

# newsgroupstextclassification

January 31, 2024

## 0.1 Importing modules

```
[1]: import numpy as np
import pandas as pd
import os
```

```
/usr/local/lib/python3.6/importlib/_bootstrap.py:219: RuntimeWarning:
numpy.dtype size changed, may indicate binary incompatibility. Expected 96, got
88
```

```
    return f(*args, **kwargs)
/usr/local/lib/python3.6/importlib/_bootstrap.py:219: RuntimeWarning:
numpy.dtype size changed, may indicate binary incompatibility. Expected 96, got
88
```

```
    return f(*args, **kwargs)
```

## 0.2 Listing folders

```
[22]: folders=sorted(os.listdir(os.path.join(DATA_DIR))) # os.listdir gives a list of
↳all files in this path
folders
```

```
[22]: ['alt.atheism',
'comp.graphics',
'comp.os.ms-windows.misc',
'comp.sys.ibm.pc.hardware',
'comp.sys.mac.hardware',
'comp.windows.x',
'misc.forsale',
'rec.autos',
'rec.motorcycles',
'rec.sport.baseball',
'rec.sport.hockey',
'sci.crypt',
'sci.electronics',
'sci.med',
'sci.space',
'soc.religion.christian',
'talk.politics.guns',
```

```
[16]: folders[5]
```

```
[18]: DATA_DIR='20_newsgroups'
```

2

## > Building Vocab

```
[75]: vocab={}
      # Creating a dictionary of words and their frequency
      for i in range(len(data)): # For each key(newsgroup)
          for doc in data[folders[i]]: # For each document corresponding to
          ↪key(newsgroup)
              for word in doc.split(): # For each word in that document
                  if word.lower() not in stopWords and len(word.lower()) >= 5:
                      if word.lower() not in vocab:
                          vocab[word.lower()]=1
                      else:
                          vocab[word.lower()]+=1
      len(vocab)
```

[75]: 390170

```
[55]: # Sort the dictionary based on frequency of each 'possible' vocabulary word
      import operator
      sorted_vocab=sorted(vocab.items(),key=operator.itemgetter(1),reverse=True)
```

### 0.4.1 Building final feature list from vocab

```
[57]: # Choosing top 2000 vocab words as features
      feature_list=[]
      for key in sorted_vocab:
          feature_list.append(key[0])
      feature_list=feature_list[0:2000] # K = 2000 (number of words in vocab)
```

### 0.4.2 Transforming data into X and Y

```
[140]: Y=[] # list of newsgroups
       for i in range(len(data)):
           for doc in data[folders[i]]:
               Y.append(folders[i])
       Y=np.array(Y)
```

```
[94]: type(data[folders[1]])
```

[94]: list

```
[133]: # Each row : one doc and each column : one word from feature_list
       # Columns headers will be the names of features
       df = pd.DataFrame(columns = feature_list)

       for folder in folders:
           # Insert each file as a new row
```

```

for file in os.listdir(os.path.join(DATA_DIR, folder)):
    # Add a new row for every file
    df.loc[len(df)] = np.zeros(len(feature_list))
    with open(os.path.join(DATA_DIR, folder, file), encoding='latin-1') as f:
        opened_file = f
        for word in opened_file.read().split():
            if word.lower() in feature_list:
                df[word.lower()][len(df)-1] += 1
        #df[current_column][current_row]
df

```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

`comp.graphics`

[illegible]

`comp.graphics`

[illegible]

`comp.graphics`

`comp.graphics`

[illegible]



`comp.graphics`

`comp.graphics`

[illegible]

[illegible]

[illegible]

`comp.graphics`

[illegible]

[illegible]



`comp.graphics`

[illegible]

[illegible]

[illegible]

```
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.graphics  
comp.os.ms-windows.misc  
comp.os.ms-windows.misc  
comp.os.ms-windows.misc  
comp.os.ms-windows.misc
```

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

`comp.sys.ibm.pc.hardware`

[illegible]

`comp.sys.mac.hardware`





















[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

`comp.sys.mac.hardware`

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

`comp.windows.x`

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

`misc.forsale`

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

`misc.forsale`







`misc.forsale`

`misc.forsale`

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]















































































































