CPS842 Fall2021
Project Report
Document Clustering System
Name: Tusaif Azmat Student#:500660278

Introduction:
For this final project, I have implemented the Document Clustering System based on the knowledge gained throughout the course. For this purpose I have use the BBC news dataset provided by http://mlg.ucd.ie/datasets/bbc.html. There are 5 classes of articles in each dataset. I used the class labels as the gold standard when measured the external evaluation metrics. I have used the inverted index program from the previous assignments to process of the documents. Then implemented the clustering algorithm and from that I use the two evaluation metrics, tightness of each cluster and the purity of clusters.

Following are the java programs implemented to create a Document Clustering System.

1- Invert.java.
2- Cluster.java.
3- Gui.java (Graphical User Interface).

Invert.java:
This program is to scan the folders in bbc and put them in the two text files which are called dictionary.txt and postings.txt. This program also creates titles.txt which stores all the titles of the documents retrieved from the files which will be used in Cluster.java when printing out the evaluation metrics information. Once the scanning is done and the required information is put into the Hash Map they need to go into, a for-loop which removes the common words in both Hash Maps, if stop word removal was turned on. Once that process is done, the contents of Hash Maps, dictionary and postings are printed on to the files **dictionary.txt** and **postings.txt**, respectively. This program also creates **titles.txt** which stores all the titles of the documents retrieved from the files which will be used in Cluster.java when printing out the evaluation metrics information.

Cluster.java:
This program calculates the similarity score of each document and creates 5 clusters which hold documents that are similar to one another. When choosing the initial centroid, the first document of every folder (i.e. business, entertainment, politics, sports, and tech) is chosen.
The first iteration calculates the similarity between the centroid with all the documents and returns the document that has the highest similarity.
Once it goes through all the documents, the final 5 sets of clusters are returned. After the first iteration is completed, the 5 documents that were chosen to be the centroids are placed back into docWeight.
The next step is to perform 5 more iteration till it converges which is done in otherIteration ().
This program also prints out the tightness of each cluster and their purity. The evaluation metrics of each cluster is calculated by initially creating a Hash Map called clusterList, that stores all the clusters that were collected through firstIteration () and otherIteration(). The tightness between each cluster is calculated in otherIteration () but it is only done after completing the 5th iteration. The distance is measured by retrieving the similarity score of each cluster which were stored in the simScore array list and is subtracted from 1. Then the distance is squared, and the sum of it is assigned to their relative cluster representing the tightness between them.

User Interface:

The program interface displays the results of the clustering system. With the help of graphical user interface user can process the initial steps to perform clustering by running the invert program which creates the three text files (title.txt, dictionary.txt and posting.txt). These files will help compute the next step in document cluster system. In the next step we compute the k-means clustering algorithm. In the compute () method, both the text files that were created previously in Invert.java are scanned to calculate the vector weights of each document. After running this program, the output is the clustered results. For each cluster, show the cluster ID, 5 terms with the highest TFIDF scores and all the documents with (ID + title) in the cluster. Please also show the actual class label beside each document. At the end, you need to show the values of the chosen metrics.