Gavin Gunawardena
INFS 774
Dr. Liu
6/15/21

## INFS 774 Big Data Analytics

## ASSIGNMENT 2

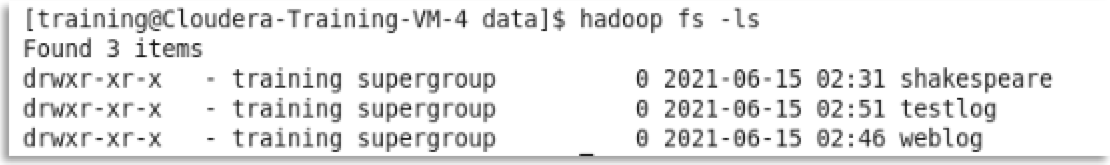
Due: 6/18/2021 + one week grace period

Please refer to "Access Hadoop VM" posted under "Content" regarding how to access the VM

**TASKS:** In this assignment you need to finish three tasks. In task 1 and task 2, you need to do some data processing with Hadoop - I provide detailed explanations for these two tasks on Page 3. **Whenever you encounter any problem, before you post your question on the discussion forum, please first read "3. Common errors" on page 3 of the document. Common errors.** Quite possibly, you will find your answers there. Task 3 will include a couple of essay questions.

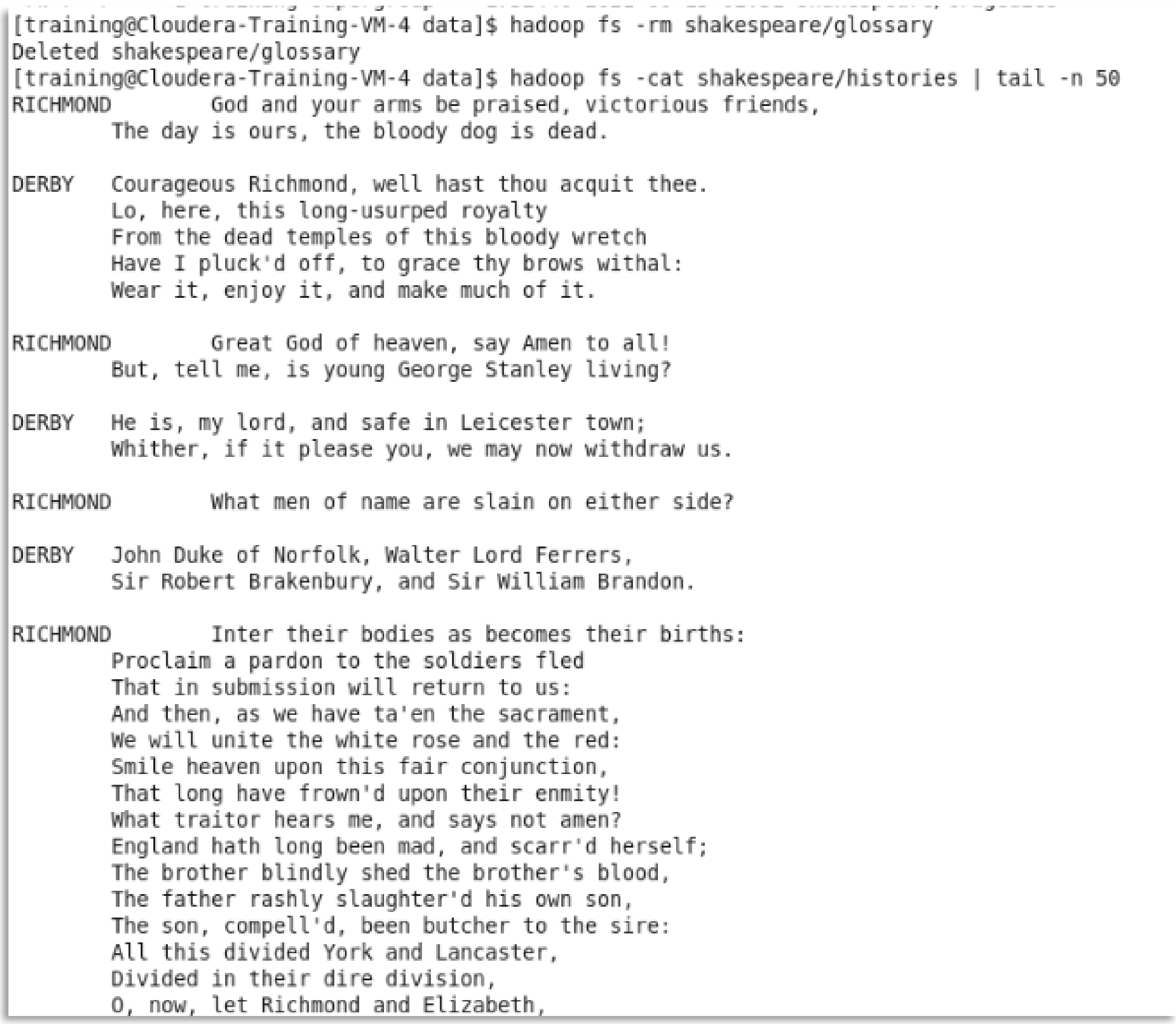
**TASK 1.** You will be working with the Cloudera Hadoop environment. Please complete the Cloudera Homework Labs – Lecture #1 (available on D2l under content -> assignments -> assignment 2). Please submit Screenshots of following steps from tutorial file:

i. Step 2-Uploading Files: After finishing 2.1 – 2.9, please run hadoop fs –ls, submit the results

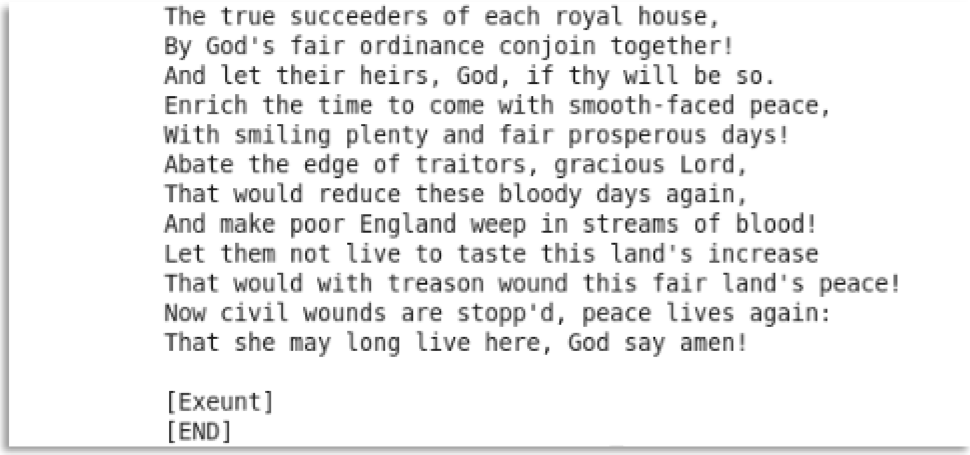


```
[training@Cloudera-Training-VM-4 data]$ hadoop fs -ls
Found 3 items
drwxr-xr-x   - training supergroup          0 2021-06-15 02:31 shakespeare
drwxr-xr-x   - training supergroup          0 2021-06-15 02:51 testlog
drwxr-xr-x   - training supergroup          0 2021-06-15 02:46 weblog
```

