Homework Assignment 2

CS 585 Natural Language Processing
Spring Semester, 2019

Dinesh Karamchandani

CWID: A20407484

Problem Description

The aim in this assignment was to implement Naive Bayes algorithm for text classification for a given IMDB like dataset of movie-reviews. The output to be produced included the code used along with a brief writeup for the metrics and evaluations done.

Data

The IMDB (like) data consisted of 25000 files of each positive and negative reviews for movies. These 25000 files from each class were further divided into test and train parts equally. Thus 25K files consisting of positive and negative reviews for each test and train.

Metrics

With the goal of implementing of implementing Naive Bayes on the given IMDB dataset, we had to implement functionality of Naive Bayes into 2 main functions *PredictLabel and PredictProb*. Further to validate the results we have functions implementing the different metrics of Accuracy, Recall and Precision, the metrics values for which are shown below. Also smoothing hyperparameter (ALPHA) was implemented to view how it affected these metrics.

Below are the metrics and their values for varying ALPHA values:

Alpha	Accuracy	Precision	Recall
0.1	0.8192	0.808537	0.83648
0.5	0.82896	0.814662	0.85168
1	0.83216	0.815597	0.8584
5	0.83196	0.81258	0.86296
10	0.83032	0.8112	0.86104

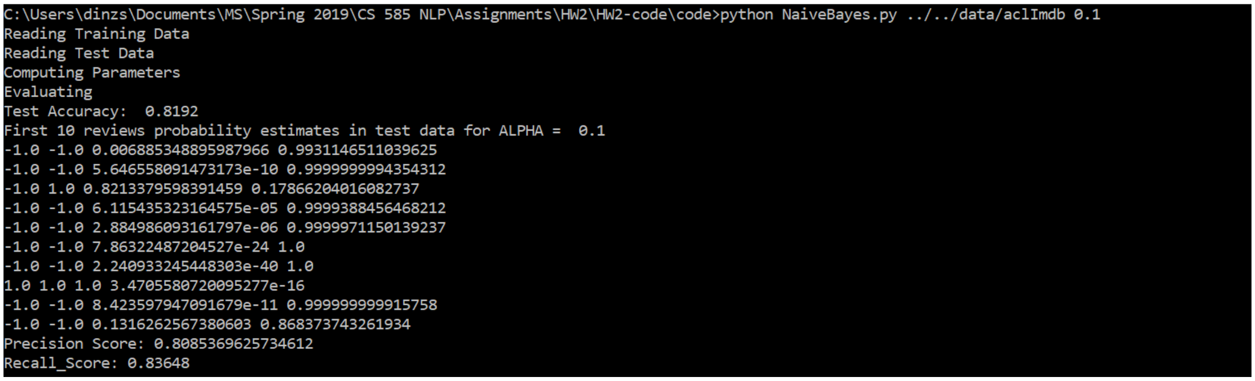
First 10 reviews probabilities

As part of the assignment we have also showcased the negative and positive probabilities of the 1st ten reviews in the training dataset, wherein the values are displayed in the below format. These probabilities are in the range [0-1] and

Actual Label, Predicted Label, Predicted positive probability, Predicted negative probability

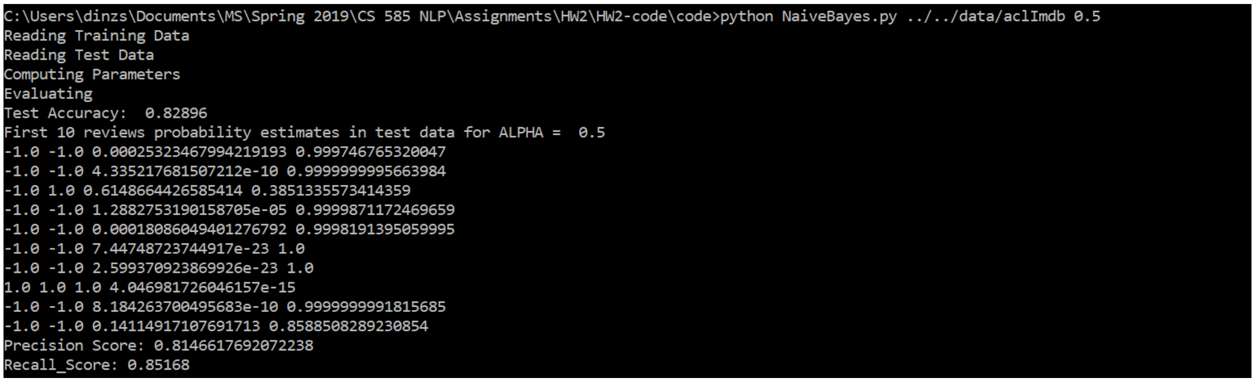
The screenshots shown below showcase the same for different values of ALPHA

