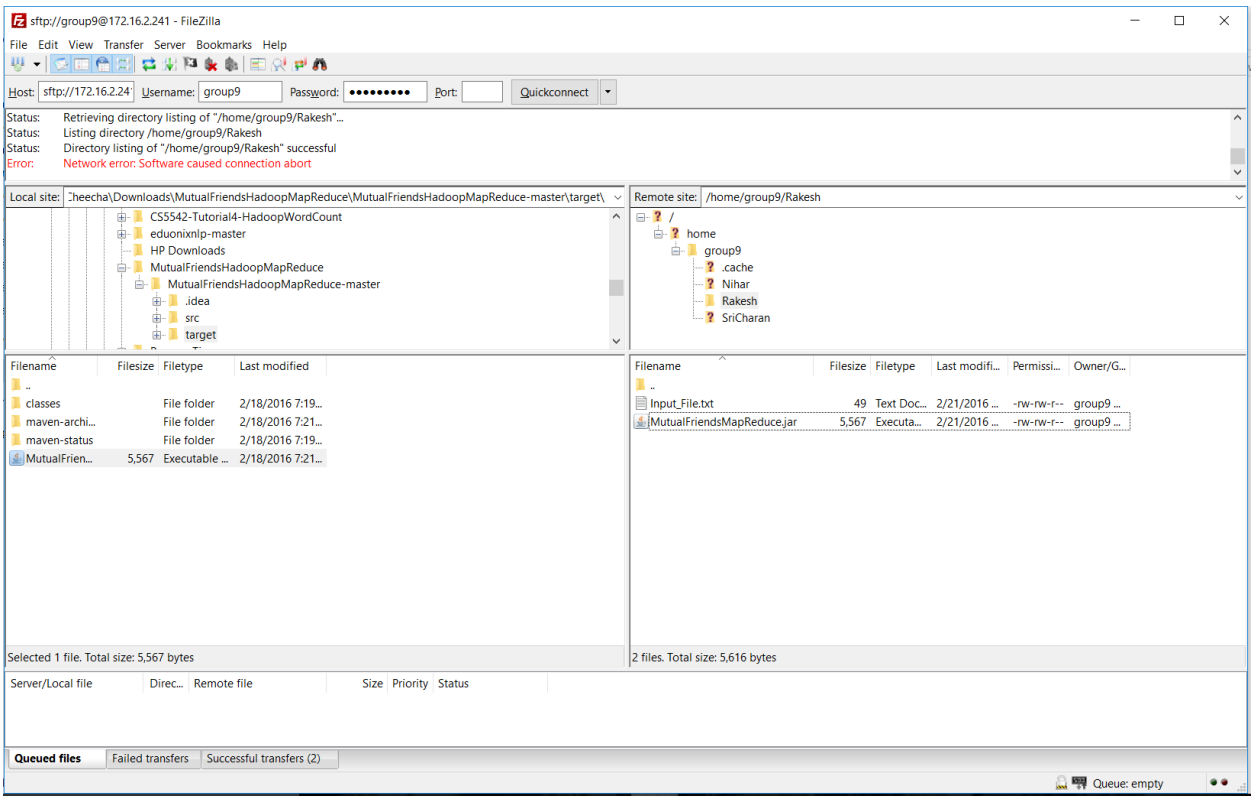
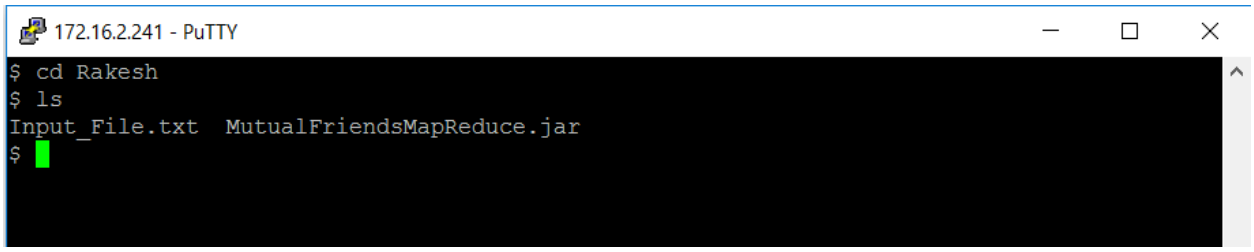
CS5542 Big Data Apps and Analytics
LAB ASSIGNMENT #4
REPORT and SCREEN SHOTS

1.HadoopMapReduce AlgorithmImplement MapReduce algorithmforfindingFacebook common friendsproblemand run theMapReduce job on Apache Hadoop.Write a report including your algorithmand result screenshots.

