ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC MÁY TÍNH

ĐỒ ÁN CUỐI KÌ

PHÂN TÍCH CẢM XÚC BÌNH LUẬN ĐÁNH GIÁ SẢN PHẨM TRÊN SÀN THƯƠNG MẠI ĐIỆN TỬ

MÔN: XỬ LÝ NGÔN NGỮ TỰ NHIÊN

Lớp: CS211.N21

GVHD: Nguyễn Trọng Chỉnh
GVTH: Đặng Văn Thìn

Nhóm sinh viên thực hiện

STT	Tên	MSSV
1	Phạm Lê Thành Phát	21521262
2	Lê Tuấn Đạt	21520699

TP. Hồ Chí Minh, 07/2023

LỜI CẢM ƠN

Lời đầu tiên, nhóm xin gửi lời cảm ơn chân thành đến quý Thầy Cô Trường Đại học Công nghệ thông tin – Đại học Quốc gia TP.HCM và Thầy Cô khoa Khoa học máy tính vì đã cung cấp cho nhóm rất nhiều những kiến thức cơ bản làm nền tảng để thực hiện đồ án môn Xử lí ngôn ngữ tự nhiên lần này.

Đặc biệt, nhóm xin gửi lời cảm ơn đến Thầy Nguyễn Trọng Chỉnh (Giảng viên hướng dẫn) và Thầy Đặng Văn Thìn (Giảng viên phụ trách thực hành). Các Thầy đã dạy chúng em từng kiến thức một, chỉ bảo trực tiếp, hướng dẫn tận tình, sửa chữa và đóng góp nhiều ý kiến quý báu giúp nhóm hoàn thành tốt đồ án của mình.

Xuất phát từ mục đích học tập môn Xử lý ngôn ngữ tự nhiên, cũng như tìm hiểu thêm về việc ứng dụng các bài toán về xử lí ngôn ngữ tự nhiên trong cuộc sống, nhóm chúng em đã thực hiện đồ án về "Phân tích cảm xúc của bình luận Tiếng Việt". Trong quá trình thực hiện đồ án, nhóm chúng em đã vận dụng những kiến thức nền tảng đã tích lũy dựa trên những kiến thức được thầy cung cấp, đồng thời kết hợp với việc học hỏi và nghiên cứu những kiến thức mới từ thầy cô, bạn bè cũng như nhiều nguồn tài liệu tham khảo, nhóm đã cố gắng thực hiện đồ án một cách tốt nhất. Tuy nhiên, vì kiến thức chuyên môn còn hạn chế và bản thân còn thiếu nhiều kinh nghiệm thực tiễn, có thể nội dung của báo cáo không tránh khỏi những thiếu sót nhưng nó là kết quả của sự nỗ lực của các thành viên trong nhóm và sự giúp đỡ của các Thầy nên mong mọi người thông cảm và góp ý để chúng em có thể hoàn thiện hơn trong tương lai. Lời góp ý chân thành của mọi người sẽ là động lực cho nhóm em trong khoảng thời gian sinh viên sắp tới, có thêm nhiều năng lượng và hết mình với học tập và nghiên cứu.

Một lần nữa xin gửi đến các Thầy lời cảm ơn chân thành và tốt đẹp nhất!

MỤC LỤC

I. GIỚI THIỆU BÀI TOÁN

Trong những năm gần đây, sự phát triển của các nền tảng thương mại điện tử đã cách mạng hóa cách mọi người mua sắm. Các nền tảng này cung cấp một cách thuận tiện và dễ tiếp cận cho người tiêu dùng mua sản phẩm và dịch vụ trực tuyến. Với sự tăng trưởng về cả mặt số lượng người dùng lẫn nội dung trên các nền tảng hiện nay, thì đánh giá sản phẩm đóng một vai trò rất lớn trong việc đánh giá chất lượng, cảm nghĩ của người mua cũng như về kinh nghiệm phát triển sản phẩm tốt hơn. Với sự phát triển lớn mạnh công nghệ, đặc biệt là mạng xã hội, đã tạo một lượng lớn dữ liệu được tạo ra từ việc đánh giá sản phẩm trực tuyến trên các nền tảng thương mai điện tử như Tiki, Shopee, Lazada. Những dữ liệu này chứa đựng những ý kiến và đánh giá chi tiết về các khía cạnh khác nhau của các sản phẩm, bao gồm cảm xúc tích cực, trung tính và tiêu cực của người mua. Tuy nhiên, xử lý và phân tích lượng lớn dữ liệu này để thu thập được những thông tin hữu ích và những hiểu biết về sản phẩm là một thách thức đối với cả con người và các phương pháp truyền thống.

Bằng cách hiểu rõ hơn về cảm nhận và ý kiến của sản phẩm, các nhà sản xuất và nhà kinh doanh có thể điều chỉnh các thành phần nguyên liệu, giá bán, thiết kế, nội dung… để tạo ra những sản phẩm tốt hơn và làm hài long khách hàng. Đối với các nhà quảng cáo và đơn vị tiếp thị, phân tích cảm xúc và đánh giá sản phẩm có thể cung cấp thông tin quan trọng về mức độ phổ biến của một sản phẩm và sự hài lòng của khách hàng với nó. Các quyết định về chiến dịch tiếp thị, việc lựa chọn đối tượng khách hàng và cách tiếp cận sẽ trở nên hiệu quả hơn khi dựa trên những thông tin chi tiết về cảm nhận và đánh giá từ người mua. Nghiên cứu này cũng mang tính ứng dụng cao, với tiềm năng để phát triển các công cụ và dịch vụ phân tích cảm xúc và đánh giá bình luận tự động. Các công ty có thể tận dụng những công cụ này để thu thập phản hồi từ khách hang nhanh chóng và hiệu quả hơn, giúp định hình chiến lược tiếp thị và phát triển các sản phẩm phù hợp với nhu cầu của thị trường.

Một trong những yếu tố quan trọng ảnh hưởng đến quyết định mua hàng của khách hàng là các đánh giá của người mua trước đó. Do đó, việc áp dụng phân tích cảm xúc dựa trên văn bản để tự động phân loại các đánh giá của khách hàng là một bài toán có ý nghĩa thực tiễn cao cho các doanh nghiệp thương mại điện tử. Bằng cách này, họ có thể nắm bắt được xu hướng và sở thích của khách hàng, cũng như nhận ra được những điểm mạnh và điểm yếu của sản phẩm và dịch vụ của mình. Tuy nhiên, việc đọc và phân tích các đánh giá này bằng cách thủ công là tốn kém thời gian và công sức, đặc biệt khi số lượng đánh giá rất lớn và đa dạng. Trong bài nghiên cứu này, chúng tôi tập trung vào bài toán phân tích cảm xúc dựa trên tập dữ liệu bình luận sách trên trang Tiki, tập dữ liệu mà chúng tôi tự thu thập được. Mục tiêu của chúng tôi là sử dụng các kỹ thuật xử lý ngôn ngữ tự nhiên (NLP) kết hợp với phương pháp biểu diễn từ vựng TF-IDF và các mô hình học máy như Logistic

Regression, Naïve Bayes để xây dựng mô hình máy học nhằm phân loại các bình luận thành ba loại: tích cực (positive), trung tính (neutral) và tiêu cực (negative). Mục tiêu cuối cùng của đề tài là đánh giá hiệu suất của mô hình máy học dựa trên các thước đo như accuracy score, precision, recall và f1-score.

Xử lý ngôn ngữ tự nhiên mang những thách thức khó khăn vì nó yêu cầu phải phân tích các luồng dữ liệu từ nhiều góc nhìn, điều này đòi hỏi rất nhiều thời gian và công sức từ con người. Nguồn dữ liệu có thể là có cấu trúc, bán cấu trúc hoặc không cấu trúc tùy thuộc vào nền tảng lưu trữ. Máy móc có thể phân tích những dữ liệu phân tán này bằng việc sử dụng những từ hay cụm từ khóa có liên quan tới nghĩa của cả câu, qua đó có thể hiểu được nghĩa của văn bản. Phân tích cảm xúc đối với ngôn ngữ tự nhiên ở dây có thể được thực hiện bằng nhiều cách, bằng các hệ thống dựa trên quy tắc, tự động hay kết hợp bằng các quy tắc thủ công và các kỹ thuật học máy. Đầu tiên trong quá trình nghiên cứu mô hình học máy này, chúng tôi muốn đề cập đến quá trình phân tích dữ liệu và xử lý để có thể dễ dàng sử dụng cho các mô hình học máy.

Việc phân tích dữ liệu thường để kiểm tra tập dữ liệu cần phải được đảm bảo rằng các nhãn được phân bố với tỉ lệ phù hợp để đưa vào huấn luyện và kiểm thử. Có thể sử dụng một số phương pháp thống kê đối với những từ được chứa trong dữ liệu, từ đó đưa ra những nhận xét khách quan để đánh giá xem tập dữ liệu có phù hợp để huấn luyện và kiểm thử hay không. Những bước này cần được thực hiện đầy đủ và cẩn thận để đảm bảo tập dữ liệu có thể phù hợp với nhu cầu sử dụng sau khi hoàn thành huấn luyện mô hình học máy. Việc xử lý dữ liệu cũng rất quan trọng vì nó đòi hỏi chúng ta phải quyết định dạng chuẩn của dữ liệu phù hợp với mô hình học máy. Chúng ta cần phải xác định loại bỏ những ký tự không liên quan đến yêu cầu cũng như những từ ngữ mang ý nghĩa mơ hồ, không ảnh hưởng đến ý nghĩa của câu trong thực tế (stop words). Tuy nhiên trong một vài trường hợp cũng cần phải giữ lại những stop words khi chúng bổ sung ý nghĩa cho câu. Ngoài ra, còn một số cách xử lý dữ liệu giúp tăng hiệu suất huấn luyện mô hình như word stemming, lemmatize, tokenize, part-of-speech, text normalize… Bước tiếp theo là biến đổi những dữ liệu của chúng ta thành những tập hợp đặc trưng số học để chắc chắn những mô hình phân loại có thể hiểu được.

Ở đây, chúng ta có thể sử dụng những phương pháp Word Embedding như TF-IDF, bag of words (BOW), Word2Vec hay GloVe. Qua phương pháp này, chúng ta có thể biến đổi những từ ngữ sang không gian vector, biểu diễn được mối liên hệ, sự tương đồng về mặt ngữ nghĩa đối với dữ liệu chung. Ngoài ra có thể sử dụng mô hình chủ đề, ví dụ như phân bổ dirichlet tiềm ẩn (LDA) thuộc lớp mô hình sinh (generative model) xác định tập hợp các chủ đề được biểu diễn bởi tập hợp các từ.

