

Siddharth Chaphekar

CSC 555 – Mining Big Data

Summer 2019

Assignment 1

PART 1 – Computation Problems

a) Discrete Computations

- $2^{10} = \mathbf{1024}$
- $4^5 = \mathbf{1024}$
- $8^5 = \mathbf{32768}$
- $837 \text{ MOD } 100 = \mathbf{37}$
- $842 \text{ MOD } 20 = \mathbf{2}$
- $16 \text{ MOD } 37 = \mathbf{16}$
- $37 \text{ MOD } 16 = \mathbf{5}$

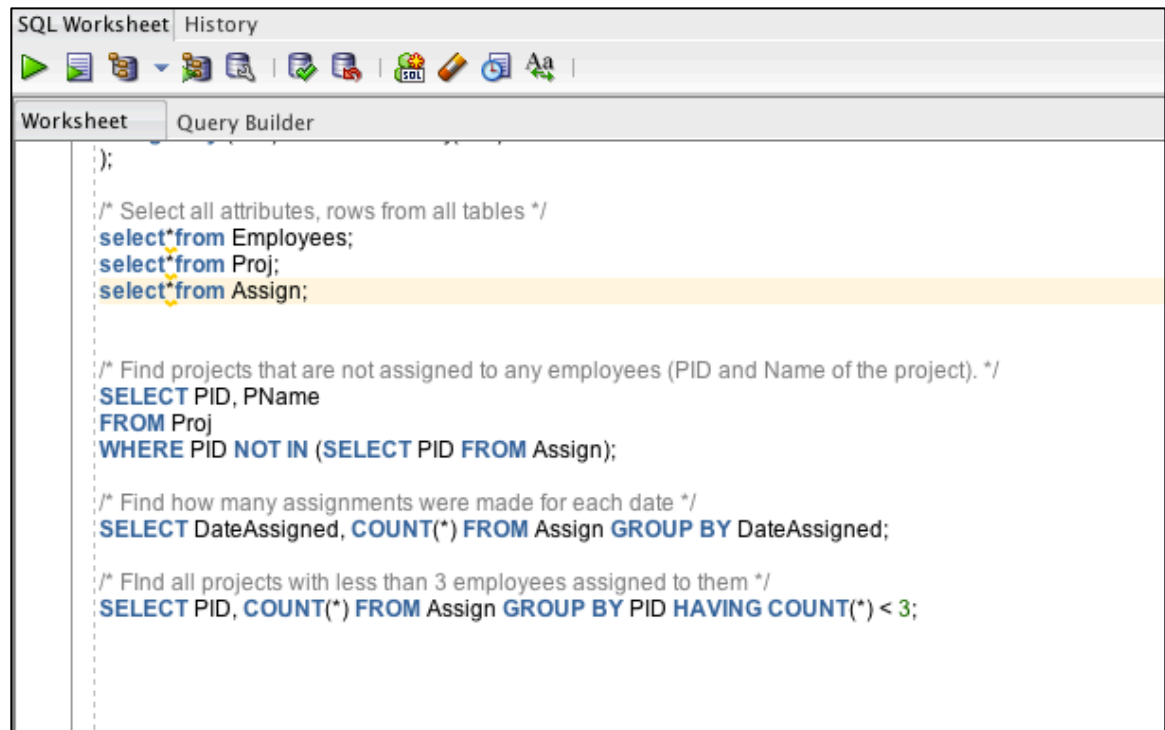
b) Vector Calculations

- $V2 - V1 = (2-1, 1-2, 2-3) = \mathbf{(1, -1, -1)}$
- $V1 + V1 = (1+1, 2+2, 3+3) = \mathbf{(2, 4, 6)}$
- $|V1| = \text{SQRT}(1^2 + 2^2 + 3^2) = \text{SQRT}(14) = \sqrt{\mathbf{14}}$
- $|V2| = \text{SQRT}(2^2 + 1^2 + 2^2) = \text{SQRT}(9) = \mathbf{3}$
- $M * V2 = \mathbf{(11, 8, 6)}$
- $M^2 = \mathbf{[(8, 4, 14), (6, 5, 11), (4, 1, 7)]}$
- $M^3 = \mathbf{[(34, 16, 60), (28, 16, 50), (16, 6, 28)]}$

