Phân loại cảm xúc của các bình luận dựa trên bộ dữ liệu British Airline

Trung Le-Chi Phan, Man Nguyen Tran
Khanh Quoc Tran, Anh Tuan-Nguyen Gia
University of Information Technology, Ho Chi Minh City, Vietnam
Vietnam National University, Ho Chi Minh City, Vietnam

Abstract

Sentiment-analysis là một trong các tác vụ chính của **Xử lý ngôn ngữ tự nhiên (NLP)**. Bài báo cáo này, với hơn 3000 dòng dữ liệu của bộ dữ liệu British Arirline, sẽ tập trung vào việc xác định và nhận diện các yếu tố cảm xúc dựa trên 1 câu văn đã được cho trước. Bằng việc sử dụng các **kỹ thuật tiền xử lý dữ liệu**, **mô hình pre-trained** và các mô hình **Machine Learning** để thực hiện việc phân tích và phân loại các bình luận thành tích cực, tiêu cực và trung bình. Sau khi đã phân tích kết quả cho thấy rằng cảm xúc chủ yếu sẽ là cảm giác tiêu cực về các vấn đề của chuyến bay như delay, dịch vụ khách hàng, ... Bài báo cáo này sẽ tập trung phân tích dữ liệu và nghiên cứu về việc áp dụng các mô hình máy học, cũng như các hướng phát triển trong tương lai

1 Giới thiệu

Phản hồi của khách hàng về cảm nhận đối với dịch vụ đang ngày càng trở nên quan trọng và trở thành mối quan tâm hàng đầu của các doanh nghiệp đối với thị trường. Sự phát triển của truyền thông và xã hội đã khiến cho việc tiếp cận và trình bày cảm nhận của bản thân đối với một tổ chức dễ dàng hơn bao giờ hết. Vì vậy, phân tích cảm xúc trở thành một công cụ quan trọng để doanh nghiệp phân tích phản hồi của khách hàng, có thông tin bổ ích về cảm xúc của khách hàng và cải thiện trải nghiệm khách hàng. Trong bài viết này, nhóm tiếp cận vấn đề trên với việc nghiên cứu phân loại cảm xúc khách hàng và áp dụng các kỹ thuật tiền xử lý

2 Bộ dữ liệu

Bộ dữ liệu được nhóm chọn từ trên Kaggle để sử dụng và nghiên cứu cho môn học này. Các thông tin về bộ dữ liệu như sau:

2.1 Nguồn gốc

- Tên bộ dữ liệu: Airways customer data
- Người đóng góp: GHASSEN KHALED
- Nguồn: Kaggle
- Link: https://www.kaggle.com/ghassenkhaled

Bộ dữ liệu được tạo ra nhằm phục vụ cho mục đích xây dựng một mô hình dự đoán để có thể tìm ra được những nhân tố ảnh hưởng đến hành vi mua vé của khách hàng. Bộ dữ liệu đã được thu thập bằng phương pháp web scraping, sử dụng công cụ BeautifulSoup. Bộ dữ liệu bao gồm 3412 điểm dữ liệu. Mỗi điểm dữ liệu bao gồm 4 thuộc tính (mô tả trong bảng sau):

Column Name	Meaning
reviews	Một đoạn phản hồi của khách hàng về hàng hàng không British Airway
rates	Điểm số được khách hàng đánh giá thông qua đoạn bình luận
date	Ngày khách hàng đăng bình luận
country	Quốc tịch của người đăng bình luận

2.2 Phân tích bộ dữ liệu

- **Stt** : không được sắp xếp tốt, gồm 3412 dòng nhưng giá trị lớn nhất là 3417.
- **Reviews**, gồm 3 dạng chính :
  - **Trip Verified** : Trip Verified | Excellent service both on the ground and on board - while their first class product might not be leading edge, the service around the entire experience was well worth the cost of the ticket. The Concorde room at Heathrow is well managed and a great place to relax prior to the trip. Flight was slightly delayed out of Heathrow but arrived on schedule in Johannesburg.
  - **Not Verified** : Not Verified | London to Cairo. First, on this 5 hour mid morning flight the only complimentary food and

drink were a tiny bag of pretzels and a small bottle of water. Even Southwest is more generous. When unable to connect my phone to order food, I hit the FA call button with no response for more than an hour. When the FA came to collect garbage I had to show him the call light and he gruffly asked me what was the matter. He used his phone to place the order.

