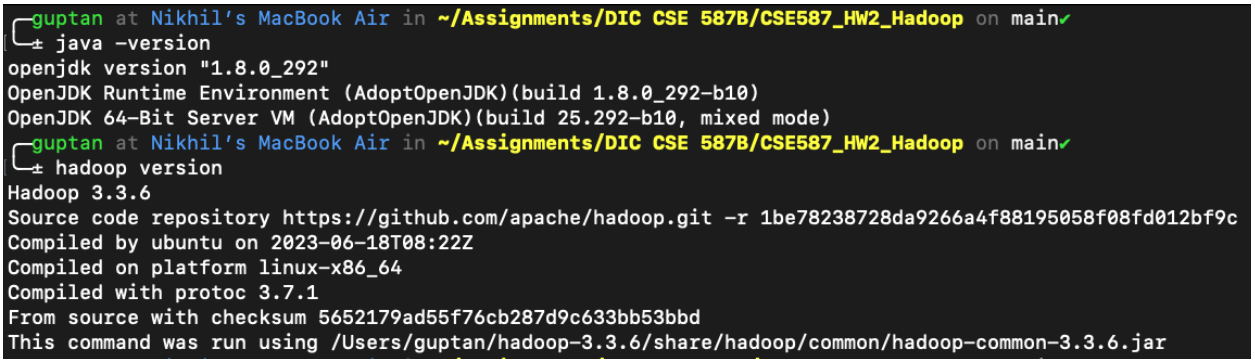
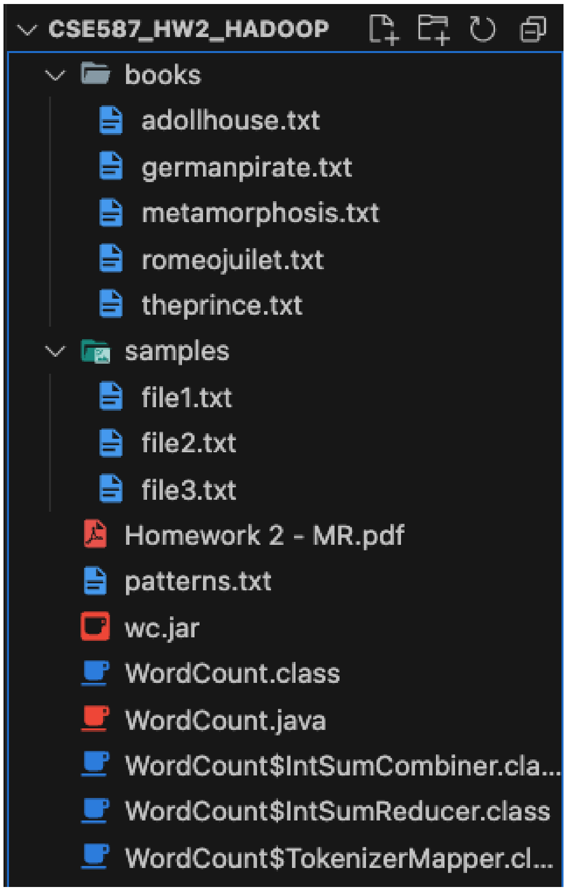
Homework 2 - MapReduce

Software Versions



```
┌guptan at Nikhil’s MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✔
└± java -version
openjdk version "1.8.0_292"
OpenJDK Runtime Environment (AdoptOpenJDK)(build 1.8.0_292-b10)
OpenJDK 64-Bit Server VM (AdoptOpenJDK)(build 25.292-b10, mixed mode)
┌guptan at Nikhil’s MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✔
└± hadoop version
Hadoop 3.3.6
Source code repository https://github.com/apache/hadoop.git -r 1be78238728da9266a4f88195058f08fd012bf9c
Compiled by ubuntu on 2023-06-18T08:22Z
Compiled on platform linux-x86_64
Compiled with protoc 3.7.1
From source with checksum 5652179ad55f76cb287d9c633bb53bbd
This command was run using /Users/guptan/hadoop-3.3.6/share/hadoop/common/hadoop-common-3.3.6.jar
```