1) ALPHA = 0.1



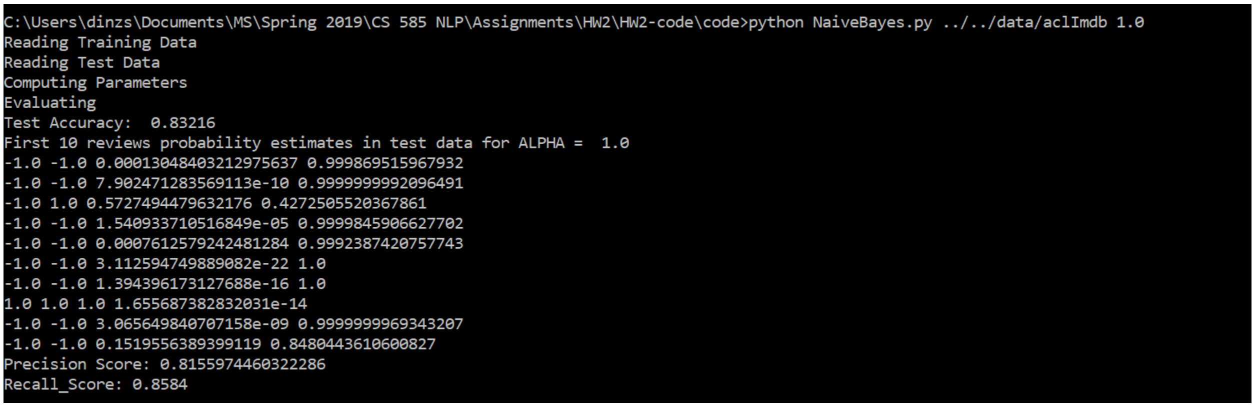
```
C:\Users\dinzs\Documents\MS\Spring 2019\CS 585 NLP\Assignments\HW2\HW2-code\code>python NaiveBayes.py ../../data/aclImdb 0.1
Reading Training Data
Reading Test Data
Computing Parameters
Evaluating
Test Accuracy:  0.8192
First 10 reviews probability estimates in test data for ALPHA =  0.1
-1.0 -1.0 0.006885348895987966 0.9931146511039625
-1.0 -1.0 5.646558091473173e-10 0.9999999994354312
-1.0 1.0 0.8213379598391459 0.17866204016082737
-1.0 -1.0 6.115435323164575e-05 0.9999388456468212
-1.0 -1.0 2.884986093161797e-06 0.9999971150139237
-1.0 -1.0 7.86322487204527e-24 1.0
-1.0 -1.0 2.240933245448303e-40 1.0
1.0 1.0 1.0 3.4705580720095277e-16
-1.0 -1.0 8.423597947091679e-11 0.999999999915758
-1.0 -1.0 0.1316262567380603 0.868373743261934
Precision Score: 0.8085369625734612
Recall_Score: 0.83648
```