– **Normal**: CPH-LHR-CPH October 2014. Air travel just keeps getting better. The latest boon on BA is the mobile phone boarding pass app. Wonderful. No more searching for a printer making life so much easier. Well done BA!

- **Rates**: Quan sát trên hình 1, ta nhận thấy được phân phối của số các điểm dữ liệu có xu hướng lệch phải. Trong đó giá trị xuất hiện nhiều nhất là 1.

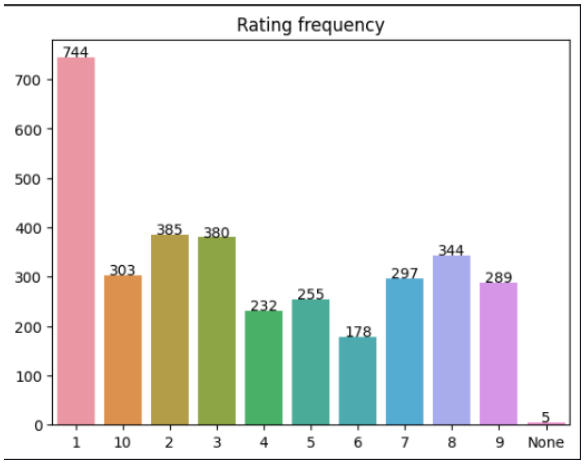

Figure 1: Biểu đồ cột về các mức rating

- **Date** : Thu thập được trải dài từ ngày 18/5/2014 đến 24/1/2023

- **Country** : Trong thuộc tính tồn tại 2 điểm dữ liệu rỗng. Các điểm dữ liệu còn lại mang giá trải dài trên 69 quốc gia bao gồm: United States, Canada, United Kingdom, ...

3 Tạo bộ nhãn cho bộ dữ liệu

3.1 Quy trình gán nhãn dữ liệu

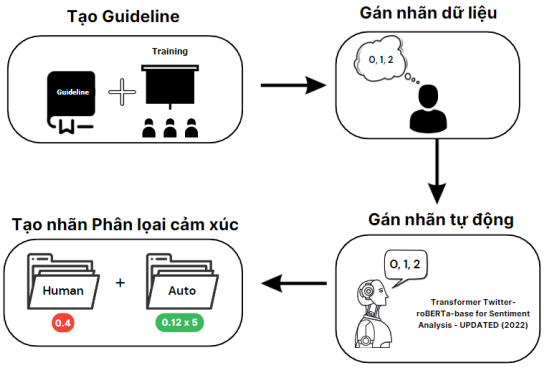

Figure 2: Quy trình gán nhãn dữ liệu

Với mục đích tận dụng bộ dữ liệu để phát triển bài toán phân loại bình luận khách hàng. Nhóm quyết định thực hiện gán nhãn "Label" (biểu thị cảm xúc của bình luận khách hàng) trên bô dữ liệu ban đầu. Công việc gán nhãn được thực hiển bởi 3 thành viên có đủ trình độ nhận thức cảm xúc. Quy trình gán nhãn sẽ được mô tả theo biểu đồ sau:

- Bước 1: Ở bước này, dữ liệu sẽ được gán nhãn bởi con người. Nhóm sẽ thảo luận và thống nhất về Guideline gán nhãn cho dữ liệu. Tiếp theo, nhóm sẽ thực hiện việc phổ biến guideline cho toàn bộ các thành viên tham gia gán nhãn.

