TRƯỜNG ĐẠI HỌC ĐIỆN LỰC

KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO CHUYÊN ĐỀ HỌC PHẦN

NGÔN NGỮ LẬP TRÌNH PYTHON

ĐỀ TÀI :

ỨNG DỤNG THUẬT TOÁN NAIVE BAYES ĐỂ PHÂN LOẠI ĐÁNH GIÁ SẢN PHẨM

Sinh viên thực hiện	**: NGUYỄN MẠNH HƯNG**
	NGUYỄN MỸ LINH
Lớp	**: D15TTNT&TGMT**
Giảng viên hướng dẫn	**: ThS. LÊ MẠNH HÙNG**
Ngành	**: CÔNG NGHỆ THÔNG TIN**
Chuyên ngành	**: TTNT & TGMT**
Khóa	**: 2020-2025**

Hà Nội, Tháng 11 Năm 2023

PHIẾU CHẤM ĐIỂM

Sinh viên thực hiện

Họ và tên	Điểm	Chữ ký	Ghi Chú
Nguyễn Mạnh Hưng 20810310347			
Nguyễn Mỹ Linh 20810310311			

Giảng viên chấm

Họ và tên	Chữ ký	Ghi chú
Giảng viên chấm 1		
Giảng viên chấm 2		

MỤC LỤC

DANH MỤC HÌNH ẢNH

LỜI NÓI ĐẦU

Trong thời đại hiện nay, Công nghệ thông tin đang trở thành một yếu tố then chốt đối với sự phát triển và tương tác trong xã hội. Sự gia tăng với tốc độ chóng mặt của dữ liệu số và thông tin trên Internet đã đặt ra những thách thức lớn về việc xử lý và hiểu biết về nội dung trực tuyến. Một trong những nhiệm vụ quan trọng trong lĩnh vực này là phân loại văn bản nơi mà mô hình Naive Bayes nổi lên như một công cụ quan trọng và hiệu quả.

Ngày nay, người dùng Internet có khả năng đánh giá và chia sẻ ý kiến của họ với tốc độ không ngừng. Những đánh giá này không chỉ đóng vai trò là nguồn thông tin quý báu về sản phẩm và dịch vụ, mà còn là một động lực mạnh mẽ đằng sau quyết định mua sắm và sự tin tưởng của khách hàng. Để hiểu rõ hơn về ý kiến của người dùng và tận dụng thông tin đánh giá, phân loại đánh giá trở thành một công cụ không thể thiếu trong phân tích dữ liệu và trí tuệ nhân tạo.

Trong hệ thống phân loại đánh giá, mô hình Naive Bayes đã chứng minh sức mạnh của mình thông qua sự đơn giản và hiệu quả. Khả năng xử lý dữ liệu lớn, độ chính xác cao và khả năng làm việc tốt với dữ liệu văn bản là những yếu tố làm nổi bật Naive Bayes trong việc đưa ra dự đoán chính xác về ý kiến của người dùng dựa trên đánh giá.

Trong báo cáo này, chúng em sẽ tìm hiểu chi tiết về cách mô hình Naive Bayes hoạt động trong phân loại văn bản và những ứng dụng thực tế của nó trong lĩnh vực Công nghệ thông tin.

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1. Đặt vấn đề

Công nghệ thông tin ngày càng được phổ biến đến tất cả mọi người, ứng dụng công nghệ thông tin vào việc lưu trữ và xử lý thông tin ngày nay được áp dụng hầu hết vào mọi loại lĩnh vực. Điều này đã tạo ra một lượng lớn dữ liệu được lưu trữ với kích thước lớn. Trong thời kỳ số hóa mạnh mẽ hiện nay, dữ liệu người dùng trên các nền tảng trực tuyến, đặc biệt là đánh giá và ý kiến, trở thành một nguồn thông tin quan trọng. Tuy nhiên, việc hiểu và phân loại đánh giá này đối với doanh nghiệp là một thách thức, do lượng thông tin lớn và độ đa dạng của ngôn ngữ tự nhiên. Điều này làm nảy sinh nhu cầu sử dụng các mô hình máy học, trong đó mô hình Naive Bayes nổi bật như một công cụ hiệu quả.

1.2. Mục tiêu nghiên cứu

Đề tài tập chung vào nghiên cứu kỹ thuật phân lớp trong khai phá dữu liệu, từ đó nắm bắt được những giải thuật làm tiền đề cho nghiên cứu và xây dựng ứng dụng cụ thể. Mục tiêu chính của đề tài là áp dụng và nghiên cứu mô hình Naive Bayes trong việc phân loại đánh giá người dùng trên các nền tảng trực tuyến. Chúng ta sẽ tập trung vào việc xây dựng một hệ thống phân loại đánh giá tự động có khả năng hiểu và tận dụng ý kiến người dùng để hỗ trợ quyết định kinh doanh.