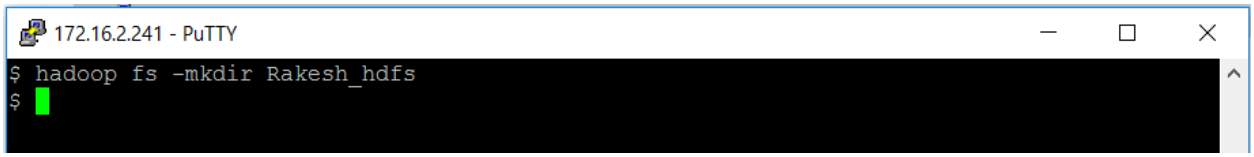
1. Push the Jar file and Input.txt file into the cloud using the FileZilla client.



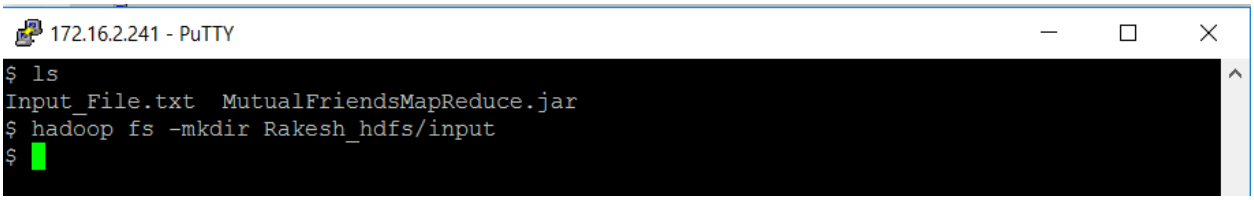
2. open the putty and run the command ls to view the files in cloud



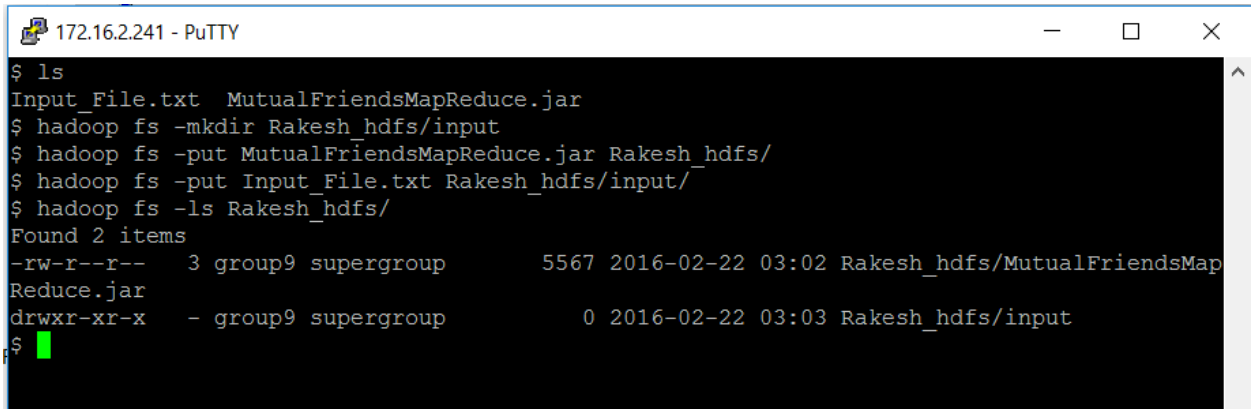
3. Create a directory in HDFS using the following command