[illegible]

[illegible]



rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.baseball  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey  
rec.sport.hockey

[illegible]

[illegible]

[illegible]

[illegible]





















[illegible]

[illegible]

[illegible]

[illegible]

[illegible]





[illegible]





















[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]











[illegible]

[illegible]



[illegible]





[illegible]

[illegible]

[illegible]



















[illegible]











[illegible]

[illegible]



[illegible]



[illegible]























[illegible]

[illegible]

[illegible]















[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]





[illegible]

[illegible]

[illegible][illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]



[illegible]



[illegible]



















[illegible]















[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]





[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]





























[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[133]:	going	something	computer	system	might	please	reply-to:	using	\
0	1.0	0.0	0.0	2.0	0.0	0.0	0.0	0.0	
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	1.0	0.0	0.0	1.0	0.0	0.0	
3	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	

4	0.0	2.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	2.0	0.0	0.0	0.0
10	0.0	2.0	0.0	1.0	1.0	0.0	0.0	0.0
11	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0
12	1.0	1.0	2.0	0.0	3.0	0.0	0.0	1.0
13	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
14	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
15	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
16	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
17	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	3.0	1.0	0.0	0.0	1.0
19	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0
20	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
21	0.0	2.0	0.0	0.0	0.0	0.0	0.0	0.0
22	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
24	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
25	0.0	0.0	1.0	0.0	0.0	0.0	1.0	0.0
26	0.0	0.0	1.0	0.0	1.0	1.0	0.0	0.0
27	0.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0
28	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
29	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...
19967	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19968	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
19969	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0
19970	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
19971	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19972	0.0	1.0	0.0	0.0	0.0	1.0	0.0	0.0
19973	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19974	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
19975	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19976	0.0	0.0	0.0	0.0	1.0	0.0	0.0	2.0
19977	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19978	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
19979	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
19980	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
19981	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19982	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19983	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19984	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19985	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
19986	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0

19987	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19988	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19989	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19990	0.0	0.0	1.0	0.0	0.0	0.0	1.0	0.0
19991	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19992	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19993	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19994	1.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
19995	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
19996	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0

	never	can't	...	homosexuals	director	data.	argic	back,	\
0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
1	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
3	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
4	0.0	0.0	...	0.0	0.0	0.0	0.0	1.0	
5	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
6	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
7	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
8	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
9	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
10	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
11	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
12	1.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
13	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
14	1.0	2.0	...	0.0	0.0	0.0	0.0	0.0	
15	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
16	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
17	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
18	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
19	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
20	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
21	1.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
22	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
23	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
24	0.0	1.0	...	0.0	0.0	0.0	0.0	0.0	
25	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
26	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
27	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
28	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
29	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
...	...	...	...	...	...	...	...	...	...
19967	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
19968	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
19969	1.0	0.0	...	0.0	0.0	0.0	0.0	0.0	
19970	1.0	0.0	...	0.0	0.0	0.0	0.0	0.0	

19971	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19972	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19973	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19974	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19975	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19976	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19977	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19978	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19979	1.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19980	2.0	1.0	...	0.0	0.0	0.0	0.0	0.0
19981	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19982	1.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19983	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19984	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19985	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19986	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19987	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19988	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19989	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19990	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19991	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19992	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19993	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19994	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19995	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
19996	1.0	0.0	...	0.0	0.0	0.0	0.0	0.0

	assumed	sure.	universal	impact	plastic
0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0
8	0.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	0.0
10	1.0	0.0	0.0	0.0	0.0
11	0.0	0.0	0.0	0.0	0.0
12	0.0	0.0	0.0	0.0	0.0
13	0.0	0.0	0.0	0.0	0.0
14	0.0	0.0	0.0	0.0	0.0
15	0.0	0.0	0.0	0.0	0.0
16	0.0	1.0	0.0	0.0	0.0
17	0.0	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	0.0	0.0

