

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

BÁO CÁO TỔNG KẾT
ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN

**ỨNG DỤNG HỌC MÁY TRONG VIỆC PHÁT HIỆN TIN GIẢ TRÊN MẠNG XÃ
HỘI**

Sinh viên thực hiện : Hoàng Kim Quang

Lớp, khoa: Khoa Học Máy Tính 1 – Công Nghệ Thông tin

Sinh viên thực hiện : Trần Văn Sơn

Lớp, khoa: Khoa Học Máy Tính 1 – Công Nghệ Thông tin

Người hướng dẫn: ThS. Lê Thị Thủy

Hà Nội - Tháng 5/ Năm 2024

BỘ CÔNG THƯỜNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

BÁO CÁO TỔNG KẾT
ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN

ỨNG DỤNG HỌC MÁY TRONG VIỆC PHÁT HIỆN TIN GIẢ TRÊN MẠNG XÃ HỘI

Sinh viên thực hiện: Hoàng Kim Quang **Nam, Nữ: Nam**

Dân tộc: Kinh

Lớp, khoa: Khoa Học Máy Tính 1 – Công Nghệ Thông Tin **Năm thứ: 3/4**

Ngành: Khoa Học Máy Tính

Sinh viên thực hiện: Trần Văn Sơn **Nam, Nữ: Nam**

Dân tộc: Kinh

Lớp, khoa: Khoa Học Máy Tính 1 – Công Nghệ Thông Tin **Năm thứ: 3/4**

Ngành: Khoa Học Máy Tính

Người hướng dẫn: ThS. Lê Thị Thủy

Hà Nội - Tháng 5/ Năm 2024

MỤC LỤC

DANH MỤC TỪ VIẾT TẮT	1
DANH MỤC BẢNG BIỂU	2
DANH MỤC HÌNH ẢNH	3
LỜI CẢM ƠN	5
MỞ ĐẦU	6
1. Lý do chọn đề tài.....	6
2. Mục đích, mục tiêu nghiên cứu	6
3. Phương pháp nguyên cứu	6
4. Đối tượng nguyên cứu	7
CHƯƠNG 1: TỔNG QUAN VỀ HỌC MÁY.....	8
1.1 Giới thiệu	8
1.2 Định nghĩa	8
1.3 Phân loại	8
1.3.1 Học có giám sát – Phân lớp	8
1.3.2 Học không giám sát – Gom cụm	10
CHƯƠNG 2: CÁC KỸ THUẬT PHÂN LỚP	11
2.1 K-nearest neighbor.....	11
2.1.1 Quy trình xây dựng.....	11
2.1.2 Khoảng cách trong không gian vector.....	11
2.1.3 Ưu nhược điểm thuật toán.....	13
2.1.4 Ví dụ.....	13
2.2 Naïve bayes	17
2.2.1 Định lý	17
2.2.2 Công thức Bayes tổng quát.....	17

2.2.3	Quy trình xây dựng	19
2.2.4	Ưu nhược điểm thuật toán.....	20
2.2.5	Ví dụ.....	21
2.3	SVM.....	24
2.3.1	Quy trình làm việc.....	24
2.3.2	Margin trong SVM.....	28
2.3.3	Lập trình tìm nghiệm cho bài toán SVM	29
2.3.4	Ưu nhược điểm thuật toán.....	29
2.4	Nơron	30
2.4.1	Kiến trúc mạng nơron.....	30
2.4.2	Một mạng NN sẽ có 3 tầng.....	31
2.4.3	Lan truyền tiến	31
2.4.4	Quy trình thuật toán	31
2.4.5	Xây dựng mô hình mạng neuron	32
CHƯƠNG 3: XÂY DỰNG ỨNG DỤNG.....		33
3.1	Tổng quan về tin giả và phát hiện tin giả.....	33
3.1.1	Tin giả	33
3.1.2	Phát hiện tin giả	35
3.2	Công cụ thực hiện.....	37
3.3	Xây dựng ứng dụng.....	38
3.3.1	Mô tả dữ liệu	38
3.3.2	Thuật toán cài đặt	41
3.3.3	Chương trình ứng dụng.....	44
KẾT LUẬN		50
TÀI LIỆU THAM KHẢO.....		51

DANH MỤC TỪ VIẾT TẮT

CNN	Convolutional Neural Network
KNN	K-Nearest Neighbors
NLP	Natural Language Processing
SVM	Recurrent Neural Network
TF-IDF	Term Frequency-Inverse Document Frequency

DANH MỤC BẢNG BIỂU

Bảng 1: Bảng dữ liệu khoản vay và khả năng vỡ nợ.....	14
Bảng 2: Bảng dữ liệu cơ sở khách hàng	22
Bảng 3: Bảng so sánh đánh giá các mô hình học máy	49

DANH MỤC HÌNH ẢNH

Hình 2.1: Biểu đồ biểu diễn dữ liệu khoản vay và khả năng vỡ nợ	16
Hình 2.2: Biểu đồ thuật toán Naive Bayes	23
Hình 2.3: Biểu đồ thuật toán SVM trường hợp 1	25
Hình 2.4: Biểu đồ thuật toán SVM trường hợp 2	25
Hình 2.5: Biểu đồ thuật toán SVM trường hợp 3	26
Hình 2.6: Biểu đồ thuật toán SVM trường hợp 4 trước	26
Hình 2.7: Biểu đồ thuật toán SVM trường hợp 4 sau.....	27
Hình 2.8: Biểu đồ biểu diễn thuật toán SVM trường hợp 5 trước.....	27
Hình 2.9: Biểu đồ biểu diễn thuật toán SVM trường hợp 5	28
Hình 2.10: Mô hình Perceptron đa tầng	30
Hình 3.1: Biểu đồ phân loại tin giả.....	33
Hình 3.2: Mô hình phân loại tin giả	36
Hình 3.3 : Các thư viện cần khai báo	38
Hình 3.4: File dữ liệu.....	39
Hình 3.5: Dữ liệu huấn luyện	40
Hình 3.6: Dữ liệu thử nghiệm.....	41
Hình 3.7: Cách thức hoạt động của mô hình.....	42
Hình 3.8: Code nhập thư viện.....	44
Hình 3.9: Code nhập và lấy dạng của dữ liệu.....	45
Hình 3.10: Code thêm thông tin	45
Hình 3.11: Code xác định labels.....	45
Hình 3.12: Code tách nhãn ra khung dữ liệu.....	46
Hình 3.13: Code chia tệp dữ liệu.....	46
Hình 3.14: Code chuyển văn bản thành các đặc trưng.....	46
Hình 3.15: Code biến đổi tệp huấn luyện và tệp thử nghiệm	46
Hình 3.16: Code sử dụng MultinomialNB	47
Hình 3.17: Code tính toán độ tin cậy.....	47
Hình 3.18: Kết quả đánh giá.....	48
Hình 3.19: Code thực nghiệm trên một dữ liệu thử nghiệm	48

Hình 3.20: Kết quả trả về	49
Hình 3.21: Biểu đồ so sánh các thuật toán	49

LỜI CẢM ƠN

Lời đầu tiên cho phép chúng em gửi lời cảm ơn sâu sắc tới các thầy cô trong khoa Công nghệ thông tin - Trường Đại học Công Nghiệp Hà Nội. Những người đã chỉ dẫn và tạo điều kiện cho em trong suốt những kỳ học vừa qua. Để hoàn thành được đề tài này, em xin được bày tỏ sự tri ân và xin chân thành cảm ơn giảng viên ThS. Lê Thị Thủy, người đã hết lòng giúp đỡ, hướng dẫn, chỉ dạy tận tình để nhóm em hoàn thành được đề tài này.

Sau nữa, chúng mình xin gửi lời cảm ơn tới các thành viên trong nhóm đã luôn bên cạnh hợp tác, góp ý, động viên, giúp đỡ cả về vật chất lẫn tinh thần trong quá trình làm đề tài này.

Trong quá trình nghiên cứu và làm đề tài, do năng lực, kiến thức, trình độ của chúng em còn hạn chế nên không tránh khỏi những thiếu sót, chúng em mong nhận được sự thông cảm và những góp ý từ quý thầy cô.

Chúng em xin chân thành cảm ơn!

Hà Nội, tháng 5, năm 2024

MỞ ĐẦU

1. Lý do chọn đề tài

Trong thời đại của thông tin kỹ thuật số và mạng xã hội, việc phát hiện và ngăn chặn sự lan truyền của tin giả trên mạng xã hội đang trở thành một thách thức đáng kể. Tin giả, hay còn gọi là tin tức giả mạo, có thể gây hại nghiêm trọng đến sự tin tưởng của người dùng và ảnh hưởng xấu đến xã hội. Điều đó việc nghiên cứu ứng dụng học máy và xử lý ngôn ngữ tự nhiên đã là một bước tiến quan nhằm khắc phục và loại bỏ những tin giả độc hại trên mạng xã hội.

Nghiên cứu đề tài này có thể được áp dụng vào thực tế, giúp cung cấp công cụ hữu ích cho các tổ chức, chính phủ và cộng đồng mạng xã hội để phát hiện và ngăn chặn tin giả. Điều này có thể góp phần tạo ra môi trường truyền thông trung thực và đáng tin cậy hơn.

2. Mục đích, mục tiêu nghiên cứu

Nguyên cứu về các phương pháp và kỹ thuật Học máy trong phân loại và dự đoán: Nghiên cứu các phương pháp và kỹ thuật Học máy như học có giám sát, học không giám sát và học tăng cường để áp dụng vào việc phát hiện tin giả trên mạng xã hội. Tìm hiểu về các mô hình và thuật toán phổ biến trong Học máy, như mạng neural, học sâu và học tăng cường.

Xây dựng mô hình Học máy để phát hiện tin giả: Tiến hành thu thập dữ liệu từ mạng xã hội, xử lý và tiền xử lý dữ liệu ngôn ngữ tự nhiên. Xây dựng một mô hình Học máy phù hợp để phân loại và dự đoán tính chân thực của các bài viết, tin tức hoặc thông tin trên mạng xã hội. Đánh giá và cải thiện hiệu suất của mô hình thông qua các phương pháp đánh giá và tinh chỉnh tham số.

Áp dụng mô hình vào thực tế và kiểm tra hiệu quả: Áp dụng mô hình Học máy đã xây dựng vào việc phát hiện tin giả trên dữ liệu thực tế từ mạng xã hội. Đánh giá hiệu quả và độ chính xác của mô hình trong việc phát hiện và ngăn chặn tin giả. So sánh với các phương pháp và công cụ hiện có để đánh giá khả năng ứng dụng của mô hình.

3. Phương pháp nguyên cứu

Thu thập và tiền xử lý dữ liệu: Thu thập dữ liệu từ các nguồn mạng xã hội phổ biến, bao gồm các bài viết.

Xây dựng tập dữ liệu huấn luyện: Xác định các bài viết, thông tin chân thực và tin giả từ tập dữ liệu thu thập được. Gán nhãn cho các mẫu dữ liệu, đánh dấu chúng là chân thực hoặc tin giả.

Lựa chọn và xây dựng mô hình Học máy: Chọn các mô hình Học máy phù hợp như mạng neural, học sâu, hoặc các thuật toán phân loại như KNN, Naive Bayes, SVM. Xây dựng mô hình dựa trên tập dữ liệu huấn luyện và tinh chỉnh các tham số của mô hình.

Đánh giá hiệu suất mô hình: Sử dụng các phương pháp đánh giá như ma trận lỗi, độ chính xác, độ phủ, độ F1 để đánh giá hiệu suất của mô hình trong việc phát hiện tin giả. So sánh kết quả với các phương pháp và công cụ khác đã được sử dụng trong lĩnh vực này.

4. Đối tượng nghiên cứu

Tin giả và thông tin chân thực: Mục tiêu là nghiên cứu các đặc điểm và mẫu tin giả để phân biệt chúng với thông tin chân thực trên mạng xã hội. Đối tượng nghiên cứu cũng bao gồm các yếu tố gây ra tin giả, quy trình lan truyền tin giả và tác động của tin giả đến công chúng và xã hội.

Phương pháp và kỹ thuật Học máy: Đối tượng nghiên cứu cũng bao gồm các phương pháp và kỹ thuật Học máy như học có giám sát, học không giám sát và học tăng cường. Nghiên cứu sẽ tìm hiểu và áp dụng các mô hình và thuật toán Học máy phù hợp để phát hiện tin giả trên mạng xã hội.

CHƯƠNG 1: TỔNG QUAN VỀ HỌC MÁY

1.1 Giới thiệu

Những năm gần đây, AI - Artificial Intelligence (Trí Tuệ Nhân Tạo), và cụ thể hơn là Machine Learning (Học Máy hoặc Máy Học) nổi lên như một bằng chứng của cuộc cách mạng công nghiệp lần thứ IV (I - động cơ hơi nước, II - năng lượng điện, III - công nghệ thông tin). Trí Tuệ Nhân Tạo đang len lỏi vào mọi lĩnh vực trong đời sống mà có thể chúng ta không nhận ra. Xe tự hành của Google và Tesla, hệ thống tự tag khuôn mặt trong ảnh của Facebook, trợ lý ảo Siri của Apple, hệ thống gợi ý sản phẩm của Amazon, hệ thống gợi ý phim của Netflix, máy chơi cờ vây AlphaGo của Google DeepMind,Không chỉ dừng lại ở đó, hiện nay AI đã có thể sáng tác các tác phẩm nghệ thuật, vẽ tranh và tương tác 1 cách chân thật với người dùng. Những ví dụ trên chỉ là một vài trong vô vàn những ứng dụng của AI/Machine Learning trong thực tế xã hội hiện đại.

