TRƯỜNG ĐẠI HỌC KINH TẾ - LUẬT

KHOA HỆ THỐNG THÔNG TIN

BÁO CÁO CUỐI KỲ MÔN HỌC

MÔN: PHƯƠNG PHÁP NGHIÊN CỨU LIÊN NGÀNH

MÃ HỌC PHẦN: 231PP0801

ỨNG DỤNG MACHINE LEARNING ĐỂ PHÂN TÍCH SẮC THÁI BÌNH LUẬN

Giảng viên hướng dẫn: TS. Nguyễn Thôn Dã

Nhóm sinh viên thực hiện:

Nguyễn Hữu Lộc	K224060791 (Leader)
Huỳnh Ngọc Châu	K224060770
Dương Mai Hân	K224060782
Trần Huỳnh Khánh Linh	K224060789
Lê Thị Mỹ Tuyến	K224060823
Trần Nhất Quý Xinh	K224060824

TP.Hồ Chí Minh, tháng 12 năm 2023

Lời cảm ơn của nhóm

Lời đầu tiên, chúng em xin cảm ơn ban lãnh đạo của Khoa Hệ thống thông tin đã sắp xếp cho chúng em một môn học ý nghĩa và hữu ích này để chúng em có thể cơ hội tìm hiểu và tiếp thu nhiều kiến thức về môn Phương pháp nghiên cứu liên ngành.

Chúng em muốn gửi lời cảm ơn chân thành đến thầy Nguyễn Thôn Dã – giảng viên giảng dạy môn này, vì đã cho nhóm được tiếp cận đề tài về Machine Learning rất hay cùng với sự giúp đỡ nhiệt tình và tận tâm để nhóm có thể hoàn thành bài báo cáo một cách tốt nhất.

Tuy nhiên, vì kiến thức và kinh nghiệm vẫn còn nhiều hạn chế, nhóm sẽ có những sai sót trong quá trình hoàn thiện bài báo cáo đồ án cuối kỳ. Chúng em luôn mong đợi và tiếp thu những đóng góp ý kiến từ thầy.

Nhóm sinh viên thực hiện

BẢNG ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH ĐỒ ÁN CUỐI KỲ

STT	Họ và tên	MSSV	Công việc	Mức độ hoàn thành
1	Nguyễn Hữu Lộc	K224060791	- Code chính; - Thuyết trình báo cáo; - Nội dung: tóm tắt, giới thiệu và kết luận.	100%
2	Huỳnh Ngọc Châu	K224060770	- Nội dung: MultinomialNB và quy trình thực hiện; - Làm powerpoint.	100%
3	Dương Mai Hân	K224060782	-Nội dung: SVM và Tiền xử lý dữ liệu; - Làm powerpoint.	100%
4	Lê Thị Mỹ Tuyến	K224060823	- Ý tưởng đề tài; - Nội dung: Decision Classifier Tree và Khám phá dữ liệu; - Làm powerpoint.	100%
5	Trần Nhất Quý Xinh	K224060824	- Nội dung: Đánh giá kết quả mô hình; - Làm powerpoint	100%
6	Trần Huỳnh Khánh Linh	K224060789	- Thuyết trình báo cáo; - Tổng hợp, chỉnh sửa và hoàn chỉnh nội dung.	100%

ỨNG DỤNG MACHINE LEARNING ĐỂ PHÂN TÍCH SẮC THÁI BÌNH LUẬN

Nhóm sinh viên[1]

Keyword:

- Thương mại điện tử
- Hành vi khách hàng
- Thuật toán
- Học máy

TÓM TẮT

Tính đến thời điểm hiện tại, sự hiểu biết sâu sắc về hành vi mua sắm trực tuyến đang được xem xét là một yếu tố chủ chốt ảnh hưởng trực tiếp đến hiệu suất kinh doanh trong lĩnh vực thương mại điện tử. Mặc dù các nghiên cứu hiện nay chủ yếu tập trung vào việc đo lường ý định mua và doanh số bán hàng, nhưng thực tế vẫn còn nhiều hạn chế trong việc áp dụng những kết quả từ các nghiên cứu đó vào thực tế kinh doanh.

Sự quyết định mua hàng không chỉ đơn thuần xuất phát từ ý định mua mà còn từ trải nghiệm mua hàng thực tế. Chúng tôi đang hướng tới việc tăng cường hiểu biết về hành vi mua sắm trực tuyến bằng cách sử dụng dữ liệu đa chiều từ phản hồi của khách hàng đối với từng sản phẩm cụ thể. Nghiên cứu này nhằm so sánh và đánh giá hiệu quả của việc sử dụng nhiều thuật toán khác nhau trong việc phân tích thái độ khi mua sắm trực tuyến của khách hàng. Chúng tôi tiến hành so sánh khả năng dự đoán giữa các phương pháp học máy khác nhau, từ đó chọn ra các mô hình phù hợp nhất có độ chính xác cao nhất. Phân tích này sẽ hỗ trợ việc thiết kế nền tảng tương tác, mở rộng kiến thức và tạo ra các dự đoán chính xác về hành vi mua sắm trực tuyến trong lĩnh vực thương mại điện tử.

1. Giới thiệu:

Trong những năm gần đây, thương mại điện tử tại Việt Nam đã trải qua một sự phát triển đáng kể. Bộ Công Thương thông báo rằng doanh thu thương mại điện tử bán lẻ tại Việt Nam năm 2022 đã tăng 20% so với năm 2021, đạt 16,4 tỷ USD, chiếm tỷ lệ 7,5% trong tổng doanh thu bán lẻ hàng hóa, dịch vụ trong cả nước. Trong cùng năm, đã có 7.893 doanh nghiệp và tổ chức cũng như 2.609 cá nhân đăng ký tài khoản tham gia thương mại điện tử, thực hiện thông báo cho 10.146 website thương mại điện tử và 660 website cung cấp dịch vụ thương mại điện tử.

Đại dịch COVID-19 đã gây biến đổi đáng kể trong thói quen mua sắm khi người tiêu dùng chuyển hướng sang các kênh bán hàng trực tuyến một cách đột ngột. Việc giao hàng và đổi trả dễ dàng, vận chuyển miễn phí nhanh chóng trên hầu hết các trang web thương mại điện tử, cùng với khả năng mua sắm mọi thứ từ nhà một cách thuận tiện, đã thúc đẩy sự tăng trưởng của mua sắm trực tuyến trong đại dịch này.

[1] Nhóm sinh viên gồm 6 thành viên: Hữu Lộc, Ngọc Châu, Mai Hân, Khánh Linh, Mỹ Tuyến, Quý Xinh

Áp lực này đang thúc đẩy các doanh nghiệp truyền thống tại Việt Nam không ngừng nâng cao hoạt động bán lẻ trực tuyến và cải thiện trải nghiệm mua sắm trực tuyến cho khách hàng. Việc mua sắm online diễn ra liên tục 24/7, không bị ràng buộc thời gian, đã thay đổi cách thức mua sắm của cả khách hàng hiện có và mới.

Tương tác của khách hàng trên các nền tảng trực tuyến đóng vai trò then chốt trong việc cung cấp thông tin chi tiết giúp doanh nghiệp hiểu rõ tâm trạng của khách hàng khi mua sản phẩm của mình.

Các doanh nghiệp tại Việt Nam đang thấy đây là cơ hội quý báu để nắm bắt thông tin sâu rộng về hành trình mua sắm và hành vi mua sắm của khách hàng thông qua việc xác định và hiểu rõ về vai trò của khách hàng và nền tảng trực tuyến, cũng như cách tương tác của họ ảnh hưởng đến việc mua sắm thực tế. Sự tiến bộ trong việc dự đoán quyết định mua hàng cá nhân hoặc nhóm đang có tác động lớn tới doanh nghiệp và thị trường thương mại điện tử tại Việt Nam.

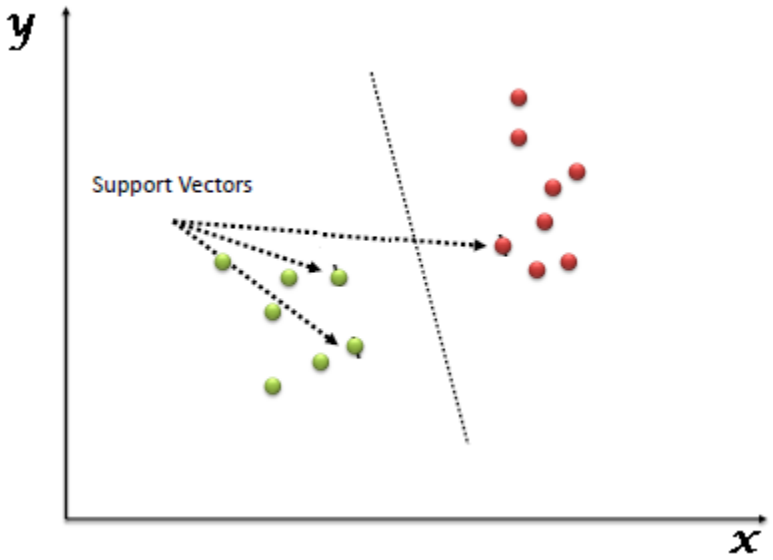
Nghiên cứu về hành vi của khách hàng được thực hiện thông qua việc thu thập dữ liệu từ các kênh, đánh giá đặc điểm của các nhận xét. Tuy nhiên, các nhận xét này thường chứa nhiều lỗi như: sai chính tả, ngữ pháp, teencode, viết tắt,... Vì vậy, việc xử lý và làm sạch dữ liệu là quan trọng sau đó để từ đó, có thể tiếp tục áp dụng các mô hình thuật toán, đánh giá từng mô hình và đưa ra giải pháp phù hợp nhất.

