CSP554—Big Data Technologies A20512400

Assignment #3 (Modules 03a & 03b, 15 points)

5) Install the mrjob library on your EMR master node.

a) ssh to the master node (/home/hadoop) as you did in assignment #2
b) Enter the following (note if the first command does not work, try the second)

sudo /usr/bin/pip3.7 install mrjob[aws]

or try:

sudo /usr/bin/pip3 install mrjob[aws]

```
pradaapss — hadoop@ip-172-31-51-250:~ — ssh -i ~/Downloads/bdt_3/emr-key-pair.pem hadoop@ec2-18-234-105-2.compute-1.amazonaws...
18 package(s) needed for security, out of 38 available
Run "sudo yum update" to apply all updates.
-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E:::::EEEEEEEEEE   M:::::M M::M M::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R::::::::::RR
  E:::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR:::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE::::EEEEEEEEE:::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-51-250 ~]$ sudo /usr/bin/pip3.7 install mrjob[aws]
WARNING: Running pip install with root privileges is generally not a good idea. Try `pip3.7 install --user` instead.
Collecting mrjob[aws]
  Downloading mrjob-0.7.4-py2.py3-none-any.whl (439 kB)
     |████████████████████████████████| 439 kB 19.3 MB/s
Requirement already satisfied: PyYAML>=3.10 in /usr/local/lib64/python3.7/site-packages (from mrjob[aws]) (5.4.1)
Collecting boto3>=1.10.0; extra == "aws"
  Downloading boto3-1.24.78-py3-none-any.whl (132 kB)
     |████████████████████████████████| 132 kB 26.8 MB/s
Collecting botocore>=1.13.26; extra == "aws"
  Downloading botocore-1.27.78-py3-none-any.whl (9.1 MB)
     |████████████████████████████████| 9.1 MB 7.4 MB/s
Collecting s3transfer<0.7.0,>=0.6.0
  Downloading s3transfer-0.6.0-py3-none-any.whl (79 kB)
     |████████████████████████████████| 79 kB 5.2 MB/s
Requirement already satisfied: jmespath<2.0.0,>=0.7.1 in /usr/local/lib/python3.7/site-packages (from boto3>=1.10.0; extra == "aws"->mrjob[a
ws]) (1.0.0)
Collecting python-dateutil<3.0.0,>=2.1
  Downloading python_dateutil-2.8.2-py2.py3-none-any.whl (247 kB)
     |████████████████████████████████| 247 kB 14.4 MB/s
Collecting urllib3<1.27,>=1.25.4
  Downloading urllib3-1.26.12-py2.py3-none-any.whl (140 kB)
     |████████████████████████████████| 140 kB 4.9 MB/s
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/site-packages (from python-dateutil<3.0.0,>=2.1->botocore>=1.13.26; extr
a == "aws"->mrjob[aws]) (1.13.0)
Installing collected packages: python-dateutil, urllib3, botocore, s3transfer, boto3, mrjob
  WARNING: The scripts mrjob, mrjob-3 and mrjob-3.7 are installed in '/usr/local/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed boto3-1.24.78 botocore-1.27.78 mrjob-0.7.4 python-dateutil-2.8.2 s3transfer-0.6.0 urllib3-1.26.12
[hadoop@ip-172-31-51-250 ~]$
```

6) Next you will set up to execute the provided WordCount.py map reduce program found in the “Assignments” section of the Blackboard. This is the exact same program we saw in class.

Step 1:

Download the two files “w.data” and “WordCount.py” to your PC or Mac. They are part of the documents included with the assignment.

Step 2:

Note to prevent confusion: the default directory of your Linux account on the Hadoop master node is “/home/hadoop.” But when we want to copy something to HDFS we will sometimes copy it to an HDFS directory beginning with “/user/hadoop.” Be aware, the Linux and HDFS file system path names have nothing to do with one another. Any similarity in naming (such as the use of the directory name “hadoop”) is just coincidental.

Now open another terminal window (but don’t use it to ssh to the master node). This will allow you to access files on your PC or MAC to upload them to the Hadoop master node.

From this terminal window use the secure copy (scp) program to move the WordCount.py file to the /home/hadoop directory of the master node.

```
pradaapss — -zsh — 187×49
pradaapss@dhcp89 ~ % scp -i /Users/pradaapss/Downloads/bdt_3/emr-key-pair.pem /Users/pradaapss/Downloads/bdt_3/WordCount.py hadoop@ec2-18-234-105-2.compute-1.amazonaws.com:/home/hadoop
/etc/profile.d/lang.sh: line 19: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory
WordCount.py                                                                                  100%  402    14.6KB/s   00:00
pradaapss@dhcp89 ~ % scp -i /Users/pradaapss/Downloads/bdt_3/emr-key-pair.pem /Users/pradaapss/Downloads/bdt_3/w.data hadoop@ec2-18-234-105-2.compute-1.amazonaws.com:/home/hadoop
/etc/profile.d/lang.sh: line 19: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory
w.data                                                                                        100%  528    21.5KB/s   00:00
```

Step 3:

Do the same for the assignment file w.data. That is move it to the directory /home/Hadoop on the Hadoop master node Linux file system.