Tiếp đến, ta sử dụng những mô hình phân loại khác nhau, có thể là những mô hình học có giám sát như Logistic Regression hay Naïve Bayes. Ngoài ra có thể kể đến những mô

hình kết hợp có thể được sử dụng đối với công việc phân loại như phân loại bỏ phiếu hay XGBoost. Sau khi chọn được những model cũng như đã xử lý những dữ liệu sao cho phù hợp với model, chúng ta tiến hành fine-tuning để mô hình học máy đạt hiệu suất cao cho một công việc cụ thể, phù hợp với nhu cầu sử dụng. Ở bước này, ta có thể áp dụng Data Augmentation, Transfer Learning hay sử dụng Hyperparameter Tuning, tìm ra những siêu tham số trong mô hình thông qua Gridsearch hoặc Randomsearch giúp mô hình đạt hiệu suất cao nhất đối với bài toán. Sau cùng, chúng ta sẽ đánh giá lại mô hình học máy bằng những thước đo như accuracy score, recall, precision hay f1-score. Việc đánh giá mô hình học máy giúp ta đảm bảo rằng mô hình hoạt động hiệu quả và không có vấn đề nào xảy ra.

Mô tả bài toán: Đầu vào là một câu bình luận mang tính cảm xúc, đầu ra là cảm xúc của bình luận đó (tích cực, tiêu cực hoặc trung tính).

Input: Giao hàng chậm, gói hàng không kỹ làm sản phẩm bị móp nặng. Liên lạc với cửa hàng thì không được đổi trả, làm ăn vô trách nhiệm.

Output: Tiêu cực.

II. TẠO NGỮ LIỆU VÀ TÁCH TỪ

2.1 Chuẩn bị ngữ liệu

Để thực hiện bài toán phân tích cảm xúc thông qua văn bản, nhóm em đã tiến hành thu thập các đánh giá của người mua hàng về các sản phẩm trên sàn thương mại điện tử Tiki, cụ thể là các sản phẩm đến từ mặt hàng sách. Lý do để nhóm chọn mặt hàng sách là vì những bình luận về sách mang tính chất chân thật nhiều hơn các mặt hàng khác, bình luận có đầu tư và thể hiện rõ cảm xúc của người đọc trên nhiều phương diện. Bộ dữ liệu có hơn 140000 bình luận với các mốc đánh giá từ 1 – 5 sao.

Dựa theo yêu cầu của đồ án môn học đó là phân tích hình thái của câu, chúng em đã chọn ra 61 câu trong bộ dữ liệu của bài toán để thực hiện việc phân tích hình thái. Các câu được chọn là các câu đơn, có chứa các tên riêng, chứa số. Các câu được chọn sẽ đáp ứng đầy đủ tiêu chí về mặt ngữ nghĩa cũng như đủ tiêu chuẩn để gọi là một câu, đồng thời nhóm cũng đã xử lí các từ viết tắt hay những khoảng trắng dư thừa. Cấu trúc của ngữ liệu sẽ như sau:

- Số lượng câu: 61 câu, trong đó:
    - 29 câu dùng cho việc huấn luyện.
    - 32 câu dùng cho việc kiểm tra.
- Số từ nhiều nhất trong một câu: 63.
- Số từ ít nhất trong một câu: 8.
- Mỗi dòng là 1 câu.
- Các từ được phân cách với nhau bằng dấu cách.
- Được lưu bằng tệp .txt.
- Chứa một số từ tên riêng tiếng Việt hoặc tiếng Hán (Ví dụ: Nhã Nam, Bác Nguyên Phong…).
- Chứa các kí hiệu độ dài, số thứ tự (Ví dụ: 12cm, quyển thứ 2…)

Tách từ (Word Segmentation) là một bài toán cơ bản trong Xử lý ngôn ngữ tự nhiên với tiếng Việt, đồng thời là bước tiền xử lý quan trọng cho các bài toán khác. Tách từ là một quá trình xử lý nhằm mục đích xác định ranh giới của các từ trong câu văn, cũng có thể hiểu đơn giản rằng tách từ là quá trình xác định các từ đơn, từ ghép… có trong câu.

Khác với tiếng Anh, một số từ tiếng Việt có thể được tạo ra bởi nhiều âm hay tiếng (syllable). Xét ví dụ từ "cá thể" được tạo ra bởi hai âm là "cá" và "thể", các từ đơn "cá" và "thể" lại có thể mang ý nghĩa khác so với từ "cá thể". Vì vậy ta cần dùng dấu gạch dưới "_" để liên kết hai âm của từ này thành "cá_thể".

Ngoài việc xác định các từ có nhiều âm tiết, chúng ta cũng cần tách các dấu câu riêng khỏi từ. Ví dụ câu "Hôm nay, tôi đi học." ta cần tách dấu "," khỏi từ "nay" và dấu chấm "." khỏi từ "học". Đây là quy ước chung cho tất cả các ngôn ngữ của bài toán tách từ

trong xử lý ngôn ngữ tự nhiên. Việc quy ước như vậy là để tạo thành chuẩn chung và để dễ xử lý hơn trong lập trình.

Phát biểu bài toán: Đầu vào là một câu hay văn bản tiếng Việt, đầu ra là câu hay văn bản đã được tách từ.

Input: "Sách lúc nhận hình thức rất ổn, không cong vênh xước gãy, đơn hàng này mình hài lòng với bên bán"

Output: "Sách lúc nhận hình_thức rất ổn , không cong vênh xước gãy , đơn hàng này mình hài_lòng với bên bán"

Có nhiều phương pháp để thực hiện tách từ như: Maximum Matching, Hidden Markov Model (HMM), Transformation based Learning (TBL)... Trong báo cáo này nhóm sử dụng một phương pháp đơn giản đó là Maximum Matching. Sau đó đem so sánh với các phương pháp tách từ khác đó là tách thủ công và sử dụng thư viện.

2.1.1 Phương pháp tách từ thủ công

Nhóm thực hiện tách từ thủ công bằng cách sử dụng từ điển tiếng Việt VLSP của GS. Hồ Tú Bảo [1] để kiểm tra các từ Tiếng Việt có nghĩa trong câu. Chúng em sẽ thực hiện tách thủ công trên chính hai thành viên của nhóm chúng em, sau đó tiến hành kiểm tra chéo và thống nhất với nhau về những lỗi sai và chọn ra những câu đúng nhất.

Quy ước khi thực hiện tách từ thủ công được nhóm thống nhất như sau:

- Giữ nguyên các từ viết tắt tiếng Việt và tiếng Anh.
- Giữ dấu của số thập phân, phân số, dấu định dạng ngày tháng thay vì tách ra giống như với các từ. (Ví dụ: "3.000.000" được giữ nguyên thay vì tách thành "3000000", "." và ".").
- Đối với từ có các âm được nối bởi dấu gạch nối "-" gắn liền (Ví dụ: "Covid-19") sẽ được giữ nguyên.
- Các từ ghép và danh từ riêng sẽ được nối bởi dấu "_".

2.1.2 Phương pháp Maximum Matching

Ý tưởng của phương pháp so khớp tối đa (Maximum Matching), hay còn được gọi là so khớp tối đa từ trái qua phải (From Left to Right Maximum Matching) là duyệt một câu vào từ trái qua phải và chọn cụm từ dài nhất có mặt trong một từ điển từ vựng đã cho. Nếu từ đó tồn tại trong từ điển, ta tách từ đó ra và di chuyển đến phần tiếp theo của câu. Nếu từ đó không tồn tại trong từ điển, ta giảm đi một ký tự và tiếp tục tìm kiếm. Quá trình này tiếp tục được lặp lại cho đến khi cụm từ tìm được có độ dài giảm dần cho đến hết câu.

Ưu điểm của phương pháp so khớp tối đa là đơn giản, dễ hiểu và chạy nhanh. Hơn nữa phương pháp chỉ cần một tệp từ điển đầy đủ là có thể tiến hành phân đoạn các văn bản, hoàn toàn không phải trải qua huấn luyện. Nhược điểm của phương pháp này là thuật toán

gặp phải nhiều nhập nhằng và phụ thuộc vào từ điển. Ví dụ, khi gặp các từ viết tắt, từ không nằm trong từ điển hoặc các từ có nhiều cách tách khác nhau, phương pháp này có thể cho kết quả không chính xác. Do đó, phương pháp Maximum Matching thường được kết hợp với các phương pháp khác để cải thiện hiệu suất và độ chính xác trong tách từ tiếng Việt. Ở đây chúng em sẽ sử dụng từ điển mở của thư viện underthesea [2] với gần 74.000 từ, là dự án tổng hợp các từ điển tiếng Việt, được phát triển bởi nhóm nghiên cứu xử lý ngôn ngữ tự nhiên tiếng Việt – underthesea. Từ điển được lưu trong file dictionary.txt.

Thuật toán có độ phức tạp O(nV) với n là số âm tiết trong chuỗi, V là số từ trong từ điển. Thuật toán của phương pháp Maximum Matching như sau (viết bằng mã giả):

```
# text: input text to be tokenized
# dictionary: list of valid vietnamese words
function maximum_matching_tokenize(text, dictionary):
    tokens = [] #store the result of token
    while text is not empty:
        found_word = False
        for i from the length of text down to 1:
            current_word = text[0:i]
            if current_word in dictionary: # find the longest matching word from dictionary
                tokens.append(current_word)
                text = text[i:]
                found_word = True
                break
        if not found_word:
            # if no matching word is found, treat the current character as a token
            tokens.append(text[0])
            text = text[1:]
    return tokens
```

2.1.3 Phương pháp tách từ bằng thư viện Underthesea

Underthesea là một bộ dữ liệu mô-đun Python mã nguồn mở và các hướng dẫn hỗ trợ nghiên cứu và phát triển trong xử lý ngôn ngữ tự nhiên tiếng Việt. Chúng cung cấp API cực kỳ dễ dàng để nhanh chóng áp dụng các mô hình NLP được đào tạo trước vào văn bản tiếng Việt của bạn, chẳng hạn như phân đoạn từ, gắn thẻ một phần của bài phát biểu (PoS), nhận dạng thực thể được đặt tên (NER), phân loại văn bản và phân tích cú pháp phụ thuộc. Underthesea được xây dựng dựa trên mô hình Markov ẩn (Hidden Markov Model) và tập dữ liệu đã được gán nhãn rộng lớn. Điều này giúp thư viện có khả năng tách từ chính xác và đáng tin cậy trong nhiều trường hợp.