2) ALPHA = 0.5



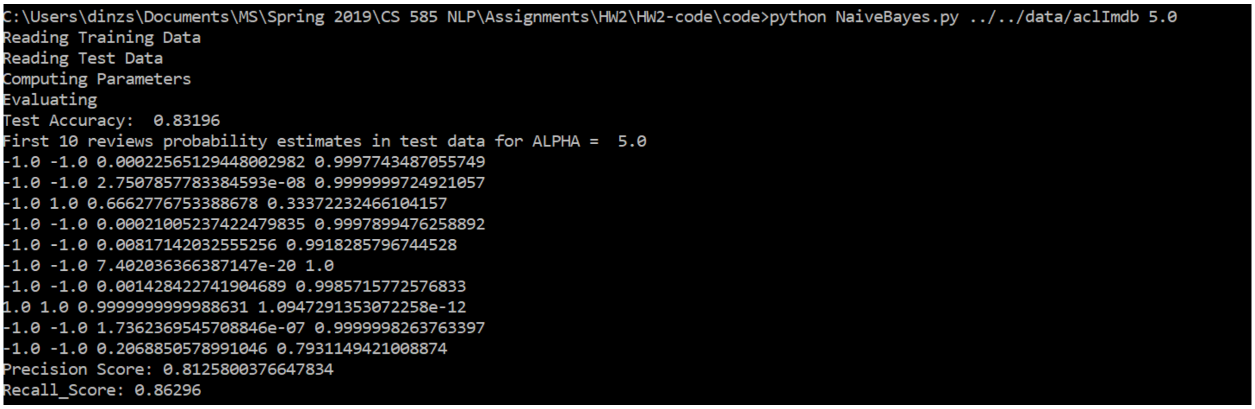
```
C:\Users\dinzs\Documents\MS\Spring 2019\CS 585 NLP\Assignments\HW2\HW2-code\code>python NaiveBayes.py ../../data/aclImdb 0.5
Reading Training Data
Reading Test Data
Computing Parameters
Evaluating
Test Accuracy:  0.82896
First 10 reviews probability estimates in test data for ALPHA =  0.5
-1.0 -1.0 0.00025323467994219193 0.999746765320047
-1.0 -1.0 4.335217681507212e-10 0.9999999995663984
-1.0 1.0 0.6148664426585414 0.3851335573414359
-1.0 -1.0 1.2882753190158705e-05 0.9999871172469659
-1.0 -1.0 0.00018086049401276792 0.9998191395059995
-1.0 -1.0 7.44748723744917e-23 1.0
-1.0 -1.0 2.599370923869926e-23 1.0
1.0 1.0 1.0 4.046981726046157e-15
-1.0 -1.0 8.184263700495683e-10 0.9999999991815685
-1.0 -1.0 0.14114917107691713 0.8588508289230854
Precision Score: 0.8146617692072238
Recall_Score: 0.85168
```

3) ALPHA = 1.0



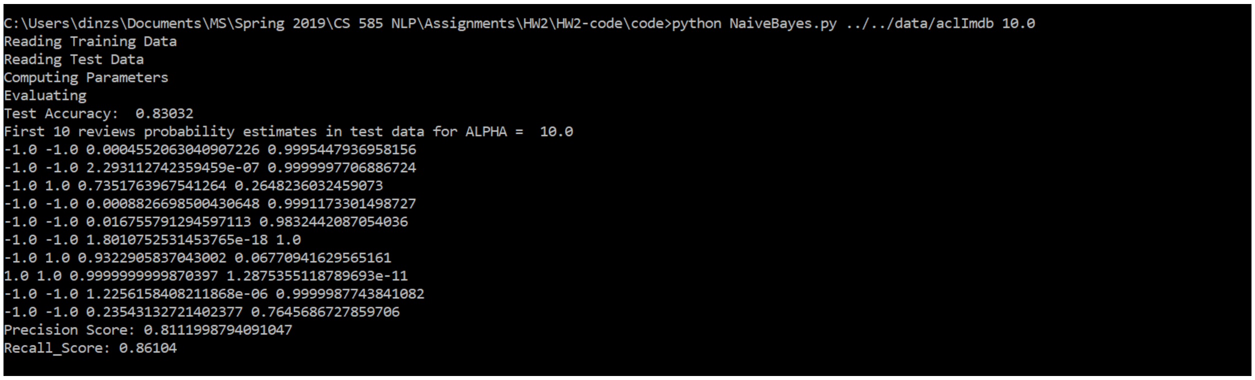
```
C:\Users\dinzs\Documents\MS\Spring 2019\CS 585 NLP\Assignments\HW2\HW2-code\code>python NaiveBayes.py ../../data/aclImdb 1.0
Reading Training Data
Reading Test Data
Computing Parameters
Evaluating
Test Accuracy:  0.83216
First 10 reviews probability estimates in test data for ALPHA =  1.0
-1.0 -1.0 0.00013048403212975637 0.999869515967932
-1.0 -1.0 7.902471283569113e-10 0.9999999992096491
-1.0 1.0 0.5727494479632176 0.4272505520367861
-1.0 -1.0 1.540933710516849e-05 0.9999845906627702
-1.0 -1.0 0.0007612579242481284 0.9992387420757743
-1.0 -1.0 3.112594749889082e-22 1.0
-1.0 -1.0 1.394396173127688e-16 1.0
1.0 1.0 1.0 1.655687382832031e-14
-1.0 -1.0 3.065649840707158e-09 0.9999999969343207
-1.0 -1.0 0.1519556389399119 0.8480443610600827
Precision Score: 0.8155974460322286
Recall_Score: 0.8584
```

