Sai Teja Chava

Set Spark home environment variable.

Versions:

Scala – 2.11

Spark – 2.3.1

Commands to run:

$SPARK_HOME/bin/spark-submit --class Task1 path_to_jar path_to_data_txt_file method to use(whether tf-idf or word count) no_of_clusters no_of_iterations

Example for Task1 with tf-idf:

$SPARK_HOME/bin/spark-submit --class Task1 ~/IdeaProjects/untitled/out/artifacts/untitled_jar/untitled.jar ~/Documents/Data_Mining_INF_553/Assignments/hw4/INF553_Assignment4/Data/yelp_reviews_clustering_small.txt T 5 20

Example for Task1 with word count:

$SPARK_HOME/bin/spark-submit --class Task1 ~/IdeaProjects/untitled/out/artifacts/untitled_jar/untitled.jar ~/Documents/Data_Mining_INF_553/Assignments/hw4/INF553_Assignment4/Data/yelp_reviews_clustering_small.txt W 5 20

Example for Task 2 K means:

$SPARK_HOME/bin/spark-submit --class Task2 ~/IdeaProjects/untitled/out/artifacts/untitled_jar/untitled.jar ~/Documents/Data_Mining_INF_553/Assignments/hw4/INF553_Assignment4/Data/yelp_reviews_clustering_small.txt K 8 20

Example for Task 2 Bisecting K-means:

$SPARK_HOME/bin/spark-submit --class Task2 ~/IdeaProjects/untitled/out/artifacts/untitled_jar/untitled.jar ~/Documents/Data_Mining_INF_553/Assignments/hw4/INF553_Assignment4/Data/yelp_reviews_clustering_small.txt B 8 20

Description of how I solved the problem/Assignment

1. For task 1, I did word clustering as spoke to TA(Prasad). Randomly selected 5 (no_of_clusters) words as cluster centers/centroids and assigned all other words to the closest centroid.
2. Once the clustering is done, all words assigned to the cluster are sorted based on their counts and top 10 words in the cluster are found.
3. I did not use built in tf-idf and calculated my own tf-idf using the following formulae

   TF = term_freq/size_of_document
   IDF = Log(total_documents / no_of_docs_containing_term)
   TF-IDF = TF * IDF

4. Once I get the centroid values, I find the top 10 words that occur in the cluster.
5. Used inbuilt hashingTf() and IDF() to calculate TF-IDF's values or feature vectors for the documents.
6. For task2, I gave the number of features to the hashingTF as 15000, since no of unique words in the documents are 14948.
7. WSSE was found for Task2 using builtin functions.