Assignment 1 - Data Analysis
CA675 - Cloud Technologies

Name	Rohit Shinde
Student ID	22260213
E-mail	rohit.shinde2@mail.dcu.ie

Git Repository link: https://gitlab.computing.dcu.ie/shinder2/ca675-cloud-technology-assign1

Project link on cloud: https://console.cloud.google.com/dataproc/clusters?region=us-central1&organizationId=999744533918&project=weighty-volt-363313

Took dataset from Kaggle named Amazon Product Review (Electronics category) Dataset Link

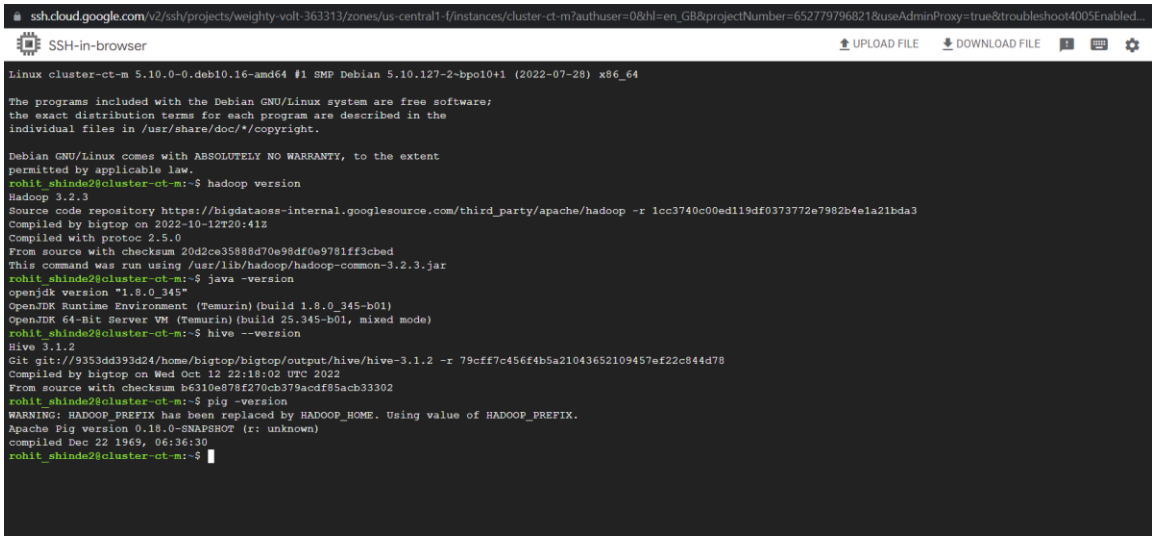
All the queries used in pig, hive, Hadoop are uploaded in gitlab repository Link

Task 1: Cloud Infrastructure Setup (AWS, GCP, Azure)

Task 1.1: Install Hadoop and create a Hadoop cluster

Used GCP Dataproc to create cluster named "cluster-ct" with 1 master node and 3 slave nodes.

Task 1.2: Install MapReduce, Pig and Hive to use the cluster created in Task 1.1



Task 2: Dataset

Task 2.1: Choose a relevant dataset

Dataset from Kaggle- Link then reduce that data size using Jupyter notebook python and divided data into chunks as follows:

```
import pandas as pd
import numpy as np
df=pd.read_json('Electronics.json', lines=True, nrows=500)
df
cnt=0
for df in pd.read_json('D:\CA675\Assignment 1\Electronics.json', lines=True,
chunksize=200000):
                    print(df.shape)
                    cnt=cnt+1
                    path=f"Electronics/review dataset{cnt}.csv"
                    df.to_csv(path)
```

