

In this document per-user TF-IDF of the top 10 terms for each of the top 10 users is being calculated.

### Steps Prior to calculate TF-IDF:

1. Dataset Final\_PostData.csv is cleaned using basic sed command to achieve a clean csv file. Using the sed command new lines with spaces have been replaced.

**Command:** sed 'a;N;\$!ba;s/\n//g' Final\_PostData.csv > Clean\_Final.csv

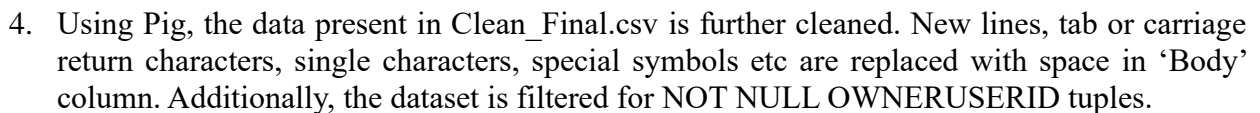
```
shradha_shivani2@cluster-hadoop-m:~$ sudo sed 'a;N;$!ba;s/\n//g' Final_PostData.csv > Clean_Final.csv
shradha_shivani2@cluster-hadoop-m:~$ ls -ltr
total 756088
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 2727761 Jul  3 2020 apache-ivy-2.5.0-bin.tar.gz.1
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 278813748 Jul  3 2020 apache-hive-3.1.2-bin.tar.gz
drwxrwxr-x 10 shradha_shivani2 shradha_shivani2 4096 Oct 25 18:36 apache-hive-3.1.2-bin
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 28469 Oct 25 21:33 apache-ivy-2.5.0-bin.tar.gz
drwxrwxr-x 3 shradha_shivani2 shradha_shivani2 4096 Oct 25 23:06 Pig
-rw-r--r-- 1 root root 2615 Oct 25 23:15 pig_1635203662479.log
drwxr-xr-x 2 root root 4096 Oct 25 23:24 combined
-rw-r--r-- 1 root root 2617 Oct 25 23:24 pig_1635203850422.log
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 223258 Oct 26 00:00 hiveResults.csv
drwxrwxr-x 3 shradha_shivani2 shradha_shivani2 4096 Oct 26 15:14 MapReduce
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 248559656 Oct 26 15:36 Final_PostData.csv
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 243835097 Oct 26 15:37 Clean_Final.csv
shradha_shivani2@cluster-hadoop-m:~$
```

2. Basic cleaned file 'Clean\_Final.csv' is moved to '/user/CA1' in HDFS

**Command:** hadoop fs -put Clean\_Final.csv /user/CA1

```
shradha_shivani2@cluster-hadoop-m:~$ hadoop fs -put Clean_Final.csv /user/CA1/
shradha_shivani2@cluster-hadoop-m:~$ hadoop fs -ls
ls: '.': No such file or directory
shradha_shivani2@cluster-hadoop-m:~$ hadoop fs -ls /user/CA1
Found 4 items
-rw-r--r-- 2 shradha_shivani2 hadoop 243835097 2021-10-26 15:48 /user/CA1/Clean_Final.csv
-rw-r--r-- 2 shradha_shivani2 hadoop 248559656 2021-10-25 23:16 /user/CA1/Final_PostData.csv
drwxr-xr-x - shradha_shivani2 hadoop 0 2021-10-26 15:43 /user/CA1/TFIDF
drwxr-xr-x - root hadoop 0 2021-10-25 23:28 /user/CA1/combined
shradha_shivani2@cluster-hadoop-m:~$
```