1.3. Ý nghĩa của đề tài

Nghiên cứu này không chỉ mang lại lợi ích thực tiễn cho doanh nghiệp trong việc hiểu biết ý kiến của khách hàng một cách tự động và nhanh chóng, mà còn góp phần vào sự phát triển của lĩnh vực trí tuệ nhân tạo và phân loại dữ liệu văn bản. Việc áp dụng mô hình Naive Bayes trong lĩnh vực này có thể mở ra những triển vọng mới trong nghiên cứu và ứng dụng của trí tuệ nhân tạo trong xử lý ý kiến người dùng. Đồng thời, đề tài cũng hướng tới việc cung cấp giải pháp hiệu quả cho doanh nghiệp trong quản lý và tối ưu hóa chất lượng dịch vụ dựa trên phản hồi người dùng.

CHƯƠNG 2: TỔNG QUAN VỀ NAÏVE BAYES

2.1. Giới thiệu về thuật toán

Naive Bayes Classification (NBC) là một thuật toán dựa trên định lý Bayes về lý thuyết xác suất để đưa ra các phán đoán cũng như phân loại dữ liệu dựa trên các dữ liệu được quan sát và thống kê. Naive Bayes Classification là một trong những thuật toán được ứng dụng rất nhiều trong các lĩnh vực Machine learning dùng để đưa các dự đoán chính xác nhất dự trên một tập dữ liệu đã được thu thập, vì nó khá dễ hiểu và độ chính xác cao. Nó thuộc vào nhóm Supervised Machine Learning Algorithms (thuật toán học có hướng dẫn), tức là máy học từ các ví dụ từ các mẫu dữ liệu đã có.

Ví dụ như ta có thể ứng dụng vào việc thiết kế một ứng dụng nghe nhạc có thể phán đoán được sở thích của nghe nhạc của người dùng dựa trên các hành vi như nhấn nút "thích" bài hát, "nghe đi nghe" lại nhiều lần các bài hát, "bỏ qua" các bài hát không thích …. Dựa trên tập dữ liệu đó ta có thể áp dụng NBC để tính toán ra các phong cách nhạc mà người dùng thích nhất, từ đó chúng ta có thể đưa ra các "gợi ý" nghe nhạc gần đúng nhất cho người dùng từ việc học hỏi từ những thói quen đó.

2.2. Định luật Bayes

Định lý Bayes cho phép tính xác suất xảy ra của một sự kiện ngẫu nhiên A khi biết sự kiện liên quan B đã xảy ra. Xác suất này được ký hiệu là P(A|B), và đọc là "xác suất của A nếu có B". Đại lượng này được gọi xác suất có điều kiện hay xác suất hậu nghiệm vì nó được rút ra từ giá trị được cho của B hoặc phụ thuộc vào giá trị đó. Theo định lí Bayes, xác suất xảy ra A khi biết B sẽ phụ thuộc vào 3 yếu tố:

- Xác suất xảy ra A của riêng nó, không quan tâm đến B. Kí hiệu là P(A) và đọc là xác suất của A. Đây được gọi là xác suất biên duyên hay xác suất tiên nghiệm, nó là "tiên nghiệm" theo nghĩa rằng nó không quan tâm đến bất kỳ thông tin nào về B.

- Xác suất xảy ra B của riêng nó, không quan tâm đến A. Kí hiệu là P(B) và đọc là "xác suất của B". Đại lượng này còn gọi là hằng số chuẩn hóa (normalising constant), vì nó luôn giống nhau, không phụ thuộc vào sự kiện A đang muốn biết.
- Xác suất xảy ra B khi biết A xảy ra. Kí hiệu là P(B|A) và đọc là "xác suất của B nếu có A". Đại lượng này gọi là khả năng (likelihood) xảy ra B khi biết A đã xảy ra. Chú ý không nhầm lẫn giữa khả năng xảy ra B khi biết A và xác suất xảy ra A khi biết B.

Tóm lại định lý Bayes sẽ giúp ta tính ra xác suất xảy ra của một giả thuyết bằng cách thu thập các bằng chứng nhất quán hoặc không nhất quán với một giả thuyết nào đó. Khi các bằng chứng tích lũy, mức độ tin tưởng vào một giả thuyết thay đổi. Khi có đủ bằng chứng, mức độ tin tưởng này thường trở nên rất cao hoặc rất thấp, tức là xác xuất sảy ra giả thuyết sẽ thay đổi thì các bằng chứng liên quan đến nó thay đổi.

Công thức của định luật Bayes được phát biểu như sau:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (1)$$

- P(A|B) là xác suất xảy ra sự kiện A khi biết sự kiện liên quan B đã xảy ra.
- P(B|A) là xác suất xảy ra sự kiện B khi biết sự kiện liên quan Á đã xảy ra.
- P(A) là xác suất xảy ra của riêng A mà không quan tâm đến B.
- P(B) là xác suất xảy ra của riêng B mà không quan tâm đến A.

Ở trên ta có thể thấy xác suất sảy ra của giả thuyết A phụ thuộc và xác suất của giả thuyết B, nhưng trong thực tế xác suất A có thể phụ thuộc vào xác suất của nhiều các giác thuyết khác có thể là B1, B2, B3 … Bn.