2. Mô hình áp dụng

2.1 SVM(Support vector machine)

❖ *Lý thuyết:* (Stecanella, 2017)

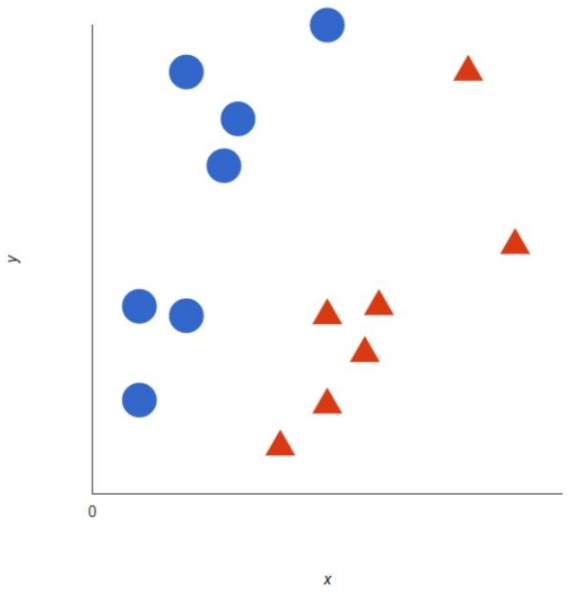
SVM là một thuật toán giám sát, nó có thể sử dụng cho cả việc phân loại hoặc đệ quy. Tuy nhiên nó được sử dụng chủ yếu cho việc phân loại. Trong thuật toán này, chúng ta vẽ đồ thị dữ liệu là các điểm trong n chiều (ở đây n là số lượng các tính năng bạn có) với giá trị của mỗi tính năng sẽ là một phần liên kết. Sau đó chúng ta thực hiện tìm "đường bay" (hyper-plane) phân chia các lớp. Hyper-plane nó chỉ hiểu đơn giản là 1 đường thẳng có thể phân chia các lớp ra thành hai phần riêng biệt.



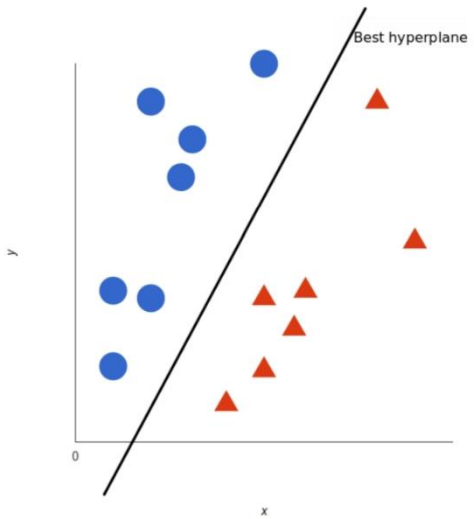
Support Vectors hiểu một cách đơn giản là các đối tượng trên đồ thị tọa độ quan sát, Support Vector Machine là một biên giới để chia hai lớp tốt nhất.

❖ *Quy trình hoạt động:* (Stecanella, 2017)

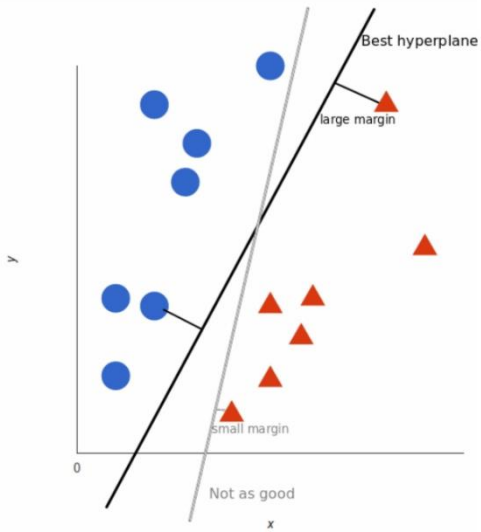
Hãy tưởng tượng có hai thẻ: red và blue và dữ liệu có hai tính năng: x và y. Một trình phân loại với một cặp tọa độ (x, y) sẽ xuất ra nếu nó có màu đỏ hoặc xanh lam, vẽ biểu đồ dữ liệu đào tạo đã được dán nhãn của trên một mặt phẳng:



Máy vectơ hỗ trợ lấy các điểm dữ liệu này và xuất ra siêu phẳng (trong hai chiều, nó chỉ đơn giản là một đường) để phân tách các thẻ tốt nhất. Đường này là ranh giới quyết định: bất cứ thứ gì nằm về một bên của nó sẽ phân loại là màu xanh lam và bất cứ thứ gì nằm về phía bên kia là màu đỏ.

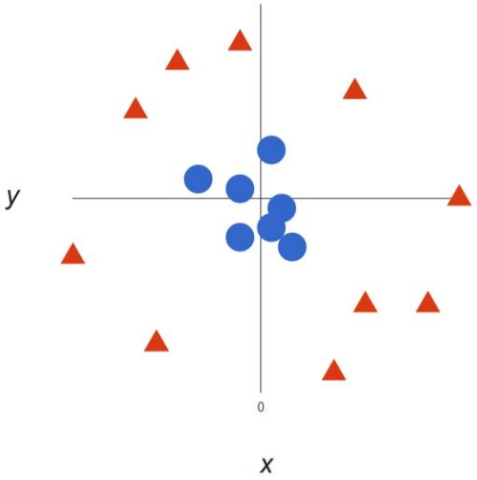


Đối với SVM, đây là thẻ tối đa hóa lợi nhuận từ cả hai thẻ. Nói cách khác: siêu phẳng (hãy nhớ trong trường hợp này là một đường thẳng) có khoảng cách đến phần tử gần nhất của mỗi thẻ là lớn nhất.



- Dữ liệu phi tuyến

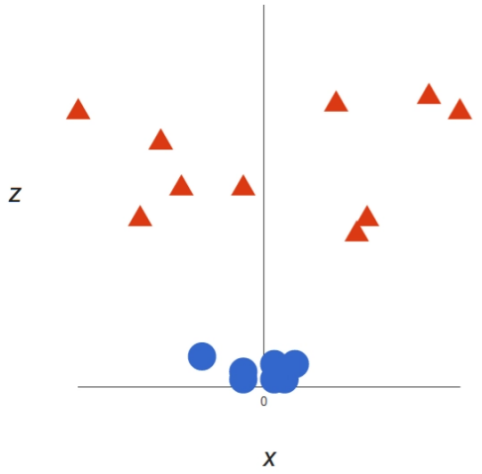
Dữ liệu có thể phân tách tuyến tính - có thể vẽ một đường thẳng để phân tách màu đỏ và màu xanh lam. Tuy nhiên không phải lúc nào cũng dễ dàng, có những lúc dữ liệu phức tạp hơn, dưới đây là một trong những trường hợp:



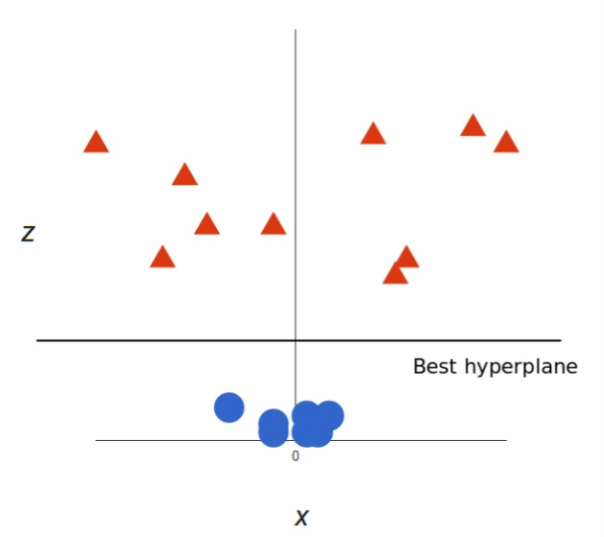
Khá rõ ràng là không có ranh giới quyết định tuyến tính (một đường thẳng ngăn cách cả hai thẻ). Tuy nhiên, các vectơ được phân tách rất rõ ràng.

Vì vậy hãy thêm chiều thứ ba. Bình thường sẽ có hai chiều: x và y. Tạo một thứ nguyên z mới và quy định rằng nó phải được tính theo một cách nhất định sẽ thuận tiện hơn: $z = x^2 + y^2$.

Điều này sẽ cho một không gian ba chiều. Lấy một phần không gian đó, ta sẽ có:

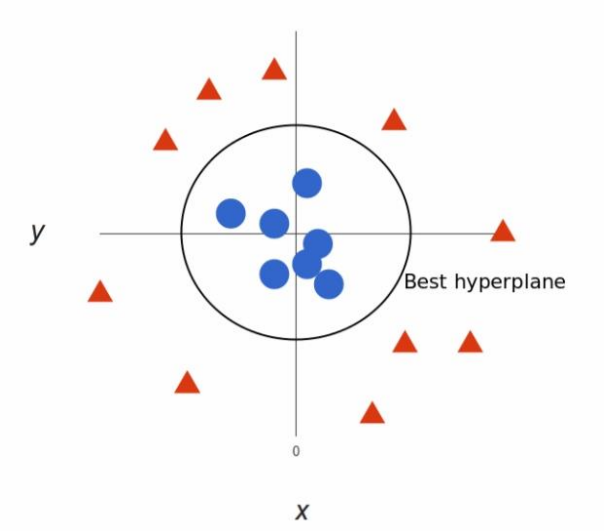


Sau đó thuật toán sẽ làm tiếp:



Đây là không gian ba chiều nên siêu phẳng là một mặt phẳng song song với trục x tại một z nhất định (giả sử z = 1).

