



**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

KHOA KHOA HỌC MÁY TÍNH



BÁO CÁO ĐỒ ÁN MÔN MÁY HỌC

Phân loại đánh giá trong lĩnh vực khách sạn

Lớp: **CS114.K21**

Giảng viên: **Lê Đình Duy**

Phạm Nguyễn Trường An

Nhóm thực hiện: **LoViTa**

1. Nguyễn Duy Nhật	18520118
2. Đỗ Nguyễn Thuận Phong	18520126
3. Nguyễn Dương Trúc Phương	18520133

MỤC LỤC

HỢP ĐỒNG THÀNH LẬP NHÓM.....	3
BẢNG PHÂN CÔNG CÔNG VIỆC.....	6
I. Mục đích.....	6
II. Bảng phân công công việc cụ thể	7
BẢNG ĐÁNH GIÁ CÔNG VIỆC CỤ THỂ	7
THÔNG TIN ĐỒ ÁN.....	8
I.Giới thiệu chung:	8
1.1 Đặt vấn đề:.....	8
1.2 Giới thiệu bài toán :	8
1.3 Ứng dụng của chương trình:.....	8
II. Bài toán:	8
2.1 Dữ liệu trong bài toán:	8
2.1.1 Thu thập dữ liệu:.....	8
2.1.2 Tiến hành gán nhãn:.....	9
2.1.3 Nhận xét về dữ liệu:.....	10
2.2 Feature Engineering:	11
a) Tách từ theo dấu cách:	11
b) Tách từ theo nghĩa của từ (sử dụng Pyvi):	11
c) CountVectorizer:	12
d) TF-IDF:	12
2.3 Model:	13
a) BernoulliNB:	13
b) LogisticRegression:	13
c) SVC:.....	13
d) Decision tree:	14
e) Random Forest:	14
2.4 Nhận xét:	14
III. Demo:	16
KẾT LUẬN	18
I. Hạn chế:	18
II. Hướng phát triển:	18

HỢP ĐỒNG THÀNH LẬP NHÓM

Tên nhóm: **LoViTa**

Danh sách thành viên:

STT	Tên thành viên	MSSV	Chức vụ
1	Nguyễn Duy Nhật	18520118	Thành viên
2	Đỗ Nguyễn Thuận Phong	18520126	Thành viên
3	Nguyễn Dương Trúc Phương	18520133	Nhóm trưởng

Mục đích thành lập:

- Tìm hiểu, hiểu biết về máy học.
- Nâng cao kỹ năng làm việc nhóm, thuyết trình.
- Thúc đẩy khả năng tìm tòi hiểu biết.
- Hoàn thành tốt các nhiệm vụ (đề án) mà khoa học đề ra.

Quy tắc làm việc đúng:

- Tham gia ít nhất 80% các buổi họp của nhóm.
- Thống nhất giờ giấc, ý thức đúng giờ.
- Tích cực tham gia bàn luận, đóng góp ý kiến cá nhân - Tôn trọng mọi người.
- Biết nhận lỗi, sửa lỗi, lắng nghe góp ý của mọi người.
- Có tinh thần trách nhiệm cao với công việc.
- Kết quả của việc bàn luận phải được sự đáp ứng của 2/3 thành viên.

Quy tắc làm việc sai:

- Nếu trễ họp 30 phút sẽ bị khiển trách và trừ điểm. Lần 2 sẽ bị loại ra khỏi buổi họp và đánh vắng buổi đó.
- Nếu không hoàn thành công việc được giao sẽ bị loại khỏi nhóm.
- Nếu công việc không hoàn thành đúng thời hạn đã giao sẽ bị trừ điểm.
- Nghỉ họp không có lý do, không thông báo trước.

Hình thức họp nhóm:

- Họp nhóm, trao đổi thông tin qua mạng (Facebook, Gmail, số điện thoại).
- Họp nhóm tại nơi thích hợp, có mặt đầy đủ của các thành viên: Phòng tự học ký túc xá, thư viện trường.

Vai trò các thành viên trong nhóm:

Thành viên	Lãnh đạo và phân công công việc	Tìm kiếm tài liệu	Thiết kế báo cáo	Thuyết trình
Nguyễn Duy Nhật		X	X	X
Đỗ Nguyễn Thuận Phong		X	X	X
Nguyễn Dương Trúc Phương	X	X	X	X

Tiêu chuẩn đánh giá hiệu quả hoạt động nhóm:

Đặc điểm	Tỷ trọng	Xuất sắc	Tốt	Bình Thường	Kém
Thái độ làm việc	30%	Nhiệt tình trong công việc, giúp đỡ quan tâm mọi người	Đề cao tinh thần trách nhiệm công việc, hoàn thành đúng hạn	Làm đủ việc được giao	Không có ý thức làm việc, trễ- thiếu thành phẩm
Quản lý thời gian	10%	Luôn hoàn thành công việc trước hạn và tới sớm chuẩn bị cho các cuộc họp nhóm	Luôn đúng giờ trong công việc và họp mặt nhóm	Hoàn thành nhiệm vụ đúng hạn với sự nhắc nhở	Không hoàn thành nhiệm vụ được giao và thường tới trễ các buổi họp
Giải quyết vấn đề	30%	Tích cực tìm kiếm, bàn luận xử lý vấn đề tối ưu	Tham khảo ý kiến, hỏi han giúp đỡ cách giải quyết vấn đề phát sinh	Đóng góp các ý kiến có thể giúp đỡ giải quyết các vấn đề đưa ra	Không tham gia vào việc góp ý – giải quyết các vấn đề phát sinh