3. Logged in to pig terminal using HCatalog. The Command used is **pig -useHCatalog**.



```
grunt> loadPost = load 'hdfs:///user/CA1/Clean_Final.csv' using
org.apache.pig.piggybank.storage.CSVExcelStorage(',',
'YES_MULTILINE','NOCHANGE','SKIP_INPUT_HEADER')
as(id:int,posttypeid:int,acceptedanswerid:int,
parentid:int,creationdate:DATETIME,deletiondate:DATETIME,score:int,viewcount:int,body
:chararray,owneruserid:int,ownerdisplayname:chararray,lasteditoruserid:int,lasteditordisplay
name:chararray,lasteditdate:DATETIME,lastactivitydate:DATETIME,title:chararray,tags:cha
rarray,answercount:int,commentcount:int,favoritecount:int,closeddate:DATETIME,commun
ityowneddate:DATETIME,contentlicense:chararray);
2021-10-26 15:55:56,451 [main] INFO org.apache.hadoop.conf.Configuration.deprecation -
yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use
yarn.system-metrics-publisher.enabled
grunt>
```

```
grunt> cleanPost = FOREACH loadPost GENERATE id, score, owneruserid,
REPLACE(REPLACE(REPLACE(REPLACE(REPLACE((REPLACE(body,['\r\n']+,""), '<[
^>]*>'
,
'), '[^a-zA-Z\s\']+'
,
'), '(?=\S*[''])'([a-zA-Z\']+'
), '(?<![\w\~])\w(?:[\w\~])?')'
, '[ ]{2,};'') AS body;
```

```
grunt> loadPost = load 'hdfs:///user/CA1/Clean_Final.csv' using org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'YES_MULTILINE', 'NOCHANGE', 'SKIP_INPUT_HEADER') as(id:int,ptid:int,creationdate:DATETIME,deletiondate:DATETIME,score:int,viewcount:int,body:chararray,owneruserid:int,ownerdisplayname:chararray,lasteditoruserid:int,lasteditordisplayname:chararray,commentcount:int,favoritecount:int,closeddate:DATETIME,communityowneddate:DATETIME,contentlicense:chararray);
2021-10-26 15:55:56,451 [main] INFO org.apache.hadoop.conf.Configuration.deprecation - yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
grunt>
grunt>
grunt>
grunt> cleanPost = FOREACH loadPost GENERATE id, score, owneruserid, REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(REPLACE(body, '\r\n', '')), '<[^>]*>', ' '), '[^a-zA-Z\s\']+', ' '), '(?=\s+[\'])([a-zA-Z\']+)','(?![\w\-\'])(\w(?:[\w\-\']|' '[ ]{2,}', ' ') AS body;
grunt>
```

grunt> filter\_data = FILTER post\_data BY (owneruserid is not null);

```
grunt> filter_data = FILTER cleanPost BY (owneruserid is not null);
grunt> |
```

### **Computation of TF-IDF:**

1. In this section, the users are grouped together, their total score of their various posts is calculated and arranged in descending order . The result is just limited to top 10. So, in other terms top 10 posts of top 10 users are selected as per their total score.

#### **Command:**

```
grunt> select_distinct_users_post = GROUP filter_data BY owneruserid;
grunt> select_users_by_max_score = FOREACH select_distinct_users_post GENERATE
group AS userid, SUM(filter_data.score) AS maxscore;
grunt> select_users_by_max_score_desc_order = ORDER select_users_by_max_score BY
maxscore DESC;
grunt> select_data_limit_10 = LIMIT select_users_by_max_score_desc_order 10;
grunt> select_top_10_user_id = FOREACH select_data_limit_10 GENERATE userid;
grunt> select_posts_by_10_users = JOIN filter_data BY owneruserid, select_top_10_user_id
BY userid;
grunt> select_posts_by_10_users = FOREACH select_posts_by_10_users GENERATE
filter_data::owneruserid, LOWER(TRIM(REPLACE(filter_data::body, '[ ]{2,}', ' '))) AS
filter_data::body
```

```
grunt> select_distinct_users_post = GROUP filter_data BY owneruserid;
grunt> select_users_by_max_score = FOREACH select_distinct_users_post GENERATE group AS userid, SUM(filter_data.score) AS maxscore;
grunt> select_users_by_max_score_desc_order = ORDER select_users_by_max_score BY maxscore DESC;
grunt> select_data_limit_10 = LIMIT select_users_by_max_score_desc_order 10;
grunt> select_top_10_user_id = FOREACH select_data_limit_10 GENERATE userid;
grunt> select_posts_by_10_users = JOIN filter_data BY owneruserid, select_top_10_user_id BY userid;
grunt> select_posts_by_10_users = FOREACH select_posts_by_10_users GENERATE filter_data::owneruserid, LOWER(TRIM(REPLACE(filter_data::body,'[]{2,}',' '))) AS filter_data::body;
grunt> |
```