1.2 Định nghĩa

Trí tuệ nhân tạo là trí thông minh được thể hiện bằng máy móc, trái ngược với trí thông minh tự nhiên của con người. Trí tuệ nhân tạo có khả năng bắt chước các chức năng "nhận thức" mà con người thường phải liên kết với tâm trí, như "học tập" và "giải quyết vấn đề". Trí tuệ nhân tạo có nhiều ứng dụng trong các lĩnh vực khác nhau, như xử lý ngôn ngữ tự nhiên, thị giác máy tính, trò chơi điện tử và xe tự lái. [18]

Machine Learning là một tập con của AI. Machine Learning là một lĩnh vực nhỏ của Khoa Học Máy Tính, nó có khả năng tự học hỏi dựa trên dữ liệu đưa vào mà không cần phải được lập trình cụ thể. Những năm gần đây, khi mà khả năng tính toán của các máy tính được nâng lên một tầm cao mới và lượng dữ liệu khổng lồ được thu thập bởi các hãng công nghệ lớn, Machine Learning đã tiến thêm một bước dài và một lĩnh vực mới được ra đời gọi là Deep Learning (Học Sâu). Deep Learning đã giúp máy tính thực thi những việc tưởng chừng như không thể vào 10 năm trước: phân loại cả ngàn vật thể khác nhau trong các bức ảnh, tự tạo chú thích cho ảnh, bắt chước giọng nói và chữ viết của con người, giao tiếp với con người, hay thậm chí cả sáng tác văn hay âm nhạc.

1.3 Phân loại

1.3.1 Học có giám sát – Phân lớp

Học có giám sát là thuật toán dự đoán đầu ra của một dữ liệu mới dựa trên các cặp (*dữ liệu đầu vào, dữ liệu đầu ra*) đã biết từ trước. Cặp dữ liệu này còn được gọi là (*data, label*), tức (*dữ liệu, nhãn*). Học có giám sát là nhóm thuật toán phổ biến nhất trong các thuật toán học máy hiện nay.

Mục đích của thuật toán là xây dựng ra hàm ánh xạ một cách tốt nhất có thể để khi bạn có dữ liệu đầu vào mới và bạn có thể dự đoán các biến đầu ra cho dữ liệu đó một cách chính xác.

Quá trình đó được gọi là việc học có giám sát bởi vì toàn bộ quá trình vận hành của thuật toán học từ tập dữ liệu đầu vào được cung cấp từ trước, việc này có thể được coi là một “giáo viên” giám sát quá trình học tập. Chúng ta biết trước được kết quả câu trả lời đúng, cung cấp cho thuật toán và sau đó, thuật toán sẽ thực thi lặp đi lặp lại làm cho việc dự đoán cho dữ liệu đầu vào liên tục được “giáo viên” hoàn thiện và kết quả của sự dự đoán sẽ ngày càng được cải thiện và chính xác. Việc học dừng lại khi thuật toán đạt được mức hiệu suất ở mức chấp nhận được.

Thuật toán học có giám sát còn được tiếp tục chia nhỏ ra thành hai loại chính, đó là **phân loại** và **hồi quy**.

1.3.1.1 Phân loại

Một bài toán được gọi là classification nếu các label của input data được chia thành một số hữu hạn nhóm. Cụ thể là nhóm đầu ra đều bên trong một lớp. Nếu thuật toán cố gán nhãn đầu vào thành hai lớp riêng biệt, nó được gọi là phân loại nhị phân. Chọn giữa nhiều hơn hai lớp được gọi là phân loại đa lớp.

Ví dụ: Gmail xác định xem một email có phải là spam hay không; các hãng tín dụng xác định xem một khách hàng có khả năng thanh toán nợ hay không.

Điểm mạnh: Cây phân loại thực hiện rất tốt trong thực tế nghiên cứu

Điểm yếu: Do bản thân thuật toán không bị giới hạn, vì vậy các cây riêng lẻ dễ bị quá mức.

1.3.1.2 Hồi quy

Nếu *label* không được chia thành các nhóm mà là một giá trị thực cụ thể. Ví dụ: một căn nhà rộng x m², có y phòng ngủ và cách trung tâm thành phố z km sẽ có giá là bao nhiêu?

1.3.2 Học không giám sát – Gộp cụm

Trong thuật toán này, chúng ta không biết được outcome hay nhãn mà chỉ có dữ liệu đầu vào. Thuật toán học không giám sát sẽ dựa vào cấu trúc của dữ liệu để thực hiện một công việc nào đó, ví dụ như phân nhóm (clustering) hoặc giảm số chiều của dữ liệu (dimension reduction) để thuận tiện trong việc lưu trữ và tính toán. Một cách toán học, Học không giám sát là khi chúng ta chỉ có dữ liệu vào X mà không biết nhãn Y tương ứng.

Những thuật toán loại này được gọi là Học không giám sát vì không giống như Học giám sát, chúng ta không biết câu trả lời chính xác cho mỗi dữ liệu đầu vào. Giống như khi ta học, không có thầy cô giáo nào chỉ cho ta biết đó là chữ A hay chữ B. Cụm không giám sát được đặt tên theo nghĩa này. Các bài toán Học không giám sát được tiếp tục chia nhỏ thành hai loại:

1.3.2.1 Phân nhóm

Một bài toán phân nhóm toàn bộ dữ liệu X thành các nhóm nhỏ dựa trên sự liên quan giữa các dữ liệu trong mỗi nhóm. Ví dụ: phân nhóm khách hàng dựa trên hành vi mua hàng. Điều này cũng giống như việc ta đưa cho một đứa trẻ rất nhiều mảnh ghép với các hình thù và màu sắc khác nhau, ví dụ tam giác, vuông, tròn với màu xanh và đỏ, sau đó yêu cầu trẻ phân chúng thành từng nhóm. Mặc dù không cho trẻ biết mảnh nào tương ứng với hình nào hoặc màu nào, nhiều khả năng chúng vẫn có thể phân loại các mảnh ghép theo màu hoặc hình dạng.

1.3.2.2 Association

Là bài toán khi chúng ta muốn khám phá ra một quy luật dựa trên nhiều dữ liệu cho trước. Ví dụ: những khách hàng nữ mua quần áo thường có xu hướng mua thêm giày dép hoặc túi xách; những khán giả xem phim Spider Man thường có xu hướng xem thêm phim Bat Man, dựa vào đó tạo ra một hệ thống gợi ý khách hàng (Recommendation System), thúc đẩy nhu cầu mua sắm.

CHƯƠNG 2: CÁC KỸ THUẬT PHÂN LỚP

2.1 K-nearest neighbor

2.1.1 Quy trình xây dựng

Ta có D là tập các điểm dữ liệu đã được gán nhãn và A là dữ liệu chưa được phân loại. Ta bắt đầu thực hiện thuật toán. Ta bắt tay vào tính khoảng cách, áp dụng 1 trong các cách tính khoảng cách Euclid, Manhattan, Minkowski, Minkowski hoặc tính theo trọng số từ dữ liệu mới A đến tất cả các dữ liệu khác đã được phân loại trong D . Sau khi tính hết toàn bộ khoảng cách nêu trên, chọn K (K là tham số tự định nghĩa) khoảng cách nhỏ nhất, hay nói cách khác là lấy ra K đối tượng mà có khoảng cách nhỏ nhất đã được tính toán.

Kiểm tra danh sách các lớp có khoảng cách nhỏ nhất và đếm số lượng xuất hiện của mỗi lớp. Lấy ra đúng lớp xuất hiện nhiều nhất trong K đối tượng được lấy ra. Từ đó rút ra kết luận: Lớp của dữ liệu mới nhập vào là lớp đã nhận được ở trên.

Thế nhưng, vấn đề mang tính ảnh hưởng nhất là lấy giá trị K sao cho phù hợp, tức là cần bao nhiêu điểm ở gần hay cách nói khác là cần bao nhiêu “láng giềng” gần nhất với dữ liệu cần phân loại?

Theo các chuyên gia, chúng ta nên thực hiện các phép thử, chạy nhiều mô hình KNN với các giá trị K khác nhau và bắt đầu thực nghiệm từ 1 “láng giềng”, nghĩa là với $K = 1$, sau đó kiểm tra độ hiệu quả của từng mô hình để có thể tìm được giá trị K nào sẽ là tối ưu nhất cho bài toán của mình. Tuy nhiên, do dữ liệu mỗi tập sẽ có tính khác biệt, đặc điểm khác nhau, vì vậy, việc xác định được giá trị K nào sẽ là phù hợp sẽ gặp nhiều khó khăn. Tuy nhiên, nếu ra có thể tìm được giá trị K phù hợp cho bài toán, hiệu quả cũng như độ chính xác của bài toán sẽ được nâng lên đáng kể.

2.1.2 Khoảng cách trong không gian vector

Trong không gian một chiều, việc đo đạc khoảng cách giữa 2 điểm là một công việc hết sức quen thuộc là lấy giá trị tuyệt đối của hiệu giữa hai giá trị được xét. Ở trong không gian hai chiều, tức là một mặt phẳng, ta có thể sử dụng nhiều công thức để tính khoảng

cách khác nhau, và ta thường sử dụng công thức tính khoảng cách Euclid để thực hiện việc này.

Việc đo khoảng cách giữa hai điểm dữ liệu nhiều chiều là rất cần thiết trong học máy để phục vụ việc đánh giá xem điểm nào là điểm gần nhất so với một điểm khác trong không gian. Vì vậy mà khái niệm norm ra đời. Norm có nhiều loại:

- Chuẩn norm 1: Là tổng các giá trị tuyệt đối của các phần tử vector, công thức toán học như sau:

$$\|\vec{x}\|_1 = \sum_{i=1}^N |x_i|$$

- Chuẩn norm 2: là căn bậc hai của tổng bình phương của mỗi phần tử vector, công thức toán học như sau:

$$\|\vec{x}\|_2 = \sqrt{\sum_{i=1}^N |x_i|^2}$$

- Chuẩn giới hạn âm: là giá trị tuyệt đối nhỏ nhất trong số các phần tử của vector, công thức toán học như sau:

$$\|\vec{x}\|_{+\infty} = |x_i|$$

- Chuẩn giới hạn dương: ngược lại với chuẩn giới hạn âm là giá trị tuyệt đối lớn nhất trong số các phần tử của vector, công thức toán học như sau:

$$\|\vec{x}\|_{-\infty} = |x_i|$$

- Chuẩn norm p, có công thức như sau:

$$L_p = \|\vec{x}\|_p = \sqrt[p]{\sum_{i=1}^N |x_i|^p}$$

Để xác định khoảng cách giữa hai vector x và y , người ta thường áp dụng một hàm số lên vector hiệu: $f = x - y$. Một hàm số được dùng để đo các vector sẽ cần có một vài tính chất đặc biệt.

2.1.3 Ưu nhược điểm thuật toán

Ưu điểm của thuật toán

Thuật toán đơn giản, dễ dàng triển khai. Độ phức tạp tính toán nhỏ. Xử lý tốt với tập dữ liệu nhiều.

Nhược điểm của thuật toán

Thuật toán dễ dàng cài đặt và sử dụng, vì vậy với giá trị K nhỏ dễ gặp nhiều dẫn tới kết quả đưa ra không chính xác. Hơn nữa, vì hoàn toàn không sử dụng tới dữ liệu được cung cấp sẵn mà chỉ sử dụng khi thực thi thuật toán nên thuật toán cần nhiều thời gian để thực hiện do phải tính toán khoảng cách với tất cả các đối tượng trong tập dữ liệu. Cũng vì lý do đó, ta phải chuyển đổi kiểu dữ liệu thành các yếu tố định tính.

- Trong lĩnh vực giáo dục: Phân loại các học sinh theo hoàn cảnh, học lực, tính cách để tìm ra phương án hỗ trợ, tạo điều kiện cho các em để các em có điều kiện học tập, phát triển tốt hơn.

- Trong lĩnh vực thương mại điện tử: Phân loại khách hàng theo sở thích, theo thói quen để hỗ trợ bán hàng hay xây dựng hệ thống khuyến nghị, tư vấn cụ thể dựa trên dữ liệu thu thập được.

Trong lĩnh vực kinh tế: Giúp dự báo các sự kiện kinh tế trong tương lai, dự báo tình hình thời tiết, xác định xu hướng thị trường để phục vụ lên kế hoạch đầu tư.

2.1.4 Ví dụ

Giả sử có một ngân hàng có một tập dữ liệu gồm 10000 khách hàng đã mở các khoản vay khác nhau, sau một thời gian cụ thể, ngân hàng đã xác định được 8000 khách hàng thanh toán các khoản vay nợ đúng hạn và còn có 2000 khách hàng không thanh toán vay

nợ đúng hạn. Ngân hàng tổng hợp toàn bộ dữ liệu và phân loại ra các khách hàng này theo khả năng nợ vỡ nợ hay không vỡ nợ.

Dựa vào đó, ta có dữ liệu đầu vào:

Đặt Y là có khả năng vỡ nợ, N là không có khả năng vỡ nợ.

Ta có các biến đầu vào là:

- Age: Độ tuổi

- Loan: Khoản vay

- Default: Khả năng vỡ nợ

- Khoảng cách được thuật toán tính sẽ là Distance được tính theo công thức tính khoảng cách Euclid

Công thức khoảng cách Euclid: $\sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$

Chúng ta có dữ liệu là tuổi, khoản vay và khả năng vỡ nợ như bảng sau:

Bảng 1: Bảng dữ liệu khoản vay và khả năng vỡ nợ

Age	Loan	Default	Distance
25	\$40,000	N	102000
35	\$60,000	N	82000
45	\$80,000	N	62000
20	\$20,000	N	122000
35	\$120,000	N	2000
52	\$18,000	N	124000
23	\$95,000	Y	47000
40	\$62,000	Y	80000

60	\$100,000	Y	42000
48	\$220,000	Y	78000
33	\$150,000	Y	8000
...
48	\$142,000	?	?

Thuật toán KNN cho rằng những dữ liệu tương tự nhau sẽ tồn tại gần nhau trong một không gian, từ đó công việc của chúng ta là sẽ tìm k điểm gần với dữ liệu cần kiểm tra nhất. Việc tìm khoảng cách giữa 2 điểm cũng có nhiều công thức có thể sử dụng, tùy trường hợp mà chúng ta lựa chọn cho phù hợp.

