Gaurav Bishnoi
CSCI 5832: Natural Language Processing
Programming Assignment 4
Date: April 21, 2016

How to run Code:
Run the python file "bishnoi_gaurav_semantic.py". Place the test files "posi.txt" and "nega.txt" in the same folder with python file. "posi.txt" contains positive opinion words and "nega.txt" file contain negative opinion words.

Additional Features Implemented:
1. Data of positive and negative opinions by Hu and Liu
2. Implemented the data given in NLTK Book by removing 'stopwords' and punctuations.

Classifiers Used:
1. Naïve Bayes
2. Maximum Entropy

Final Code:
Classifier: Naïve Bayes
Data Set: Data of positive and negative opinions by Hu and Liu

Different Codes and their results:

I first tried the code given in section 1.3, Chapter 6, NLTK Book. It was giving accuracy in the region of 0.65 for 10-fold. Then I removed some unnecessary/irrelevant features from list of maximum-probability words by using 'stopwords' function of 'nltk.corpus'. I have added this code between comment symbols, titled "Extra Code 1".
Results: Accuracy = 0.68

```
C:\Python34\python.exe E:/Programming
/python/untitled/try.py
Accuracy for 1 fold: 0.71
Accuracy for 2 fold: 0.67
Accuracy for 3 fold: 0.68
Accuracy for 4 fold: 0.715
Accuracy for 5 fold: 0.715
Accuracy for 6 fold: 0.705
Accuracy for 7 fold: 0.64
Accuracy for 8 fold: 0.685
Accuracy for 9 fold: 0.665
Accuracy for 10 fold: 0.655
0.6839999999999999

Process finished with exit code 0
```

Then I tried Maximum Entropy Classifier with 200 most frequent words from NLTK movie review corpus. Because there is lot of calculation involved in this and using 2000 word set was giving errors (citing system performance). Also, it takes 100 iterations for 1 fold, so I run the code for just 1 fold, i.e. training set of first 100 reviews (random) and test set of remaining 1900 reviews. I have added this code between comment symbols, titled "Extra Code 2".

Results: Accuracy for first fold: 0.58

```
C:\Python34\python.exe E:/Programming/python/untitled/try.py
  ==> Training (100 iterations)

      Iteration    Log Likelihood    Accuracy

      ---------------------------------------
             1          -0.69315          0.496
             2          -0.69253          0.506
             3          -0.69197          0.532
             4          -0.69141          0.562
             5          -0.69086          0.581
             6          -0.69031          0.594
             7          -0.68977          0.607
             8          -0.68922          0.611
             9          -0.68869          0.614
            10          -0.68815          0.619
            11          -0.68762
```

16	0.621		
17	12	-0.68709	
	0.618		
18	13	-0.68656	
	0.616		
19	14	-0.68604	
	0.616		
20	15	-0.68552	
	0.615		
21	16	-0.68500	
	0.614		
22	17	-0.68448	
	0.619		
23	18	-0.68397	
	0.619		
24	19	-0.68346	
	0.621		
25	20	-0.68296	
	0.620		
26	21	-0.68245	
	0.620		
27	22	-0.68196	
	0.619		
28	23	-0.68146	
	0.619		
29	24	-0.68096	
	0.619		
30	25	-0.68047	
	0.619		
31	26	-0.67999	

…

…

```
 91          0.624
 92              87          -0.65514
             0.624
 93              88          -0.65480
             0.624
 94              89          -0.65446
             0.625
 95              90          -0.65413
             0.624
 96              91          -0.65380
             0.624
 97              92          -0.65347
             0.624
 98              93          -0.65314
             0.624
 99              94          -0.65282
             0.624
100              95          -0.65249
             0.624
101              96          -0.65217
             0.624
102              97          -0.65185
             0.624
103              98          -0.65153
             0.624
104              99          -0.65122
             0.624
105           Final          -0.65090
             0.624
106 Accuracy for 1 fold: 0.58
```

At last, I used 'Data of positive and negative opinions by Hu and Liu' for better results. It has separate list of positive and negative words. It increased the accuracy to a great extent.
Results: 0.82

```
C:\Python34\python.exe E:/Programming
/python/untitled/try.py
6789
Accuracy for 1 fold: 0.825
Accuracy for 2 fold: 0.785
Accuracy for 3 fold: 0.86
Accuracy for 4 fold: 0.825
Accuracy for 5 fold: 0.855
Accuracy for 6 fold: 0.85
Accuracy for 7 fold: 0.85
Accuracy for 8 fold: 0.77
Accuracy for 9 fold: 0.76
Accuracy for 10 fold: 0.82
Net Accuracy: 0.82

Process finished with exit code 0
```