- Next, the resultant set is stored in a folder named 'TFIDF' under '/user/CA1/'

**Command:**

```
Grunt>STORE select_posts_by_10_users INTO '/user/CA1/TFIDF' USING
org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'NO_MULTILINE', 'NOCHANGE', 'S
KIP_OUTPUT_HEADER');
```

```
system-metrics-publisher.enabled
grunt> STORE select_posts_by_10_users INTO '/user/CA1/TFIDF' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'NO_MULTILINE', 'NOCHANGE', 'SKIP_OUTPUT_HEADER');
```

- Successful logs are obtained for data storing in TFIDF location and the files part-r-00000 and \_SUCCESS log can be seen under this location.

```
2021-10-26 16:49:40,711 [main] INFO org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at cluster-hadoop-m/10.164.0.3:8032
2021-10-26 16:49:40,712 [main] INFO org.apache.hadoop.yarn.client.AHSProxy - Connecting to Application History server at cluster-hadoop-m/10.164.0.3:10200
2021-10-26 16:49:40,714 [main] INFO org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redirecting to job histo
ry server
2021-10-26 16:49:40,736 [main] INFO org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at cluster-hadoop-m/10.164.0.3:8032
2021-10-26 16:49:40,736 [main] INFO org.apache.hadoop.yarn.client.AHSProxy - Connecting to Application History server at cluster-hadoop-m/10.164.0.3:10200
2021-10-26 16:49:40,738 [main] INFO org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redirecting to job histo
ry server
2021-10-26 16:49:40,754 [main] WARN org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Encountered Warning FIELD_DISCARDED_TYPE_CONVERSION_FAILED
184 time(s).
2021-10-26 16:49:40,754 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
grunt>
grunt>
grunt>
grunt> fs -ls /user/CA1/TFIDF
Found 2 items
-rw-r--r-- 2 shradha_shivani2 hadoop 0 2021-10-26 16:49 /user/CA1/TFIDF/_SUCCESS
-rw-r--r-- 2 shradha_shivani2 hadoop 125193 2021-10-26 16:49 /user/CA1/TFIDF/part-r-00000
grunt> |
```

**Implementing TFIDE:**

- 1<sup>st</sup> the file stored in '/user/CA1/TFIDF/part-r-00000' is merged with the local folder.