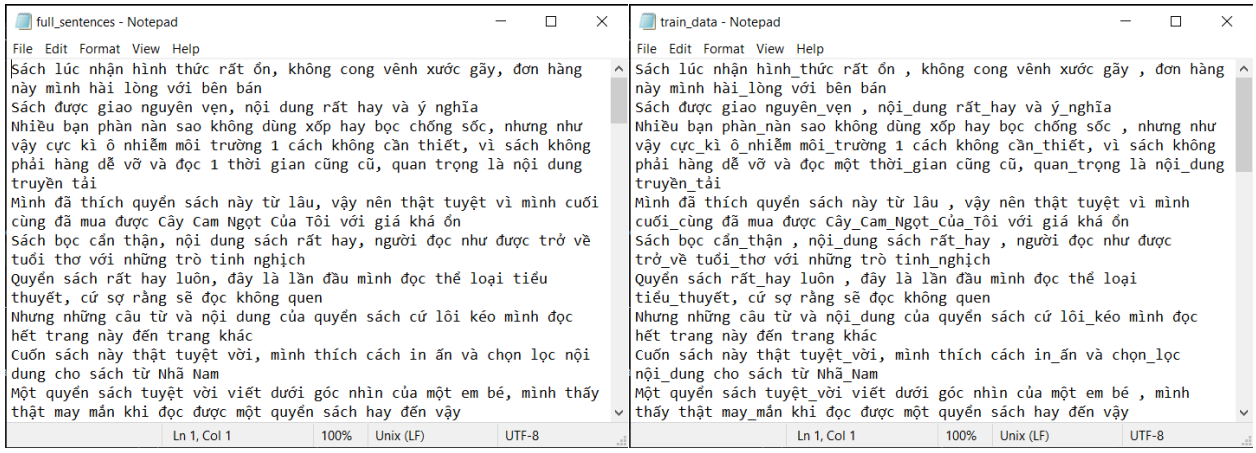
2.2 Tách từ

Chúng em thực hiện lần lượt bước sau:

- Tách thủ công: nhóm tiến hành phân công thực hiện tách từ thủ công, dựa vào từ điển VLSP tạo thành dữ liệu tách từ chuẩn. Sau khi tách thủ công chúng em sẽ phân chia các tập như sau:
  - full_sentences.txt: chứa dữ liệu của toán bộ các câu được chọn để tách.
  - data_byhand.txt: chứa toàn bộ câu đã được tách thủ công, từ đó chúng em chia làm các tập nhỏ hơn là:
    - train_data.txt: gồm 29 câu huấn luyện đã được tách thủ công.
    - test_data.txt: gồm 32 câu kiểm tra nguyên mẫu.
    - result_byhand.txt: gồm 32 câu kiểm tra đã được tách thủ công.

  Lí do nhóm tách như vậy vì nghĩ rằng đối với nguồn từ điển để thực hiện cho việc tách từ bằng phương pháp Maximum Matching là chưa đủ, do đó để có thể đạt hiệu suất tốt hơn nhóm sẽ cho từ điển học thêm trong các câu nhóm đã tách thủ công. Để có thể kiểm chứng nhóm sẽ tiền hành so sánh điểm số trước và sau khi cho thuật toán học.

- Tách từ bằng Maximum Matching: cài đặt thuật toán, tiến hành cho thuật toán học những từ đã được tách thủ công từ tệp train_data.txt, nếu như những từ đó chưa có trong bộ từ điển. Tiếp theo đó sử dụng thuật toán để tách từ ở tệp test_data.txt, sau đó lưu kết quả vào tệp result_maxmat.txt với thuật toán được học và tệp result_maxmat_notrain.txt với thuật toán không được học.
- Tách từ bằng thư viện underthesea: nhóm sẽ tiến hành tách từ trên tệp test_data.txt, sau đó lưu kết quả vào tệp result_underthesea.txt.



Hình 1: sơ bộ về các tệp được xử lí.

2.3 Các vấn đề gặp phải

Với từ điển đã được tạo sẵn, chúng ta có thể dễ dàng tách các từ ghép trong câu. Tuy nhiên cách tách từ trái sang phải đôi lúc cũng có những lỗi về mặt ngữ nghĩa. VD: 'Tôi tập thể dục hằng ngày' → 'Tôi', 'tập thể', 'dục', 'hằng ngày'. Để giải quyết vấn đề này, chúng ta tiến hành chạy thêm tách từ phải sang trái để so sánh kết quả, giúp tăng độ chính xác.

2.3 Kết quả

Sau khi tiến hành tách từ ta thu được số lượng từ được tách như sau:

- Tách từ thủ công: 803 từ
- Tách từ bằng Maximum Matching không huấn luyện: 766 từ
- Tách từ bằng Maximum Matching được huấn luyện: 761 từ
- Tách từ bằng underthesea: 794 từ

Nhóm sẽ so sánh dựa trên số từ được tách có mặt trong tệp result_byhand.txt cho các tệp đã được tách bằng hai phương pháp Maximum Matching và Underthesea. Kết quả như sau:

Phương pháp	Độ chính xác
Maximum Matching (không huấn luyện)	85.71%
Maximum Matching (được huấn luyện)	87.76%
Underthesea	88.78%

Nhận xét: kết quả cho ra bởi Underthesea khá ấn tượng, dễ hiểu vì theo công bố của nhóm thì trên thang điểm F1-score độ chính xác của họ đạt đến 97%. Trong khi đó phương pháp Maximum Matching cho kết quả khá tốt, đã có sự cải thiện so với việc được học thêm từ mới, lí do có thể là vì tập từ điển được sử dụng thiếu từ hoặc có những từ chưa chuẩn xác trong từ điển, và các danh từ riêng và danh từ ghép phức tạp thì bộ từ điển không nhận

biết được, dẫn đến các trường hợp nhận biết sai. Bộ từ điển hiện tại khoảng gần 74000 từ tuy nhiên nhóm có để ý thấy có một số từ không được công nhận, hoặc trong bộ từ điển VLSP không có nên nhóm đánh giá là không chuẩn, một phần vì bộ từ điển là từ điển mở, được thu thập từ nhiều nơi nên việc kiểm chứng tính đúng đắn sẽ bị hạn chế.

III. BỘ DỮ LIỆU BÀI TOÁN

3.1 Thông tin dữ liệu và tiền xử lí dữ liệu

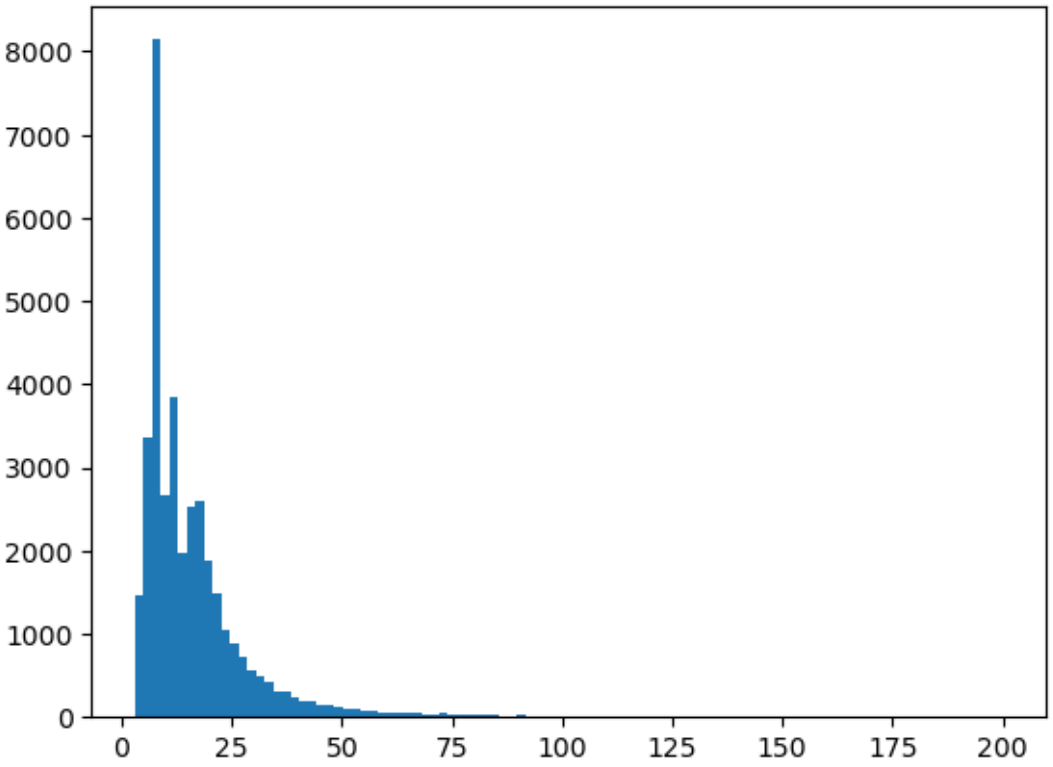
Bộ dữ liệu "Tiki Book Reviews" bao gồm tổng cộng 140000 bình luận được lấy từ Tiki, trong đó mỗi bình luận sẽ có mã số sản phẩm, mã số bình luận, tên bình luận, số lượng lượt tích, mã số người bình luận, xếp hạng sản phẩm và nội dung bình luận. Ngoài ra đi kèm với thông tin của những bình luận là thông tin của các cuốn sách được sử dụng để lấy dữ liệu, bao gồm hơn 1700 cuốn. Trong đó số lượng đánh giá cụ thể như sau: 1 sao (4495 đánh giá), 2 sao (2701 đánh giá), 3 sao (5276 đánh giá), 4 sao (14295 đánh giá), 5 sao (114514 đánh giá). Cấu trúc bộ dữ liệu như sau:

	product_id	comment_id	title	thank_count	customer_id	rating	comment
0	74021317	12559756	Cực kì hài lòng	313	22051463	5	Có những người bước đến, họ lấp đầy hạnh phúc ...
1	74021317	16979365	Cực kì hài lòng	6	27791831	5	Thấy nhiều bạn chê tiki gói hàng quá, may sao ...
2	74021317	14069617	Cực kì hài lòng	25	17748750	5	Bìa cực xinh, tiki giao hàng nhanh, sách không...
3	74021317	8569824	Cực kì hài lòng	57	410797	5	Sách lúc nhận hình thức rất ổn, không cong vên...
4	74021317	18368714	Cực kì hài lòng	0	28545286	5	Một cuốn sách rất đáng đọc về tình yêu thương ...