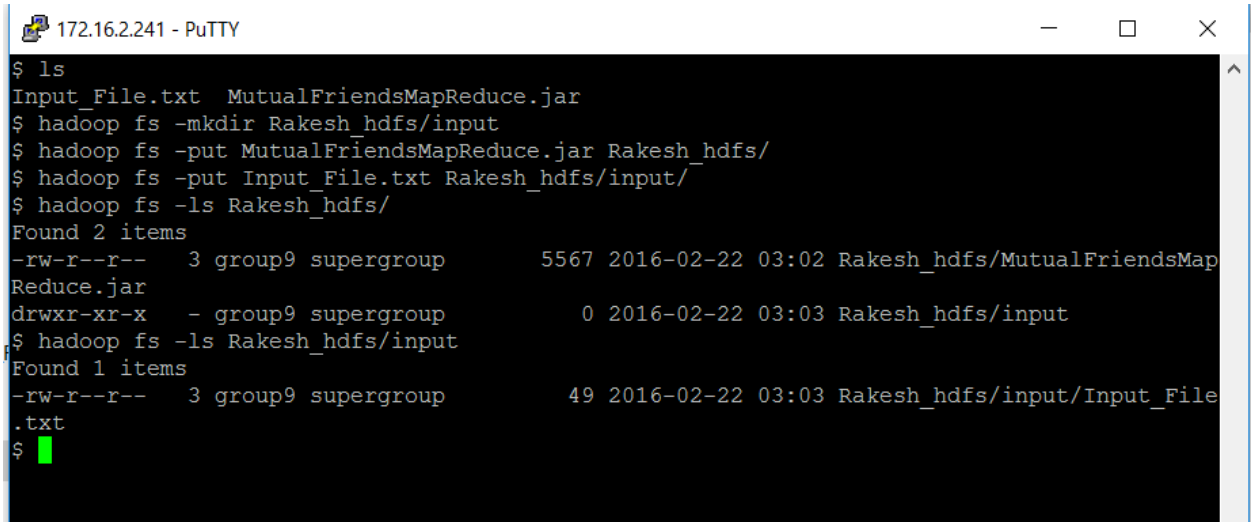
4. Create a input directory to move the input files into the folder



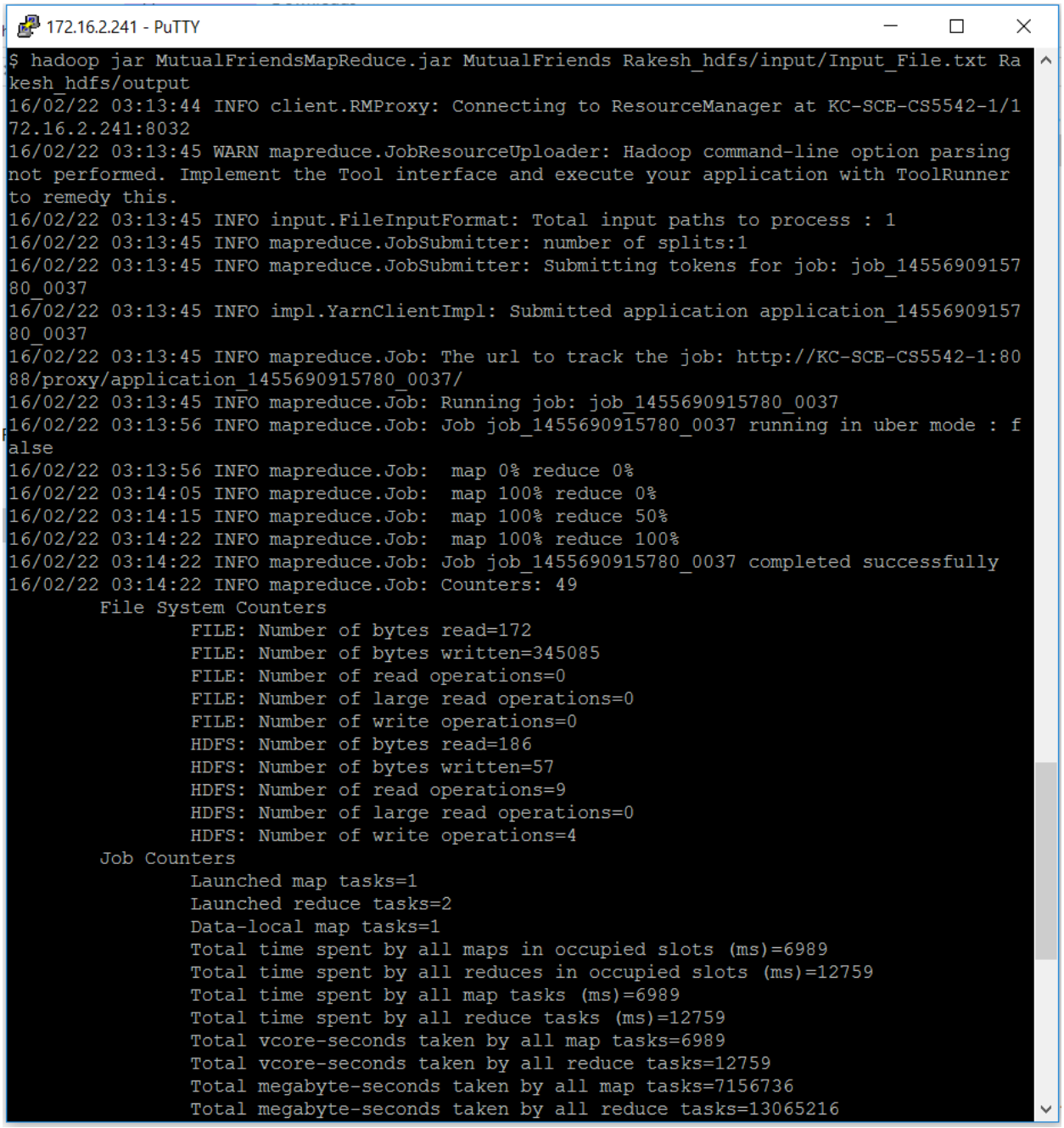
5. Move the input files such as Input_file.txt and jar files in to the Hadoop HDFS using command hadoop fs –put input_directory target_directory