**Command:** `hadoop fs -getmerge /user/CA1/TFIDF/part-r-00000/ TFIDF.csv`

TFIDF.csv file gets created under /home/shradha\_shivani2/MapReduce

- The top 10 users listed in TFIDF.csv file are split into individual text files using splitTopUser.py python program. The resultant shows 10 different text files , one for each user.

```
drwxr-xr-x 1 shradha_shivani2 shradha_shivani2 125193 Oct 27 15:08 TFIDF.csv
drwxr-xr-x 2 shradha_shivani2 shradha_shivani2 4096 Oct 27 15:24 tfidfResults
shradha_shivani2@cluster-hadoop-m:~/MapReduce$ vi splitTopUsers.py
shradha_shivani2@cluster-hadoop-m:~/MapReduce$ vi
vi view viewres vigpg vigr vim vim.basic vim.tiny vimdiff vimtutor vipw virtualenv visudo
shradha_shivani2@cluster-hadoop-m:~/MapReduce$ vi splitTopUsers.py
shradha_shivani2@cluster-hadoop-m:~/MapReduce$ python3 splitTopUsers.py
shradha_shivani2@cluster-hadoop-m:~/MapReduce$ ls -lrt
total 492
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 1559 Oct 26 00:02 mapper1.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 587 Oct 26 00:02 mapper2.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 322 Oct 26 00:02 mapper3.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 412 Oct 26 00:03 mapper4.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 1033 Oct 26 00:03 reducer1.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 769 Oct 26 00:03 reducer2.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 739 Oct 26 00:03 reducer3.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 556 Oct 26 00:08 sortResults.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 176502 Oct 26 00:16 hadoop-streaming-3.2.2.jar
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 344 Oct 26 17:16 TFIDF_Result.txt
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 1090 Oct 27 14:02 mapreduce.sh
drwxr-xr-x 2 shradha_shivani2 shradha_shivani2 4096 Oct 27 14:44 TFIDF
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 125193 Oct 27 15:08 TFIDF.csv
drwxr-xr-x 2 shradha_shivani2 shradha_shivani2 4096 Oct 27 15:24 tfidfResults
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 350 Oct 27 15:45 splitTopUsers.py
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 18417 Oct 27 15:45 6069.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 6422 Oct 27 15:45 4883.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 9659 Oct 27 15:45 9951.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 16866 Oct 27 15:45 51816.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 27143 Oct 27 15:45 49153.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 7432 Oct 27 15:45 95592.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 3814 Oct 27 15:45 89904.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 2935 Oct 27 15:45 87234.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 8968 Oct 27 15:45 63051.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 23537 Oct 27 15:45 179736.txt
shradha_shivani2@cluster-hadoop-m:~/MapReduce$
```

- All the 10 text files are placed in HDFS under /data/userData

```
shradha_shivani2@cluster-hadoop-m:~/MapReduce$ ls -lrt
total 488
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 1559 Oct 26 00:02 mapper1.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 587 Oct 26 00:02 mapper2.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 322 Oct 26 00:02 mapper3.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 412 Oct 26 00:03 mapper4.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 1033 Oct 26 00:03 reducer1.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 769 Oct 26 00:03 reducer2.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 739 Oct 26 00:03 reducer3.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 556 Oct 26 00:08 sortResults.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 176502 Oct 26 00:16 hadoop-streaming-3.2.2.jar
drwxr-xr-x 2 shradha_shivani2 shradha_shivani2 4096 Oct 27 14:44 TFIDF
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 125193 Oct 27 15:08 TFIDF.csv
drwxr-xr-x 2 shradha_shivani2 shradha_shivani2 4096 Oct 27 15:24 tfidfResults
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 350 Oct 27 15:45 splitTopUsers.py
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 18417 Oct 27 15:45 6068.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 6422 Oct 27 15:45 4883.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 9659 Oct 27 15:45 9951.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 16866 Oct 27 15:45 51816.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 27143 Oct 27 15:45 49153.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 7432 Oct 27 15:45 95592.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 3814 Oct 27 15:45 89904.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 2935 Oct 27 15:45 87234.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 8968 Oct 27 15:45 63051.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 23537 Oct 27 15:45 179736.txt
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 1049 Oct 27 15:57 mapreduce.sh
shradha_shivani2@cluster-hadoop-m:~/MapReduce$ hadoop fs -put *.txt /data/userData
shradha_shivani2@cluster-hadoop-m:~/MapReduce$ hadoop fs -ls /data/userData
Found 10 items
-rw-rw-r-- 2 shradha_shivani2 hadoop 23537 2021-10-27 15:59 /data/userData/179736.txt
-rw-rw-r-- 2 shradha_shivani2 hadoop 6422 2021-10-27 15:59 /data/userData/4883.txt
-rw-rw-r-- 2 shradha_shivani2 hadoop 27143 2021-10-27 15:59 /data/userData/49153.txt
-rw-rw-r-- 2 shradha_shivani2 hadoop 16866 2021-10-27 15:59 /data/userData/51816.txt
-rw-rw-r-- 2 shradha_shivani2 hadoop 18417 2021-10-27 15:59 /data/userData/6068.txt
-rw-rw-r-- 2 shradha_shivani2 hadoop 8968 2021-10-27 15:59 /data/userData/63051.txt
-rw-rw-r-- 2 shradha_shivani2 hadoop 2935 2021-10-27 15:59 /data/userData/87234.txt
-rw-rw-r-- 2 shradha_shivani2 hadoop 3814 2021-10-27 15:59 /data/userData/89904.txt
-rw-rw-r-- 2 shradha_shivani2 hadoop 7432 2021-10-27 15:59 /data/userData/95592.txt
-rw-rw-r-- 2 shradha_shivani2 hadoop 9659 2021-10-27 15:59 /data/userData/9951.txt
shradha_shivani2@cluster-hadoop-m:~/MapReduce$
```