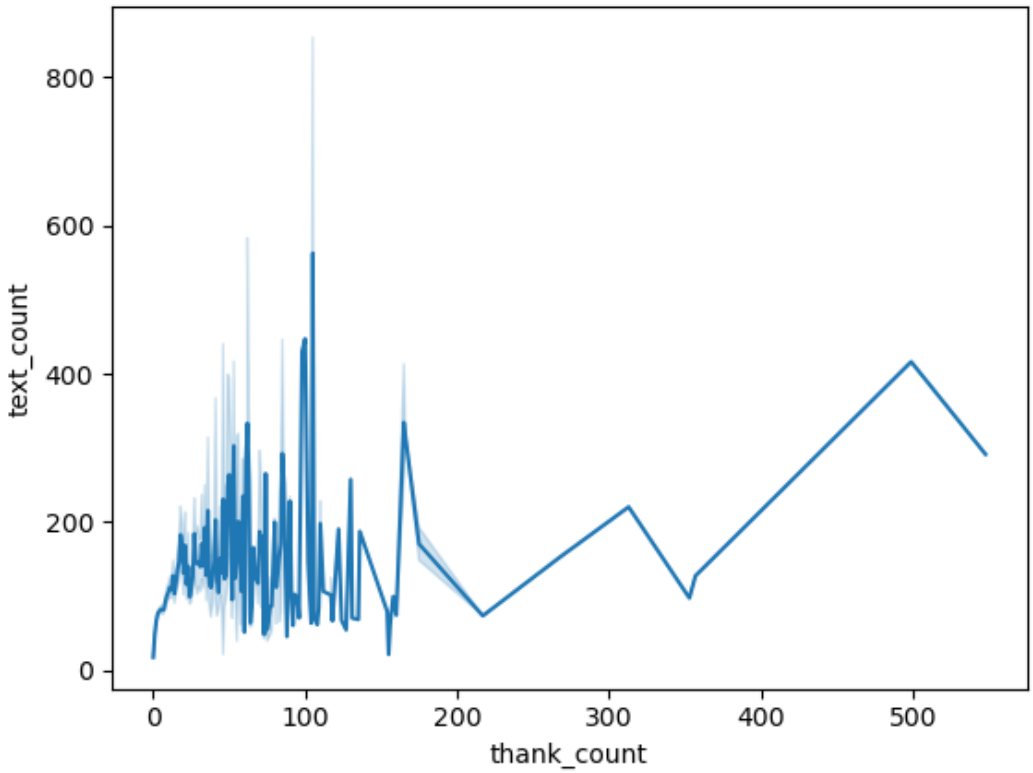
Hình 2: Sơ bộ về bộ dữ liệu.

```
#   Column       Non-Null Count   Dtype
---  ------       --------------   -----
0   product_id   141281 non-null  int64
1   comment_id   141281 non-null  int64
2   title        141277 non-null  object
3   thank_count  141281 non-null  int64
4   customer_id  141281 non-null  int64
5   rating       141281 non-null  int64
6   comment      103263 non-null  object
```

Hình 3: Thông tin các cột trong bộ dữ liệu.



Hình 4: Số lượng từ trong các câu bộ dữ liệu.



Hình 5: Độ tương quan giữa đặc trưng thank_count và text_count.

Từ bảng đo độ tương quan giữa các đặc trưng, ta thấy rằng có tương quan thuận cao giữa thank_count và text_count, điều này cho thấy những bình luận nhiều chữ nhận được nhiều thích hơn, điều này có thể là do những bình luận dài thường có nhiều thông tin hơn, dễ hiểu hơn, dễ đánh giá sản phẩm hơn.

Như đã được nêu ở trên, phần dữ liệu về những đánh giá sách ở trong bộ dữ liệu ban đầu vẫn chưa sẵn sàng để được đưa vào mô hình học máy để tiến hành huấn luyện phân loại được. Giải thích của việc này phần lớn là vì mô hình phân loại học máy cần dữ liệu ở dạng đã được quy định từ trước chứ không phải ở dạng đoạn văn ban đầu. Tiếng việt là một ngôn ngữ rất khó, đòi hỏi khả năng xử lí các văn bản phải tốn khá nhiều bước và có nhiều quy chuẩn khác nhau. Hơn nữa, do phần lớn dữ liệu ở đây được thu thập ở trên mạng internet nên có thể có những tag HTML hay những từ viết tắt. Những kỹ thuật tiền xử lý dữ liệu được chúng em sử dụng trong bài nghiên cứu bao gồm:

• Loại bỏ những URL (web links) – Trong những đánh giá có thể tồn tại những đường dẫn không liên quan, không ảnh hưởng tới nội dung của đánh giá. Việc loại bỏ những đường dẫn này cũng giúp làm giảm số chiều của dữ liệu, góp phần cải thiện hiệu suất hoạt động của mô hình học máy.

• Phân tích từ vựng - Chúng em sử dụng hàm phân tách văn bản đầu vào thành một danh sách các từ sử dụng word_tokenizer từ thư viện underthesea chuyên dùng để xử lý ngôn ngữ tự nhiên tiếng việt. Điều này được thực hiện để chia nhỏ văn bản đầu vào thành các từ riêng lẻ, giúp cho việc phân tích văn bản được thực hiện bởi các mô hình diễn ra dễ dàng hơn, góp phần gia tăng hiệu suất hoạt động của mô hình.

• Loại bỏ những từ dừng (stop words) – Thông qua việc sử dụng một danh sách các stop words có từ trước, chúng em tiến hành loại bỏ những từ như là “bạn”, “có chăng”, “này” . . . và những từ tương tự bởi chúng không mang lại ý nghĩa cho đánh giá cho sản phẩm. Những từ này chỉ được sử dụng để hỗ trợ các từ mang ý nghĩa chính. Loại bỏ những từ này ngoài giúp giảm chiều dữ liệu cũng một phần gia tăng hiệu suất hoạt động của mô hình học máy.

• Loại bỏ những đánh giá trùng lặp - Trong bộ dữ liệu này, có tồn tại những đánh giá trùng lặp với nhãn giống nhau. Trong quá trình tiến hành phân tích bộ dữ liệu, chúng em phát hiện có rất nhiều đánh giá bị trùng lặp. Các sàn thương mại điện tử hiện nay đều có chức năng gợi ý những đánh giá có sẵn, tuy nhiên có nhiều đánh giá như vậy thì mô hình sẽ học những thứ không cần thiết và làm tăng thời gian huấn luyện. Những đánh giá này phần lớn có số lượng từ ít và mang những nhãn cảm xúc giống nhau, loại bỏ những đánh giá sẽ góp phần làm giảm kích thước bộ dữ liệu.

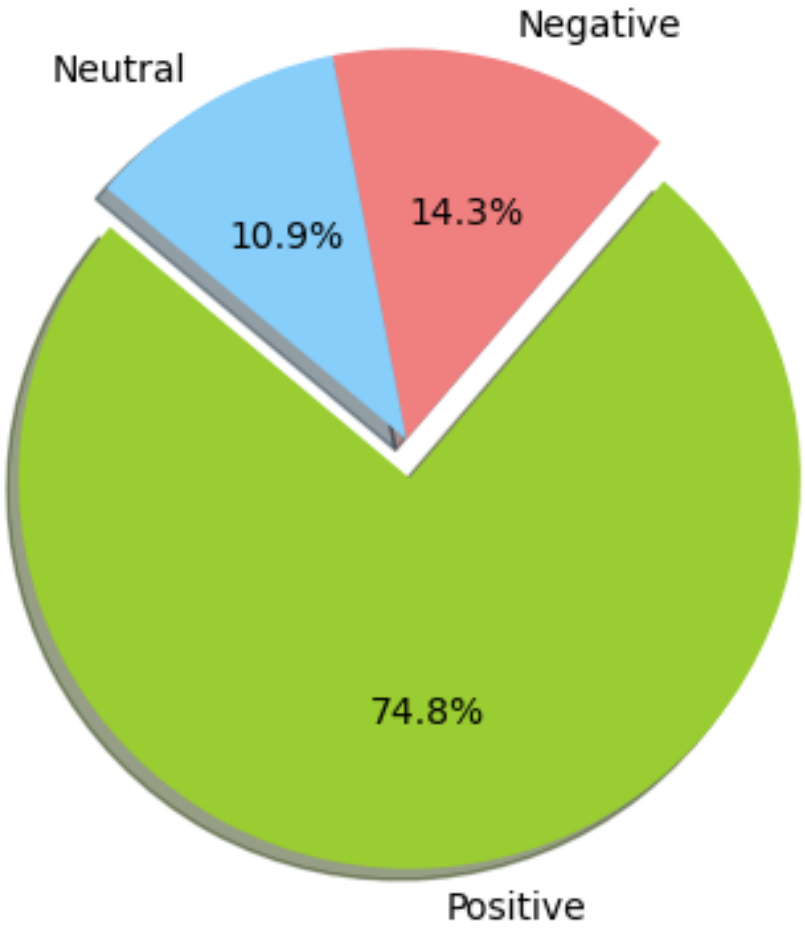
• Chuyển hóa các từ ngữ không chuẩn hoặc những thứ thể hiện cảm xúc nhưng máy không thể hiểu như: chuyển emojicon sang các từ ngữ thể hiện cảm xúc của chúng, chuyển các từ teencode về chuẩn nghĩa gốc trong tiếng việt, chuyển các từ tiếng anh có lẫn trong câu về tiếng việt, loại bỏ các từ ngữ sai, loại bỏ các dấu cách giữa các từ bị dư. Điều này làm cho câu trở thành những câu đạt chuẩn về ngữ nghĩa cũng như thể hiện đúng cảm xúc của người bình luận. Mục tiêu hàng đầu chính là phân tích cảm xúc của bình luận, do đó việc này là hoàn toàn cần thiết để giúp mô hình có thể hiểu đúng ý nghĩa của từ ngữ cũng như áp dụng vào việc dự đoán chính xác hơn.

• Chuẩn hóa unicode Tiếng Việt: chuyển đổi các từ đang ở dạng bảng mã 1252 về utf-8, tức là chuyển đổi các từ được gõ bằng Unicode tổ hợp sang Unicode dựng sẵn. Có những từ tuy nhìn giống nhau nhưng lại không giống nhau, đơn giản vì chúng được viết trên bảng mã khác nhau. Chuyển đổi về một dạng sẽ giúp mô hình giảm thời gian huấn luyện cũng như cho hiệu suất tốt hơn.

• Xử lí từ đặc biệt: những từ như “quá đúng” hay “cũng bị” thường cũng chỉ mang nghĩa như những từ gốc đó là “đúng” và “bị” nhưng với một sắc thái cao hơn. Tuy nhiên với thư viện underthesea thì những từ đó không nằm trong từ điển, cho nên khi tách từ sẽ không tách được những từ đặc biệt như vậy, trong khi đó chúng cũng chỉ mang nghĩa tương tự những từ gốc. Do đó chúng em đã chuyển đổi chúng trở thành cụm từ mà mô hình có thể học được như “quá_đúng”, “cũng_bị” thay vì học từ “quá” và “đúng”, “cũng” và “bị” dẫn đến việc mô hình hiểu sai nghĩa của câu.

• Xử lí pos-tag: khi tiến hành tách từ bằng thư viện underthesea thì sẽ xuất hiện các pos-tag đứng bên các từ được tách, thì chúng em sẽ loại bỏ đi để trở thành những từ bình thường.