2.3. Các loại phân phối xác suất

2.3.1. Gaussian Naïve Bayes (Phân phối chuẩn)

Mô hình này được sử dụng chủ yếu trong loại dữ liệu mà các thành phần là các biến liên tục.

- Giả định rằng các biến đầu vào có phân phối Gaussian (phân phối chuẩn).
- Được sử dụng khi giá trị của các biến đầu vào được đo lường dưới dạng số thực.

Định nghĩa 2.19 (Phân phối chuẩn). Biến ngẫu nhiên liên tục X được gọi là tuân theo luật phân phối chuẩn (normal distribution) với tham số μ, σ^2, ký hiệu là $X \sim \mathcal{N}(\mu, \sigma^2)$, nếu hàm mật độ xác suất của X có dạng

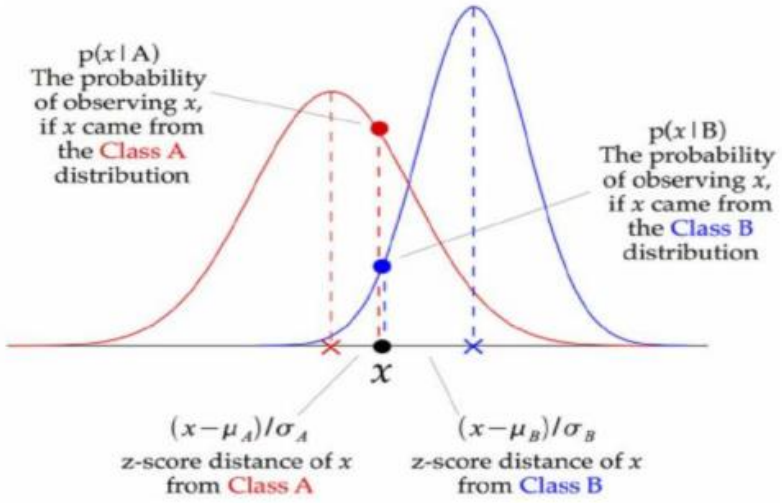
$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad x \in \mathbb{R} \tag{2.38}$$

ở đây e và π được lấy xấp xỉ lần lượt là 2,71828 và 3,14159.

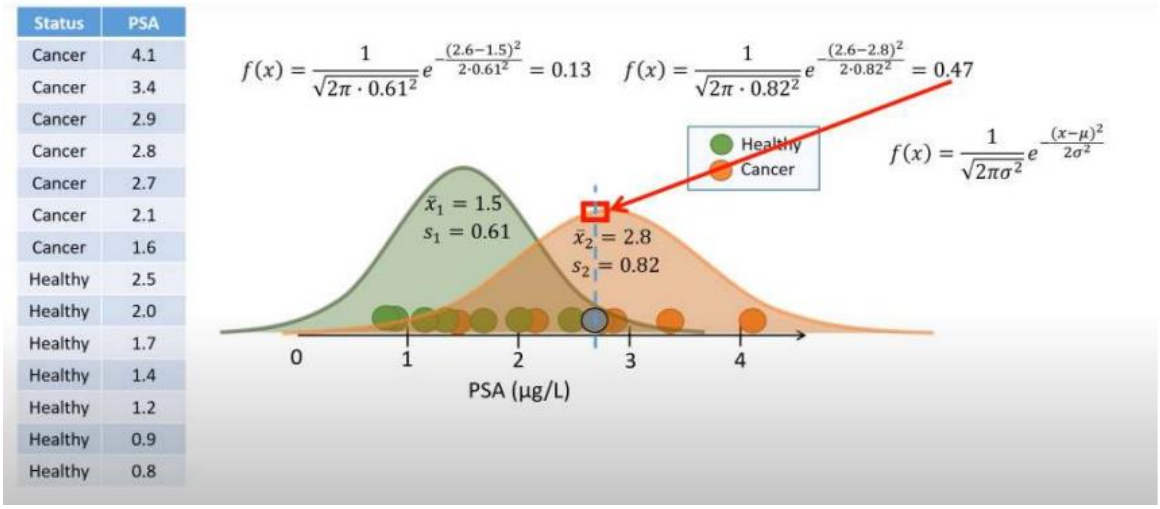
Hình 2.1: Định nghĩa phân phối chuẩn

Khi các đặc trưng nhận giá trị liên tục, ta giả sử các đặc trưng đó có phân phối Gaussian. Khi đó,likelihood(khả năng xảy ra) sẽ có dạng:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp\left(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right)$$