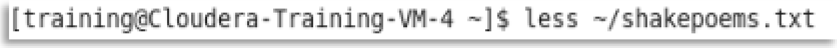
ii. Step 3-Viewing and manipulating files: Result from step 3



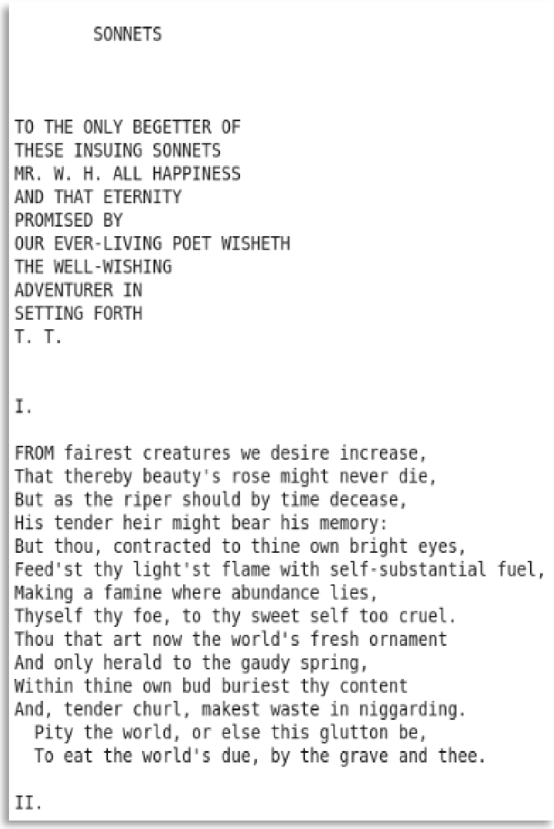
```
[training@Cloudera-Training-VM-4 data]$ hadoop fs -rm shakespeare/glossary
Deleted shakespeare/glossary
[training@Cloudera-Training-VM-4 data]$ hadoop fs -cat shakespeare/histories | tail -n 50
RICHMOND        God and your arms be praised, victorious friends,
        The day is ours, the bloody dog is dead.

DERBY   Courageous Richmond, well hast thou acquit thee.
        Lo, here, this long-usurped royalty
        From the dead temples of this bloody wretch
        Have I pluck'd off, to grace thy brows withal:
        Wear it, enjoy it, and make much of it.

RICHMOND        Great God of heaven, say Amen to all!
        But, tell me, is young George Stanley living?

DERBY   He is, my lord, and safe in Leicester town;
        Whither, if it please you, we may now withdraw us.

RICHMOND        What men of name are slain on either side?

DERBY   John Duke of Norfolk, Walter Lord Ferrers,
        Sir Robert Brakenbury, and Sir William Brandon.

RICHMOND        Inter their bodies as becomes their births:
        Proclaim a pardon to the soldiers fled
        That in submission will return to us:
        And then, as we have ta'en the sacrament,
        We will unite the white rose and the red:
        Smile heaven upon this fair conjunction,
        That long have frown'd upon their enmity!
        What traitor hears me, and says not amen?
        England hath long been mad, and scarr'd herself;
        The brother blindly shed the brother's blood,
        The father rashly slaughter'd his own son,
        The son, compell'd, been butcher to the sire:
        All this divided York and Lancaster,
        Divided in their dire division,
        O, now, let Richmond and Elizabeth,
```



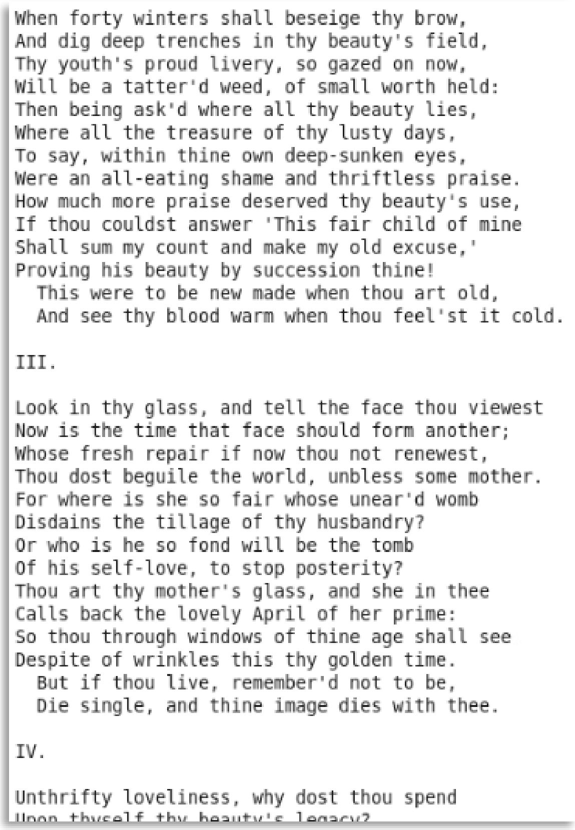
```
            The true succeeders of each royal house,
            By God's fair ordinance conjoin together!
            And let their heirs, God, if thy will be so.
            Enrich the time to come with smooth-faced peace,
            With smiling plenty and fair prosperous days!
            Abate the edge of traitors, gracious Lord,
            That would reduce these bloody days again,
            And make poor England weep in streams of blood!
            Let them not live to taste this land's increase
            That would with treason wound this fair land's peace!
            Now civil wounds are stopp'd, peace lives again:
            That she may long live here, God say amen!

            [Exeunt]
            [END]
```



```
[training@Cloudera-Training-VM-4 ~]$ less ~/shakepoems.txt
```



