**Miner2 username** – bot\_olof

**Mason userid** - aghadge

**Best public score** – 0.69

**Best Rank** – 203

**K Nearest Neighbor Classifier Report**

**Introduction**

Implemented KNN classifier to classify move reviews as either positive or negative. Value of k = 81 with bag of words model and cosine similarity measure to calculate similarity between two reviews. Each review represented as a dictionary of indexes of words in the bag of words and frequency of words in that review. Cross validation with 10 folds is used to verify the model.

**Approach**

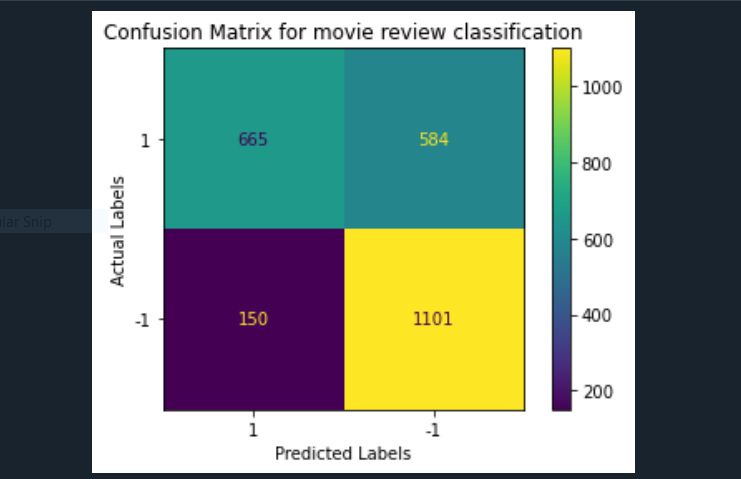
1. **Preprocessing**

I split the training review data file into list of lists of string e.g. [[review1], [review2] …].

Splitting was done keeping ‘#EOF’ as the delimiter. Then the class labels of reviews in the training data were separated and stored in a list called label. Each training review was then converted to lower case and was then tokenized, and punctuation marks were removed. The tokenized words obtained earlier were then traversed on and then stemmed. Along with stemming a bag of words from training review data was also created in this step. Also, each review record (both training and testing) were converted into dictionaries of the form **{index of word in bag of words: frequency of word in this particular review}**.

Bag of words implementation consists of a dictionary which has the preprocessed word as key and value will be the index of word in this bag of words. A dictionary named ‘bag\_freq’ is maintained to store the frequencies of words in the bag of words.

Words with frequencies below a certain threshold are removed from the bag of words representation.

1. **Classification**
   1. Each test review record was compared with all training review records with cosine similarity measure.
   2. If a word from the test record existed in the corresponding train review record, then their word frequencies in that review was used to compute the numerator of the cosine similarity. For the denominator, summation of multiplication of frequencies of words in test reviews is multiplied with summation of multiplication of frequencies of words in training reviews and then taken square root of.
   3. The output of this similarity between each test review and all training reviews is stored in a dictionary. Keys in each dictionary of test record represents the training review number and value represents the cosine similarity of that test record with the training record number represented by the key.
   4. The cosine similarity dictionary is then sorted in decreasing order and top ‘k’ similarities is fetched as the k nearest neighbors.
   5. The training review number stored as key in the cosine similarity dictionary allows to fetch the label of that training review.
   6. The labels of the k training reviews are summed and if this calculation is less than 0 then majority of the labels of the k training reviews is -1 and +1 if the summation is more than 0.
2. **Experiments** 
   1. Tried different values of frequency thresholds. Words having frequencies below this threshold were removed from the bag of words.
   2. Performed stemming on words and removed punctuation marks.
   3. Tried using Manhattan distance instead of cosine similarity. For Manhattan distance, k least distant training reviews to the test review were used to signify greater similarity.
   4. Implemented cross validation to verify the model for all the different techniques mentioned above along with trying different values of k nearest neighbors.
   5. Below is a confusion matrix for a single fold of 10-fold cross validator and value of k in KNN classifier as 81 and accuracy = 70.64%.
   6. 
3. **Conclusion**

Successfully studied and implemented KNN classifier using bag of words implementation approach with 69% accuracy on miner.