TEXT CLASSIFICATION

In this problem, we are asked to implement the Naïve Bayes algorithm to predict the rating of the movie reviews taken from IMDB data set.

(a)

Here I will implement Naïve Bayes algorithm to classify each of the reviews into one of the given categories.

I have used Laplace smoothing to avoid any zero probabilities. (c = 1). Also I have computed the probabilities in the log scale to avoid underflow issues.

In the reviews I have also done some pre-processing. Like: making all words in lowercase so as to treat Like and like same. Also I have removed the whitespaces and punctuation marks.

After implementing the Naïve Bayes method for categorizing the reviews (refer code file for the implementation), accuracies obtained are as follows:

Results:

Training set accuracy = 68.44%

Test set accuracy = 38.476%

(b)

In this I have computed the test set accuracy by randomly assigning one of the rating categories as the target class for each of the reviews. (Random prediction)

Test set accuracy for random prediction: 12.60%

I have also computed the test set accuracy by assigning each review the target class which occurs most in the training set. (Majority prediction)

Test set accuracy for majority prediction: 20.088%

Observations:
The test set accuracy has dropped in both the cases (Random & majority prediction) compared to (a).
Majority prediction accuracy is higher than the Random prediction accuracy.

Our algorithm gives around 19% improvement in accuracy over majority prediction.

(c)

I have drawn the confusion matrix for the test set of (a).

Confusion Matrix:

Actual classes

Predicted classes	[1]	[2]	[3]	[4]	[7]	[8]	[9]	[10]
[1]	[[4274.	1588.	1363.	1038.	399.	423.	332.	796.]
[2]	[87.	50.	56.	48.	10.	13.	6.	12.]
[3]	[154.	183.	229.	209.	80.	64.	23.	43.]
[4]	[257.	273.	491.	673.	262.	167.	94.	104.]
[7]	[36.	55.	127.	224.	424.	310.	154.	179.]
[8]	[61.	54.	120.	229.	520.	720.	468.	561.]
[9]	[20.	6.	13.	26.	73.	132.	122.	177.]
[10]	[133.	93.	142.	188.	539.	1021.	1145.	3127.]]

Observations:

Category 1 has the highest value of the diagonal entry of the confusion matrix. It means that class 1 has been predicted most correctly. This class has highest predicted accuracy.

In the confusion matrix, more is the value of diagonal entries, more better the model is. High value of diagonal entry means high prediction accuracy.

We can also see that class 10 has the 2nd highest prediction accuracy.

One more observation is that, classes 1-4 has been predicted more towards class 1 (1^{st} four columns of 1^{st} row) and classes 7-10 have been predicted more towards class 10. (last four columns of last row)

(d)

In this section, I have done some pre-processing on the input reviews.
I have removed the STOP WORDS from the reviews. Also I have done STEMMING before feeding the reviews to the Naïve Bayes algorithm.

After doing these pre-processing, a new model has been created.

Training set accuracy: 67.98%

Test set accuracy: 38.448%

Observations:
The test set accuracy is more or less same for both the cases.

- When data is having stopwords
- When stop words are removed and data is stemmed

So there is marginal change in the accuracy for the test set as well as for the training set, when stop words are removed and stemming is done.

(e)

Here I have to do feature engineering. Means I have to extract the features which improve the overall accuracy of the test set reviews.
Applied Bigram technique to create the features.

Train set accuracy: 99.50%

Test set accuracy: 39.336%

So Bigram has improved the test accuracy by almost 1%.

I also tried removing least frequent words from the review and balancing the unbalanced data, but that did not improve the test accuracy. The accuracy for them was coming out to be around 30%.

Also tried, making good review words list and bad review words list. Then on prediction, I computed the review words occurrence in the good word list and bad word list. If occurrence of good words is more than bad words for the review predicted to be in class 1-4(using NB), then I assign that review any random class 7 to 10 and vice-versa. But accuracy for that also coming out to be 20.52%