19	0.0	0.0	0.0	0.0	0.0
20	0.0	0.0	0.0	0.0	0.0
21	0.0	0.0	0.0	0.0	0.0
22	0.0	0.0	1.0	0.0	0.0
23	0.0	0.0	0.0	0.0	0.0
24	0.0	0.0	0.0	0.0	0.0
25	0.0	0.0	0.0	0.0	0.0
26	0.0	0.0	0.0	0.0	0.0
27	0.0	0.0	0.0	0.0	0.0
28	0.0	0.0	0.0	0.0	0.0
29	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...
19967	0.0	0.0	0.0	0.0	0.0
19968	0.0	0.0	0.0	0.0	0.0
19969	0.0	0.0	0.0	0.0	0.0
19970	0.0	0.0	0.0	0.0	0.0
19971	0.0	0.0	0.0	1.0	0.0
19972	0.0	0.0	0.0	0.0	0.0
19973	0.0	0.0	0.0	0.0	0.0
19974	0.0	0.0	0.0	0.0	0.0
19975	0.0	0.0	0.0	0.0	0.0
19976	0.0	0.0	0.0	0.0	0.0
19977	0.0	0.0	0.0	0.0	0.0
19978	0.0	0.0	0.0	0.0	0.0
19979	0.0	0.0	0.0	0.0	0.0
19980	0.0	0.0	0.0	0.0	0.0
19981	0.0	0.0	0.0	0.0	0.0
19982	0.0	0.0	0.0	0.0	0.0
19983	0.0	0.0	0.0	0.0	0.0
19984	0.0	0.0	0.0	0.0	0.0
19985	0.0	0.0	0.0	0.0	0.0
19986	0.0	0.0	0.0	0.0	0.0
19987	0.0	0.0	0.0	0.0	0.0
19988	0.0	0.0	0.0	0.0	0.0
19989	0.0	0.0	0.0	0.0	0.0
19990	0.0	0.0	0.0	0.0	0.0
19991	0.0	0.0	0.0	0.0	0.0
19992	0.0	0.0	0.0	0.0	0.0
19993	0.0	0.0	0.0	0.0	0.0
19994	0.0	0.0	0.0	0.0	0.0
19995	0.0	0.0	0.0	0.0	0.0
19996	0.0	0.0	0.0	0.0	0.0

[19997 rows x 2000 columns]

```
[134]: X=df.values
```



```
[135]: X
```

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

### 0.4.3 Splitting X and Y into training and testing data

```
[192]: from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(X,Y,random_state=0,test_size=0.
↪25)
```

### 0.4.4 Using the inbuilt Multinomial Naive Bayes

```
[193]: from sklearn.naive_bayes import MultinomialNB
clf=MultinomialNB()
clf.fit(x_train,y_train)
```

```
[193]: MultinomialNB(alpha=1.0, class_prior=None, fit_prior=True)
```

```
[194]: clf.score(x_test,y_test)
```

```
[194]: 0.8476
```

### 0.4.5 Implementing Multinomial Naive Bayes from scratch

```
[195]: def fit(x_train,y_train):
        result={}
        result["total_data"]=len(y_train)
        class_labels=set(y_train)
        for current_label in class_labels:
            result[current_label]={}
            current_rows=(y_train==current_label)
            x_train_current=x_train[current_rows]
            y_train_current=y_train[current_rows]
            total_words=0
            for i in range(len(feature_list)):
                result[current_label][feature_list[i]]=x_train_current[:,i].sum()
                total_words+=x_train_current[:,i].sum()
            result[current_label]["total_count"]=total_words
        return result
```

```
[196]: def probability(x,dictionary,current_class):
        output=np.log(dictionary[current_class]["total_count"])-np.
        ↪log(dictionary["total_data"])
        for i in range(len(feature_list)):
            current_word_count=dictionary[current_class][feature_list[i]]+1
            ↪
        ↪total_word_count=dictionary[current_class]["total_count"]+len(feature_list)
            current_word_probability=np.log(current_word_count)-np.
        ↪log(total_word_count)
            for j in range(int(x[i])): # if the frequency of word in test data
        ↪point is zero then we wont consider it.
                output+=current_word_probability
        return output
```

```
[211]: def predictSingleClass(x,dictionary):
        best_class=-1000
        best_prob=-1000
        firstRun=True
        possible_classes=dictionary.keys()
        for current_class in possible_classes:
            if current_class=="total_data":
                continue
            current_class_probability=probability(x,dictionary,current_class)
            if(firstRun==True or current_class_probability>best_prob):
                best_class=current_class
                best_prob=current_class_probability
            firstRun=False
        return best_class
```

```
[212]: def predict(X_test,dictionary):
        Y_pred=[]
        num = 0
        for x in X_test:
            Y_pred.append(predictSingleClass(x,dictionary))
        return Y_pred
```

```
[213]: dictionary=fit(x_train,y_train)
```

```
[214]: y_pred=predict(x_test,dictionary)
```

