TRƯỜNG ĐẠI HỌC THỦY LỢI

KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO BÀI TẬP LỚN MÔN HỌC

HỌC MÁY

ĐỀ TÀI: DỰ ĐOÁN MỨC ĐỘ HÀI LÒNG CỦA KHÁCH HÀNG VỀ CHẤT LƯỢNG DỊCH VỤ VẬN CHUYỂN DỰA TRÊN THUẬT TOÁN NAVIE BAYES

Giảng viên hướng dẫn: Đào Thị Thúy Quỳnh

Sinh viên thực hiện: Vũ Văn Chức

Phạm Thanh Hải

Hà Nội, 2022

MỤC LỤC

LỜI MỞ ĐẦU

Trong quá tình nghiên cứu đề tài: “Nghiên cứu và đưa ra dự đoán mức độ hài lòng của khách hàng về chất lượng dịch vụ vận chuyển bằng thuật toán Navie Bayes”, nhóm em vô cùng cảm ơn vì đã được cô Đào Thị Thúy Quỳnh tạo điều kiện thuận lợi, hướng dẫn tận tình và giúp đỡ trong suốt quá trình học tập và nghiên cứu.

Bài nghiên cứu chứa dữ liệu thứ cấp, được tham khảo từ các bài báo nghiên cứu của nhiều tác giả nhằm đưa ra những giải pháp giúp tăng độ hài lòng của khách hàng đối với dịch vụ vận chuyển của các nhà hàng.

Bài nghiên cứu vẫn còn những thiếu sót về mặt kiến thức chuyên môn do thời gian và nguồn dữ liệu còn bị hạn chế. Nhóm chúng em mong nhận được sự đóng góp ý kiến của cô và các bạn để đề tài được hoàn thiện hơn.

Chúng em xin chân thành cảm ơn!

I. TỔNG QUAN VỀ ĐỀ TÀI

1.1. Giới thiệu về bài toán

"Trải nghiệm khách hàng là tổng trải nghiệm của khách hàng đối với thương hiệu của bạn trên tất cả các điểm tiếp xúc trong hành trình của khách hàng; từ khám phá ban đầu cho đến trạng thái chuyển đổi."

Sự hài lòng của khách hàng cho thấy mức độ hài lòng của khách hàng đối với dịch vụ mà họ nhận được. Nhưng sự hài lòng là yếu tố chủ quan, vì nó liên quan đến cảm nhận hoặc trải nghiệm về sản phẩm, dịch vụ. Như vậy, công bằng mà nói, sự hài lòng của khách hàng có liên hệ mật thiết với trải nghiệm của khách hàng. Khách hàng sẵn sàng trả nhiều tiền hơn để nâng cấp trải nghiệm của mình. Như vậy, trải nghiệm tốt hơn và nhờ đó cải thiện sự hài lòng của khách hàng giúp nâng cao giá trị sản phẩm hoặc dịch vụ của các nhà hàng. Nhất là trong thời đại công nghệ 4.0 như hiện nay, việc mua đồ ăn thông qua các nền tảng online như Shopeefood, Grab, Gojek… dần trở nên phổ biến. Vì vậy, việc nghiên cứu và đánh giá mức độ hài lòng của khách hàng là ưu tiên hàng đầu của nhà hàng cũng như của bên giao vận nhằm không chỉ cải thiện chất lượng sản phẩm mà còn nâng cao chất lượng dịch vụ để có thể thu hút nhiều lượng khách hàng mới và giữ chân được những khách hàng quen thuộc.

Từ bài toán thực tế trên, nhóm chúng em quyết định nghiên cứu và đưa ra hướng xử lý cho bài toán dự đoán mức độ hài lòng của khách hàng dựa trên các yếu tố đặc trưng như chất lượng đồ ăn, giá thành để đánh giá về chất lượng dịch vụ vận chuyển bằng thuật toán Navie Bayes. Từ đó để hiểu rõ hơn về thuật toán cũng như đưa ra cái nhìn tổng quát về ứng dụng thực tế của thuật toán trong cuộc sống.