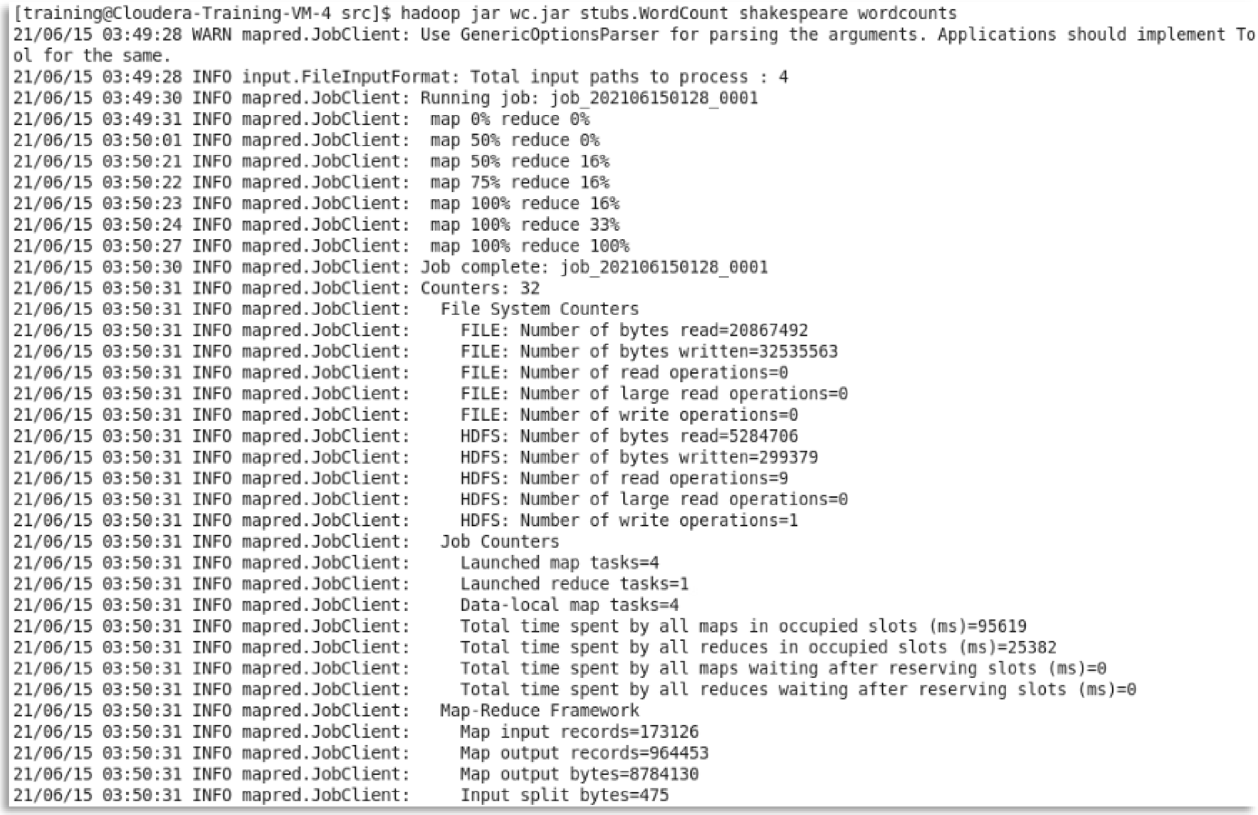
```
        SONNETS



TO THE ONLY BEGETTER OF
THESE INSUING SONNETS
MR. W. H. ALL HAPPINESS
AND THAT ETERNITY
PROMISED BY
OUR EVER-LIVING POET WISHETH
THE WELL-WISHING
ADVENTURER IN
SETTING FORTH
T. T.


I.

FROM fairest creatures we desire increase,
That thereby beauty's rose might never die,
But as the riper should by time decease,
His tender heir might bear his memory:
But thou, contracted to thine own bright eyes,
Feed'st thy light'st flame with self-substantial fuel,
Making a famine where abundance lies,
Thyself thy foe, to thy sweet self too cruel.
Thou that art now the world's fresh ornament
And only herald to the gaudy spring,
Within thine own bud buriest thy content
And, tender churl, makest waste in niggarding.
  Pity the world, or else this glutton be,
  To eat the world's due, by the grave and thee.

II.
```



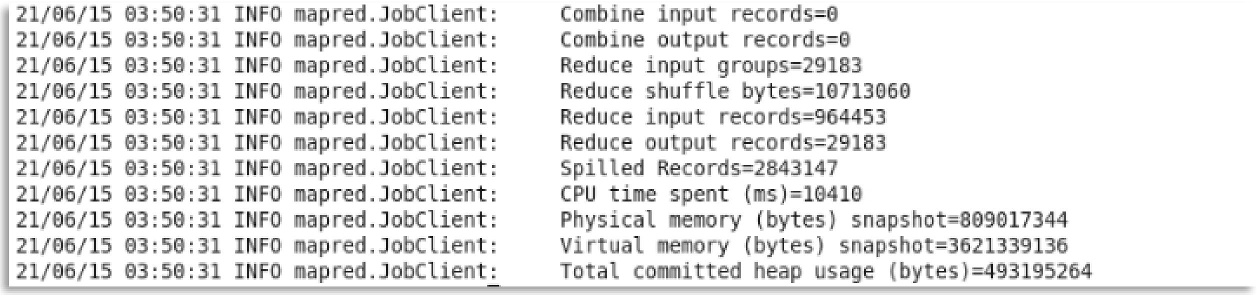
```
When forty winters shall beseige thy brow,
And dig deep trenches in thy beauty's field,
Thy youth's proud livery, so gazed on now,
Will be a tatter'd weed, of small worth held:
Then being ask'd where all thy beauty lies,
Where all the treasure of thy lusty days,
To say, within thine own deep-sunken eyes,
Were an all-eating shame and thriftless praise.
How much more praise deserved thy beauty's use,
If thou couldst answer 'This fair child of mine
Shall sum my count and make my old excuse,'
Proving his beauty by succession thine!
  This were to be new made when thou art old,
  And see thy blood warm when thou feel'st it cold.

III.

Look in thy glass, and tell the face thou viewest
Now is the time that face should form another;
Whose fresh repair if now thou not renewest,
Thou dost beguile the world, unbless some mother.
For where is she so fair whose unear'd womb
Disdains the tillage of thy husbandry?
Or who is he so fond will be the tomb
Of his self-love, to stop posterity?
Thou art thy mother's glass, and she in thee
Calls back the lovely April of her prime:
So thou through windows of thine age shall see
Despite of wrinkles this thy golden time.
  But if thou live, remember'd not to be,
  Die single, and thine image dies with thee.

IV.

Unthrifty loveliness, why dost thou spend
Upon thyself thy beauty's legacy?
```

**TASK 2.** You will be working with the Cloudera Hadoop environment. Please complete the Cloudera Homework Labs – Lecture #2 (available on D2l under content -> assignments -> assignment 2). Please submit Screenshots of following steps from tutorial file:

i. Compiling and Submitting a MapReduce Job: step 5

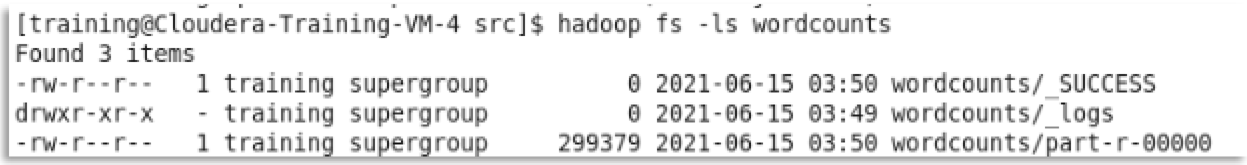


```
[training@Cloudera-Training-VM-4 src]$ hadoop jar wc.jar stubs.WordCount shakespeare wordcounts
21/06/15 03:49:28 WARN mapred.JobClient: Use GenericOptionsParser for parsing the arguments. Applications should implement To
ol for the same.
21/06/15 03:49:28 INFO input.FileInputFormat: Total input paths to process : 4
21/06/15 03:49:30 INFO mapred.JobClient: Running job: job_202106150128_0001
21/06/15 03:49:31 INFO mapred.JobClient:  map 0% reduce 0%
21/06/15 03:50:01 INFO mapred.JobClient:  map 50% reduce 0%
21/06/15 03:50:21 INFO mapred.JobClient:  map 50% reduce 16%
21/06/15 03:50:22 INFO mapred.JobClient:  map 75% reduce 16%
21/06/15 03:50:23 INFO mapred.JobClient:  map 100% reduce 16%
21/06/15 03:50:24 INFO mapred.JobClient:  map 100% reduce 33%
21/06/15 03:50:27 INFO mapred.JobClient:  map 100% reduce 100%
21/06/15 03:50:30 INFO mapred.JobClient: Job complete: job_202106150128_0001
21/06/15 03:50:31 INFO mapred.JobClient: Counters: 32
21/06/15 03:50:31 INFO mapred.JobClient:   File System Counters
21/06/15 03:50:31 INFO mapred.JobClient:     FILE: Number of bytes read=20867492
21/06/15 03:50:31 INFO mapred.JobClient:     FILE: Number of bytes written=32535563
21/06/15 03:50:31 INFO mapred.JobClient:     FILE: Number of read operations=0
21/06/15 03:50:31 INFO mapred.JobClient:     FILE: Number of large read operations=0
21/06/15 03:50:31 INFO mapred.JobClient:     FILE: Number of write operations=0
21/06/15 03:50:31 INFO mapred.JobClient:     HDFS: Number of bytes read=5284706
21/06/15 03:50:31 INFO mapred.JobClient:     HDFS: Number of bytes written=299379
21/06/15 03:50:31 INFO mapred.JobClient:     HDFS: Number of read operations=9
21/06/15 03:50:31 INFO mapred.JobClient:     HDFS: Number of large read operations=0
21/06/15 03:50:31 INFO mapred.JobClient:     HDFS: Number of write operations=1
21/06/15 03:50:31 INFO mapred.JobClient:   Job Counters
21/06/15 03:50:31 INFO mapred.JobClient:     Launched map tasks=4
21/06/15 03:50:31 INFO mapred.JobClient:     Launched reduce tasks=1
21/06/15 03:50:31 INFO mapred.JobClient:     Data-local map tasks=4
21/06/15 03:50:31 INFO mapred.JobClient:     Total time spent by all maps in occupied slots (ms)=95619
21/06/15 03:50:31 INFO mapred.JobClient:     Total time spent by all reduces in occupied slots (ms)=25382
21/06/15 03:50:31 INFO mapred.JobClient:     Total time spent by all maps waiting after reserving slots (ms)=0
21/06/15 03:50:31 INFO mapred.JobClient:     Total time spent by all reduces waiting after reserving slots (ms)=0
21/06/15 03:50:31 INFO mapred.JobClient:   Map-Reduce Framework
21/06/15 03:50:31 INFO mapred.JobClient:     Map input records=173126
21/06/15 03:50:31 INFO mapred.JobClient:     Map output records=964453
21/06/15 03:50:31 INFO mapred.JobClient:     Map output bytes=8784130
21/06/15 03:50:31 INFO mapred.JobClient:     Input split bytes=475
```



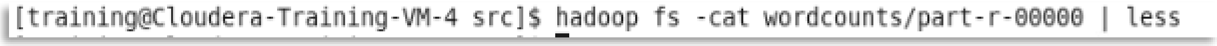
```
21/06/15 03:50:31 INFO mapred.JobClient:     Combine input records=0
21/06/15 03:50:31 INFO mapred.JobClient:     Combine output records=0
21/06/15 03:50:31 INFO mapred.JobClient:     Reduce input groups=29183
21/06/15 03:50:31 INFO mapred.JobClient:     Reduce shuffle bytes=10713060
21/06/15 03:50:31 INFO mapred.JobClient:     Reduce input records=964453
21/06/15 03:50:31 INFO mapred.JobClient:     Reduce output records=29183
21/06/15 03:50:31 INFO mapred.JobClient:     Spilled Records=2843147
21/06/15 03:50:31 INFO mapred.JobClient:     CPU time spent (ms)=10410
21/06/15 03:50:31 INFO mapred.JobClient:     Physical memory (bytes) snapshot=809017344
21/06/15 03:50:31 INFO mapred.JobClient:     Virtual memory (bytes) snapshot=3621339136
21/06/15 03:50:31 INFO mapred.JobClient:     Total committed heap usage (bytes)=493195264
```

ii. Compiling and Submitting a MapReduce Job: step 7

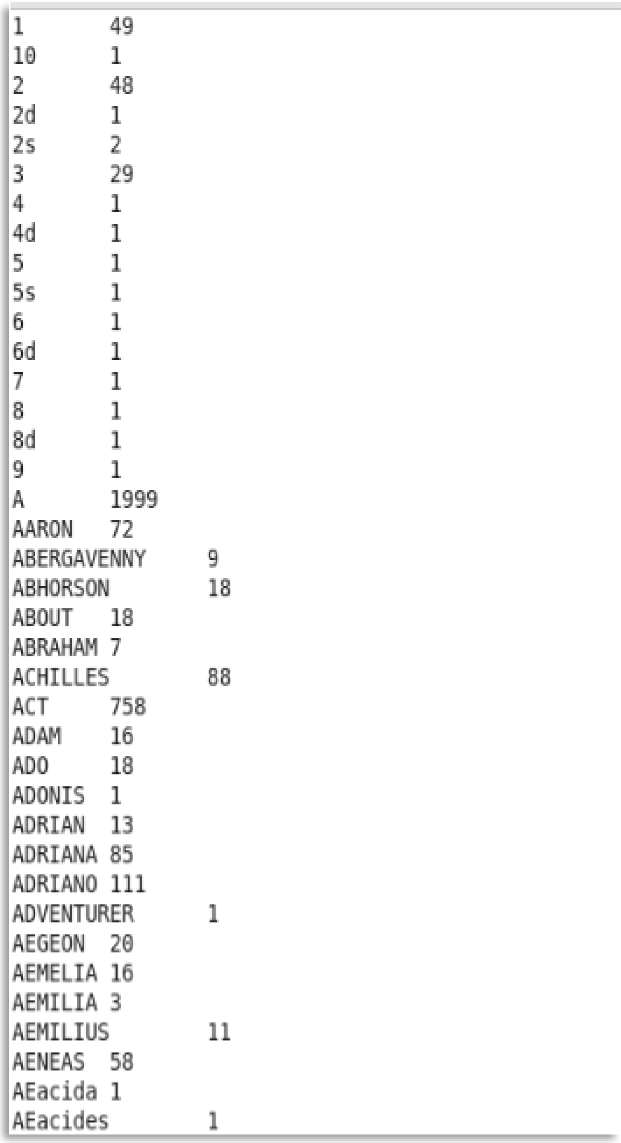


```
[training@Cloudera-Training-VM-4 src]$ hadoop fs -ls wordcounts
Found 3 items
-rw-r--r--   1 training supergroup          0 2021-06-15 03:50 wordcounts/_SUCCESS
drwxr-xr-x   - training supergroup          0 2021-06-15 03:49 wordcounts/_logs
-rw-r--r--   1 training supergroup     299379 2021-06-15 03:50 wordcounts/part-r-00000
```

