Description File

Java – 1.8
Scala – 2.11
Spark – 2.3.1

Approach

The main class name is Task1.scala in the JAR file Ashir_Alam_Clustering.jar. The algorithm reads a .txt file to an RDD and we make a term frequency, word count and document frequency RDD. We then use Word Count and TF-IDF for 2 different features cluster computation. As expected, the error for TFIDF is much less than Word count as it is normalized. We then apply the K-Means algorithm as discussed in class using Euclidean distance. It computes the clusters and we then take the top 10 elements from each cluster.

For task 2, we use the MLLIB function of K Means and Bissecting K Means algorithm. We use the TF-IDF feature for this too and supply the dense vector of TF-IDF to the KMeans and Bissecting K-Means libraries of MLLIB. We save the clusters result to a JSON file.

Command Line

Task1:
spark-submit --driver-memory 4g --class Task1 Ashir_Alam_Clustering.jar <input_path> <feature> <num_clusters> <num_iterations>

Task2:
spark-submit --driver-memory 4g --class Task2 Ashir_Alam_Clustering.jar <input_path> <algorithm> <num_clusters> <num_iterations>
Eg:

```
spark-submit --driver-memory 4g --class Task1 Ashir_Alam_Clustering.jar
/Users/ashiralam/Downloads/INF553_Assignment4/Data/yelp_reviews_clustering_small.txt T
5 20


spark-submit --driver-memory 4g --class Task2 Ashir_Alam_Clustering.jar
/Users/ashiralam/Downloads/INF553_Assignment4/Data/yelp_reviews_clustering_small.txt B
8 20
```