c) Coin Flip Probabilities

- $\text{HTHT} = 0.6 \times 0.4 \times 0.6 \times 0.4 = \mathbf{0.0576}$
- $\text{THTT} = 0.4 \times 0.6 \times 0.4 \times 0.4 = \mathbf{0.0384}$
- $\text{Exactly 1 Head} = \{\text{HTTT}, \text{THTT}, \text{TTHT}, \text{TTTH}\} = \mathbf{0.1536}$
- $\text{Exactly 1 Tail} = \{\text{THHH}, \text{HTHH}, \text{HHTH}, \text{HHHT}\} = 4 \times 0.0864 = \mathbf{0.3456}$

d) SQL Queries



The screenshot shows an 'SQL Worksheet' window with a toolbar at the top containing icons for running queries, saving, and other database operations. Below the toolbar are two tabs: 'Worksheet' and 'Query Builder'. The 'Worksheet' tab is active, displaying a list of SQL queries. The first query is a simple selection from three tables. The second query finds projects not assigned to any employees. The third query counts assignments by date. The fourth query finds projects with fewer than three assignments.

```
);

/* Select all attributes, rows from all tables */
select*from Employees;
select*from Proj;
select*from Assign;

/* Find projects that are not assigned to any employees (PID and Name of the project). */
SELECT PID, PName
FROM Proj
WHERE PID NOT IN (SELECT PID FROM Assign);

/* Find how many assignments were made for each date */
SELECT DateAssigned, COUNT(*) FROM Assign GROUP BY DateAssigned;

/* Find all projects with less than 3 employees assigned to them */
SELECT PID, COUNT(*) FROM Assign GROUP BY PID HAVING COUNT(*) < 3;
```

e) Mining of Massive Datasets, Ex. 1.3.3

The hash function $h(x)$ will only give values between 0 and 15 since it is $x \text{ MOD } 15$. If we choose $c = 1$, then we can get hash keys that are divided equally into all of the buckets, but we choose $c = 2$ then we only get hash key distributions across buckets 0,2,4,6,8, ... etc. Similar for $c = 3$, $c = 4$, etc. But for $c = 16$, we again get equal distribution, so **c should be one more than multiple of bucket size**. The number should be co-prime. So 2 and 4 and 16 are indeed good, but 3 is actually bad because 15 is 3×5 .

f) MapReduce Implementation

The Map function transforms input into meaningful key-value pairs. For example, if you were dealing with a dataset containing cities and states, it would combine those two attributes to form a key; the value would be a combination of other attributes related to that city. So, a MapReduce job generally divides the input data into smaller blocks that are processed in parallel by the map tasks.

PART 2 – Linux Problems

1) Contents of copied file displayed on terminal screen

```
Desktop — ec2-user@ip-172-31-10-146:~ — ssh -i schaphekar.pem ec2-user@ec2-54-153-117-2...
[Siddharths-MBP:~ Siddharth$ chmod 600 private_key_file.pem]
chmod: private_key_file.pem: No such file or directory
[Siddharths-MBP:~ Siddharth$ ssh -i "schaphekar.pem" ec2-user@ec2-54-153-117-22.us-west-1.compute.amazonaws.com]
Warning: Identity file schaphekar.pem not accessible: No such file or directory.
The authenticity of host 'ec2-54-153-117-22.us-west-1.compute.amazonaws.com (54.153.117.22)' can't be established
.
ECDSA key fingerprint is SHA256:sFJJ0/kQ0DLuQtFC3UnwhitKukPGcsHCrTnto6RprV4.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-54-153-117-22.us-west-1.compute.amazonaws.com' (ECDSA) to the list of known hosts
.
ec2-user@ec2-54-153-117-22.us-west-1.compute.amazonaws.com: Permission denied (publickey,gssapi-keyex,gssapi-with
-mic).
[Siddharths-MBP:~ Siddharth$ chmod 400 schaphekar.pem]
chmod: schaphekar.pem: No such file or directory
[Siddharths-MBP:~ Siddharth$ cd Desktop]
[Siddharths-MBP:Desktop Siddharth$ chmod 400 schaphekar.pem]
[Siddharths-MBP:Desktop Siddharth$ ssh -i "schaphekar.pem" ec2-user@ec2-54-153-117-22.us-west-1.compute.amazonaws.
com]

  __|  __|_ )
  _| (  /  Amazon Linux 2 AMI
  ___|\\___|___|

https://aws.amazon.com/amazon-linux-2/
[[ec2-user@ip-172-31-10-146 ~]$ nano myfile.txt
[[ec2-user@ip-172-31-10-146 ~]$ ls
myfile.txt
[[ec2-user@ip-172-31-10-146 ~]$ cat myfile.txt
This is my text file for CSC555.
[[ec2-user@ip-172-31-10-146 ~]$ cp myfile.txt mycopy.txt
[[ec2-user@ip-172-31-10-146 ~]$ ls
mycopy.txt  myfile.txt
[[ec2-user@ip-172-31-10-146 ~]$ nano mycopy.txt
[[ec2-user@ip-172-31-10-146 ~]$ cat mycopy.txt
This is my other text file for CSC555.
[[ec2-user@ip-172-31-10-146 ~]$
```