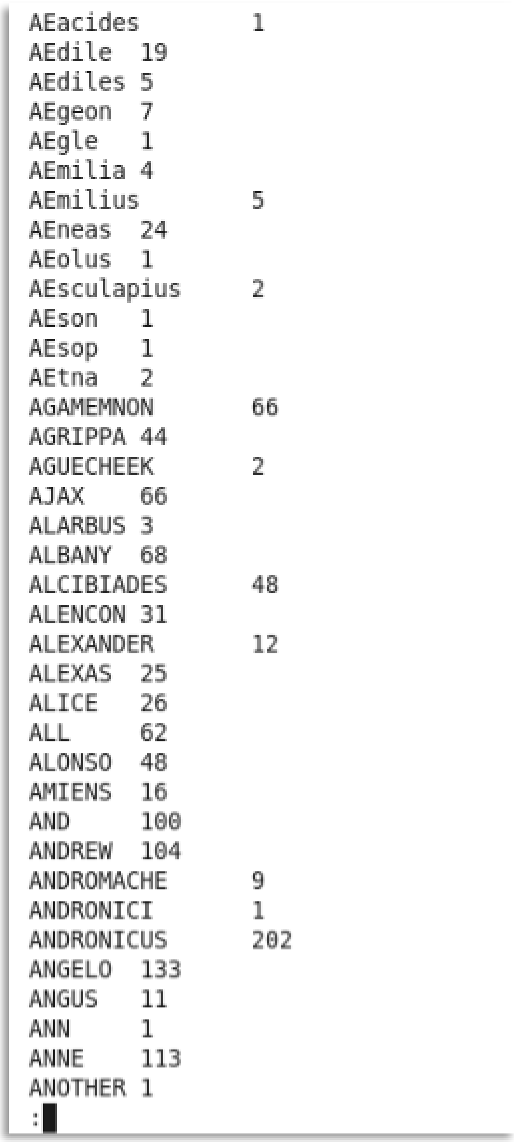
iii. Compiling and Submitting a MapReduce Job: step 8



```
[training@Cloudera-Training-VM-4 src]$ hadoop fs -cat wordcounts/part-r-00000 | less
```

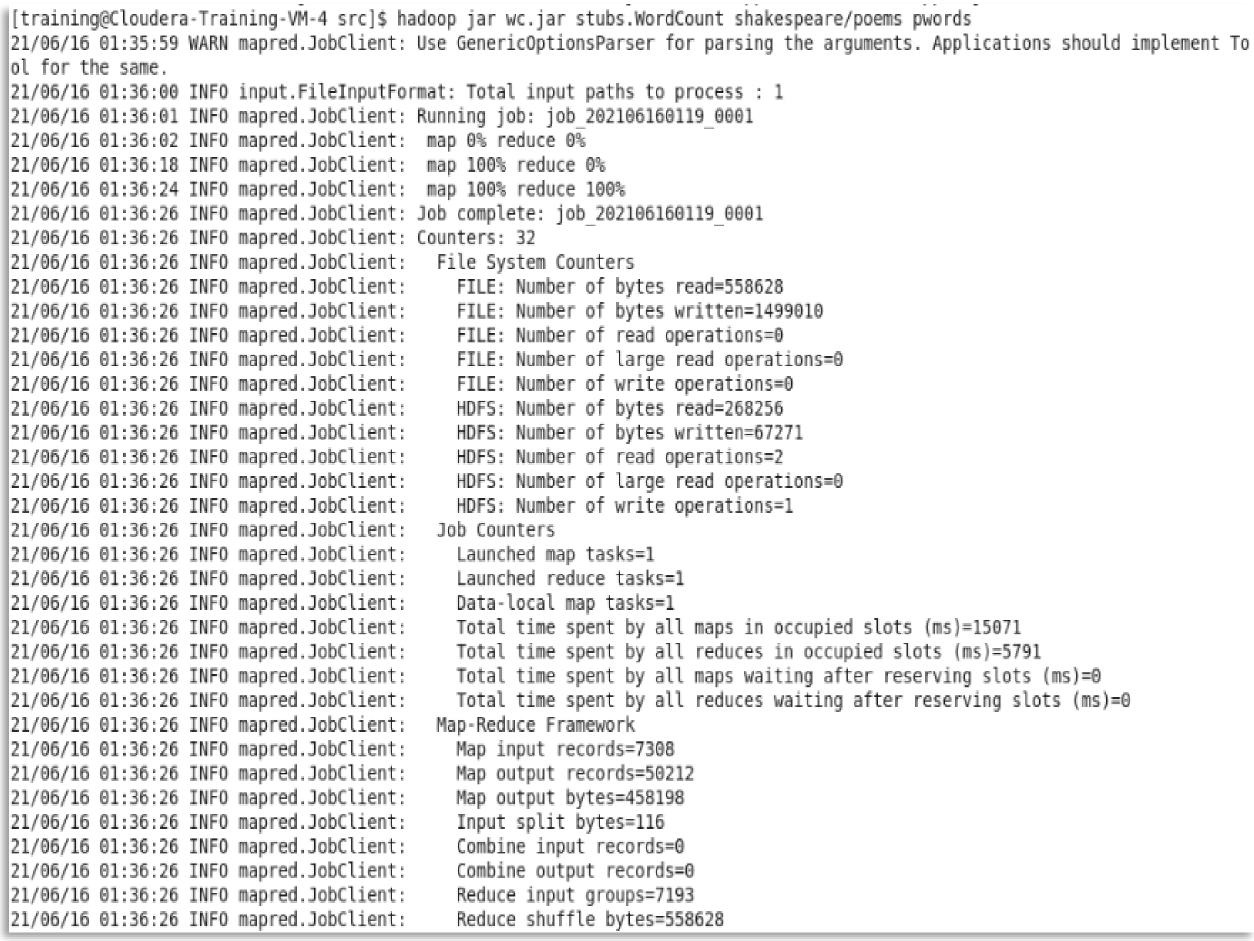


```
1       49
10      1
2       48
2d      1
2s      2
3       29
4       1
4d      1
5       1
5s      1
6       1
6d      1
7       1
8       1
8d      1
9       1
A       1999
AARON   72
ABERGAVENNY     9
ABHORSON        18
ABOUT   18
ABRAHAM 7
ACHILLES        88
ACT     758
ADAM    16
ADO     18
ADONIS  1
ADRIAN  13
ADRIANA 85
ADRIANO 111
ADVENTURER      1
AEGEON  20
AEMELIA 16
AEMILIA 3
AEMILIUS        11
AENEAS  58
AEacida 1
AEacides        1
```