Sau khi thực hiện quá trình tiền xử lí, nhận thấy được một điều rằng dữ liệu còn nhiều nội dung chưa chính xác với hình thái và đúng với bình luận, chúng em quyết định gán nhãn thủ công kết hợp tự động cho hơn 35000 đánh giá và hoàn thành bộ dữ liệu. Số từ nhiều nhất của bộ dữ liệu này sau khi tiền xử lí là 200 và ít nhất sau khi tiền xử lí là 3. Tỉ lệ các câu trong bộ dữ liệu ở bước cuối cùng như sau:



Hình 6: Biểu đồ thể hiện phân bố của các nhãn bình luận

Với bộ dữ liệu này, nhãn cảm xúc của những bình luận được gán thành positive (tích cực), neutral (trung tính) và negative (tiêu cực). Tuy nhiên số lượng các câu mang các nhãn cảm xúc lại không đồng đều, cụ thể là có: 27643 bình luận mang nhãn positive, 5279 bình luận mang nhãn negative và 4010 bình luận mang nhãn trung tính. Do tính không đồng đều của dữ liệu, đồng thời nhóm đã xem xét các từ xuất hiện trong hai nhãn trung tính và tiêu cực có nhiều từ ngữ xuất hiện khá giống nhau nên nhóm đã quyết định huấn luyện mô hình theo 2 cách:

- Cách 1: vẫn huấn luyện theo cách gán 3 loại nhãn cho dữ liệu. Không sử dụng thêm công cụ bổ trợ nào.
- Cách 2: gộp các bình luận có nhãn negative và neutral vào chung 1 nhóm là negative, do đó ghi gộp thì việc phân loại sẽ được tăng hiệu suất vì các mô hình học máy thường sẽ đạt hiệu suất tốt khi phân loại nhị phân.

Sau khi phân phối nhãn dữ liệu, tiếp theo là phân chia dữ liệu thành các tập huấn luyện và tập kiểm thử. Phân chia này giúp đánh giá hiệu suất và độ chính xác của mô hình. Thông thường, một tỷ lệ phân chia phổ biến là 80-20 hoặc 70-30, tức là 80% (70%) dữ liệu cho tập huấn luyện và 20% (30%) dữ liệu cho tập kiểm thử. Tỷ lệ này phụ thuộc vào kích thước của tập dữ liệu và đặc điểm của bộ dữ liệu. Trong bài nghiên cứu này, sau khi áp dụng kiểm chứng chéo ngẫu nhiên (Cross Validation) với 42 lần random_state với model đại diện là Logistic Regression và mỗi tỷ lệ kiểm thử đã được nêu ở trên, kết quả cho thấy với test_size = 0.3 thì hiệu suất huấn luyện cho ra là cao nhất. Qua đấy, chúng em tiến hành sử dụng tỷ lệ huấn luyện - kiểm thử là 70-30.

IV. CÁC MÔ HÌNH HỌC MÁY

Trong quá trình nghiên cứu, chúng em đã sử dụng thư viện Lazy Predict [3], một thư viện giúp xây dựng nhiều mô hình cơ bản mà không cần nhiều mẫu và giúp hiểu mô hình nào hoạt động tốt hơn mà không cần bất kỳ điều chỉnh tham số nào. Nhờ sử dụng Lazy Predict chúng em đã chọn ra được những mô hình có hiệu suất tốt đó là Logistic Regression, RandomForestClassifier, SVC và KneighboirsClassifier.

Sau khi cho chạy thử các mô hình đã được lựa chọn thì chúng em nhận thấy rằng mô hình Logistic Regression cho kết quả tốt nhất. Sau đó chúng em quyết định sự dụng hai mô hình chính để huấn luyện và tối ưu đó là Logistic Regression và Naıve Bayes.

4.1 Mô Hình Logistic Regression

Logistic Regression (Loh, 2011) [3] là một phương pháp thống kê được sử dụng để biểu thị mối quan hệ giữa biến kết quả nhị phân, một hoặc nhiều biến dự đoán. Ý tưởng cơ bản của hồi quy Logistic là sử dụng hàm Logistic để ánh xạ các biến dự báo vào một thang đo có xác suất từ 1 đến 0, từ đó có thể dự báo xác suất biến kết quả có thể nhận được giá trị mong muốn.

Hàm logistic được xác định bởi phương trình:

$$p = \frac{1}{1 + e^{-z}}$$

Trong đó p là xác suất biến kết quả nhận giá trị quan tâm, z là tổ hợp tuyến tính của các biến dự báo. Các hệ số của mô hình hồi quy logistic được ước lượng bằng phương pháp ước lượng hợp lý cực đại (Maximum likelihood estimation). Trong đó, đánh giá xem các giá trị hệ số nào sẽ cung cấp khả năng làm cho xác suất dự báo của biến kết quả gần nhất có thể với các giá trị thực tế của biến kết quả trong tập dữ liệu. Hiệu suất của mô hình hồi quy logistic có thể được đánh giá bằng các chỉ số đo lường sự phù hợp, như là độ lệch, AIC hoặc BIC.

Trong bài nghiên cứu, nhóm chúng em sử dụng mô hình học máy Logistic Regression của thư viện sklearn (Pedregosa et al., 2011) [4]. Sau quá trình tinh chỉnh sử dụng GridsearchCV để tinh chỉnh mô hình với tham số C có giá trị [0.001, 0.01, 0.1, 1, 10, 100] thì kết quả cho thấy tham số C với giá trị 100 cho phân loại 2 nhãn và 10 cho phân loại 3 nhãn. Tham số này đóng vai trò thay đổi sự chặt chẽ của việc phân loại trong mô hình hồi quy. Giá trị C càng nhỏ thì mô hình có xu hướng phân loại linh hoạt hơn, cho phép các điểm dữ liệu bị phân loại sai giữa các lớp. Giá trị C càng lớn thì mô hình có xu hướng phân loại chặt chẽ hơn, mô hình sẽ ưu tiên tìm một ranh giới quyết định rõ ràng hơn giữa các lớp.

4.2 Mô Hình Naïve Bayes

Naïve Bayes là một thuật toán máy học được dựa trên định lý Bayes về giả định tính độc lập giữa các đặc trưng. Thuật toán này dưa trên xác suất để dự đoán hay phân lớp nhãn của một mẫu dựa trên xác suất tiên nghiệm và xác suất hậu nghiệm. Naïve Bayes có cơ chế hoạt động dựa trên định lý Bayes, một định lý cơ bản trong xác suất thống kê. Công thức Bayes cho phép tính xác suất hậu nghiệm (xác suất của một biến cố xảy ra sau khi có thông tin mới) dựa trên xác suất tiên nghiệm (xác suất của một biến cố diễn ra trước khi có thông tin mới) và thông tin mới đó. Xác suất hậu nghiệm được tính như sau:

$$P(y|x) = \frac{P(x|y) \cdot P(y)}{P(x)}$$

Trong đó, P(y|x) là xác suất của lớp y khi biết x, P(x|y) là xác suất của lớp x khi biết y, P(y) là xác suất tiên nghiệm của lớp y và P(x) là xác suất đặc trưng x. Thông qua việc tính toán xác suất diễn ra của từng lớp, ta có thể tiến hành phân loại dựa vào phân lớp có xác suất cao nhất.

Trong bài nghiên cứu này, nhóm chúng em sử dụng thuật toán multinomialNB của thư viện sklearn (Pedregosa et al., 2011) [4]. Trong multinomial Naïve Bayes (multimonialNB), ta giả định các đặc trưng là biến rời rạc và tuân theo phân phối đa thức (multinomial). Trong mô hình multinomialNB thuộc thư viện sklearn, tham số alpha là một tham số quan trọng, tham gia điều chỉnh mức độ ảnh hưởng của xác suất tiên nghiệm của các đặc trưng khi biết lớp trong quá trình tính toán. Nếu giá trị của alpha lớn, mô hình sẽ đặt mức độ ảnh hưởng cao cho xác suất tiên nghiệm và giảm khả năng bị overfitting. Ngược lại, với alpha nhỏ thì mô hình sẽ tập trung vào dữ liệu huấn luyện và có khả năng bị overfitting.

Trong quá trình nghiên cứu, qua sử dụng GridSearchCV để điều chỉnh tham số alpha [0.001, 0.01, 0.1, 1.0, 10]. Kết quả thu được là với tham số 1.0 cho phân loại 2 nhãn và 0.1 cho phân loại 3 nhãn, mô hình multinomialNB hoạt động tốt nhất.

4.3 Trích xuất đặc trưng

Trong xử lý ngôn ngữ tự nhiên, trích xuất đặc trưng là quá trình chuyển đổi văn bản thành một tập hợp các đặc trưng số học để có thể được sử dụng để huấn luyện mô hình học máy. Các phương pháp trích xuất đặc trưng có thể được áp dụng ở đây đã được kể đến ở phần giới thiệu như TF-IDF, bag of words, word2vec hay GloVe. Trong bài nghiên cứu này, nhóm chúng em sử dụng TF-IDF như phương pháp trích xuất đặc trưng dữ liệu đối với những đánh giá sáchtrong bộ dữ liệu ày.

Như nội dung được trình bày tại chương về trích xuất đặc trưng trong cuốn sách (Uther et al., 2011) [5], TF-IDF (Term frequency-Inverse document frequency) là một phương pháp đánh trọng số các từ xuất hiện trong văn bản.. Phương pháp này thường sử dụng để biểu

diễn các văn bản dưới dạng các vector (cho mục đích phân loại, phân cụm, trực quan hóa, truy xuất…). Giả sử: T = $\{t_1,...,t_n\}$ là tập hợp của tất cả các từ xuất hiện trong tập dữ liệu đang xét. Khi đó, một văn bản d_i được biểu diễn bởi vectơ có n chiều giá trị thực $x_i = (x_{i1},…,x_{in})$ với mỗi đối tượng tương ứng với một từ có thể có trong T.

Phương pháp này cân nhắc tới 2 yếu tố chính là tần số xuất hiện của từ trong văn bản và tần suất nghịch đảo của từ trong tập dữ liệu:

- Tần số xuất hiện của từ (TF): đo lường tần số suất hiện của từ trong văn bản, tần số càng cao, từ đó xuất hiện càng nhiều trong văn bản.
- Tần suất nghịch đảo của từ (IDF): đo lường mức độ quan trọng của từ trong toàn bộ tập dữ liệu. Tần suất nghịch đảo càng cao có nghĩa rằng từ đó xuất hiện ít trong toàn bộ tập dữ liệu.