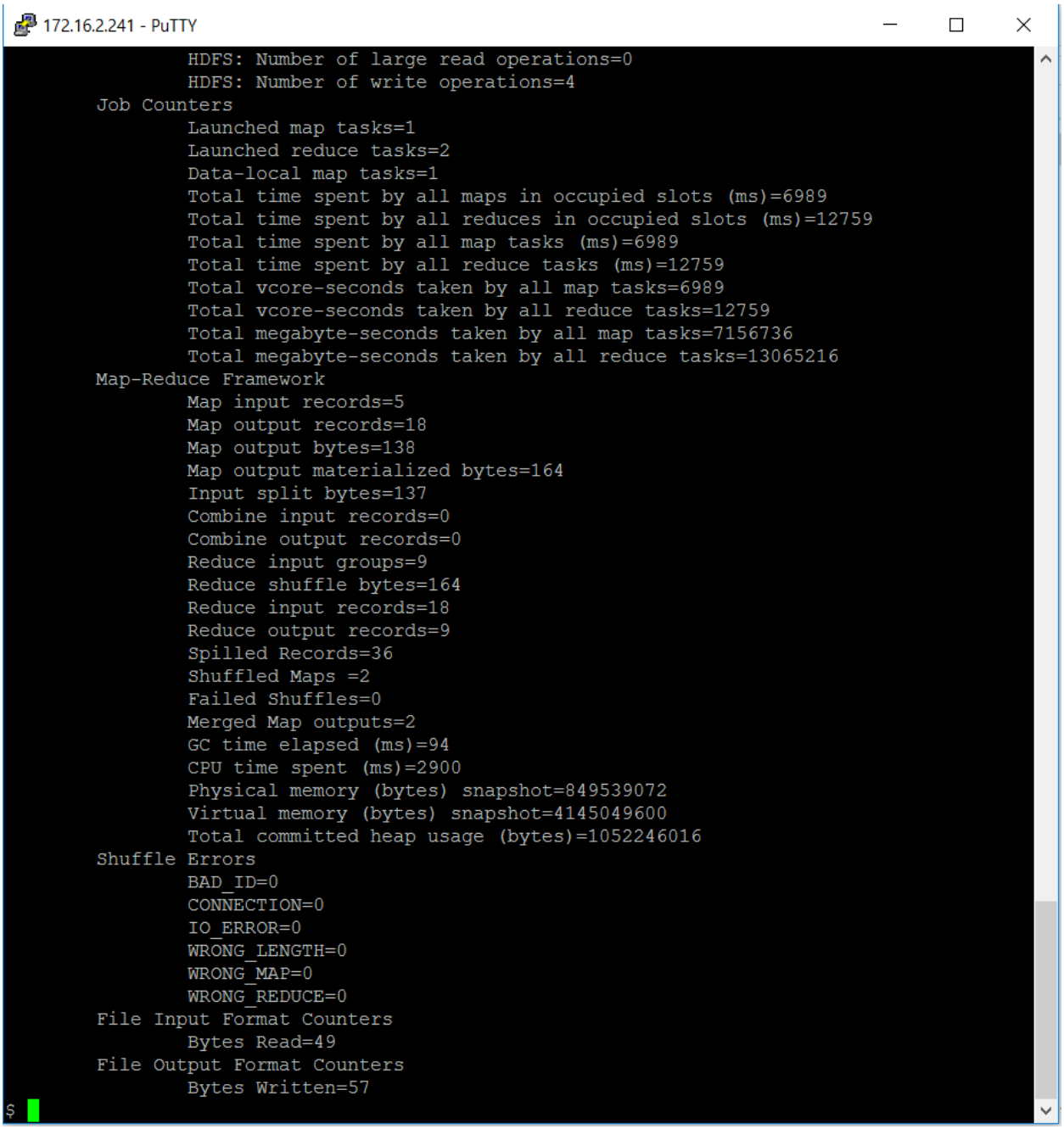


6. View the files in HDFS filesystem directory

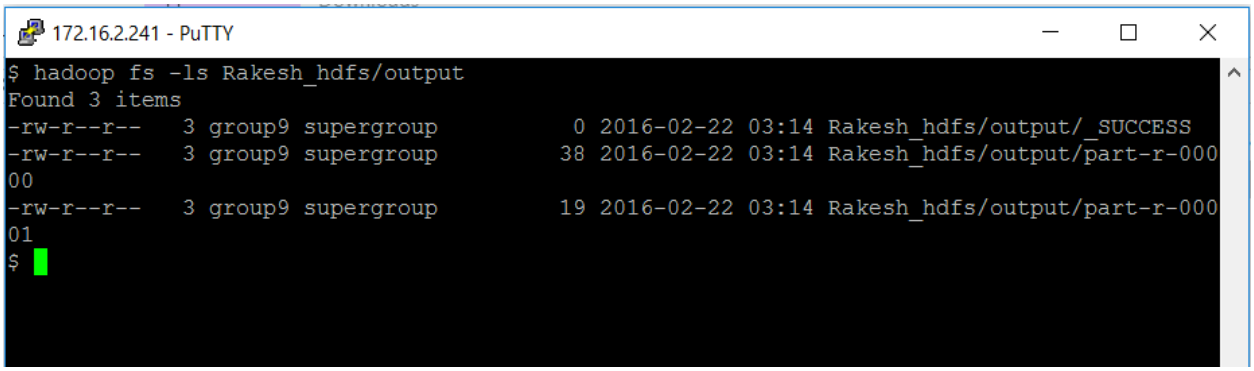


7. Run the Jar file using hadoop jar /path to Jar classname input_path output_path

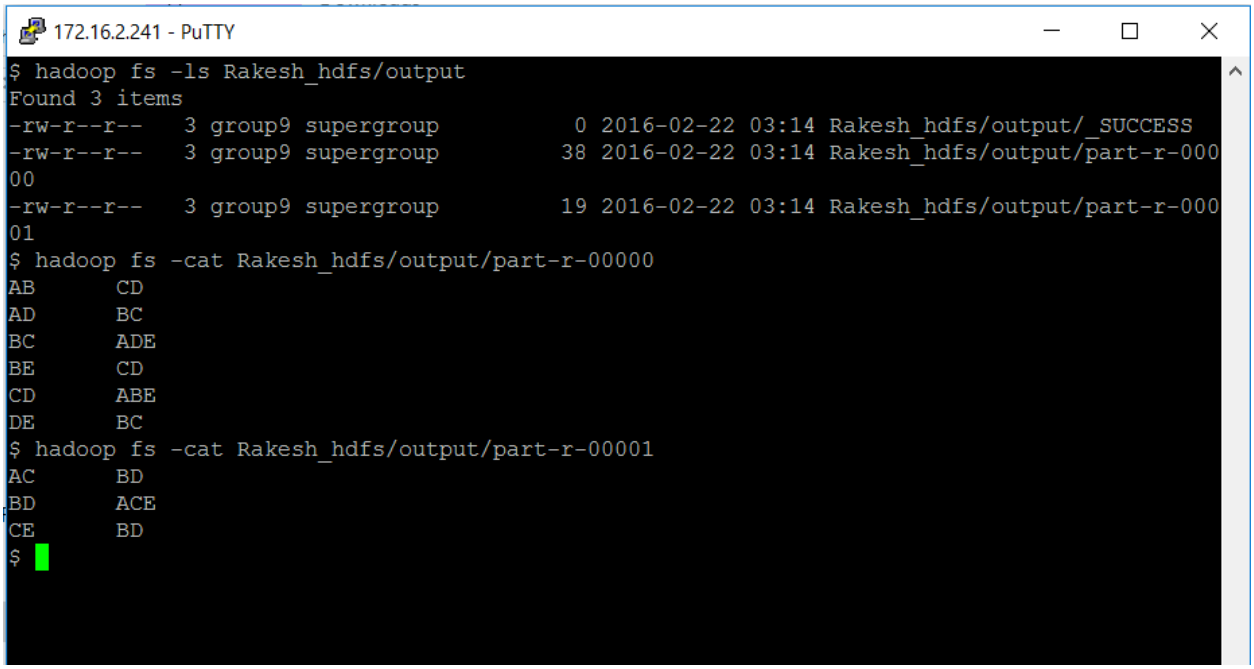



172.16.2.241 - PuTTY
HDFS: Number of large read operations=0
HDFS: Number of write operations=4
Job Counters
Launched map tasks=1
Launched reduce tasks=2
Data-local map tasks=1
Total time spent by all maps in occupied slots (ms)=6989
Total time spent by all reduces in occupied slots (ms)=12759
Total time spent by all map tasks (ms)=6989
Total time spent by all reduce tasks (ms)=12759