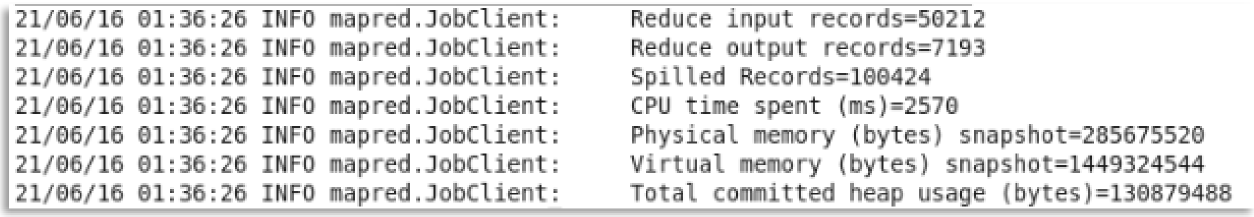


```
AEacides        1
AEdile  19
AEdiles 5
AEgeon  7
AEgle   1
AEmilia 4
AEmilius        5
AEneas  24
AEolus  1
AEsculapius     2
AEson   1
AEsop   1
AEtna   2
AGAMEMNON       66
AGRIPPA 44
AGUECHEEK       2
AJAX    66
ALARBUS 3
ALBANY  68
ALCIBIADES      48
ALENCON 31
ALEXANDER       12
ALEXAS  25
ALICE   26
ALL     62
ALONSO  48
AMIENS  16
AND     100
ANDREW  104
ANDROMACHE      9
ANDRONICI       1
ANDRONICUS      202
ANGELO  133
ANGUS   11
ANN     1
ANNE    113
ANOTHER 1
:
```

iv. Compiling and Submitting a MapReduce Job: step 9 (replicating steps 7 and 8 for pwords)

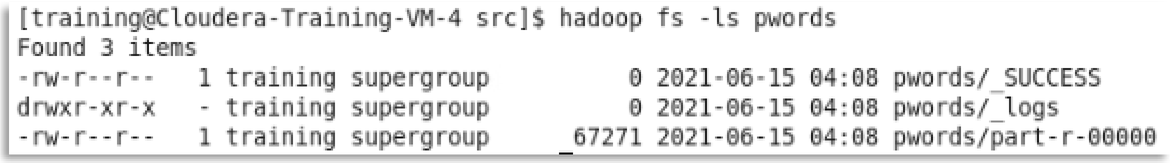


```
[training@Cloudera-Training-VM-4 src]$ hadoop jar wc.jar stubs.WordCount shakespeare/poems pwords
21/06/16 01:35:59 WARN mapred.JobClient: Use GenericOptionsParser for parsing the arguments. Applications should implement To
ol for the same.
21/06/16 01:36:00 INFO input.FileInputFormat: Total input paths to process : 1
21/06/16 01:36:01 INFO mapred.JobClient: Running job: job_202106160119_0001
21/06/16 01:36:02 INFO mapred.JobClient:  map 0% reduce 0%
21/06/16 01:36:18 INFO mapred.JobClient:  map 100% reduce 0%
21/06/16 01:36:24 INFO mapred.JobClient:  map 100% reduce 100%
21/06/16 01:36:26 INFO mapred.JobClient: Job complete: job_202106160119_0001
21/06/16 01:36:26 INFO mapred.JobClient: Counters: 32
21/06/16 01:36:26 INFO mapred.JobClient:   File System Counters
21/06/16 01:36:26 INFO mapred.JobClient:     FILE: Number of bytes read=558628
21/06/16 01:36:26 INFO mapred.JobClient:     FILE: Number of bytes written=1499010
21/06/16 01:36:26 INFO mapred.JobClient:     FILE: Number of read operations=0
21/06/16 01:36:26 INFO mapred.JobClient:     FILE: Number of large read operations=0
21/06/16 01:36:26 INFO mapred.JobClient:     FILE: Number of write operations=0
21/06/16 01:36:26 INFO mapred.JobClient:     HDFS: Number of bytes read=268256
21/06/16 01:36:26 INFO mapred.JobClient:     HDFS: Number of bytes written=67271
21/06/16 01:36:26 INFO mapred.JobClient:     HDFS: Number of read operations=2
21/06/16 01:36:26 INFO mapred.JobClient:     HDFS: Number of large read operations=0
21/06/16 01:36:26 INFO mapred.JobClient:     HDFS: Number of write operations=1
21/06/16 01:36:26 INFO mapred.JobClient:   Job Counters
21/06/16 01:36:26 INFO mapred.JobClient:     Launched map tasks=1
21/06/16 01:36:26 INFO mapred.JobClient:     Launched reduce tasks=1
21/06/16 01:36:26 INFO mapred.JobClient:     Data-local map tasks=1
21/06/16 01:36:26 INFO mapred.JobClient:     Total time spent by all maps in occupied slots (ms)=15071
21/06/16 01:36:26 INFO mapred.JobClient:     Total time spent by all reduces in occupied slots (ms)=5791
21/06/16 01:36:26 INFO mapred.JobClient:     Total time spent by all maps waiting after reserving slots (ms)=0
21/06/16 01:36:26 INFO mapred.JobClient:     Total time spent by all reduces waiting after reserving slots (ms)=0
21/06/16 01:36:26 INFO mapred.JobClient:   Map-Reduce Framework
21/06/16 01:36:26 INFO mapred.JobClient:     Map input records=7308
21/06/16 01:36:26 INFO mapred.JobClient:     Map output records=50212
21/06/16 01:36:26 INFO mapred.JobClient:     Map output bytes=458198
21/06/16 01:36:26 INFO mapred.JobClient:     Input split bytes=116
21/06/16 01:36:26 INFO mapred.JobClient:     Combine input records=0
21/06/16 01:36:26 INFO mapred.JobClient:     Combine output records=0
21/06/16 01:36:26 INFO mapred.JobClient:     Reduce input groups=7193
21/06/16 01:36:26 INFO mapred.JobClient:     Reduce shuffle bytes=558628
```

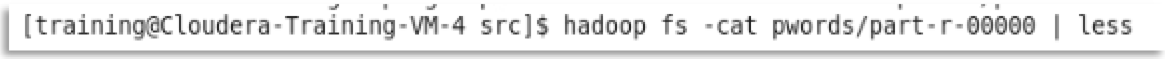


```
21/06/16 01:36:26 INFO mapred.JobClient:     Reduce input records=50212
21/06/16 01:36:26 INFO mapred.JobClient:     Reduce output records=7193
21/06/16 01:36:26 INFO mapred.JobClient:     Spilled Records=100424
21/06/16 01:36:26 INFO mapred.JobClient:     CPU time spent (ms)=2570
21/06/16 01:36:26 INFO mapred.JobClient:     Physical memory (bytes) snapshot=285675520
21/06/16 01:36:26 INFO mapred.JobClient:     Virtual memory (bytes) snapshot=1449324544
21/06/16 01:36:26 INFO mapred.JobClient:     Total committed heap usage (bytes)=130879488
```