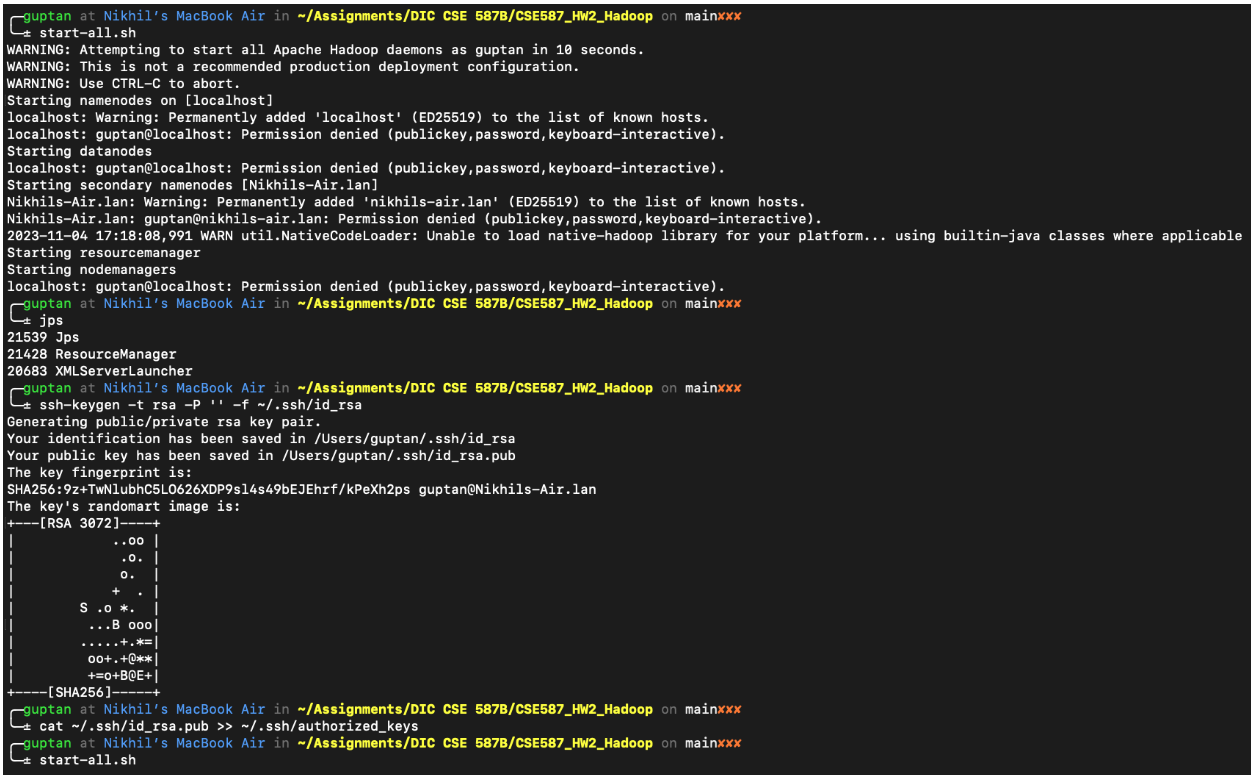
Folder Structure



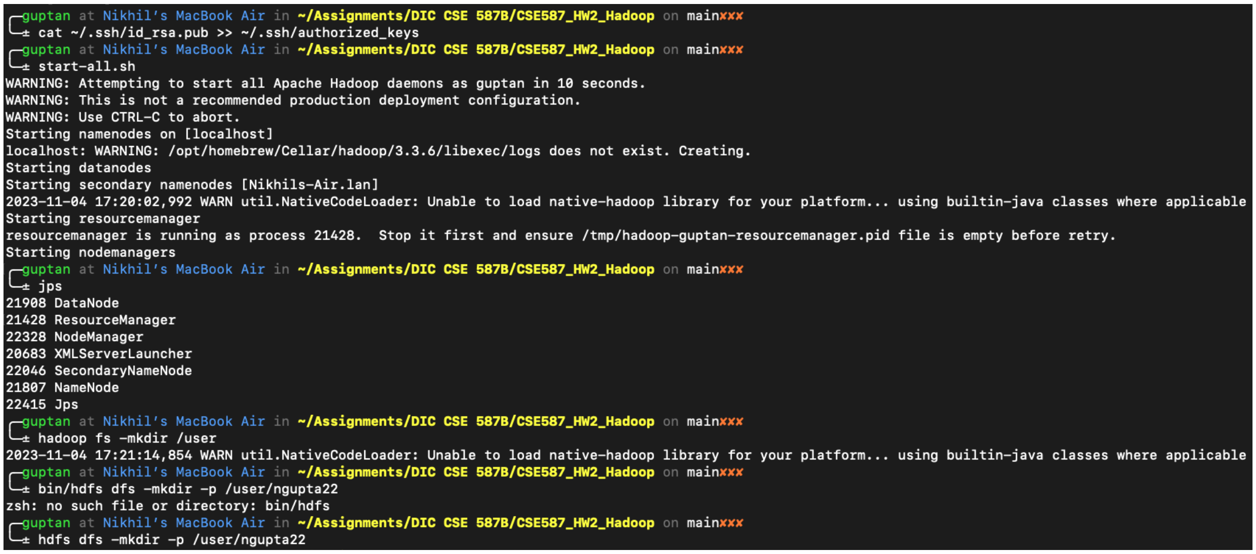
```
CSE587_HW2_HADOOP
  books
    adollhouse.txt
    germanpirate.txt
    metamorphosis.txt
    romeojuilet.txt
    theprince.txt
  samples
    file1.txt
    file2.txt
    file3.txt
  Homework 2 - MR.pdf
  patterns.txt
  wc.jar
  WordCount.class
  WordCount.java
  WordCount$IntSumCombiner.cla...
  WordCount$IntSumReducer.class
  WordCount$TokenizerMapper.cl...
```

- List of 5 books in books folder
- 3 samples text files to check if code is first functioning correctly or not, inside samples folder
- List of altogether 98 **stop words** in patterns.txt, uploaded on HDFS
    - A,an,the,in,my,has,as,if,do,have,had,on,at,of,for,by,with,to,up,down,and,or,not,but,is,am,are,was,were,be,being,been,it,this,that,these,those,i,me,myself,we,us,our,ours,you,your,yours,he,him,his,she,her,hers,it's,its,they,them,their,theirs,what,which,who,whom,whose,here,there,when,where,why,how,all,any,both,each,few,more,most,other,some,such,no,nor,not,only,own,same,so,than,too,very,s,t,can,will,just,don,should,now
- A single main Java application file as WordCount.java

Hadoop Installation

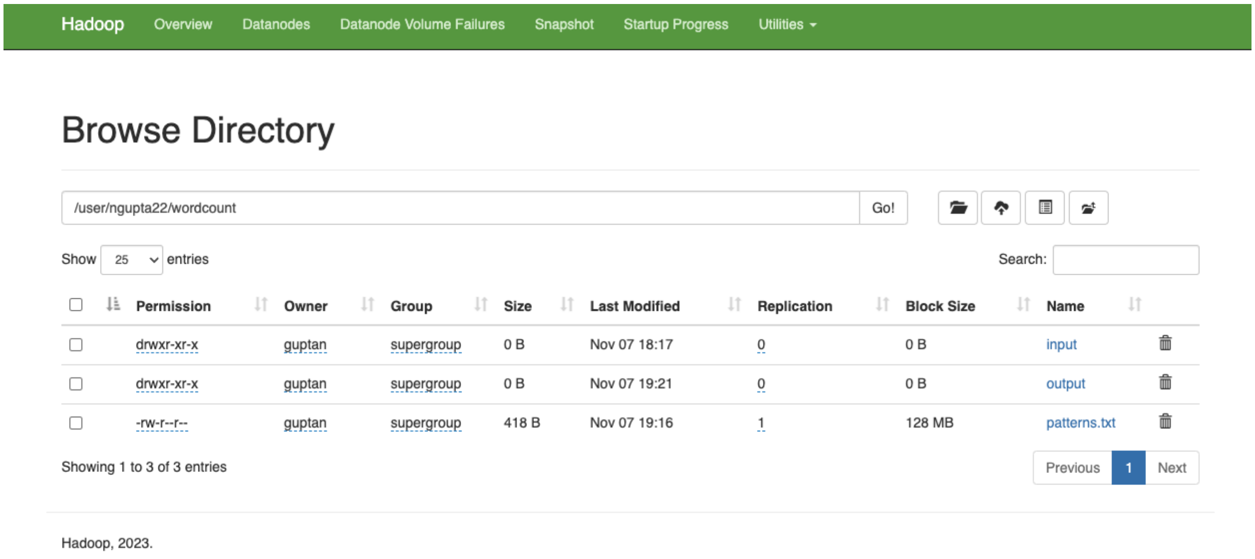


```
╭─guptan at Nikhil’s MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✘✘✘
╰─± start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as guptan in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
localhost: Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
localhost: guptan@localhost: Permission denied (publickey,password,keyboard-interactive).
Starting datanodes
localhost: guptan@localhost: Permission denied (publickey,password,keyboard-interactive).
Starting secondary namenodes [Nikhils-Air.lan]
Nikhils-Air.lan: Warning: Permanently added 'nikhils-air.lan' (ED25519) to the list of known hosts.
Nikhils-Air.lan: guptan@nikhils-air.lan: Permission denied (publickey,password,keyboard-interactive).
2023-11-04 17:18:08,991 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting resourcemanager
Starting nodemanagers
localhost: guptan@localhost: Permission denied (publickey,password,keyboard-interactive).
╭─guptan at Nikhil’s MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✘✘✘
╰─± jps
21539 Jps
21428 ResourceManager
20683 XMLServerLauncher
╭─guptan at Nikhil’s MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✘✘✘
╰─± ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
Your identification has been saved in /Users/guptan/.ssh/id_rsa
Your public key has been saved in /Users/guptan/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:9z+TwNlubhC5LO626XDP9sl4s49bEJEhrf/kPeXh2ps guptan@Nikhils-Air.lan
The key's randomart image is:
+---[RSA 3072]----+
|            ..oo |
|             .o. |
|             o.  |
|            +  . |
|        S .o *.  |
|         ...B ooo|
|        .....+.*=|
|         oo+.+@**|
|         +=o+B@E+|
+----[SHA256]-----+
╭─guptan at Nikhil’s MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✘✘✘
╰─± cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
╭─guptan at Nikhil’s MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✘✘✘
╰─± start-all.sh
```



```
╭─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✗✗✗
╰─± cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
╭─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✗✗✗
╰─± start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as guptan in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
localhost: WARNING: /opt/homebrew/Cellar/hadoop/3.3.6/libexec/logs does not exist. Creating.
Starting datanodes
Starting secondary namenodes [Nikhils-Air.lan]
2023-11-04 17:20:02,992 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting resourcemanager
resourcemanager is running as process 21428.  Stop it first and ensure /tmp/hadoop-guptan-resourcemanager.pid file is empty before retry.
Starting nodemanagers
╭─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✗✗✗
╰─± jps
21908 DataNode
21428 ResourceManager
22328 NodeManager
20683 XMLServerLauncher
22046 SecondaryNameNode
21807 NameNode
22415 Jps
╭─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✗✗✗
╰─± hadoop fs -mkdir /user
2023-11-04 17:21:14,854 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
╭─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✗✗✗
╰─± bin/hdfs dfs -mkdir -p /user/ngupta22
zsh: no such file or directory: bin/hdfs
╭─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on main✗✗✗
╰─± hdfs dfs -mkdir -p /user/ngupta22
```