- In this section, TFIDF is implemented in Hadoop using Python scripts. Altogether there are 4 mapper and 3 reducer python program files. The implementation takes place in four phases. The first phase uses three mappers and three reducers. The last phase uses the fourth mapper to generate a single file with 10 users word list and its TF-IDF value. The output of one phase is fed as an input for the next phase. Each text file is given as an input to mapreduce.sh script one by one. For each of the 10 text files, the algorithm is run and the resultant text file for each 10 text files gets merged under /home/shradha\_shivani2/MapReduce/tfidfResults/

The mapreducer.sh script consists of below commands:

```
hadoop jar hadoop-streaming-3.2.2.jar -files /home/shradha_shivani2/MapReduce/mapper1.py,/home/shradha_shivani2/MapReduce/reducer1.py -mapper 'python3 mapper1.py' -reducer 'python3 reducer1.py' -input hdfs:///data/userData/$1 -output hdfs:///data/output1
```

```
hadoop jar hadoop-streaming-3.2.2.jar -files /home/shradha_shivani2/MapReduce/mapper2.py,/home/shradha_shivani2/MapReduce/reducer2.py -mapper 'python3 mapper2.py' -reducer 'python3 reducer2.py' -input hdfs:///data/output1/ -output hdfs:///data/output2
```

```
hadoop jar hadoop-streaming-3.2.2.jar -files /home/shradha_shivani2/MapReduce/mapper3.py,/home/shradha_shivani2/MapReduce/reducer3.py -mapper 'python3 mapper3.py' -reducer 'python3 reducer3.py' -input hdfs:///data/output2/ -output hdfs:///data/output3
```

```
hadoop jar hadoop-streaming-3.2.2.jar -files /home/shradha_shivani2/MapReduce/mapper4.py -mapper 'python3 mapper4.py' -input hdfs:///data/output3/ -output hdfs:///data/output4
```

```
hadoop fs -getmerge hdfs:///data/output4/ /home/shradha_shivani2/MapReduce/tfidfResults/$1
```

```
hadoop fs -rm -r hdfs:///data/output*
```

5. The history command shows that individual mapper reducer algorithms

```
764 bash mapreduce.sh 6068.txt
765 cd tfidResults/
766 ls -lrt
767 more 6068.txt
768 !
769 cd ..
770 ls -lrt
771 bash mapreduce.sh 4883.txt
772 ls -lrt
773 bash mapreduce.sh 9951.txt
774 ls -lrt
775 bash mapreduce.sh 51816.txt
776 ls -lrt
777 bash mapreduce.sh 49153.txt
778 ls -lrt
779 bash mapreduce.sh 95592.txt
780 ls -lrt
781 bash mapreduce.sh 89904.txt
782 ls -lrt
783 bash mapreduce.sh 87234.txt
784 ls -lrt
785 bash mapreduce.sh 63051.txt
786 ls -lrt
787 bash mapreduce.sh 179736.txt
```