- Bước 2: Các thành viên sẽ bắt đầu công việc gán nhãn, sau khi hoàn tất nhiệm vụ gán nhãn, nhóm sẽ kiểm tra lại chất lượng của các nhãn đã được gán để đảm bảo không có sai sót của con người xuất hiện.

- Bước 3: Sau khi hoàn thành việc gán và kiểm tra các nhãn, nhóm tiến hành tính toán độ đồng thuận dựa trên F1-score (macro).Đồng thời, nhóm tiến hành tạo nhãn tự động thông qua việc sử dụng mô hình Twitter-roBERTa-base để kết hợp với các nhãn đã được gán để tạo ra một loại nhãn sau cùng.

- Bước 4: Nhóm tiến hành công việc tính toán tỉ lệ giữa các nhãn do người gán và máy gán. Trong đó trọng số do người đánh sẽ được đặt nặng hơn. Sau khi tính toán xong nhóm sẽ đưa ra bộ nhãn cuối cùng và hoàn tất quy trình gán nhãn.

3.2 Guideline gán nhãn

Do mỗi con người khi đọc bình luận sẽ có cảm nhận khác nhau. Chính vì vậy, để đảm bảo tính thực tế cũng như đặt ra một thử thách khi tạo một bộ dữ liệu có thể giúp máy tính học được như suy nghĩ con người. Nhóm quyết định các nhãn sẽ được gán dựa trên cảm xúc của cá nhân mỗi người khi đọc.Tuy nhiên, với có các trường hợp cảm xúc quá rõ ràng thì annotators phải chú ý theo hướng dẫn. Guideline của nhóm bao gồm các ý chính sau:

- Các annotators sẽ đọc các bình luận và dựa trên cảm xúc cá nhân để gán nhãn cho bình luận đó. Các nhãn được gán sẽ có giá trị:

Label	Value
0	Tiêu cực
1	Tích cực
2	Trung tính

- Những câu mang tính tuyệt đối được gán nhãn như sau:
  - "Tôi sẽ tránh BA", "Tôi hoàn toàn thất vọng bởi dịch vụ của BA", "BA xứng đáng nhận điểm dưới mức trung bình"...: nhãn 0.
  - "Tôi hoàn toàn hài lòng với dịch vụ của BA", "Trải nghiệm tốt với BA": nhãn 1.
- Những câu mang tính chung chung, nhưng có thiên hướng về đánh giá về một phía tích cực hoặc tiêu cực:
  - "Tôi thấy dịch vụ khá tốt, nhân viên thân thiện và niềm nở, đồ ăn được bày ra rất thẩm mỹ, tôi có một giấc ngủ ngon nhưng có một điểm trừ nho nhỏ là nhạc phát từ máy phát thanh khá lâu đời, không hợp với thời đại chung": nhãn 1.
  - "Khá buồn vì tôi đã rất tin tưởng BA nhưng chất lượng hiện tại lại giảm sút khá nhiều, đồ ăn bị nguội, nước uống thì không được cung cấp nếu không yêu cầu mà trong khi lại bay ở chuyến bay dài, khoang máy bay có vẻ không được vệ sinh kỹ, thậm chí tôi còn quẹt phải bụi! Nhưng điểm cứu vớt được là tiếp viên lại rất ân thân thiện và niềm nở, nói chung tôi không thể đổ lỗi cho họ được": nhãn 0.
- Những câu mang tính chung chung, khen chê đều, kể chuyện như:
  - "Tôi thấy BA làm đồ ăn khá ngon, nhưng máy bay khá cũ kĩ": nhãn 2
  - "Tôi chấm BA ở mức điểm trung bình": nhãn 2
  - "Chuyến bay bị trì hoãn, khi lên máy bay thì trong khoang khá trang trọng, đồ ăn ngon, nhân viên thân thiện nhưng đồ uống thì không được tiếp dù đi chuyến bay dài, nhạc phát trên đài khá nhỏ và rè, nói chung cũng bình thường": nhãn 2
- Các bạn tham gia gán nhãn sẽ thực việc gắn nhãn trên nên tảng Google Sheet. Toàn bộ quá trị gán nhãn sẽ được các bạn chọn thông qua một list lựa chọn cố định để loại bỏ khả năng nhập sai định dạng, sai chính tả,...