Hadoop Web Application on Localhost with all the required folders and files



Browse Directory

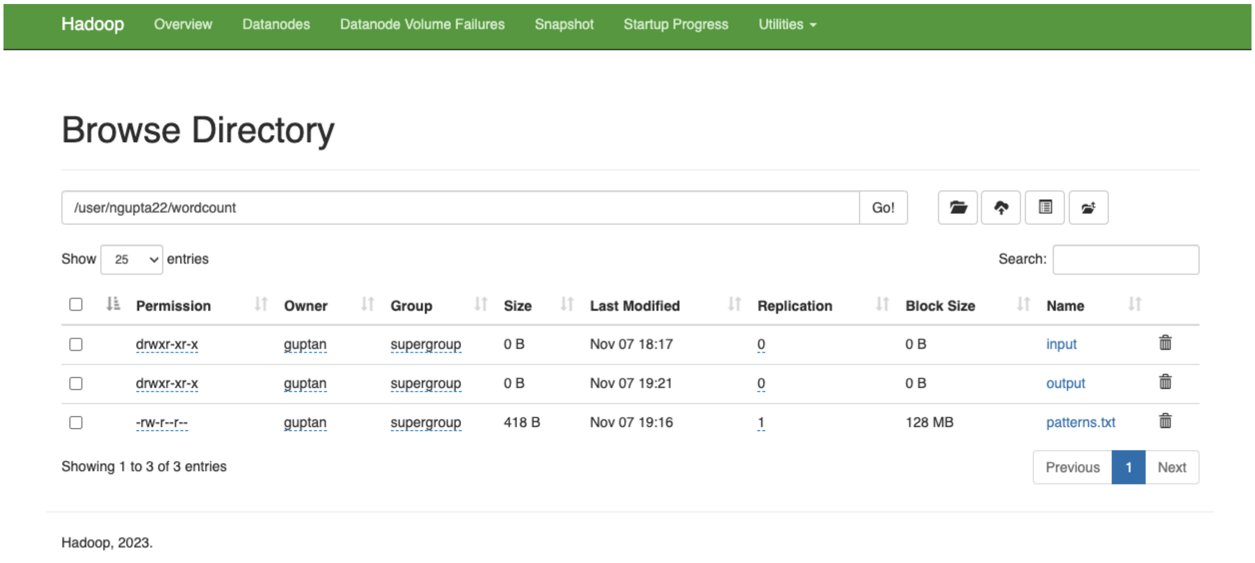
/user/ngupta22/wordcount Go!

Show 25 entries   Search:

□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
□	drwxr-xr-x	guptan	supergroup	0 B	Nov 07 18:17	0	0 B	input	🗑
□	drwxr-xr-x	guptan	supergroup	0 B	Nov 07 19:21	0	0 B	output	🗑
□	-rw-r--r--	guptan	supergroup	418 B	Nov 07 19:16	1	128 MB	patterns.txt	🗑

Showing 1 to 3 of 3 entries

Hadoop, 2023.



Browse Directory

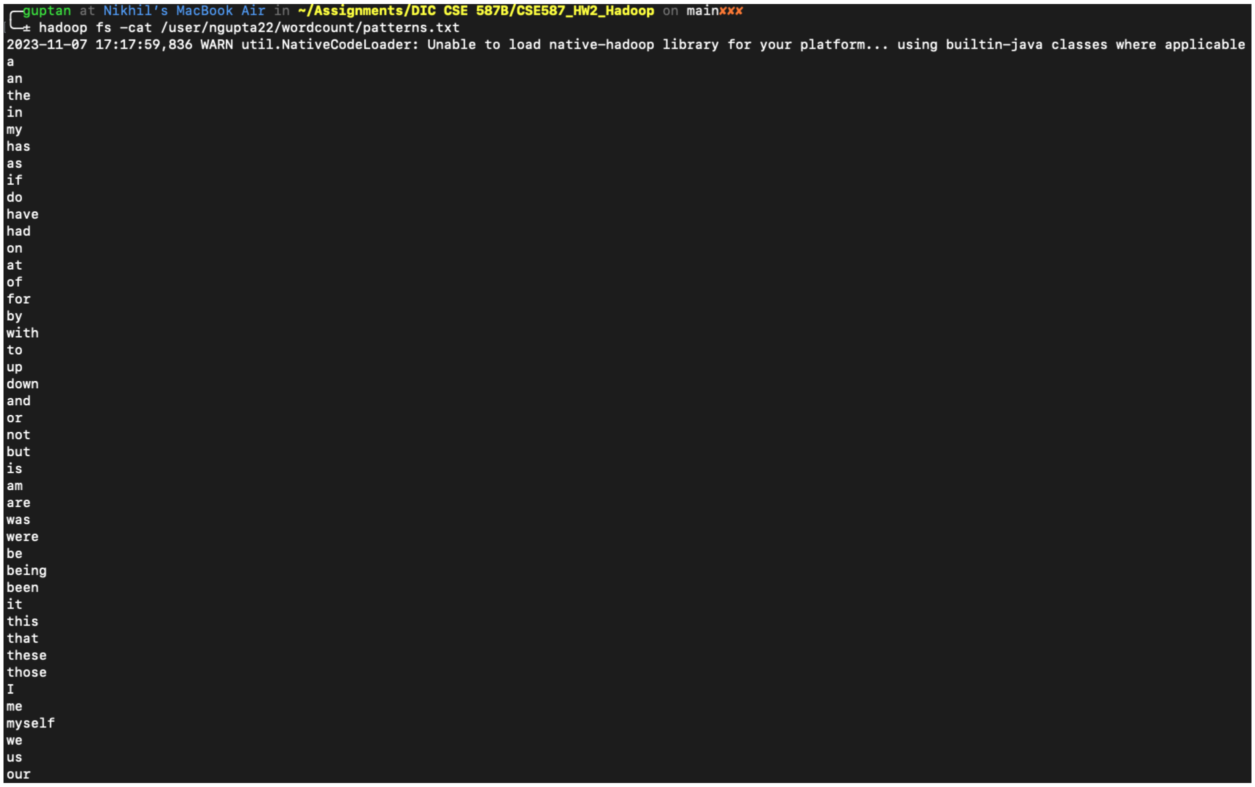
/user/ngupta22/wordcount

□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
□	drwxr-xr-x	guptan	supergroup	0 B	Nov 07 18:17	0	0 B	input
□	drwxr-xr-x	guptan	supergroup	0 B	Nov 07 19:21	0	0 B	output
□	-rw-r--r--	guptan	supergroup	418 B	Nov 07 19:16	1	128 MB	patterns.txt

Showing 1 to 3 of 3 entries

Hadoop, 2023.