Dữ liệu cần phân loại của chúng ta là {age: 48, loan: 142000}. Đây là dữ liệu 2 chiều và chúng ta cần dự đoán người này thuộc nguy cơ vỡ nợ hay không. Chúng ta sẽ dùng một cách khá phổ biến để tính khoảng cách là Euclidean. Ví dụ ở hàng đầu tiên khoảng cách sẽ được tính:

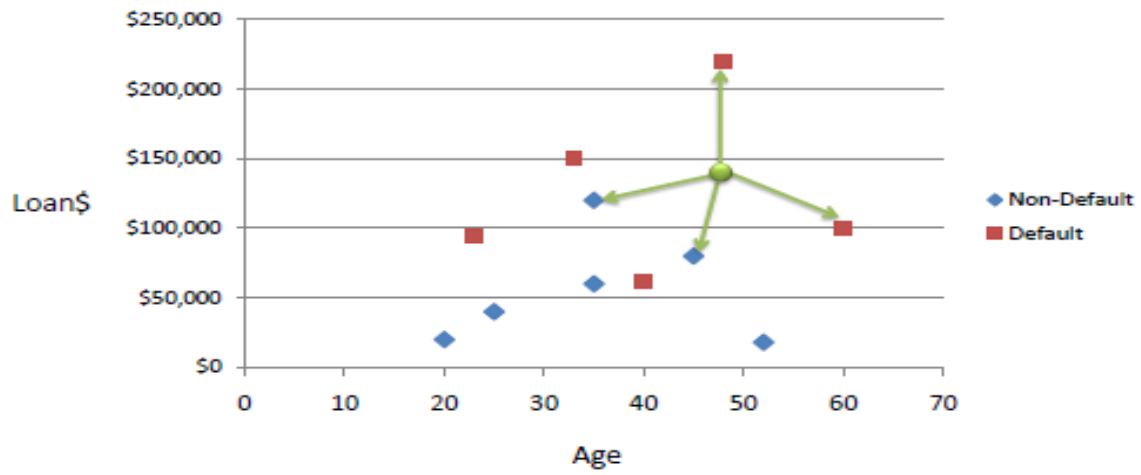
$$\sqrt{(48 - 25)^2 + (142000 - 40000)^2}$$

Thực hiện tương tự, ta sẽ tính được khoảng cách ở cột Distance, từ đó chọn ra $k = 3$ khoảng cách nhỏ nhất (gần với dữ liệu vào nhất). Với 3 khoảng cách này chúng ta nhận được 3 label là (Yes, No, Yes). Trong 3 label này Yes xuất hiện nhiều hơn nên chúng ta sẽ đưa ra dự đoán người này có khả năng vỡ nợ.

Vì đây là dữ liệu 2 chiều nên chúng ta cũng có thể biểu diễn dữ liệu trong hệ tọa độ như hình:

Trên hệ tọa độ này chúng ta dễ dàng nhận thấy cách chúng ta chọn k điểm gần nhất. Nhưng với dữ liệu lớn, nhiều chiều thì việc biểu diễn dữ liệu trên một không gian là

không hề dễ dàng.



Hình 2.1: Biểu đồ biểu diễn dữ liệu khoản vay và khả năng vỡ nợ

2.2 Naïve bayes

2.2.1 Định lý

Với $P(B) > 0$:

$$P(A|B) = \frac{P(AB)}{P(B)}$$

Suy ra:

$$P(AB) = P(A|B) P(B) = P(B|A) P(A)$$

Công thức Bayes:

$$\begin{aligned} P(A|B) &= \frac{P(AB)}{P(A)} = \frac{P(A|B)P(B)}{P(A)} = \frac{P(A|B)P(B)}{P(AB)+P(A\bar{B})} \\ &= \frac{P(A|B)P(B)}{P(AB)+P(A\bar{B})} = \frac{P(A|B)P(B)}{P(A|B)P(B)+P(A|\bar{B})P(\bar{B})} \end{aligned}$$

Với A, B là 2 biến cố

2.2.2 Công thức Bayes tổng quát

Với $P(A) > 0$ và $\{B_1, B_2, \dots, B_n\}$ là một hệ đầy đủ các biến cố:

Tổng xác suất của hệ bằng 1:

$$\sum_{k=1}^n P(B_k) = 1$$

Từng đôi một xung khắc:

$$P(B_i \cap B_j) = 0$$

Khi đó, ta có:

$$P(A) = \frac{P(A|B_k)P(B_k)}{P(A)} = \frac{P(A|B_k)P(B_k)}{\sum_{i=1}^n P(A|B_i)P(B_j)}$$

Trong đó ta gọi A là một chứng cứ (evidence) (trong bài toán phân lớp A sẽ là một phần tử dữ liệu), B là một giả thiết nào để cho A thuộc về một lớp C nào đó. Trong bài toán phân lớp chúng ta muốn xác định giá trị $P(B/A)$ là xác suất để giả thiết B là đúng với chứng

cứ A thuộc vào lớp C với điều kiện ra đã biết các thông tin mô tả A. $P(B|A)$ là một xác suất hậu nghiệm (posterior probability hay posteriori probability) của B với điều kiện A.

Giả sử tập dữ liệu khách hàng của chúng ta được mô tả bởi các thuộc tính tuổi và thu nhập, và một khách hàng X có tuổi là 25 và thu nhập là 2000\$. Giả sử H là giả thiết khách hàng đó sẽ mua máy tính, thì $P(H|X)$ phản ánh xác suất người dùng X sẽ mua máy tính với điều kiện ta biết tuổi và thu nhập của người đó.

Ngược lại $P(H)$ là xác suất tiên nghiệm (prior probability hay priori probability) của H. Nó là xác suất một khách hàng sẽ mua máy tính mà không cần biết các thông tin về tuổi hay thu nhập của họ. Hay nói cách khác, xác suất này không phụ thuộc vào yếu tố X. Tương tự, $P(X|H)$ là xác suất của X với điều kiện H (likelihood), nó là một xác suất hậu nghiệm. Ví dụ, nó là xác suất người dùng X (có tuổi là 25 và thu nhập là \$200) sẽ mua máy tính với điều kiện ta đã biết người đó sẽ mua máy tính. Cuối cùng $P(X)$ là xác suất tiên nghiệm của X. Trong ví dụ trên, nó sẽ là xác suất một người trong tập dữ liệu sẽ có tuổi 25 và thu nhập \$2000.

$$\text{Posterior} = \text{Likelihood} * \text{Prior} / \text{Evidence}$$

Trong thực tế, dữ liệu thu thập được có nhu cầu để thực hiện bài toán phân loại là dữ liệu rời rạc. Do đó, để bài toán phân loại bằng phương pháp Bayes có tính ứng dụng thực tế, công việc đầu tiên phải làm là ước lượng hàm mật độ xác suất từ dữ liệu rời rạc. Có rất nhiều phương pháp tham số cũng như phi tham số để thực hiện công việc này. Trong bài báo cáo này, tôi sử dụng phương pháp hàm hạt nhân, một phương pháp cho đến thời điểm hiện tại có nhiều ưu điểm và được áp dụng nhất. Hàm mật độ n chiều ước lượng bằng phương pháp này có dạng:

$$f(x) = \frac{1}{Nh_1 h_2 \dots h_n} \sum_{i=1}^N \prod_{j=1}^n K_j \left(\frac{x_i - x_{ij}}{h_j} \right)$$

Trong đó:

- h_j là tham số tron cho biến thứ j .
- K_j là hàm hạt nhân của biến thứ j ,
- x_i là chiều thứ i , x_{ij} là số liệu thứ i của biến thứ j
- N là số phần tử của mẫu

Mặc dù tại thời điểm hiện tại, về mặt lý thuyết, chúng ta vẫn chưa khẳng định việc chọn xác suất tiên nghiệm theo phương pháp nào là hợp lý, tuy nhiên dựa trên các ứng dụng thực tế cho ta thấy việc chọn theo dựa vào tập mẫu phối đều thường cho kết quả tốt nhất. Công thức tính:

$$q_1 = q_2 = \dots = q_c = \frac{1}{c} \cdot q_i = \frac{n_i+1}{N+n}$$

Trong đó n_i là số các phần tử trong w_i , n là số chiều và N là số những phần tử của tập mẫu.

Để tính sai số Bayes, Pham-Gia et al (2006) đã tìm các biểu thức giải tích cụ thể để xác định trong một số trường hợp đặc biệt của phân phối một chiều cho hai tổng thể. Trong trường hợp nhiều tổng thể một chiều, chúng tôi đã thiết lập chương trình xác định biểu thức giải tích cụ thể hàm cực đại, để từ đó tính tích phân chúng và xác định chính xác sai số Bayes. Khi có nhiều chiều, việc xác định hàm cực đại của các $g_i(x)$ vô cùng phức tạp, ngay cả trường hợp hai tổng thể có phân phối chuẩn (xem Pham-Gia et al., 2008). Chúng tôi sử dụng cách tính gần đúng hàm cực đại của các hàm mật độ xác suất bằng phương pháp Monte-Carlo, để từ đó tính sai số Bayes cho trường hợp k tổng thể n chiều.

2.2.3 Quy trình xây dựng

Gọi D là tập dữ liệu huấn luyện, trong đó mỗi phần tử dữ liệu X được biểu diễn bằng một vector chứa n giá trị thuộc tính $A_1, A_2, \dots, A_n = \{x_1, x_2, \dots, x_n\}$

Giả sử có m lớp C_1, C_2, \dots, C_m . Cho một phần tử dữ liệu X , bộ phân lớp sẽ gán nhãn cho X là lớp có xác suất hậu nghiệm lớn nhất. Cụ thể, bộ phân lớp Bayes sẽ dự đoán X

thuộc vào lớp C_i nếu và chỉ nếu: $P(C_i|X) > P(C_j|X)$ ($1 \leq i, j \leq m, i \neq j$) Giá trị này sẽ tính dựa trên định lý Bayes.

Để tìm xác suất lớn nhất, ta nhận thấy các giá trị $P(X)$ là giống nhau với mọi lớp nên không cần tính. Do đó ta chỉ cần tìm giá trị lớn nhất của $P(X | C_i) * P(C_i)$. Chú ý rằng $P(C_i)$ được ước lượng bằng $|D_i|/|D|$, trong đó D_i là tập các phần tử dữ liệu thuộc lớp C_i . Nếu xác suất tiên nghiệm $P(C_i)$ cũng không xác định được thì ta coi chúng bằng nhau $P(C_1) = P(C_2) = \dots = P(C_m)$, khi đó ta chỉ cần tìm giá trị $P(X|C_i)$ lớn nhất.

Khi số lượng các thuộc tính mô tả dữ liệu là lớn thì chi phí tính toán $P(X|C_i)$ là rất lớn, do đó có thể giảm độ phức tạp của thuật toán Naive Bayes giả thiết các thuộc tính độc lập nhau. Khi đó ta có thể tính: $P(X|C_i) = P(x_1|C_i) \dots P(x_n|C_i)$.

2.2.4 Ưu nhược điểm thuật toán

Ưu điểm thuật toán

Thuật toán có tính giả định độc lập, hoạt động tốt cho nhiều bài toán cũng như miền dữ liệu và ứng dụng. Hơn nữa, thuật toán tuy đơn giản nhưng lại hoàn toàn đủ tốt để giải quyết nhiều bài toán như phân lớp văn bản, lọc spam, ... thế nhưng độ phức tạp khi tính toán của quá trình training là bằng 0. Vì vậy mà việc dự đoán kết quả dữ liệu mới rất đơn

giản, kết hợp với tri thức tiên nghiệm và dữ liệu quan sát được sẽ cho ra kết quả với độ chính xác cao. Và cho kết quả tốt khi có sự chênh lệch số lượng giữa các lớp phân loại.

Huấn luyện mô hình (ước lượng tham số) dễ và nhanh

Nhược điểm thuật toán

Giả định độc lập (ưu điểm cũng chính là nhược điểm) hầu hết các trường hợp thực tế trong đó có các thuộc tính trong các đối tượng thường phụ thuộc lẫn nhau.

Vấn đề zero

Mô hình không được huấn luyện bằng phương pháp tối ưu mạnh và chặt chẽ. Tham số của mô hình là các ước lượng xác suất điều kiện đơn lẻ. Không tính đến sự tương tác giữa các ước lượng này.

2.2.5 Ví dụ

Ví dụ 1: Phân các bệnh nhân thành 2 lớp ung thư và không ung thư. Giả sử xác suất để một người bị ung thư là 0.008 tức là $P(\text{cancer}) = 0.008$; và $P(\text{nocancer}) = 0.992$. Xác suất để bệnh nhân ung thư có kết quả xét nghiệm dương tính là 0.98 và xác suất để bệnh nhân không ung thư có kết quả dương tính là 0.03 tức là $P(+/\text{cancer}) = 0.98$, $P(+/\text{nocancer}) = 0.03$. Bây giờ giả sử một bệnh nhân có kết quả xét nghiệm dương tính.

Ta có:

$$P(+/\text{cancer}) P(\text{cancer}) = 0.98 * 0.008 = 0.0078$$

$$P(+/\text{nocancer}) P(\text{nocancer}) = 0.03 * 0.992 = 0.0298$$

Như vậy, $P(+/\text{nocancer}) P(\text{nocancer}) \gg P(+/\text{cancer}) P(\text{cancer})$.

Do đó ta xét đoán rằng, bệnh nhân là không ung thư.