Annotator_id	F1_score
0	0.698
1	0.736
2	0.714

Table 1: F1-score của các người gán nhãn

3.3 Tính độ đồng thuận

Sau khi các bạn hoàn thành gán nhãn, nhóm sẽ chọn ra ngẫu nhiên 100 điểm dữ liệu đã được gán nhãn để làm tập ground truth. Tập ground truth này sau đó sẽ được xóa nhãn và gửi đến tất cả các thành viên để gán nhãn. Các giá trị Label các thành viên gán sau đó sẽ được so sánh với tập ground truth ban đầu và tính độ đồng thuận. Nhóm sử dụng chỉ số F1-score(macro) [6.1.1].

4 Xử lý dữ liệu:

4.1 Các vấn đề của bộ dữ liệu:

Sau khi hoàn thành quy trình gán nhãn. Nhóm tiến hành quan sát và nhận thấy mốt số sai sót tồn đọng trong bộ dữ liệu:

- Trong thuộc tính "Reviews" có tồn tại các câu bình luận trùng lặp nội dung với nhau (9 điểm dữ liệu).
- Trong các câu bình luận có sự xuất hiện của lỗi chính tả.
- Thuộc tính "Rates" không phản ánh đúng với sự hài lòng của khách hàng với dịch vụ.
- Bộ dữ liệu mất cân bằng (lệch về nhãn 0).

4.2 Cách khắc phục

4.2.1 Các bình luận trùng nhau

Do số lượng câu bị trung lập khá nhỏ so với quy mô bộ dữ liệu (chiếm 9/3412) và không có tác động mạnh đến dữ liệu. Chính vì vậy đối với vấn đề này nhóm sẽ trực tiếp loại bỏ các điểm dữ liệu trùng lặp.

4.2.2 Sửa lỗi chính tả

Nhóm sử dụng thư viện gingerit đã có sẵn để tiến hành việc sửa lỗi chính tả cho toàn bộ dữ liệu. Tuy nhiên trong bộ dữ liệu tồn tại một số lưỡng rất lớn các từ ngữ viết tắt và các từ ngữ mang trá trị biểu đạt. Ví dụ: “BA” – “British Airline”, “A370” – “Máy bay hiệu A370”,... Các từ này sẽ được thư viện gingerit nhận xét là các từ sai chính tả và sửa lỗi, gây ra sự sai lệch thông tin. Do đó, nhóm sẽ tiến hành cho gingerit bỏ qua các từ ngữ mang tính viết tắt.

4.2.3 Thuộc tính "Rates" không phản ứng chính xác giá trị

Theo như biểu đồ trên, ta có thể thấy sộ lượng điểm dữ liệu mà thuộc tính rates không phản ứng chính xác trên nhãn 0 (nhãn 0 nhưng rates lớn hơn 5) chiếm tỉ lệ xx, trên nhãn 1 (nhãn 1 nhưng rates bé hơn 4) chiếm xx. Tổng thể dố lượng điểm sai lệch chiếm xx. Do số lượng thuộc tính rates sai lệch chiếm tỉ lệ khá cao sẽ gây nhiễu và làm ảnh hưởng đến bộ dữ liệu, nhóm quyết định sẽ không sử dụng và loại bỏ thuộc tính này.

4.2.4 Mất cân bằng dữ liệu

Hiện thực ngày nay, việc dữ liệu mất cân bằng là một vấn đề chắc chắn sẽ gặp phải và tồn tại trong nhiều bộ dữ liệu. Chính vì vậy, nhóm sẽ tạm thời chấp nhận sự mất cân bằng và biến nó thành một thách thức cho quy trình sử dụng và huấn luyện mô hình.