Góp ý	20%	Sẵn sàng nêu ra ý kiến cá nhân, thảo luận và đánh giá cùng mọi người	Tự tin nêu ý kiến của mình	Phải đợi nhắc nhở mới góp ý	Không tham gia vào việc thảo luận
Giữ liên lạc	10%	Mọi người luôn luôn có thể liên lạc	Có 1 cách liên lạc nhất định	Liên lạc không ổn định nhưng biết chủ động liên lạc lại	Không thể liên lạc

Tiêu chí đánh giá thành viên cuối khóa học:

Dựa vào tỷ trọng trong bảng tiêu chuẩn đánh giá hiệu quả hoạt động nhóm, ta sẽ đánh giá từng thành viên theo thang điểm như sau:

Điểm 10: làm tốt việc được giao, đúng hạn, có chất lượng, giúp đỡ thành viên khác, tích cực, chủ động trong công việc.

Điểm 8-9: làm tốt việc được giao, đúng hạn, có chất lượng, giúp đỡ thành viên khác.

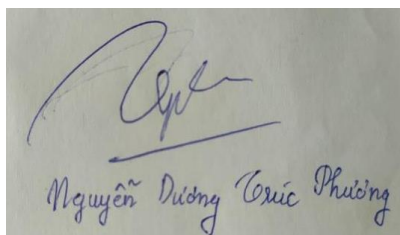
Điểm 6-7: Hoàn thành công việc được giao, kết quả chấp nhận được, vi phạm một số điều lệ nhóm.

Điểm 1-5: Chưa hoàn thành công việc được giao, ít hợp tác.

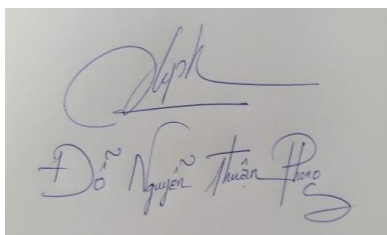
Điểm 0: Bị loại ra khỏi nhóm.

Mọi thành viên trong nhóm đều đồng ý các quy định trên và chấp hành những quy định của nhóm nêu trên.

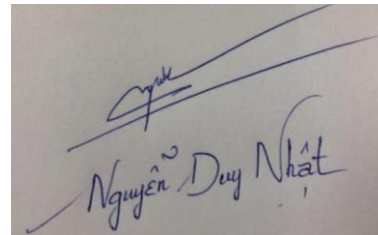
Nguyễn Dương Trúc Phương



Đỗ Nguyễn Thuận Phong



Nguyễn Duy Nhật



BẢNG PHÂN CÔNG CÔNG VIỆC

I. Mục đích

Hoàn thành những công việc của cả nhóm, tìm ra những vấn đề cần thiết để hoàn thành mục tiêu.

Nhằm phân bổ thời gian cần thiết cho từng cá nhân và cả đội. Bắt buộc hoàn thành theo đúng thời hạn deadline

Nội dung công việc và yêu cầu thực hiện

STT	Nội dung công việc	Yêu cầu cần đạt
1	Ý tưởng	Cùng nhau tìm hiểu về các mô hình máy học và các dự án có sẵn và đưa ra những ý tưởng hay. Tìm được hướng đi đúng cho đề án.
2	Hiện thực hóa chương trình	Xây dựng một chương trình.
3	Thuật giải	Sử dụng các mô hình máy học được học và tìm hiểu thêm.
4	Kiểm tra, chạy thử chương trình.	Khi phát hiện lỗi trong kiểm tra và chạy thử nghiệm phải xử lý được. Cần chạy thử nhiều lần và thử nhiều trường hợp khác nhau để soát lỗi nhiều nhất có thể
5	Cải thiện chương trình	Nâng cao thuật toán. Giảm thời gian chạy thuật giải. Đưa ra những kết quả với độ chính xác cao hơn.
6	Viết báo cáo, quay video thực hiện chương trình và thuyết trình.	Báo cáo phải bám sát yêu cầu mà giảng viên đã đề ra. Câu cú gọn gàng, không dài dòng nhưng đầy đủ chi tiết, trung thực với những gì nhóm đã làm được trong thời gian qua. Thuyết trình cần nêu rõ những gì mình đã làm được, trung thực trong từng câu nói mình nói ra.

II. Bảng phân công công việc cụ thể

STT	Họ tên	Chức vụ	Công việc cụ thể được phân công
1	Nguyễn Duy Nhật	Thành viên	Xây dựng ý tưởng. Kiểm tra, chạy thử chương trình. Thuyết trình. Viết báo cáo. Soạn powerpoint. Xây dựng giao diện.
2	Đỗ Nguyễn Thuận Phong	Thành viên	Xây dựng ý tưởng. Viết chương trình và xây dựng giao diện. Kiểm tra, chạy thử chương trình. Thuyết trình. Viết báo cáo
3	Nguyễn Dương Trúc Phương	Nhóm trưởng	Xây dựng ý tưởng. Kiểm tra, chạy thử chương trình. Viết báo cáo. Thuyết trình. Soạn powerpoint. Xây dựng giao diện.

