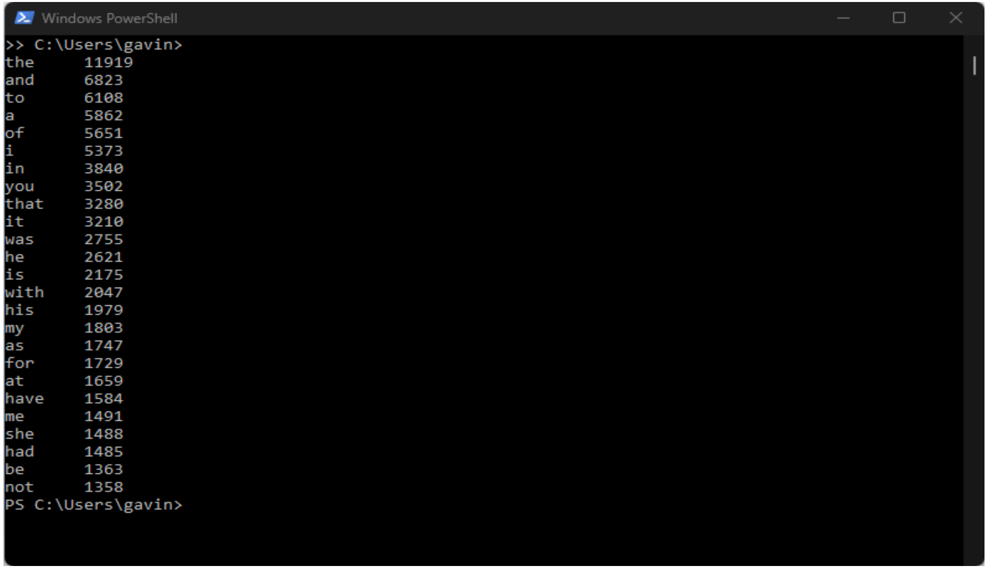
Homework #2 – MapReduce

Books Used:

AdvofSherlockHolmes.txt, AliceInWonderland.txt, DollsHouse.txt, GreatGatsby.txt, RomeoandJuliet.txt

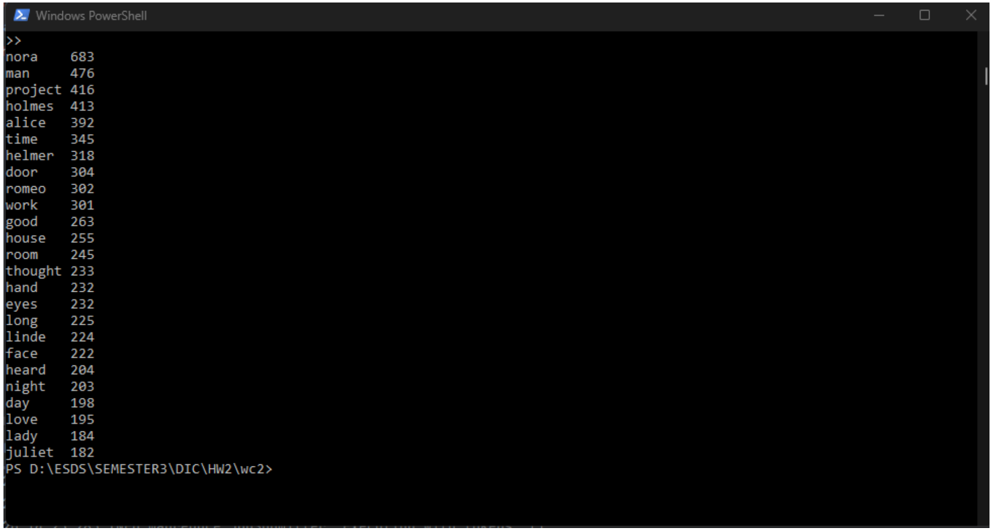
Analysis:

1) What are the 25 most common words and the number of occurrences of each when you do not remove stop words?
OUTPUT: 25 most common words with no. of occurrences with stop words



```
>> C:\Users\gavin>
the     11919
and     6823
to      6108
a       5862
of      5651
i       5373
in      3840
you     3502
that    3280
it      3210
was     2755
he      2621
is      2175
with    2047
his     1979
my      1803
as      1747
for     1729
at      1659
have    1584
me      1491
she     1488
had     1485
be      1363
not     1358
PS C:\Users\gavin>
```

2) What are the 25 most common words and the number of occurrences of each when you do remove stop words?
OUTPUT: 25 most common words with no. of occurrences without stop words

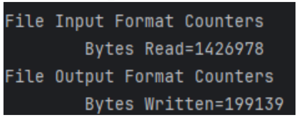


```
>>
nora    683
man     476
project 416
holmes  413
alice   392
time    345
helmer  318
door    304
romeo   302
work    301
good    263
house   255
room    245
thought 233
hand    232
eyes    232
long    225
linde   224
face    222
heard   204
night   203
day     198
love    195
lady    184
juliet  182
PS D:\ESDS\SEMESTER3\DIC\HW2\wc2>
```