Cat Pattern.txt from HDFS



```
┌─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on mainxxx
└─± hadoop fs -cat /user/ngupta22/wordcount/patterns.txt
2023-11-07 17:17:59,836 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
a
an
the
in
my
has
as
if
do
have
had
on
at
of
for
by
with
to
up
down
and
or
not
but
is
am
are
was
were
be
being
been
it
this
that
these
those
I
me
myself
we
us
our
```

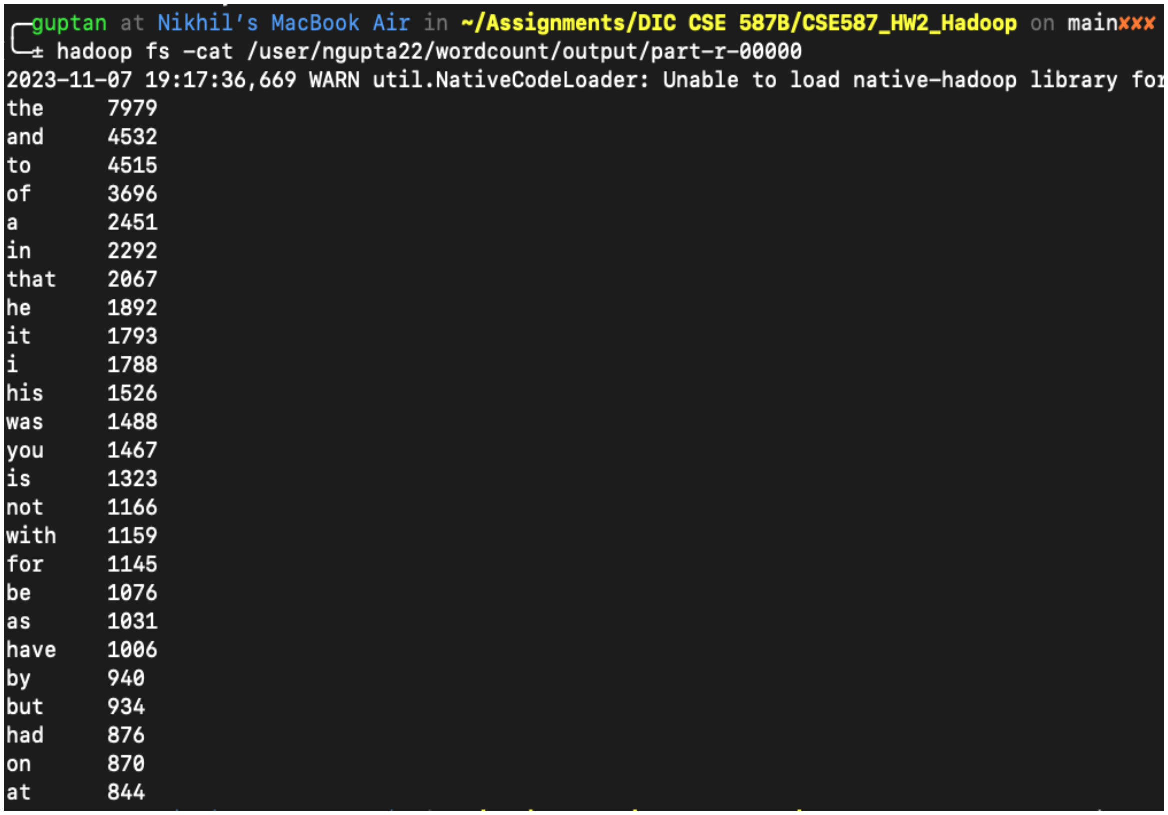
Useful Hadoop commands throughout homework

- hadoop fs -mkdir /user/ngupta22/wordcount/input/samples
- hadoop com.sun.tools.javac.Main WordCount.java
- jar cf wc.jar WordCount*.class
- hadoop fs -ls /user/ngupta22/wordcount/input/
- hadoop fs -cat /user/ngupta22/wordcount/input/file1.txt
- hadoop fs -rm -r /user/ngupta22/wordcount/output
- hadoop jar wc.jar WordCount /user/ngupta22/wordcount/input/books /user/ngupta22/wordcount/output
- hadoop fs -cat /user/ngupta22/wordcount/output/part-r-00000
- hadoop fs -rm /user/ngupta22/wordcount/input/file3.txt
- hadoop fs -put books/* /user/ngupta22/wordcount/input/books
- hadoop jar wc.jar WordCount /user/ngupta22/wordcount/input/samples /user/ngupta22/wordcount/output -skip /user/ngupta22/wordcount/patterns.txt

Analysis

1. What are the 25 most common words and the number of occurrences of each when you do not remove stopwords?

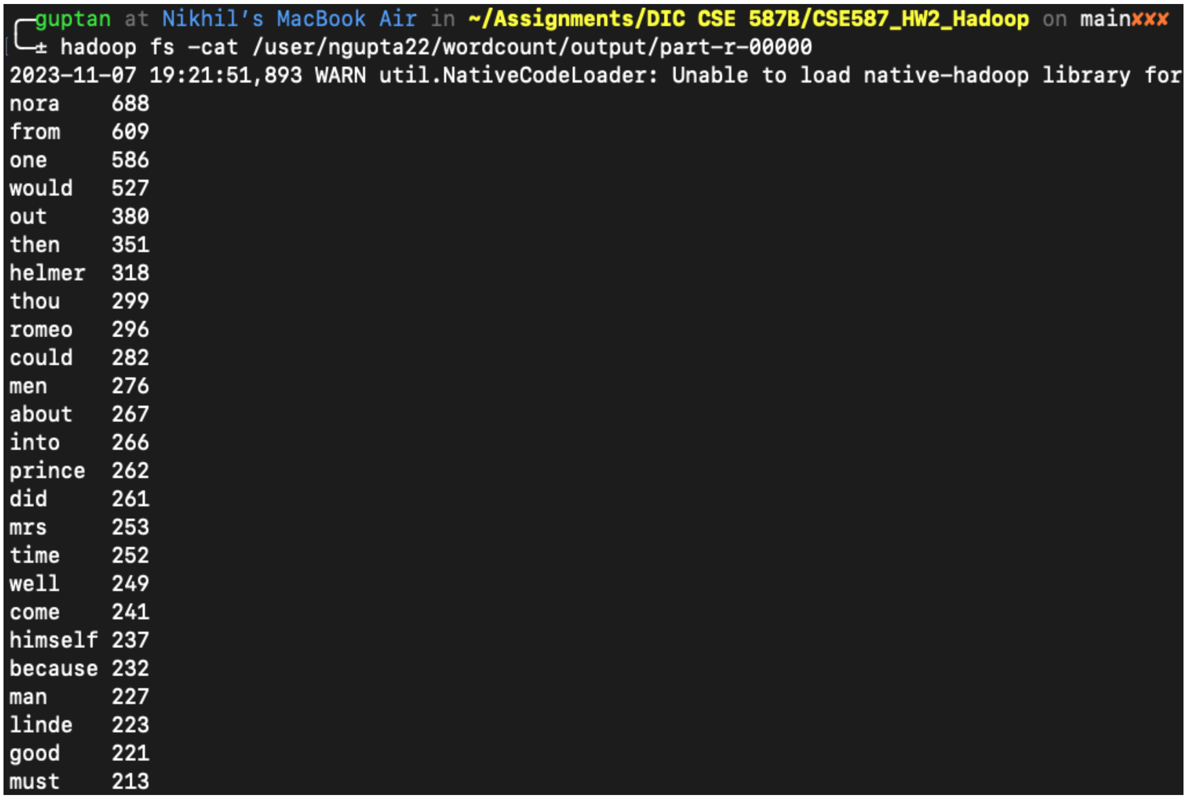
   Below is an attached screenshot of the output showing 25 most common words and the number of occurrences of each without removing stop words.



```
┌─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on mainxxx
└─± hadoop fs -cat /user/ngupta22/wordcount/output/part-r-00000
2023-11-07 19:17:36,669 WARN util.NativeCodeLoader: Unable to load native-hadoop library for
the     7979
and     4532
to      4515
of      3696
a       2451
in      2292
that    2067
he      1892
it      1793
i       1788
his     1526
was     1488
you     1467
is      1323
not     1166
with    1159
for     1145
be      1076
as      1031
have    1006
by      940
but     934
had     876
on      870
at      844
```

