Task_2_solution

September 4, 2021

0.0.1 Task 2

[ ]:

```
import numpy as np
import re
```

[ ]:

```
sample_listofreview=['Food was damn good!:)','good food. recommended','YOU␣
 ↪PEOPLE ARE THE BEST!!!','It tasted very bad..... too bad service as well']
print(sample_listofreview)
```

```
['Food was damn good!:)', 'good food. recommended', 'YOU PEOPLE ARE THE
BEST!!!', 'It tasted very bad… too bad service as well']
```

[ ]:

```
sample=sample_listofreview[2]
sample=re.sub('[!@#$%.^&*()_+}{":?><"}]','',sample)
print(sample)
sample=sample.lower()
print('\n')
print(sample)
```

```
YOU PEOPLE ARE THE BEST


you people are the best
```

Preprocessing the data

[ ]:

```
processed_reviews=[]
for review in sample_listofreview:
    for words in review:
        review=re.sub('[!@#$%^&*.()_+}{":?><"}]','',review)
        review=review.lower()
    processed_reviews.append(review)

print(processed_reviews)
```

```
['food was damn good', 'good food recommended', 'you people are the best', 'it
tasted very bad too bad service as well']
```

Collecting the uniques words

[ ]:

```
unique_words=[]
for review in processed_reviews:
    for word in review.split():
        if word not in unique_words:
            unique_words.append(word)

print('list of unique words',unique_words)
```

```
list of unique words ['food', 'was', 'damn', 'good', 'recommended', 'you',
'people', 'are', 'the', 'best', 'it', 'tasted', 'very', 'bad', 'too', 'service',
'as', 'well']
```

[ ]:

```
feature_matrix=np.zeros((len(processed_reviews),len(unique_words)))
for n,review in enumerate(processed_reviews):
    for word in review.split():
        feature_matrix[n][unique_words.index(word)]=review.count(word)

feature_matrix
```

[ ]:

```
array([[1., 1., 1., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0.],
       [1., 0., 0., 1., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0.],
       [0., 0., 0., 0., 0., 1., 1., 1., 1., 1., 0., 0., 0., 0., 0., 0.,
        0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 1., 1., 2., 1., 1.,
        2., 1.]])
```

Assignment task

0.1 1. Data Preprocessing

[ ]:

```
import pandas as pd
import numpy as np
import string

df = pd.read_csv('../../datasets/Tweets.csv')

df['text']
```

[ ]:

```
0                       @VirginAmerica What @dhepburn said.
1        @VirginAmerica plus you've added commercials t…
2        @VirginAmerica I didn't today… Must mean I n…
3        @VirginAmerica it's really aggressive to blast…
4        @VirginAmerica and it's a really big bad thing…
                                 …
14635    @AmericanAir thank you we got on a different f…
```

```
14636    @AmericanAir leaving over 20 minutes Late Flig…
14637    @AmericanAir Please bring American Airlines to…
14638    @AmericanAir you have my money, you change my …
14639    @AmericanAir we have 8 ppl so we need 2 know h…
Name: text, Length: 14640, dtype: object
```

```
[ ]: df['airline_sentiment'].value_counts()
```

```
[ ]: negative    9178
     neutral     3099
     positive    2363
     Name: airline_sentiment, dtype: int64
```

```
[ ]: print(df.iloc[:,-3].value_counts(normalize = True))
```

```
negative    0.626913
neutral     0.211680
positive    0.161407
Name: airline_sentiment, dtype: float64
```

0.1.1 1a. Remove Special Characters

```
[ ]: df['text'].replace(regex=True, inplace=True, to_replace=r'[@_!#$%^&*()<>?/\|}{~:
      ↪]', value=r'')

     df['text']
```

```
[ ]: 0                         VirginAmerica What dhepburn said.
     1        VirginAmerica plus you've added commercials to…
     2        VirginAmerica I didn't today… Must mean I ne…
     3        VirginAmerica it's really aggressive to blast …
     4        VirginAmerica and it's a really big bad thing …
                                    …
     14635    AmericanAir thank you we got on a different fl…
     14636    AmericanAir leaving over 20 minutes Late Fligh…
     14637    AmericanAir Please bring American Airlines to …
     14638    AmericanAir you have my money, you change my f…
     14639    AmericanAir we have 8 ppl so we need 2 know ho…
     Name: text, Length: 14640, dtype: object
```