Took file number 6 as random file named "review dataset.csv" and then uploaded it to bucket for further process

```
In [10]: cnt=0
         for df in pd.read_json('D:\CA675\Assignment 1\Electronics.json', lines=True, chunksize=200000):
             print(df.shape)
             cnt=cnt+1
             path=f"Electronics/review dataset{cnt}.csv"
             df.to_csv(path)

         (200000, 12)
         (200000, 12)
         (200000, 12)
         (200000, 12)
         (200000, 12)
         (200000, 12)
         (200000, 12)
         (200000, 12)
```

Task 2.2: Load data into chosen cloud technology (GCP)

For this task, uploaded the file "review dataset.csv" to bucket and then used Hadoop commands to load the data from bucket to pig directory

```
rohit_shinde2@cluster-ct-m:~$ hadoop fs -mkdir /pig
rohit_shinde2@cluster-ct-m:~$ hadoop fs -cp 'gs://bucket-cc-2722/data/review dataset.csv' /pig
rohit_shinde2@cluster-ct-m:~$ hadoop fs -ls /pig
Found 1 items
-rw-r--r--   2 rohit_shinde2 hadoop  128432665 2022-10-27 23:40 /pig/review dataset.csv
rohit_shinde2@cluster-ct-m:~$
```

```
rohit_shinde2@cluster-ct-m:~$ pig
WARNING: HADOOP_PREFIX has been replaced by HADOOP_HOME. Using value of HADOOP_PREFIX.
2022-10-28 00:05:56,744 INFO pig.ExecTypeProvider: Trying ExecType : LOCAL
2022-10-28 00:05:56,745 INFO pig.ExecTypeProvider: Trying ExecType : MAPREDUCE
2022-10-28 00:05:56,745 INFO pig.ExecTypeProvider: Picked MAPREDUCE as the ExecType
2022-10-28 00:05:56,791 [main] INFO  org.apache.pig.Main - Apache Pig version 0.18.0-SNAPSHOT (r: unknown) compiled Dec 22 1969, 06:36:30
2022-10-28 00:05:56,791 [main] INFO  org.apache.pig.Main - Logging error messages to: /home/rohit_shinde2/pig_1666915556788.log
2022-10-28 00:05:56,809 [main] INFO  org.apache.pig.impl.util.Utils - Default bootup file /home/rohit_shinde2/.pigbootup not found
2022-10-28 00:05:57,163 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtrack
er.address
2022-10-28 00:05:57,163 [main] INFO  org.apache.pig.backend.hadoop.executionengine.HExecutionEngine - Connecting to hadoop file system at: hdfs://cluster
-ct-m
2022-10-28 00:05:58,375 [main] INFO  org.apache.pig.PigServer - Pig Script ID for the session: PIG-default-d377caf7-706a-4b08-aeb1-6a07ba6580f6
2022-10-28 00:05:58,552 [main] INFO  org.apache.hadoop.yarn.client.api.impl.TimelineClientImpl - Timeline service address: cluster-ct-m:8188
2022-10-28 00:05:58,884 [main] INFO  org.apache.pig.backend.hadoop.PigATSClient - Created ATS Hook
2022-10-28 00:05:58,910 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - yarn.resourcemanager.system-metrics-publisher.enabled is deprecat
ed. Instead, use yarn.system-metrics-publisher.enabled
grunt> register /home/rohit_shinde2/pig/piggybank.jar
2022-10-28 00:06:06,346 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - yarn.resourcemanager.system-metrics-publisher.enabled is deprecat
ed. Instead, use yarn.system-metrics-publisher.enabled
grunt>
```

