Restaurant Sentiment Analysis

Name of Members:-

1. Arjun Nandlal Bhardwaj (M210683CA)
2. Alok Pandey (M210701CA)

Definition:-

The purpose of this analysis is to build a prediction model
to predict whether a review on the restaurant is positive or negative.

Code for Training Data

```
import pandas as pd

# import data
dataset = pd.read_csv(r"/Users/alokpandey181/Downloads/python project/
a1_RestaurantReviews_HistoricDump.tsv”, delimiter='\t', quoting=3)
print(dataset.shape)
print(dataset.head())

# data cleaning
# By default, NLTK (Natural Language Toolkit) includes a list of 40 stop words, including: “a”, “an”,
“the”, “of”, “in”, etc. The stopwords in nltk are the most common words in data.

import re
import nltk
nltk.download('stopwords')

from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer

ps = PorterStemmer()

all_stopwords = stopwords.words('english')
all_stopwords.remove('not')

# filteration start
corpus = []

for i in range(0, 900):
    review = re.sub('[^a-zA-Z]', ' ', dataset['Review'][i])
    review = review.lower()
    review = review.split()
    review = [ps.stem(word) for word in review if not word in set(all_stopwords)]
    review = ' '.join(review)
    corpus.append(review)

print(corpus)
```

```
# data transformation
# It is used to transform a given text into a vector on the basis of the frequency (count)
# of each word that occurs in the entire text.

from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=1420)
X = cv.fit_transform(corpus).toarray()
y = dataset.iloc[:, -1].values

# Pickling is a way to convert a python object (list, dict, etc.) into a character stream.
# Saving BoW dictionary to later use in prediction

import pickle

bow_path = r"/Users/alokpandey181/Downloads/python project/c1_BoW_Sentiment_Model.pkl"
pickle.dump(cv, open(bow_path, "wb"))

# Dividing dataset into training and test set
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=0)

# Model fitting (Naive Bayes)
from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(X_train, y_train)

# Exporting NB Classifier to later use in prediction
# Especially used to execute tasks parallely
import joblib
joblib.dump(classifier, r"/Users/alokpandey181/Downloads/python project/
c2_Classifier_Sentiment_Model")

# model performance
y_pred = classifier.predict(X_test)
from sklearn.metrics import accuracy_score
print(accuracy_score(y_test, y_pred))
```

Code for Testing Data

```
import pandas as pd

#import data
dataset = pd.read_csv(r"/Users/alokpandey181/Downloads/python project/
a2_RestaurantReviews_FreshDump.tsv", delimiter = '\t', quoting = 3)

l=len(dataset)
print(dataset.shape)
print(dataset.head())


#data cleaning
import re
import nltk
nltk.download('stopwords')

# By default, NLTK (Natural Language Toolkit) includes a list of 40 stop words, including: “a”, “an”,
“the”, “of”, “in”, etc. The stopwords in nltk are the most common words in data.

from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
ps = PorterStemmer()

all_stopwords = stopwords.words('english')
all_stopwords.remove('not')

#filteration start
corpus=[]

for i in range(0, l):
  review = re.sub('[^a-zA-Z]', ' ', dataset['Review'][i])
  review = review.lower()
  review = review.split()
  review = [ps.stem(word) for word in review if not word in set(all_stopwords)]
  review = ' '.join(review)
  corpus.append(review)

print(corpus)


# Loading BoW dictionary
# Pickling is a way to convert a python object (list, dict, etc.) into a character stream.
import pickle
cvFile=r"/Users/alokpandey181/Downloads/python project/c1_BoW_Sentiment_Model.pkl"
cv = pickle.load(open(cvFile, "rb"))

X_fresh = cv.transform(corpus).toarray()
print(X_fresh.shape)

#Predictions (via sentiment classifier)
import joblib
classifier = joblib.load(r"/Users/alokpandey181/Downloads/python project/
c2_Classifier_Sentiment_Model")

y_pred = classifier.predict(X_fresh)
print(y_pred)

dataset['predicted_label'] = y_pred.tolist()
print(dataset.head())
```