Total vcore-seconds taken by all map tasks=6989
Total vcore-seconds taken by all reduce tasks=12759
Total megabyte-seconds taken by all map tasks=7156736
Total megabyte-seconds taken by all reduce tasks=13065216
Map-Reduce Framework
Map input records=5
Map output records=18
Map output bytes=138
Map output materialized bytes=164
Input split bytes=137
Combine input records=0
Combine output records=0
Reduce input groups=9
Reduce shuffle bytes=164
Reduce input records=18
Reduce output records=9
Spilled Records=36
Shuffled Maps =2
Failed Shuffles=0
Merged Map outputs=2
GC time elapsed (ms)=94
CPU time spent (ms)=2900
Physical memory (bytes) snapshot=849539072
Virtual memory (bytes) snapshot=4145049600
Total committed heap usage (bytes)=1052246016
Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
File Input Format Counters
Bytes Read=49
File Output Format Counters
Bytes Written=57
$

8. View the files in output directory
We can see the _SUCCESS for successive run of job and output reduced key value pairs in part-r-* files



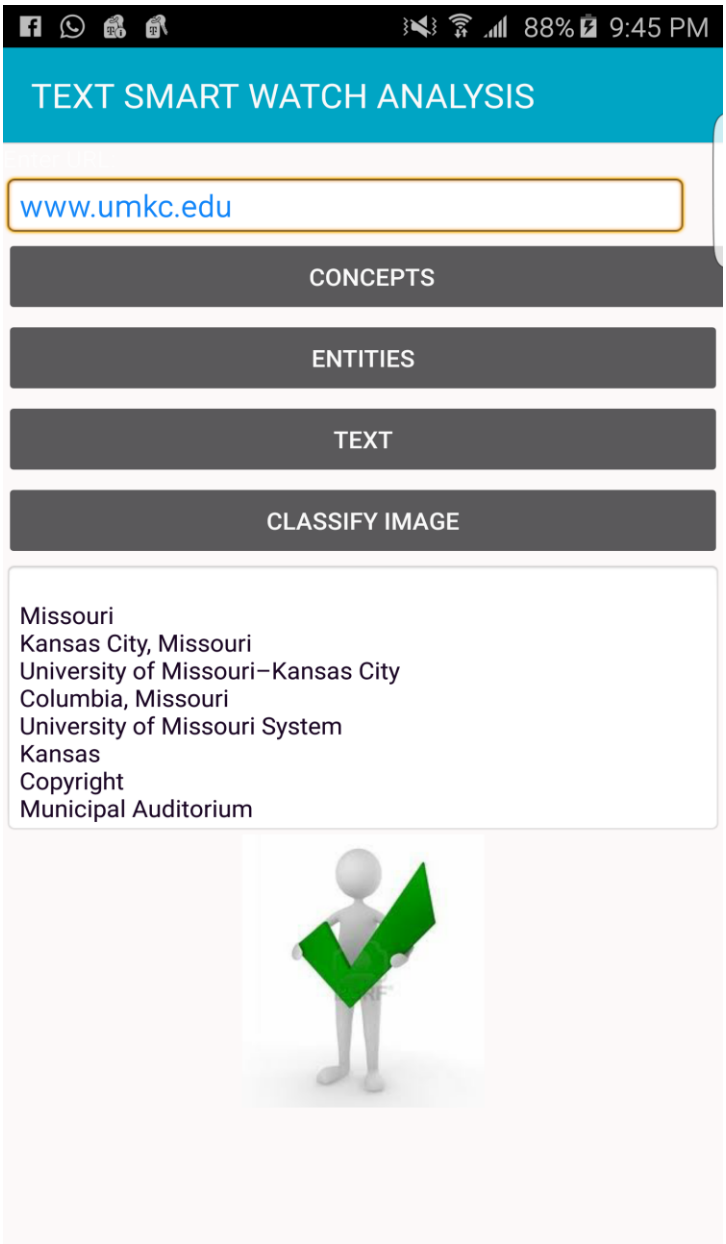
9. View the output files using cat command and we can see the Reduced key value pairs.
The common friends for the each pair can see the value of those index.