Trọng số của x_{ij} tương ứng với từ t_j trong văn bản d_i thường là 1 tích của ba giá trị. Một phần phụ thuộc vào tần số xuất hiện của từ t_j trong văn bản d_i (IDF), một phần phụ thuộc vào mức độ quan trọng của t_j trong toàn bộ tập dữ liệu. Thông thường, trọng số TF-IDF được tính bằng công thức:

$$\text{TF-IDF}(t_j, d_i) = \text{TF}_{t_j,d_i} \times \text{IDF}(t_j) \quad (1)$$

Trong đó:

- $TF_{tj,di}$ là tần số xuất hiện của từ t_j trong d_j, được tính bằng nhiều cách khác nhau, ví dụ như đếm số lần xuất hiện của từ t_j trong d_i hoặc sử dụng các phương pháp chuẩn hóa tần số.
- IDF(t_j) là tần suất nghịch đảo của từ t_j. Nó được tính bằng cách lấy tỷ lệ giữa tổng số tài liệu trong tập dữ liệu (N) và số từ chứa từ t_j (DF(t_l)), sau đó lấy log cơ số e của kết quả. Công thức chính xác là:

$$\text{IDF}(t_j) = \log_e\left(\frac{N}{\text{DF}(t_j)}\right)$$

Sau khi tiến hành quá trình trích xuất đặc trưng bằng việc sử dụng TfidfVectorizer của thư viện sklearn (Pedregosa et al., 2011) [4], kết quả cho ra là ma trận có kích thước (35582, 107) biểu thị cho 35582 đánh giá được lọc ra trong tập dữ liệu với 107 đặc trưng được trích xuất từ các đánh giá trong tập dữ liệu.

4.4 Xây dựng mô hình

Bước 1: Nhập thư viện và dữ liệu, kiểm tra dữ liệu lại 1 lần nữa với việc xoá bỏ bình luận lặp và bình luận rỗng.

Bước 2: Tạo vector trích xuất đặc trưng cho dữ liệu.

Bước 3: Sử dụng Cross Validation để chia tệp ngẫu nhiên với test_size = [0.3, 0.2], tương ứng với chia tệp theo tỉ lệ 70-30 và 80-20. Ở đây chúng em nhận thấy độ chính xác khi chia theo tỉ lệ 0.3 cao hơn so với khi chia theo tỉ lệ 0.2 (77.64% so với 77.56%). Do mức độ chênh nhau không nhiều nên chúng em quyết định chọn tỉ lệ chia tệp sẽ là 0.3, tương ứng với 70% cho tập train và 30% cho tập test.

Bước 4: Nhập các model được sử dụng để so sánh, ở đây chúng em sử dụng Random Forest Classifier, KNN, Logistic Regression, SVC và Naïve Bayes (cho mô hình 3 nhãn) và XGB Classifier, Logistic Regression, Extra Trees Classifier, Naïve Bayes và KNN (cho mô hình 2 nhãn). Lí do chúng em chọn mô hình này là vì đã sử dụng Lazy Predict để thử nghiệm và chọn ra những mô hình đạt hiệu suất tốt. Sau khi chạy chúng em có kết quả như sau:

	Model	Accuracy Mean	Accuracy 3 * STD	Time
0	XGBClassifier	0.804794	0.010561	0 days 00:00:05.081269
1	LogisticRegression	0.800940	0.010891	0 days 00:00:00.359201
2	ExtraTreesClassifier_100	0.800739	0.011180	0 days 00:00:24.785652
3	ExtraTreesClassifier_50	0.799133	0.008925	0 days 00:00:15.768028
4	MultinomialNB	0.787610	0.015868	0 days 00:00:00.155178
5	KNeighborsClassifier_7	0.776127	0.011583	0 days 00:00:05.515771
6	KNeighborsClassifier_5	0.773116	0.010857	0 days 00:00:06.405106
7	KNeighborsClassifier_3	0.757940	0.009355	0 days 00:00:06.336375

Hình 7: Độ chính xác của các mô hình khi chạy trên 3 nhãn.

	Model	Accuracy Mean	Accuracy 3 * STD	Time
0	LogisticRegression	0.772534	0.008332	0 days 00:00:00.701078
1	RandomForestClassifier_100	0.768561	0.007829	0 days 00:00:22.964205
2	RandomForestClassifier_50	0.767197	0.007273	0 days 00:00:16.666974
3	MultinomialNB	0.765350	0.002913	0 days 00:00:00.259208
4	KNeighborsClassifier_7	0.734570	0.013840	0 days 00:00:06.146669
5	SVC	0.726222	0.010050	0 days 00:00:34.016580
6	KNeighborsClassifier_5	0.721687	0.017547	0 days 00:00:06.918908
7	KNeighborsClassifier_3	0.706036	0.020010	0 days 00:00:07.733388

Hình 8: Độ chính xác của các mô hình khi chạy trên 2 nhãn.

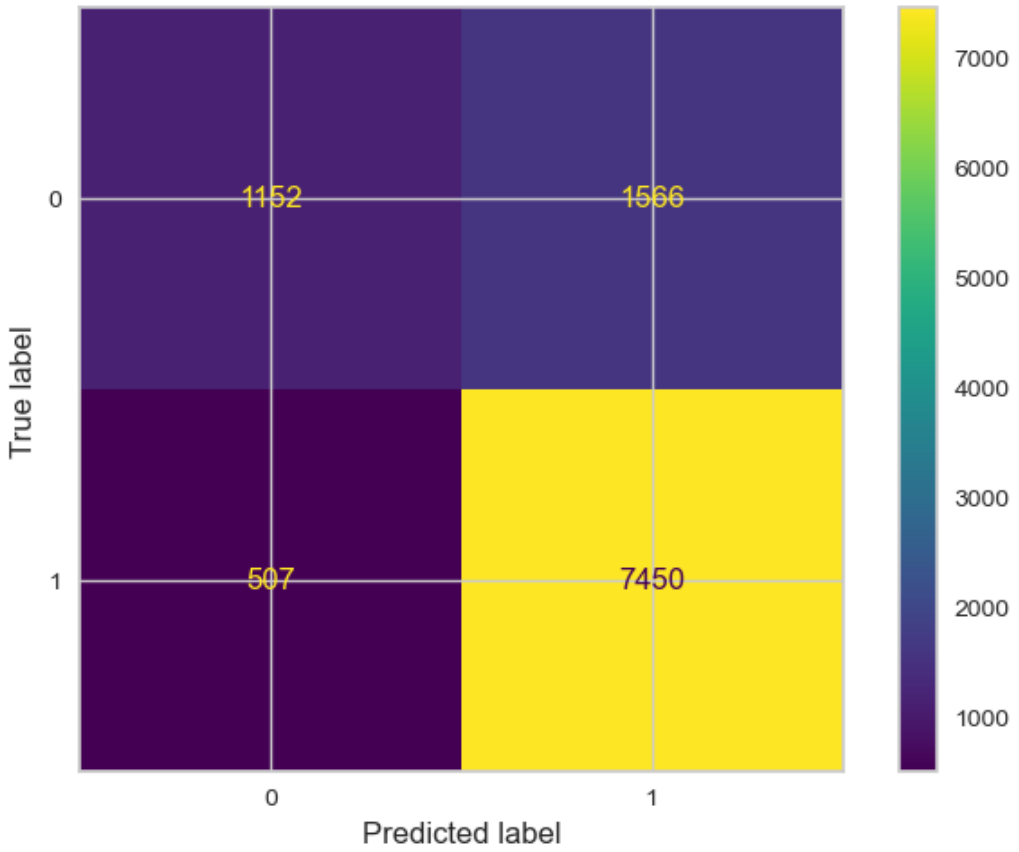
Mô hình Logistic Regression là mô hình có độ ổn định tốt khi đều đạt kết quả cao ở phân loại 2 nhãn và 3 nhãn. Do đó chúng em sẽ chọn Logistic Regression và Naive Bayes để tối ưu.

Bước 5: Tối ưu mô hình

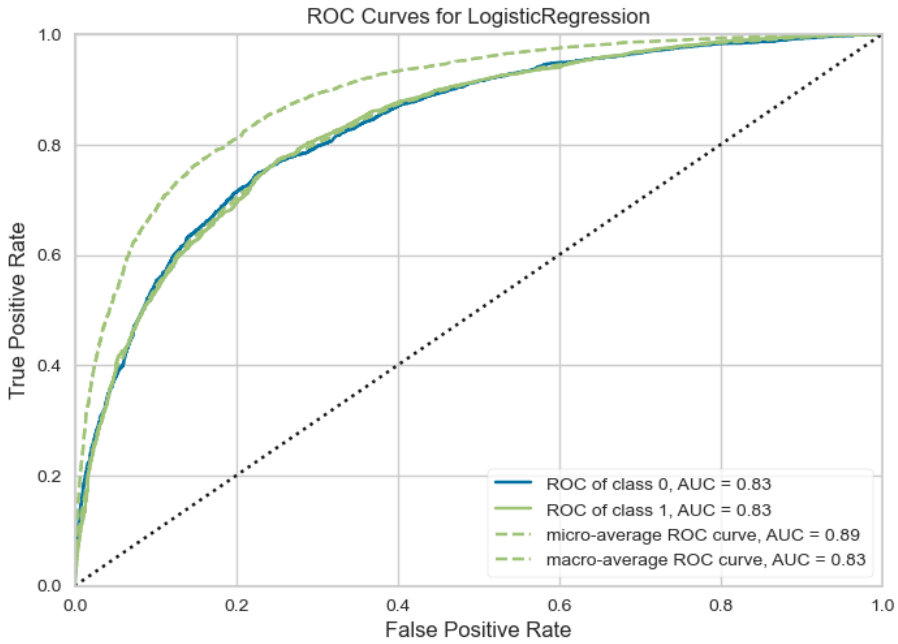
1. Với mô hình 2 nhãn:

- Tối ưu mô hình Logistic Regression với các tham số sau:
    - solvers = ['newton-cg', 'lbfgs', 'liblinear']
    - penalty = ['l2']
    - c_values = [100, 10, 1.0, 0.1, 0.01]