1.2. Giới thiệu thuật toán Navie Bayes

1.2.1. Khái niệm

Naive Bayes Classification (NBC) – thuật toán phân loại Naive Bayes - 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, được ứng dụng rất nhiều trong các lĩnh vực Machine learning dùng để đưa các dự đoán có độ chính xác cao, dựa trên một tập dữ liệu đã được thu thập. NBC thuộc vào nhóm học máy có giám sát.

Có ba loại Mô hình Naive Bayes, được đưa ra dưới đây:

- ***Gaussian***: Mô hình Gaussian giả định rằng các tính năng tuân theo phân phối chuẩn. Điều này có nghĩa là nếu các yếu tố dự đoán nhận các giá trị liên tục thay vì rời rạc, thì mô hình sẽ giả định rằng các giá trị này được lấy mẫu từ phân phối Gaussian.
- ***Đa thức***: Trình phân loại Naïve Bayes Đa thức được sử dụng khi dữ liệu được phân phối đa thức. Nó chủ yếu được sử dụng cho các vấn đề phân loại tài liệu, nó có nghĩa là một tài liệu cụ thể thuộc về danh mục nào, chẳng hạn như Thể thao, Chính trị, giáo dục, v.v. Bộ phân loại sử dụng tần suất của các từ cho các yếu tố dự đoán.
- ***Bernoulli:*** Bộ phân loại Bernoulli hoạt động tương tự như bộ phân loại Đa thức, nhưng các biến dự đoán là các biến Booleans độc lập. Chẳng hạn như nếu một

từ cụ thể có mặt hoặc không có trong tài liệu. Mô hình này cũng nổi tiếng với nhiệm vụ phân loại tài liệu.

1.2.2 Ứng dụng trong thực tế

Thuật toán Naive Bayes Classification được áp dụng vào các loại ứng dụng sau:
- 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 phát hiện sự cố...
- 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 dữ liệu. 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 được sử dụng rất nhiều để xây dựng hệ thống gợi ý dựa trên thói quen của người dùng.

1.3. Các bước xử lý cho bài toán

Các bước xử lý bài toán được tiến hành theo mô hình dưới đây:

Bước 1: Download dữ liệu từ trang https://www.kaggle.com/datasets/mohamedharris/restaurant-order-details

Bước 2: Xử lý dữ liệu để dữ liệu ở dạng chuẩn hóa (làm sạch dữ liệu)

Bước 3: Training mô hình và dự đoán với thư viện sklearn Naive Bayes

Bước 4: Mô hình đưa ra kết quả dự đoán

Bước 5: Đánh giá kết quả thực nghiệm

II. THUẬT TOÁN PHÂN LOẠI NAVIE BAYES

2.1. Giải thuật thuật toán phân loại Navie Bayes

Trong giai đoạn huấn luyện ta có mợt tập mẫu, mỗi mẫu đươc cho bởi cặp $<x_i, y_i>$ trong đó:

- x_i *là vector đặc trưng(thuộc tính)*
- y_i *là nhãn phân loại,* $y_i \in C$*(C là các tập nhãn)*

Sau khi huấn luyện xong, bộ phận phân loại cần dự đoán nhãn y cho mẫu mới $x=<x1,x2,\ldots x_n>$

$$y = \text{argmax}_{cj \in C}\ P(c_j \mid x1,x2,\ldots,x_n)$$

- Sử dụng quy tắc Bayes

$$y = \text{argmax}_{cj \in C} \frac{P(x1,x2,\ldots,x_n \mid c_j)P(c_j)}{P(c_j \mid x1,x2,\ldots,xn)} = \text{argmax}_{cj \in C}\ P(x1,x2,\ldots,x_n \mid c_j)P(c_j)$$

trong đó:

- $P(c_j)$ là tần suất quan sát thấy nhãn c_j trên tập dữ liệu D $\frac{count(cj)}{|D|}$
- *$P(x_i|c_j)$ là số lần xuất hiện x_i cùng với c_j chia cho số lần xuất hiện c_j* $\frac{count(xi,cj)}{count(cj)}$

$$P(x1,x2,...,x_n|c_j) = P(x_1|c_j)P(x_2|c_j) \dots P(x_n|c_j)$$

Khắc phục vấn đề xác suất điều kiện bằng zero:

- Nếu trong dữ liệu huấn luyện không có đối tượng X nào có thuộc tính lớp C khi đó xác suất điều kiện $P(x_i|c_j)$ sẽ bằng 0.
- Khi phân lớp, nếu có một đối tượng nào mang thuộc tính này thì xác suất phân vào lớp C luôn bằng 0.
- Khắc phục bằng cách ước lượng theo công thức Laplace coretion:

$$\frac{count(xi,cj)+1}{count(cj)+mj}$$

trong đó: *m_j là số lượng các thuộc tính của lớp C tương ứng.*

2.2. Đặc điểm

2.2.1 Ưu điểm

- Giả định độc lập: hoạt động tốt cho nhiều bài toán/miền dữ 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.