2) Files in CSC555 directory

```
Desktop — ec2-user@ip-172-31-10-146:~/CSC555 — ssh -i schaphekar.pem ec2-user@ec2-54-1...

__|__|__| )
_| ( / Amazon Linux 2 AMI
---|\\___|___|

https://aws.amazon.com/amazon-linux-2/
[[ec2-user@ip-172-31-10-146 ~]$ nano myfile.txt
[[ec2-user@ip-172-31-10-146 ~]$ ls
myfile.txt
[[ec2-user@ip-172-31-10-146 ~]$ cat myfile.txt
This is my text file for CSC555.
[[ec2-user@ip-172-31-10-146 ~]$ cp myfile.txt mycopy.txt
[[ec2-user@ip-172-31-10-146 ~]$ ls
mycopy.txt  myfile.txt
[[ec2-user@ip-172-31-10-146 ~]$ nano mycopy.txt
[[ec2-user@ip-172-31-10-146 ~]$ cat mycopy.txt
This is my other text file for CSC555.
[[ec2-user@ip-172-31-10-146 ~]$ cp myfile.txt filedelete.txt
[[ec2-user@ip-172-31-10-146 ~]$ ls
filedelete.txt  mycopy.txt  myfile.txt
[[ec2-user@ip-172-31-10-146 ~]$ rm filedelete.txt
[[ec2-user@ip-172-31-10-146 ~]$ ls
mycopy.txt  myfile.txt
[[ec2-user@ip-172-31-10-146 ~]$ mkdir CSC555
[[ec2-user@ip-172-31-10-146 ~]$ cd CSC555
[[ec2-user@ip-172-31-10-146 CSC555]$ pwd
/home/ec2-user/CSC555
[[ec2-user@ip-172-31-10-146 CSC555]$ cd ..
[[ec2-user@ip-172-31-10-146 ~]$ ls
CSC555  mycopy.txt  myfile.txt
[[ec2-user@ip-172-31-10-146 ~]$ mv myfile.txt CSC555/
[[ec2-user@ip-172-31-10-146 ~]$ mv mycopy.txt CSC555/
[[ec2-user@ip-172-31-10-146 ~]$ cd CSC555
[[ec2-user@ip-172-31-10-146 CSC555]$ ls
mycopy.txt  myfile.txt
[[ec2-user@ip-172-31-10-146 CSC555]$
```

3) Unzipping myzipfile.zip

```
[[ec2-user@ip-172-31-10-146 CSC555]$ mv myzipfile.zip /home/ec2-user/
[[ec2-user@ip-172-31-10-146 CSC555]$ unzip myzipfile.zip
unzip: cannot find or open myzipfile.zip, myzipfile.zip.zip or myzipfile.zip.ZIP.
[[ec2-user@ip-172-31-10-146 CSC555]$ cd
[[ec2-user@ip-172-31-10-146 ~]$ unzip myzipfile.zip
Archive:  myzipfile.zip
  extracting: mycopy.txt
  extracting: myfile.txt
[[ec2-user@ip-172-31-10-146 ~]$
```