Những gì còn lại là ánh xạ nó trở lại hai chiều:



❖ *Ưu điểm, nhược điểm*

SVM – một thuật toán học máy mạnh mẽ, có những ***ưu điểm*** đáng chú ý sau:

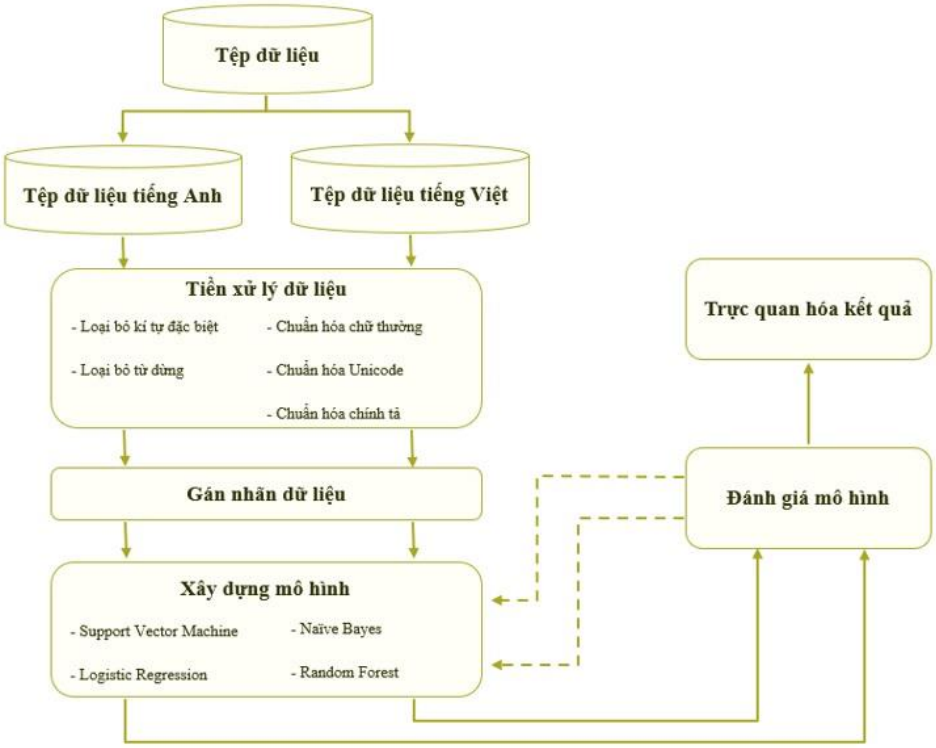
- Hiệu quả trong không gian nhiều chiều: SVM thể hiện độ hiệu quả cao trong việc xử lý dữ liệu chiều cao, nơi số lượng tính năng vượt quá số lượng mẫu. Ngay cả khi số lượng đặc trưng lớn, SVM vẫn hoạt động tốt và có khả năng xử lý dữ liệu nhiều chiều một cách hiệu quả, làm cho nó phù hợp với các ứng dụng có nhiều đặc trưng.
- Linh hoạt: SVM có khả năng áp dụng cho cả bài toán phân loại và hồi quy. Sự hỗ trợ của nhiều hàm hạt nhân khác nhau mang lại linh hoạt, giúp mô hình nắm bắt mối quan hệ phức tạp trong dữ liệu. Điều này làm cho SVM có thể sử dụng cho nhiều loại công việc khác nhau.
- Hiệu quả trong trường hợp dữ liệu hạn chế: SVM có thể hoạt động hiệu quả ngay cả khi tập dữ liệu huấn luyện nhỏ. Sự sử dụng vectơ hỗ trợ đảm bảo chỉ một số điểm dữ liệu quan trọng ảnh hưởng đến ranh giới quyết định, điều này làm cho SVM phù hợp khi dữ liệu hạn chế.

Nhược điểm của SVM: mặc dù SVM được rộng rãi ưa chuộng vì những lợi ích đã được đề cập, nhưng nó cũng mang theo một số hạn chế và thách thức:

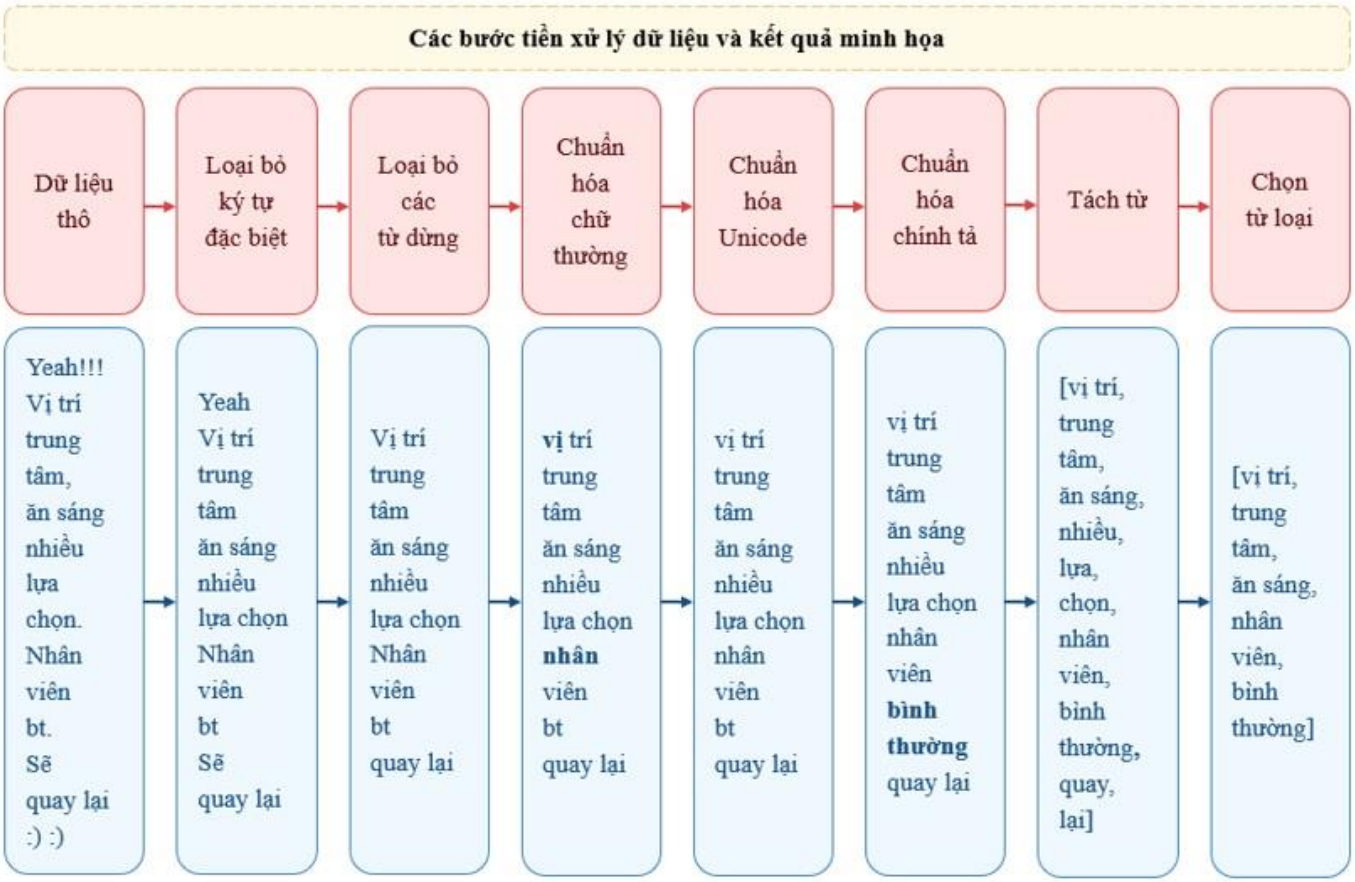
- Tính toán phức tạp: SVM có thể đòi hỏi nhiều tính toán, đặc biệt khi đối mặt với các tập dữ liệu lớn. Thời gian huấn luyện và yêu cầu bộ nhớ tăng lên đáng kể theo kích thước của tập dữ liệu huấn luyện.
- Nhạy cảm với tham số: SVM bao gồm các tham số như tham số chính quy hóa và lựa chọn hàm kernel. Hiệu suất của SVM có thể phụ thuộc nhiều vào cách thiết lập các tham số này. Việc điều chỉnh không đúng có thể dẫn đến kết quả không tối ưu hoặc thời gian huấn luyện lâu dài.
- SVM có thể gặp vấn đề về khả năng mở rộng khi áp dụng cho các tập dữ liệu cực kỳ lớn. Việc đào tạo một SVM trên hàng triệu mẫu có thể trở nên không thực tế do hạn chế về bộ nhớ và tính toán.