In this case copy the file from the Linux “/home/hadoop” directory to the Hadoop file system (HDFS), say to the directory “/user/hadoop”

```
pradaapss — hadoop@ip-172-31-51-250:~ — ssh -i ~/Downloads/bdt_3/emr-key-pair.pem hadoop@ec2-18-234-105-2.compute-1.amazonaws.com...
command [genericOptions] [commandOptions]

[hadoop@ip-172-31-51-250 ~]$ hadoop fs -ls /
Found 4 items
drwxr-xr-x   - hdfs hdfsadmingroup          0 2022-09-22 06:00 /apps
drwxrwxrwt   - hdfs hdfsadmingroup          0 2022-09-22 06:03 /tmp
drwxr-xr-x   - hdfs hdfsadmingroup          0 2022-09-22 06:00 /user
drwxr-xr-x   - hdfs hdfsadmingroup          0 2022-09-22 06:00 /var
[hadoop@ip-172-31-51-250 ~]$ ls
WordCount.py  w.data
[hadoop@ip-172-31-51-250 ~]$ hadoop fs -copyFromLocal /home/hadoop/WordCount.py /user/hadoop
[hadoop@ip-172-31-51-250 ~]$ hadoop fs -ls /user/hadoop
Found 1 items
-rw-r--r--   1 hadoop hdfsadmingroup        402 2022-09-22 06:20 /user/hadoop/WordCount.py
[hadoop@ip-172-31-51-250 ~]$ hadoop fs -copyFromLocal /home/hadoop/w.data /user/hadoop
[hadoop@ip-172-31-51-250 ~]$ hadoop fs -ls /user/hadoop
Found 2 items
-rw-r--r--   1 hadoop hdfsadmingroup        402 2022-09-22 06:20 /user/hadoop/WordCount.py
-rw-r--r--   1 hadoop hdfsadmingroup        528 2022-09-22 06:21 /user/hadoop/w.data
[hadoop@ip-172-31-51-250 ~]$
```

Step 4:

Now execute the following

```
python WordCount.py -r hadoop hdfs:///user/hadoop/w.data
```

Note there must be three slashes in “hdfs:///” as “hdfs://” indicates that the file you are reading from is in the hadoop file system and the “/user” is the first part of the path to that file. Also note that sometimes copying and pasting this command from the assignment document does not work and it needs to be entered manually.

Check that it produces some reasonable output.

Note, the above command will erase all output files in hdfs. If you want to keep the output use the following command instead:

python WordCount.py -r hadoop hdfs:///user/hadoop/w.data - -output-dir /user/hadoop/some-non-existent-directory

```
pradaapss — hadoop@ip-172-31-51-250:~ — ssh -i ~/Downloads/bdt_3/emr-key-pair.pem hadoop@ec2-18-234-105-2.compute-1.amazonaws.com — 165×55
[hadoop@ip-172-31-51-250 ~]$ python WordCount.py -r hadoop hdfs:///user/hadoop/w.data
No configs found; falling back on auto-configuration
No configs specified for hadoop runner
Looking for hadoop binary in $PATH...
Found hadoop binary: /usr/bin/hadoop
Using Hadoop version 2.10.1
Looking for Hadoop streaming jar in /home/hadoop/contrib...
Looking for Hadoop streaming jar in /usr/lib/hadoop-mapreduce...
Found Hadoop streaming jar: /usr/lib/hadoop-mapreduce/hadoop-streaming.jar
Creating temp directory /tmp/WordCount.hadoop.20220922.062325.075336
uploading working dir files to hdfs:///user/hadoop/tmp/mrjob/WordCount.hadoop.20220922.062325.075336/files/wd...
Copying other local files to hdfs:///user/hadoop/tmp/mrjob/WordCount.hadoop.20220922.062325.075336/files/
Running step 1 of 1...
  packageJobJar: [] [/usr/lib/hadoop/hadoop-streaming-2.10.1-amzn-4.jar] /tmp/streamjob1820944988870503871.jar tmpDir=null
  Connecting to ResourceManager at ip-172-31-51-250.ec2.internal/172.31.51.250:8032
  Connecting to Application History server at ip-172-31-51-250.ec2.internal/172.31.51.250:10200
  Connecting to ResourceManager at ip-172-31-51-250.ec2.internal/172.31.51.250:8032
  Connecting to Application History server at ip-172-31-51-250.ec2.internal/172.31.51.250:10200
  Loaded native gpl library
  Successfully loaded & initialized native-lzo library [hadoop-lzo rev 049362b7cf53ff5f739d6b1532457f2c6cd495e8]
  Total input files to process : 1
  number of splits:4
  Submitting tokens for job: job_1663826484077_0001
  resource-types.xml not found
  Unable to find 'resource-types.xml'.
  Adding resource type - name = memory-mb, units = Mi, type = COUNTABLE
  Adding resource type - name = vcores, units = , type = COUNTABLE
  Submitted application application_1663826484077_0001
  The url to track the job: http://ip-172-31-51-250.ec2.internal:20888/proxy/application_1663826484077_0001/
  Running job: job_1663826484077_0001
  Job job_1663826484077_0001 running in uber mode : false
   map 0% reduce 0%
   map 50% reduce 0%
   map 75% reduce 0%
   map 100% reduce 0%
   map 100% reduce 100%
  Job job_1663826484077_0001 completed successfully
  Output directory: hdfs:///user/hadoop/tmp/mrjob/WordCount.hadoop.20220922.062325.075336/output
Counters: 50
        File Input Format Counters
                Bytes Read=1320
        File Output Format Counters
                Bytes Written=652
        File System Counters
                FILE: Number of bytes read=632
                FILE: Number of bytes written=1129682
                FILE: Number of large read operations=0
                FILE: Number of read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=1768
                HDFS: Number of bytes written=652
                HDFS: Number of large read operations=0
                HDFS: Number of read operations=15
                HDFS: Number of write operations=2
        Job Counters
```