0.1.2 1b. Remove Non English Alphabets

```
[ ]: df['text'].replace(regex=True, inplace=True, to_replace=r'[^A-Za-z0-9 ]+',␣
      ↪value=r'')

     df['text']
```

```
[ ]: 0                         VirginAmerica What dhepburn said
     1        VirginAmerica plus youve added commercials to …
     2        VirginAmerica I didnt today Must mean I need t…
     3        VirginAmerica its really aggressive to blast o…
     4        VirginAmerica and its a really big bad thing a…
                                    …
     14635    AmericanAir thank you we got on a different fl…
     14636    AmericanAir leaving over 20 minutes Late Fligh…
     14637    AmericanAir Please bring American Airlines to …
     14638    AmericanAir you have my money you change my fl…
     14639    AmericanAir we have 8 ppl so we need 2 know ho…
     Name: text, Length: 14640, dtype: object
```

0.1.3 1c. Remove Numerical Values

```
[ ]: # Regex to remove all the numbers and ending space to match readability
     df['text'].replace(regex=True, inplace=True, to_replace=r'\d+\s*', value=r'')

     df['text']
```

```
[ ]: 0                         VirginAmerica What dhepburn said
     1        VirginAmerica plus youve added commercials to …
     2        VirginAmerica I didnt today Must mean I need t…
     3        VirginAmerica its really aggressive to blast o…
     4        VirginAmerica and its a really big bad thing a…
                                    …
     14635    AmericanAir thank you we got on a different fl…
     14636    AmericanAir leaving over minutes Late Flight N…
     14637    AmericanAir Please bring American Airlines to …
     14638    AmericanAir you have my money you change my fl…
     14639    AmericanAir we have ppl so we need know how ma…
     Name: text, Length: 14640, dtype: object
```

0.1.4 1d. All words in lower case

```
[ ]: df["text"] = df["text"].apply(lambda x: x.lower())

     df["text"]
```

```
[ ]: 0                         virginamerica what dhepburn said
     1        virginamerica plus youve added commercials to …
     2        virginamerica i didnt today must mean i need t…
     3        virginamerica its really aggressive to blast o…
     4        virginamerica and its a really big bad thing a…
                                    …
     14635    americanair thank you we got on a different fl…
     14636    americanair leaving over minutes late flight n…
```

```
14637    americanair please bring american airlines to …
14638    americanair you have my money you change my fl…
14639    americanair we have ppl so we need know how ma…
Name: text, Length: 14640, dtype: object
```

0.1.5 1e. Remove Punctuation

```
[ ]: df['text'].replace(regex=True, inplace=True, to_replace=r'[%s]' % string.
      ↪punctuation, value=r'')

     df['text']
```

```
[ ]: 0                          virginamerica what dhepburn said
     1        virginamerica plus youve added commercials to …
     2        virginamerica i didnt today must mean i need t…
     3        virginamerica its really aggressive to blast o…
     4        virginamerica and its a really big bad thing a…
                                     …
     14635    americanair thank you we got on a different fl…
     14636    americanair leaving over minutes late flight n…
     14637    americanair please bring american airlines to …
     14638    americanair you have my money you change my fl…
     14639    americanair we have ppl so we need know how ma…
     Name: text, Length: 14640, dtype: object
```