2.2.2 Nhược điểm

- Có thể xảy ra vấn đề zero nếu như dữ liệu đầu vào quá ít (đã nêu cách giải quyết ở phía trên)
- 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)

III.CHƯƠNG TRÌNH DỰ ĐOÁN MỨC ĐỘ HÀI LÒNG CỦA KHÁCH HÀNG VỀ CHẤT LƯỢNG DỊCH VỤ VẬN CHUYỂN DỰA TRÊN THUẬT TOÁN NAVIE BAYES

3.1.Dữ liệu thực nghiệm

Bộ dữ liệu training bao gồm 5 nhãn như sau:

- Quantity of Items: Số lượng món ăn
- Order amount: Số lượng đơn hàng
- Delivery Time Taken: Thời gian vận chuyển(phút)
- Customer Rating-Food: Đánh giá về chất lượng đồ ăn
- Customer Rating-Delivery: Đánh giá chất lượng dịch vụ vận chuyển

3.2. Các bước xử lý bài toán (chi tiết)

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import category_encoders as ce
from sklearn.preprocessing import RobustScaler
from sklearn.naive_bayes import GaussianNB
import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

import warnings
warnings.filterwarnings('ignore')

# B1: đọc file dữ liệu
data = 'Orders.csv'
df = pd.read_csv(data, header=None, sep=',')
col_names = ['Quantity of Items', 'Order Amount', 'Delivery Time Taken (mins)',
'Customer Rating-Food', 'Customer Rating-Delivery']
df.columns = col_names

# Khai báo vector đặc trưng và biến mục tiêu
X = df.drop(['Customer Rating-Delivery'], axis=1)
y = df['Customer Rating-Delivery']

# B2: Tách dữ liệu thành tập huấn luyện và kiểm tra riêng biệt
# Chia X và y thành các tập huấn luyện và kiểm tra
from sklearn.model_selection import train_test_split

# lấy ra 10% tập dữ liệu để làm dữ liệu test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.1,
random_state = 0)
```

```
print(X_test)
# hiển thị các biến phân loại
categorical = [col for col in X_train.columns if X_train[col].dtypes == 'O']
numerical = [col for col in X_train.columns if X_train[col].dtypes != 'O']

# Mã hóa các biến còn lại với mã hóa một lần mã hoá
encoder = ce.OneHotEncoder(cols=['Quantity of Items', 'Order Amount', 'Delivery
Time Taken (mins)', 'Customer Rating-Food'])
X_train = encoder.fit_transform(X_train)
X_test = encoder.transform(X_test)

cols = X_train.columns
scaler = RobustScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
X_train = pd.DataFrame(X_train, columns=[cols])
X_test = pd.DataFrame(X_test, columns=[cols])

# B3: Training
# Huấn luyện bộ phân loại Gaussian Naive Bayes trên tập huấn luyện
# Khởi tạo mô hình
gnb = GaussianNB()

# Đổ dữ liệu
gnb.fit(X_train, y_train)

# Dự đoán kết quả
y_pred = gnb.predict(X_test)
print(y_pred)

# Điểm chính xác của mô hình
from sklearn.metrics import accuracy_score
print('Model accuracy score: {0:0.4f}'. format(accuracy_score(y_test, y_pred)))