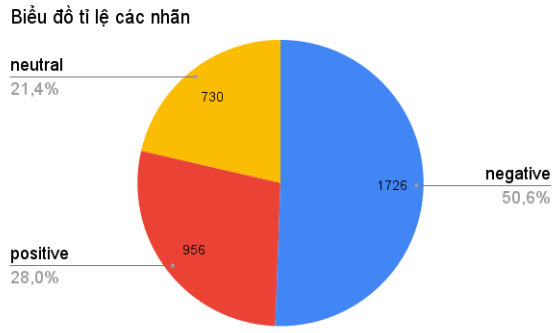

Figure 3: Biểu đồ tỉ lệ các nhãn trong bộ dữ liệu

4.3 Biến đổi các thuộc tính của dữ liệu

4.3.1 Tạo thêm thuộc tính Status

Như đã trình bày ở phần III-B, ta thấy các câu bình luận được phân thành ba loại khác nhau (Trip Verified, Not Verified, Normal). Tuy nhiên phần phân loại này là một phần của các câu bình luận chứ chưa trở thành một thuộc tính. Chính vì vậy, nhóm tách các phần phân loại câu bình luận thành một thuộc tính Status sẽ bao gồm các biến phân loại mang các giá trị về tình trạng của câu bình luận.

4.3.2 Xử lý các dữ liệu Date time

Trong bộ dữ liệu, ta có một thuộc tính "Date" ghi nhận ngày khách hàng đã viết bình luận dưới dạng: "24th January 2023". Nhóm sẽ tạo thêm 3 thuộc tính mới từ thuộc tính này bao gồm: "Day", "Month", "Year". Đồng thời, thực hiện chuẩn hóa thuộc tính "Date" về lại thành dạng chuẩn như sau: "MM/DD/YYYY"

5 Mô Hình

Ở bài toán này, đầu vào của mô hình sẽ là một câu bình luận của hãng hàng không Anh. Câu bình luận sẽ đi qua bước tiền xử lý dữ liệu và được trích xuất đặc trưng (sử dụng TfidfVectorizer để mã hóa thành một ma trận A[MxN]). Trong đó, M là số lượng điểm dữ liệu đưa vào huấn luyện, N là số lượng vocab tối đa được chọn lọc sau khi sử dụng TfidfVectorizer để trích xuất các đặc trưng của dữ liệu bình luận. Tiếp đến, các ma trận này sẽ trở thành đầu vào để huấn luyện các mô hình học máy và phân loại cảm xúc vào các nhãn. Tại bước này, nhóm thực hiện huấn luyện dữ liệu trên nhiều loại mô hình khác nhau và đánh giá hiệu suất, điểm mạnh, điểm yếu của từng mô hình để có thể chọn ra mô hình mang hiệu suất tốt nhất cho bài toán.

5.1 XGBoost

- Gradient boosting là một thuật toán phổ biến trong lớp thuật toán Boosting. Gradient Boost xem vấn đề tăng cường (boosting problem) là một vấn đề tối ưu hóa, trong đó nó sử dụng một hàm mất mát (loss function) và cố gắng giảm thiểu lỗi. Đây là lý do tại sao nó được gọi là Gradient boost, vì nó được lấy cảm hứng từ sự giảm dần độ dốc (gradient descent). Gradient Boost giúp cải thiện mãnh mẽ bằng việc kết hợp các mô hình yếu thành những mô hình mạnh, mà trong đó mỗi mô hình mới được huấn luyện để giảm độ lỗi của các mô

hình sử dụng trước đó. Với mỗi vòng lặp, thuật toán sẽ tính toán gradient của hàm dựa trên các dự đoán của của mô hình chung trước đó và sau đó huấn luyện trên mô hình yếu mới để cực tiểu các sự mất mát này.