pradaapss — hadoop@ip-172-31-51-250:~ — ssh -i ~/Downloads/bdt_3/emr-key-pair.pem hadoop@ec2-18-234-105-2.compute-1.amazonaws.com — 165×55

```
        Job Counters
                Data-local map tasks=4
                Killed map tasks=1
                Launched map tasks=4
                Launched reduce tasks=1
                Total megabyte-milliseconds taken by all map tasks=74190336
                Total megabyte-milliseconds taken by all reduce tasks=13605888
                Total time spent by all map tasks (ms)=48301
                Total time spent by all maps in occupied slots (ms)=2318448
                Total time spent by all reduce tasks (ms)=4429
                Total time spent by all reduces in occupied slots (ms)=425184
                Total vcore-milliseconds taken by all map tasks=48301
                Total vcore-milliseconds taken by all reduce tasks=4429
        Map-Reduce Framework
                CPU time spent (ms)=5190
                Combine input records=95
                Combine output records=80
                Failed Shuffles=0
                GC time elapsed (ms)=1337
                Input split bytes=448
                Map input records=6
                Map output bytes=891
                Map output materialized bytes=805
                Map output records=95
                Merged Map outputs=4
                Physical memory (bytes) snapshot=2040258560
                Reduce input groups=65
                Reduce input records=80
                Reduce output records=65
                Reduce shuffle bytes=805
                Shuffled Maps =4
                Spilled Records=160
                Total committed heap usage (bytes)=1610612736
                Virtual memory (bytes) snapshot=17872625664
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
job output is in hdfs:///user/hadoop/tmp/mrjob/WordCount.hadoop.20220922.062325.075336/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/WordCount.hadoop.20220922.062325.075336/output...
"a"     3
"all"   1
"an"    1
"and"   1
"are"   1
"as"    4
"available"     1
"be"    3
"by"    1
"cluster"       2
"combine"       1
"contained"     1
```

pradaapss — hadoop@ip-172-31-51-250:~ — ssh -i ~/Downloads/bdt_3/emr-key-pair.pem hadoop@ec2-18-234-105-2.compute-1.amazonaws.com — 165×55

```
"defined"       1
"dependencies"  1
"do"    1
"either"        1
"executed"      1
"explains"      1
"file"  2
"first" 1
"following"     1
"for"   1
"hadoop"        1
"how"   2
"in"    1
"individual"    1
"is"    2
"job"   4
"machine"       1
"map"   1
"more"  2
"mrjob" 1
"must"  1
"nodes" 1
"of"    1
"on"    4
"or"    2
"oriented"      1
"our"   1
"program"       1
"python"        1
"reduce"        1
"reference"     1
"run"   1
"runners"       1
"script"        1
"second"        1
"sections"      1
"see"   1
"submitted"     1
"task"  2
"that"  1
"the"   4
"things"        1
"those" 1
"to"    3
"two"   1
"uploaded"      1
"versions"      1
"well"  1
"when"  1
"will"  1
"within"        1
"writing"       2
"your"  5
Removing HDFS temp directory hdfs:///user/hadoop/tmp/mrjob/WordCount.hadoop.20220922.062325.075336...
Removing temp directory /tmp/WordCount.hadoop.20220922.062325.075336...
```

5) Now slightly modify the WordCount.py program. Call the new program WordCount2.py.

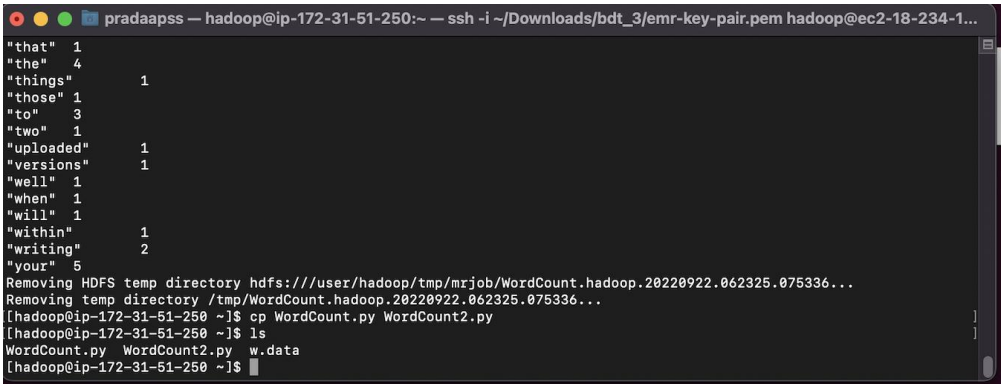
Instead of counting how many words there are in the input documents (w.data), modify the program to count how many words begin with the small letters a-n and how many begin with anything else.