# Điểm chính xác của tập huấn luyện
y_pred_train = gnb.predict(X_train)
print('Training-set accuracy score: {0:0.4f}'. format(accuracy_score(y_train,
y_pred_train)))
```

IV. KẾT QUẢ THỰC NGHIỆM VÀ ĐÁNH GIÁ

Bộ data x_test máy đã tách :

	Quantity of Items	Order Amount	Delivery Time Taken (mins)	Customer Rating-Food
90	5	710	45	4
254	7	815	24	4
284	5	377	24	3
446	3	634	11	4
339	3	590	31	1
15	7	872	44	5
407	4	573	16	3
278	3	352	30	3
159	5	616	41	2
153	7	719	21	3
241	5	370	48	2
250	4	224	22	2
306	3	639	49	2
439	3	770	27	5
171	4	353	44	3
297	1	65	23	2
469	5	542	39	5
402	6	971	20	2
154	7	772	49	5
37	4	474	19	2
205	7	691	47	5
367	4	787	44	1
240	5	568	48	3
108	5	830	49	3
45	4	478	15	2
421	4	751	30	3
21	6	686	35	3
425	5	546	11	3
96	4	300	27	2
233	6	945	36	4
434	7	837	50	1
118	4	391	20	2
124	4	376	46	3
191	5	691	41	2
375	6	1011	27	3
360	4	777	13	3
312	7	872	43	2
450	7	882	25	3
295	5	372	40	4
238	2	86	23	3
318	4	782	48	1

```
46                  7          744                  45                  3
470                 4          476                  37                  5
221                 5          844                  15                  5
76                  7          934                  30                  2
1                   5          633                  47                  5
213                 1           36                  10                  2
326                 4          508                  23                  3
419                 4          733                  12                  3
102                 2           78                  32                  5
364                 6         1160                  31                  3
```

```
#B4: Dự đoán kết quả
y_pred = gnb.predict(X_test)
print(y_pred)
```

```
[51 rows x 4 columns]
['3' '1' '1' '4' '5' '3' '5' '4' '3' '3' '5' '3' '4' '3' '3' '3' '2' '5'
 '3' '2' '3' '3' '5' '3' '3' '3' '3' '3' '3' '3' '3' '5' '3' '3' '3' '3'
 '5' '1' '2' '2' '5' '3' '5' '5' '3' '4' '5' '3' '3' '5' '5']
```

Đánh giá mức độ chính xác của kết quả dự đoán :

```
#B5: Đánh giá kết quả thực nghiệm
# Điểm chính xác của mô hình
from sklearn.metrics import accuracy_score
print('Model accuracy score: {0:0.4f}'. format(accuracy_score(y_test, y_pred)))

# Điểm chính xác của tập huấn luyện
y_pred_train = gnb.predict(X_train)
print('Training-set accuracy score: {0:0.4f}'. format(accuracy_score(y_train,
y_pred_train)))
```

```
Model accuracy score: 0.1765
Training-set accuracy score: 0.8733
```

V.KẾT LUẬN

- Thuật toán Naive Bayes dựa trên cơ sở định lí Bayes và đặc biệt phù hợp cho các trường hợp phân loại có kích thước đầu vào là lớn.
- Cơ sở của thuật toán Naïve Bayes là dựa vào định lý Bayes. Định lý Bayes là kết quả của lý thuyết xác suất, nó đề cập đến xác suất có điều kiện của biến ngẫu nhiên A biết phân bố xác suất của A và phân bố xác suất của B khi A đã xảy ra.
- Phân loại văn bản bằng định lý Bayes là phương pháp phân loại có giám sát, đây là phương pháp quan trọng trong xử lý ngôn ngữ tự nhiên. Nó dễ sử dụng, dễ cài đặt nhưng đem lại hiệu quả vô cùng cao.
- Chúng ta có các thuật toán thay thế như Suppory Vector Machine (SVM) hay Neural Networks khi gặp các vấn đề liên quan đến xử lý ngôn ngữ tự nhiên(Natural Language Processing). Khi đặt NBC (Naive Bayes Classifier) bên cạnh các thuật toán phức tạp, đòi hỏi nhiều thời gian và nguồn lực thì chúng lại trở nên lợi thế với thiết kế đơn giản cho các bài toán phân loại. Hơn nữa chúng còn được nhận xét là nhanh, đáng tin cậy và chính xác trong một số trường hợp của NLP.

TÀI LIỆU THAM KHẢO

https://www.kaggle.com/code/prashant111/naive-bayes-classifier-in-python

https://www.datacamp.com/tutorial/naive-bayes-scikit-learn

https://jakevdp.github.io/PythonDataScienceHandbook/05.05-naive-bayes.html

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