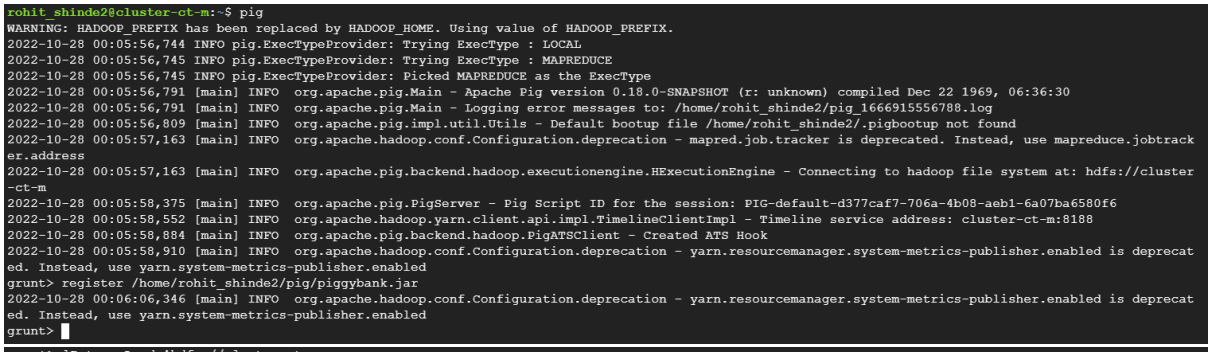
```
grunt> lData = Load 'hdfs://cluster-ct-m
>> /pig/review dataset.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','YES_MULTILINE') AS
>> (sno: chararray, id: chararray, reviewerID: chararray, asin: chararray, reviewerName: chararray, helpful: chararray, reviewText: chararray, overall: i
nt, summary: chararray, unixReviewTime: chararray, reviewTime: chararray, category: chararray, class: int);
2022-10-28 00:23:50,959 [main] ERROR org.apache.pig.tools.grunt.Grunt - ERROR 1200: <line 1, column 13>  mismatched input ''hdfs://cluster-ct-m
/pig/review dataset.csv'' expecting QUOTEDSTRING
Details at logfile: /home/rohit_shinde2/pig_1666916580611.log
grunt> lData = Load 'hdfs://cluster-ct-m/pig/review dataset.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','YES_MULTILINE') AS
>> (sno: chararray, id: chararray, reviewerID: chararray, asin: chararray, reviewerName: chararray, helpful: chararray, reviewText: chararray, overall: i
nt, summary: chararray, unixReviewTime: chararray, reviewTime: chararray, category: chararray, class: int);
2022-10-28 00:24:32,859 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - yarn.resourcemanager.system-metrics-publisher.enabled is deprecat
ed. Instead, use yarn.system-metrics-publisher.enabled
grunt>
```

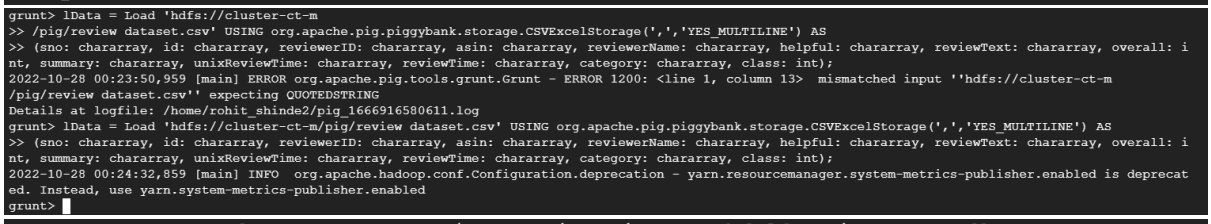
Task 3: Clean and process the data using Pig and/or Hive

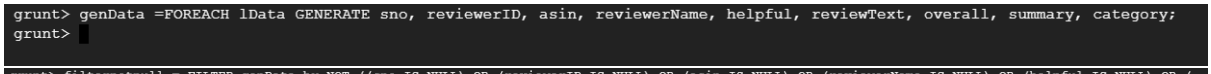
Cleaned the data with the use of pig as it is faster as compared to hive and it uses a multi-query approach. The data had line-break characters and commas so to overcome this, used piggybank library- Link and registered it in pig. To enable the CSV read multi-line in pig, registered the piggybank.jar file.

