Classification of News Data

Vikram Ahuja(201256040)
Manas Tewari(201256031)
Asif Hussain(201225086)

WHAT IS DOCUMENT CLASSIFICATION

Document classification is a problem in various field which involves the task of assigning documents to one or multiple classes or categories.

The documents may be text music or pictures however in this project we will be dealing with only texts.

Documents may be classified according to their subject or some other attributes (such as document type, author etc).

DATASET

- The dataset we are using is the standard Twenty News Group Dataset.
- This dataset consist of 20000 articles taken from 20 newsgroup.
- Each newsgroup is stored in a different subdirectory with 1000 articles in each subdirectory.
- Each article is stored as a file consisting of text only.

OUR APPROACH

- We used tf-idf feature to give scores in the document.
- Boosted proper nouns across all documents.
- Comparing different classification algorithms like K-means, NN and 3NN, and Naive Bayes with and without boosting.
- Result calculated for unigram, bigram and trigram for Naive Bayes. Bigram and Trigram result for KNN required more computation power.
- KNN and Naive bayes coded from scratch. Libraries used for Kmeans.

CLEANING UP DATASET

- Total Unigrams -> 134341 , Bigrams -> 30809040 , Trigrams -> 4651881
- Top 5000 words were taken as stop words
- Uni/Bi/Trigrams with frequency less than 4 were removed.
- Unigrams Reduced to 35120, Bigrams to 169654 and Trigrams to 208504
- In Boosting count of frequency of Proper Nouns in each document was doubled.
- Train : Test Data = 4 : 1

NAIVE BAYES APPROACH

In bayes approach we use some likelihood of each unigram in a document with some prior probability and then the outcome is selected based on the highest posterior probability. Evidence is scaling factor ignored by us.

The formula used is

$$p(C_k|\mathbf{x}) = \frac{p(C_k)\ p(\mathbf{x}|C_k)}{p(\mathbf{x})}.$$

or in plain english

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}.$$

1 OR 3-NN APPROACH

This is a supervised approach in which k parameters(here 1 and 3) are chosen from training data through testing to have the highest accuracy and then new unknown data is classified into a group basing on the shortest cosine similarity distance from parameters.

Tested with 6 and 20 classes . Tf idf score to create vectors

Extremely sparse tf-idf matrix : nearly 0.05% filled(unigrams). Will be less for Bi/Trigrams.

Cos Similarity

Cosine Similarity (d1, d2) = Dot product(d1, d2) / ||d1|| * ||d2||

Dot product (d1,d2) = d1[0] * d2[0] + d1[1] * d2[1] * … * d1[n] * d2[n]

||d1|| = square root(d1[0]2 + d1[1]2 + ... + d1[n]2)

||d2|| = square root(d2[0]2 + d2[1]2 + ... + d2[n]2)

Weighted Approach: Adding cos similarity of same classes and using it as weights(looking at whole class instead of one).

K-MEANS(Unsupervised) APPROACH

This approach attempts to split the data set into k(20) clusters using k centroids and classify the data using these centroids producing a set of clusters. Centroids are then set to arithmetic mean of cluster and this process is repeated till centroid is stable.

Sklearn library was used for this task.

WHY THESE CLASSIFIERS?

- Naive Bayes and KNN are supervised methods and Kmeans is Unsupervised.
- Naive bayes runs well for small dataset and assumes that feature is independent.
- Use Nearest Neighbour approach instead of SVM’s and Neural Networks because it is more intuitive and works fast for smaller values of 1 and 3.
- Kmeans being unsupervised, comparison of Supervised and Unsupervised classification is done.
- We didn’t use Kernel SVM’s because vectors are very large and time taken to predict will be extremely high.

OBSERVATION (ACCURACY)

K-Means => 43%
k-Means(Boost by 1) => 51%

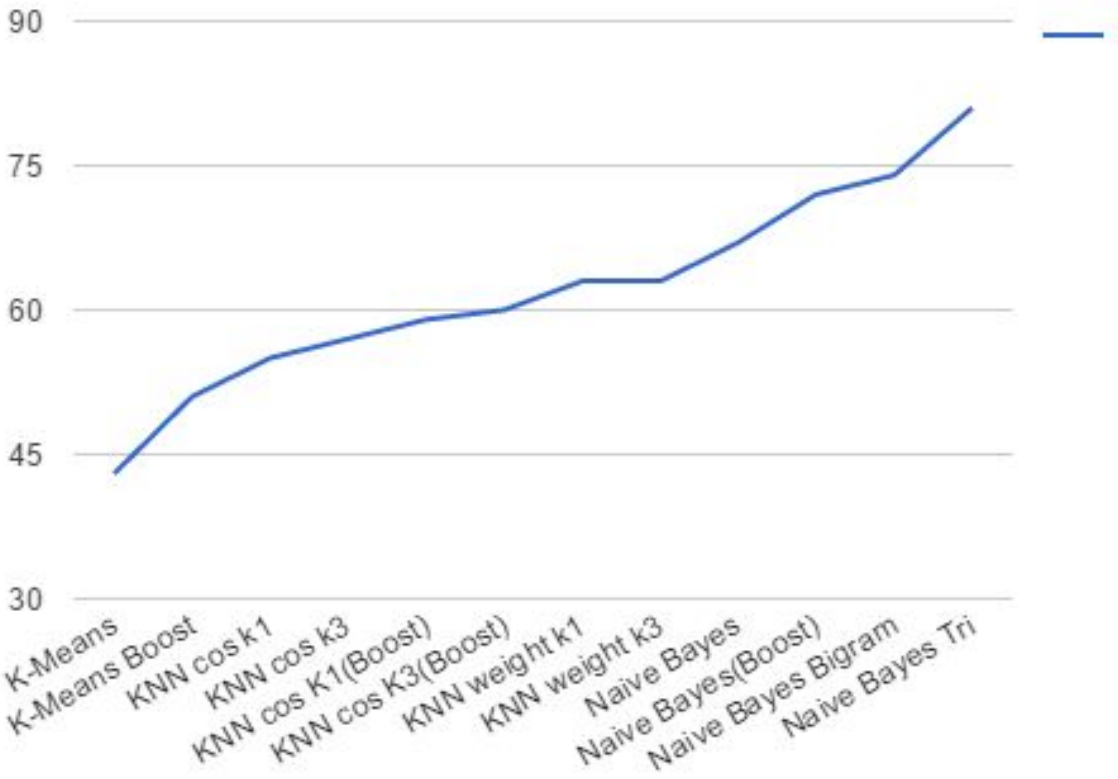
Naive Bayes Unigrams (No Boost)=>67%
Naive Bayes Unigrams (Boost by 1)=>72%
Naive Bayes Bigram=>74%
Naive Bayes Trigram=>81%

Nearest Neighbour/s(unigrams):

- Cosine similarity:
  - k(1) =>55%
  - k(3) =>57%
  - WIth Boost by 1 => 59% and 60% Respectively
- Weight

  63% in both the cases(An increase)

GRAPH


90
75
60
45
30
K-Means
K-Means Boost
KNN cos k1
KNN cos k3
KNN cos K1(Boost)
KNN cos K3(Boost)
KNN weight k1
KNN weight k3
Naive Bayes
Naive Bayes(Boost)
Naive Bayes Bigram
Naive Bayes Tri