Ví dụ 2: Cơ sở dữ liệu khách hàng:

Bảng 2: Bảng dữ liệu cơ sở khách hàng

ID	Tuổi	Thu nhập	Sinh viên	Đánh giá tín dụng	Mua máy tính
1	Youth	High	No	Fair	No
2	Youth	High	No	Excellent	No
3	Middle	High	No	Fair	Yes
4	Senior	Medium	No	Fair	Yes
5	Senior	Low	Yes	Fair	Yes
6	Senior	Low	Yes	Excellent	No
7	Middle	Low	Yes	Excellent	Yes
8	Youth	Medium	No	Fair	Yes
9	Youth	Low	Yes	Fair	Yes
10	Senior	Medium	Yes	Fair	Yes

Giả sử có một khách hàng mới X có các thuộc tính $X = (\text{age} = \text{youth}, \text{income} = \text{medium}, \text{student} = \text{yes}, \text{credit_rating} = \text{fair})$

Bây giờ cần xác định xem khách hàng X có thuộc lớp C_{yes} (mua máy tính) hay không, ta tính toán như sau: $P(C_{\text{yes}}) = 9/14 = 0.357$

Các xác suất thành phần:

$$P(\text{age} = \text{youth} | C_{\text{yes}}) = 2/9 = 0.222$$

$$P(\text{age} = \text{youth} | C_{\text{no}}) = 3/5 = 0.6$$

$$P(\text{income} = \text{medium} | C_{\text{yes}}) = 4/9 = 0.444$$

$$P(\text{income} = \text{medium} | C_{\text{no}}) = 2/5 = 0.4$$

$$P(\text{student} = \text{yes} | C_{\text{yes}}) = 6/9 = 0.667$$

$$P(\text{student} = \text{yes} | C_{\text{no}}) = 1/5 = 0.2$$

$$P(\text{credit_rating} = \text{fair} | C_{\text{yes}}) = 6/9 = 0.667$$

$$P(\text{credit_rating} = \text{fair} | C_{\text{no}}) = 2/5 = 0.2$$

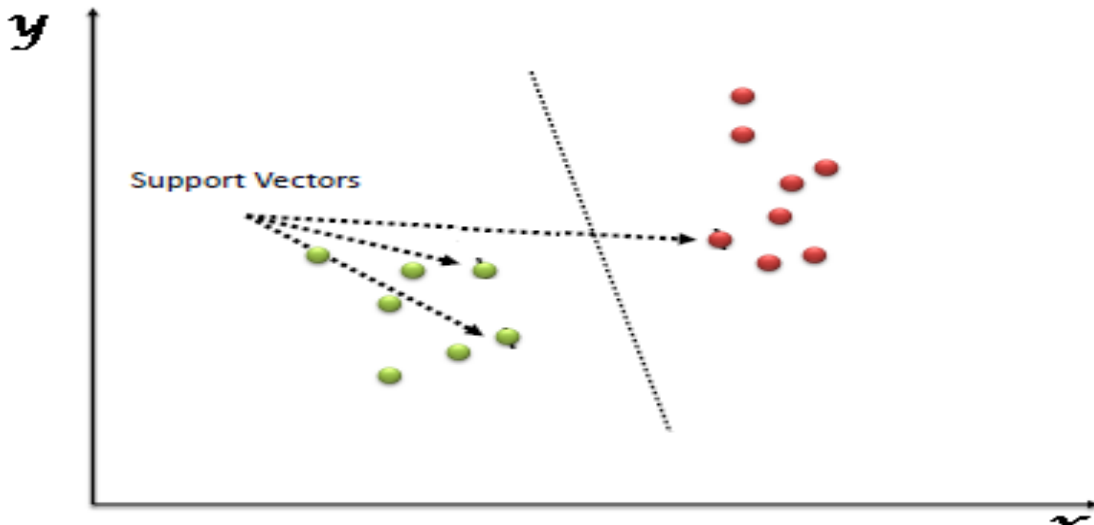
Cuối cùng:

$$P(X | C_{\text{yes}}) = 0.222 * 0.444 * 0.667 * 0.667 = 0.044$$

$$P(X | C_{\text{no}}) = 0.604 * 0.2 * 0.4 = 0.019$$

$$P(X | C_{\text{yes}}) * P(C_{\text{yes}}) = 0.044 * 0.643$$

$$P(X | C_{\text{no}}) * P(C_{\text{no}}) = 0.019 * 0.357 = 0.007$$



Hình 2.2: Biểu đồ thuật toán Naive Bayes

Từ kết quả này ta thấy $P(X | C_{\text{yes}}) P(C_{\text{yes}})$ có giá trị lớn nhất, do đó thuật toán Bayes sẽ kết luận là khách hàng X sẽ mua máy tính.

2.3 SVM

SVM là phương pháp học có giám sát do Vladimir N. Vapnik đề xuất vào năm 1995, và ngày càng được sử dụng phổ biến trong đa dạng lĩnh vực, đặc biệt là lĩnh vực phân loại mẫu và nhận dạng mẫu.

Phương pháp này thực hiện phân lớp dựa trên nguyên lý cực tiểu hóa rủi ro có cấu trúc SRM – một trong số các phương pháp phân lớp giám sát không tham số tinh vi nhất hiện tại.

Bài toán phân lớp sử dụng SVM có 2 mục đích chính. Thứ nhất, tìm một siêu phẳng có biên cực đại giữa các lớp mẫu âm và mẫu dương. Thứ hai, cực tiểu hóa các mẫu không phân chia được trong tập huấn luyện.

2.3.1 Quy trình làm việc

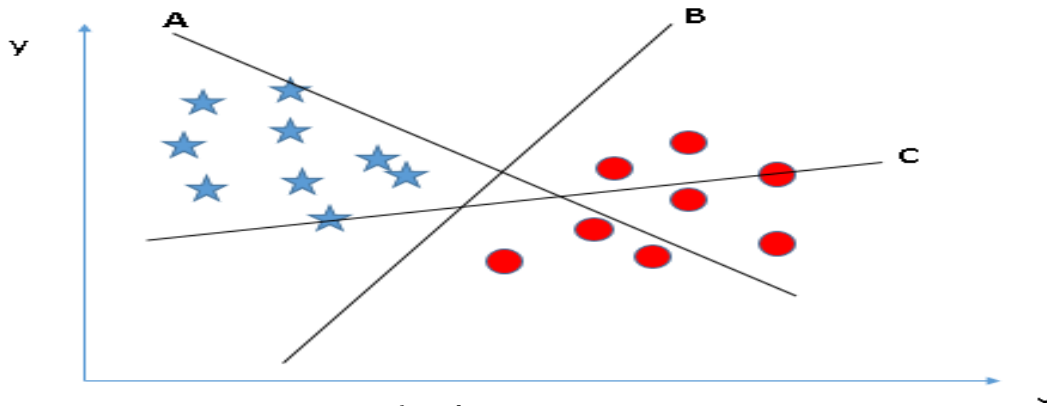
Support Vectors hiểu một cách đơn giản là các đối tượng trên đồ thị tọa độ quan sát, Support Vector Machine là một biên giới để chia hai lớp tốt nhất.

Chúng ta sẽ theo các tiêu chí sau:

Xác định siêu mặt phẳng bên phải (Tình huống 1):

Ở đây, có 3 đường hyper-lane (A, B and C). Bây giờ đường nào là hyper-lane đúng cho nhóm ngôi sao và hình tròn.

Quy tắc số một để chọn 1 hyper-lane, chọn một hyperplane để phân chia hai lớp tốt nhất. Trong ví dụ này chính là đường B.

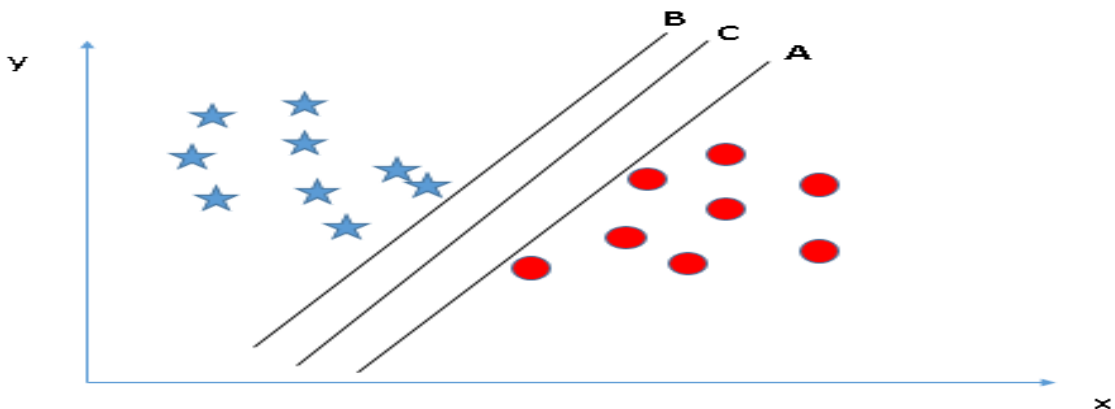


Hình 2.3: Biểu đồ thuật toán SVM trường hợp 1

Xác định siêu mặt phẳng bên phải (Tình huống 2):

Ở đây chúng ta cũng có 3 đường hyperplane (A, B và C), theo quy tắc số 1, chúng đều thỏa mãn.

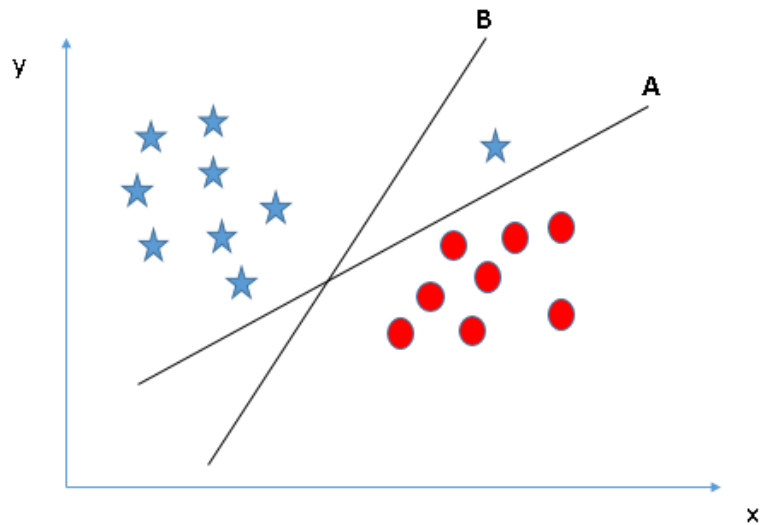
Quy tắc thứ hai chính là xác định khoảng cách lớn nhất từ điều gần nhất của một lớp nào đó đến đường hyperplane. Khoảng cách này được gọi là "Margin", Hãy nhìn hình bên dưới, trong đây có thể nhìn thấy khoảng cách margin lớn nhất đây là đường C. Cần nhớ nếu chọn làm hyper-lane có margin thấp hơn thì sau này khi dữ liệu tăng lên thì sẽ sinh ra nguy cơ cao về việc xác định nhầm lớp cho dữ liệu.



Hình 2.4: Biểu đồ thuật toán SVM trường hợp 2

Xác định siêu mặt phẳng bên phải (Tình huống 3):

Sử dụng các nguyên tắc đã nêu trên để chọn ra hyperplane cho trường hợp sau:

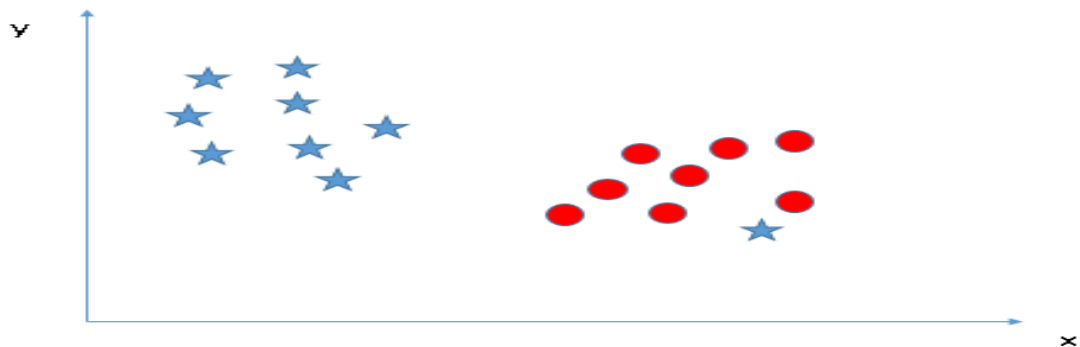


Hình 2.5: Biểu đồ thuật toán SVM trường hợp 3

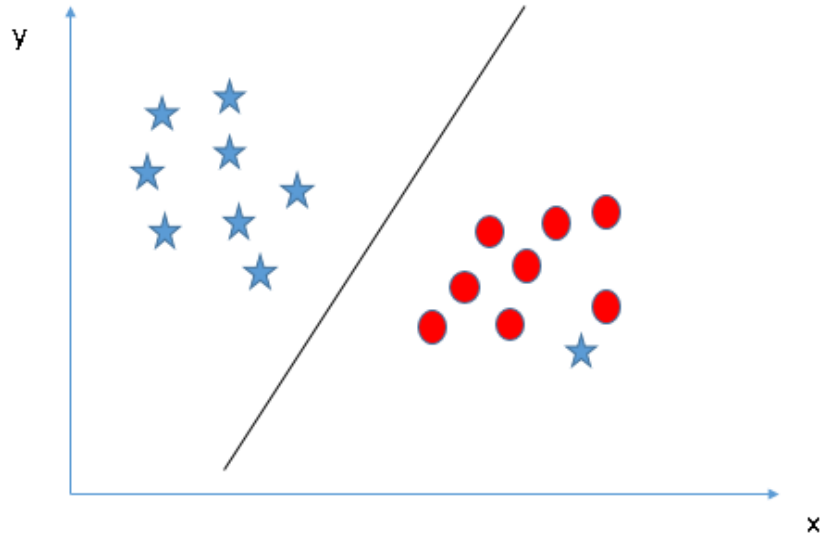
Có thể có một vài người sẽ chọn đường B bởi vì nó có margin cao hơn đường A, nhưng đây sẽ không đúng bởi vì nguyên tắc đầu tiên sẽ là nguyên tắc số 1, chúng ta cần chọn hyperplane để phân chia các lớp thành riêng biệt. Vì vậy đường A mới là lựa chọn chính xác.

Phân loại hai lớp (Tình huống 4):

Tiếp theo hãy xem hình bên dưới, không thể chia thành hai lớp riêng biệt với 1 đường thẳng, để tạo 1 phần chỉ có các ngôi sao và một vùng chỉ chứa các điểm tròn.



Hình 2.6: Biểu đồ thuật toán SVM trường hợp 4 trước

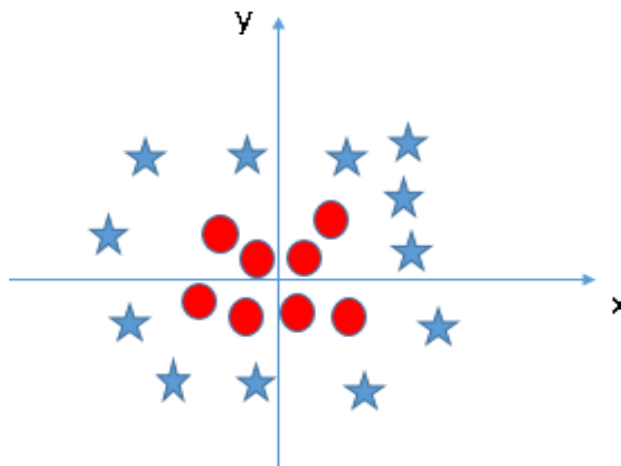


Hình 2.7: Biểu đồ thuật toán SVM trường hợp 4 sau

Ở đây sẽ chấp nhận, một ngôi sao ở bên ngoài cuối được xem như một ngôi sao phía ngoài hơn, SVM có tính năng cho phép bỏ qua các ngoại lệ và tìm ra hyperplane có biên giới tối đa. Do đó có thể nói, SVM có khả năng mạnh trong việc chấp nhận ngoại lệ.