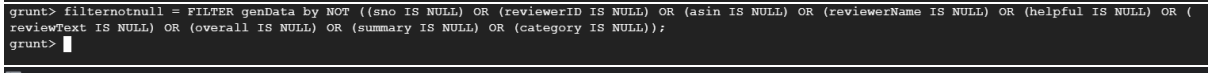
```
grunt> register /home/rohit_shinde2/pig/piggybank.jar
```

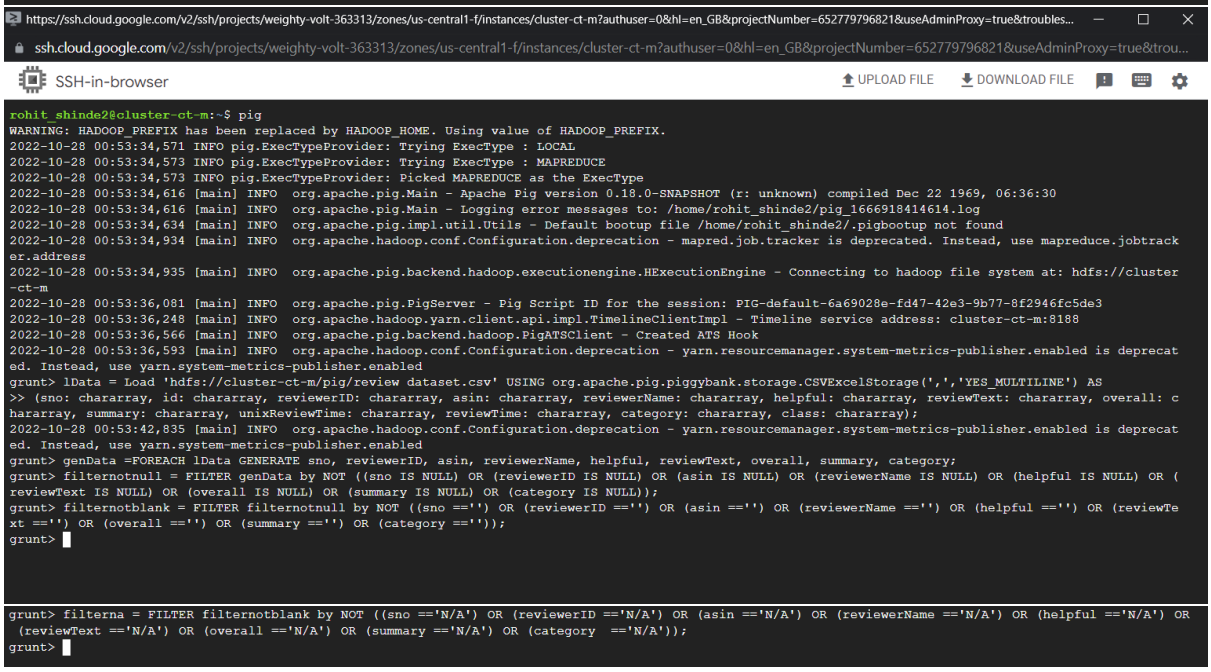
The data is cleaned in pig by checking null values, "N/A" values, blank values, etc. and finally the cleaned data is stored with store query. All the queries used for cleaning data using pig is uploaded on git repository- Link.











```
grunt> STORE filterna INTO '/CleanData' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','YES_MULTILINE');
```

Then put the cleaned data named "data_input.csv" to local path directory as well as in google bucket.

```
rohit_shinde2@cluster-ct-m:~$ hadoop fs -ls /CleanData/
Found 2 items
-rw-r--r--   2 rohit_shinde2 hadoop          0 2022-10-28 01:13 /CleanData/_SUCCESS
-rw-r--r--   2 rohit_shinde2 hadoop  114353394 2022-10-28 01:13 /CleanData/part-m-00000
rohit_shinde2@cluster-ct-m:~$ hadoop fs -rm /CleanData/_SUCCESS
Deleted /CleanData/_SUCCESS
rohit_shinde2@cluster-ct-m:~$ hadoop fs -getmerge /CleanData/ /home/rohit_shinde2/data_input.csv
rohit_shinde2@cluster-ct-m:~$ hadoop fs -put data_input.csv 'gs://bucket-cc-2722/cleandata'
rohit_shinde2@cluster-ct-m:~$
```

Task 4: Ham and Spam using Pig and/or Hive

Task 4.1: Query processed data to differentiate ham and spam part of the dataset

Used hive for this task.

