Tiange Wang
3717659

CS165A MP1 Report

********* IMPORTANT: PLEASE RUN THE CODE WITH PYTHON3 *********

This Project is completed on python version 3.7.2 with external libraries numpy and scipy. The code is separated into 2 major portions, which are extraction and analysis. In the extraction portion, first it takes in 2 input training files which contain positive and negative training data. A list of common punctuation marks and stop words are provided at the beginning to provide additional accuracy and reduce dictionary size for faster runtime. For the actual extraction procedure, the loop iterates through every review in both the positive training set and the negative training set. 2 dictionaries are constructed for each category. Whenever a new word is encountered, it is put into the dictionary of its respective review class with a value of 1. If a word already exists in the dictionary, increment the value by 1. In addition to constructing the positive and negative dictionary, an additional step to count up how many reviews contain a certain word for computing the TF-IDF score in the analysis step. Reviews are separated by the special
</br> symbol, which is used in my code to differentiate different reviews. For every word, every character is converted into lower case. If any punctuation marks are detected, then the word is simply discarded. Also, for every review, a new dictionary is initialized to store all unique words in this review. Once a review has been parsed, every unique word in this dictionary is added to another dictionary which counts the number of reviews for a certain word. After all words from the positive and negative training data are extracted, another dictionary is created to store every word in all reviews, including both positive and negative. After all extractions are complete, the code moves onto the second portion, which is analysis using the extracted features. There are three total classifiers, which divides the code into 3 portions. The first portion is the gaussian naïve bayes classifier using the BoW feature. For this classifier, I constructed a vector for every review which contains the frequency of each word and sum up the result into a new vector. Once all of them have been iterated through, I divide this vector by the number of reviews that I have iterated through to get the mean vector. As for the variance vector, I iterate through the entire training set again but using each vector to subtract the mean vector and square the result. The whole set's sum is also put into a new vector and divided by the number of reviews at the end, which is the variance vector. Once

both vectors have been obtained, the public reviews can be iterated through by using the mean and variance obtained from the previous 2 vectors computed. For each review, their probabilities of word being in either positive class or negative class are computed through the normal distribution and multiplied together to reach the end result. Whichever is higher determines what this review is classified to. For the gaussian naïve bayes classifier with TF-IDF feature, the idea is the same as the BoW feature, except the difference being a tf-idf score is calculated instead of simply counting the frequency. The tf score is computed by parsing through each review and the idf score is computed by using the word count when doing the initial extraction. The last portion of the analysis part is the multinomial naïve bayes classifier. The classifier is don't through the formula (occurrence + 1) / (total words in class + dictionary length). The +1 is the Laplace smoothing used to deal with words that don't exist in the training set. Each word has a probability for their respective class which is ended up multiplied together to acquire the final result. Whichever class's probability is higher determines what the review is classified as.

Result of the 3 classifiers:

```
---------------
Final Result:
---------------
Gaussian Naive Bayes classifier using BoW feature:
-Total Prediction: 6000
-Correct Prediction: 3462
-Accuracy: 0.577
---------------
Gaussian Naive Bayes classifier using TF-IDF feature:
-Total Prediction: 6000
-Correct Prediction: 3138
-Accuracy: 0.523
---------------
Multinomial Naive Bayes classifier using BoW feature:
-Total Prediction: 6000
-Correct Prediction: 4443
-Accuracy: 0.7405
```