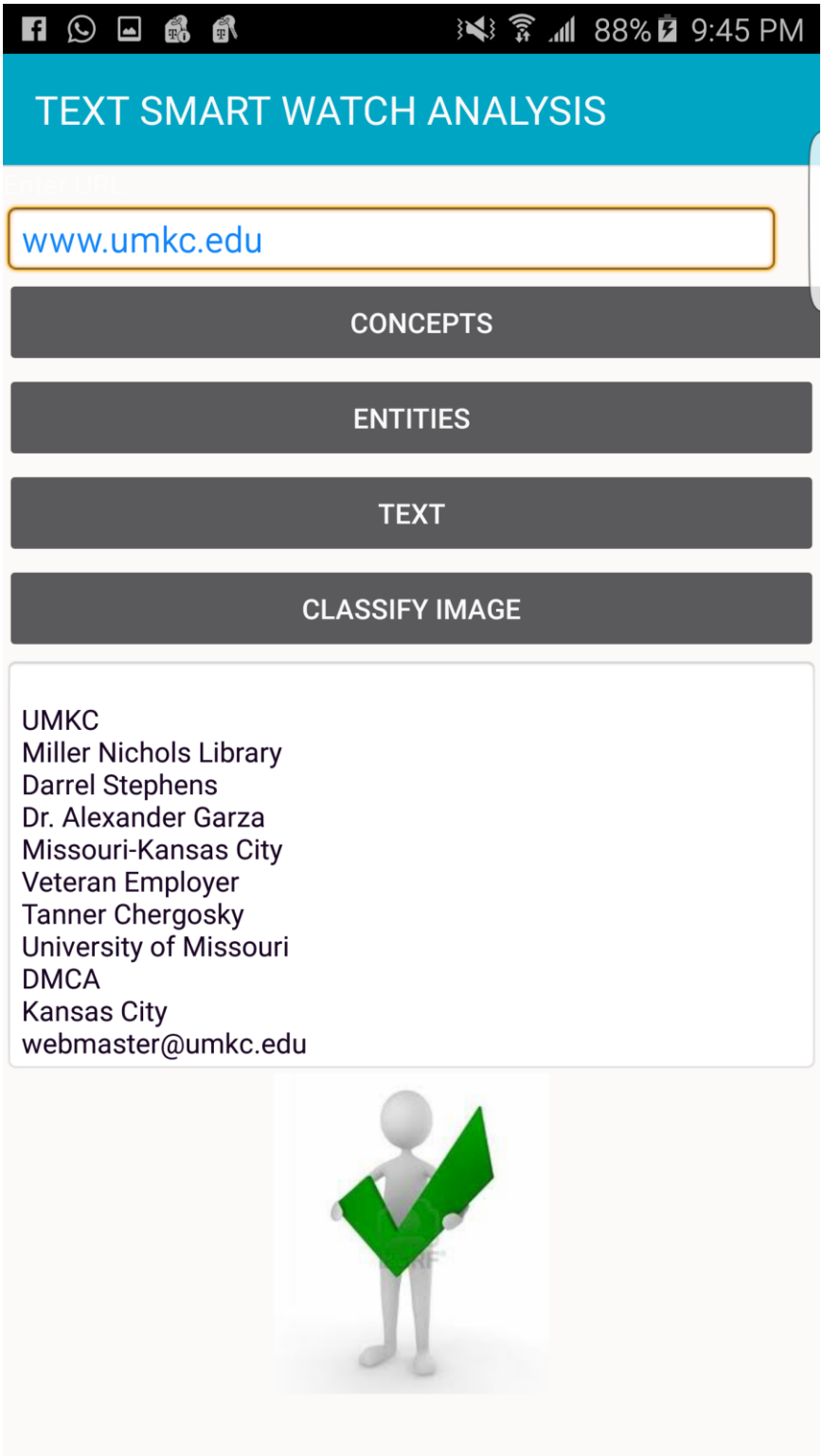
2. Smartphone/WatchApplicationImplement asmartwatch/smartphone applicationusing existingspeech services/image services (e.g., IBM Alchemyapi, Face++) related to your project.