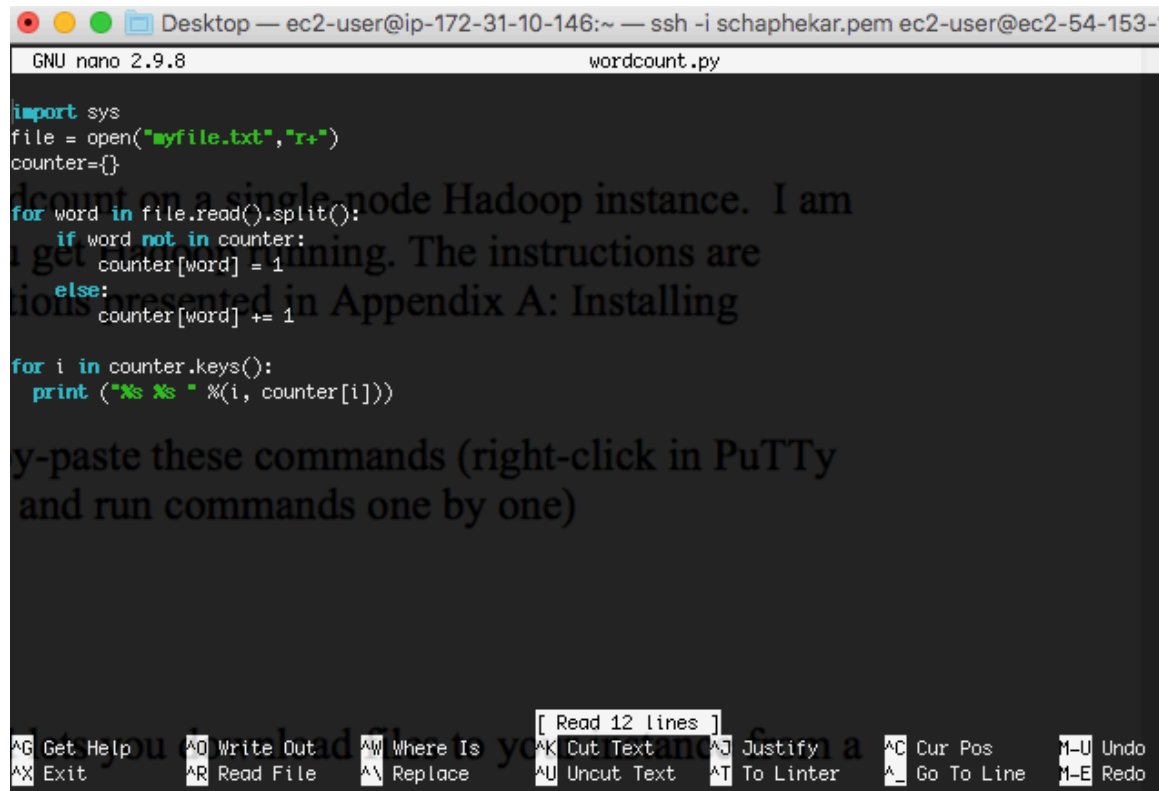
3) Size of grail file = 9 bytes + Permission denied

```
Desktop — ec2-user@ip-172-31-10-146:~ — ssh -i schaphekar.pem ec2-user@ec2-54-153-117-2...
7
8
9
10      Written as was performed in the feature film
11      -----
12      Transcribed by Adam R. Jones
13      Helpers: Hans ten Cate, Rich Jackman, Malcolm Dickinson, Bret Shefter
14
15
16      Monty Python and the Holy Grail - (c) 1974 - Python (Monty) Pictures, Ltd.
17
18
19
20      |-----|
21      | The Cast: (in order of appearance) |
22      |-----|
23
24      KING ARTHUR  Graham Chapman
25      PATSY        Terry Gilliam
26      SOLDIER #1   Michael Palin
27      SOLDIER #2   John Cleese
28      CART-MASTER Eric Idle
[ec2-user@ip-172-31-10-146 ~]$ cat myfile.txt > redirect1.txt
[ec2-user@ip-172-31-10-146 ~]$ ls -lh > redirect2.txt
[ec2-user@ip-172-31-10-146 ~]$ cat mycopy.txt >> myfile.txt
[ec2-user@ip-172-31-10-146 ~]$ chmod u-r myfile.txt
[ec2-user@ip-172-31-10-146 ~]$ cat myfile.txt
cat: myfile.txt: Permission denied
[ec2-user@ip-172-31-10-146 ~]$
```

4) Python word counter, executed on myfile.txt

```
[ec2-user@ip-172-31-10-146 ~]$ nano wordcount.py
[ec2-user@ip-172-31-10-146 ~]$ python wordcount.py
for 2
This 2
text 2
is 2
other 1
file 2
my 2
CSC555: 2
[ec2-user@ip-172-31-10-146 ~]$
```

5) Python word counter code



The screenshot shows a terminal window with a nano editor. The title bar indicates the user is 'ec2-user' on a machine with IP '172-31-10-146', connected via SSH. The editor is editing a file named 'wordcount.py'. The Python code is as follows:

```
import sys
file = open("myfile.txt", "r+")
counter={}

for word in file.read().split():
    if word not in counter:
        counter[word] = 1
    else:
        counter[word] += 1

for i in counter.keys():
    print ("%s %s" % (i, counter[i]))
```

The bottom of the screen shows the nano editor's command palette with various shortcuts like ^G for Get Help, ^O for Write Out, ^W for Where Is, etc.

