có ít mẫu, như "Toxic", dẫn đến hiệu suất thấp hơn trong việc dự đoán nhãn này.

Mô hình **LSTM**, mặc dù có khả năng xử lý dữ liệu chuỗi và học ngữ cảnh dài hạn, nhưng trong thực nghiệm này chỉ đạt được độ chính xác 52%. Điều này cho thấy rằng LSTM yêu cầu nhiều tài nguyên tính toán và không phải là lựa chọn tốt nhất cho bài toán hiện tại với dữ liệu hiện có.

Tổng quan, kết quả thực nghiệm cung cấp cái nhìn toàn diện về hiệu suất của các mô hình phân loại cảm xúc. Việc lựa chọn mô hình phù hợp nên dựa trên yêu cầu cụ thể của bài toán, cân nhắc giữa độ chính xác, tốc độ triển khai và khả năng xử lý dữ liệu phức tạp. Các kết quả và phân tích này sẽ giúp định hướng cho các bước tiếp theo trong việc cải thiện và áp dụng các mô hình phân loại cảm xúc trong thực tế.

Tổng kết

- **Tầm quan trọng của phân loại cảm xúc bình luận trên mạng xã hội**

Trong thời đại số hóa, mạng xã hội đã trở thành nền tảng chính để con người chia sẻ ý kiến và bày tỏ cảm xúc. Việc phân loại cảm xúc từ các bình luận không chỉ giúp doanh nghiệp hiểu rõ hơn về tâm lý khách hàng mà còn đóng vai trò quan trọng trong nhiều lĩnh vực khác như marketing, chăm sóc khách hàng và nghiên cứu xã hội. Việc phát hiện các ngôn từ mang tính độc hại còn góp phần xây dựng môi trường giao tiếp trực tuyến lành mạnh.

- **Quy trình thực hiện**

Nhóm chúng em đã tiến hành nghiên cứu và triển khai hệ thống phân tích cảm xúc từ các bình luận trên mạng xã hội bằng cách sử dụng các mô hình học máy tiên tiến gồm Logistic Regression, Naive Bayes, SVM và LSTM. Quy trình thực hiện gồm các bước từ chuẩn bị và làm sạch dữ liệu, gán nhãn, tiền xử lý, huấn luyện mô hình, đánh giá hiệu suất, tối ưu và triển khai mô hình.

- **Các mô hình học máy và hiệu suất**
- **Naive Bayes**: Hoạt động tốt với dữ liệu nhỏ và đơn giản nhưng gặp khó khăn với dữ liệu phức tạp và không độc lập.
    - Ưu điểm: Đơn giản, nhanh chóng, hiệu quả với dữ liệu nhỏ.
    - Nhược điểm: Giả định các đặc trưng độc lập, hiệu suất thấp với dữ liệu phức tạp.
- **Logistic Regression**: Đơn giản, hiệu quả với các bài toán phân loại tuyến tính nhưng có hạn chế với dữ liệu phi tuyến tính.
    - Ưu điểm: Dễ hiểu, hiệu quả, không yêu cầu điều chỉnh nhiều.
    - Nhược điểm: Giới hạn trong phân loại tuyến tính, hiệu suất thấp với dữ liệu phức tạp.
- **SVM**: Mạnh mẽ với dữ liệu phức tạp, giảm thiểu overfitting nhưng yêu cầu tài nguyên tính toán cao và khó tối ưu hóa.
    - Ưu điểm: Mạnh mẽ, giảm thiểu overfitting, hiệu quả với dữ liệu nhiều chiều.
    - Nhược điểm: Yêu cầu tài nguyên tính toán cao, khó tối ưu hóa.
- **LSTM**: Khả năng học ngữ cảnh dài hạn, hiệu quả trong các bài toán chuỗi thời gian nhưng đòi hỏi nhiều tài nguyên tính toán và thời gian huấn luyện.

- Ưu điểm: Ghi nhớ thông tin lâu dài, hiệu quả trong các bài toán tuần tự.
- Nhược điểm: Yêu cầu tài nguyên tính toán cao, phức tạp.

- **Kết quả thực nghiệm**

Trong các thử nghiệm thực tế, các mô hình Logistic Regression và SVM cho thấy độ chính xác cao hơn so với Naive Bayes và LSTM. Logistic Regression đạt độ chính xác 70%, SVM đạt 66%, trong khi Naive Bayes đạt 68% và LSTM đạt 52%. Các chỉ số đánh giá như precision, recall và F1-score cũng được tính toán để cung cấp cái nhìn chi tiết về hiệu suất của từng mô hình.

- **Đóng góp và ứng dụng thực tiễn**

Kết quả nghiên cứu không chỉ giúp chúng em hiểu rõ hơn về các phương pháp học máy mà còn mang lại những ứng dụng thực tiễn quan trọng. Hệ thống phân loại cảm xúc và phát hiện ngôn từ độc hại này có thể được áp dụng để cải thiện dịch vụ khách hàng, tối ưu chiến lược marketing và xây dựng môi trường trực tuyến an toàn và lành mạnh hơn.