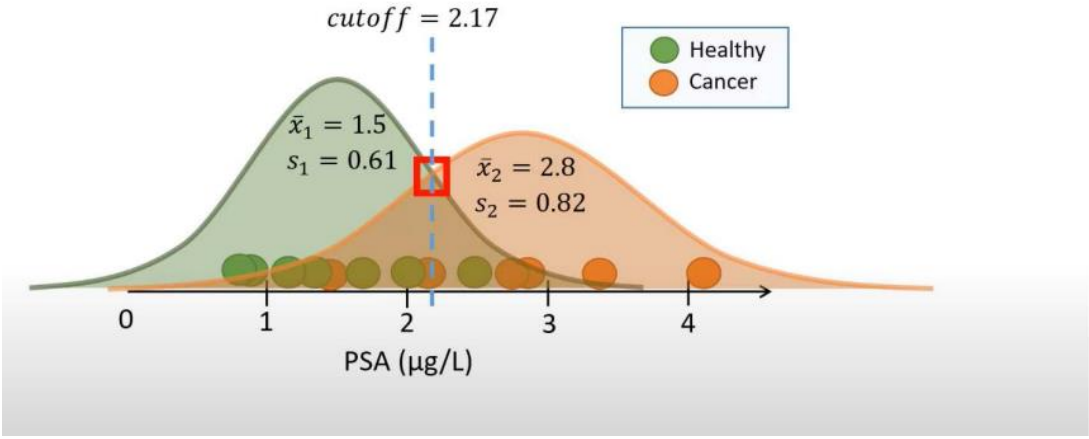


Hình 2.2: Cách hoạt động của bộ phân loại Gaussian Naive Bayes (GNB).



Hình 2.3: Ví dụ về phân loại theo Gaussian Naive Bayes

Xét ví dụ trên ta nhận thấy các điểm dữ liệu ban đầu (xanh, cam) liên tục, nếu có một điểm mới (xám) nằm trong phạm vi PSA=2.6 thì Naïve bayes có xu hướng dự đoán thuộc lớp Cancer. Tuy nhiên như vậy là không khả quan. Vì vậy ta sử dụng Gaussian Naïve Bayes. Sử dụng công thức phân phối chuẩn ta vẽ được đồ thị phân phối của 2 lớp dữ liệu. Qua đó tại điểm x = 2.6, thì x sẽ thuộc lớp có giá trị hàm phân phối cao hơn (Cancer).



Hình 2.4: Ví dụ về phân loại theo Gaussian Naive Bayes

Xét thấy có một điểm mà ở đó 2 đường cong có độ cao bằng nhau, ta gọi đó là điểm cutoff (trong bài này giá trị cutoff xấp xỉ bằng 2.17). Nếu những người có chỉ số PSA nhỏ hơn 2.17 người đó xếp ở lớp Healthy và ngược lại.

2.3.2. Bernoulli Naive Bayes

Mô hình này được áp dụng cho các loại dữ liệu mà biến đầu vào có giá trị nhị phân (0 hoặc 1), mô tả sự xuất hiện hoặc vắng mặt của một đặc trưng.

Thích hợp cho các tác vụ phân loại nhị phân, nơi giá trị của biến đầu vào chỉ là 0 hoặc 1.

- Lược đồ Bernoulli:

1. Giả sử ta tiến hành n phép thử độc lập.
2. Trong mỗi phép thử chỉ có hai trường hợp: hoặc sự kiện A xảy ra, hoặc sự kiện A không xảy ra (tức là xảy ra A).
3. Xác suất xảy ra A trong mỗi phép thử đều bằng p (tức là P(A) = p) và xác suất không xảy ra A trong mỗi phép thử đều bằng q = 1 − p (tức là P(A) = 1 − p).

   => Những bài toán thỏa mãn cả ba điều kiện trên được gọi là tuân theo lược đồ Béc-nu-li (hay dãy phép thử Béc-nu-li).

- Công thức Bernoulli:

Định lý 1.1. Trong lược đồ Béc–nu–li (hay dãy phép thử Béc–nu–li)

(a) Xác suất để sự kiện A xuất hiện đúng k lần, ký hiệu là $P_n(k)$, được xác định bởi

$$P_n(k) = C_n^k p^k (1-p)^{n-k}, \quad k = 0, 1, \ldots, n \tag{1.19}$$

(b) Xác suất để sự kiện A xuất hiện từ k_1 đến k_2 lần, ký hiệu là $P_n(k_1, k_2)$:

$$P_n(k_1, k_2) = \sum_{k=k_1}^{k_2} P_n(k) = \sum_{k=k_1}^{k_2} C_n^k p^k (1-p)^{n-k} \tag{1.20}$$

Hình 2. 5: Công thức Bernoulli

- Phân phối Bernoulli: Như đã nói ở trên phép thử Bernoulli là phép thử chỉ cho 2 kết quả duy nhất là A với xác xuất p và với xác suất q = 1 – p. Biến ngẫu nhiên X tuân theo phân phối Bernoulli như sau:

$$p(x) = P[X = x] = \begin{cases} q = 1 - p & x = 0 \\ p & x = 1 \end{cases} \tag{2}$$

- Quy tắc cho Bernoulli Naïve Bayes:

$$P(x_i \mid y) = P(i \mid y)x_i + (1 - P(i \mid y))(1 - x_i) \quad (3)$$

2.3.3. Multinomial Naive Bayes

Multinomial Naive Bayes là một phân phối xác suất đa thức, được sử dụng để mô hình hóa xác suất của các sự kiện rời rạc.