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

```
[ [201  2  0  0  0  0  0  0  0  0  1  0  1  4  2  1  0  3
    5 64]
  [ 0 202 13  1  2  5  2  3  0  0  1  4  4  2  9  0  2  1
```

```

0 0]
[ 0 12 204 5 3 8 3 1 1 0 0 1 3 0 0 0 0 0
2 0]
[ 0 6 12 207 8 3 10 1 0 0 0 1 3 0 1 0 1 0
0 1]
[ 1 6 1 8 217 0 6 2 0 0 1 0 4 0 1 0 0 0
0 0]
[ 1 7 4 2 0 215 2 3 1 0 0 2 0 1 0 0 1 0
0 0]
[ 0 1 2 7 4 1 220 5 4 0 0 0 7 4 1 0 2 1
3 0]
[ 1 3 1 4 1 0 2 228 7 1 1 0 6 1 1 0 3 0
6 1]
[ 1 3 0 1 0 1 1 9 263 0 0 0 2 4 0 0 0 2
7 2]
[ 0 1 1 0 0 0 0 0 0 244 1 0 1 1 2 0 2 0
3 0]
[ 0 0 0 0 0 1 2 0 2 2 224 0 0 2 2 0 0 0
0 0]
[ 1 1 1 0 0 1 0 1 1 0 0 212 1 1 0 0 3 3
4 0]
[ 1 3 6 5 1 3 6 4 1 0 0 4 210 4 2 0 0 0
1 1]
[ 3 3 2 0 0 0 2 1 0 0 0 0 1 223 2 0 3 3
5 1]
[ 0 3 1 0 0 2 1 3 1 1 0 0 0 1 216 0 0 0
3 2]
[ 2 0 1 0 0 0 1 1 0 0 0 0 0 1 0 250 0 2
1 11]
[ 0 0 0 0 0 0 0 5 1 0 0 3 0 2 0 0 207 6
47 23]
[ 2 0 0 0 0 0 2 0 1 0 0 2 0 0 0 1 0 242
9 3]
[ 2 0 0 0 0 0 1 2 1 0 2 4 1 1 5 0 12 18
146 21]
[ 17 0 0 0 0 0 0 0 0 0 0 0 0 4 2 0 13 0
17 106]]

```

	precision	recall	f1-score	support
alt.atheism	0.86	0.71	0.78	284
comp.graphics	0.80	0.80	0.80	251
comp.os.ms-windows.misc	0.82	0.84	0.83	243
comp.sys.ibm.pc.hardware	0.86	0.81	0.84	254
comp.sys.mac.hardware	0.92	0.88	0.90	247
comp.windows.x	0.90	0.90	0.90	239
misc.forsale	0.84	0.84	0.84	262
rec.autos	0.85	0.85	0.85	267
rec.motorcycles	0.93	0.89	0.91	296

rec.sport.baseball	0.98	0.95	0.97	256
rec.sport.hockey	0.97	0.95	0.96	235
sci.crypt	0.91	0.92	0.92	230
sci.electronics	0.86	0.83	0.85	252
sci.med	0.87	0.90	0.88	249
sci.space	0.88	0.92	0.90	234
soc.religion.christian	0.99	0.93	0.96	270
talk.politics.guns	0.83	0.70	0.76	294
talk.politics.mideast	0.86	0.92	0.89	262
talk.politics.misc	0.56	0.68	0.61	216
talk.religion.misc	0.45	0.67	0.54	159
avg / total	0.86	0.85	0.85	5000

[ ]:

[ ]:

[ ]:

[ ]:

[ ]: