Programming Assignment Report

Student: **Vagdevi Junnuri**
Student#: **014926366**

Rank - 26
Accuracy - 0.8058

Approach :
Before passing the training and test datasets through the classifier, I have preprocessed the data. The preprocessing steps involved:

Dataset cleaning:

1. Removing HTML tags, numbers, emails, IDs. This is done using regular expressions.
2. Removed any emojis which may effect the meaning of the review.
3. Converted to lowercase
4. Pruning stop words
5. Splitted individual words
6. Removed stop words by using the nltk package
7. Performed Lemmatization

TF_IDF:

1. Converted the above cleaned trained and test datasets into sparse matrices

Normalization:

Performed normalization on the sparse matrices

Similarity measure:

Used cosine similarity as a similarity measure to find the L2 norm between the normalized sparse matrices of training and test datasets

Finding k nearest neighbors:

The k nearest neighbors are found using the calculated similarities vectors and a declared k value

kNN classifier:

Used kNN classifier to find the sentiments in the test dataset.

Used the training dataset sentiments and the k nearest neighbors to classify the test dataset sentiments.

Methodology:

Data cleaning:

1. Since the data contains information about IMDB movie reviews which is obtained from online, I have removed any unwanted HTML tags as a process of cleaning the dataset.
2. The dataset might contain any other meta information like numbers,email Ids,Urls etc which is highly unwanted in order to do sentiment analysis and might even effect the accuracy is not removed. So I have removed it using regular expressions.
3. Any kind of emojis which may affect the meaning of the reviews have also been pruned.

4. Since the reviews need to be purely words, all the other unnecessary characters have been removed and the obtained result of words have been converted to lower case and split into individual words.
5. If stop words are present, they are removed using the nltk package.
6. Lemmatization is performed for faster processing and decrease noise.Used WordNetLemmatizer from nltk package.

TF_IDF: The training and test datasets are converted into sparse matrices in order to save storage and computing times.

Normalization: Performed data normalization for effective data processing.

Cosine similarity: I have opted to use cosine similarity as a distance measure since the dataset contains text data and since I am working with sparse vectors.Also cosine similarity was fast when computing similarities between training and test data compared to others.

kNN classifier: I have used kNN classifier in order to do sentiment analysis on the movie reviews data.

After calculating the cosine similarity between the training and test data , the similarity vectors are passed to the kNN classifier for classifying the sentiments.
I have ranked the top values based on their neighbours.
Iterated over the training sentiment labels to check for sentiment that has +1.
Maintained the count of the +1(positive) sentiment label in a variable.
If this count variable is greater than half of the k value, the label is predicted as +1(positive) otherwise negative.

I have used kNN classifier since it has high accuracy rate and faster processing time. Since the given training dataset was labeled, using kNN classifier seemed justified.

Associated parameter:
Choosing the optimal value of k is very important in kNN classifier.
Tried a vast range of k values to obtain better accuracy.