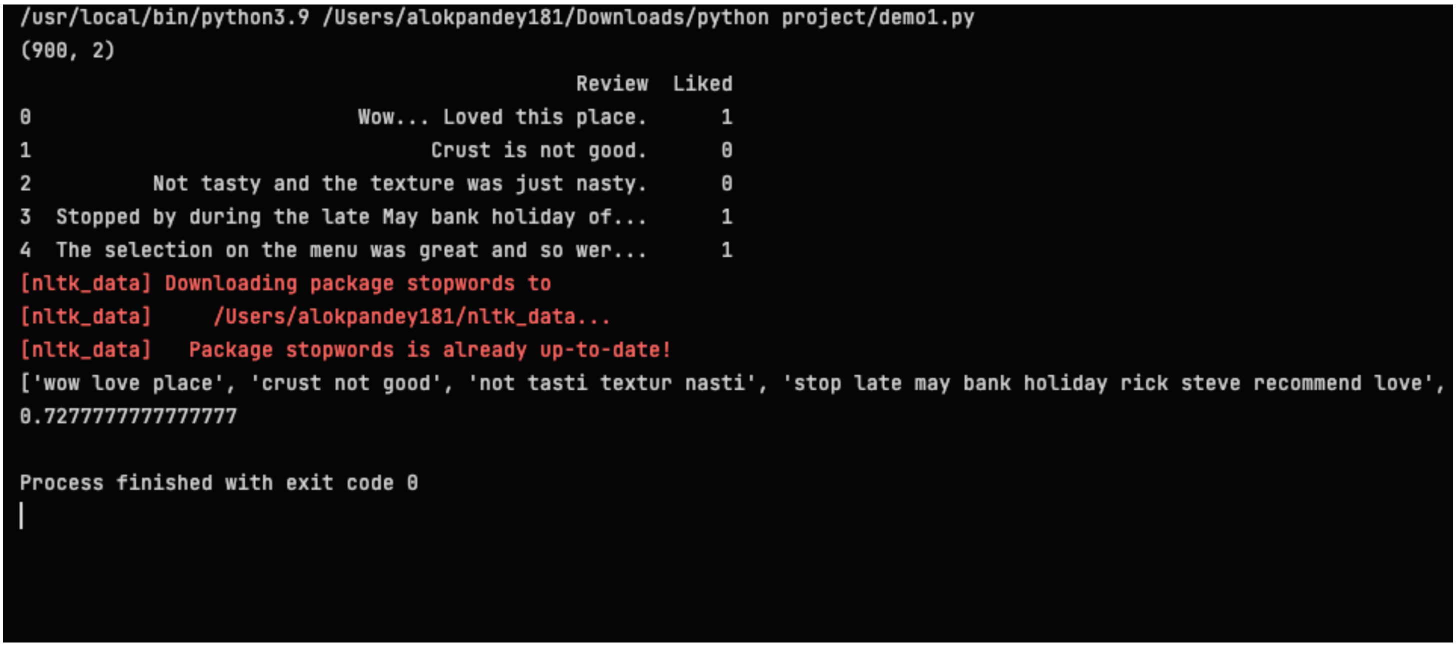
```
dataset.to_csv(r"/Users/alokpandey181/Downloads/python project/
c3_Predicted_Sentiments_Fresh_Dump.tsv", sep='\t', encoding='UTF-8', index=False)

#print pic hart
count1 = (dataset['predicted_label'] == 1).sum()
count0 = (dataset['predicted_label'] == 0).sum()

# Creating dataset
Sentiments = ['Good Comments', 'Bad Comments']
data = [count1,count0]

# Creating plot
from matplotlib import pyplot as plt
plt.pie(data, labels=Sentiments,autopct='%1.1f%%', startangle=90)
plt.show()
```

