Lab 6: Spam Filtering using Multinominal NB

Name: Ezhilarasan C

Roll NO: 225229151

In [66]:
```
# Step :1
```

In [1]:
```
import pandas as pd
```

In [10]:
```
dr=pd.read_csv("SMSSpamCollection.csv",encoding ='Windows-1252')
```

In [11]:
```
dr
```

Out[11]:

	label	text	Unnamed: 2	Unnamed: 3	Unnamed: 4
0	ham	Go until jurong point, crazy.. Available only ...	NaN	NaN	NaN
1	ham	Ok lar... Joking wif u oni...	NaN	NaN	NaN
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	NaN	NaN	NaN
3	ham	U dun say so early hor... U c already then say...	NaN	NaN	NaN
4	ham	Nah I don't think he goes to usf, he lives aro...	NaN	NaN	NaN
...	...	...	...	...	...
5567	spam	This is the 2nd time we have tried 2 contact u...	NaN	NaN	NaN
5568	ham	Will Ì_ b going to esplanade fr home?	NaN	NaN	NaN
5569	ham	Pity, * was in mood for that. So...any other s...	NaN	NaN	NaN
5570	ham	The guy did some bitching but I acted like i'd...	NaN	NaN	NaN
5571	ham	Rofl. Its true to its name	NaN	NaN	NaN

5572 rows × 5 columns

In [12]:
```
dr.head()
```

Out[12]:

	label	text	Unnamed: 2	Unnamed: 3	Unnamed: 4
0	ham	Go until jurong point, crazy.. Available only ...	NaN	NaN	NaN
1	ham	Ok lar... Joking wif u oni...	NaN	NaN	NaN
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	NaN	NaN	NaN
3	ham	U dun say so early hor... U c already then say...	NaN	NaN	NaN
4	ham	Nah I don't think he goes to usf, he lives aro...	NaN	NaN	NaN

In [13]:
```
dr.tail()
```

Out[13]:

	label	text	Unnamed: 2	Unnamed: 3	Unnamed: 4
5567	spam	This is the 2nd time we have tried 2 contact u...	NaN	NaN	NaN
5568	ham	Will ì_ b going to esplanade fr home?	NaN	NaN	NaN
5569	ham	Pity, * was in mood for that. So...any other s...	NaN	NaN	NaN
5570	ham	The guy did some bitching but I acted like i'd...	NaN	NaN	NaN
5571	ham	Rofl. Its true to its name	NaN	NaN	NaN

In [14]:
```
dr.shape
```

Out[14]:
```
(5572, 5)
```

In [15]:
```
dr.size
```

Out[15]:
```
27860
```

In [16]:
```
dr.columns
```

Out[16]:
```
Index(['label', 'text', 'Unnamed: 2', 'Unnamed: 3', 'Unnamed: 4'], dtype='object')
```

In [19]:
```
dr.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5572 entries, 0 to 5571
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   label       5572 non-null   object
 1   text        5572 non-null   object
 2   Unnamed: 2  50 non-null     object
 3   Unnamed: 3  12 non-null     object
 4   Unnamed: 4  6 non-null      object
dtypes: object(5)
memory usage: 217.8+ KB
```

In [20]:
```
dr.isnull()
```

Out[20]:

	label	text	Unnamed: 2	Unnamed: 3	Unnamed: 4
0	False	False	True	True	True
1	False	False	True	True	True
2	False	False	True	True	True
3	False	False	True	True	True
4	False	False	True	True	True
...	...	...	...	...	...
5567	False	False	True	True	True
5568	False	False	True	True	True
5569	False	False	True	True	True
5570	False	False	True	True	True
5571	False	False	True	True	True

5572 rows × 5 columns

In [27]:
```
dr.drop(['Unnamed: 2','Unnamed: 3','Unnamed: 4'],axis=1,inplace=True)
```