❖ *Áp dụng cho bài toán sentiment analysis*

Bức tranh tổng quan về mô hình nghiên cứu được thể hiện qua ảnh dưới đây. Nguồn dữ liệu cho nghiên cứu được tổng hợp từ một nền tảng trực tuyến. Dữ liệu thô, sau khi thu thập, trải qua quá trình tiền xử lý để đảm bảo chất lượng và độ chính xác khi xây dựng mô hình. Bước tiếp theo là việc gán nhãn cho dữ liệu, và sau đó, nghiên cứu sẽ tiến hành xây dựng mô hình máy học để phân tích quan điểm. Việc đánh giá độ chính xác của mô hình cũng được thực hiện để đảm bảo hiệu suất cao nhất. Các bước này sẽ được lặp lại một cách liên tục để đạt được mức hiệu suất tối đa. Cuối cùng, các kết quả từ mô hình sẽ được trực quan hóa thông qua việc sử dụng các biểu đồ. (Hồ & al, 2023)



- *Tiền xử lý:* đối với mô hình máy học, việc làm sạch dữ liệu đóng vai trò quan trọng. Tập dữ liệu thô có thể chứa thông tin không liên quan, ảnh hưởng đến hiệu suất mô hình. Nghiên cứu sẽ thực hiện quá trình làm sạch dữ liệu theo sơ đồ quy trình. (Hồ & al, 2023)



- Loại bỏ kí tự đặc biệt: Là những ký tự không thuộc bảng chữ cái, là những siêu liên kết, khoảng trắng và dấu câu,...
- Loại bỏ các từ dừng và những từ cần thiết để hình thành câu nhưng không ảnh hưởng đến giá trị quan điểm của câu.
- Đưa chữ hoa về chữ thường để tránh trùng ý.
- Chuyển đổi ký tự Unicode tổ hợp thành Unicode dựng sẵn vì chúng có bộ mã hoàn toàn khác nhau, dù những ký tự này giống nhau.
- Chuẩn hóa chính tả và chuyển đổi các từ viết tắt thành một hình thức chuẩn.
- Tách từ: tiến hành tách từ để phân tích.
- Chọn từ loại: tiến hành trích xuất lại các từ loại để tiến hành huấn luyện nhằm tăng độ chính xác của mô hình.

Nghiên cứu hiện tại không quan tâm đến trọng số của các từ khóa, với tất cả được xem xét như nhau. Điều này có thể coi là một thiếu sót trong nghiên cứu, góp phần làm giảm độ chính xác của mô hình. Các nghiên cứu tiếp theo có thể tập trung vào việc đánh giá và xem xét cả trọng số và tần suất xuất hiện của từ khóa trong mỗi bình luận để cải thiện khía cạnh này. (Hồ & al, 2023)

- Gán nhãn dữ liệu

Nghiên cứu này tập trung vào việc phân tích ý kiến thông qua kỹ thuật học máy có giám sát. Sau quá trình huấn luyện và đánh giá mô hình, chúng ta thu được một mô hình có khả năng phân loại quan điểm trong các bài đánh giá chưa được gán nhãn. Bằng cách thực hiện đầy đủ và theo thứ tự các bước tiền xử lý dữ liệu như đã mô tả trước đó, nghiên cứu đã tạo ra một tập dữ liệu sạch để tiến hành phân tích và biểu thị thông qua đồ thị. (Hồ & al, 2023)

2.2 Decision Tree Classifier

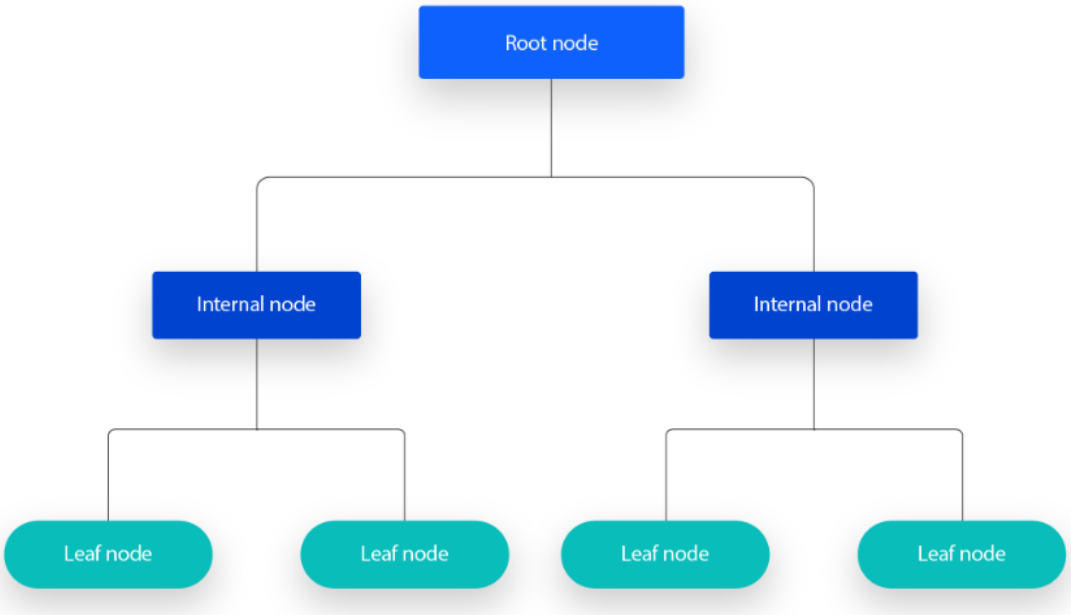
Decision Tree Classifier là một thuật toán học máy được sử dụng phổ biến trong các bài toán phân loại (Classification). Nó xây dựng một cây quyết định từ dữ liệu huấn luyện, trong đó mỗi nút trong cây đại diện cho một thuộc tính và mỗi cành đại diện cho một quy tắc quyết định.

❖ *Lý thuyết*

Là một mô hình học máy phân loại và dự đoán dựa trên việc xây dựng một cây quyết định logic. Mô hình này sử dụng các quy tắc logic để phân loại dữ liệu vào các lớp khác nhau. Cây quyết định được xây dựng bằng cách chia tập dữ liệu thành các phân vùng nhỏ dựa trên các thuộc tính và giá trị của chúng. Quá trình này tiếp tục đến khi một điều kiện dừng được đáp ứng, ví dụ như không còn các điểm dữ liệu nào trong một phân vùng hoặc không còn thuộc tính nào để chia. (Staff, 2023)

Cấu trúc model:

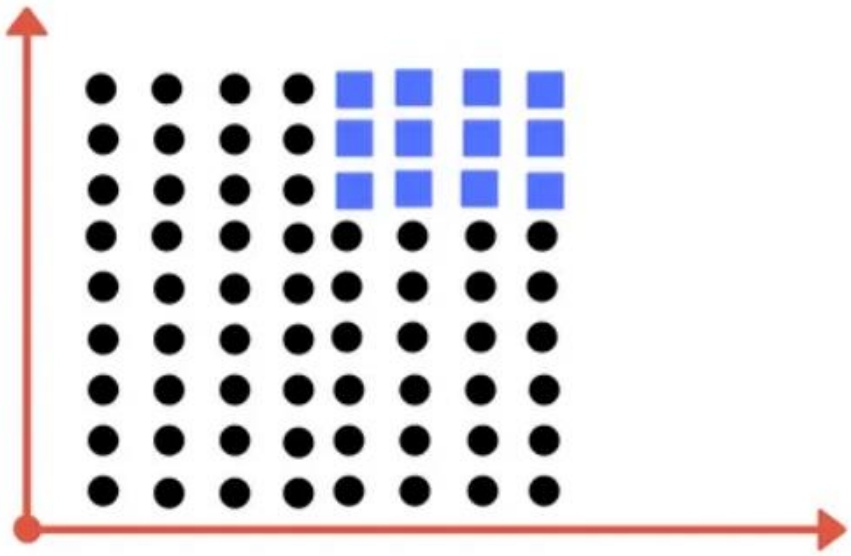
Nó có cấu trúc cây phân cấp, bao gồm nút gốc, nhánh, nút bên trong và nút lá.



Decision Tree bắt đầu bằng nút gốc, không có bất kỳ nhánh nào đến. Các nhánh đi từ nút gốc sau đó sẽ đi vào các nút bên trong, còn được gọi là nút quyết định. Dựa trên các tính năng có sẵn, cả hai loại nút đều tiến hành đánh giá để tạo thành các tập hợp con đồng nhất, được biểu thị bằng các nút lá hoặc nút cuối. Các nút lá đại diện cho tất cả các kết quả có thể xảy ra trong tập dữ liệu. (Staff, 2023)

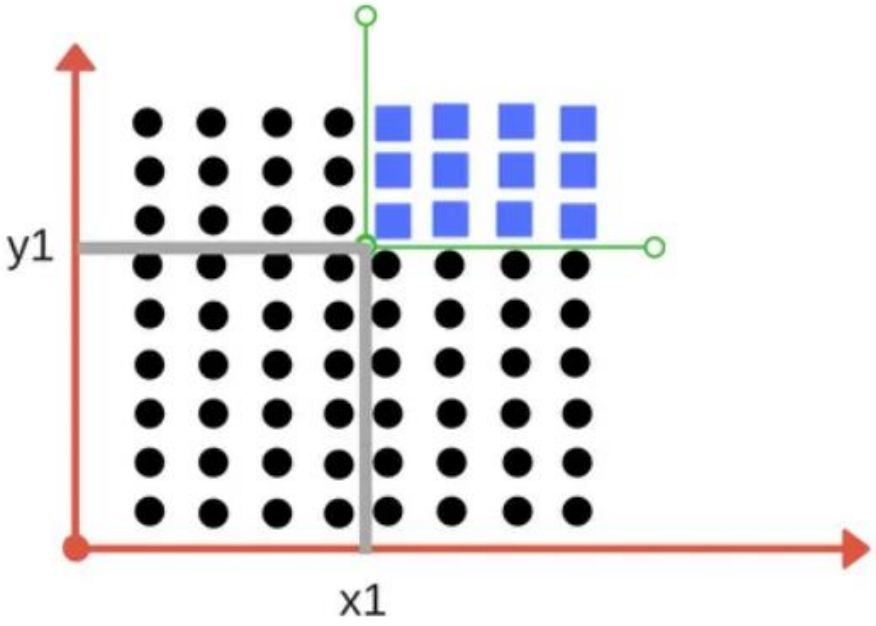
❖ *Quy trình hoạt động:*

Giả sử chúng ta có biểu đồ sau cho hai lớp được biểu thị bằng hình tròn màu đen và hình vuông màu xanh lam. Có thể vẽ một đường phân cách duy nhất?