BẢNG ĐÁNH GIÁ CÔNG VIỆC CỤ THỂ

STT	Họ và tên thành viên	Công việc được giao	Mức độ hoàn thiện
1	Nguyễn Duy Nhật	Xây dựng ý tưởng	Hoàn thành
		Kiểm tra, chạy thử chương trình.	Test, tìm và sửa lỗi trong quá trình chạy.
		Thuyết trình.	Hoàn thành.
		Mức độ đóng góp: 100%	
2	Đỗ Nguyễn Thuận Phong	Xây dựng ý tưởng	Hoàn thành
		Viết chương trình	Hoàn thành
		Kiểm tra, chạy thử chương trình	Sửa lỗi kịp thời
		Thuyết trình.	Hoàn thành
		Mức độ đóng góp: 100%	
3	Nguyễn Dương Trúc Phương	Xây dựng ý tưởng.	Hoàn thành
		Kiểm tra, chạy thử chương trình.	Test, tìm và sửa lỗi trong quá trình chạy.
		Viết báo cáo, thuyết trình.	Hoàn thành
		Mức độ đóng góp: 100%	

THÔNG TIN ĐỒ ÁN

I. Giới thiệu chung:

1.1 Đặt vấn đề:

Với sự phát triển của đất nước, cùng với đó là những phát triển về dịch vụ du lịch. Để đáp ứng được những yêu cầu của khách du lịch nước nhà thì việc chọn những khách sạn đáp ứng tốt những tiêu chí mà khách hàng mong muốn. Vì thế, việc phân loại đâu là bình luận tích cực hay tiêu cực về khách sạn là vấn đề cần thiết, nhất là trong bối cảnh Internet đang rất phổ biến. Việc phân loại này sẽ giúp khách hàng đưa ra những lựa chọn phù hợp nhất cho chuyến du lịch, nghỉ dưỡng.

1.2 Giới thiệu bài toán :

Là chương trình phân loại bình luận của khách hàng về khách sạn (nhân viên, giá cả, thức ăn,...).

Input: một câu bình luận về khách sạn

Output: bình luận đó là bình luận tích cực hay tiêu cực trong đó:

-Bình luận tích cực là những bình luận tốt, hài lòng về chất lượng khách sạn .

Ví dụ: Đồ ăn ở đây rất ngon

Nhân viên phục vụ thân thiện

-Bình luận tiêu cực là những bình luận chê, không hài lòng, góp ý về chất lượng khách sạn.

Ví dụ: Phòng ở rất bẩn

Cần xem lại phục vụ nhân viên

1.3 Ứng dụng của chương trình:

Dựa vào đánh giá có thể đưa ra chất lượng về khách sạn (phòng ở, thức ăn,...).

Xây dựng hệ thống Recommended Systems.

Là cơ sở ứng dụng vào các bài toán cao hơn:

Phân loại thành nhiều lớp: tích cực, tiêu cực, trung tính...,

Phân loại sâu hơn vào những vấn đề của khách sạn trên bình luận tích cực hoặc tiêu cực. Ví dụ: trong bình luận tiêu cực có thể phân ra bình luận tiêu cực về mặt nào (phòng ở, nhân viên,...)

II. Bài toán:

2.1 Dữ liệu trong bài toán:

2.1.1 Thu thập dữ liệu:

Sử dụng data miner để lấy dữ liệu từ trang <https://www.agoda.com/vi-vn/>

Lưu dữ liệu dưới dạng file (.xlsx)

	Column 1
750	Nhiều muỗi quá, nhưng có lẽ khách sạn nào cũng vậy Phòng khá mới và sạch sẽ, nhân viên nhiệt tình!
751	Resort đẹp, điểm trừ hơi xa trung tâm
752	Ngoài việc đi lại vào trung tâm thì xa còn lại tất cả mọi điểm tuyệt vời
753	Đồ ăn ngon tuy nhiên chưa phong phú, không gian đẹp, nhân viên thân thiện
754	Resort đẹp, dịch vụ tuyệt vời tuy nhiên hơi xa trung tâm thành phố
755	Đáng để quay lại
756	Resort khá đẹp, nhân viên chuyên nghiệp, vị trí thuận lợi
757	Ưu điểm: không gian đẹp, phục vụ tốt Nhược điểm: đi qua nhiều mồ mả.
758	Dịch vụ spa chưa trau chuốt
759	Voi nhưng gì hiện tại ma The Shells đã mang đến cho tôi trong 3 ngày nghỉ thì quá tuyệt, cách dọn tiếp, thức ăn, phục vụ thì k còn gì bàn. Tôi sẽ quay lại The Shells lần sau
760	Khung cảnh đẹp, tiện nghi, đồ ăn rất ok. Mỗi tội đi qua nhiều mồ mả.
761	Cần thuê một đầu bếp giỏi về, với lại phải để sẵn khăn giấy trên bàn ăn.
762	Mình có bị huỷ 1 phòng do 2 bạn đi cùng thời tiết xấu bị lỡ chuyến bay. Bên khách sạn có đổi phòng huỷ thêm 1 đêm cho 2 vợ chồng. Rất nhiệt tình giúp đỡ
763	Phòng ốc sạch, tiện nghi, nhà hàng nấu ăn ngon. Vị trí tương đối thuận lợi.
764	View đẹp, phòng tiện nghi và sạch sẽ, nhân viên thân thiện và chuyên nghiệp. Nhà hàng đồ ăn ngon tuy nhiên phục vụ chậm khách sạn cần khắc phục điều này để hoàn thiện
765	Ocean Bungalow ở khu vực riêng biệt, ngay biển, có hồ bơi riêng, xung quanh sạch sẽ, cây xanh còn ít. Thực đơn nhà hàng ít món, kì nghỉ 3 ngày trở đi thật sự sẽ trở nên nhàm chán.
766	Khách sạn đẹp, không gian rộng rãi, thoải mái, nhân viên rất nhiệt tình
767	Không gian sạch. Rong. Thoang mát. Hồ bơi đẹp
768	Phòng đẹp, view đẹp. Tuy nhiên ăn sáng quá đơn điệu.
769	Nhân viên nhiệt tình, thân thiện.