2. What are the 25 most common words and the number of occurrences of each when you do remove stopwords?

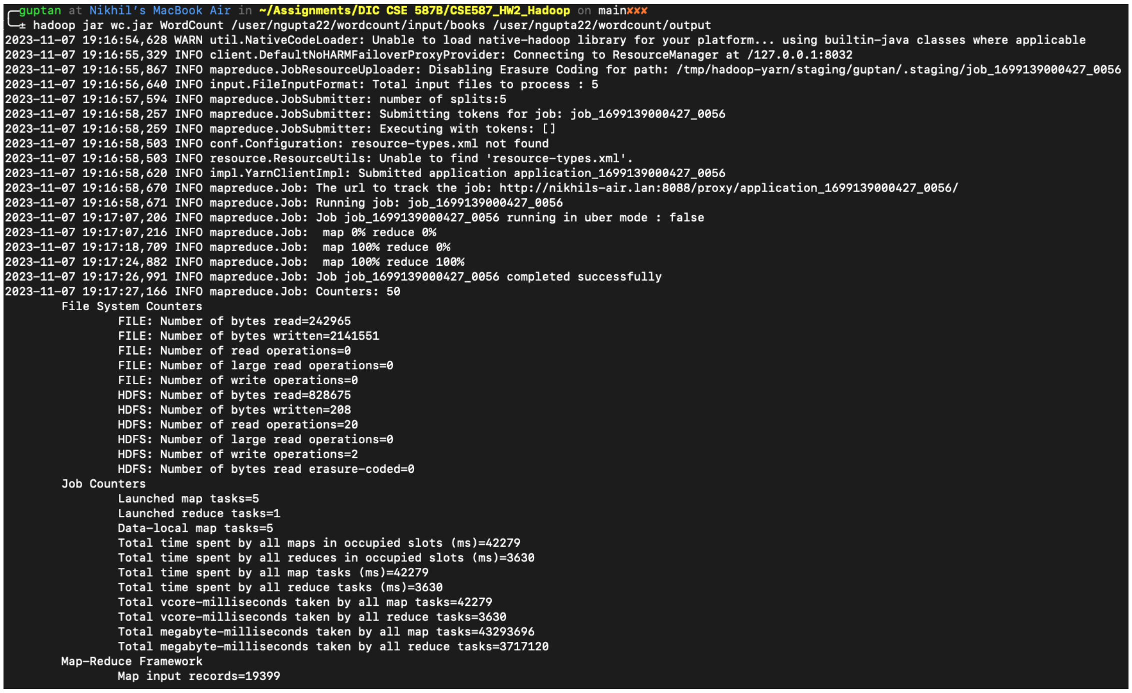
   Below is an attached screenshot of the output showing 25 most common words and the number of occurrences of each without removing stop words.



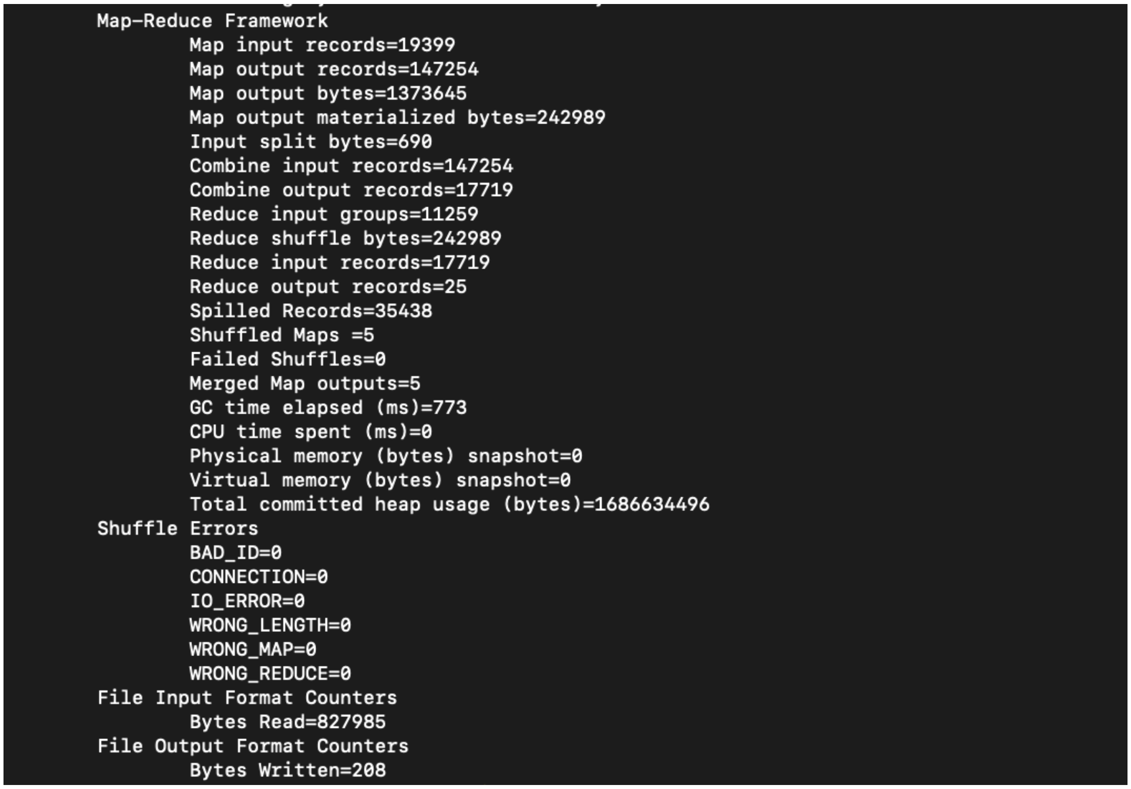
```
┌─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on mainxxx
└─± hadoop fs -cat /user/ngupta22/wordcount/output/part-r-00000
2023-11-07 19:21:51,893 WARN util.NativeCodeLoader: Unable to load native-hadoop library for
nora    688
from    609
one     586
would   527
out     380
then    351
helmer  318
thou    299
romeo   296
could   282
men     276
about   267
into    266
prince  262
did     261
mrs     253
time    252
well    249
come    241
himself 237
because 232
man     227
linde   223
good    221
must    213
```

3. Based on the output of your application, how does removing stop words affect the total amount of bytes output by your mappers? Name one concrete way that this would affect the performance of your application.