PART 3 – Word Count Lab

1) Verifying that the file was uploaded to HDFS

```
Desktop — ec2-user@ip-172-31-10-146:~ — ssh -i schaphekar.pem ec2-user@ec2-54-153-117-22.us-west-1.compute.am...
[ec2-user@ip-172-31-10-146 ~]$ start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /home/ec2-user/hadoop-2.6.4/logs/yarn-ec2-user-resourcemanager-ip-172-31-10-146.us-west-1.compute.int
ernal.out
localhost: starting nodemanager, logging to /home/ec2-user/hadoop-2.6.4/logs/yarn-ec2-user-nodemanager-ip-172-31-10-146.us-west-1.compute.i
nternal.out
[ec2-user@ip-172-31-10-146 ~]$ mr-jobhistory-daemon.sh start historyserver
starting historyserver, logging to /home/ec2-user/hadoop-2.6.4/logs/mapred-ec2-user-historyserver-ip-172-31-10-146.us-west-1.compute.int
ernal.out
[ec2-user@ip-172-31-10-146 ~]$ jps
633 SecondaryNameNode
1090 NodeManager
1397 JobHistoryServer
969 ResourceManager
506 NameNode
655 DataNode
1439 Jps
[ec2-user@ip-172-31-10-146 ~]$ hadoop fs -mkdir /data
[ec2-user@ip-172-31-10-146 ~]$ wget http://rasinsrv07.cstcis.cti.depaul.edu/CSC555/bioproject.xml
--2019-07-01 01:40:37-- http://rasinsrv07.cstcis.cti.depaul.edu/CSC555/bioproject.xml
Resolving rasinsrv07.cstcis.cti.depaul.edu (rasinsrv07.cstcis.cti.depaul.edu)... 140.192.39.95
Connecting to rasinsrv07.cstcis.cti.depaul.edu (rasinsrv07.cstcis.cti.depaul.edu)[140.192.39.95]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 231149003 (220M) [text/xml]
Saving to: 'bioproject.xml'

100%[=====] 231,149,003 10.4MB/s in 22s

2019-07-01 01:40:59 (10.3 MB/s) - 'bioproject.xml' saved [231149003/231149003]

[ec2-user@ip-172-31-10-146 ~]$ hadoop fs -put bioproject.xml /data/
[ec2-user@ip-172-31-10-146 ~]$ hadoop fs -ls /data
Found 1 items
-rw-r--r-- 1 ec2-user supergroup 231149003 2019-07-01 01:42 /data/bioproject.xml
[ec2-user@ip-172-31-10-146 ~]$
```

2) Occurrences of “arctic”

```

Bytes Read=231153099
File Output Format Counters
Bytes Written=20056175
[ec2-user@ip-172-31-10-146 ~]$ hadoop fs -du /data/wordcount1/
0      /data/wordcount1/_SUCCESS
20056175 /data/wordcount1/part-r-00000
[ec2-user@ip-172-31-10-146 ~]$ hadoop fs -cat /data/wordcount1/part-r-00000 | grep arctic
&lt;I&gt;hol&lt;I&gt; 28
&lt;I&gt;hol&lt;I&gt;/&lt;/&gt;. 8
&lt;I&gt;hol&lt;I&gt;/&lt;I&gt;. 1
&lt;I&gt;pale&lt;I&gt;/&lt;I&gt;. 4
&lt;I&gt;hol&lt;I&gt;/&lt;I&gt;. 1
(Ant&lt;I&gt;
(Ant&lt;I&gt;) 1
(Ant&lt;I&gt;), 11
<Label>Ant&lt;I&gt; 1
<Name>Ant&lt;I&gt; 3
<Name>Ant&lt;I&gt; 1
<Strain>Ant&lt;I&gt; 1
<Title>Ant&lt;I&gt; 5
Ant&lt;I&gt; 137
Ant&lt;I&gt;, 1
Ant&lt;I&gt;. 2
Ant&lt;I&gt;.</Description> 1
Ant&lt;I&gt;.</Title> 1
Ant&lt;I&gt;</Title> 4
Ant&lt;I&gt; 16
Ant&lt;I&gt;.</Title> 1
Ant&lt;I&gt;, 9
Ant&lt;I&gt;. 24
Ant&lt;I&gt;.&lt;I&gt; 3
Ant&lt;I&gt;.</Description> 19
Ant&lt;I&gt;</Description> 2
Ant&lt;I&gt;</Name> 1
Ant&lt;I&gt;</Title> 6