Thường được sử dụng cho các biến đầu vào mà mô tả tần suất xuất hiện của một từ (hoặc một đặc trưng) trong một tài liệu.

Phù hợp cho các biến đầu vào có giá trị là số nguyên dương, thường là đếm số lần xuất hiện của từng từ trong văn bản.

- Phân phối đa thức: Giả sử có n phép thử độc lập, mỗi phép thử cho kết quả là một trong số k nhóm với mỗi nhóm có xác suất tương ứng xác định.

Giả sử p_i, for $i = \overline{1,k}$ là xác suất rơi vào nhóm i tương ứng trong k nhóm, ta có:

$$\sum_{i=1}^{k} p_i = 1$$

Nếu biến ngẫu nhiên $X_i \in \{0, 1, \ldots, n\}$, for $i = \overline{1,k}$ thể hiện số lần xuất hiện của sự kiện nhóm i, ta có:

$$\sum_{i=1}^{k} x_i = n$$

Ta có công thức tính xác suất cho phân phối đa thức:

- Áp dụng Multinomial Naive Bayes cho bài toán phân loại tài liệu văn bản:

	Doc	Word	Class
Tranning	1	Chinese Bejing Chinese	c
	2	Chinese Chinese Shanghai	c
	3	Chinese Macao	c
	4	Tokyo Japan Chinese	j
Test	5	Chinese Chinese Chinese Tokyo Japan	?

Hình 2.6: Ví dụ áp dụng Multinomial Naive Bayes

Có 5 văn bản trong đó 4 văn bản được phân loại vào 2 lớp c và j. Vậy văn bản thứ 5 sẽ thuộc class nào?

Conditional Probabilities:

P(Chinese|*c*) = (5+1) / (8+6) = 6/14 = 3/7
P(Tokyo|*c*) = (0+1) / (8+6) = 1/14
P(Japan|*c*) = (0+1) / (8+6) = 1/14
P(Chinese|*j*) = (1+1) / (3+6) = 2/9
P(Tokyo|*j*) = (1+1) / (3+6) = 2/9
P(Japan|*j*) = (1+1) / (3+6) = 2/9

Cuối cùng chúng ta tính ra được xác xuất để chọn class:

- P(c|d5) = 3/4 * (3/7)^3 * 1/14 * 1/14 = 0.0003
- P(j|d5) = 1/4 * (2/9)^3 * 2/9 * 2/9 = 0.0001

Vậy là chúng ta chọn loại c cho **Doc5**

Hình 2.7: Ví dụ áp dụng Multinomial Naive Bayes

2.4. Ví dụ thực tế

Trong một vụ thu hoạch ở một đồn điền trang trại các người làm đã thu hoạch được hơn 1000 trái cây các loại được phân loại thành 3 nhóm trái cây chính là "Chuối (banana)", "Cam (orange)" và "các loại trái cây khác (other fruit)" và được phân l oại thành các kiểu như loại trái cây "dài (long), "không dài (not long), "ngọt (sweet)", "không ngọt (not sweet)", "màu vàng (yellow)", "không phải màu vàng (not yellow)":

Type	Long	Not Long	Sweet	Not Sweet	Yellow	Not Yellow	Total
Banana	400	100	350	150	450	50	500
Orange	0	300	150	150	300	0	300
Other Fruit	100	100	150	50	50	150	200
Total	500	500	650	350	800	200	1000

Hình 2.8: Ví dụ thực tế

Bây giờ bài toàn đặt ra là tính ra tỷ lệ một quả chuối có thuộc tính là "màu vàng, dài, và ngọt" với tỷ lệ quả cảm và các loại hoa quả khác có cũng có thuộc tính là "màu vàng, dài, và ngọt".

Áp dụng định lý Bayes ta sẽ có 3 công thức tính cho 3 loại trái cây như sau:

Tỷ lệ quả chuối với thuộc tính "vàng, dài và ngọt":

P(Long|Banana) = 400/500 = 0.8

P(Sweet|Banana) = 350/500= 0.7

P(Yellow|Banana) =450/500= 0.9

P(Banana) = 500/1000 = 0.5

P(Long) = 500/1000 = 0.5

P(Sweet) = 650/1000 = 0.65

P(Yellow) = 800/1000= 0.8

P(Banana|Long,Sweet,Yellow) = (0.8 * 0.7 * 0.9 * 0.5) / (0.5 * 0.65 * 0.8) = 0.97

Tức là tỷ lệ chuối với thuộc tính "vàng, dài và ngọt" là 97%

Tương tự ta cũng có thể tính ra tỷ lệ quả cam với thuộc tính "vàng dài và ngọt" với công thức sau:

Do tỷ lệ P(Long|Orange) = 0/500 = 0 cho nên P(Orange|Long, Sweet, Yellow) = 0 tức là tỷ lệ quả cam với thuộc tính "vàng dài và ngọt" là 0%.