Bạn có thể vẽ đường phân chia đơn cho các lớp này không?

Chúng ta sẽ cần nhiều hơn một dòng để chia thành các lớp. Một cái gì đó tương tự như hình ảnh sau đây:

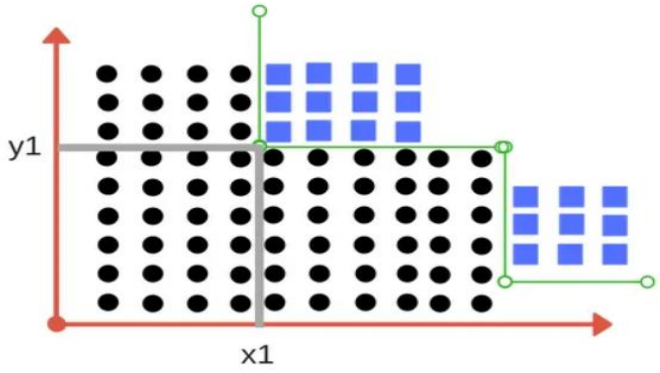


Chúng ta cần hai dòng một cho ngưỡng x và ngưỡng cho y.

Ở đây chúng ta cần hai dòng, một dòng phân tách theo giá trị ngưỡng x và dòng còn lại dành cho giá trị ngưỡng y.

➔Như vậy ta đã đoán được Decision tree classifier sẽ cố gắng thực hiện điều gì.

Trình phân loại cây quyết định, lặp đi lặp lại việc chia khu vực làm việc (sơ đồ) thành phần phụ bằng cách xác định các đường (lặp đi lặp lại vì có thể có hai vùng xa nhau cùng loại được chia cho nhau như trong hình bên dưới).



❖ *Decision Tree Classifier trong sentiment analysis:*

Decision Tree Classifier có thể được áp dụng trong sentiment analysis bằng cách sử dụng các đặc trưng của văn bản để phân loại chúng thành các lớp cảm xúc khác nhau chẳng hạn như tích cực, tiêu cực hoặc trung lập. Các đặc trưng có thể bao gồm từ khóa, đếm từ, phân phối tần suất từ, hoặc các đặc trưng ngữ nghĩa khác. Quá trình xây dựng Decision Tree sẽ tìm kiếm các đặc trưng tốt nhất để phân chia dữ liệu thành các lớp cảm xúc. Cụ thể:

Thu thập và chuẩn bị tập dữ liệu với các văn bản đã được gán nhãn theo các lớp cảm xúc. Dữ liệu này bao gồm các mẫu văn bản và nhãn cảm xúc tương ứng.

Biểu diễn các văn bản thành các vectơ đặc trưng để Decision Tree có thể xử lý. Các phương pháp như Bag-of-Words, TF-IDF được sử dụng để chuyển đổi các văn bản thành vectơ số.

Chọn đặc trưng phân chia tốt nhất để tách dữ liệu thành các nhánh con. Đặc trưng này được chọn dựa trên mức độ tách biệt giữa các lớp cảm xúc. Một số phương pháp phổ biến để đo lường mức độ tách biệt bao gồm hệ số Gini, thông tin độ, hoặc hàm entropy.Sau khi chọn đặc trưng phân chia, tính toán giá trị phân chia của nó. Giá trị phân chia đo lường mức độ tách biệt mà đặc trưng phân chia đem lại khi áp dụng nó vào dữ liệu. Giá trị phân chia cao hơn cho thấy đặc trưng phân chia tốt hơn.

Tiếp theo, dữ liệu được tách thành các nhánh con dựa trên giá trị của đặc trưng phân chia. Mỗi nhánh con tương ứng với một giá trị của đặc trưng và chứa các mẫu dữ liệu có giá trị tương ứng đó.

Quá trình xây dựng mô hình được lặp lại trên mỗi nhánh con để tìm các đặc trưng phân chia tiếp theo. Quá trình này tiếp tục cho đến khi một điều kiện dừng được đáp ứng, ví dụ như đạt đến độ sâu tối đa hoặc không còn đặc trưng phân chia nào tốt hơn.

Khi quá trình xây dựng cây hoàn thành, các nút lá được định nghĩa và gán nhãn với lớp cảm xúc tương ứng. Các mẫu dữ liệu trong cùng một nhánh con sẽ được phân loại vào cùng một lớp cảm xúc.

Sau khi cây quyết định được xây dựng, bạn cần đánh giá hiệu suất của nó bằng cách sử dụng tập kiểm tra hoặc các phương pháp đánh giá khác. Nếu cây quá khớp hoặc không đạt đủ hiệu suất, ta có thể áp dụng các biện pháp như cắt tỉa (pruning) hoặc điều chỉnh các siêu tham số để cải thiện mô hình.

Quá trình xây dựng Decision Tree trong sentiment analysis có sự tương đồng với quá trình xây dựng Decision Tree trong các bài toán khác. Tuy nhiên, điểm khác biệt chính là trong sentiment analysis, đặc trưng phân chia được chọn dựa trên mức độ tách biệt giữa các lớp cảm xúc. Mục tiêu là tìm các đặc trưng mà có khả năng phân chia tốt giữa các mẫu dữ liệu tích cực, tiêu cực và trung tính.

❖ *Ưu điểm của Decision Tree Classifier trong sentiment analysis:*

- Dễ hiểu và diễn giải: Cây quyết định tạo ra các quy tắc dễ hiểu và có thể được diễn giải, giúp người dùng hiểu rõ lý do quyết định phân loại.
- Xử lý dữ liệu phi tuyến: Cây quyết định có khả năng xử lý dữ liệu phi tuyến mà không cần phải thực hiện biến đổi đặc trưng.
- Tính toán đơn giản: So với một số mô hình phức tạp khác, cây quyết định có thể được xây dựng và dự đoán nhanh chóng.

❖ *Nhược điểm của Decision Tree trong sentiment analysis:*

- Dễ bị quá khớp (dễ dẫn đến hiện tượng overfitting): Cây quyết định có thể dễ dàng bị quá khớp dữ liệu huấn luyện, làm giảm khả năng tổng quát hóa cho dữ liệu mới.

- Nhạy cảm với sự thay đổi nhỏ: Một số thay đổi nhỏ trong dữ liệu đầu vào có thể dẫn đến sự thay đổi lớn trong cấu trúc của cây quyết định, làm cho nó không ổn định và khó diễn giải.
- Khó xử lý dữ liệu không hoàn hảo: Cây quyết định có thể không hiệu quả khi xử lý dữ liệu có nhiễu hoặc thiếu, và cần phải thực hiện các phương pháp xử lý dữ liệu trước khi xây dựng cây.

2.3 MultinomialNB

❖ *Lý thuyết và quy trình thực hiện:*

Việc sắp xếp tài liệu tự động ngày càng trở nên quan trọng vì xử lý và sắp xếp tài liệu thủ công tốn nhiều thời gian và không phải là một giải pháp khả thi do số lượng tài liệu rất lớn. Mô hình Naive Bayes là một mô hình rất nổi tiếng để phân loại văn bản hiệu quả, nhanh chóng.

Multinomial Naive Bayes (MultinomialNB) là mô hình phân loại theo xác suất dựa trên định lý Bayes:

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)},$$

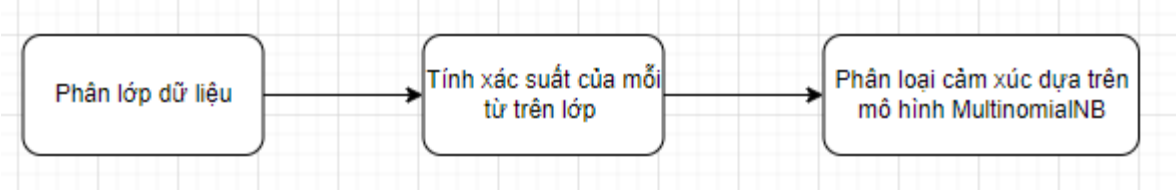
Naive Bayes là một trong nhóm các thuật toán áp dụng định lý Bayes với một giả định khá ngây thơ - đúng nghĩa đen của từ Naïve, rằng mọi features đầu vào đều độc lập với nhau. Features ở đây có thể được hiểu là danh sách các biến đầu vào: độ tuổi, giới tính, mức lương, tình trạng hôn nhân, ... Ví dụ 2 biến độc lập là: size giày và giới tính của bạn. Ví dụ 2 biến phụ thuộc là: số tiền quảng cáo bỏ ra và doanh số thu được. (Abbas & al, 2019)

❖ *Ưu điểm và nhược điểm của MultinomialNB*

- Mô hình Multinomial Naive Bayes (MultinomialNB) mang đến nhiều ưu điểm cho các ứng dụng phân loại văn bản. Đầu tiên, nó dễ triển khai và đào tạo, đặc biệt là với các tập dữ liệu lớn. Hiệu suất của mô hình thường rất tốt khi áp dụng vào các tác vụ phân loại văn bản, đặc biệt là khi có nhiều lớp và mẫu dữ liệu. Mô hình cũng có khả năng làm mới với dữ liệu mới mà không yêu cầu đào tạo lại toàn bộ mô hình, giúp nó duy trì tính linh hoạt trong môi trường thay đổi.
- Tuy nhiên, MultinomialNB không phải là một giải pháp hoàn hảo và nó mang theo một số nhược điểm. Mô hình dựa trên giả định về sự độc lập giữa các đặc trưng, điều này thường không phản ánh đúng thực tế, gây giảm hiệu suất khi dữ liệu không tuân theo giả định này. Nó cũng không xử lý được thông tin về thứ bậc giữa các từ trong văn bản và yếu đối với các từ hiếm xuất hiện. Điều này làm cho việc chọn mô hình phụ thuộc vào đặc tính cụ thể của tập dữ liệu và yêu cầu tiền xử lý dữ liệu kỹ lưỡng.