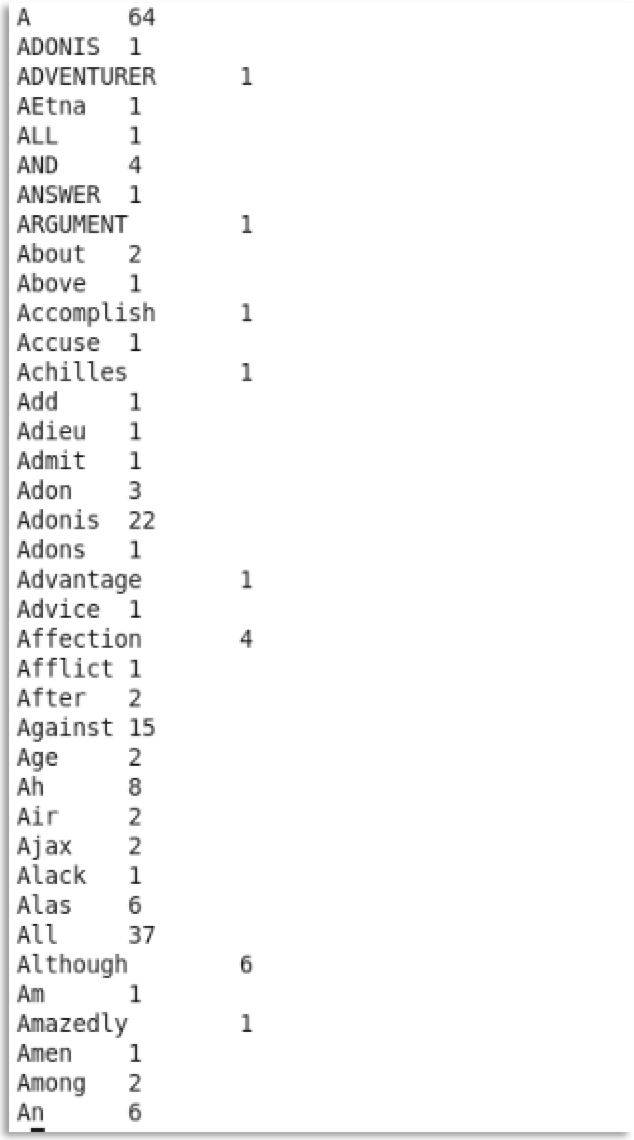
*Recreation of steps 7-8 but for the individual file, poems, and utilizing pwords*



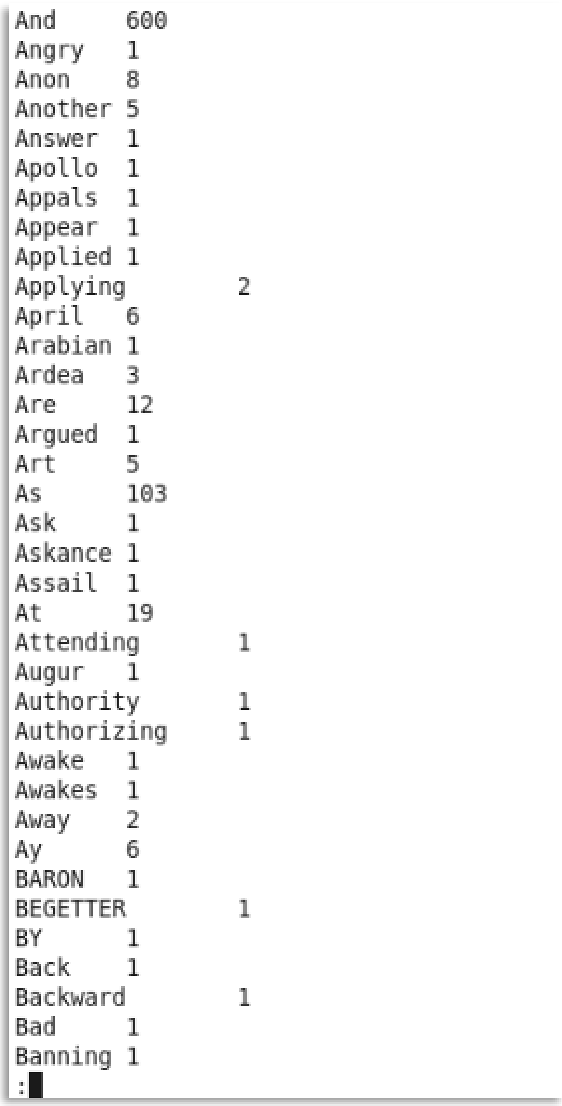
```
[training@Cloudera-Training-VM-4 src]$ hadoop fs -ls pwords
Found 3 items
-rw-r--r--   1 training supergroup          0 2021-06-15 04:08 pwords/_SUCCESS
drwxr-xr-x   - training supergroup          0 2021-06-15 04:08 pwords/_logs
-rw-r--r--   1 training supergroup      67271 2021-06-15 04:08 pwords/part-r-00000
```



```
[training@Cloudera-Training-VM-4 src]$ hadoop fs -cat pwords/part-r-00000 | less
```



```
A       64
ADONIS  1
ADVENTURER      1
AEtna   1
ALL     1
AND     4
ANSWER  1
ARGUMENT        1
About   2
Above   1
Accomplish      1
Accuse  1
Achilles        1
Add     1
Adieu   1
Admit   1
Adon    3
Adonis  22
Adons   1
Advantage       1
Advice  1
Affection       4
Afflict 1
After   2
Against 15
Age     2
Ah      8
Air     2
Ajax    2
Alack   1
Alas    6
All     37
Although        6
Am      1
Amazedly        1
Amen    1
Among   2
An      6
```



```
And     600
Angry   1
Anon    8
Another 5
Answer  1
Apollo  1
Appals  1
Appear  1
Applied 1
Applying        2
April   6
Arabian 1
Ardea   3
Are     12
Argued  1
Art     5
As      103
Ask     1
Askance 1
Assail  1
At      19
Attending       1
Augur   1
Authority       1
Authorizing     1
Awake   1
Awakes  1
Away    2
Ay      6
BARON   1
BEGETTER        1
BY      1
Back    1
Backward        1
Bad     1
Banning 1
:
```

**TASK3**.

Q1: Using the study materials provided as a starting point, and any others that you may refer to, write in your own words a brief synopsis (200-300 words – roughly half a page, font 12 font, single spacing) of what you understand by the term "big data analytics". Some ideas for discussion -- How is the term similar or different than other related terms/disciplines? In building your arguments, consider whether the industry term “big data” is a hype and if so, to what extent. What is novel, unique about this term with regards to the practical and research implications that it presents?

Big Data Analytics, a concept mentioned frequently in the past decade in tech and business media, is the analytics of the large and growing streams of data that organizations collect or can collect. The scale of the data referred to in Big Data is immense to the point that it is often measured in terabytes and petabytes when in storage and when streaming from various data sources. Documentation from IBM mentions that studies have shown how companies with a competitive advantage in analytics are twice as likely to outperform their competitors. Thus, there is a major incentive for companies to analyze the data they collect or can collect during their business operations.

The term “Big Data” is likely being used to hype the analytics industry. Analytics has been around and used by businesses for centuries, but lately the demand for it has increased. Businesses who sell analytics products or services have an incentive to generate hype for the field in order to bolster their products and services. Companies that employ analytics teams also have an incentive to generate hype in order to increase the supply of analysts in the workforce and by doing so, improve the average skill level in the field and lower or stabilize the average wage for the field. Thus, in current times, the term “Big Data” has become a major part of modern-day economies and even cultures as there is a demand for it from businesses and thus growing interest in it.