The output file should look something like

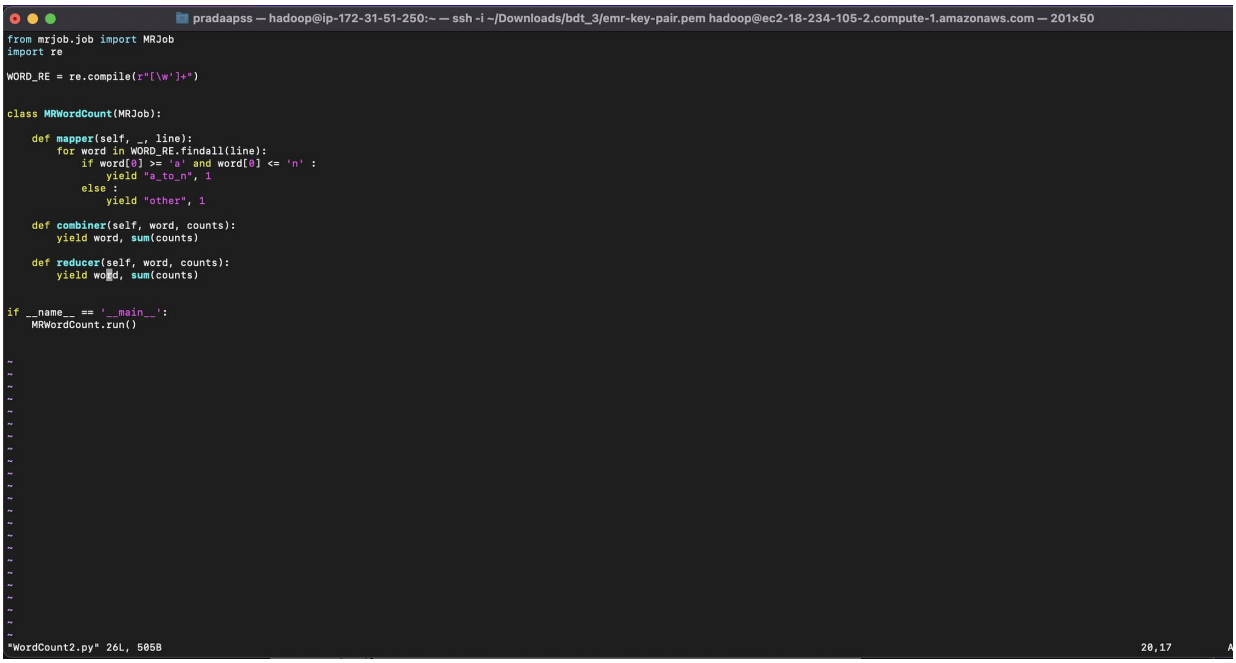
a_to_n, 12

other, 21

Now execute the program and see what happens.



6) (5 points) Submit a copy of this modified program and a screen shot of the results of the program's execution as the output of your assignment.



```
[hadoop@ip-172-31-51-250 ~]$ vim WordCount2.py
[hadoop@ip-172-31-51-250 ~]$ python WordCount2.py -r hadoop hdfs:///user/hadoop/w.data
No configs found; falling back on auto-configuration
No configs specified for hadoop runner
Looking for hadoop binary in $PATH...
Found hadoop binary: /usr/bin/hadoop
Using Hadoop version 2.10.1
Looking for Hadoop streaming jar in /home/hadoop/contrib...
Looking for Hadoop streaming jar in /usr/lib/hadoop-mapreduce...
Found Hadoop streaming jar: /usr/lib/hadoop-mapreduce/hadoop-streaming.jar
Creating temp directory /tmp/WordCount2.hadoop.20220922.063826.317850
uploading working dir files to hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.063826.317850/files/wd...
Copying other local files to hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.063826.317850/files/
Running step 1 of 1...
  packageJobJar: [] [/usr/lib/hadoop/hadoop-streaming-2.10.1-amzn-4.jar] /tmp/streamjob1307815257719306590.jar tmpDir=null
  Connecting to ResourceManager at ip-172-31-51-250.ec2.internal/172.31.51.250:8032
  Connecting to Application History server at ip-172-31-51-250.ec2.internal/172.31.51.250:10200
  Connecting to ResourceManager at ip-172-31-51-250.ec2.internal/172.31.51.250:8032
  Connecting to Application History server at ip-172-31-51-250.ec2.internal/172.31.51.250:10200
  Loaded native gpl library
  Successfully loaded & initialized native-lzo library [hadoop-lzo rev 049362b7cf53ff5f739d6b1532457f2c6cd495e8]
  Total input files to process : 1
  number of splits:4
  Submitting tokens for job: job_1663826484077_0002
  resource-types.xml not found
  Unable to find 'resource-types.xml'.
  Adding resource type - name = memory-mb, units = Mi, type = COUNTABLE
  Adding resource type - name = vcores, units = , type = COUNTABLE
  Submitted application application_1663826484077_0002
  The url to track the job: http://ip-172-31-51-250.ec2.internal:20888/proxy/application_1663826484077_0002/
  Running job: job_1663826484077_0002
  Job job_1663826484077_0002 running in uber mode : false
   map 0% reduce 0%
   map 50% reduce 0%
   map 75% reduce 0%
   map 100% reduce 0%
   map 100% reduce 100%
  Job job_1663826484077_0002 completed successfully
  Output directory: hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.063826.317850/output
Counters: 50
        File Input Format Counters
                Bytes Read=1320
        File Output Format Counters
                Bytes Written=23
        File System Counters
                FILE: Number of bytes read=78
                FILE: Number of bytes written=1128542
                FILE: Number of large read operations=0
                FILE: Number of read operations=0
                FILE: Number of write operations=0
```

```
                HDFS: Number of read operations=15
                HDFS: Number of write operations=2
        Job Counters
                Data-local map tasks=4
                Killed map tasks=1
                Launched map tasks=4
                Launched reduce tasks=1
                Total megabyte-milliseconds taken by all map tasks=60496896
                Total megabyte-milliseconds taken by all reduce tasks=12484608
                Total time spent by all map tasks (ms)=39386
                Total time spent by all maps in occupied slots (ms)=1890528
                Total time spent by all reduce tasks (ms)=4064
                Total time spent by all reduces in occupied slots (ms)=390144
                Total vcore-milliseconds taken by all map tasks=39386
                Total vcore-milliseconds taken by all reduce tasks=4064
        Map-Reduce Framework
                CPU time spent (ms)=5980
                Combine input records=95
                Combine output records=6
                Failed Shuffles=0
                GC time elapsed (ms)=1125
                Input split bytes=448
                Map input records=6
                Map output bytes=996
                Map output materialized bytes=144
                Map output records=95
                Merged Map outputs=4
                Physical memory (bytes) snapshot=2067492864
                Reduce input groups=2
                Reduce input records=6
                Reduce output records=2
                Reduce shuffle bytes=144
                Shuffled Maps =4
                Spilled Records=12
                Total committed heap usage (bytes)=1637875712
                Virtual memory (bytes) snapshot=17884938240
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
job output is in hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.063826.317850/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.063826.317850/output...
"a_to_n"        46
"other" 49
Removing HDFS temp directory hdfs:///user/hadoop/tmp/mrjob/WordCount2.hadoop.20220922.063826.317850...
Removing temp directory /tmp/WordCount2.hadoop.20220922.063826.317850...
[hadoop@ip-172-31-51-250 ~]$
```