❖ *Áp dụng Multinomial naive bayes vào Sentiment analysis:* (Abbas & al, 2019)

Mô hình phân tích cảm xúc



Bước 1: Phân phối lớp (Class Distribution):

Sử dụng ký hiệu π_C để biểu diễn tỷ lệ văn bản thuộc lớp c.

$$\pi_c = \frac{class_c}{\sum_{n=1}^{N} class_n}$$

Bước 2: Tính xác suất của mỗi từ theo lớp (Probability of each word per class):

Bước 3: Bộ phân loại Naive Bayes Multinomial (Multinomial Naive Bayes Classifier):

- Sử dụng phân phối xác suất và tỷ lệ văn bản thuộc mỗi lớp để tính xác suất tổng cộng cho mỗi văn bản.
- Áp dụng log để tránh tình trạng underflow khi thực hiện nhân nhiều giá trị nhỏ.
- Thêm một yếu tố IDF (Inverse Document Frequency) để xem xét tần suất xuất hiện của các từ trong toàn bộ tập dữ liệu.

Bước 4: Smoothing và xử lý stop words:

- Sử dụng Laplace Smoothing để xử lý các từ không xuất hiện trong dữ liệu huấn luyện.
- Thêm IDF để xử lý stop words và làm cho mô hình chính xác hơn.

Bước 5: Biểu thức toàn diện của mô hình (Optimal Model):

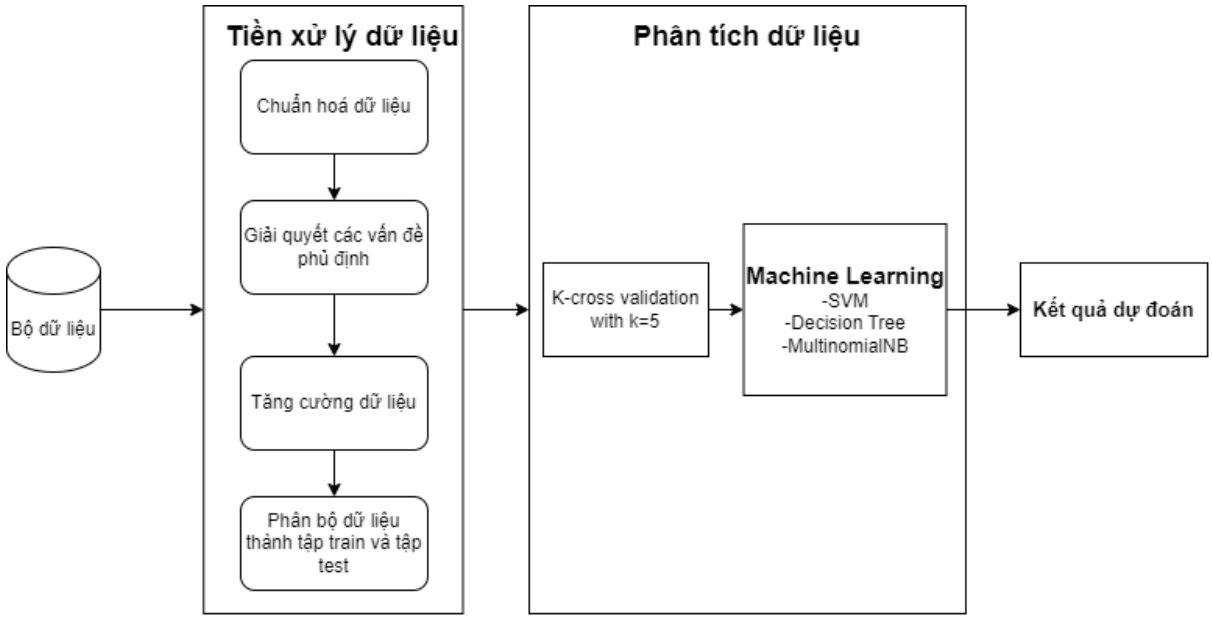
- Kết hợp các yếu tố trên để định nghĩa mô hình Naive Bayes Multinomial cuối cùng, được biểu diễn bằng công thức:

$$\Pr(c) = \log \pi_c + \prod_{w=1}^{|V|} \log(1 + f_w) \log(\Pr(w|c))$$

3. Quá trình thực hiện

3.1 Quy trình thực hiện

Sơ đồ thực hiện bắt đầu từ xử lý dữ liệu cho đến đánh giá mô hình



3.2 Khám phá dữ liệu

❖ *Mô tả dữ liệu*

Dữ liệu sentiment analysis mẫu được lấy từ các nguồn như comment trên sàn thương mại điện tử và bình luận trên mạng xã hội,... Dữ liệu thu thập ở dạng text bao gồm mã số và phần nội dung chưa được chuẩn hóa chứa các từ tiếng anh, chuyên ngành, sai chính tả, teencode, và đặc biệt chứa khá nhiều các emoji cảm xúc. Quan sát dữ liệu thấy có khá nhiều nhiễu, gán nhãn sai và lấy từ các trang thương mại điện tử nên từ ngữ lộn xộn, thường không theo văn phong chuẩn mực, cần phải có bước chuẩn hóa cần thiết.

Dữ liệu này được sử dụng để phân loại ý kiến của người dùng thành các nhãn tích cực, tiêu cực hoặc trung tính. Mỗi mẫu dữ liệu trong tập dữ liệu bao gồm một câu hoặc một đoạn văn bản ngắn và nhãn tương ứng cho ý kiến được diễn đạt trong đó.

Mục đích chính của dữ liệu này là xây dựng và huấn luyện các mô hình machine learning để tự động phân loại và đánh giá cảm xúc từ các đoạn văn bản tương tự.

❖ *Cấu trúc dữ liệu*

Tập dữ liệu sentiment analysis mẫu có thể được tổ chức dưới dạng một bảng dữ liệu (data frame) với hai cột chính: "Text" và "Label". Cột "Text" chứa các đoạn văn bản hoặc câu được lấy từ các nguồn khác nhau, trong khi cột "Label" chứa nhãn tương ứng cho ý kiến trong đoạn văn bản đó. Nhãn có thể được biểu diễn dưới dạng số nguyên hoặc chuỗi ký tự, ví dụ: 0 cho tích cực, 1 cho tiêu cực và 2 cho trung tính.

❖ *Quy trình chuẩn hóa*

- Loại bỏ các ký tự đặc biệt và dấu câu không cần thiết, chỉ giữ lại các từ và cụm từ quan trọng.
- Chuyển đổi các chữ hoa thành chữ thường để giảm sự phân biệt do viết hoa.
- Loại bỏ các từ không có ý nghĩa như "is", "ừm", "ờ", "thì", v.v. (các stop words).
- Thực hiện việc tách từ (tokenization) để chia các đoạn văn thành các từ riêng biệt.
- Áp dụng quá trình stemming hoặc lemmatization để chuẩn hóa các từ về dạng gốc (ví dụ: "okeyyy" thành "ok").
- Chuyển đổi các icon thành các mức đánh giá độ tích tích tiêu cực như (dưới 3 sao: tiêu cực)
- Chia tập dữ liệu thành tập huấn luyện và tập kiểm tra để đánh giá hiệu suất của mô hình.

3.3 Tiền xử lý dữ liệu

Giải pháp *là tập trung vào dữ liệu hơn mô hình*. Tập trung xử lý dữ liệu, loại bỏ nhiễu, gán nhãn lại các dữ liệu gắn nhãn sai. Lý do thực hiện tập trung vào dữ liệu hơn là vì dữ liệu được phát hiện có nhiều nhiễu khác, gán nhãn sai và lấy từ các trang thương mại điện tử nên từ dịch vụ quét, thường không theo văn phong chuẩn mực, cần phải có bước chuẩn hóa.

Mô hình được sử dụng là SVM và tính năng quen thuộc TF-IDF (5-gram). Lý do sử dụng SVM vì SVM khá phù hợp với các bài toán có ít dữ liệu nhưng nhiều tính năng. Mô hình này vẫn cho kết quả khá tốt, thời gian train/dự đoán khá nhanh. Cuối cùng là giải pháp sử dụng Error Analysis để phân bổ lại các dữ liệu gắn nhãn sai.