Created database named db to start with the hive queries:
hive> create database db;
hive> use db;

Created table electronics and loaded the cleaned data from pig:

```
hive> select count(*) as total_count from electronics;
Query ID = rohit_shinde2_20221101115828_923976cb-dab0-41ed-b5f9-e4eb78752ee4
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1667302556211_0001)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 8.16 s
----------------------------------------------------------------------------------------------
OK
total_count
197013
Time taken: 8.987 seconds, Fetched: 1 row(s)
hive>
```

Downloaded bag of words from link, extracted them with link and uploaded it in bucket then

Then created bag of words table named bag2 and loaded the bag of words into it. Also created table named wordcount and counter to count the words in the bag. The queries are uploaded in gitlab repository- Link.

After doing this, the words are selected from bag of words on the basis of their count with respect to summary column from the table electronics.

```
OK
counter.word    counter.count
great   36090
works   12649
product 10450
price   8473
excellent       5973
camera  5726
perfect 4578
quality 4553
sound   4059
mount   3850
value   2949
awesome 2920
money   2728
little  2716
garmin  2375
better  2245
cable   2216
player  2123
battery 1936
speakers        1903
worked  1886
cheap   1853
worth   1678
expected        1418
amazing 1376
portable        1361
small   1335
review  1284
headphones      1273
drive   1270
Time taken: 6.002 seconds, Fetched: 30 row(s)
hive>
```

Created spam and ham table separately using bag of words (Words used from bag of words as spam words are excellent, awesome, speakers, works, product, price). Queries are uploaded on gitlab repository- Link.

Task 4.2: Find the top 10 spam accounts

hive> select sno, custID, custname, helpful, overall, category, summary from db.spam order by helpful desc limit 10;

```
hive> select sno,custID,custname,helpful,overall,category,summary from db.spam order by helpful desc limit 10;
Query ID = rohit_shinde2_20221101122415_33a6e776-ded3-4794-826f-5bb47c90a46b
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1667302556211_0002)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 5.58 s
----------------------------------------------------------------------------------------------
OK
sno     custid  custname        helpful overall category        summary
1194421 ADNEBAXSYYXS6   Osmos   [98, 100]       5       Electronics     Amazing TV especially for the price
1001741 A3NCIN6TNL0MGA  M. JEFFREY MCMAHON "herculodge" [970, 1013]     4       Electronics     Good Value at This Price Point
1050930 A3GPH7UNBO1X9E  Gary E. Meyer    [97, 105]       4       Electronics     works very well for spotting scopes
1113806 A124RSNIZX0J7M  Jim     [95, 96]        4       Electronics     Great Camera for the Price, beware accessory creep
1179777 A386NVAVQV5WUO  Richard D. Cappetto "RickDC"    [95, 100]       5       Electronics     WOW what a great camera for this price, outstanding!
1026404 A321B3UJKTWLOL  sallen  [93, 102]       5       Electronics     Simply Awesome!!!
1178223 A2FWZ58M6IQMD7  Doc Stew        [92, 98]        5       Electronics     Awesome backpack for school!
1026101 A39KJOCZGCM8P1  Chris Lee Mullins       [92, 98]        4       Electronics     Easy to setup, works great with MacOS X 10.4.8
1062816 A35I7SM1SHRQ91  T. Burbey       [92, 94]        4       Electronics     quality speakers, just very finicky
1017530 A2JW236794LPMB  D. Dwyer "amazonian1000"        [92, 111]       5       Electronics     Great HD player (especially for the price)...
Time taken: 6.401 seconds, Fetched: 10 row(s)
hive>
```