In [28]:
```
dr.head()
```

Out[28]:

	label	text
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

In [65]:
```
# Step:2
```

In [29]:
```
dr['text'].value_counts().sum()
```

Out[29]: 5572

In [64]:
```
# Step:3
```

In [30]:
```
dr.groupby(['label']).count()
```

Out[30]:

	text
label	
ham	4825
spam	747

In [63]:
```
# Step:4
```

In [39]:
```
y = dr['label']
X = dr['text']
```

In [40]:
```
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=42
```

In [62]:
```
# Step: 5
```

In [41]:
```
from nltk.corpus import stopwords
def process_text(msg):
    punctuations = '''!()-[]:;"\,<>./?@#${}%^_~*&'''
    nopunc = [char for char in msg if char not in punctuations]
    nopunc = ''.join(nopunc)
    return [word for word in nopunc.split()
            if word.lower() not in stopwords.words('english')]
```

In [42]:
```
import nltk
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\1mscdsa51\AppData\Roaming\nltk_data...
[nltk_data]   Unzipping corpora\stopwords.zip.
```

Out[42]:
```
True
```

In [61]:
```
# Step :6
```

In [43]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
df1 = TfidfVectorizer(use_idf=True,
analyzer = process_text,
ngram_range=(1,3),
min_df=1,
stop_words = 'english')
df1
```

Out[43]:
```
TfidfVectorizer(analyzer=<function process_text at 0x000001A2AE4B3430>,
                ngram_range=(1, 3), stop_words='english')
```

In [45]:
```
a = df1.fit_transform(X_train)
a1 = df1.transform(X_test)
```

In [60]:
```
# Step:7
```

In [46]:
```
from sklearn.naive_bayes import MultinomialNB
cl = MultinomialNB()
cl.fit(a,y_train)
```

Out[46]:
```
MultinomialNB()
```

In [59]:
```
# Step :8
```

In [47]:
```
y_pred = cl.predict(a1)
y_pred
```

Out[47]:
```
array(['ham', 'ham', 'ham', ..., 'ham', 'ham', 'spam'], dtype='<U4')
```

In [58]:
```
# Step :9
```

In [48]:
```
from sklearn.metrics import confusion_matrix
confusion_matrix(y_test,y_pred)
```

Out[48]:
```
array([[965,   0],
       [ 39, 111]], dtype=int64)
```

In [49]:
```
from sklearn.metrics import classification_report
print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

         ham       0.96      1.00      0.98       965
        spam       1.00      0.74      0.85       150

    accuracy                           0.97      1115
   macro avg       0.98      0.87      0.92      1115
weighted avg       0.97      0.97      0.96      1115
```

In [57]:
```
# Step 10
```

In [50]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
df2 = TfidfVectorizer(use_idf=True,
analyzer = process_text,
ngram_range=(1,2),
min_df=1,
stop_words = 'english')
df2
```

Out[50]:
```
TfidfVectorizer(analyzer=<function process_text at 0x000001A2AE4B3430>,
                ngram_range=(1, 2), stop_words='english')
```

In [51]:
```
b = df2.fit_transform(X_train)
b1= df2.transform(X_test)
```

In [52]:
```
from sklearn.naive_bayes import MultinomialNB
cl = MultinomialNB()
cl.fit(b,y_train)
```

Out[52]:
```
MultinomialNB()
```

In [54]:
```
y1_pred = cl.predict(b1)
y1_pred
```

Out[54]:
```
array(['ham', 'ham', 'ham', ..., 'ham', 'ham', 'spam'], dtype='<U4')
```

In [55]:
```
confusion_matrix(y_test,y1_pred)
```

Out[55]:
```
array([[965,   0],
       [ 39, 111]], dtype=int64)
```

In [56]:
```
print(classification_report(y_test,y1_pred))
```

```
              precision    recall  f1-score   support

         ham       0.96      1.00      0.98       965
        spam       1.00      0.74      0.85       150

    accuracy                           0.97      1115
   macro avg       0.98      0.87      0.92      1115
weighted avg       0.97      0.97      0.96      1115
```

In [ ]:
```

```