2.1.2 Tiến hành gắn nhãn:

Các bước thực hiện gắn nhãn:

Thực hiện lần lượt với từng đánh giá, mỗi đánh giá sẽ được thực hiện gồm các bước sau:

Bước 1: Xem đánh giá đó gồm 1 câu hay từ 2 câu trở lên. Nếu là 1 câu thì chuyển đến bước 3 và nếu đánh giá có từ 2 câu trở lên thì đến bước 2.

Bước 2: Tách đánh giá đó thành nhiều đánh giá với mỗi đánh giá là một câu.

Ví dụ. Tại đánh giá bên dưới có hai câu.

3	Lựa chọn tốt cho du lịch nhóm ,gia đình . Phòng rộng rãi ,sạch sẽ
---	-------------------------------------------------------------------

Nên tách thành hai đánh giá.

	Lựa chọn tốt cho du lịch nhóm ,gia đình.
3	Phòng rộng rãi ,sạch sẽ.

Lưu ý: Dấu chấm cuối câu là không cần thiết, có hay không đều được.

Bước 3: Sửa lại những câu viết chưa được hoàn chỉnh. Như những câu có chữ viết tắt, viết thiếu dấu, ... Đối với những câu có xen tiếng anh vào, những câu vô nghĩa thì **xóa** luôn.

Ví dụ câu cần được sửa lại: (Câu trên là đã sửa lại, câu dưới là câu gốc).

211	Hồ bơi rất lớn và đẹp.
212	Ho bơi rat lon va dep.

Ví dụ:

395	Tôi thích nhất Buffet ở đây.
-----	------------------------------

Sửa chữ buffet lại thành ăn sáng nếu có thể hoặc có thể xóa luôn cũng được.

Những từ tiếng anh khuyến khích nên dịch (những từ liên quan đến lĩnh vực khách sạn): hotel, buffet, homestay..... Đặc biệt những chữ như ok, good, great, bad.... thì nên giữ lại nguyên trạng.

Ví dụ: Ks này rất ok.

Thì có thể sửa lại thành: Khách sạn này rất ok.

Bước 4: Gán nhãn. Nếu là khen thì đánh số 1 và chê thì là 0 vào cột cạnh bên.

Trong lúc gán nhãn sẽ gặp một số câu không rõ là khen hay chê thì có thể xóa, hoặc những câu cần vận dụng những câu khác mới biết được nó khen hay chê thì cũng xóa.

Ví dụ: Tôi đặt phòng tại khách sạn 5 sao. Mà ngày đến nhận phòng cứ ngỡ khách sạn 2 hay 3 sao.

Có thể xóa luôn cả 2 câu.

Dưới đây là file sau khi đã tiến hành bước gán nhãn

	text	label
0	Không có chào đón như trong giới thiệu	0
1	Lựa chọn tốt cho du lịch nhóm ,gia đình	1
2	phòng yên tĩnh dù sát mặt đường lớn , sạch sẽ.	1
3	phòng sạch đẹp	1
4	Khách sạn sạch sẽ, gần biển, gần siêu thị Lotte gần quán cafe sân vườn rộng đẹp.	1
5	Khách sạn mới, sạch và đẹp	1
6	Nhà hàng hơi chật	0
7	Tốt, tôi sẽ trở lại đây khi đến Vũng Tàu	1
8	Khách sạn ngay trung tâm, tránh được khu nhà nghỉ khách sạn đông đúc	1
9	Tốt	1
10	hệ thống nước ko tốt nên nước nóng ko ổn định, lúc nóng lúc lạnh ko đều	0
11	Khách sạn đẹp vừa túi tiền, gần lottle có cho mượn xe đạp miễn phí.	1
12	Khách sạn mới, đẹp, rất sạch, yên tĩnh, tôi đã có những ngày nghỉ rất dễ chịu ở đây. .	1

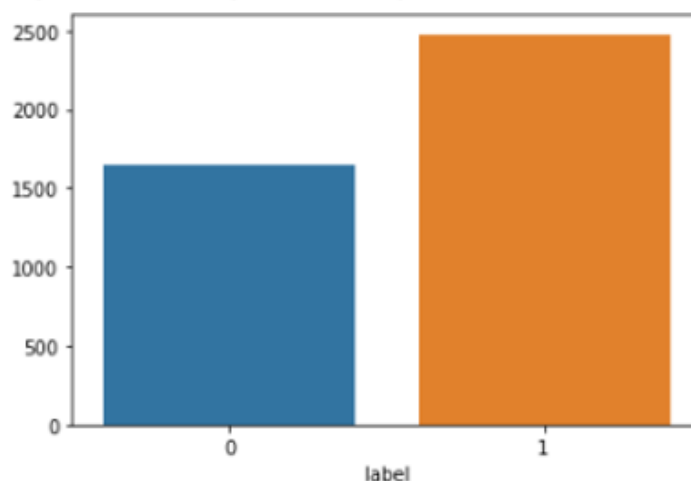
2.1.3 Nhận xét về dữ liệu:

Dữ liệu gồm tất cả:

- 2 cột: text và label. Cột text là bình luận thu thập được. Cột label là nhãn tương ứng với bình luận đó (1 là bình luận tích cực, 0 là bình luận tiêu cực).