7) Now do the same as the above for the files Salaries.py and Salaries.tsv. The “.tsv” file holds department and salary information for Baltimore municipal workers. Have a look at Salaries.py for the layout of the “.tsv” file and how to read it in to our map reduce program

```
pradaapss — hadoop@ip-172-31-51-250:~ — ssh -i ~/Downloads/bdt_3/emr-key-pair.pem hadoop@ec2-18-234-105-2.compute-1.amazonaws.com — 202×56
[hadoop@ip-172-31-51-250 ~]$ hadoop fs -ls /user/hadoop
Found 6 items
-rw-r--r--   1 hadoop hdfsadmingroup        411 2022-09-22 06:49 /user/hadoop/Salaries.py
-rw-r--r--   1 hadoop hdfsadmingroup    1538148 2022-09-22 06:44 /user/hadoop/Salaries.tsv
-rw-r--r--   1 hadoop hdfsadmingroup        402 2022-09-22 06:20 /user/hadoop/WordCount.py
drwxr-xr-x   - hadoop hdfsadmingroup          0 2022-09-22 06:23 /user/hadoop/tmp
-rw-r--r--   1 hadoop hdfsadmingroup    2438233 2022-09-22 06:43 /user/hadoop/u.data
-rw-r--r--   1 hadoop hdfsadmingroup        528 2022-09-22 06:21 /user/hadoop/w.data
[hadoop@ip-172-31-51-250 ~]$ ls
Salaries.py  Salaries.tsv  WordCount.py  WordCount2.py  u.data  w.data
```

8) Execute the Salaries.py program to make sure it works. It should print out how many workers share each job title.

```
pradaapss — hadoop@ip-172-31-51-250:~ — ssh -i ~/Downloads/bdt_3/emr-key-pair.pem hadoop@ec2-18-234-105-2.compute-1.amazonaws.com — 160×42
[hadoop@ip-172-31-51-250 ~]$ cp Salaries.py Salaries2.py
[hadoop@ip-172-31-51-250 ~]$ vim Salaries2.py
[hadoop@ip-172-31-51-250 ~]$ python Salaries2.py -r hadoop hdfs:///user/hadoop/Salaries.tsv
No configs found; falling back on auto-configuration
No configs specified for hadoop runner
Looking for hadoop binary in $PATH...
Found hadoop binary: /usr/bin/hadoop
Using Hadoop version 2.10.1
Looking for Hadoop streaming jar in /home/hadoop/contrib...
Looking for Hadoop streaming jar in /usr/lib/hadoop-mapreduce...
Found Hadoop streaming jar: /usr/lib/hadoop-mapreduce/hadoop-streaming.jar
Creating temp directory /tmp/Salaries2.hadoop.20220922.070240.985953
uploading working dir files to hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.070240.985953/files/wd.
Copying other local files to hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.070240.985953/files/
Running step 1 of 1...
  packageJobJar: [] [/usr/lib/hadoop/hadoop-streaming-2.10.1-amzn-4.jar] /tmp/streamjob6429260499980111358.jar tmpDir=null
  Connecting to ResourceManager at ip-172-31-51-250.ec2.internal/172.31.51.250:8032
  Connecting to Application History server at ip-172-31-51-250.ec2.internal/172.31.51.250:10200
  Connecting to ResourceManager at ip-172-31-51-250.ec2.internal/172.31.51.250:8032
  Connecting to Application History server at ip-172-31-51-250.ec2.internal/172.31.51.250:10200
  Loaded native gpl library
  Successfully loaded & initialized native-lzo library [hadoop-lzo rev 049362b7cf53ff5f739d6b1532457f2c6cd495e8]
  Total input files to process : 1
  number of splits:4
  Submitting tokens for job: job_1663826484077_0004
  resource-types.xml not found
  Unable to find 'resource-types.xml'.
  Adding resource type - name = memory-mb, units = Mi, type = COUNTABLE
  Adding resource type - name = vcores, units = , type = COUNTABLE
  Submitted application application_1663826484077_0004
  The url to track the job: http://ip-172-31-51-250.ec2.internal:20888/proxy/application_1663826484077_0004/
  Running job: job_1663826484077_0004
  Job job_1663826484077_0004 running in uber mode : false
   map 0% reduce 0%
   map 50% reduce 0%
   map 100% reduce 0%
   map 100% reduce 100%
  Job job_1663826484077_0004 completed successfully
  Output directory: hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.070240.985953/output
Counters: 50
        File Input Format Counters
                Bytes Read=1564110
```

```
Screenshot 2022-09-22 at 1.55.02 AM
Terminal  Shell  Edit  View  Window  Help                                  Thu 22 Sep  1:55 AM
pradaapss — hadoop@ip-172-31-51-250:~ — ssh -i ~/Downloads/bdt_3/emr-key-pair.pem hadoop@ec2-18-234-105-2.compute-1.amazonaws.com — 202×56
"UTILITY METER READER II"      12
"UTILITY METER READER SUPT II"  1
"UTILITY METER READER SUPV"     5
"UTILITY POLICY ANALYST"        1
"Urban Forester"        7
"VEHICLE IDENTIFICATION INSPECT"        1
"VEHICLE PROCESSOR"     9
"VICE PRESIDENT CITY COUNCIL"   1
"VICTIM/WITNESS COORDINATOR SAO"        11
"VOLUNTEER SERVICE COORDINATOR" 1
"VOLUNTEER SERVICE WORKER"      1
"Volunteer Service Coordinator" 1
"WASTE WATER PLANT COORDINATOR" 2
"WASTE WATER PLANT MANAGER"     2
"WASTE WATER PLANT OPNS SUPV"   2
"WATER PUMPING ASST MANAGER"    2
"WATER SERVICE INSPECTOR"       4
"WATER SERVICE REPRESENTATIVE"  12
"WATER TREATMENT ASST MANAGER"  2
"WATER TREATMENT TECHNICIAN II" 17
"WATER TREATMENT TECHNICIAN III"        8
"WATER TREATMENT TECHNICIAN SUP"        6
"WATERSHED MAINT SUPV"  3
"WATERSHED MANAGER"     1
"WATERSHED RANGER II"   5
"WATERSHED RANGER III"  3
"WATERSHED RANGER SUPERVISOR"   1
"WEB DEVELOPER" 1
"WELDER"        8
"WHITEPRINT MACHINE OPR"        1
"WORK STUDY STUDENT"    18
"WORKER'S COMPENSATION CONTRACT"        1
"WWW Chief of Engineering"      1
"WWW Division  Manager I"       1
"WWW Division Manager II"       5
"Waste Water Maint Mgr Instrum" 1
"Waste Water Maintenance Mgr Me"        1
"Waste Water Opns Tech II Pump" 10
"Waste Water Opns Tech II Sanit"        81
"Waste Water Tech Supv I Pump"  6
"Waste Water Tech Supv II Pump" 1
"Waste Water Tech Supv II Sanit"        10
"Waste Water Techn Supv I Sanit"        19
"Water Systems Pumping Supv"    1
"Water Systems Treatment Manage"        1
"Water Systems Treatment Supv"  2
"YOUTH DEVELOPMENT TECH"        3
"ZONING ADMINISTRATOR"  1
"ZONING APPEALS ADVISOR BMZA"   1
"ZONING APPEALS OFFICER"        1
"ZONING ENFORCEMENT OFFICER"    1
"ZONING EXAMINER I"     2
"ZONING EXAMINER II"    1
Removing HDFS temp directory hdfs:///user/hadoop/tmp/mrjob/Salaries.hadoop.20220922.065213.756206...
Removing temp directory /tmp/Salaries.hadoop.20220922.065213.756206...
[hadoop@ip-172-31-51-250 ~]$
```