I have done the SmartPhone application to perform the Text analysis and Image analysis using the Alchemy API.
I have used the Alchemy API to perform the ENTITY,CONCEPT and TEXT analysis over the TEXT or URL provided as input.
And also used the IMAGE CLASSIFY from the Alchemy API to classify the image given as Input.

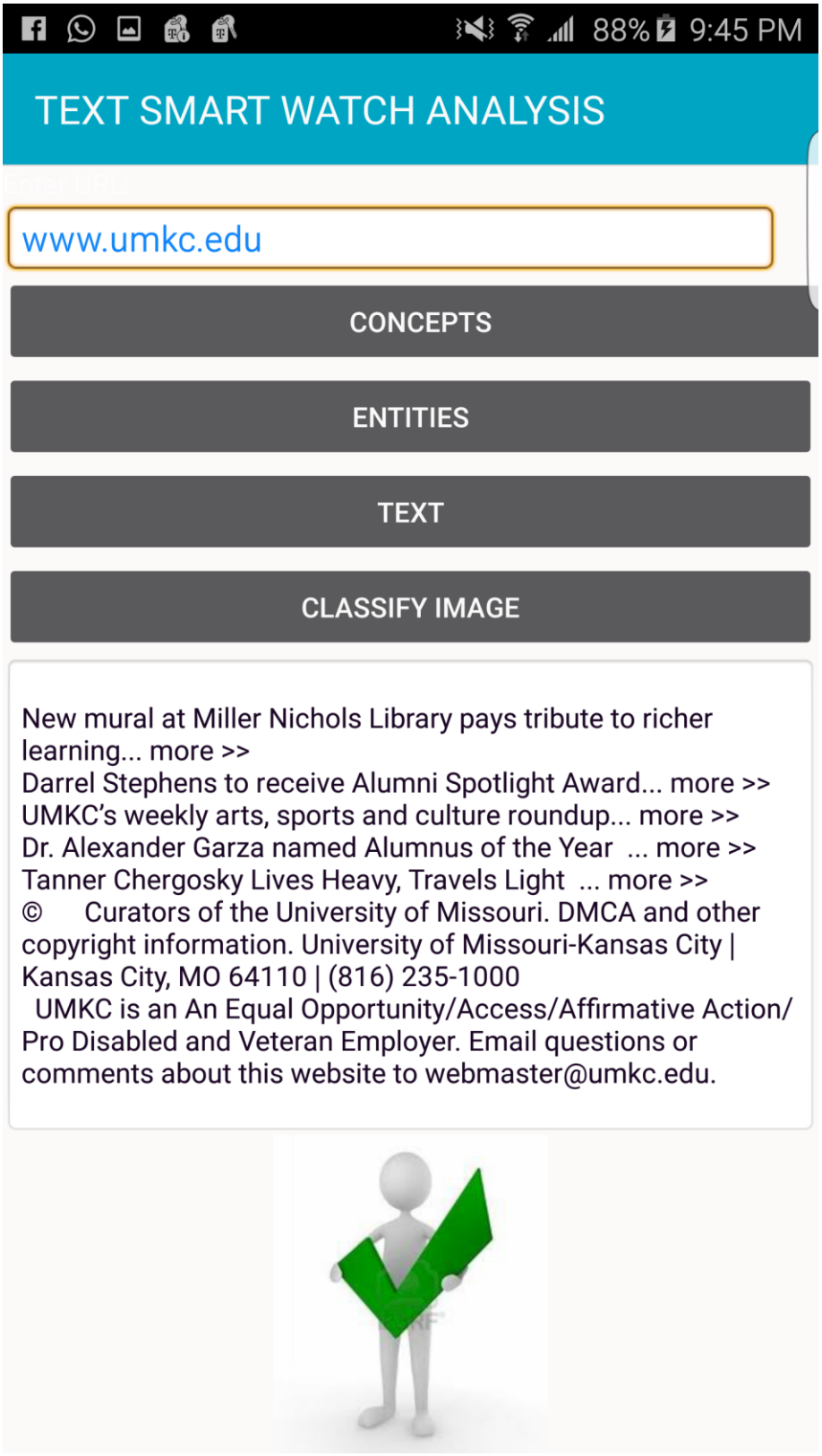
1. Concepts analysis of url " www.umkc.edu" using Alchemy API



2. ENTITY analysis on the URL www.umkc.edu provided as input



3. TEXT analysis of URL www.umkc.edu



4. IMAGE CLASSIFICATION on the image provided