Best params: {'C': 100, 'penalty': 'l2', 'solver': 'liblinear'}



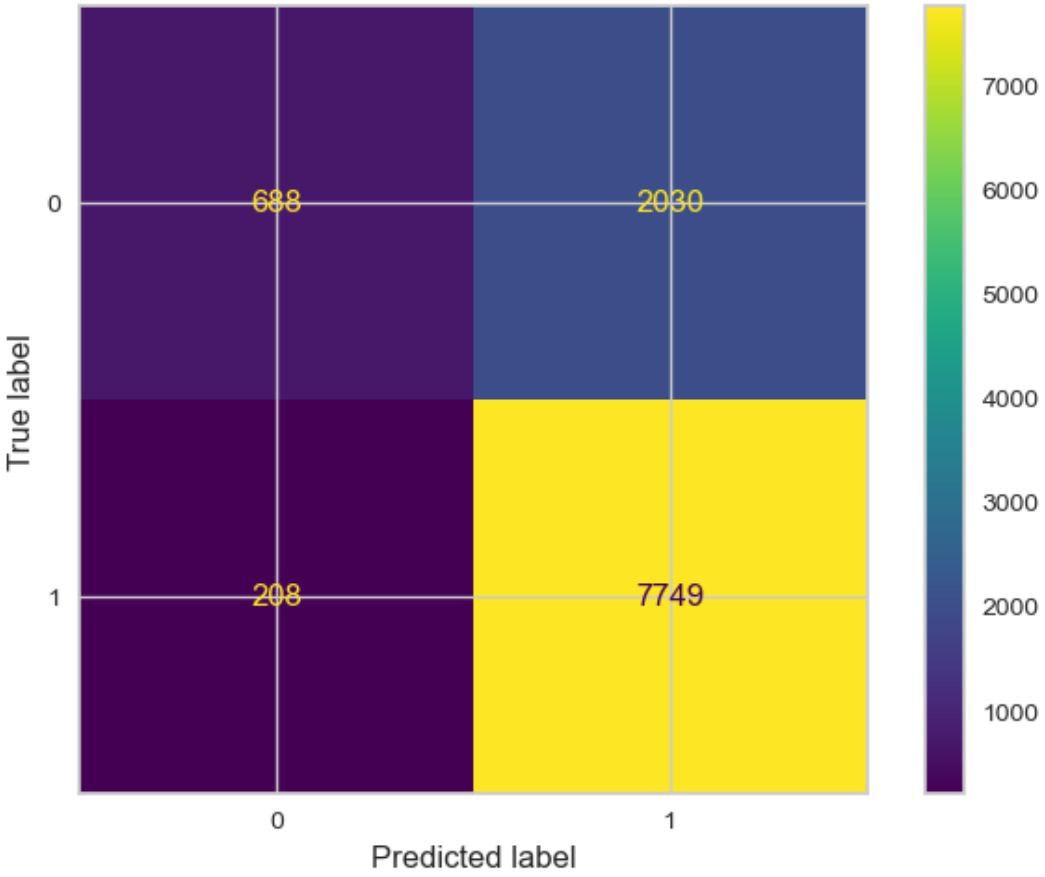
Hình 9: Confusion Matrix của mô hình Logistic Regression trên 2 nhãn.



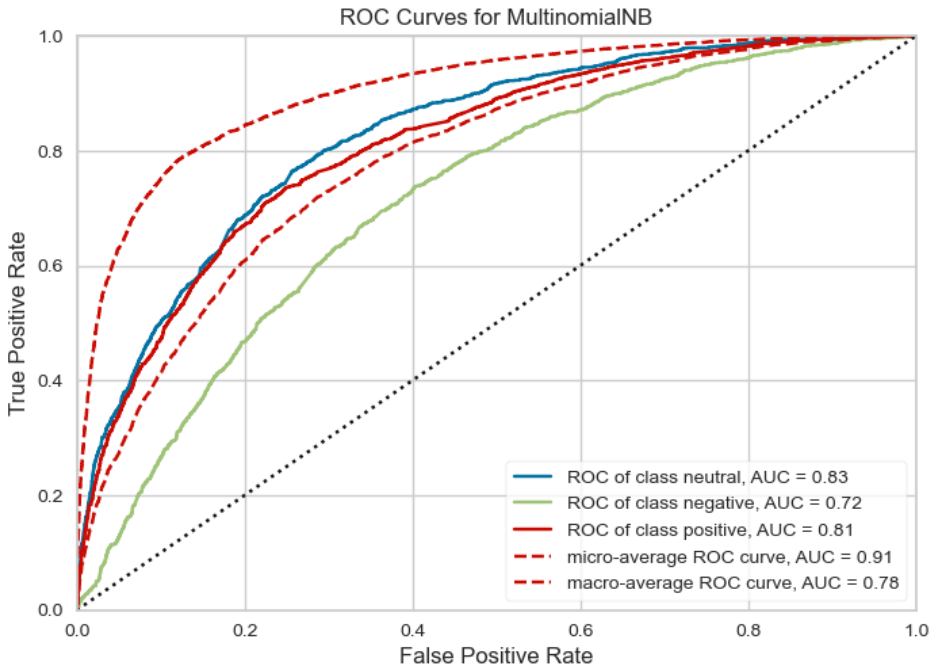
Hình 10: AUC – ROC của mô hình Logistic Regression trên 2 nhãn.

- Tối ưu mô hình Naive Bayes với các tham số sau:
  - alpha = [0.1, 0.5, 1.0, 10.0]
  - fit_prior = [True, False]

Best params: {‘alpha’: 1.0, ‘fit_prior’: True}



Hình 11: Confusion Matrix của mô hình Naïve Bayes trên 2 nhãn.

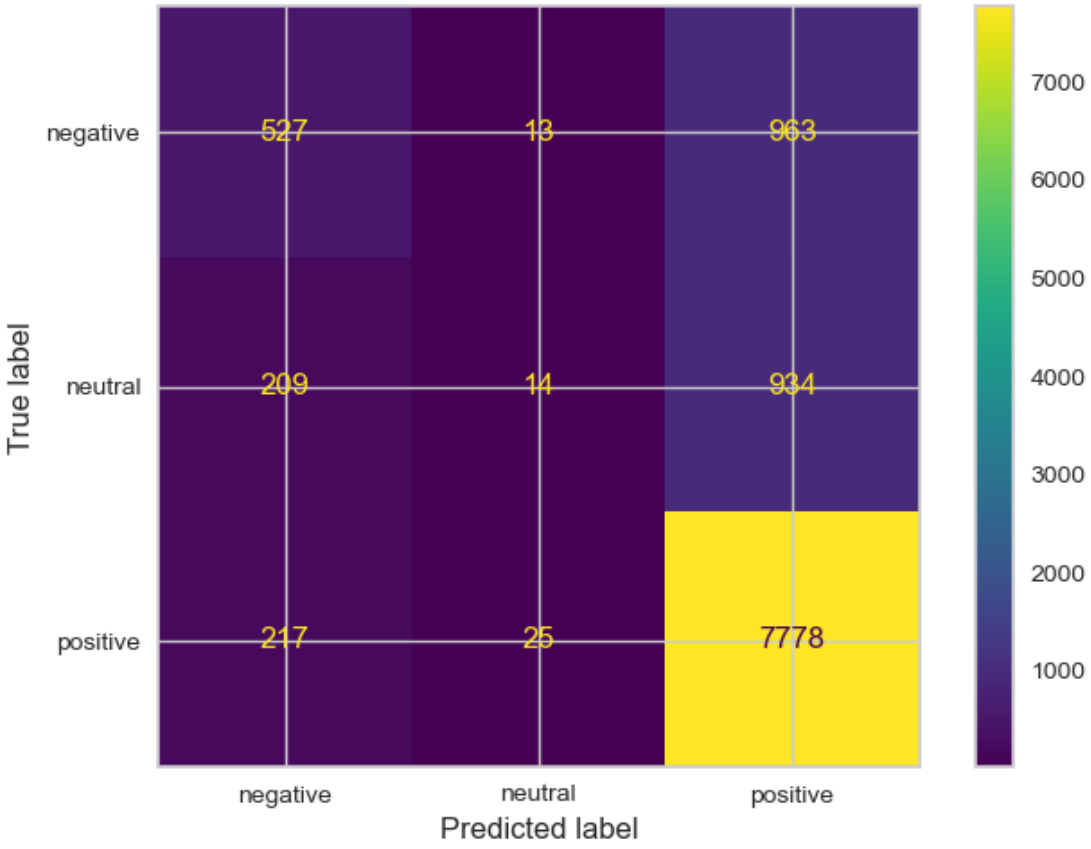


Hình 12: AUC – ROC của mô hình Naïve Bayes trên 2 nhãn.

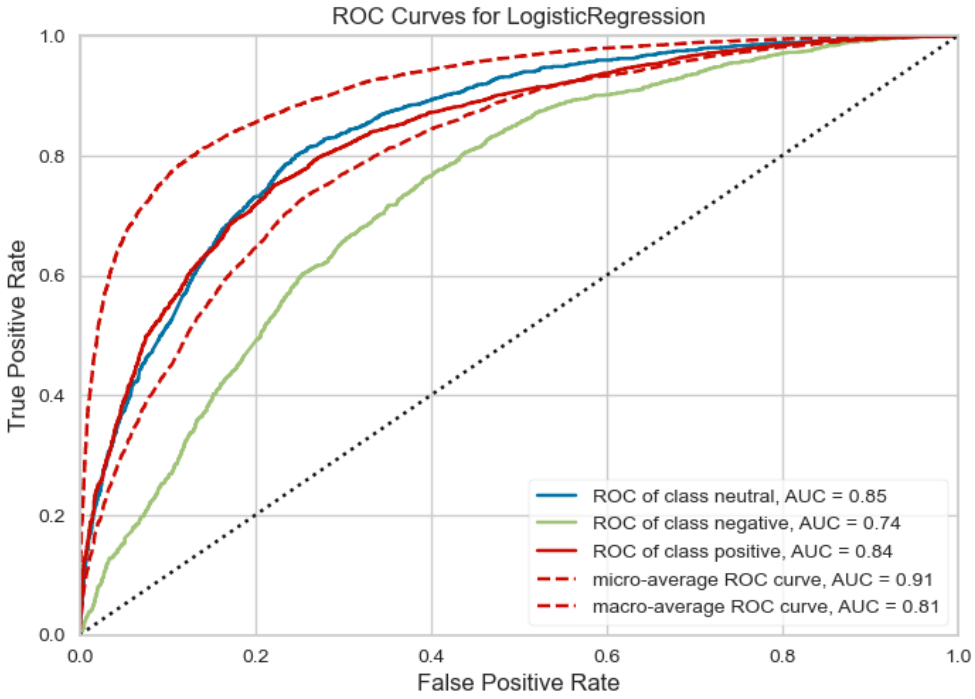
2. Với mô hình 3 nhãn:

- Tối ưu mô hình Logistic Regression với các tham số sau:
    - solvers = ['newton-cg', 'lbfgs', 'liblinear']
    - penalty = ['l2']
    - c_values = [100, 10, 1.0, 0.1, 0.01]