9) Now modify the Salaries.py program. Call it Salaries2.py

Instead of counting the number of workers per department, change the program to provide the number of workers having High, Medium or Low annual salaries. This is defined as follows:

High	100,000.00 and above
Medium	50,000.00 to 99,999.99
Low	0.00 to 49,999.99

The output of the program should be something like the following (in any order):

High 20

Medium 30

Low 10

Some important hints:

- The annual salary is a string that will need to be converted to a float.
- The mapper should output tuples with one of three keys depending on the annual salary: High, Medium and Low
- The value part of the tuple is not a salary. (What should it be?)

Now execute the program and see what happens.

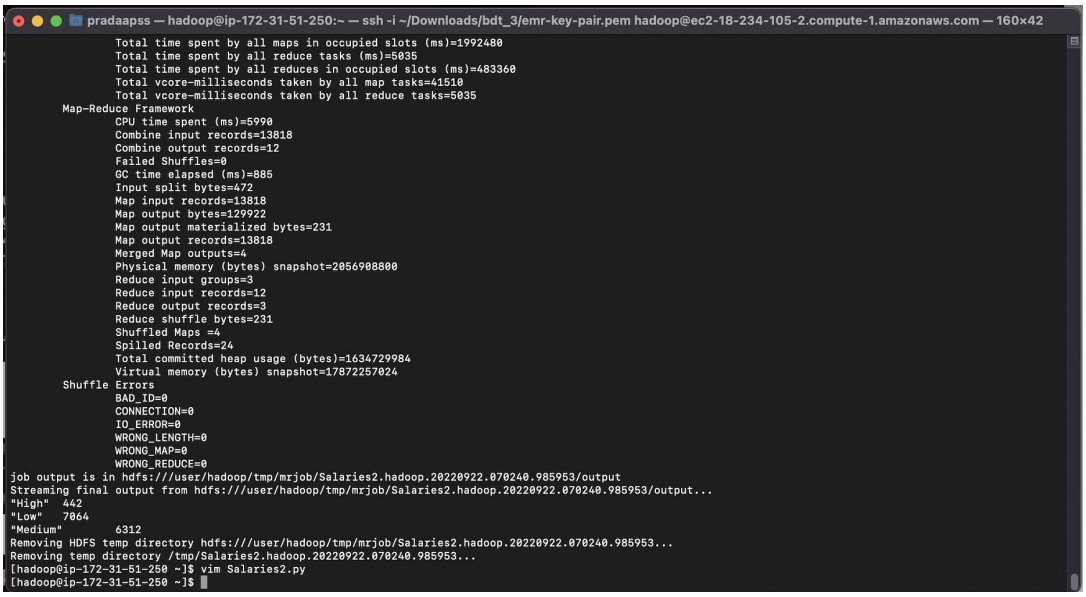
9) (5 points) Submit a copy of this modified program and a screen shot of the results of the program's execution as the output of your assignment.

Salaries2.py

```python
from mrjob.job import MRJob

class MRSalaries(MRJob):

    def mapper(self, _, line):
        (name,jobTitle,agencyID,agency,hireDate,annualSalary,grossPay) = line.split('\t')
        if(float(annualSalary) >= 0.0 and float(annualSalary) <= 49999.99 :
            yield "Low", 1
        if(float(annualSalary) >= 50000.00 and float(annualSalary) <= 99999.99 :
             yield "Medium", 1
        else :
             yield "High", 1

    def combiner(self, annualSalary, counts):
        yield annualSalary, sum(counts)

    def reducer(self, annualSalary, counts):
        yield annualSalary, sum(counts)


if __name__ == '__main__':
    MRSalaries.run()
```