Task 4.3: Find the top 10 ham accounts

```
hive> select sno, custID, custname, helpful, overall, category, summary
from db.ham order by helpful desc limit 10;
```

```
hive> select sno,custID,custname,helpful,overall,category,summary from db.ham order by helpful desc limit 10;
Query ID = rohit_shinde2_20221101122652_6dbe2e8b-2054-4a4f-becf-ec3e8e5a1e03
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1667302556211_0002)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 4.94 s
----------------------------------------------------------------------------------------------
OK
sno     custid  custname        helpful overall category        summary
1046116 A399PFIZ68MS9N  George A. Sopko [99, 118]       3       Electronics     Huge disappointment!
1130831 A1377JHZW2UTEF  Super Movie Fan Man     [99, 115]       3       Electronics     Nice Scope Crippled by an Unstable mount & Flimsy Tripod
1175073 A251RUG92V4RV   Jack of IL      [99, 111]       1       Electronics     Buyer beware - most posted review no longer apply!
1075622 A16KADFBQHVG3Y  John Bonanno    [99, 111]       2       Electronics     Not Very Robust
1151695 A269M8M0L0EIL4  Amazon Customer [99, 106]       1       Electronics     Bass rumble
1172973 A1RYRGZ75IBQNY  Sydney Jane     [99, 105]       2       Electronics     Bait & Switch
1078864 A8P5ZBH50PPOR   Amazon Customer [98, 98]        2       Electronics     A pain at first to install
1136760 AWZR0O65DL2Q    Nameless Faceless User  [97, 110]       1       Electronics     Stay away
1124180 A37D1ZP8GBHE38  Sprout  [96, 97]        2       Electronics     Stick with the X-230
1059907 A3IQ2ODI3FHFDV  H. Hojda        [96, 113]       1       Electronics     CONSUMER ALERT
Time taken: 5.88 seconds, Fetched: 10 row(s)
hive>
```

Task 5: TF-IDF using MapReduce

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

By using pig cleaned and processed the CSV file which was exported from spam table and this data is fed to mapper and reducer script in python.

--Hive—
In hive, created the table named spamtf and inserted the data into it from table spam.

--Pig—
With the help of pig data is loaded and cleaned again in which null values, blank values and "N/A" values, etc are removed. Final data was stored in local path /pigstore2

Then for mapper reducer the python files are copied from bucket folder pyfiles to local directory /python.

```
-- for copying file from bucket to hadoop file system
hadoop fs -cp 'gs://bucket-cc-2722/pyfiles/*' /python
-- for copying file from hadoop file system  to local directory
hadoop fs -get /python* /home/rohit_shinde2/python2/
```

Hadoop commands to execute mapper and reducer:

```
rohit_shinde2@cluster-ct-m:~$ hadoop jar /usr/lib/hadoop/hadoop-streaming.jar -file /home/rohit_shinde2/python2
/mapper1.py /home/rohit_shinde2/python2/reducer1.py -mapper "python mapper1.py" -reducer "python reducer1.py" -
input /pigstore2/part-r-00000 -output /mapred/spam/output1
```

```
rohit_shinde2@cluster-ct-m:~$ hadoop jar /usr/lib/hadoop/hadoop-streaming.jar -file /home/rohit_shinde2/python2
/mapper2.py /home/rohit_shinde2/python2/reducer2.py -mapper "python mapper2.py" -reducer "python reducer2.py" -
input /mapred/spam/output1/part-0000* -output /mapred/spam/output2
```

```
rohit_shinde2@cluster-ct-m:~$ hadoop jar /usr/lib/hadoop/hadoop-streaming.jar -file /home/rohit_shinde2/python2
/mapper3.py /home/rohit_shinde2/python2/reducer3.py -mapper "python mapper3.py" -reducer "python reducer3.py" -
input /mapred/spam/output2/part-0000* -output /mapred/spam/output3
```

```
rohit_shinde2@cluster-ct-m:~$ hadoop jar /usr/lib/hadoop/hadoop-streaming.jar -files /home/rohit_shinde2/python
2/mapper4.py -mapper "python mapper4.py" -input /mapred/spam/output3/part-0000* -output /mapred/spam/output4
```

