TASK 5

The TF – IDF task will be carried out by the readiy avaibly python code for mapper and reducer

Task5_Electronics = Load 'hdfs://cluster-assignment1-m/tfidf/spam/000000_0' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','YES_MULTILINE') AS (ID:chararray,reviewerID:chararray,helpful:chararray,overall:chararray,summary:chararray,review text:chararray);

select id,overall,helpful,reviewername,summary from dbElectronics_task4.hamSelected order by helpful desc limit 10;

Process to be repeated for each python code file (4 Mapper and 3 Reducer)

>>>hadoop jar /usr/lib/hadoop/hadoop-streaming.jar -file /home/hamza_qadeer2/pythonfiles/mapper1.py /home/hamza_qadeer2/pythonfiles/reducer1.py - mapper "python mapper1.py" -reducer "python reducer1.py" -input /hivefiles30/part-r-00000 - output /mapred/spam/output1

>>>hadoop jar /usr/lib/hadoop/hadoop-streaming.jar -file /home/hamza_qadeer2/pythonfiles/mapper2.py /home/hamza_qadeer2/pythonfiles/reducer2.py - mapper "python mapper2.py" -reducer "python reducer2.py" -input /hivefiles30/part-r-00000 - output /mapred/spam/output2

>>>hadoop jar /usr/lib/hadoop/hadoop-streaming.jar -file /home/hamza_qadeer2/pythonfiles/mapper3.py /home/hamza_qadeer2/pythonfiles/reducer3.py - mapper "python mapper3.py" -reducer "python reducer3.py" -input /hivefiles30/part-r-00000 - output /mapred/spam/output3

>>>hadoop jar /usr/lib/hadoop/hadoop-streaming.jar -file /home/hamza_qadeer2/pythonfiles/mapper4.py /home/hamza_qadeer2/mapper "python mapper4.py" " -input /hivefiles30/part-r-00000 - output /mapred/spam/output1

5.1 Using MapReduce to calculate the TF-IDF of the top 10 spam keywords for each top 10 spam accounts

>>> SELECT * FROM (SELECT id,word,tfidf, rank() over (PARTITION BY id ORDER BY tfidf DESC) as rank FROM Task_5.tfidf_spam DISTRIBUTE BY id SORT BY id desc) ranks WHERE rank < 10 ORDER BY id, rank;

5.2 Using MapReduce to calculate the TF-IDF of the top 10 ham keywords for each top 10 ham accounts

>>> SELECT * FROM (SELECT id,word,tfidf, rank() over (PARTITION BY id ORDER BY tfidf DESC) as rank FROM Task_5.tfidf_ham DISTRIBUTE BY id SORT BY id desc) hamR WHERE rank < 10 ORDER BY id, rank;