- Các dự đoán của mô hình mới sau đó sẽ được thêm vào mô hình chung và quá trình sẽ được lặp lại cho đến hết vòng lặp hoặc khi gặp một điều kiện dừng nào đó. Tuy nhiên, khác với AdaBoost, Gradient Boosting khớp mô hình cơ sở mới dựa trên sai số dư (residual error) của mô hình cơ sở trước đó

- XGBoost là một thư viện đã cải tiến của Gradient Boosting được thiết kế nhằm mục đích đạt hiệu quả cao, mền dẻo hơn và nhỏ gọn hơn. Nó áp dụng các thuật toán máy học dưới bộ khung của Gradient Boosting. XGBoost cung cấp một thuật toán cải thiện cây song song, thứ sẽ giải quyết nhiều vấn đề về khoa học dữ liệu khác nhau theo một cách nhanh chóng và chính xác. Các đặc trưng khác biệt của XGBoost làm nó khác biệt so với các thuật toán Gradient Boosting khác bao gồm:
  - Các hình thức phát cây một cách thông minh
  - Tỉ lệ co cụm lại của các lá
  - Newton Boosting
  - Các thông số ngẫu nhiên khác
  - Chọn đặc trưng tự động
  - Cải thiện cấu trúc cây song song với những khoảng thưa thớt
  - Cấu trúc truy cập vào các khối một cách hiệu quả cho việc huấn luyện cây quyết định acl

5.2 Support Vector Machine

- Support Vector Machine (SVM) là một thuật toán thuộc nhóm Supervised Learning (học có giám sát) dùng để phân chia dữ liệu (classification) thành các nhóm riêng biệt (bài toán phân lớp).

- Ý tưởng cơ bản của SVM là tạo ra một siêu phẳng (hyperplane) trong không gian đa chiều, nơi mỗi điểm dữ liệu được biểu diễn bởi một vector. Siêu phẳng này tách rời các điểm dữ liệu của các lớp khác nhau sao cho khoảng cách từ các điểm tới siêu phẳng là lớn nhất. Điều này đảm bảo rằng SVM sẽ tìm ra một đường ranh giới phân chia tối ưu giữa các lớp, giúp phân loại các điểm dữ liệu mới một cách chính xác.

- Thuật toán SVM tính toán các hệ số của siêu phẳng thông qua việc tối ưu hóa một hàm mục tiêu. Hàm mục tiêu này kết hợp việc tìm ra đường ranh giới tối ưu và đảm bảo rằng các điểm dữ liệu nằm đúng vị trí. Việc tìm hiểu SVM cũng bao gồm quá trình chọn và điều chỉnh các tham số quan trọng như hàm kernel, hằng số ưu tiên và tỷ lệ mất mát (loss).

- SVM không chỉ phù hợp với dữ liệu tuyến tính, mà còn có thể áp dụng các hàm kernel khác nhau để xử lý dữ liệu phi tuyến. Hàm kernel cho phép chúng ta biến đổi không gian dữ liệu ban đầu thành một không gian mới, nơi việc phân loại trở nên dễ dàng hơn.

6 Đánh Giá

6.1 Chỉ số đánh giá

Nhóm thực hiện việc huấn luyện bốn mô hình trên 2 tập dữ liệu được gán nhãn khác nhau để có thể so sánh và đánh giá hiệu suất của các mô hình. Đối với việc đánh giá mô hình, nhóm sử dụng chỉ số accuracy và F1 (macro) để đánh giá hiệu suất mô hình.

6.1.1 F1

Chỉ số F1 đo đạt độ chính xác sử dụng Precision **p** và Recall **r**. Precision là tỉ lệ của các True Positive **tp** đến tất cả những giá trị Positive đã được dự đoán **tp + fp**. Recall là tỉ lệ giữa các True Positive **tp** đến tất cả các giá trị Positive thực tế **tp + fn** Chỉ số F1 được tính bởi công thức:

$$F1 = 2\frac{p \cdot r}{p + r}$$

Trong đó:

$$p = \frac{tp}{tp + fp}, r = \frac{tp}{tp + fn}$$

6.1.2 Accuracy

Độ chính xác (Accuracy) là tỷ lệ các trường hợp được dự báo đúng trên tổng số các trường hợp. Độ chính xác được tính bằng công thức:

$$Accuracy = \frac{TP + TN}{TotalSample}$$

6.2 Đánh giá trên tập dữ liệu

Model	F1	Accuracy
XGBoost	0.769	0.800
SVM	0.767	0.798

Table 2: Hiệu suất mô hình

7 Thử nghiệm

7.1 Cân bằng nhãn từ bộ dữ liệu

- Qua Figure 3 trên, ta có thể thấy được sự mất cân bằng trong tỉ lệ giữa các nhãn được gán. Cụ thể thì dữ liệu sẽ bị lệch về nhãn 0 (50.6%). Nhận thấy việc dữ liệu bị mất cân bằng nên nhóm quyết định thử nghiệm bằng phương pháp giảm số lượng nhãn 0 mà các mô hình được học để hạn chế các trường hợp mô hình học lệch về 1 nhãn gây ảnh hưởng đến kết quả dự đoán. Do chưa tìm được công cụ phù hợp để sinh ra thêm các điểm dữ liệu nên nhóm sẽ thử nghiệm dưới hình thức giảm số lượng nhãn 0 để cân bằng tỉ lệ với 2 nhãn còn lại.

- Có thể thấy được từ Table 3, hiệu suất tổng thể bị giảm, cả hai mô hình SVM và XGBoost bị giảm cả về Accuracy và F1, nên nhóm quyết định loại bỏ phương án thử nghiệm này để áp dụng vào tổng thể thử nghiệm dữ liệu.

7.2 Điều chỉnh các siêu tham số

Nhóm tiến hành cải thiện các mô hình bằng cách sử dụng GridSearchCV để tìm ra siêu tham số cho các mô hình để có được hiệu suất tốt nhất, Table 4 chỉ ra hiệu quả các mô hình cho ra và đã đưa ra các tham số tốt nhất cho từng mô hình, trong đó, mô hình SVM mang lại hiệu suất cao hơn so với XGBoost.

7.3 Tổng Kết Đánh Giá:

8 Phân tích lỗi:

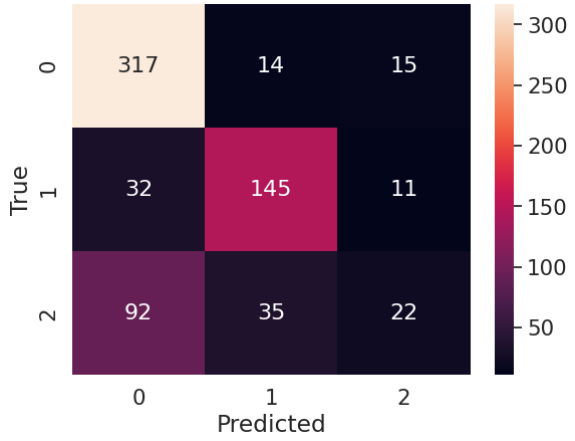

Figure 4: Ma trận nhầm lẫn

Dựa vào ma trận nhầm lẫn Figure 4, ta thấy được số lượng nhãn 0 được dự đoán đúng tương đối nhiều. Tuy nhiên số lượng nhãn 2 được dự đoán đúng là rất thấp và thường bị dự đoán nhầm thành nhãn 0. Qua đó, ta thấy được các thông số của mô hình sau khi huấn luyện đã có xu hướng dự đoán lệch về nhãn 0.

9 Hướng phát triển:

Do hiện tại bộ dữ liệu bị mất cân bằng khá nghiêm trọng dẫn đến sự chênh lệch trong mô hình. Chính vì vậy nhóm có dự định sẽ cân bằng các nhãn trong tương lai. Bên cạnh đó, bộ dữ liệu hiên tại có số lượng tương đối nhỏ, chưa đủ nhiều để có thể mang lại các hiệu quả rõ rệt việc bổ sung về kích thước bộ dữ liệu sẽ là cần thiết trong tương lai. Ngoài ra bình luận là một loại dữ liệu sẽ thay đổi theo thời gian. Ví dụ như ngôn ngữ teencode của các thế hệ sẽ khác nhau tùy thuộc vào tốc độ phát triển và trình độ phát triển của xã hội. Vì vậy để có thể đảm bảo tính cập nhật theo thời đại, nhóm có định hướng sẽ biến bộ dữ liệu trở thành kiểu dữ liệu streaming với các luồng bình luận sẽ liên tục được ghi tự đồng vào data và được lưu trữ bằng các hệ thống lưu trữ dữ liệu lớn.