4) ALPHA = 5.0



```
C:\Users\dinzs\Documents\MS\Spring 2019\CS 585 NLP\Assignments\HW2\HW2-code\code>python NaiveBayes.py ../../data/aclImdb 5.0
Reading Training Data
Reading Test Data
Computing Parameters
Evaluating
Test Accuracy:  0.83196
First 10 reviews probability estimates in test data for ALPHA =  5.0
-1.0 -1.0 0.00022565129448002982 0.9997743487055749
-1.0 -1.0 2.7507857783384593e-08 0.9999999724921057
-1.0 1.0 0.6662776753388678 0.33372232466104157
-1.0 -1.0 0.00021005237422479835 0.9997899476258892
-1.0 -1.0 0.00817142032555256 0.9918285796744528
-1.0 -1.0 7.402036366387147e-20 1.0
-1.0 -1.0 0.001428422741904689 0.9985715772576833
1.0 1.0 0.9999999999988631 1.0947291353072258e-12
-1.0 -1.0 1.7362369545708846e-07 0.9999998263763397
-1.0 -1.0 0.2068850578991046 0.7931149421008874
Precision Score: 0.8125800376647834
Recall_Score: 0.86296
```

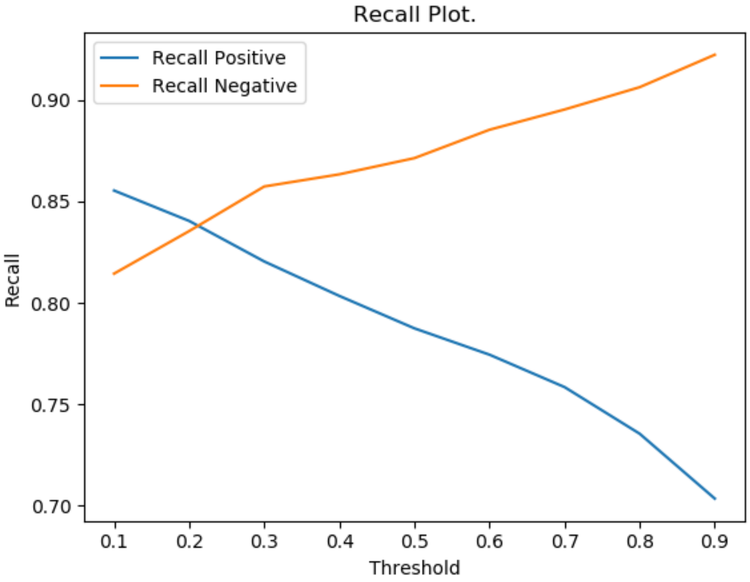
5) ALPHA = 10.0

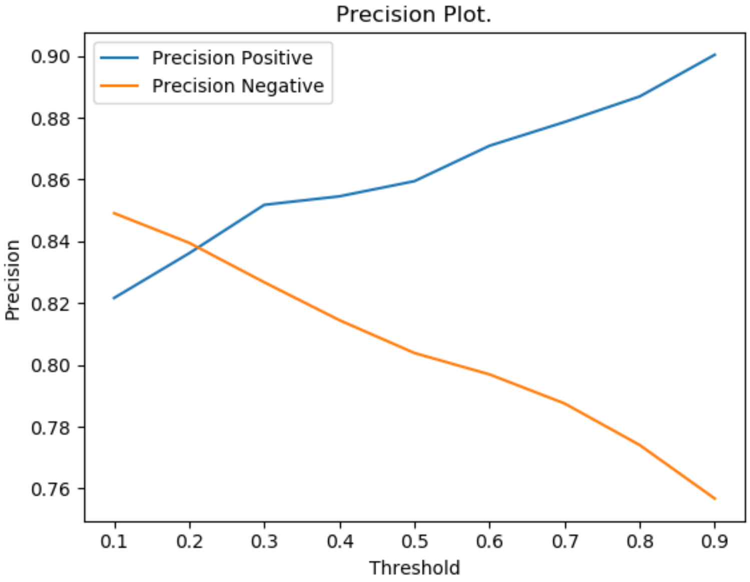


```
C:\Users\dinzs\Documents\MS\Spring 2019\CS 585 NLP\Assignments\HW2\HW2-code\code>python NaiveBayes.py ../../data/aclImdb 10.0
Reading Training Data
Reading Test Data
Computing Parameters
Evaluating
Test Accuracy:  0.83032
First 10 reviews probability estimates in test data for ALPHA =  10.0
-1.0 -1.0 0.0004552063040907226 0.9995447936958156
-1.0 -1.0 2.293112742359459e-07 0.9999997706886724
-1.0 1.0 0.7351763967541264 0.2648236032459073
-1.0 -1.0 0.0008826698500430648 0.9991173301498727
-1.0 -1.0 0.016755791294597113 0.9832442087054036
-1.0 -1.0 1.8010752531453765e-18 1.0
-1.0 1.0 0.9322905837043002 0.06770941629565161
1.0 1.0 0.9999999999870397 1.2875355118789693e-11
-1.0 -1.0 1.2256158408211868e-06 0.9999987743841082
-1.0 -1.0 0.23543132721402377 0.7645686727859706
Precision Score: 0.8111998794091047
Recall_Score: 0.86104
```

Precision and Recall Graphs

After changing the PredictProb function to include the threshold limit, the recall and precision values were calculated and plotted on graphs as showcased below.



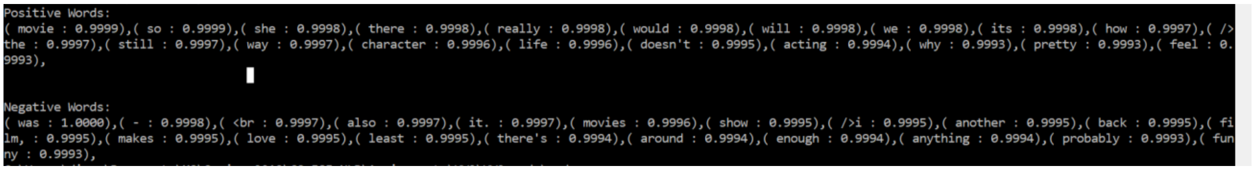


Relationship observed:

From the above figures we can see the relationship of Recall and Precision with threshold, where in as the threshold values increases, for positive classes the precision also increases and recall decreases. Whereas in the negative classes the precision decreases and recall increases as the threshold limit increases.

Features

Lastly, we have also evaluated the probabilities of all the positive and negative words in the vocabulary such that they are sorted in descending order by their weight. Below are the Top 20 Positive and Negative words based on that for ALPHA = 1.0



```
Positive Words:
( movie : 0.9999),( so : 0.9999),( she : 0.9998),( there : 0.9998),( really : 0.9998),( would : 0.9998),( will : 0.9998),( we : 0.9998),( its : 0.9998),( how : 0.9997),( />
the : 0.9997),( still : 0.9997),( way : 0.9997),( character : 0.9996),( life : 0.9996),( doesn't : 0.9995),( acting : 0.9994),( why : 0.9993),( pretty : 0.9993),( feel : 0.
9993),


Negative Words:
( was : 1.0000),( - : 0.9998),( <br : 0.9997),( also : 0.9997),( it. : 0.9997),( movies : 0.9996),( show : 0.9995),( />i : 0.9995),( another : 0.9995),( back : 0.9995),( fi
lm, : 0.9995),( makes : 0.9995),( love : 0.9995),( least : 0.9995),( there's : 0.9994),( around : 0.9994),( enough : 0.9994),( anything : 0.9994),( probably : 0.9993),( fun
ny : 0.9993),
```

Analysis :

From my understanding the words like movie, character, acting, pretty are coming as the most positive words indicate that in the given sample of positive reviews apart from there being many instances they are portraying a positive factor. This may be to due to sentences like “a good movie”, “strong character” , “splendid acting”, “pretty actress” or so on. Also, words like how, we, will ,feel are also present due to examples like “how realistic was the look and feel of the animations”,etc.

For the negative words we find words like was, also, another, film are present in the top list due to abundance of examples having sentences like “also we felt … did better in another film”. The words love, funny,enough are also present which means that they were mostly used in sarcasm for the negative reviews.