Tìm siêu phẳng để phân tách các lớp (Tình huống 5)

Trong trường hợp dưới đây, không thể tìm ra 1 đường hyperplane tương đối để chia các lớp, vậy làm thế nào để SVM phân tách dữ liệu thành hai lớp riêng biệt? Cho đến bây giờ chúng ta chỉ nhìn vào các đường tuyến tính hyperplane.

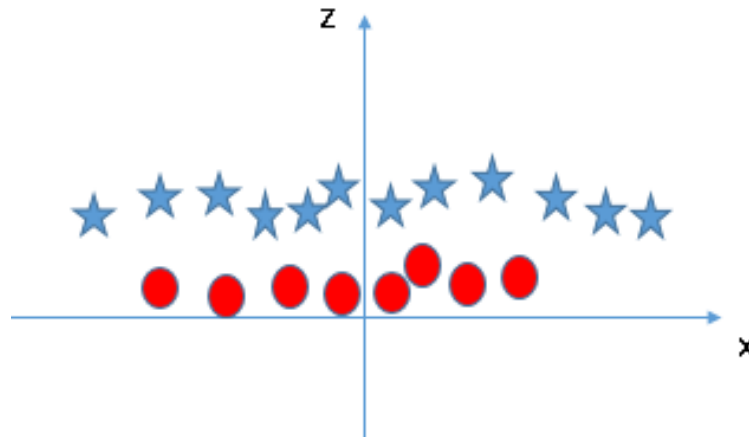


Hình 2.8: Biểu đồ biểu diễn thuật toán SVM trường hợp 5 trước

SVM có thể giải quyết vấn đề này, Khá đơn giản, nó sẽ được giải quyết bằng việc

thêm một tính năng, Ở đây chúng ta sẽ thêm tính năng $z = x^2 + y^2$. Bây giờ dữ liệu sẽ được biến đổi theo trục x và z như sau

Trong sơ đồ trên, các điểm cần xem xét là: Thứ nhất, tất cả dữ liệu trên trục z sẽ là số dương vì nó là tổng bình phương x và y .



Hình 2.9: Biểu đồ biểu diễn thuật toán SVM trường hợp 5

Trên biểu đồ các điểm tròn đỏ xuất hiện gần trục x và y hơn vì thế z sẽ nhỏ hơn \Rightarrow nằm gần trục x hơn trong đồ thị (z, x) . Trong SVM, rất dễ dàng để có một siêu phẳng tuyến tính để chia thành hai lớp. Nhưng có một câu hỏi sẽ nảy sinh đây là: Chúng ta có cần phải thêm một tính năng phân chia này bằng tay hay không?

Câu trả lời là Không, bởi vì SVM có một kỹ thuật được gọi là kỹ thuật hạt nhân, đây là tính năng có không gian đầu vào có chiều sâu thấp và biến đổi nó thành không gian có chiều cao hơn, điều đó nghĩa là nó không phân chia các vấn đề thành các vấn đề riêng biệt, các tính năng này được gọi là hạt nhân. Nói một cách đơn giản nó thực hiện một số biến đổi dữ liệu phức tạp, sau đó tìm ra quá trình tách dữ liệu dựa trên các nhãn hoặc đầu ra mà chúng ta đã xác định trước.

2.3.2 Margin trong SVM

Margin là khoảng cách giữa siêu phẳng đến 2 điểm dữ liệu gần nhất tương ứng với các phân lớp.

SVM cố gắng tối ưu thuật toán bằng cách tìm các cách để tối đa hóa giá trị margin

này, từ đó tìm ra siêu phẳng đẹp nhất, tối ưu nhất để phân 2 lớp dữ liệu.

Cách tính margin:

- Trong không gian hai chiều, người ta tính margin giữa hai đường thẳng bằng công

thức: $margin = \frac{2}{\sqrt{w_1^2 + w_2^2}}$

- Với không gian nhiều chiều, ta phải tìm phương trình siêu phẳng có phương trình:

$$w^T x + b = 0$$

Margin sẽ được tính theo công thức: $margin = \frac{2}{\|w\|}$

2.3.3 Lập trình tìm nghiệm cho bài toán SVM

Tìm nghiệm cho SVM ta sử dụng trực tiếp thư viện sklearn.

2.3.4 Ưu nhược điểm thuật toán

Ưu điểm thuật toán

Xử lý trên không gian số chiều cao: SVM là một công cụ tính toán hiệu quả trong không gian chiều cao, trong đó đặc biệt áp dụng cho các bài toán phân loại văn bản và phân tích quan điểm nơi chiều có thể cực kỳ lớn.

Tiết kiệm bộ nhớ: Do chỉ có một tập hợp con của các điểm được sử dụng trong quá trình huấn luyện và ra quyết định thực tế cho các điểm dữ liệu mới nên chỉ có những điểm cần thiết mới được lưu trữ trong bộ nhớ khi ra quyết định.

Tính linh hoạt - phân lớp thường là phi tuyến tính. Khả năng áp dụng Kernel mới cho phép linh động giữa các phương pháp tuyến tính và phi tuyến tính từ đó khiến cho hiệu suất phân loại lớn hơn.

Nhược điểm thuật toán

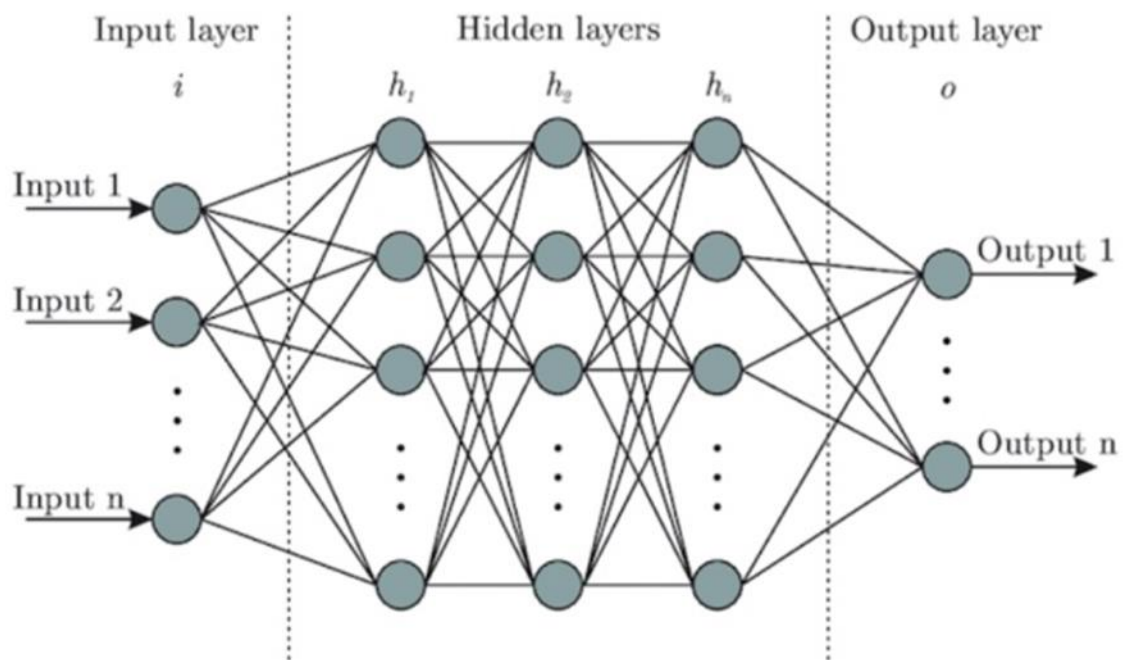
Bài toán số chiều cao: Trong trường hợp số lượng thuộc tính (p) của tập dữ liệu lớn hơn rất nhiều so với số lượng dữ liệu (n) thì SVM cho kết quả khá tồi. Chưa thể hiện rõ tính xác suất: Việc phân lớp của SVM chỉ là việc cố gắng tách các đối tượng vào hai lớp được phân tách bởi siêu phẳng SVM. Điều này chưa giải thích được xác suất xuất hiện của

một thành viên trong một nhóm là như thế nào. Tuy nhiên hiệu quả của việc phân lớp có thể được xác định dựa vào khái niệm margin từ điểm dữ liệu mới đến siêu phẳng phân lớp mà chúng ta đã bàn luận ở trên.

2.4 Neuron

2.4.1 Kiến trúc mạng neuron

Mạng NN là sự kết hợp của các tầng perceptron hay còn được gọi là perceptron đa tầng (*multilayer perceptron*) như hình vẽ bên dưới:



Hình 2.10: Mô hình Perceptron đa tầng

Chú thích:

- input layer: Tầng đầu vào
- hidden layer: Tầng ẩn
- output layer: Tầng đầu ra

Mỗi mô hình luôn có 1 tầng đầu vào, 1 tầng đầu ra, có thể có hoặc không các tầng ẩn. Tổng số tầng trong mô hình được quy ước là số tầng – 1 (Không tính tầng đầu vào).

2.4.2 Một mạng NN sẽ có 3 tầng

Tầng đầu vào (input layer): Là tầng bên trái cùng của mạng thể hiện cho các đầu vào của mạng.

Tầng đầu ra (output layer): Là tầng bên phải cùng của mạng thể hiện cho các đầu ra của mạng.

Tầng ẩn (hidden layer): Là tầng nằm giữa tầng vào và tầng ra thể hiện cho việc suy luận logic của mạng.

Lưu ý: một NN chỉ có 1 tầng vào và 1 tầng ra nhưng có thể có nhiều tầng ẩn.

2.4.3 Lan truyền tiến

Tất cả các nốt mạng (nơ-ron) được kết hợp đôi một với nhau theo một chiều duy nhất từ tầng vào tới tầng ra. Tức là mỗi nốt ở một tầng nào đó sẽ nhận đầu vào là tất cả các nốt ở tầng trước đó mà không suy luận ngược lại. Hay nói cách khác, việc suy luận trong mạng NN là suy luận tiến.

2.4.4 Quy trình thuật toán

Yêu cầu

Cho trước m bộ dữ liệu để đào tạo thuật toán. Bộ dữ liệu thứ i chứa vector input $x^{(i)}$ và vector output $y^{(i)}$. Nhiệm vụ của ta là tìm ra các hệ số $W_{pq}^{(l)}$ của mô hình Neural Network đã chọn trước.

Thuật toán

Bước 1: Chọn các giá trị $W_{pq}^{(l)}$ ngẫu nhiên và một giá trị learning rate α .

Bước 2: Liên tiếp lặp lại các phép biến đổi với các đạo hàm riêng được tính bằng thuật toán Backpropagation.

$$W_{pq}^{(l)} := W_{pq}^{(l)} - \alpha J'_{W_{pq}^{(l)}}(W)$$

Bước 3: Thuật toán dừng lại khi $J(W)$ thay đổi rất nhỏ hoặc trị tuyệt đối các đạo hàm riêng rất nhỏ. Nếu thuật toán không thể kết thúc thì chọn giá trị α nhỏ hơn rồi quay lại

Bước 2.

2.4.5 Xây dựng mô hình mạng neuron

Số nút ở lớp input là số phần tử thuộc một vector input không tính phần tử 1.

Thông thường ta sẽ chỉ dùng 1 lớp ẩn cho mạng neuron. Càng nhiều lớp ẩn sẽ càng có khả năng giải quyết các bài toán phức tạp với độ chính xác cao hơn nhưng thời gian thực hiện sẽ lâu hơn. Nếu có nhiều lớp ẩn thì số nút trong các lớp đó thường được chọn bằng nhau.

Trong các bài toán phân loại 2 lớp thì ta chỉ cần 1 nút ở lớp output là đủ giống như Logistic Regression. Với các bài toán phân loại k lớp thì ta cần k nút ở lớp output. Khi đó ta cần biến đổi output của dữ liệu đào tạo dưới dạng vector. Ví dụ với $k = 3$ thì output sẽ nhận một trong ba giá trị.

$$[1 \ 0 \ 0], [0 \ 1 \ 0], [0 \ 0 \ 1]$$

Đối với input mới giả sử kết quả phương trình giả thuyết trả về là:

$$[0.2 \ 0.9 \ 0.1]$$

Ta sẽ dự đoán nó thuộc lớp phân loại thứ 2 do thành phần thứ 2 của vector output có giá trị lớn nhất.

CHƯƠNG 3: XÂY DỰNG ỨNG DỤNG

3.1 Tổng quan về tin giả và phát hiện tin giả

3.1.1 Tin giả

Thuật ngữ "tin giả" là một khái niệm tương đối mới và cho đến nay vẫn chưa có một định nghĩa chung được thống nhất về tin tức giả mạo hay tin giả (Fake News). Theo từ điển Oxford "Tin giả là thông tin sai sự thật được phát sóng hoặc xuất bản dưới dạng tin tức nhằm mục đích lừa đảo hoặc có động cơ chính trị. Tin giả tạo ra sự nhầm lẫn đáng kể của công chúng về các sự kiện hiện tại.

Tin giả bùng nổ trên phương tiện truyền thông xã hội, đang xâm nhập vào các kênh truyền thông chính". Học giả về truyền thông Nolan Higdson đã định nghĩa "Tin tức giả là nội dung sai sự thật hoặc gây hiểu lầm được trình bày dưới dạng tin tức và được truyền đạt dưới các định dạng bao gồm truyền thông nói, viết, in, điện tử và kỹ thuật số".

Tin tức giả mạo cũng đề cập đến những câu chuyện bịa đặt có rất ít hoặc không có sự thật và khó có thể xác minh được. Thậm chí rộng hơn, sau kỳ bầu cử tổng thống Mỹ năm 2020, người ta đã mở rộng ý nghĩa của "tin tức giả" để bao gồm cả các tin tức tiêu cực về niềm tin và hành động cá nhân của họ.

