Project One

CSE 408: Group 11

Names:
Carter Kwon: 1208955809
Daniel Davidson: 1205519532
Bryce Pedroza: 1208763448
Jason Pajas: 1207462980
Due: 9/18/18

Table of Contents

Introduction:

The goal of this project was to successfully categorize documents using various algorithms that perform text classification including k-NN (k-nearest neighbors) and bag of words and to perform sentiment analysis on text files in order to obtain an overall positive or negative connotation from the text within reviews. Each method we used gave us different accuracy and each had issues that affected the overall accuracy of the classification process which are described in detail below.

Implementation:

For this assignment we chose to break the work up into sections where Carter and Bryce worked on k-NN classification and testing in part one, Jason worked on part two of the assignment implementing Text Sentiment Analysis, and Daniel worked on documentation. We then discussed the results and some of the issues that we had encountered such as low accuracy of the results and ways to increase the reliability of the algorithms such as changing an algorithms threshold values or nearest neighbor k values.

Results:

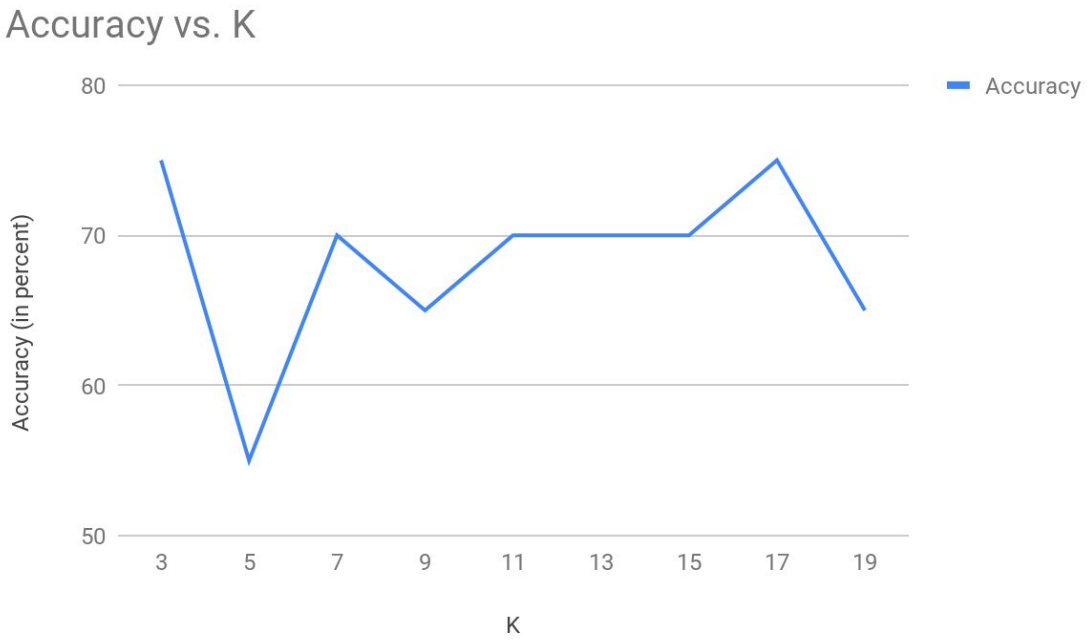
Question One:

One complication we ran into when classifying using the bag of words method was setting a threshold level that would remove words that held key meaning in the context of the sentence only because they did not appear enough to meet the threshold, for example an overall positive review might contain the word “good” twice, but with a threshold of three it would be removed from the calculation affecting the accuracy of the classification and producing a false negative. We did find however that our sentiment analysis produced overall accurate results. k-NN was a fairly reliable algorithm that worked best with low

and high k values, it was able to produce an accuracy of 75% but did not work as well with mid ranged k values.

Question Two:

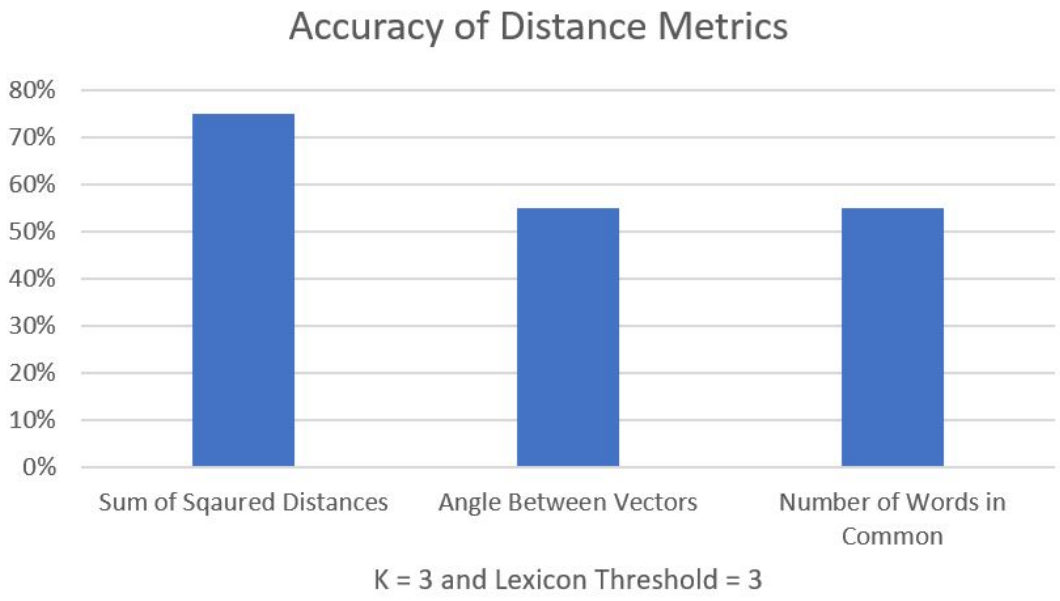
We achieved our highest level of accuracy using the sum of squares distances (ssd) method. Although, the other methods were not far behind with an average accuracy of around 50%. Attached below is a graph showing accuracy vs. different K values using ssd and a threshold of 3. The highest accuracy came from using K = 3, and K = 17 with an accuracy of 75%.



Question Three:

There were three different distance-metrics that were observed for this project: sum of squared distances (SSD), the angle between vectors and number of words in common. At face value, it would appear that number of words in common would be a valuable metric to take into

consideration for correctly classifying positive and negative text articles. There are two baskets or words, positive and negative, and the article will be classified in relation to the number of words that are present from the article in each basket of words. If there are more positive words in an article, it should be classified as positive, and vice versa. However, upon further examination of our results, it was determined that number of words in common was not a particularly effective method at article classification. Its accuracy was found to be 55% with the constraints mentioned in the table below.



There are many reasons why the number of words in common is not accurate at classifying articles as positive and negative, but the main reason can be the organization and overall structure of an article is not being properly taken into account. Even though words are classified as positive or negative, the overall structure of the article is not considered with this method. The method that was found to be more successful however, was the sum of squared distances. Our

testing found the SSD to produce results that better matched what was expected with 20% more accuracy. The constraints in the bar chart above led to a 75% accuracy when classifying positive vs. negative articles with SSD. This is observed because SSD takes the organization of the article more into account than only measuring the number of words in common. Results that have similar weights in the same positions will cause the SSD to be smaller, which will yield a better result because the difference of the two points at the same index is taken into account. This can be observed in the equation given in lecture the notes.

$$\sum_{i=1}^{n}(X_i - Y_i)^2$$

Question Four:

On the text sentiment analysis portion of the project our highest positive score on a negative review was on 03.txt with a sentiment score of 3.12. The lowest negative score on a positive review was on 18.txt with a sentiment score of -4.58. The set of words for the negative review 03.txt that confused the sentiment analysis system was {philanthropic, time, create, make, available, simply, sake, better, such, will, relevant, about, practice, character, incorrectly, nonsensical, visually, interesting, point, acting, personal, brilliant, live such, lofty, will}. All these words are positive and when added up, provided a high positive sentiment on a negative review. As for the positive review, the set of words in 18.txt that confused the sentiment analysis system was {unlike, by, do, not, have, other cross, small, phone}. Just like the positive words in the negative review, all these words are negative and when added up, provided a high negative sentiment on a positive review.

Conclusion:

In conclusion we discovered that each method of solving classification comes with its own drawbacks and that classifying language is a difficult science. Accuracy ratings are not always perfect, but in general they can be relied upon to give reasonable confidence in how an article of text should be classified. Sentiment analysis is one of the easier classifications to solve due to its binary nature of being strictly a positive or negative sentiment, but general document classification can be difficult for most other types of classification. Going forward, a more elegant solution would be to develop a sentiment analysis algorithm that would take sentence structure into account when assigning a classification for a piece of text. As seen in our analysis of question four, there were articles of text that were incorrectly classified because the the algorithm would see negative words and assign a more negative sentiment to the article, even though the article should have been classified as positive. By figuring out a way consider the organization of the words, the classification of articles would be much more accurate.