- 4124 dòng:

Trong đó: có 2476 bình luận tích cực - label 1 (60,04%) và 1648 bình luận tiêu cực - label 0 (39,96%).



2.2 Feature Engineering:

Tách từ và tiền xử lý dữ liệu:

a) Tách từ theo dấu cách:

```
def standardize_data(t):
    t = str(t)
    t = t.lower()
    t = t.replace(' ', ' ').replace('.', ' ') \
        .replace(", ", " ").replace(";", " ") \
        .replace(":", " ").replace("?", " ") \
        .replace("!", " ").replace("?", " ") \
        .replace("-", " ").replace("?", " ")
    t = t.strip()
    return t
```

Không phân tích theo nghĩa tiếng việt _ phân theo dấu cách

```
df['text'] = df['text'].apply(standardize_data)
df
```

	text	label
0	không có chào đón như trong giới thiệu	0
1	lựa chọn tốt cho du lịch nhóm gia đình	1
2	phòng yên tĩnh dù sát mặt đường lớn sạch sẽ	1
3	phòng sạch đẹp	1
4	khách sạn sạch sẽ gần biển gần siêu thị lott...	1
...
4119	thời thì treo biển không nhận khách đi oto cho...	0
4120	12 giờ đêm về thì khách sạn tối hìn	0
4121	nói 3 sao thì hơi quá	0
4122	thua khách sạn mình ở nhà trang vẫn 3 sao	0

b) Tách từ theo nghĩa của từ (sử dụng Pyvi):

```
!pip install pyvi
```

```
Requirement already satisfied: pyvi in /usr/local/lib/python3.6/dist-packages (0.1)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.6/dist-packages (from pyvi) (0.22.2.post1)
Requirement already satisfied: sklearn-crfsuite in /usr/local/lib/python3.6/dist-packages (from pyvi) (0.3.6)
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.6/dist-packages (from scikit-learn->pyvi) (0.16.0)
Requirement already satisfied: numpy>=1.11.0 in /usr/local/lib/python3.6/dist-packages (from scikit-learn->pyvi) (1.18.5)
Requirement already satisfied: scipy>=0.17.0 in /usr/local/lib/python3.6/dist-packages (from scikit-learn->pyvi) (1.4.1)
Requirement already satisfied: tabulate in /usr/local/lib/python3.6/dist-packages (from sklearn-crfsuite->pyvi) (0.8.7)
Requirement already satisfied: tqdm>=2.0 in /usr/local/lib/python3.6/dist-packages (from sklearn-crfsuite->pyvi) (4.41.1)
Requirement already satisfied: python-crfsuite>=0.8.3 in /usr/local/lib/python3.6/dist-packages (from sklearn-crfsuite->pyvi) (0.9.7)
Requirement already satisfied: six in /usr/local/lib/python3.6/dist-packages (from sklearn-crfsuite->pyvi) (1.15.0)
```

```
from pyvi import ViTokenizer
def Token(t):
    return ViTokenizer.tokenize(t)
```

Pyvi

```
df['text'] = df['text'].apply(standardize_data)
df['text'] = df['text'].apply(Token)
df
```

	text	label
0	không có chào_đón như trong giới_thiệu	0
1	lựa_chọn tốt cho du_lịch nhóm gia_đình	1
2	phòng yên_tĩnh dù sát mặt_đường lớn sạch_sẽ	1
3	phòng sạch đẹp	1
4	khách_sạn sạch_sẽ gần biển gần siêu_thị lotte ...	1
...
4119	thời_thì treo biển không nhận khách đi oto cho...	0
4120	12 giờ đêm về thì khách_sạn tối hìn	0
4121	nói 3 sao thì hơi quá	0
4122	thua khách_sạn mình ở nhà trang vẫn 3 sao	0
4123	vẫn nằm ở trục đường chính một chiều đồng ngệt...	0

c) *CountVectorizer*:

CountVectorizer phương thức để chuyển đổi dữ liệu văn bản thành các vector vì mô hình chỉ có thể xử lý dữ liệu số. Trong *CountVectorizer*, chúng ta chỉ đếm số lần một từ xuất hiện trong tài liệu dẫn đến sai lệch có lợi cho hầu hết các từ thường xuyên. điều này kết thúc trong việc bỏ qua các từ hiếm có thể giúp xử lý dữ liệu của chúng ta hiệu quả hơn.

```
In [99]: X=df['text']
y=df['label']
emb=CountVectorizer().fit(X)
X_train,X_test, y_train, y_test=train_test_split(X, y, test_size=0.2)
X_train=emb.transform(X_train)
X_test=emb.transform(X_test)
```

d) *TF-IDF*:

TF-IDF (Term Frequency – Inverse Document Frequency) là 1 kĩ thuật sử dụng trong khai phá dữ liệu văn bản. Trọng số này được sử dụng để đánh giá tầm quan trọng của một từ trong một văn bản. Giá trị cao thể hiện độ quan trọng cao và nó phụ thuộc vào số lần từ xuất hiện trong văn bản nhưng bù lại bởi tần suất của từ đó trong tập dữ liệu. Một vài biến thể của tf-idf thường được sử dụng trong các hệ thống tìm kiếm như một công cụ chính để đánh giá và sắp xếp văn bản dựa vào truy vấn của người dùng. Tf-idf cũng được sử dụng để lọc những từ stopwords trong các bài toán như tóm tắt văn bản và phân loại văn bản.