For final outcome, created table named spamtfidf and loaded the final data from mapper and reducer to spamtfidf and extracted top 10 spam accounts with their tf-idf values as below:

```
hive> SELECT * FROM (SELECT id,word,tfidf, rank() over (PARTITION BY id ORDER BY tfidf DESC) as rank FROM db.spamtfidf DISTRIBUTE BY id SORT BY id desc) rk WHERE rank < 10 ORDER BY id, rank;
WARNING: Order/Sort by without limit in sub query or view [rk] is removed, as it's pointless and bad for performance.
Query ID = rohit_shinde2_20221104204853_38fd4ec3-721c-4414-b5e9-d8ebc0a79f2a
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1667594468392_0002)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 ..... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ..... container     SUCCEEDED      1          1        0        0       0       0
Reducer 3 ..... container     SUCCEEDED      1          1        0        0       0       0
Reducer 4 ..... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 04/04  [==========================>>] 100%  ELAPSED TIME: 10.76 s
----------------------------------------------------------------------------------------------
OK
rk.id   rk.word rk.tfidf        rk.rank
1001741 player  0.767528        1
1001741 value   0.767528        1
1001741 especially      0.536479        3
1017530 finicky 0.575646        1
1017530 hd      0.575646        1
1017530 price   0.173287        3
1026101 quality 0.767528        1
1026404 amazing 2.302585        1
1026404 great   0.916291        2
1050930 setup   0.575646        1
1050930 good    0.575646        1
1050930 well    0.575646        1
1050930 very    0.402359        4
1062816 simply  0.767528        1
1062816 easy    0.767528        1
1062816 point   0.767528        1
1113806 creep   0.460517        1
1113806 beware  0.460517        1
1113806 awesome 0.460517        1
1113806 camera  0.321888        4
1179777 scopes  0.767528        1
1179777 accessory       0.767528        1
1179777 wow     0.767528        1
1194421 spotting        0.767528        1
1194421 tv      0.767528        1
```

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

The same steps are used to find out top 10 ham records with tf-idf values.

```
hive> SELECT * FROM (SELECT id,word,tfidf, rank() over (PARTITION BY id ORDER BY tfidf DESC) as rank FROM db.hamtfidf DISTRIBUTE BY id SORT BY id desc) rk WHERE rank < 10 ORDER BY id, rank;
WARNING: Order/Sort by without limit in sub query or view [rk] is removed, as it's pointless and bad for performance.
Query ID = rohit_shinde2_20221104205128_c4b967b7-ddb2-4e46-80e4-6e36af2c8302
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1667594468392_0002)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 ..... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ..... container     SUCCEEDED      1          1        0        0       0       0
Reducer 3 ..... container     SUCCEEDED      1          1        0        0       0       0
Reducer 4 ..... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 04/04  [==========================>>] 100%  ELAPSED TIME: 6.13 s
----------------------------------------------------------------------------------------------
OK
rk.id   rk.word rk.tfidf        rk.rank
1075622 longer  2.302585        1
1078864 install 2.302585        1
1078864 pain    2.302585        1
1130831 huge    0.460517        1
1130831 nice    0.460517        1
1130831 scope   0.460517        1
1130831 bass    0.460517        1
1130831 mount   0.460517        1
1130831 tripod  0.460517        1
1130831 posted  0.460517        1
1136760 robust  2.302585        1
1136760 stay    2.302585        1
1175073 flimsy  0.767528        1
1175073 away    0.767528        1
1175073 buyer   0.767528        1
Time taken: 7.007 seconds, Fetched: 15 row(s)
hive>
```