Sau đây là các bước trong quá trình làm sạch dữ liệu:

- Đầu tiên, cập nhật các đường dẫn chứa từ điển tích cực, tiêu cực, phủ định để phục vụ cho quá trình phân loại.
- Tiếp theo là xác định các ký tự có dấu trong tiếng Việt
  - Tiếng Việt có 2 cách bỏ dấu, đưa ra 1 chuẩn. Ví dụ chữ "Hoa" và "Hoà" đều được chấp nhận trong tiếng Việt. Ngoài ra còn một số trường hợp chữ chữ lỗi cũng cần chuẩn hóa lại (các trường hợp chữ như: "Giao hàngnhanh" xử lý y sẽ tốt hơn).
- Sau đó viết làm loại bỏ dấu của ký tự đi
- Viết hàm để chuẩn hóa các từ teencode, emoji, sai chính tả, tiếng anh,...sang từ biểu thị sentiment
  - Remove các ký tự kéo dài và chuyển thành chữ bình thường
    Ví dụ: Áo đẹp quáaaaaaaa ➔ Áo đẹp quá.
    Ví dụ: ÁO ➔ áo
  - Chuẩn hóa tiếng Việt, xử lý emoji, chuẩn hóa tiếng anh, thuật ngữ
    Emojs mang ý nghĩa tích cực (positive): '💯','💗' và Emojs mang nghĩa tiêu cực (nagative): '👎','😣'.
    "okie"➔"ok", "okey"➔"ok", authentic➔"chuẩn chính hãng",vv…
  - Loại bỏ các kí tự thừa thãi. Ví dụ: ,,..
    Loại bỏ dấu câu (dấu chấm câu) và các ký tự nhiễu thành khoảng trống.
    Sau đó thực hiện tokennize lại các câu và bỏ dấu "_" trong các token.
- Xử lý vấn đề phủ định

- TF-IDF không thể xử lý lớp phủ vấn đề được định nghĩa trong tình cảm tiền toán.
  Ví dụ: Cái áo này rất đẹp và Cái áo này nghĩ đẹp sẽ không khác nhau nhiều khi chọn tính năng tf-idf, giải pháp của là biến đẹp thành tích cực , hay không tệ thành không tiêu cực bằng cách dùng từ tâm điển và lớp phủ từ điển.
- Nếu có từ phủ định thì thực hiện đổi thành 'notpos' hoặc 'notnag'
- Thêm feature cho những sentiment words vào cuối câu nếu nó không có từ phủ định
  VD: áo này đẹp ➔áo này đẹp positive
  Hay: áo này chẳng đẹp ➔ áo này notpos
- Remove nốt những ký tự thừa thãi như " ", ký tự đặc biệt

- Tạo class với các phương thức để truy cập vào dữ liệu: hàm đọc dữ liệu, hàm mô tả dữ liệu, hàm chuyển đổi dữ liệu, chuẩn hoá dữ liệu:

- Hàm đọc dữ liệu từ file và chia thành các comment riêng biệt của từng khách hàng
- Đối với một khách hàng tạo ra một dictionary bao gồm ID, lable, và phần comment.
- Viết hàm thực hiện kết hợp hai phương thức _load_raw_data và _create_row để chuyển tất cả dữ liệu thành một danh sách hoàn chỉnh

- Cuối cùng là chuyển đổi dữ liệu test thành dạng phù hợp để test mô hình.

4. Đánh giá kết quả mô hình

4.1 Các kĩ thuật đánh giá mô hình

Machine Learning Train/Test là một phương pháp dùng để ước tính hiệu suất của các mô hình học máy, nó được gọi là Train/Test là vì nó được chia thành 2 tập dữ liệu: tập dữ liệu để huấn luyện (Train) và tập dữ liệu để kiểm thử (Test).

Ở đây nhóm chia theo tỷ lệ 70/30, tức là tập train có 70% dữ liệu trong tập dữ liệu gốc để mô hình học và tạo ra các quy tắc tổng quát, tập test có 30% dữ liệu còn lại để sử dụng như một tập kiểm tra độc lập để đánh giá hiệu suất của mô hình. Tỷ lệ trên giúp nhóm tránh được các trường hợp như mô hình chưa khớp khi sử dụng quá ít dữ liệu để huấn luyện (ví dụ như tỷ lệ 5/5) hay mô hình quá khớp khi sử dụng quá nhiều dữ liệu để huấn luyện (ví dụ như tỷ lệ 9/1). (Galarnyk, 2022)

Sau khi huấn luyện tập dữ liệu theo tỷ lệ 70/30 thì mô hình được huấn luyện lại trên toàn bộ tập dữ liệu, bao gồm cả tập train và tập test. Điều này khiến mô hình học được các đặc điểm của cả hai tập dữ liệu. Sau khi huấn luyện lần 2, mô hình sẽ được sử dụng để dự đoán các nhãn của các mẫu dữ liệu trong tập train, các mẫu dữ liệu được dự đoán sai sẽ được gán nhãn lại dựa trên kết quả dự đoán của mô hình. Quá trình này được lặp đi lặp lại nhiều lần cho đến khi độ chính xác của mô hình trên tập train đạt được một mức nhất định. (Galarnyk, 2022)

Cách huấn luyện này được tiến hành dựa trên hiện tượng Overfitting. Overfitting là hiện tượng khi mô hình học máy xây dựng thể hiện được quá chi tiết bộ dữ liệu huấn luyện, tức là cả dữ liệu nhiễu hay dữ liệu bất thường đều được chọn và học để đưa ra quy luật cho mô hình học máy. Khi áp dụng những quy luật này vào bộ dữ liệu mới có thể ảnh hưởng đến độ chính xác của mô hình. ("Vấn đề Overfitting & Underfitting trong Machine Learning,", 2019)

Trong trường hợp này, nếu mô hình được huấn luyện với tỉ lệ 70/30 đạt độ chính xác cao tức là tập dữ liệu được gán nhãn tốt và mô hình không bị overfitting. Tuy nhiên, nếu mô hình được huấn luyện lại trên toàn bộ tập dữ liệu chỉ đạt độ chính xác bằng hoặc cao hơn một chút thì có thể có nhiều mẫu được gán nhãn sai.

Nhóm đã lặp lại quá trình huấn luyện nhiều lần để gán nhãn lại cho các mẫu có nhãn sai lệch, từ đó giúp mô hình học được các đặc điểm chính xác, cải thiện độ chính xác của mô hình trên tập dữ liệu mới.

K-fold cross-validation là một phương pháp đánh giá các mô hình dự đoán trong đó các tập dữ liệu được chia thành k tập con. Quá trình học của máy có k lần và trong mỗi lần, một tập con được sử dụng để kiểm tra và (k - 1) tập con còn lại được sử dụng để huấn luyện. Phương pháp này hỗ trợ cho việc đánh giá, lựa chọn và điều chỉnh các tham số, cung cấp thước đo đáng tin cậy hơn về mức độ hiệu quả của mô hình. Nhóm áp dụng phương pháp cross-validation với k = 5 để đánh giá lại các mô hình đã chọn. (Pandian, 2023)

Train	Validation	Kết quả
Fold 1 Fold 2 Fold 3 Fold 4	Fold 5	a1
Fold 1 Fold 2 Fold 3 Fold 5	Fold 4	a2
Fold 1 Fold 2 Fold 5 Fold 4	Fold 3	a3
Fold 1 Fold 5 Fold 3 Fold 4	Fold 2	a4
Fold 5 Fold 2 Fold 3 Fold 4	Fold 1	a5

Các chỉ số mà nhóm dùng để đánh giá mô hình gồm có:

- *Precision:* Là tỉ lệ số điểm đúng tích cực trong số những điểm được phân loại là tích cực (bao gồm đúng tích cực và đúng tiêu cực), precision cao tức là độ chính xác của các điểm tìm được là cao.
- *Recall:* Là tỉ lệ số điểm đúng tích cực trong số những điểm thực sự là tích cực (bao gồm đúng tích cực và sai tiêu cực), recall cao tức là việc tỉ lệ đúng tích cực cao và tỉ lệ bỏ sót các điểm thực sự tích cực thấp.
- *F1-score:* Là trung bình có trọng số của precision và recall, có giá trị nằm trong nửa khoảng (0,1], F1-score càng cao thì bộ phân lớp càng tốt.
- *Support:* Là số lượng điểm dữ liệu thuộc lớp đó.
- *Accuracy:* Là tỉ lệ số điểm được phân loại đúng trên tổng số điểm.
- *Macro avg:* Là trung bình cộng của các precision và recall của tất cả các lớp.
- *Weight avg:* Là trung bình cộng có trọng số của các precision và recall của tất cả các lớp trong đó trọng số được tính bằng tỷ lệ phần trăm số lượng điểm dữ liệu trong lớp đó.