Best params: {‘C’: 10, ‘penalty’: ‘l2’, ‘solver’: ‘lbfgs’}



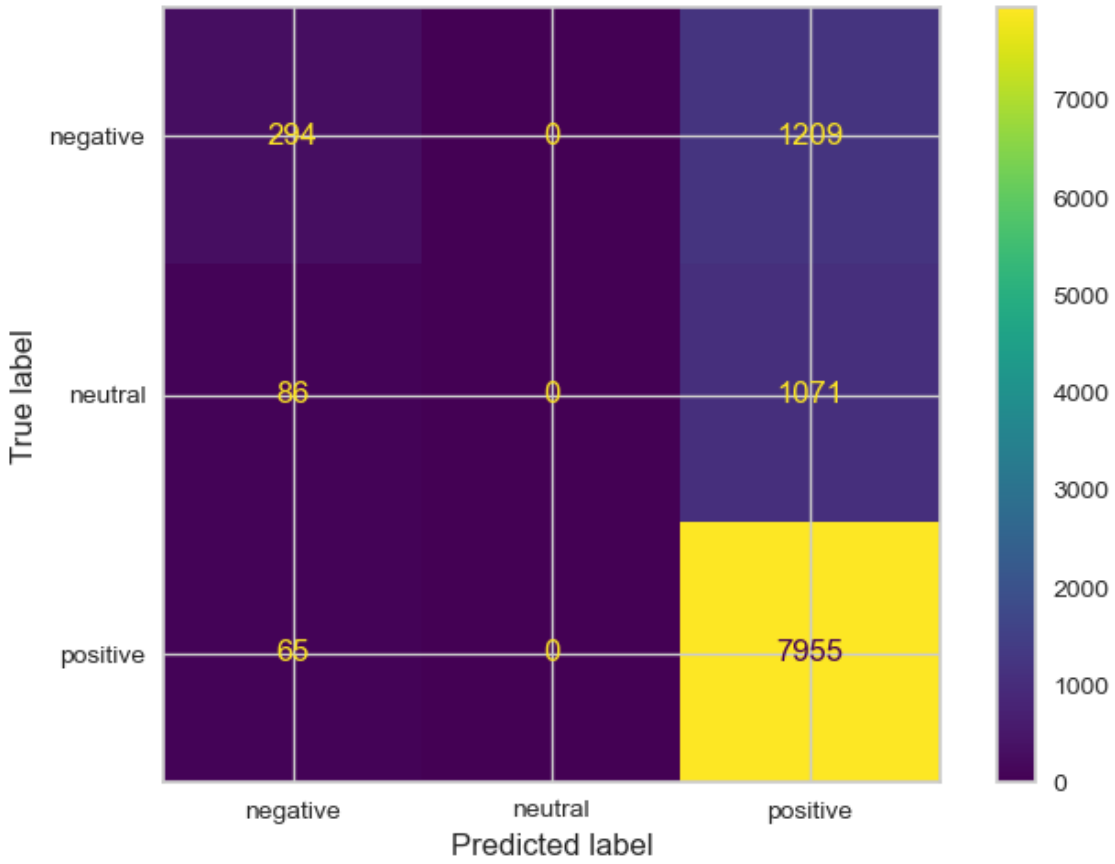
Hình 13: Confusion Matrix của mô hình Logistic Regression trên 3 nhãn.



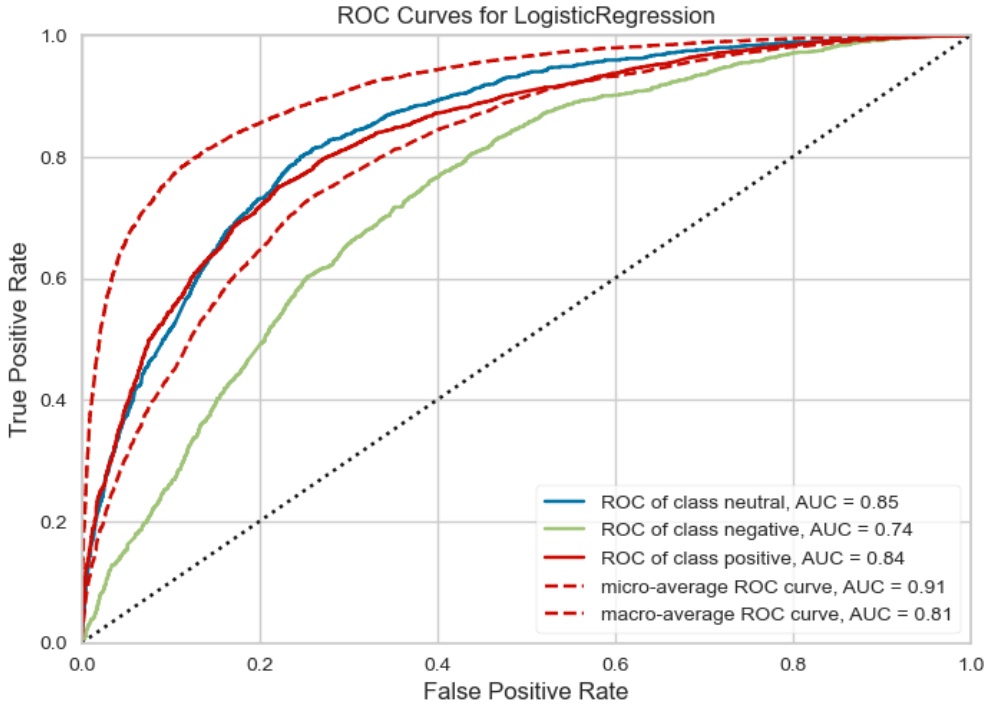
Hình 14: AUC – ROC của mô hình Logistic Regression trên 3 nhãn.

- Tối ưu mô hình Naïve Bayes với các tham số sau:
  - alpha = [0.1, 0.5, 1.0, 10.0]
  - fit_prior = [True, False]

Best params: {'alpha': 0.1, 'fit_prior': True}



Hình 15: Confusion Matrix của mô hình Naïve Bayes trên 2 nhãn.



Hình 16: AUC – ROC của mô hình Naïve Bayes trên 3 nhãn.

V. KẾT QUẢ NGHIÊN CỨU

Sau khi thực hiện những phân loại với những mô hình học máy được nêu ở trên, chúng em kiểm tra hiệu năng của mô hình sử dụng những thước đo như accuracy score, recall, precision và f1-score:

Thuật toán máy học	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.81	0.79	0.81	0.79
Naïve Bayes	0.79	0.79	0.79	0.75

Bảng 1: Bảng đo hiệu suất những thuật toán học máy (với phân loại 2 nhãn)

Thuật toán máy học	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.78	0.71	0.78	0.72
Naïve Bayes	0.77	0.68	0.77	0.7

Bảng 2: Bảng đo hiệu suất những thuật toán học máy (với phân loại 3 nhãn)

Qua bảng đo hiệu suất ở trên, ta có thể rút ra những nhận xét về từng mô hình như sau:

• Logistic Regression: Thuật toán Logistic Regression đạt được mức độ chính xác (accuracy) cao nhất trong số các thuật toán được thử nghiệm. Độ chính xác lần lượt đạt 81% và 78% khi phân loại trên 2 nhãn và 3 nhãn. Trong khi đó về độ phủ (recall) thì khi phân loại theo cả 2 cách đều đạt được chỉ số khá tốt.

• Naïve Bayes: Thuật toán Naïve Bayes đạt được mức độ chính xác (accuracy) thấp hơn so với Logistic Regression khi được thử nghiệm. Độ chính xác lần lượt đạt 79% và 77% khi phân loại trên 2 nhãn và 3 nhãn. Độ phủ thì tất nhiên sẽ thấp hơn Logistic Regression tuy nhiên không nhiều.

Tổng quan, các thuật toán máy học đều đạt được mức độ chính xác khá cao, tuy nhiên việc tối ưu tham số không làm tăng lên hiệu suất cho các mô hình quá nhiều. Có thể do dữ liệu huấn luyện không đủ, tham số khởi tạo không tốt hoặc bị overfiting. Vấn đề này nhóm sẽ tiến hành giải quyết trong tương lai gần.

VI. TỔNG KẾT

Trong nghiên cứu này, chúng tôi đã tập trung vào bài toán phân tích cảm xúc dựa trên tập dữ liệu bình luận sách từ trang Tiki.vn. Mục tiêu của chúng tôi là sử dụng kỹ thuật xử lý ngôn ngữ tự nhiên (NLP) kết hợp với các mô hình học máy như Logistic Regression và Na¨ıve Bayes để xây dựng một mô hình phân loại các bình luận thành hai nhãn: tích cực (positive), trung tính (neutral) tiêu cực (negative).

Chúng tôi đã sử dụng phương pháp TF-IDF để trích xuất đặc trưng từ văn bản và áp dụng các mô hình học máy để huấn luyện và đánh giá mô hình. Thước đo được sử dụng để đánh giá mô hình là accuracy score, precision, recall và f1-score.

Kết quả cho thấy cả hai phương pháp học máy và xử lý ngôn ngữ tự nhiên đều cho kết quả phân loại chính xác và có hiệu suất cao trong việc phân loại đánh giá sách. Mô hình học máy Logistic Regression và Na¨ıve Bayes cho hiệu suất khá tốt. Tuy nhiên nhóm gặp vấn đề ở phần tối ưu mô hình và sẽ cố gắng hoàn thành sớm nhất có thể.

Trong tương lai, có một số hướng phát triển tiềm năng cho nghiên cứu này. Đầu tiên, có thể tăng cường các kỹ thuật xử lý ngôn ngữ tự nhiên bằng cách sử dụng các phương pháp khác nhau như word embedding, mô hình ngôn ngữ tiên đoán (pre-trained language models) để cải thiện hiệu suất phân loại. Thứ hai, có thể thử nghiệm với các mô hình học máy khác nhau hoặc sử dụng mô hình học sâu (deep learning) để đạt được kết quả tốt hơn.

TÀI LIỆU THAM KHẢO

[1] GS. Hồ Tú Bảo. Đề tài KC01.01/06-10 "Nghiên cứu phát triển một số sản phẩm thiết yếu về xử lí tiếng nói và văn bản Tiếng Việt' (VLSP).

[2] Vũ Anh (anhv.ict91@gmail.com), từ điển mở tiếng Việt Underthesea, https://github.com/undertheseanlp/dictionary.

[3] Wei-Yin Loh. 2011. Classification and regression trees. WIREs Data Mining and Knowledge Discovery, 1(1):14–23.

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

[5] William Uther, Dunja Mladenic, Massimiliano Ciaramita, Bettina Berendt, Aleksander Kołcz, Marko Grobelnik, Dunja Mladenic, Michael Witbrock, John Risch, Shawn Bohn, and et al. 2011. Tf–idf. Encyclopedia of Machine Learning, page 986–987.

BẢNG PHÂN CÔNG ĐÁNH GIÁ THÀNH VIÊN

Họ và tên	MSSV	Phân công	Đánh giá
Phạm Lê Thành Phát	21521262	- Thu thập dữ liệu - Tạo ngữ liệu, tách từ - Tiền xử lí dữ liệu - Viết mô hình và kiểm nghiệm - Viết báo cáo	Hoàn thành tốt
Lê Tuấn Đạt	21520699	- Thu thập dữ liệu - Tạo ngữ liệu, tách từ - So sánh kết quả giữa các mô hình, kết luận - Viết báo cáo, làm slide	Hoàn thành tốt