pradaapss — hadoop@ip-172-31-51-250:~ — ssh -i ~/Downloads/bdt_3/emr-key-pair.pem hadoop@ec2-18-234-105-2.compute-1.amazonaws.com — 160×42

```
[hadoop@ip-172-31-51-250 ~]$ cp Salaries.py Salaries2.py
[hadoop@ip-172-31-51-250 ~]$ vim Salaries2.py
[hadoop@ip-172-31-51-250 ~]$ python Salaries2.py -r hadoop hdfs:///user/hadoop/Salaries.tsv
No configs found; falling back on auto-configuration
No configs specified for hadoop runner
Looking for hadoop binary in $PATH...
Found hadoop binary: /usr/bin/hadoop
Using Hadoop version 2.10.1
Looking for Hadoop streaming jar in /home/hadoop/contrib...
Looking for Hadoop streaming jar in /usr/lib/hadoop-mapreduce...
Found Hadoop streaming jar: /usr/lib/hadoop-mapreduce/hadoop-streaming.jar
Creating temp directory /tmp/Salaries2.hadoop.20220922.070240.985953
uploading working dir files to hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.070240.985953/files/wd...
Copying other local files to hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.070240.985953/files/
Running step 1 of 1...
  packageJobJar: [] [/usr/lib/hadoop/hadoop-streaming-2.10.1-amzn-4.jar] /tmp/streamjob6429260499980111358.jar tmpDir=null
  Connecting to ResourceManager at ip-172-31-51-250.ec2.internal/172.31.51.250:8032
  Connecting to Application History server at ip-172-31-51-250.ec2.internal/172.31.51.250:10200
  Connecting to ResourceManager at ip-172-31-51-250.ec2.internal/172.31.51.250:8032
  Connecting to Application History server at ip-172-31-51-250.ec2.internal/172.31.51.250:10200
  Loaded native gpl library
  Successfully loaded & initialized native-lzo library [hadoop-lzo rev 049362b7cf53ff5f739d6b1532457f2c6cd495e8]
  Total input files to process : 1
  number of splits:4
  Submitting tokens for job: job_1663826484077_0004
  resource-types.xml not found
  Unable to find 'resource-types.xml'.
  Adding resource type - name = memory-mb, units = Mi, type = COUNTABLE
  Adding resource type - name = vcores, units = , type = COUNTABLE
  Submitted application application_1663826484077_0004
  The url to track the job: http://ip-172-31-51-250.ec2.internal:20888/proxy/application_1663826484077_0004/
  Running job: job_1663826484077_0004
  Job job_1663826484077_0004 running in uber mode : false
   map 0% reduce 0%
   map 50% reduce 0%
   map 100% reduce 0%
   map 100% reduce 100%
  Job job_1663826484077_0004 completed successfully
  Output directory: hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.070240.985953/output
Counters: 50
        File Input Format Counters
                Bytes Read=1564110
```



```
pradaapss — hadoop@ip-172-31-51-250:~ — ssh -i ~/Downloads/bdt_3/emr-key-pair.pem hadoop@ec2-18-234-105-2.compute-1.amazonaws.com — 160×42
                Total time spent by all maps in occupied slots (ms)=1992480
                Total time spent by all reduce tasks (ms)=5035
                Total time spent by all reduces in occupied slots (ms)=483360
                Total vcore-milliseconds taken by all map tasks=41510
                Total vcore-milliseconds taken by all reduce tasks=5035
        Map-Reduce Framework
                CPU time spent (ms)=5990
                Combine input records=13818
                Combine output records=12
                Failed Shuffles=0
                GC time elapsed (ms)=885
                Input split bytes=472
                Map input records=13818
                Map output bytes=129922
                Map output materialized bytes=231
                Map output records=13818
                Merged Map outputs=4
                Physical memory (bytes) snapshot=2056908800
                Reduce input groups=3
                Reduce input records=12
                Reduce output records=3
                Reduce shuffle bytes=231
                Shuffled Maps =4
                Spilled Records=24
                Total committed heap usage (bytes)=1634729984
                Virtual memory (bytes) snapshot=17872257024
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
job output is in hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.070240.985953/output
Streaming final output from hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.070240.985953/output...
"High"  442
"Low"   7064
"Medium"        6312
Removing HDFS temp directory hdfs:///user/hadoop/tmp/mrjob/Salaries2.hadoop.20220922.070240.985953...
Removing temp directory /tmp/Salaries2.hadoop.20220922.070240.985953...
[hadoop@ip-172-31-51-250 ~]$ vim Salaries2.py
[hadoop@ip-172-31-51-250 ~]$
```

11) Now copy the file u.data from the assignment to /user/hadoop. This is similar to the file used for some examples in Module 03b. NOTE: unlike the slide deck examples, this version of u.data has fields separated by commas and not tabs.

12) (5 points) Review the slides 55-61 in lecture notes Module 3b. Now write a program to perform the task of outputting a count of the number of movies each user (identified via their user id) reviewed.

Output might look something like the following:

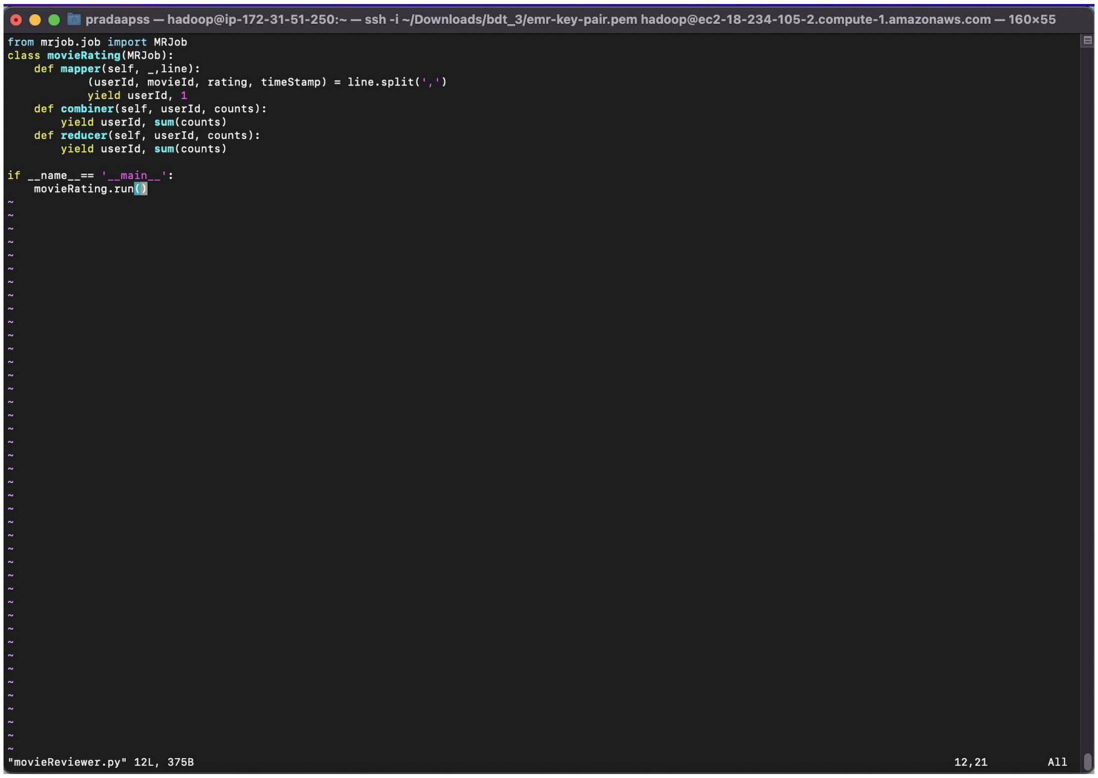
186: 2

192: 2

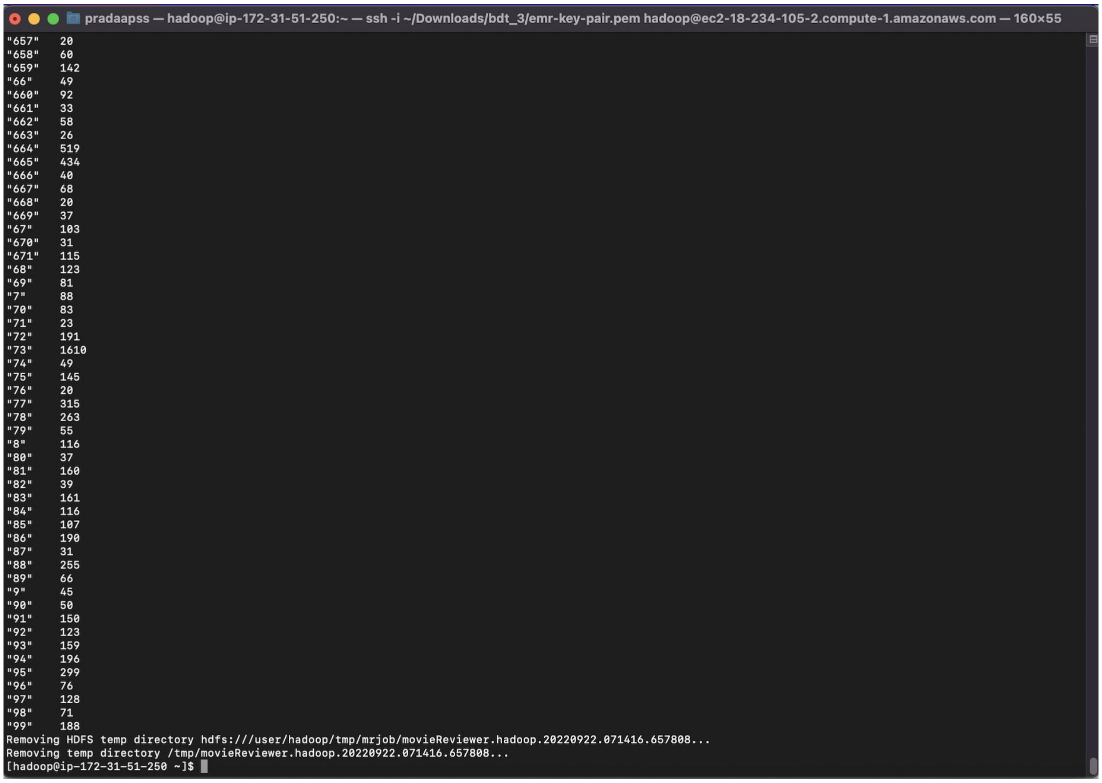
112: 1

etc.

Submit a copy of this program and a screen shot of the results of the program's execution (only 10 lines or so of the result) as the output of your assignment.



```
from mrjob.job import MRJob
class movieRating(MRJob):
    def mapper(self, _,line):
            (userId, movieId, rating, timeStamp) = line.split(',')
            yield userId, 1
    def combiner(self, userId, counts):
        yield userId, sum(counts)
    def reducer(self, userId, counts):
        yield userId, sum(counts)

if __name__== '__main__':
    movieRating.run()
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"movieReviewer.py" 12L, 375B                                                                                                      12,21         All
```



```
"657"   20
"658"   60
"659"   142
"66"    49
"660"   92
"661"   33
"662"   58
"663"   26
"664"   519
"665"   434
"666"   40
"667"   68
"668"   20
"669"   37
"67"    103
"670"   31
"671"   115
"68"    123
"69"    81
"7"     88
"70"    83
"71"    23
"72"    191
"73"    1610
"74"    49
"75"    145
"76"    20
"77"    315
"78"    263
"79"    55
"8"     116
"80"    37
"81"    160
"82"    39
"83"    161
"84"    116
"85"    107
"86"    190
"87"    31
"88"    255
"89"    66
"9"     45
"90"    50
"91"    150
"92"    123
"93"    159
"94"    196
"95"    299
"96"    76
"97"    128
"98"    71
"99"    188
Removing HDFS temp directory hdfs:///user/hadoop/tmp/mrjob/movieReviewer.hadoop.20220922.071416.657808...
Removing temp directory /tmp/movieReviewer.hadoop.20220922.071416.657808...
[hadoop@ip-172-31-51-250 ~]$ 
```

13) Remember to terminate your EMR cluster and remove your S3 bucket.

Cluster Terminated!!