TF: Term Frequency (Tần suất xuất hiện của từ) là số lần từ xuất hiện trong văn bản. Vì các văn bản có thể có độ dài ngắn khác nhau nên một số từ có thể xuất hiện nhiều lần trong một văn bản dài hơn là một văn bản ngắn. Như vậy, term frequency thường được chia cho độ dài văn bản(tổng số từ trong một văn bản).

$$tf(t, d) = \frac{f(t, d)}{\max\{f(w, d) : w \in d\}}$$

Trong đó:

$tf(t, d)$: tần suất xuất hiện của từ t trong văn bản d

$f(t, d)$: Số lần xuất hiện của từ t trong văn bản d

$\max(\{f(w, d) : w \in d\})$: Số lần xuất hiện của từ có số lần xuất hiện nhiều nhất trong văn bản d

IDF: Inverse Document Frequency(Nghịch đảo tần suất của văn bản), giúp đánh giá tầm quan trọng của một từ . Khi tính toán TF , tất cả các từ được coi như có độ quan trọng bằng nhau. Nhưng một số từ như “is”, “of” và “that” thường xuất hiện rất nhiều lần nhưng độ quan trọng là không cao. Như thế chúng ta cần giảm độ quan trọng của những từ này xuống.

$$idf(t, D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

Trong đó:

$idf(t, D)$: giá trị idf của từ t trong tập văn bản

$|D|$: Tổng số văn bản trong tập D

$|\{d \in D : t \in d\}|$: thể hiện số văn bản trong tập D có chứa từ t .

Cơ số logarit trong công thức này không thay đổi giá trị idf của từ mà chỉ thu hẹp khoảng giá trị của từ đó. Vì thay đổi cơ số sẽ dẫn đến việc giá trị của các từ thay đổi bởi một số nhất định và tỷ lệ giữa các trọng lượng với nhau sẽ không thay đổi. (nói cách khác, thay đổi cơ số sẽ

không ảnh hưởng đến tỷ lệ giữa các giá trị IDF). Việc sử dụng logarit nhằm giúp giá trị tf-idf của một từ nhỏ hơn, do chúng ta có công thức tính tf-idf của một từ trong 1 văn bản là tích của tf và idf của từ đó.

Cụ thể, chúng ta có **công thức tính tf-idf** hoàn chỉnh như sau: $\text{tfidf}(t, d, D) = \text{tf}(t, d) \times \text{idf}(t, D)$

Khi đó: Những từ có giá trị TF-IDF cao là những từ xuất hiện nhiều trong văn bản này, và xuất hiện ít trong các văn bản khác. Việc này giúp lọc ra những từ phổ biến và giữ lại những từ có giá trị cao (từ khoá của văn bản đó).

TF-IDF

```
In [88]: X_train,X_test, y_train, y_test=train_test_split(df['text'],df['label'],test_size=0.2)
emb=TfidfVectorizer()
emb.fit(df['text'])
X_train=emb.transform(X_train)
X_test=emb.transform(X_test)
```

2.3 Model:

a) BernoulliNB:

Mô hình phân lớp nhị phân là hữu ích nếu các vector đặc trưng cũng phân lớp nhị phân (tức là được đánh số 0 và 1). Phân loại văn bản sẽ được dựa trên một “bag of word”, trong đó, mỗi từ vựng sẽ được đánh số là 0 – với những từ không có trong văn bản đang xem xét và 1 – với những từ xuất hiện trong văn bản đang xem xét.

```
from sklearn.naive_bayes import BernoulliNB
model=BernoulliNB()
model.fit(X_train,y_train)
predict=model.predict(X_test)
print('Accuracy: ',accuracy_score(y_test,predict))
print(classification_report(y_test,predict))
print(confusion_matrix(y_test,predict))
```

b) LogictisRegression:

Phương pháp hồi quy logistic là một mô hình hồi quy nhằm dự đoán giá trị đầu ra rời rạc (*discrete target variable*) y ứng với một véc-tơ đầu vào x . Việc này tương đương với chuyện phân loại các đầu vào x vào các nhóm y tương ứng.

```
from sklearn.linear_model import LogisticRegression
model=LogisticRegression()
model.fit(X_train,y_train)
predict=model.predict(X_test)
print('Accuracy: ',accuracy_score(y_test, predict))
print(classification_report(y_test,predict))
print(confusion_matrix(y_test,predict))
```

c) SVC:

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

```
from sklearn.svm import SVC
model=SVC().fit(X_train,y_train)
predict=model.predict(X_test)
print('Accuracy: ',accuracy_score(y_test, predict))
print('Classification report:\n',classification_report(y_test,predict))
print('Confusion matrix: \n',confusion_matrix(y_test,predict))
```

d) Decision tree:

Việc quan sát, suy nghĩ và ra các quyết định của con người thường được bắt đầu từ các câu hỏi. Machine learning cũng có một mô hình ra quyết định dựa trên các câu hỏi. Mô hình này có tên là *cây quyết định* (*decision tree*).

```
from sklearn.tree import DecisionTreeClassifier
model=DecisionTreeClassifier().fit(X_train,y_train)
predict=model.predict(X_test)
print('Accuracy: ',accuracy_score(y_test, predict))
print(classification_report(y_test,predict))
print(confusion_matrix(y_test,predict))
```

e) *Random Forest*:

Đây là phương pháp xây dựng một tập hợp rất nhiều cây quyết định và sử dụng phương pháp voting để đưa ra quyết định về biến target cần được dự báo.

```
from sklearn.ensemble import RandomForestClassifier
model=RandomForestClassifier().fit(X_train,y_train)
predict=model.predict(X_test)
print('Accuracy: ',accuracy_score(y_test, predict))
print(classification_report(y_test,predict))
print(confusion_matrix(y_test,predict))
```

2.4 Nhận xét:

Nhận xét các model:

-Tách từ theo dấu cách:

Model	Bernouli NB	Logistic Regression	SVC	Decision Trees	RandomForest
CountVectorizer	88,97	93,57	93,69	88,73	94,18
TF-IDF Vectorizer	88,96	94,67	95,27	87,15	93,21

-Tách từ theo Pyvi:

Model	Bernouli NB	Logistic Regression	SVC	Decision Trees	RandomForest
CountVectorizer	88,85	94,54	94,06	89,82	94,54
TF-IDF Vectorizer	88,85	94,67	95,51	89,45	94,67

Với mỗi phương pháp tách từ (theo dấu cách và theo nghĩa), sẽ sử dụng 2 phương pháp Feature Engineering(CountVectorizer và TF-IDF) và 5 model do Scikit-Learn hỗ trợ và thu được 20 Accuracy ứng với 20 trường hợp trên. Đa số các Accuracy đều trên 85%, và các model khi sử dụng TF-IDF cao hơn sử dụng CountVectorizer. Accuracy cao nhất ở 20 trường hợp trên là: 95,51% (Tách từ theo Pyvi, TF-IDF Vectorizer, model SVC). Model này còn có các thông số như sau:

Classification report:

	precision	recall	f1-score	support
0	0.94	0.96	0.95	347
1	0.97	0.95	0.96	478
accuracy			0.96	825
macro avg	0.95	0.96	0.95	825
weighted avg	0.96	0.96	0.96	825

Confusion matrix:

```
[[333 14]
 [ 23 455]]
```

Accuracy: 95,51% là độ chính xác và được tính là số lượng bình luận dự đoán đúng / tổng số bình luận đem dự đoán.

Classification report: trong tập test có 347 bình luận nhãn 0, 478 bình luận nhãn 1. Trong đó f1-score của nhãn 0 và 1 tương ứng là 95% và 96% với

Và **f1-score** trung bình theo cách tính weighted avg (có nhân với tỷ lệ của số lượng nhãn) là 0.96 với $(0.96 = 0.95 \cdot (347/825) + 0.96 \cdot (478/825))$

Confusion matrix: Trong 347 bình luận nhãn 0 thì model dự đoán chính xác 333 bình luận và 478 bình luận nhãn 1 thì model dự đoán chính xác 455 bình luận \Rightarrow accuracy = $(333+455) / 825 = 0,9551$. Và đây cũng là model để tiến hành demo.

III. Demo:

Bình luận tích cực:

```
[25] 1 comment=input()
      2 predict, comment1= du_doan(comment)
      3 print('-----')
      4 for i in range(len(predict)):
      5     if comment1[i]!=" and comment1[i]!=' ':
      6         print(comment1[i])
      7     print('-> Bình luận tích cực' if predict[i] else '-> Bình luận tiêu cực')
```



Sạch sẽ, nhân viên phục vụ nhiệt tình

Sạch sẽ, nhân viên phục vụ nhiệt tình
-> Bình luận tích cực

```
[26] 1 comment=input()
      2 predict, comment1= du_doan(comment)
      3 print('-----')
      4 for i in range(len(predict)):
      5     if comment1[i]!=" and comment1[i]!=' ':
      6         print(comment1[i])
      7     print('-> Bình luận tích cực' if predict[i] else '-> Bình luận tiêu cực')
```



Tiện nghi và nhân viên nhiệt tình, dễ thương

Tiện nghi và nhân viên nhiệt tình, dễ thương
-> Bình luận tích cực

```
[27] 1 comment=input()
      2 predict, comment1= du_doan(comment)
      3 print('-----')
      4 for i in range(len(predict)):
      5     if comment1[i]!=" and comment1[i]!=' ':
      6         print(comment1[i])
      7     print('-> Bình luận tích cực' if predict[i] else '-> Bình luận tiêu cực')
```



Mình bị đau chân và chỉ chủ đổi phòng cho mình cho thuận tiện đi lại, điều đó thật tuyệt vời. Chỉ chủ thân thiện, chỗ ở sạch sẽ.