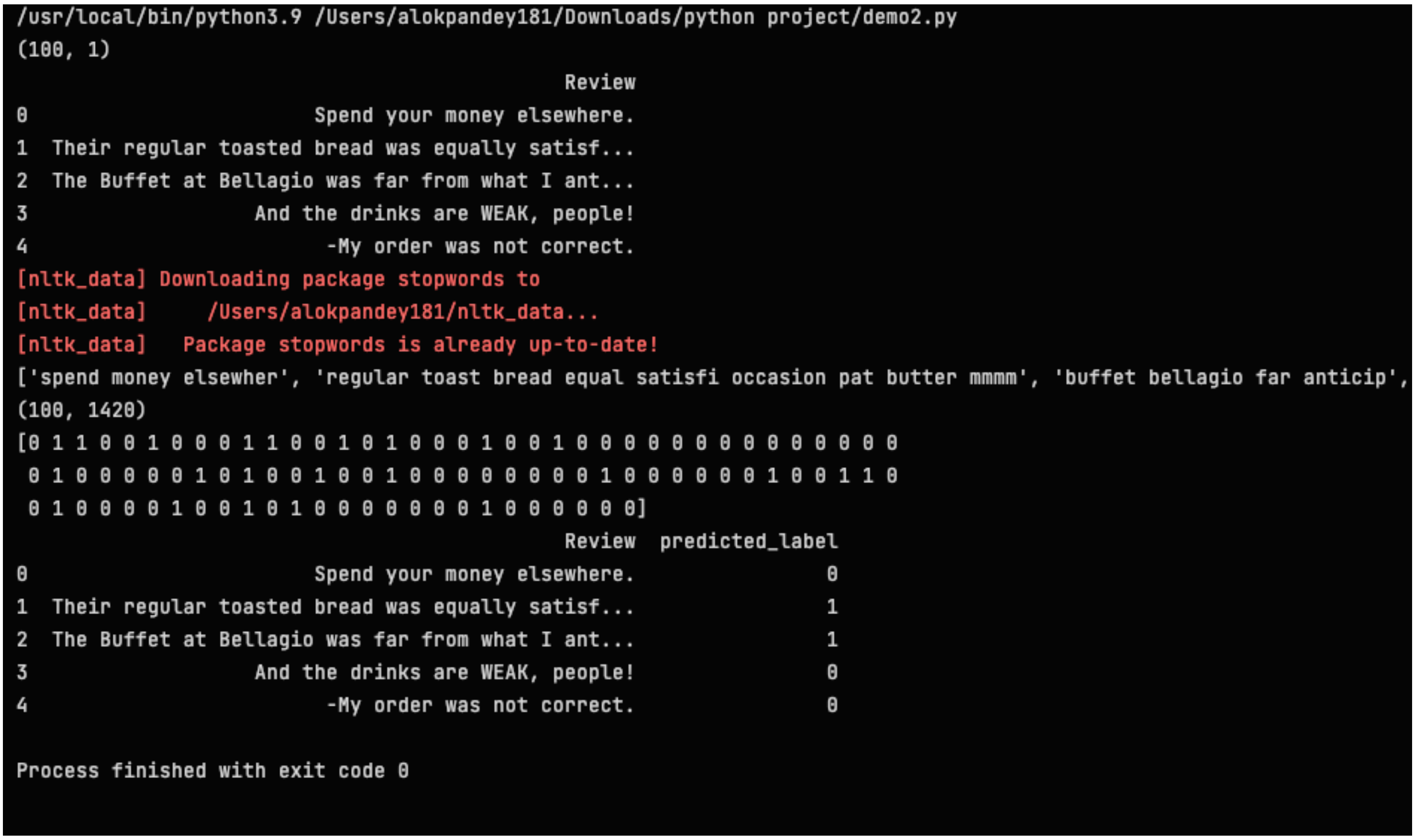
Training Data Output



```
/usr/local/bin/python3.9 /Users/alokpandey181/Downloads/python project/demo1.py
(900, 2)
                                            Review  Liked
0                         Wow... Loved this place.      1
1                               Crust is not good.      0
2          Not tasty and the texture was just nasty.      0
3  Stopped by during the late May bank holiday of...      1
4  The selection on the menu was great and so wer...      1
[nltk_data] Downloading package stopwords to
[nltk_data]     /Users/alokpandey181/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
['wow love place', 'crust not good', 'not tasti textur nasti', 'stop late may bank holiday rick steve recommend love',
0.7277777777777777

Process finished with exit code 0
|
```

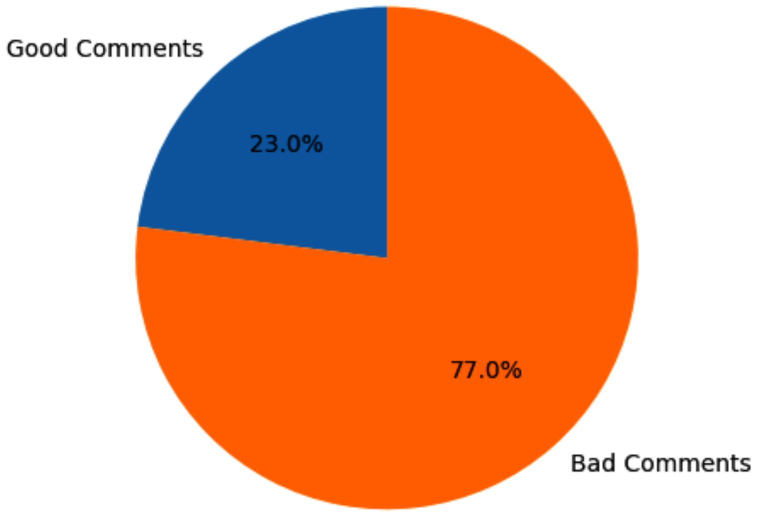
Testing Data Output



```
/usr/local/bin/python3.9 /Users/alokpandey181/Downloads/python project/demo2.py
(100, 1)
                                              Review
0                         Spend your money elsewhere.
1  Their regular toasted bread was equally satisf...
2  The Buffet at Bellagio was far from what I ant...
3                    And the drinks are WEAK, people!
4                          -My order was not correct.
[nltk_data] Downloading package stopwords to
[nltk_data]     /Users/alokpandey181/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
['spend money elsewher', 'regular toast bread equal satisfi occasion pat butter mmmm', 'buffet bellagio far anticip',
(100, 1420)
[0 1 1 0 0 1 0 0 0 1 1 0 0 1 0 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 1 0 0 0 0 0 1 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 1 1 0
 0 1 0 0 0 0 1 0 0 1 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0]
                                              Review  predicted_label
0                         Spend your money elsewhere.                0
1  Their regular toasted bread was equally satisf...                1
2  The Buffet at Bellagio was far from what I ant...                1
3                    And the drinks are WEAK, people!                0
4                          -My order was not correct.                0

Process finished with exit code 0
```

Output



Good Comments
Bad Comments