Cũng thế ta ốp công thức Bayes để tính các trái cây còn lại với thuộc tính "vàng dài và ngọt" với công thức sau:

P(Long|Other Fruit) = 100/200= 0.5

P(Sweet|Other Fruit) = 150/200 = 0.75

P(Yellow|Other Fruit) =50/200= 0.25, P(Other Fruit) = 200/1000 = 0.2

P(Banana|Long,Sweet,Yellow) = 0.5 * 0.75 * 0.25 * 0.2 / (0.5 * 0.65 * 0.8) = 0.072

Tức là tỷ lệ các trái cây khác có thuộc tính "vàng dài và ngọt" chỉ là khoảng 7,2%

=> Vậy suy ra với trái cây với ba thuộc tính là "Vàng, dài và ngọt" thì có khả năng cao nhất đó là quả chuối.

Chúng ta có thể ứng dụng Naive Bayes Classification để tính tỷ lệ xác suất với rất nhiều các dạng bài toán khác nhau, với dữ liệu càng nhiều thì độ chính xác của thuật toán sẽ càng cao, và khi dữ liệu thay đổi thì kết quả cũng thay đổi theo.

2.5. Ứng dụng của thuật toán

- *Real time Prediction:* NBC chạy khá nhanh nên nó thích hợp áp dụng ứng dụng nhiều vào các ứng dụng chạy thời gian thực, như hệ thống cảnh báo, các hệ thống trading…
- *Multi class Prediction:* Nhờ vào định lý Bayes mở rộng ta có thể ứng dụng vào các loại ứng dụng đa dự đoán, tức là ứng dụng có thể dự đoán nhiều giả thuyết mục tiêu.
- *Text classification/ Spam Filtering/ Sentiment Analysis:* NBC cũng rất thích hợp cho các hệ thống phân loại văn bản hay ngôn ngữ tự nhiên vì tính chính xác của nó lớn hơn các thuật toán khác. Ngoài ra các hệ thống chống thư rác cũng rất ưu chuộng thuật toán này. Và các hệ thống phân tích tâm lý thị trường cũng áp dụng NBC để tiến hành phân tích tâm lý người dùng ưu chuộng hay không ưu chuộng các loại sản phẩm nào từ việc phân tích các thói quen và hành động của khách hàng.
- *Recommendation System:* Naive Bayes Classifier và Collaborative Filtering được sử dụng rất nhiều để xây dựng cả hệ thống gợi ý, ví dụ như xuất hiện các quảng cáo mà người dùng đang quan tâm nhiều nhất từ việc học hỏi thói quen sử dụng internet của người dùng, hoặc như ví dụ đầu bài viết đưa ra gợi ý các bài hát tiếp theo mà có vẻ người dùng sẽ thích trong một ứng dụng nghe nhạc…

2.6. Ưu điểm và nhược điểm của Naive Bayes

***Ưu điểm**

- Giả định độc lập: hoạt động tốt cho nhiều bài toán/miền sữ liệu và ứng dụng.
- Đơn giản nhưng đủ tốt để giải quyết nhiều bài toán như phân lớp văn bản, lọc spam,..
- Cho phép kết hợp tri thức tiền nghiệm (prior knowledge) và dữ liệu quan sát được (obserwed data).
- Tốt khi có sự chệnh lệch số lượng giữa các lớp phân loại.
- Huấn luyện mô hình (ước lượng tham số) dễ và nhanh.

***Nhược điểm**

- 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 (đã nêu cách giải quyết ở phía trên)
- Mô hình không được huẩn luyện bằng phượng pháp tối ưu mạnh và chặt chẽ.
- Tham số mủa mô hình là các ước lượng xác suất điều kiện đơn lẻ.
- Không tính đến sự tương tác giữa các ước lượng này.

CHƯƠNG 3: CHƯƠNG TRÌNH

3.1. Thu thập và tiền xử lý dữ liệu

**** Chuẩn bị dữ liệu:***

Dữ liệu được thu thập từ các đánh giá sản phẩm của người dùng trên trang thương mại điện tử. Dữ liệu gồm có: 9826 mẫu đánh giá 1* đại diện cho đánh giá tiêu cực (negative) và 9727 mẫu đánh giá 5* đại diện cho đánh giá tích cực (positive).