T2. Discuss 2 motivational scenarios (200-300 words – roughly half a page, font 12 font, single spacing) where use of “big data analytics” can play a significant role in drastically changing the current status quo/problem situation. You should use cases other than those mentioned in the book (Harness the Power of Big Data) and video resources. Feel free to look for other resources on the Web and academic databases.

While researching this topic, I found two scenarios form the past few years where big data analytics were used to, in the first scenario, investigate possible government corruption in a developing nation (Thomas, 2018), and in the second scenario, combat drug abuse (New Jersey Institute of Technology, 2019).

The first scenario I researched was about the Directorate of Science, Technology, and Innovation (DSTI) in Sierra Leone and their use of big data analytics to attempt to locate 4000 vehicles that went missing during the transition period of a new government in 2018. DSTI which is a department within the office of the president of Sierra Leone analyzed registration data from Sierra Leone's Road Safety Authority to determine which of these vehicles were simply transferred to other government departments and which were transferred into private and commercial use. These findings were then transferred to the country's Anti-Corruption Commission so that action could be taken.

In the second scenario I researched, a team lead by Dr. Hai Pha from NJIT's Ying Wu College of Computing has been analyzing social media posts, geospatial data and emergency services responses to track and monitor drug abuse in communities. They developed a system called DrugTracker that creates heat maps and statistical charts with this data to help find hotspots of drug abuse that treatment centers and counselors can target. The main purpose being to create a system where organizations trying to help drug addicts could get real time data instead of the yearly data that is currently available.

## References

New Jersey Institute of Technology. (2019, July 25). *Research uses big data to track and treat drug abuse*. Retrieved from Medical Xpress: https://medicalxpress.com/news/2019-07-big-track-drug-abuse.html

Thomas, A. R. (2018, November 7). *Scientists in Sierra Leone use big data to fight corruption*. Retrieved from Sierra Leone Telegraph: https://www.thesierraleonetelegraph.com/scientists-in-sierra-leone-use-big-data-to-fight-corruption/

**Task 1 and 2 explanation:**

1. Lab 1:

Lab 1 is mainly about uploading a dataset to the Hadoop machine. Let’s say you have a dataset you want to process using Hadoop, and this dataset is on your windows machine. You first need to transfer the dataset to the Linux machine using ssh or ftp. In our assignment, the dataset “shakespeare” is already in the Linux system, so you don’t need to do file transfer from Windows to Linux. The dataset is now in the Linux file system, but not in the Hadoop HDFS. You still need to upload the file to Hadoop HDFS. How to access HDFS from your Linux system? You usually type “hadoop fs“ or sometimes “hdfs fs”. You use the command “hadoop fs –put sourceOnLinux destinationOnHDFS” to upload a dataset to HDFS. It’s important to remember that the Linux file system and HDFS are two different file systems. They have different sets of commands. You need to be clear about in which file system your target dataset is located and then use the corresponding commands. In order to a MapReduce job, you need to upload the dataset to HDFS.

In lab 1, the target dataset is shakespeare.tar.gz. It is placed under the folder ~/training_materials/developer/data in the Linux file system. You first process the file in the Linux file system. You need to:

1. Go to the directory ~/training_materials/developer/data using the command “cd”.
2. Since this file is a zip file, you need to unzip it by typing “tar zxvf shakespeare.tar.gz”. Then you should have a folder called “shakespeare” that includes 5 files.

Next, you upload this “shakespeare” folder to the Hadoop HDFS by inputting “hadoop fs -put shakespeare /user/training/shakespeare”. The Shakespeare folder and its contents will be put into a “remote” HDFS directory named /user/training/shakespeare. Now you can type “hadoop fs -ls shakespeare” to see what’s inside the folder. It turns out that the folder Shakespeare includes five files (glossary, poems, histories, comedies, tragedies) and you want to remove the file glossary by typing “hadoop fs -rm shakespeare/glossary”. Here, you remove the file in HDFS. Actually, you can also remove the file from the shakespeare folder in the Linux system first and then upload the folder to HDFS.

2. Lab 2:

In Lab 1, you upload the shakespeare folder that includes the four files (poems, histories, comedies, tragedies) to HDFS. In lab 2, you want to run a MapReduce job to count the number of occurrences of each word in the folder. Lab 2 is about preparing and running some existing MapReduce code.

The MapReduce code is in the folder ~/workspace/wordcount/src/stubs (stubs is actually a java package, but can be roughly understood as a folder). The folder contains several Java files. You need to first do “ javac -classpath `hadoop classpath` stubs/*.java” to compile these java files into .class files that contain java bytecode. Then, you type “jar cvf wc.jar stubs/*.class” to create a jar. A jar file in Java is kind of like a combination of the “zip” and“.exe” file in Windows. It is a zip file that includes the .class files and it is executable. The above is just the common procedure for creating an executable jar file in java. When you run a MapReduce job in java, you always need to create a jar file first.

If you are familiar with java, you should know that to run an executable jar, you need to use the command “java –jar”. In Hadoop, we use “hadoop jar”. You type “hadoop jar wc.jar stubs.WordCount shakespeare wordcounts” to run the mapreduce job. This hadoop jar command says that the JAR file to use is wc.jar, and the main method is in stubs.WordCount (when you do java programming, you always need to have a main method), the input directory is called “shakespere” in the HDFS user root directory (/user/training in our case) and the output directory for storing the results is called wordcounts (the full path should be /user/training/wordcounts). Your java code will then count how many times each word appears in the folder “shakespeare”. The results including the key-value pairs can be found in a file “part-r-0000” in the output folder “wordcount”. You can type “hadoop fs -cat wordcounts/part-r-00000 | less” to view the results

In lab 2, you also need to do “hadoop jar wc.jar stubs.WordCount shakespeare/poems pwords” to count the occurrences of each word in the file “sharepeare/poems”. The output folder is called “pwords”.

3. Common errors:

Please note that in Hadoop, we don’t overwrite files. Hence, if you want to re-run your MapReduce job, you need to first remove the output folder.

Below, I discuss some of the common errors you will probably encounter when you work on this assignment.

1. If you see the error: File does not exist: /user/training/shakespeare/shakespeare when you run the mapreduce code, it basically means that the folder "shakespeare" in your HDFS has somehow been messed up. Obviously, you do not have a file called "shakespeare" under the folder "shakespeare". "hadoop fs -ls shakespeare" should list only 4 text files including comedies, tragedies, histories, and poems. If you see "shakespeare" in the list, it means something went wrong when you upload the dataset - maybe you uploaded the dataset multiple times. In HDFS, you usually do not overwrite an existing dataset. You need to remove the dataset and upload it again. Hence, to deal with the error, you need to remove this shakepeare folder under "/user/training"and then re-upload the data. You can do “hadoop fs –rm –r –f shakespeare” to remove the Shakespeare folder. After you remove the folder,  you can type "hadoop fs -ls" to verify. Now you shouldn't see the folder "shakepeare". Then you can go back to step 2.3 in lecture 1 lab and re-upload the data.
2. Please note that in lab2 step 2. You need type "javac -classpath `hadoop classpath` stubs/*java". You need to use backquotes (rather than single quotes) to enclose the text "hadoop classpath".