4.2 Đánh giá kết quả của các mô hình

Sau khi áp dụng các kỹ thuật để đánh giá 3 mô hình DecisionTreeClassifier và MultinomialNB và SVM trên tập dữ liệu, nhóm đã thu được kết quả như sau:

Bảng 1: Kết quả huấn luyện DecisionTreeClassifier

```
DATASET LEN 32146
TRAIN 70/30

              precision    recall  f1-score   support

           1      0.854     0.855     0.855      4438
           0      0.876     0.875     0.876      5206

    accuracy                          0.866      9644
   macro avg      0.865     0.865     0.865      9644
weighted avg      0.866     0.866     0.866      9644

TRAIN OVERFITING

              precision    recall  f1-score   support

           1      1.000     1.000     1.000     14766
           0      1.000     1.000     1.000     17380

    accuracy                          1.000     32146
   macro avg      1.000     1.000     1.000     32146
weighted avg      1.000     1.000     1.000     32146

CROSSVALIDATION 5 FOLDS: 0.8689 (+/- 0.0082)
```

Ở bảng 1, các chỉ số khi huấn luyện với tỷ lệ 70/30 nhìn chung khá ổn định với mức tỷ lệ từ 85,4 - 87,6%. Có thể cho rằng khi huấn luyện với tỷ lệ 70/30 thì mô hình DecisionTreeClassifier có độ chính xác tổng thể tốt. Tuy nhiên ở lần huấn luyện thứ 2, tất cả các chỉ số đều "bất thường", tỷ lệ lên đến 100%. Độ chính xác của mô hình ở lần huấn luyện với tỷ lệ 70/30 gần tương đương với độ chính xác khi sử dụng kỹ thuật cross-validation 5 folds là 86,89%, trong khi đó độ chính xác ở lần huấn luyện thứ 2 lại quá cao, chênh lệch tới 13,11% so với độ chính xác khi sử dụng kỹ thuật cross-validation 5 folds cũng như lần huấn luyện với tỷ lệ 70/30 mặc dù có độ lệch chuẩn thấp nhất là 0,0082, điều này cho thấy mô hình không thể dự đoán chính xác được các điểm dữ liệu mới, có thể kết luận rằng mô hình DecisionTreeClassifier đã gặp hiện tượng overfitting.

Bảng 2: Kết quả huấn luyện MultinomialNB:

```
DATASET LEN 32146
TRAIN 70/30

              precision    recall  f1-score   support

           1      0.841     0.952     0.893      4438
           0      0.954     0.846     0.897      5206

    accuracy                          0.895      9644
   macro avg      0.897     0.899     0.895      9644
weighted avg      0.902     0.895     0.895      9644

TRAIN OVERFITING

              precision    recall  f1-score   support

           1      0.856     0.960     0.905     14766
           0      0.962     0.863     0.910     17380

    accuracy                          0.907     32146
   macro avg      0.909     0.911     0.907     32146
weighted avg      0.913     0.907     0.908     32146

CROSSVALIDATION 5 FOLDS: 0.8970 (+/- 0.0109)
```

Ở bảng 2, nhìn chung các chỉ số ở mô hình Multinomial NB khá ổn định với khoảng tỷ lệ kéo dài từ 84,1 - 95,2% ở lần huấn luyện với tỷ lệ 70/30 và có cải thiện được độ chính xác ở lần huấn luyện thứ 2 với tỷ lệ từ 85,6 - 96%. Bên cạnh đó, mô hình Multinomial NB có độ chính xác là 89,70%, gần 90% khi sử dụng kỹ thuật cross-validation 5 folds, gần tương đương với trung bình độ chính xác ở 2 lần huấn luyện, với độ lệch chuẩn cao hơn mô hình DecisionTreeClassifier một chút là 0,0109. Có thể kết luận rằng, mô hình Multinomial NB hoạt động khá ổn định, tốt hơn so với mô hình DecisionTreeClassifier.

Bảng 3: Kết quả huấn luyện SVM:

	precision	recall	f1-score	support
1	0.918	0.939	0.928	4438
0	0.947	0.929	0.938	5206
accuracy			0.933	9644
macro avg	0.932	0.934	0.933	9644
weighted avg	0.934	0.933	0.933	9644

TRAIN OVERFITING

	precision	recall	f1-score	support
1	0.973	0.979	0.976	14766
0	0.982	0.977	0.980	17380
accuracy			0.978	32146
macro avg	0.978	0.978	0.978	32146
weighted avg	0.978	0.978	0.978	32146

CROSSVALIDATION 5 FOLDS: 0.9318 (+/- 0.0101)

Ở bảng 3, có thể thấy rằng các chỉ số của mô hình SVM cao nhất trong 3 mô hình, tất cả các chỉ số đều trên 90% với khoảng tỷ lệ từ 91,8 - 94,7% ở lần huấn luyện với tỷ lệ 70/30 và độ chính xác cũng được cải thiện đáng kể ở lần huấn luyện thứ 2 với khoảng tỷ lệ từ 97,3 - 98,2%. Ngoài ra, mô hình SVM có độ chính xác khi sử dụng kỹ thuật cross-validation 5 folds là 93,18%, với độ lệch chuẩn 0,0101 nằm giữa mô hình DecisionTreeClassifier và Multinomial NB. Có thể nói rằng mô hình SVM là mô hình ổn định nhất trong số 3 mô hình mà nhóm dùng để huấn luyện.

Sau khi so sánh kết quả ở 3 bảng, có thể thấy được mô hình SVM là mô hình có độ chính xác cao nhất, tiếp theo là mô hình Multinomial NB và cuối cùng là mô hình DecisionTreeClassifier, vì vậy nhóm quyết định sử dụng mô hình SVM để tiến hành phân loại sắc thái bình luận.

5. Kết luận

Nghiên cứu về độ hài lòng đã xem xét việc dự đoán mức độ yêu thích của khách hàng trên một nền tảng trực tuyến dựa trên tương tác của họ với nền tảng này. Để làm điều này, nhóm đã sử dụng một bộ dữ liệu duy nhất bao gồm dữ liệu số hiệu khách hàng cùng với dữ liệu liên quan đến sự tương tác của khách hàng với nền tảng thương mại điện tử trực tuyến, cụ thể là các bình luận, góp ý trên các nền tảng.

Bằng cách thực hiện cả ba mô hình học máy DecisionTreeClassifier(), MultinomialNB, LinearSVC để phân loại mức độ tích cực và tiêu cực giữa các bình luận. Cuối cùng đánh giá mức độ tin cậy của từng thuật toán dựa trên logic overfitting và kỹ thuật validate cross-5 fold để đánh giá các mô hình.

Mô hình được tối ưu hóa về thời gian cũng như dung lượng cần xử lý nhưng vẫn đảm bảo độ chính xác, bởi việc thực hiện chuẩn hóa dữ liệu được thực hiện trước đó qua khá nhiều bước làm đầu vào cho việc thực hiện model rất sạch sẽ. Đồng thời cũng áp dụng việc xây dựng ma trận số từ CountVectorizer và ma trận từ ngữ TF-IDF giúp việc thực hiện các thuật toán học máy được thực hiện nhanh và logic hơn.

Cuối cùng lựa chọn mô hình LinearSVC với độ chính xác cao nhất. Kết quả của mô hình xuất ra là các nhãn tích cực hay tiêu cực đối với mỗi bình luận, chính vì vậy có thể sử dụng kết quả cho các bài toán khác hay mở rộng mức phân tích của bài toán. Ví dụ như khi phân loại được các khách hàng với mức hài lòng khi lần đầu mua hàng, kết quả có thể tiếp tục được phân tích để dự đoán xác suất khách hàng sẽ mua hàng vào lần tiếp theo.

Mặc dù khá hoàn thiện về độ chính xác tuy nhiên mô hình cũng tồn tại những hạn chế nhất định. Điều đó đến từ hai nguyên nhân là từ việc xử lý dữ liệu do các từ ngữ không chuẩn hóa được tái chuẩn hóa dựa trên một tập các dữ liệu chuẩn nên sẽ có sự hạn chế về khả năng chuẩn hóa nếu các từ không nằm trong tập cho trước. Đông thời việc lựa chọn mô hình LinearSVC cũng có những mặt bất cập nhất là khi thực hiện phân tích các dữ liệu lớn, chuyên nghiệp. LinearSVC không phù hợp với dữ liệu lớn với số lượng mẫu quá lớn và có thể gặp vấn đề về thời gian huấn luyện. Thứ hai, mô hình này chỉ tạo ra ranh giới phẳng tuyến tính, không thể xử lý dữ liệu phi tuyến. Ngoài ra, LinearSVC cần tiền xử lý cẩn thận để đạt hiệu suất tốt.

Danh mục tài liệu tham khảo

[1] *"Vấn đề Overfitting & Underfitting trong Machine Learning,"*. (2019, 4 2). Retrieved from Trí tuệ nhân tạo: https://trituenhantao.io/kien-thuc/van-de-overfitting-underfitting-trong-machine-learning

[2] Abbas, M., & al, e. (2019). Multinomial Naive Bayes Classification Model for Sentiment. *IJCSNS International Journal of Computer Science and Network Security*.

[3] Galarnyk, M. (2022, 2 28). *Understanding Train Test Split*. Retrieved from builtin.com: https://builtin.com/data-science/train-test-split

[4] Hồ, T. T., & al, e. (2023). Analysis of online user sentiment and behavior in the tourism sector in Vietnam based on reviews and comments. *VNUHCM Journal of Economics, Business and Law*.

[5] Pandian, S. (2023, 11 17). *K-Fold Cross Validation Technique and its Essentials*. Retrieved from analyticsvidhya.com: https://www.analyticsvidhya.com/blog/2022/02/k-fold-cross-validation-technique-and-its-essentials/#:~:text=K%2Dfold%20cross%2Dvalidation%20is,estimate%20the%20model%27s%20generalization%20performance

[6] Staff, I. (2023). *What is a Decision Tree?* Retrieved from IBM.com: https://www.ibm.com/topics/decision-trees

[7] Stecanella, B. (2017, 6 22). *Monkeylearn.com*. Retrieved from Support Vector Machines (SVM) Algorithm Explained: https://monkeylearn.com/blog/introduction-to-support-vector-machines-svm/