Phân loại tin giả:



Hình 11: Biểu đồ phân loại tin giả

Thông tin sai lệch (misinformation) là thông tin được phổ biến mà không có ý định

gây hại. Thông tin sai lệch có thể được chia thành hai loại chính:

- + Kết nối sai (false connection): Đây là trường hợp khi tiêu đề, hình ảnh hoặc chú thích không phù hợp với nội dung. Ví dụ, một tiêu đề gây chú ý nhưng không liên quan đến nội dung thực tế, hoặc sử dụng hình ảnh không phù hợp để thu hút sự quan tâm. Ví dụ, sử dụng các hình ảnh kinh dị hoặc hấp dẫn để lôi kéo người đọc truy cập.

- + Nội dung gây hiểu lầm (misleading content): Đây là trường hợp sử dụng thông tin sai lệch và gây hiểu lầm cho người đọc. Ví dụ, quảng cáo hoặc trang web cố gắng đánh lừa khách hàng để truy cập vào các trang web không an toàn. Nội dung gây hiểu lầm có thể bao gồm cả những nội dung có thể được coi là lừa đảo, gian lận hoặc có hại cho khách truy cập trang web thông qua các tuyên bố không có căn cứ, ưu đãi miễn phí hoặc hứa hẹn về giảm giá, quảng cáo gây hiểu lầm và quảng bá các sản phẩm và dịch vụ của bên thứ ba.

Thông tin giả mạo (disinformation) là thông tin được tạo ra và chia sẻ bởi những người có ý định gây hại. Thông tin giả mạo có thể được chia thành ba loại chính:

- + Bối cảnh sai (false context): Loại thông tin giả mạo này được sử dụng để mô tả nội dung xác thực nhưng đã được điều chỉnh lại theo những cách nguy hiểm. Ví dụ, sử dụng một sự kiện thực tế như một cơ sở để truyền tải ý đồ chính trị hoặc gây rối.

- + Nội dung mạo danh (imposter content): Đây là loại thông tin giả mạo bằng cách sử dụng danh tính hoặc tin tức từ nhân vật hoặc nguồn tin uy tín. Người tạo tin giả sẽ cố gắng giả mạo là nội dung do những cá nhân, tổ chức nổi tiếng cung cấp hoặc đã được họ chấp nhận. Ví dụ, việc sử dụng tên những người nổi tiếng để quảng cáo sai sự thật đã trở thành một vấn đề phổ biến gây khó khăn cho người tiêu dùng.

- + Nội dung bị thao túng (manipulated content): Đây là trường hợp khi một khía cạnh nào đó của nội dung chính hãng bị thay đổi. Điều này thường liên quan đến ảnh hoặc video. Ví dụ, cố tình chỉnh sửa ảnh hoặc video để truyền tải thông tin sai lệch hoặc gây hiểu lầm.

Thông tin độc hại (Mal-information): Đây là loại thông tin được chia sẻ dưới danh nghĩa "chính hãng" nhưng với mục đích gây hại hoặc tạo ra hậu quả tiêu cực. Có một số hình thức thông tin độc hại như sau:

+ Rò rỉ (Leaks): Rò rỉ thông tin là khi thông tin bí mật hoặc nhạy cảm được tiết lộ cho những người hoặc bên không có quyền truy cập hoặc phổ biến thông tin đó. Ví dụ, trong các cuộc bầu cử tổng thống hoặc các sự kiện chính trị quan trọng, thông tin được cho là rò rỉ từ các nguồn không chính thức có thể tạo ra khó khăn và tranh cãi trong quá trình bầu cử hoặc gây ảnh hưởng đến quyết định của cử tri.

+ Quấy rối (Harassment): Đây là hành vi sử dụng lời nói, hình ảnh hoặc hành động nhằm xúc phạm, làm nhục hoặc gây tổn thương cá nhân hoặc tổ chức khác. Trên mạng xã hội, các hành vi quấy rối có thể bao gồm việc gửi tin nhắn xúc phạm, phản đối công khai hoặc lan truyền thông tin sai lệch nhằm hạ thấp hoặc tạo thành hình ảnh xấu cho một cá nhân, nhóm hoặc tổ chức.

+ Gây chia rẽ và thù hận (Hate speech): Đây là loại thông tin biểu đạt sự căm phẫn, phỉ báng hoặc tạo ra sự kỳ thị đối với một cá nhân hoặc nhóm dựa trên thuộc tính như chủng tộc, dân tộc, giới tính, tình dục, tôn giáo, tuổi tác, khuyết tật về thể chất hoặc tinh thần. Nội dung gây chia rẽ và thù hận có thể gây ra sự căng thẳng xã hội, gây tổn hại đến tinh thần và quyền lợi của các cá nhân hoặc nhóm bị nhắm đến.

3.1.2 Phát hiện tin giả

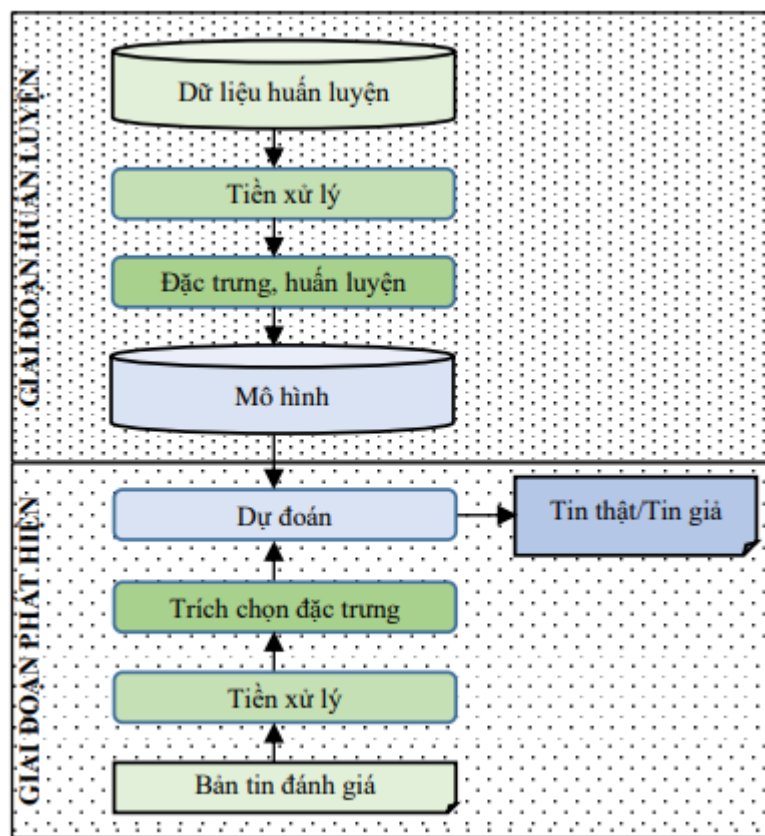
Việc phát hiện tin giả một cách thủ công thường liên quan đến tất cả các kỹ thuật và quy trình mà một người có thể sử dụng để xác minh tin tức. Tuy nhiên, lượng dữ liệu trực tuyến được tạo ra hàng ngày là quá lớn.

Hơn nữa, thông tin lan truyền trực tuyến rất nhanh nên việc kiểm tra thủ công nhanh chóng trở nên không hiệu quả và thiếu thực tế. Việc kiểm tra thủ công gặp khó khăn lớn nhất khi mở rộng quy mô xác minh do khối lượng dữ liệu được tạo ra quá lớn và nhanh. Do đó, nhiệm vụ phát hiện tự động tin giả là một nhu cầu cấp bách và quan trọng. Các nghiên cứu trước đây đã cho thấy sự khác biệt về khái niệm cũng như sự tương đồng giữa nhiều thuật ngữ liên quan đến “tin giả”. Bên cạnh đó, các nghiên cứu cũng chỉ ra các phương pháp để xác minh tin giả.

Tuy nhiên, để phát hiện tự động tin giả thì cần phải có các nghiên cứu sâu hơn. Các nghiên cứu hiện nay đang tiến thêm một bước nữa bằng cách xác định các đặc điểm hoặc

chỉ số hoạt động cụ thể liên quan đến bản tin để trên cơ sở đó có thể kết hợp xử lý ngôn ngữ tự nhiên (NLP) và đưa vào thuật toán học máy nhằm phân biệt một cách đáng tin cậy giữa các loại nội dung khác nhau được gắn với nhãn là “tin tức giả mạo”. Hệ thống phát hiện tự động tin giả sẽ giúp xác minh một tin tức là giả hay thật mà không cần sự can thiệp trực tiếp của con người. Có nhiều kỹ thuật và cách tiếp cận khác nhau được sử dụng trong nghiên cứu phát hiện tin giả. Các kỹ thuật và cách tiếp cận này phụ thuộc vào quan điểm và mục đích truy vết của người phát triển.

Vậy chúng tôi đưa ra mô hình tổng quát của hướng tiếp cận như hình sau:



Hình 12: Mô hình phân loại tin giả

Mô hình này bao gồm các giai đoạn chính sau:

Bước đầu tiên ta thu thập dữ liệu. Ở giai đoạn này, cần thu thập một tập dữ liệu đủ lớn và đa dạng để xây dựng cơ sở dữ liệu huấn luyện. Tập dữ liệu này bao gồm các bản tin đã được gắn nhãn là tin giả hoặc tin thật. Trong trường hợp học máy giám sát, tất cả dữ liệu huấn luyện đều phải được gắn nhãn. Trường hợp học bán giám sát bao gồm cả dữ liệu đã

và chưa được gán nhãn.

Đến bước tiền xử lý, giai đoạn này giúp làm sạch dữ liệu và loại bỏ thông tin không có ích. Các kỹ thuật xử lý ngôn ngữ tự nhiên được áp dụng để tiền xử lý dữ liệu, ví dụ như loại bỏ stopwords, chuẩn hóa từ ngữ, loại bỏ ký tự đặc biệt, và tách từ.

Tiếp theo là trích chọn đặc trưng giai đoạn này tập trung vào việc trích xuất những đặc trưng ngôn ngữ cần thiết để phục vụ cho việc phân loại và nhận dạng nội dung. Các đặc trưng này có thể bao gồm từ vựng, n-gram, vector hóa từ, đếm từ, hoặc các phương pháp khác để biểu diễn dữ liệu dưới dạng số.

Giai đoạn huấn luyện mô hình, trên cơ sở các đặc trưng đã trích xuất, mô hình được huấn luyện bằng cách sử dụng các thuật toán lựa chọn. Quá trình này giúp xây dựng mô hình đặc trưng có khả năng phân loại và nhận dạng tin giả và tin thật.

Dự đoán là giai đoạn này sử dụng mô hình đã được huấn luyện để dự đoán xem một bản tin cụ thể có phải là tin giả hay tin thật không. Các đặc trưng của bản tin được so sánh với mô hình đặc trưng đã tạo ra trong giai đoạn huấn luyện để đưa ra quyết định cuối cùng.

Trong bài nghiên cứu này, ta chỉ giới thiệu hướng tiếp cận khá phổ biến hiện nay là dựa trên các kỹ thuật học máy (Machine Learning) với các phương pháp truyền thống (Naïve Bayes, KNN, SVM,...). Các phương pháp này đều dựa trên phân tích nội dung để dự đoán tin giả.

Vậy bài toán đặt ra là:

Input	Output
Bài báo gồm các thông tin: tác giả, tiêu đề, nội dung bài viết.	Bài báo là tin giả hoặc tin thật

3.2 Công cụ thực hiện

Để phát triển các mô-đun chương trình, ta sử dụng ngôn ngữ lập trình Python. Python là ngôn ngữ được sử dụng phổ biến nhất trong học máy và thư viện mô hình được chọn để sử dụng là thư viện Scikit-learn (sklearn), sử dụng hàm MultinomialNB. Bên cạnh đó đề

tài có xử lý ngôn ngữ tự nhiên nên ta sử dụng lớp `TfidfVectorizer` để thực hiện đề tài. Môi trường được chọn để thử nghiệm là PyCharm và Google Colab (Google Colaboratory) vì đây là một dịch vụ miễn phí của Google nhằm hỗ trợ nghiên cứu và học tập về trí tuệ nhân tạo, có GPU để chạy các chương trình Python và hỗ trợ Học máy. Đặc biệt, trên môi trường Colaboratory có cài sẵn các thư viện học máy phổ biến. Ngoài ra, ta cũng có thể cài thêm các thư viện khác để chạy nếu cần.

Ngoài ra, ta thực hiện liên kết Google Colaboratory với Google Drive để lưu trữ và truy xuất dữ liệu nên rất tiện để sử dụng.

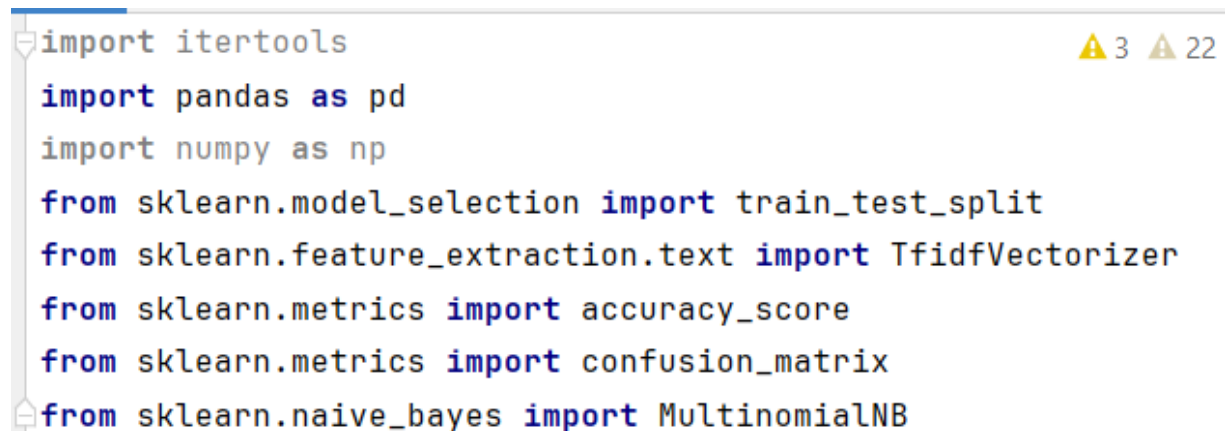
Để thực hiện đề tài ta cài đặt các thư viện bằng lệnh sau:

```
pip3 install pandas
```

```
pip3 install sklearn
```

```
pip3 install numpy
```

Tất cả các thư viện cần thiết:



```
import itertools
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.naive_bayes import MultinomialNB
```

Hình 13 : Các thư viện cần khai báo

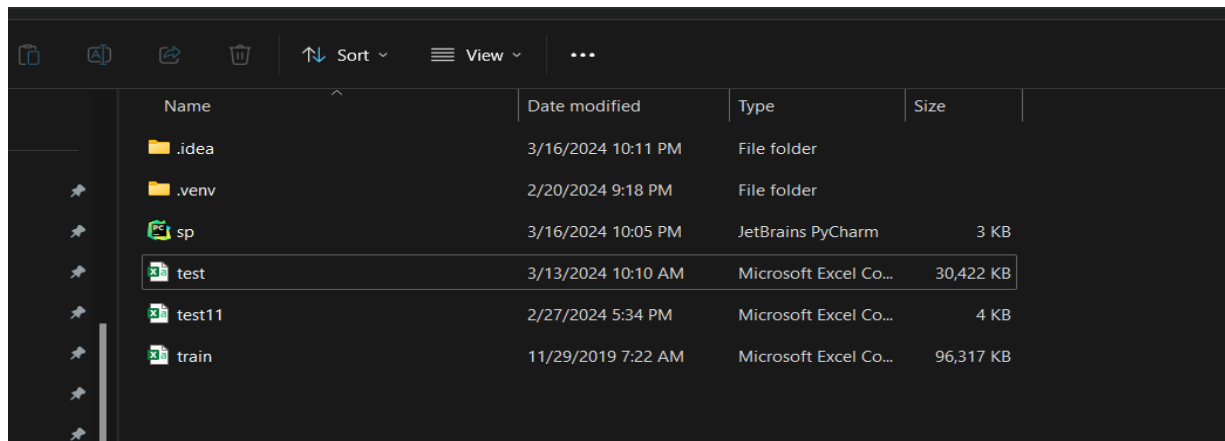
3.3 Xây dựng ứng dụng

3.3.1 Mô tả dữ liệu

Để kiểm chứng các phương pháp trình bày ở trên, ta đã tiến hành thử nghiệm với bộ dữ liệu Kaggle

Dữ liệu Kaggle là một nguồn tài nguyên dữ liệu trực tuyến được cung cấp bởi Kaggle, một nền tảng cộng đồng phổ biến cho các nhà khoa học dữ liệu và các chuyên gia trong lĩnh vực học máy và khoa học dữ liệu. Kaggle cung cấp cho người dùng truy cập vào hàng ngàn bộ dữ liệu từ nhiều lĩnh vực khác nhau như y học, tài chính, hình ảnh, ngôn ngữ tự nhiên, thị giác máy tính, và kể cả dữ liệu để thực hiện đề tài.

Chúng ta đã lấy các file dữ liệu sau:



Name	Date modified	Type	Size
.idea	3/16/2024 10:11 PM	File folder	
.env	2/20/2024 9:18 PM	File folder	
sp	3/16/2024 10:05 PM	JetBrains PyCharm	3 KB
test	3/13/2024 10:10 AM	Microsoft Excel Co...	30,422 KB
test11	2/27/2024 5:34 PM	Microsoft Excel Co...	4 KB
train	11/29/2019 7:22 AM	Microsoft Excel Co...	96,317 KB

Hình 14: File dữ liệu

Dữ liệu huấn luyện:

+Kích thước: 94.14 MB

+Số tin trong file: 20800 tin

+Định dạng: CSV

+Dữ liệu được chia thành các cột sau: id, tiêu đề, tác giả, văn bản và nhãn. Những thành phần thuộc tính này là thành phần cơ bản của một bài báo, tin tức hay của một bài viết,...

A	B	C	D	E
id	title	author	text	label
0	See Comey's Let	Darrell Lucus	ber 30, 2016 Subscribe Jason Chaffetz on the s when Comey sent his now-infamous letter ann tained classified information. Not long after this son Chaffetz (@jasoninthehouse) October 28, z. The Utah Republican had already vowed to i tter. "Democratic Ranking Members on the rele osive investigation, and neither Chaffetz nor his achine. That may make for good theater, but the isanship. He didn't even have the decency to n ublican House leadership has given its full supp is an unapologetic liberal. His desire to stand u	1
1	FLYNN: Hillary C	Daniel J. Flynn	Ever get the feeling your life circles the rounda	0
2	Why the Truth M	Consortiumnews	Why the Truth Might Get You Fired October 29 The tension between intelligence analysts and By Lawrence Davidson For those who might wonder why foreign polic Back in the early spring of 2003, George W. Bu For our purposes, we will concentrate on the b The short answer is Bush wanted, indeed need But the nuclear weapons gambit proved more What we had was a U.S. leadership cadre who So the U.S. and its allies insisted that the Unite On March 19, 2003, Bush launched the invas Social and Behavioral Sciences to the Rescue The various U.S. intelligence agencies were th A "partnership" is being forged between the Of Despite this effort, it is almost certain that the " The Believers It is simply not true, as the ODNI leaders seem Therefore, if someone feeds them "snake oil," Listen to Charles Gaukel, of the National Intelli I can certainly tell you what it means historica On the other hand, as long as what you're selli What does this sad tale tell us? If you want to : It has happened this way so often, and in so m	1

Hình 15: Dữ liệu huấn luyện

Nhưng ta chỉ quan tâm là **nhãn và cột văn bản** . Các **văn bản** cột chứa các **nội dung** của bài viết, trong khi các **nhãn** cột tượng trưng cho bài viết là thực tế hay không.

Dữ liệu thử nghiệm:

+Kích thước: 29.7 MB

+Số tin trong file: 5200 tin

+Định dạng: CSV

+Dữ liệu được chia thành các cột sau: id, tiêu đề, tác giả, văn bản. (Tương tự với dữ liệu huấn luyện nhưng bỏ đi thành phần “nhãn”)

id	title	author	text
20807	Weekly Featured Profile – Randy Shannon	Trevor Loudon	You are here: Home / *Articles of the Bound* / Weekly Featured Profile – Randy Shannon Weekly Featured Profile – Randy Shannon October 31, 2016, 7:21 am by Trevor Loudon Leave a Comment 0 KeyWiki.org Randy Shannon Randy Shannon is a Beaver County , Pennsylvania Democratic Party activist. "A Democratic victory in 2016 with a bigger progressive caucus can tax Wall Street, end austerity and discrimination, and put the nation to work building the solar infrastructure we desperately need." "We need progressives like Sanders, who support working families, running for President, for Senate, and for Congress wherever possible," said Randy Shannon , convener of the Sanders for President PA Exploratory Committee. Randy Shannon was a student leader in the 1960's at Duke University . He left Duke to organize campus groups for labor, peace, women's equality and civil rights in the South as a staff member of the Southern Student Organizing Committee . In Nashville , he was a leader of the anti-Vietnam War movement and the Free Angela Davis campaign. He was an organizer for the National Welfare Rights Organization and led a local delegation to the 1972 Democratic Convention in Miami to fight for a \$6400 guaranteed income. He ran in the TN 5th Congressional District Democratic primary in 1972 successfully targeting a right wing anti-busing candidate. He worked as a welder and organized rank and file workers as a member of Teamsters Local 327. He moved to Pittsburgh in 1976 and still works in the R&D sector of the basic materials industry. In PA he organized the Pittsburgh Youth Movement for Jobs , was active in the peace movement and progressive politics. In 1982, he moved to Beaver County and helped organize Beaver County Fightback , to defend the home ownership of unemployed steelworkers in the Ohio River Valley. He lead the Jesse Jackson campaign in the 4th and 22nd CDs of PA opening an office in Aliquippa in 1988. He also helped organize a ballot access campaign for Dennis Kucinich in 2004. He also helped organize the Beaver County Campaign for Nuclear Disarmament and is still active in Beaver County Peace Links . Up until 1991, Randy Shannon was a member of the Communist Party USA . Shannon helped organize the first Citizens Congressional Hearing on Medicare for All, chaired by Rep. Dennis Kucinich in Aliquippa , PA in May 2005 and was an early advocate in Progressive Democrats of America for Medicare for All. He has worked for ten years building a local chapter of PDA that reflects the progressive coalition of minority, labor and progressive activists. For the last nine years he has been a member of the National Coordinating Committee of the Committees of Correspondence for Democracy and Socialism and a member of Democratic Socialists of America . Randy Shannon attended the 6th National Convention of the Committees of Correspondence for Democracy and Socialism (CCDS) at San Francisco's Whitcomb Hotel, July 23-26, 2009. The "Building the Progressive Majority: Race, Class and Gender" plenary discussion began a series of panel and workshop discussions. The plenary panel consisted of reports highlighting work of CCDS activists in the South, in the Heartland "rustbelt states," on the West Coast and New England and the East Coast. Randy Shannon's report on Western Pennsylvania and the dire conditions in the wake of de-industrialization was particularly moving. He described independent political work with groups like Progressive Democrats of America in raising the consciousness and unity of the working class and Black community, and then in turn ally with forces like the Congressional Progressive Caucus in the Congress to defeat the right and advance progressive planks in Obama's economic package. He stressed the importance of ending the wars and healthcare reform, especially HR 676 "Medicare for All."

Hình 16: Dữ liệu thử nghiệm

Để nhanh chóng chúng ta sẽ chỉ lấy một tin với id là 20807 để chạy thử nghiệm để ra kết quả.

3.3.2 Thuật toán cài đặt

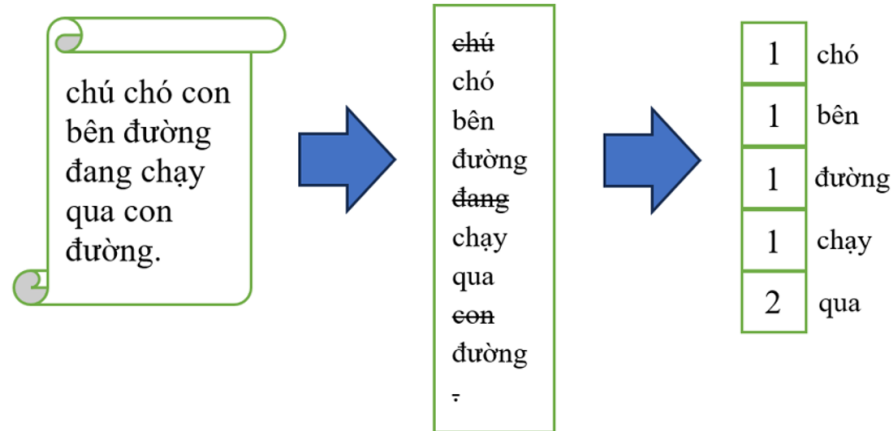
Chúng ta sẽ kết hợp mô hình học máy và NLP (xử lý ngôn ngữ tự nhiên) để phát hiện tin giả.

Mô hình học máy Naïve Bayes, SVM, KNN được sử dụng chúng ta đã nêu ra ở chương 2.

Kiến thức NLP được chúng ta áp dụng là:

Chuyển các văn bản sang các vector

Trong xử lý ngôn ngữ tự nhiên, chúng ta cần biểu diễn các câu, từ và văn bản dưới dạng các vector (vectorization) để máy tính có thể áp dụng các thuật toán và phương pháp học máy. Một phương pháp phổ biến là sử dụng mô hình "Túi từ" (Bag-of-Words) hoặc "Túi từ và Tf-Idf" (Term Frequency-Inverse Document Frequency).



Hình 17: Cách thức hoạt động của mô hình

Trong mô hình "Túi từ", mỗi từ trong văn bản được coi là một đặc trưng và được đếm số lần xuất hiện trong văn bản. Các từ sẽ được biểu diễn thành một vector dựa trên số lần xuất hiện của chúng. Mô hình "Túi từ và Tf-Idf" cải tiến bằng cách tính toán giá trị Tf-Idf cho từng từ, kết hợp tần số xuất hiện (Term Frequency - Tf) và tần số nghịch đảo của văn bản (Inverse Document Frequency - Idf).

Quá trình chuyển các văn bản thành các vector trong xử lý ngôn ngữ là một bước quan trọng để máy tính có thể hiểu và xử lý ngôn ngữ tự nhiên, và nó cung cấp cơ sở cho nhiều ứng dụng như phân loại văn bản, dịch máy, phân tích cảm xúc và nhiều hơn nữa.

Để hiểu rõ hơn về "Túi từ và Tf-Idf" (Term Frequency-Inverse Document Frequency), hãy xem một ví dụ cụ thể:

Văn bản 1: "Tôi thích mèo"

Văn bản 2: "Tôi thích chó"

Văn bản 3: "Tôi ghét nhện"

Tần suất từ (Tf)

Tf đại diện cho tần suất xuất hiện của một từ trong một văn bản. Chúng ta đếm số lần xuất hiện của mỗi từ trong mỗi văn bản.

Giá trị Tf cho mỗi văn bản:

| "Tôi" | "thích" | "mèo" | "chó" | "ghét" | "nhện" |

Văn bản 1 | 1 | 1 | 1 | 0 | 0 | 0 |

Văn bản 2 | 1 | 1 | 0 | 1 | 0 | 0 |

Văn bản 3 | 1 | 0 | 0 | 0 | 1 | 1 |

Tần suất ngược của văn bản (Idf)

Idf đo độ thông tin của một từ trong toàn bộ tập hợp các văn bản. Nó được tính là logarithm cơ số tự nhiên của tổng số văn bản chia cho số văn bản chứa từ đó.

Giá trị Idf cho mỗi từ:

| "Tôi" | "thích" | "mèo" | "chó" | "ghét" | "nhện" |

Idf | 0 | 0 | 1.09 | 1.09 | 1.09 | 1.09 |

Tf-Idf

Tf-Idf là tích của Tf và Idf. Nó đại diện cho sự quan trọng của một từ trong một văn bản cụ thể so với toàn bộ tập hợp các văn bản.