Cuối cùng, với bộ dữ liệu này, nhóm cũng có thể tận dụng để sử dụng theo một góc nhìn khác để thay đổi bài toán từ phân loại thành các bài toán khác hoặc tiếp cận bằng các phương pháp học sâu để có thể mang lại hiệu quả cũng như tạo ra nhiều giá trị hơn như: tìm ra các nội dung phê bình trong bình luân để áp dụng thực tế vào việc cải tiến chất

Models	Accuracy	F1-score (macro)	F1-score (weighted)
SVM	0.688	0.604	0.639
XGBoost	0.664	0.614	0.641

Table 3: Kết quả việc cân bằng lại nhãn của bộ dữ liệu

Model	Parameters	Accuracy	F1-macro	F1-weighted
SVM	C=10, gamma = 'scale', kernel = 'rbf'	0.740	0.624	0.709
XGB	learning_rate=0.15, n_estimators= 100, objective = 'multi:softprob'	0.728	0.612	0.698

Table 4: Điều chỉnh tham số cho các mô hình

lượng của doanh nghiệp.

10 Kết luận

Trong bài nghiên cứu này, nhóm đã đánh giá hiệu suất của các mô hình học máy khác nhau trong việc phân loại các nhãn cảm xúc dựa trên các bình luận của khách hàng trên bộ dữ liệu British Airways Reviews.

Nhóm đã áp dụng kỹ thuật tiền xử lý dữ liệu và tạo thêm các nhãn gồm nhãn gán tay, nhãn gán bằng máy (kết hợp giữa 5 nhãn gán máy từ các comment của original, unicode, stopword, stemming) – từ mô hình **Twitter-roBERTa-base for Sentiment Analysis - UPDATED** và **TfidfVectorizer** chuẩn hóa các câu văn bản thành vector đặc trưng dùng để huấn luyện các mô hình.

References

[1] Francesco Barbieri, Jose Camacho-Collados, Leonardo Neves, and Luis Espinosa-Anke. Tweeteval: Unified benchmark and comparative evaluation for tweet classification. arXiv preprint arXiv:2010.12421, 2020.

[2] Steven Bird, Ewan Klein, and Edward Loper. Natural language processing with Python: analyzing text with the natural language toolkit. " O'Reilly Media, Inc.", 2009.

[3] Jose Camacho-Collados and Mohammad Taher Pilehvar. On the role of text preprocessing in neural network architectures: An evaluation study on text categorization and sentiment analysis. arXiv preprint arXiv:1707.01780, 2017.

[4] Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, pages 785–794, New York, NY, USA, 2016. ACM.

[5] Bijoyan Das and Sarit Chakraborty. An improved text sentiment classification model using tf-idf and next word negation, 2018.

[6] Rufael Fekadu, Anteneh Getachew, Yishak Tadele, Nuredin Ali, and Israel Goytom. Machine learning models evaluation and feature impor- tance analysis on npl dataset, 2022.

[7] Sachin Kumar and Mikhail Zymbler. A machine learning approach to analyze customer satisfaction from airline tweets. Journal of Big Data, 6(1):1–16, 2019.

[8] Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, and Jose Camacho-Collados. Timelms: Diachronic language models from twitter. arXiv preprint arXiv:2202.03829, 2022.

[9] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vander- plas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duch- esnay. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830, 2011.

[10] Sabu M. Thampi, Erol Gelenbe, Mohammed Atiquzzaman, Vipin Chaudhary, and Kuan-Ching Li, editors. Advances in Computing and Network Communications. Springer Singapore, 2021.