```

```

ant&lt;I&gt;</Title> 1
ant&lt;I&gt;um 32
ant&lt;I&gt;um</Name> 3
ant&lt;I&gt;um</OrganismName> 3
ant&lt;I&gt;us 31
ant&lt;I&gt;us&lt;I&gt;/&lt;I&gt;. 4
ant&lt;I&gt;us&lt;I&gt;/&lt;I&gt;/&lt;I&gt;. 1
ant&lt;I&gt;us). 1
ant&lt;I&gt;us, 1
ant&lt;I&gt;us</Name> 5
ant&lt;I&gt;us</OrganismName> 5
arctic 21
arctic 27
arctic&lt;I&gt;/&lt;I&gt;) 2
arctic&lt;I&gt;/&lt;I&gt;. 3
arctic&lt;I&gt;/&lt;I&gt;. 1
arctic.</Description> 2
arctic.</Name> 5
arctic.</OrganismName> 5
arcticus 31
arcticus&lt;I&gt;/&lt;I&gt;. 2
arcticus</Name> 4
arcticus</OrganismName> 4
hol&lt;I&gt; 77
humans.Ant&lt;I&gt; 1
pale&lt;I&gt; 66
pale&lt;I&gt;</Name> 1
sub-Ant&lt;I&gt; 4
sub-arctic 4
subant&lt;I&gt; 1
subant&lt;I&gt;us 7
subant&lt;I&gt;us</Name> 1
subant&lt;I&gt;us</OrganismName> 1
sub&lt;I&gt; 21
[ec2-user@ip-172-31-10-146 ~]$

```


3) Job completion time

```
Desktop — ec2-user@ip-172-31-10-146:~ — ssh -i schaphekar.pem ec2-user@ec2-54-153-117-22.us-west-1.compute.amazonaws.com
real    0m0.164s
user    0m0.144s
sys      0m0.018s
[ec2-user@ip-172-31-10-146 ~]$ time hadoop jar hadoop-2.6.4/share/hadoop/mapreduce/hadoop-mapreduce-examples-2.6.4.jar wordcount /data/bioproject.xml /data/wordcount1
19/07/01 02:05:16 INFO client.RMProxy: Connecting to ResourceManager at localhost/127.0.0.1:8032
org.apache.hadoop.mapred.FileAlreadyExistsException: Output directory hdfs://localhost/data/wordcount1 already exists
    at org.apache.hadoop.mapreduce.lib.output.FileOutputFormat.checkOutputSpecs(FileOutputFormat.java:146)
    at org.apache.hadoop.mapreduce.JobSubmitter.checkSpecs(JobSubmitter.java:267)
    at org.apache.hadoop.mapreduce.JobSubmitter.submit(JobSubmitter.java:140)
    at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1297)
    at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1294)
    at java.security.AccessController.doPrivileged(Native Method)
    at javax.security.auth.Subject.doAs(Subject.java:422)
    at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1656)
    at org.apache.hadoop.mapreduce.Job.submit(Job.java:1294)
    at org.apache.hadoop.mapreduce.Job.waitForCompletion(Job.java:1315)
    at org.apache.hadoop.examples.WordCount.main(WordCount.java:87)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:498)
    at org.apache.hadoop.util.ProgramDriver$ProgramDescription.invoke(ProgramDriver.java:71)
    at org.apache.hadoop.util.ProgramDriver.run(ProgramDriver.java:144)
    at org.apache.hadoop.examples.ExampleDriver.main(ExampleDriver.java:74)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:498)
    at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
    at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
real    0m3.106s
user    0m2.668s
sys      0m0.095s
[ec2-user@ip-172-31-10-146 ~]$
```