Giá trị Tf-Idf cho mỗi từ trong mỗi văn bản:

| "Tôi" | "thích" | "mèo" | "chó" | "ghét" | "nhện" |

Văn bản 1 | 0 | 0 | 1.09 | 0 | 0 | 0 |

Văn bản 2 | 0 | 0 | 0 | 1.09 | 0 | 0 |

Văn bản 3 | 0 | 0 | 0 | 0 | 1.09 | 1.09 |

Trong ví dụ này, các từ "mèo," "chó," "ghét," và "nhện" có giá trị Tf-Idf khác không trong các văn bản tương ứng của chúng, cho thấy sự quan trọng của chúng trong các văn bản đó so với các từ khác. Từ "Tôi" có giá trị Tf-Idf bằng không trong tất cả các văn bản, cho thấy nó không cung cấp nhiều thông tin phân loại. Công thức tính toán Tf-Idf cho một từ trong một văn bản cụ thể như sau:

$$\text{Tf-Idf} = \text{Tf} * \log(N / \text{DF})$$

Trong đó:

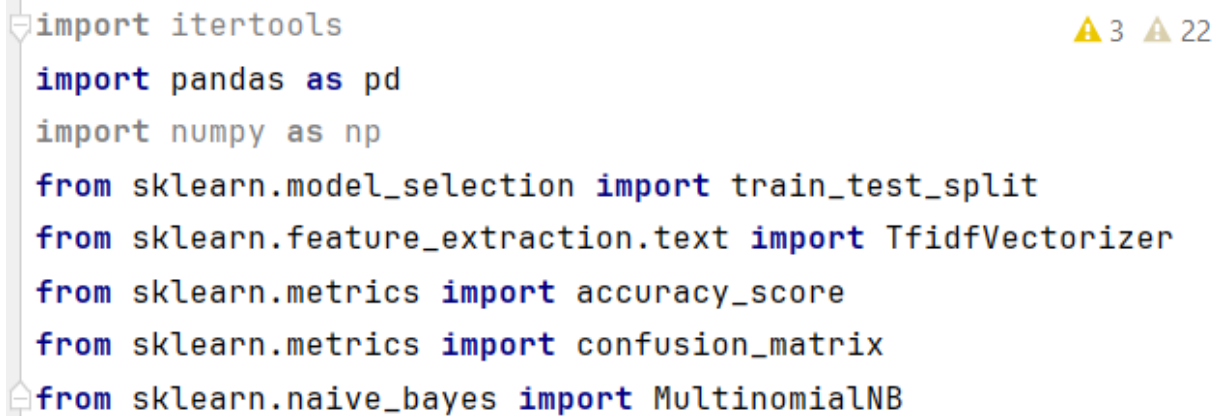
- + Tf là tần số xuất hiện của từ trong văn bản.
- + N là tổng số văn bản trong tập dữ liệu.
- + DF là số lượng văn bản trong tập dữ liệu chứa từ đó.

Bằng cách tính toán Tf-Idf cho các từ trong mỗi văn bản, chúng ta có thể biểu diễn các văn bản dưới dạng các vector số học, trong đó mỗi thành phần trong vector tương ứng với giá trị Tf-Idf của từ trong văn bản đó. Điều này cho phép chúng ta sử dụng các phương pháp và thuật toán dựa trên số học để xử lý và phân tích các văn bản.

Giờ ta sẽ biểu diễn hết toàn bộ dữ liệu thành vector đã có ở trên để áp dụng được mô hình Naive Bayes.

3.3.3 Chương trình ứng dụng

Chúng tôi sẽ bắt đầu bằng cách nhập tất cả các thư viện cần thiết:



```
import itertools
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.naive_bayes import MultinomialNB
```

Hình 18: Code nhập thư viện

Bây giờ là lúc nhập tập dữ liệu của chúng tôi và lấy hình dạng của dữ liệu.


```
# Import dataset
df=pd.read_csv('train.csv')

# Get the shape
df.shape
```

Hình 19: Code nhập và lấy dạng của dữ liệu

Rõ ràng là tệp csv chứa tập dữ liệu gồm 20800 hàng và năm tính năng riêng biệt (cột). Để có thêm một số thông tin chi tiết liên quan đến dữ liệu, tôi sẽ tiếp tục bằng cách đi đầu.

```
# Get the head
df.head()
```

Hình 20: Code thêm thông tin

Bây giờ ta có thể xem bốn bản ghi đầu tiên. Bằng cách đó, ta thấy rằng tập dữ liệu được chia thành các cột sau: id, tiêu đề, tác giả, văn bản và nhãn.

Các tính năng được quan tâm là **nhãn và cột văn bản**. Các **văn bản** cột chứa các **nội dung** của bài viết, trong khi các **nhãn** cột tượng trưng cho dù bài viết là thực tế hay không.

Điều này đã được tạo sẵn ở dạng nhị phân sử dụng '1' và '0' bằng cách sử dụng:

‘1’ for FAKE NEWS

‘0’ for RELIABLE NEWS

```
# Change the labels
df.loc[(df['label'] == 1), ['label']] = 'FAKE'
df.loc[(df['label'] == 0), ['label']] = 'REAL'
```

Hình 21: Code xác định labels

Bây giờ ta sẽ tách các nhãn khỏi phần còn lại của khung dữ liệu.

```
# Isolate the labels
labels = df.label
labels.head()
```

Hình 22: Code tách nhãn ra khỏi dữ liệu

Khi hoạt động trước đó đã kết thúc, tập dữ liệu phải được chia thành hai bộ riêng biệt. 80% dữ liệu sẽ được sử dụng để đào tạo mô hình và 20% còn lại sẽ dùng làm dữ liệu thử nghiệm.

```
#Split the dataset
x_train,x_test,y_train,y_test=train_test_split(*arrays: df['text'].values.astype('str'), labels, test_size=0.2, random_state=7)
```

Hình 23: Code chia tập dữ liệu

Bây giờ ta sẽ khai báo một TfidfVectorizer sử dụng các từ dừng từ tiếng Anh (phụ thuộc vào ngôn ngữ của bài báo) và cho phép tần suất tài liệu lên đến 0,7.

```
#Initialize a TfidfVectorizer
tfidf_vectorizer=TfidfVectorizer(stop_words='english', max_df=0.7)
```

Hình 24: Code chuyển văn bản thành các đặc trưng

Bây giờ ta có một vectorizer, ta sẽ điều chỉnh và biến đổi nó trên tập huấn luyện và cũng biến đổi nó trên tập thử nghiệm.

```
# Fit & transform naive_bayes train set, transform test set
tfidf_train=tfidf_vectorizer.fit_transform(x_train)
tfidf_test=tfidf_vectorizer.transform(x_test)
```

Hình 25: Code biến đổi tập huấn luyện và tập thử nghiệm

MultinomialNB sẽ được khởi tạo tương tự ta cũng có thể sử dụng SVC của thư viện sklearn.svm và KNeighborsClassifier của sklearn.neighbors. Để kết hợp vào mô hình, ta sử dụng “y_train” và “tfidf_train”.

```
# Initialize the and fit training sets
nb_classifier = MultinomialNB()
nb_classifier.fit(tfidf_train,y_train)
```

Hình 26: Code sử dụng MultinomialNB

Cuối cùng, sử dụng vectorizer để dự đoán liệu một bài báo có đáng tin cậy hay không và sẽ tính toán độ chính xác của mô hình.

```
# Predict and calculate accuracy
y_pred=nb_classifier.predict(tfidf_test)
score=accuracy_score(y_test,y_pred)
print(f'Accuracy: {round(score*100,2)}%')

# Build confusion matrix
cm = confusion_matrix(y_test,y_pred, labels=['FAKE','REAL'])
print("Our model successfully predicted", cm[0][0], "positives")
print("Our model successfully predicted", cm[1][1], "negatives.")
print("Our model predicted", cm[0][1], "false positives")
print("Our model predicted", cm[1][0], "false negatives")
```

Hình 27: Code tính toán độ tin cậy

Từ ma trận, chúng ta có thể đưa ra các kết quả đánh giá sau:

```

Accuracy: 85.62%
Our model successfully predicted 1525 positives
Our model successfully predicted 2037 negatives.
Our model predicted 575 false positives
Our model predicted 23 false negatives

```

Hình 28: Kết quả đánh giá

Dự đoán: Chương trình lấy một tin tức (dữ liệu với id là “20807”) bất kì của dữ liệu thử nghiệm và trả kết quả nhận dạng được tin giả hay thật.

```

# Load the test data from a CSV file
test_data = pd.read_csv(filepath_or_buffer='test11.csv', encoding='latin-1')

# Preprocess the test data
test_text = test_data['text']. values.astype('str')
print(test_text)
tfidf_test = tfidf_vectorizer. transform(test_text)

# Make predictions
prediction = nb_classifier.predict(tfidf_test)

# Map the predicted label to "This news is fake" or "This news is real"
if prediction[0] == 'FAKE':
    result = "This news is fake"
else:
    result = "This news is real"

# Print the result
print(result)

```

Hình 29: Code thực nghiệm trên một dữ liệu thử nghiệm

Kết quả trả lại là:

This news is real

Process finished with exit code 0

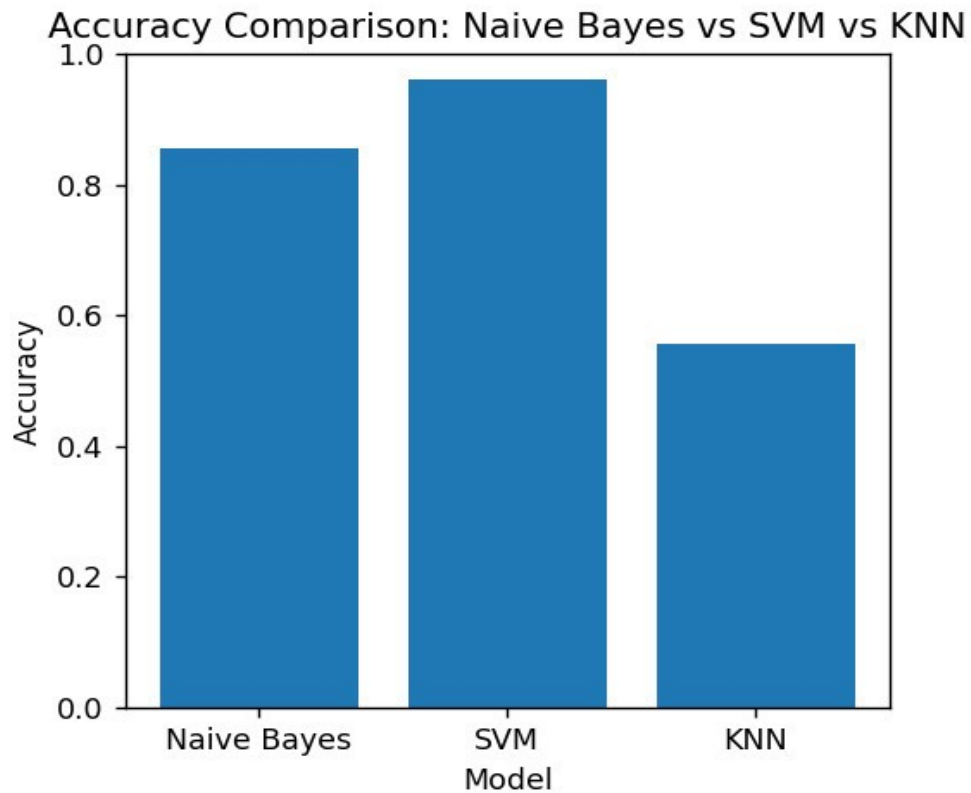
Hình 30: Kết quả trả về

Sau khi chạy thử thành công thuật toán Naïve Bayes, chúng ta có thể chạy tiếp hai thuật toán SVM, KNN và có kết quả sau:

Bảng 3: Bảng so sánh đánh giá các mô hình học máy

Naïve Bayes accuracy	SVM accuracy	KNN accuracy
85.62%	96.23%	55.6%

Ta có biểu đồ so sánh:



Hình 31: Biểu đồ so sánh các thuật toán

KẾT LUẬN

Trong quá trình nghiên cứu, chúng em đã cố gắng hết sức để tìm hiểu, học hỏi, nhưng vì khả năng còn nhiều giới hạn nên chưa giải quyết được tất cả những vấn đề đặt ra. Chúng em mong nhận được sự chỉ bảo của thầy cô.

Những kết quả đạt được:

- Sự hiểu biết về mô hình Naïve Bayes, KNN, SVM và hiệu suất của nó.
- Sự hiểu biết về quá trình chuyển đổi các văn bản thành các vector để máy tính có thể hiểu và xử lý ngôn ngữ tự nhiên.
- Từ những điều học được, hiểu biết thêm về AI khi chúng được áp dụng vào thực tế đời sống.

Những hạn chế:

- Kiến thức vẫn chưa đủ để tìm ra những mô hình phù hợp hơn.
- Yếu tố địa phương và văn hóa: Các mô hình nhận biết tin giả có thể bị ảnh hưởng bởi yếu tố địa phương và văn hóa.
- Độ chính xác của nhãn dữ liệu: Xác định chính xác nhãn (fake/real) cho từng tin tức trong tập dữ liệu có thể là một thách thức. Sự chủ quan và mâu thuẫn trong đánh giá tin tức có thể dẫn đến sự không đồng nhất trong nhãn và ảnh hưởng đến hiệu suất của mô hình.
- Độ phức tạp của ngôn ngữ và ngữ cảnh: Tin giả có thể sử dụng các chiêu trò và kỹ thuật để lừa đảo hệ thống nhận biết. Các ngôn ngữ lạc quan, nghĩa vụ, hoặc các chiêu trò ngữ cảnh khác có thể gây khó khăn trong việc phân loại chính xác tin giả.
- Độ tính vi của tin giả ngày càng tăng, họ luôn cải thiện các kỹ năng tránh để phát hiện ra. Đó là cũng là một thách thức cho đề tài
- Bộ dữ liệu: Muốn có một hệ thống phát hiện tin giả trước hết phải có bộ dữ liệu đủ lớn và luôn liên tục cập nhật. Tuy nhiên, các tin tức liên tục phát sinh khiến khối lượng rất lớn nên việc thu thập khó khăn.

TÀI LIỆU THAM KHẢO

[1]https://vi.wikipedia.org/wiki/X%E1%BB%AD_l%C3%BD_ng%C3%B4n_ng%E1%B%AF_t%E1%BB%B1_nhi%C3%AAn

[2]<https://viblo.asia/p/ly-thuyet-ve-mang-bayes-va-ung-dung-vao-bai-toan-loc-thu-rac-07LKXzkelV4>

[3]Daniel Jurafsky, James H. Martin. 2009, Prentice-Hall 2nd edition. Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics.

[4] Võ Trung Hùng, Ninh Khánh Chi, Trần Anh Kiệt 2022, “Automatic Fake News Detection: Achievements And Challenges”.