6. Running mapreduce.sh script for each text file gives the resultant text files in tfidResults directory under /home/shradha\_shivani2/Mapreduce

```
shradha_shivani2@cluster-hadoop-m:~/MapReduce/tfidResults$ ls -lrt
total 452
-rw-r--r-- 1 shradha_shivani2 shradha_shivani2 63547 Oct 27 16:07 6068.txt
-rw-r--r-- 1 shradha_shivani2 shradha_shivani2 28611 Oct 27 16:11 4883.txt
-rw-r--r-- 1 shradha_shivani2 shradha_shivani2 41680 Oct 27 16:17 9951.txt
-rw-r--r-- 1 shradha_shivani2 shradha_shivani2 52629 Oct 27 16:23 51816.txt
-rw-r--r-- 1 shradha_shivani2 shradha_shivani2 80077 Oct 27 16:30 49153.txt
-rw-r--r-- 1 shradha_shivani2 shradha_shivani2 26274 Oct 27 16:36 95592.txt
-rw-r--r-- 1 shradha_shivani2 shradha_shivani2 18033 Oct 27 16:40 89904.txt
-rw-r--r-- 1 shradha_shivani2 shradha_shivani2 14361 Oct 27 16:46 87234.txt
-rw-r--r-- 1 shradha_shivani2 shradha_shivani2 40160 Oct 27 16:51 63051.txt
-rw-r--r-- 1 shradha_shivani2 shradha_shivani2 79283 Oct 27 16:55 179736.txt
shradha_shivani2@cluster-hadoop-m:~/MapReduce/tfidResults$
```

7. Next, sorting program sortResults.py is run to get the expected result. The resultant is captured in TFIDF\_Results\_GCP.txt and stored in /home/shradha\_shivani2/MapReduce

```
shradha_shivani2@cluster-hadoop-m:~/MapReduce$ ls -lrt
total 492
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 1559 Oct 26 00:02 mapper1.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 587 Oct 26 00:02 mapper2.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 322 Oct 26 00:02 mapper3.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 412 Oct 26 00:03 mapper4.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 1033 Oct 26 00:03 reducer1.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 769 Oct 26 00:03 reducer2.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 739 Oct 26 00:03 reducer3.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 586 Oct 26 00:08 sortResults.py
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 176502 Oct 26 00:16 hadoop-streaming-3.2.2.jar
drwxr-xr-x 2 shradha_shivani2 shradha_shivani2 4096 Oct 27 14:44 TFIDF
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 125193 Oct 27 15:08 TFIDF.csv
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 350 Oct 27 15:45 splitTopUsers.py
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 18417 Oct 27 15:45 6068.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 6422 Oct 27 15:45 4883.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 9659 Oct 27 15:45 9951.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 16866 Oct 27 15:45 51816.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 27143 Oct 27 15:45 49153.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 7432 Oct 27 15:45 95592.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 3814 Oct 27 15:45 89904.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 2935 Oct 27 15:45 87234.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 8968 Oct 27 15:45 63051.txt
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 23537 Oct 27 15:45 179736.txt
-rwxr-xr-x 1 shradha_shivani2 shradha_shivani2 1069 Oct 27 16:03 mapreduce.sh
drwxr-xr-x 2 shradha_shivani2 shradha_shivani2 4096 Oct 27 16:55 tfidfResults
-rw-rw-r-- 1 shradha_shivani2 shradha_shivani2 3923 Oct 27 16:57 TFIDF_Results_GCP.txt
shradha_shivani2@cluster-hadoop-m:~/MapReduce$
```

```
95592 Owner User ID
word tfidf_score
20 gitproject 0.051546
127 links 0.050401
25 idea 0.029782
186 path 0.020619
78 naming 0.018328
21 gituser 0.018328
87 pusheverything 0.017182
86 problem 0.016037
236 run 0.016037
226 name 0.012600

9951 Owner User ID
word tfidf_score
119 iterate 0.014815
447 use 0.013757
538 tools 0.012698
104 following 0.011640
222 very 0.011640
444 understand 0.009524
278 old 0.009524
261 generated 0.008466
542 values 0.007407
263 handcraft 0.007407

51816 Owner User ID
word tfidf_score
117 gives 0.018343
552 pyfor 0.017160
325 listbox 0.014201
571 stop 0.014201
136 length 0.013609
359 values 0.011243
551 proper 0.010651
650 noneposttime 0.009467
153 represented 0.008876
423 necessary 0.008876
```