   Below is an attached screenshot showcasing mapper’s output **without** -skip stop words:

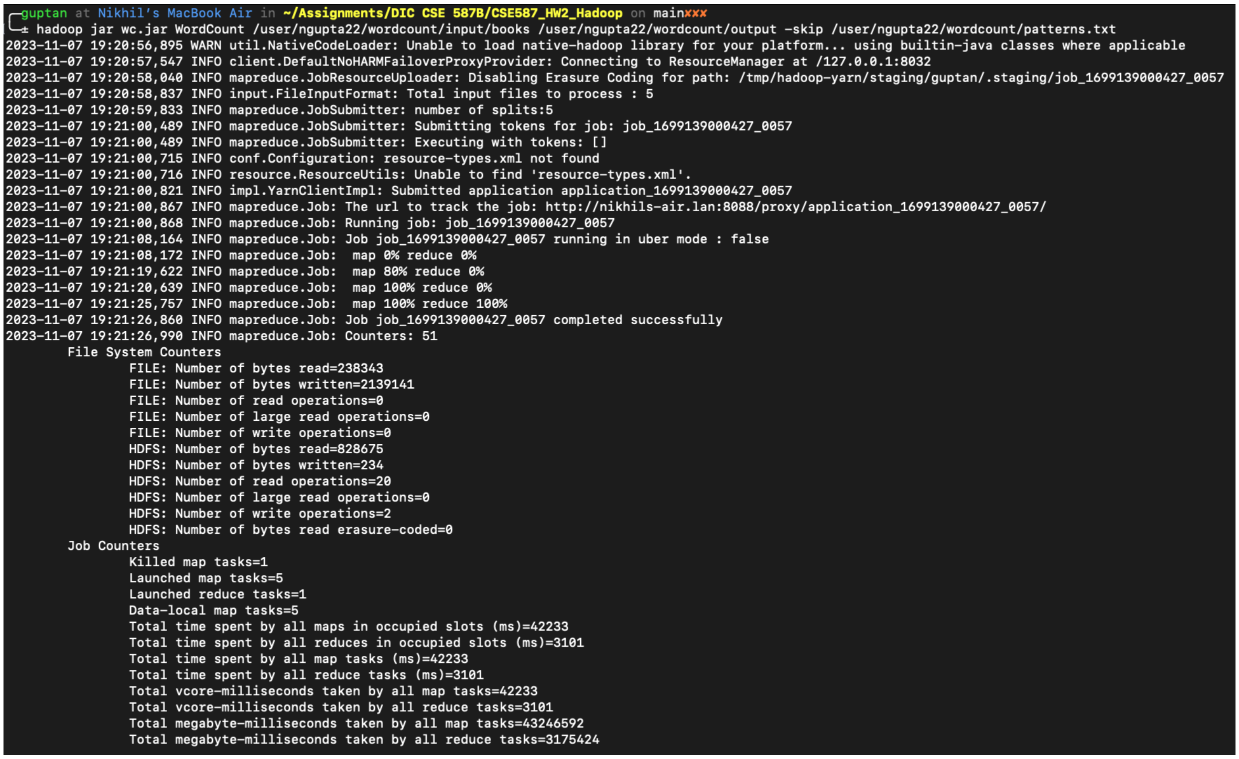


```
╭─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on mainxxx
╰─± hadoop jar wc.jar WordCount /user/ngupta22/wordcount/input/books /user/ngupta22/wordcount/output
2023-11-07 19:16:54,628 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2023-11-07 19:16:55,329 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /127.0.0.1:8032
2023-11-07 19:16:55,867 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/guptan/.staging/job_1699139000427_0056
2023-11-07 19:16:56,640 INFO input.FileInputFormat: Total input files to process : 5
2023-11-07 19:16:57,594 INFO mapreduce.JobSubmitter: number of splits:5
2023-11-07 19:16:58,257 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1699139000427_0056
2023-11-07 19:16:58,259 INFO mapreduce.JobSubmitter: Executing with tokens: []
2023-11-07 19:16:58,503 INFO conf.Configuration: resource-types.xml not found
2023-11-07 19:16:58,503 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2023-11-07 19:16:58,620 INFO impl.YarnClientImpl: Submitted application application_1699139000427_0056
2023-11-07 19:16:58,670 INFO mapreduce.Job: The url to track the job: http://nikhils-air.lan:8088/proxy/application_1699139000427_0056/
2023-11-07 19:16:58,671 INFO mapreduce.Job: Running job: job_1699139000427_0056
2023-11-07 19:17:07,206 INFO mapreduce.Job: Job job_1699139000427_0056 running in uber mode : false
2023-11-07 19:17:07,216 INFO mapreduce.Job:  map 0% reduce 0%
2023-11-07 19:17:18,709 INFO mapreduce.Job:  map 100% reduce 0%
2023-11-07 19:17:24,882 INFO mapreduce.Job:  map 100% reduce 100%
2023-11-07 19:17:26,991 INFO mapreduce.Job: Job job_1699139000427_0056 completed successfully
2023-11-07 19:17:27,166 INFO mapreduce.Job: Counters: 50
        File System Counters
                FILE: Number of bytes read=242965
                FILE: Number of bytes written=2141551
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=828675
                HDFS: Number of bytes written=208
                HDFS: Number of read operations=20
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=5
                Launched reduce tasks=1
                Data-local map tasks=5
                Total time spent by all maps in occupied slots (ms)=42279
                Total time spent by all reduces in occupied slots (ms)=3630
                Total time spent by all map tasks (ms)=42279
                Total time spent by all reduce tasks (ms)=3630
                Total vcore-milliseconds taken by all map tasks=42279
                Total vcore-milliseconds taken by all reduce tasks=3630
                Total megabyte-milliseconds taken by all map tasks=43293696
                Total megabyte-milliseconds taken by all reduce tasks=3717120
        Map-Reduce Framework
                Map input records=19399
```