Mình bị đau chân và chỉ chủ đổi phòng cho mình cho thuận tiện đi lại, điều đó thật tuyệt vời
-> Bình luận tích cực
Chỉ chủ thân thiện, chỗ ở sạch sẽ
-> Bình luận tích cực

```
[28] 1 comment=input()
      2 predict, comment1= du_doan(comment)
      3 print('-----')
      4 for i in range(len(predict)):
      5     if comment1[i]!=" and comment1[i]!=' ':
      6         print(comment1[i])
      7     print('-> Bình luận tích cực' if predict[i] else '-> Bình luận tiêu cực')
```



Bữa sáng khá vừa miệng, thức ăn rất phong Phú. Không gian khá thoải mái và tráng miệng rất ngon.

Bữa sáng khá vừa miệng, thức ăn rất phong Phú
-> Bình luận tích cực
Không gian khá thoải mái và tráng miệng rất ngon
-> Bình luận tích cực

Bình luận tiêu cực:

```
[29] 1 comment=input()
      2 predict, comment1= du_doan(comment)
      3 print('-----')
      4 for i in range(len(predict)):
      5     if comment1[i]!=" and comment1[i]!=' ':
      6         print(comment1[i])
      7     print('-> Bình luận tích cực' if predict[i] else '-> Bình luận tiêu cực')
```



Vòi sen hơi bị cũ, phòng không có cửa sổ nên hơi bí

Vòi sen hơi bị cũ, phòng không có cửa sổ nên hơi bí
-> Bình luận tiêu cực

```
[31] 1 comment=input()
      2 predict, comment1= du_doan(comment)
      3 print('-----')
      4 for i in range(len(predict)):
      5     if comment1[i]!=" and comment1[i]!=' ':
      6         print(comment1[i])
      7     print('-> Bình luận tích cực' if predict[i] else '-> Bình luận tiêu cực')
```



phòng cách âm không tốt lắm, ngồi trong phòng có thể nghe rõ ràng người trong hành lan đi và nói chuyện!

phòng cách âm không tốt lắm, ngồi trong phòng có thể nghe rõ ràng người trong hành lan đi và nói chuyện!
-> Bình luận tiêu cực

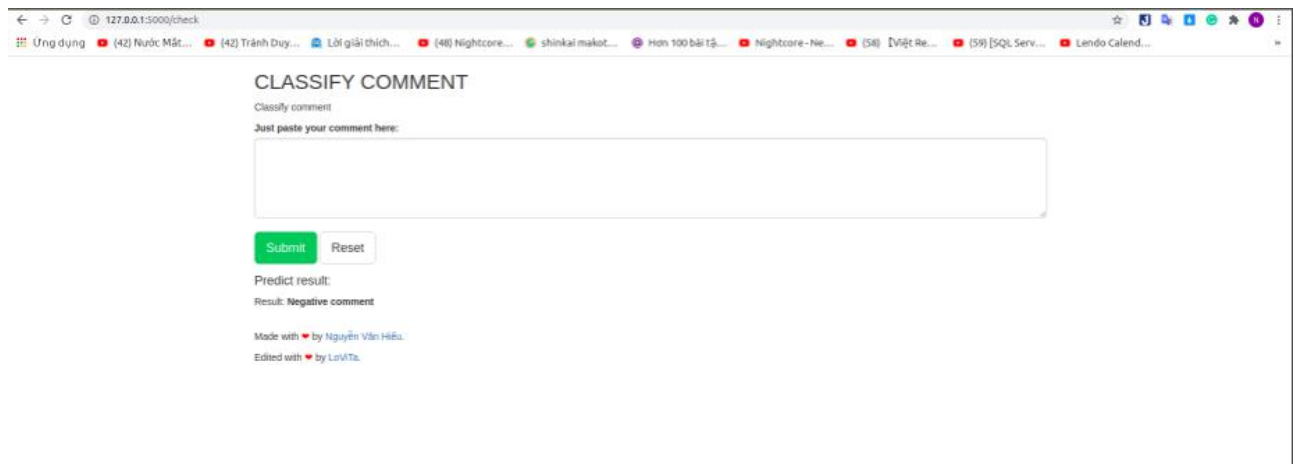
```
[32] 1 comment=input()
      2 predict, comment1= du_doan(comment)
      3 print('-----')
      4 for i in range(len(predict)):
      5     if comment1[i]!=" and comment1[i]!=' ':
      6         print(comment1[i])
      7     print('-> Bình luận tích cực' if predict[i] else '-> Bình luận tiêu cực')
```



Phòng xuống cấp so với ảnh chụp. Phòng tắm riêng tang 3 bị rò điện.

Phòng xuống cấp so với ảnh chụp
-> Bình luận tiêu cực
Phòng tắm riêng tang 3 bị rò điện
-> Bình luận tiêu cực

◊ Ngoài ra, để thuận tiện cho việc sử dụng thì nhóm em đã xây dựng thêm giao diện cho bài toán. Giao diện có cấu trúc như bên dưới:



KẾT LUẬN

I. Hạn chế:

- Vẫn còn một số câu bị phân loại sai lớp.
- Số lớp dự đoán còn hạn chế.

II. Hướng phát triển:

- Nâng cao ứng dụng để giải quyết các câu chưa phân loại được.
- Mở rộng thêm phạm vi bài toán (tăng thêm số lượng lớp và câu đánh giá) cho phù hợp với nhiều loại hình du lịch khác nhau.