0.1.6 1f. Featurize using Bag of Words technique

```
[ ]: processed_tweets = df['text']

     unique_words=[]
     for tweet in processed_tweets:
         for word in tweet.split():
             if word not in unique_words:
                 unique_words.append(word)


     feature_matrix = np.zeros((len(processed_tweets),len(unique_words)))
     for n,review in enumerate(processed_tweets):
         for word in review.split():
             feature_matrix[n][unique_words.index(word)]=review.count(word)

     feature_matrix
```

```
[ ]: array([[1., 1., 1., …, 0., 0., 0.],
            [1., 0., 0., …, 0., 0., 0.],
            [1., 0., 0., …, 0., 0., 0.],
            …,
            [0., 0., 0., …, 0., 0., 1.],
            [0., 0., 0., …, 0., 0., 0.],
            [0., 0., 0., …, 0., 0., 0.]])
```

0.2 2. Fit the Naive Bayes Classifier

0.2.1 2a. Mapping the airline_sentiment data

[ ]:
```
# Mapping values from -1 to 1

df['airline_sentiment'] = df['airline_sentiment'].map({'negative': -1,␣
 ↪'neutral': 0, 'positive': 1})
df['airline_sentiment'] = df['airline_sentiment'].astype('category')
df.head(10)
```

[ ]:
```
       tweet_id airline_sentiment  \
0  5.703060e+17                 0
1  5.703010e+17                 1
2  5.703010e+17                 0
3  5.703010e+17                -1
4  5.703010e+17                -1
5  5.703010e+17                -1
6  5.703010e+17                 1
7  5.703000e+17                 0
8  5.703000e+17                 1
9  5.702950e+17                 1

                                               text     tweet_created
0                   virginamerica what dhepburn said  24-02-2015 11:35
1  virginamerica plus youve added commercials to …  24-02-2015 11:15
2  virginamerica i didnt today must mean i need t…  24-02-2015 11:15
3  virginamerica its really aggressive to blast o…  24-02-2015 11:15
4  virginamerica and its a really big bad thing a…  24-02-2015 11:14
5  virginamerica seriously would pay a flight for…  24-02-2015 11:14
6  virginamerica yes nearly every time i fly vx t…  24-02-2015 11:13
7  virginamerica really missed a prime opportunit…  24-02-2015 11:12
8           virginamerica well i didntbut now i do d  24-02-2015 11:11
9  virginamerica it was amazing and arrived an ho…  24-02-2015 10:53
```

0.2.2 2b. Splitting the data into training and test sets and fitting the model

[ ]:
```
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score

X = feature_matrix
```

```
y = df['airline_sentiment']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25,␣
 ↪random_state=1)


nb = MultinomialNB()
nb.fit(X_train, y_train)
```

```
[ ]: MultinomialNB()
```

0.3 3. Report the Accuracy

```
[ ]: nb_y_pred = nb.predict(X_test)

     print("Accuracy of the NB model is: ", accuracy_score(y_test, nb_y_pred))
```

```
Accuracy of the NB model is:  0.7565573770491804
```

0.4 4. Compare the results from variants of other NB models

```
[ ]: from sklearn.naive_bayes import GaussianNB

     gb = GaussianNB()
     gb.fit(X_train, y_train)

     gb.score(X_test, y_test)

     gb_y_pred = gb.predict(X_test)

     print("Accuracy of the GaussianNB model is: ", accuracy_score(y_test,␣
      ↪gb_y_pred))
```

```
Accuracy of the GaussianNB model is:  0.4931693989071038
```

0.5 5. Write your own observations

- The classes have been identified using `value_counts()` method, and their distribution through `normalize` method.
- We move ahead with Multinomial Naive Bayes model instead of Bernoulli Naive Bayes model, as the data is not binary.
- Regex has been used for preprocessing the data for the preparation of the feature matrix.
- The feature matrix has been constructed using the BoW Technique.
- The classes have been mapped from `positive`, `neutral` and `negative` to `1`, `0` and `-1` respectively.

- The training and test data have been split into training and test sets using the `train_test_split` method.(ratio 3:1)
- The model has been fit using Multinomial NB, and the accuracy is reported as `0.756`
- The model has been also fit by using Gaussian NB, and the accuracy is reported as `0.493`. Thus the dataset doesn't follow a normal distribution, as the accuracy of the model is not as good as Multinomial NB.