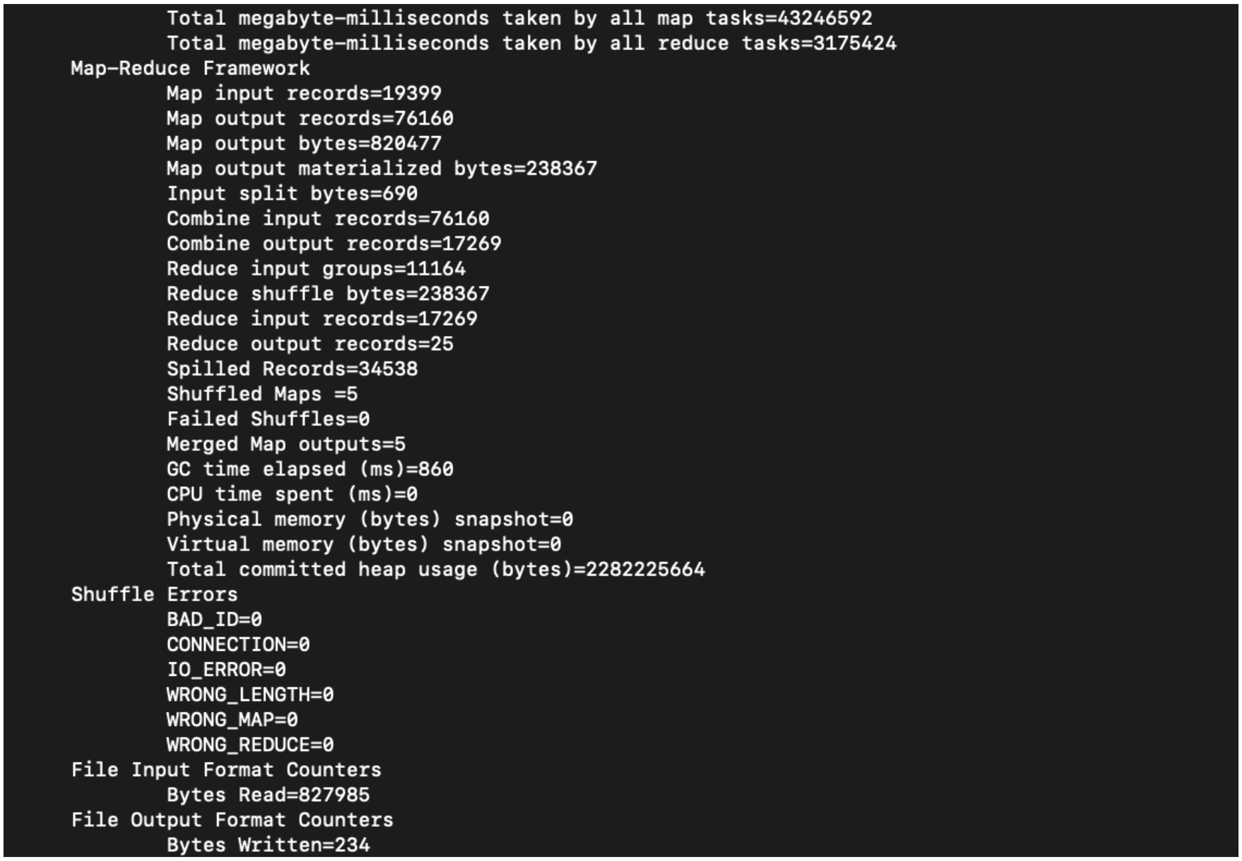


```
        Map-Reduce Framework
                Map input records=19399
                Map output records=147254
                Map output bytes=1373645
                Map output materialized bytes=242989
                Input split bytes=690
                Combine input records=147254
                Combine output records=17719
                Reduce input groups=11259
                Reduce shuffle bytes=242989
                Reduce input records=17719
                Reduce output records=25
                Spilled Records=35438
                Shuffled Maps =5
                Failed Shuffles=0
                Merged Map outputs=5
                GC time elapsed (ms)=773
                CPU time spent (ms)=0
                Physical memory (bytes) snapshot=0
                Virtual memory (bytes) snapshot=0
                Total committed heap usage (bytes)=1686634496
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=827985
        File Output Format Counters
                Bytes Written=208
```

Below is an attached screenshot showcasing mapper’s output **with** -skip stop words:



```
┌─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on mainxxx
└─± hadoop jar wc.jar WordCount /user/ngupta22/wordcount/input/books /user/ngupta22/wordcount/output -skip /user/ngupta22/wordcount/patterns.txt
2023-11-07 19:20:56,895 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2023-11-07 19:20:57,547 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /127.0.0.1:8032
2023-11-07 19:20:58,040 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/guptan/.staging/job_1699139000427_0057
2023-11-07 19:20:58,837 INFO input.FileInputFormat: Total input files to process : 5
2023-11-07 19:20:59,833 INFO mapreduce.JobSubmitter: number of splits:5
2023-11-07 19:21:00,489 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1699139000427_0057
2023-11-07 19:21:00,489 INFO mapreduce.JobSubmitter: Executing with tokens: []
2023-11-07 19:21:00,715 INFO conf.Configuration: resource-types.xml not found
2023-11-07 19:21:00,716 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2023-11-07 19:21:00,821 INFO impl.YarnClientImpl: Submitted application application_1699139000427_0057
2023-11-07 19:21:00,867 INFO mapreduce.Job: The url to track the job: http://nikhils-air.lan:8088/proxy/application_1699139000427_0057/
2023-11-07 19:21:00,868 INFO mapreduce.Job: Running job: job_1699139000427_0057
2023-11-07 19:21:08,164 INFO mapreduce.Job: Job job_1699139000427_0057 running in uber mode : false
2023-11-07 19:21:08,172 INFO mapreduce.Job:  map 0% reduce 0%
2023-11-07 19:21:19,622 INFO mapreduce.Job:  map 80% reduce 0%
2023-11-07 19:21:20,639 INFO mapreduce.Job:  map 100% reduce 0%
2023-11-07 19:21:25,757 INFO mapreduce.Job:  map 100% reduce 100%
2023-11-07 19:21:26,860 INFO mapreduce.Job: Job job_1699139000427_0057 completed successfully
2023-11-07 19:21:26,990 INFO mapreduce.Job: Counters: 51
        File System Counters
                FILE: Number of bytes read=238343
                FILE: Number of bytes written=2139141
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=828675
                HDFS: Number of bytes written=234
                HDFS: Number of read operations=20
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Killed map tasks=1
                Launched map tasks=5
                Launched reduce tasks=1
                Data-local map tasks=5
                Total time spent by all maps in occupied slots (ms)=42233
                Total time spent by all reduces in occupied slots (ms)=3101
                Total time spent by all map tasks (ms)=42233
                Total time spent by all reduce tasks (ms)=3101
                Total vcore-milliseconds taken by all map tasks=42233
                Total vcore-milliseconds taken by all reduce tasks=3101
                Total megabyte-milliseconds taken by all map tasks=43246592
                Total megabyte-milliseconds taken by all reduce tasks=3175424
```



```
                Total megabyte-milliseconds taken by all map tasks=43246592
                Total megabyte-milliseconds taken by all reduce tasks=3175424
        Map-Reduce Framework
                Map input records=19399
                Map output records=76160
                Map output bytes=820477
                Map output materialized bytes=238367
                Input split bytes=690
                Combine input records=76160
                Combine output records=17269
                Reduce input groups=11164
                Reduce shuffle bytes=238367
                Reduce input records=17269
                Reduce output records=25
                Spilled Records=34538
                Shuffled Maps =5
                Failed Shuffles=0
                Merged Map outputs=5
                GC time elapsed (ms)=860
                CPU time spent (ms)=0
                Physical memory (bytes) snapshot=0
                Virtual memory (bytes) snapshot=0
                Total committed heap usage (bytes)=2282225664
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=827985
        File Output Format Counters
                Bytes Written=234
```

Conclusion: We can notice the significant drop in overall Map-Reduce Framework numbers by removing the stop words. Below is a table with detailed comparison of few parameters:-

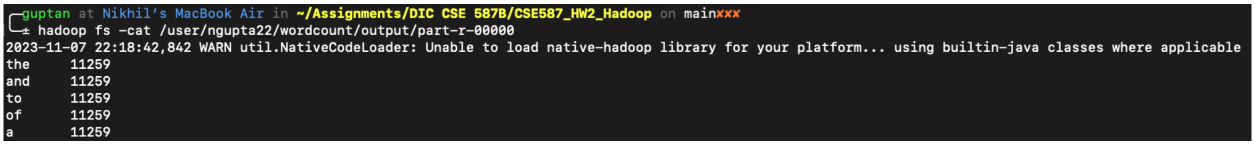
	Without removing Stop Words	Removing Stop Words
Map output records	147254	76160
Map output bytes	1373645	820477
Map output materialized bytes	242989	238367
Combine output records	17719	17269

Just for an instance from the above table we can observe that Map output records dropped from 147254 to 76160 and Map output bytes dropped from 1374645 to 820477.