**** Đọc file dữ liệu:***

```
import pandas as pd

# Đọc dữ liệu từ file CSV
df = pd.read_csv("/content/data_10k.csv")
```

```
# Nhóm số lượng dựa theo cột rating
df.groupby('rating').describe()
```

	comment		
	count	unique	top
rating			
1	9826	9823	
5	9727	9727	Hàng đẹp hơn mong đợi ,

```
df.head(10)
```

	rating	comment
0	1	Đóng gói cận thận, độ chống nhìn trộm t/bình, ...
1	1	sản phẩm chưa dùng đã hỏng, Sản phẩm không đún...
2	1	Sản phẩm giao chưa đúng theo yêu cầu, Shop khô...
3	1	trừ 1 sao vì đóng gói ẩu, chỉ lồng duy nhất tú...
4	1	*Hãy xem đánh giá 1★, Rất bức xúc😮Không đáng 1...
5	1	. Loa mới dùng 1 ngày là hông nghe tiếng nữa⛪...
6	1	.thiếu 1. Số dụng củ
7	1	?????? Nhầm hàng shop ơi cái này là cái gì :D ...
8	1	??shop đặt dây 20w về dây 27w
9	1	_QUÁ TỆ_ Khuyên thật các bạn đừng mua hàng sho...

Hình 3.1: Đọc file dữ liệu

**** Tiền xử lý dữ liệu:***

Từ hình 3.1, ta có thể thấy ở dữ liệu gốc có chứa các ký tự đặc biệt, chấm câu, hoặc ký tự. Những thứ này đều không ảnh hưởng đến nghĩa của đánh giá vì vậy ta cần loại bỏ chúng đi. Ngoài ra, ta cần chuyển đổi về văn bản dạng chữ thường để tránh sự phân biệt giữa chữ hoa và chữ thường.

```
[ ]  import re

     # Làm sạch văn bản
     def clean_text(text):

         #Loại bỏ email
         text = ' '.join([i for i in text.split() if '@' not in i])

         #Loại bỏ website
         text = re.sub('http[s]?://\S+', ' ', text)

         #Loại bỏ tất cả các dấu câu và ký tự không phải chữ cái và số.
         text = re.sub(r'[^\w\s]', ' ', text)

         #oại bỏ tất cả các khoảng trắng nhiều hơn 1 và thay thế chúng bằng một khoảng trắng duy nhất.
         text = re.sub('\s+', ' ', text)

         return text

     df["clean_comment"] = df.comment.apply(lambda x: clean_text(x))
```

Hình 3.2: Tiền xử lý dữ liệu

Văn bản sau khi tiền xử lý:

	rating	comment
0	1	Đóng gói cận thận độ chống nhìn trộm t bình Độ...
1	1	sản phẩm chưa dùng đã hỏng Sản phẩm không đúng...
2	1	Sản phẩm giao chưa đúng theo yêu cầu Shop khôn...
3	1	trừ 1 sao vì đóng gói ẩu chỉ lồng duy nhất túi...
4	1	Hãy xem đánh giá 1 Rất bức xúc Không đáng 1 s...
5	1	Loa mới dùng 1 ngày là hông nghe tiếng nữa Đú...
6	1	thiếu 1 Số dụng củ
7	1	Nhầm hàng shop ơi cái này là cái gì D mất 65k...
8	1	shop đặt dây 20w về dây 27w
9	1	_QUÁ TỆ_ Khuyên thật các bạn đừng mua hàng sho...

Hình 3.3: Văn bản sau khi tiền xử lý

	comment			
	count	unique	top	freq
rating				
negative	9672	9672	Đóng gói cận thận độ chống nhìn trộm t bình Độ...	1
positive	9672	9672	Hàng đẹp hơn mong đợi chất lượng tốt phù hợp v...	1

Hình 3.4: Văn bản sau khi tiền xử lý

Tokenization: Chia câu thành các từ (token) để dễ dàng xử lý.

Loại Bỏ Stop words: Loại bỏ các từ phổ biến như "và," "là," vì nó không ảnh hưởng nhiều đến nghĩa của đánh giá và để giảm chiều dài của vectơ đặc trưng.

```
from stop_words import get_stop_words
from underthesea import word_tokenize
from nltk.stem.porter import PorterStemmer

stopwords_vi = get_stop_words('vi')  # Danh sách stopwords tiếng Việt
list_word = []
ps = PorterStemmer()

for i in range(len(comment_df)):
        review = comment_df['comment'][i].lower()

        # Chuyển đổi thành danh sách từ sử dụng word_tokenize
        words = word_tokenize(review, format="text")

        # Tiếp tục với các bước xử lý tiếp theo nếu cần
        words = [ps.stem(word) for word in words.split() if word not in stopwords_vi]
        review = ' '.join(words)
        list_word.append(review)
```

Hình 3.5: Thực hiện thêm một số tiền xử lý

TF-IDF Vectorization: Sử dụng TF-IDF để chuyển đổi văn bản thành vectơ, biểu diễn tần suất xuất hiện của từng từ trong văn bản.

```
# Chuyển đổi dữ liệu văn bản sang dạng vector bằng TFIDF VECTORIZER
from sklearn.feature_extraction.text import TfidfVectorizer
cv=TfidfVectorizer(max_features=500)
x=cv.fit_transform(list_word).toarray()
```

```
y=df['rating']
```

Hình 3.6: Chuyển văn bản thành vectơ với TF-IDF

3.2. Triển khai mô hình

**** Chia tập dữ liệu và tạo mô hình huấn luyện:***

Chia tập dữ liệu thành 2 tập train và test lần lượt theo tỉ lệ 80% và 20%.

Lựa chọn mô hình Multinomial Naïve Bayes để huấn luyện vì mô hình này thích hợp cho dữ liệu rời rạc, như là vectơ đặc trưng của từng từ.

```
# Chia dữ liệu thành 2 tập train, test
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=0)
```

```
from sklearn.naive_bayes import MultinomialNB
clf=MultinomialNB()
clf.fit(x_train, y_train)
y_pred=clf.predict(x_test)
```

Hình 3.7: Chia tập dữ liệu và khởi tạo mô hình huấn luyện

3.3. Thử nghiệm và đánh giá kết quả

**** Kiểm tra độ chính xác của mô hình:***

Như hình ảnh dưới đây cho thấy độ chính xác của mô hình phân lớp MultinomialNB trên tập dữ liệu Test đạt được 92,4%.

```
[[1740  148]
 [ 145 1836]]
              precision    recall  f1-score   support

    negative       0.92      0.92      0.92      1888
    positive       0.93      0.93      0.93      1981

    accuracy                           0.92      3869
   macro avg       0.92      0.92      0.92      3869
weighted avg       0.92      0.92      0.92      3869

accuracy_score:  0.9242698371672267
```

Hình 3.8: Độ chính xác của mô hình huấn luyện

**** Thử nghiệm:***

```
from underthesea import word_tokenize
from sklearn.feature_extraction.text import TfidfVectorizer

# Hàm tiền xử lý cho một đoạn văn bản
def preprocess_text(text):
    # Thực hiện các bước tiền xử lý tương tự như trong quá trình huấn luyện
    words = word_tokenize(text, format="text")
    words = [ps.stem(word) for word in words.split() if word not in stopwords_vi]
    return ' '.join(words)

# Chuẩn bị dữ liệu đầu vào mới
# new_text = "áo xấu vải mỏng"
new_text = "rất tốt, Chất lượng , giao hàng nhanh , giá cả hợp lý , khuyên dùng"
preprocessed_text = preprocess_text(new_text)

# Sử dụng TfidfVectorizer để chuyển đổi đoạn văn bản thành vectơ TF-IDF
new_text_vector = cv.transform([preprocessed_text]).toarray()

# Dự đoán bằng mô hình đã huấn luyện
prediction = clf.predict(new_text_vector)

# In kết quả dự đoán
print("Đánh giá trên thuộc loại:", prediction)
```

```
Đánh giá trên thuộc loại: ['positive']
```

Hình 3.9: Thử nghiệm mô hình

Có thể thấy rằng Naïve Bayes là thuật toán tốt cho mô hình phân lớp văn bản với độ chính xác cao. Ngoài ra phương pháp trích chọn đặc trưng cũng ảnh hưởng rất lớn tới sự chính xác của mô hình.

KẾT LUẬN

Bài báo cáo đã cung cấp cho ta một cái nhìn rõ ràng hơn về việc sử dụng Multinomial Naive Bayes trong phân loại đánh giá. Kết quả thu được không chỉ cung cấp một cái nhìn chi tiết về hiệu suất của mô hình mà còn đặt ra những thách thức và hướng phát triển tiềm năng cho các nghiên cứu sau này. Điều này mở ra cơ hội cho những tiến bộ trong việc hiểu ý kiến và đánh giá trên mạng từ các nguồn đa dạng.

Kết quả thực nghiệm đã chứng minh rằng Multinomial Naive Bayes là một lựa chọn hiệu quả cho bài toán phân loại đánh giá. Đặc biệt, nó hoạt động tốt trên dữ liệu văn bản với các đặc trưng đếm, như số lần xuất hiện của từ trong văn bản.

Mặc dù Multinomial Naive Bayes mang lại hiệu suất đáng kể, nhưng còn nhiều thách thức, chẳng hạn như xử lý ngôn ngữ không chuẩn, đa dạng đánh giá, và các biểu cảm ngôn ngữ. Các nghiên cứu sau này có thể tập trung vào giải quyết những thách thức này và cải thiện mô hình.

TÀI LIỆU THAM KHẢO

[1] Thuật toán Naïve Bayes:

https://machinelearningcoban.com/2017/08/08/nbc/

[2] Tổng quan về thuật toán phân lớp Navie Bayes:

http://hoctructuyen123.net/tongquan-ve-thuat-toan-phan-lop-naive-bayes classification-nbc/

[3] Navie Bayes Classifiers:

https://www.geeksforgeeks.org/naive-bayes-classifiers/

[4] Tìm hiểu về thuật toán phân lớp Navie Bayes:

https://200lab.io/blog/tim-hieu-naive-bayes-classification-phan-1/

[5] Áp dụng thuật toán Multinomial Naïve Bayes trong xử lý ngôn ngữ tự nhiên:

https://medium.com/syncedreview/applying-multinomial-naive-bayes-to-nlp-problems-a-practical-explanation-4f5271768ebf?ref=200lab.io