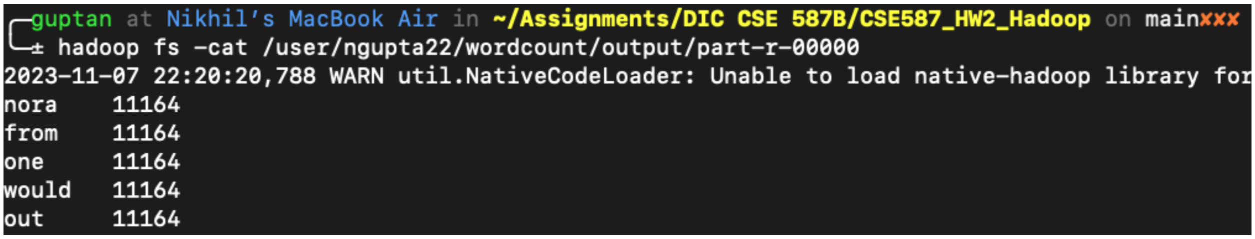
Removing stop words can significantly impact on MapReduce performance as stop words are the words that occurs frequently in any texts but carries little to no meaning, therefore by simply omitting it, it reduces the amount of data that is required to be transferred to the shuffle and reduce phase, generating less key value pairs and eventually increasing overall performance of MapReduce and speeding up processing time, all of these consuming less network bandwidth and I/O process. In the above table we can observe that by getting rid of stop words, Map output bytes dropped from **1374645** to **820477**, which is decrease in **68%.**

4. Based on the output of your application, what is the size of your keyspace with and without removing stopwords? How does this correspond to the number of stopwords you have chosen to remove?

   Below is an attached screenshot of number of keyspaces without removing and removing stop words:



```
┌─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on mainxxx
└─± hadoop fs -cat /user/ngupta22/wordcount/output/part-r-00000
2023-11-07 22:18:42,842 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
the     11259
and     11259
to      11259
of      11259
a       11259
```



```
┌─guptan at Nikhil's MacBook Air in ~/Assignments/DIC CSE 587B/CSE587_HW2_Hadoop on mainxxx
└─± hadoop fs -cat /user/ngupta22/wordcount/output/part-r-00000
2023-11-07 22:20:20,788 WARN util.NativeCodeLoader: Unable to load native-hadoop library for
nora    11164
from    11164
one     11164
would   11164
out     11164
```

- Size of keyspace without removing stopwords: 11259
- Size of keyspace with removing stopwords: 11164

  This is difference of 95 keyspace. Assuming that we take 'S' number of stopwords, our keyspace after removing stopwords should ideally be: Size of keyspace with removing stopwords + 'S'. This happenes only when all the stop words shows up as input in the keyspace atleast once.
  However in our case, we have '98' stopwords. Out of which 95 stopwords has been occurred in out input text files and rest 3 stopwords are unique, therefore contributing to the Keyspace.

5. Let's now assume you were going to run your application on the entirety of Project Gutenberg. For this question, assume that there are 100TB of input data, the data is spread over 10 sites, and each site has 20 mappers. Assume you ignore all but the 25 most common words that you listed in question 2. Furthermore, assume that your combiners have been run optimally so that each combiner will output at most 1 keyvalue pair per key.

   a. How much data will each mapper have to parse?
   -> Total input size = 100TB, spreaded over 10 sites. Mapper = 20
   Therefore, each mapper has to parse, 100/10 = 10 and then again, 10/20=**0.5TB** data

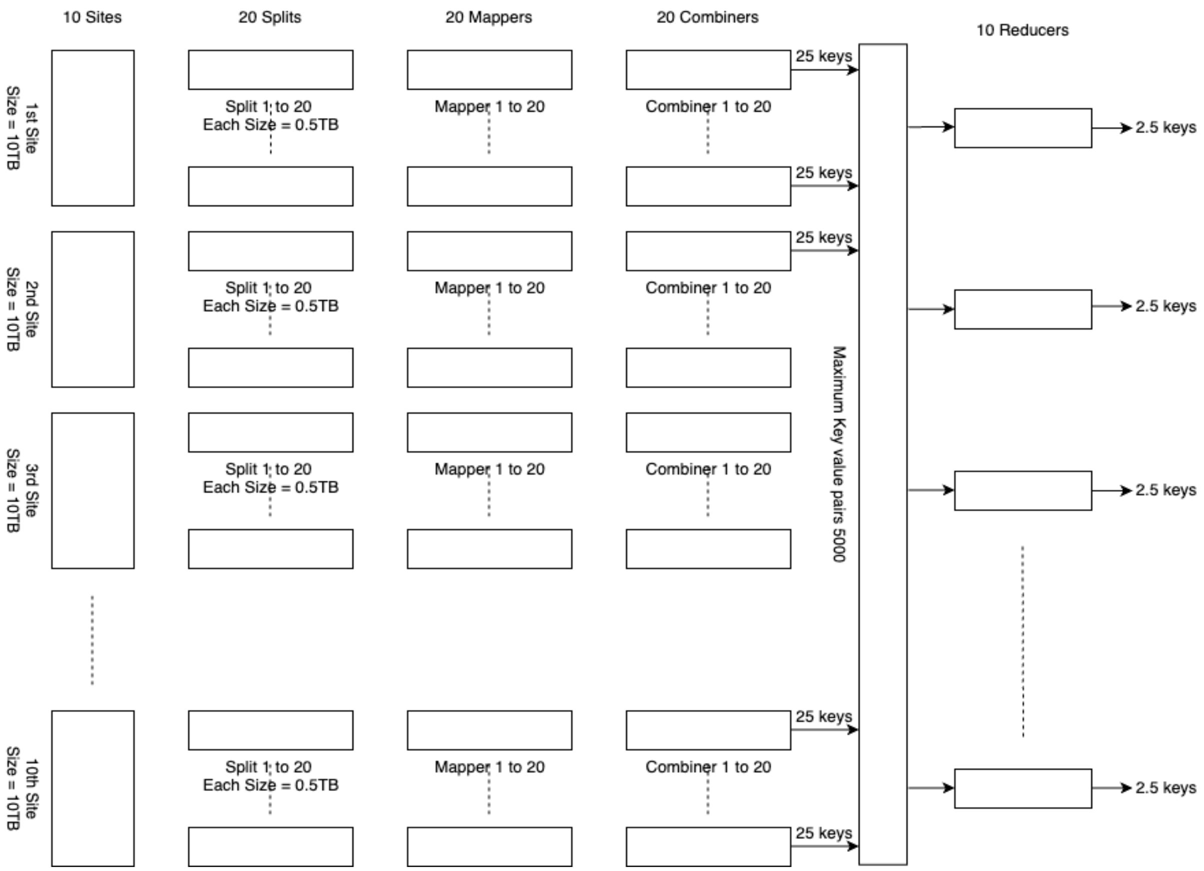
   b. What is the size of your keyspace?
   → Size of keyspace= most common words = **25**

   c. What is the maximum number of key-value pairs that could be communicated during the barrier between mapping and reducing?
   → Keyspace size = 25, number of sites = 10
   Mapper per site = 20

   Therefore, maximum number of key value pairs = 25*10*20 = **5000**

   d. Assume you are running one reducer per site. On average, how many key-value pairs will each reducer have to handle?
   → Total number of reducer = one per site = 10
   Number of words = 25
   Each reducer have to handle minimum of 25/10 = 2.5 key pairs. Also we have maximum number of key value pairs as 5000. Each site having 20 mappers. Therefore for 5000 key pairs, we have 20 reducers * 25 most common words = 500 key pairs to handle.

6. Draw the data flow diagram for question 5. The diagram should be similar to the diagram shown in the lecture. On your diagram, label the specific quantities you got for 5a,b,c, and d.



References:

- https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/SingleCluster.html
- https://www.gutenberg.org/ebooks/
- https://hadoop.apache.org/docs/stable/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html
- https://app.diagrams.net/