Reference to Stop Words:
https://gist.github.com/sebleier/554280?permalink_comment_id=3431590#gistcomment-3431590

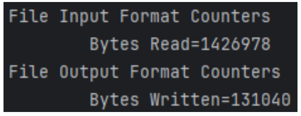
3) Based on the output of your application, how does removing stop words affect the total amount of bytes output by your mappers? Name one concrete way that this would affect the performance of your application.
Answer: The total amount of bytes that needs to be processed and sent between the map and reduce can be greatly reduced by removing the stop words from the mapper output. Stop words are removed to focus on just the meaningful words. The impact of removing stop words will change depending on the data/books we use. Removing stop words normally reduces the amount of data that is being processed and transferred between map and reduce helping to improve the performance overall.

With Stop Words: 199139 bytes



```
File Input Format Counters
        Bytes Read=1426978
File Output Format Counters
        Bytes Written=199139
```

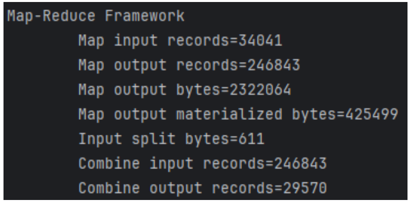
Removing Stop Words: 131040 bytes



```
File Input Format Counters
        Bytes Read=1426978
File Output Format Counters
        Bytes Written=131040
```

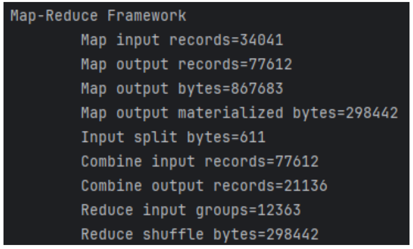
4) Based on the output of your application, what is the size of your keyspace with and without removing stopwords? How does this correspond to the number of stopwords you have chosen to remove?
Answer: The stopwords list was taken from https://gist.github.com/sebleier/554280?permalink_comment_id=3431590#gistcomment-3431590
Removing these stopwords before processing in MapReduce reduced the keyspace significantly. The keyspace are the total number of unique words in the .txt books files. Removing the stopwords shrinks because the unique stopwords are removed from the unique words in the keyspace. The impact on the size of the keyspace depends on the stopwords list. If everything from the stopwords list appear in the books, it will minus the occurrence of these. But not all stopwords will appear in the books, and not all words in the books will be on the stopwords list. So, the keyspace reduction will usually be less than the total number of stopwords in the list.

With Stop Words: (Combine Output records: 29570)



```
Map-Reduce Framework
        Map input records=34041
        Map output records=246843
        Map output bytes=2322064
        Map output materialized bytes=425499
        Input split bytes=611
        Combine input records=246843
        Combine output records=29570
```

Removing Stop Words: (Combine Output records: 21136)



```
Map-Reduce Framework
        Map input records=34041
        Map output records=77612
        Map output bytes=867683
        Map output materialized bytes=298442
        Input split bytes=611
        Combine input records=77612
        Combine output records=21136
        Reduce input groups=12363
        Reduce shuffle bytes=298442
```

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 key value pair per key.

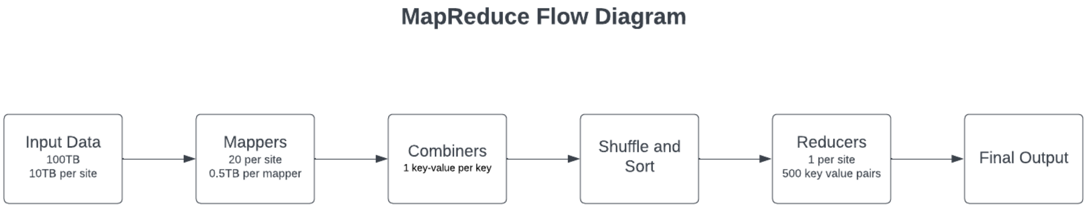
a. How much data will each mapper have to parse?
Answer: 100TB data, 10 Sites, 20 Mappers/site
100/10 = 10TB per site
10/20 = 0.5TB or 500GB per Mapper
Each of the mappers have to parse 0.5TB or 500GB of data.

b. What is the size of your keyspace?
Answer: Keyspace is the set of unique keys that the mappers can output and ignoring all but the 25 most common words that you listed in question 2, the keyspace size is 25.

c. What is the maximum number of key-value pairs that could be communicated during the barrier between mapping and reducing?
Answer: Each combiner will output at most 1 key value pair per key. 25 Keyspace. 20 mappers. 20 sites.
Total Mappers: 20 x 10 = 200. The maximum number of key-value pairs that could be communicated during the barrier between mapping and reducing is 25 x 200 = 5000
Hence, it is 5000 Key value pairs.

d. Assume you are running one reducer per site. On average, how many key-value pairs will each reducer have to handle?
Answer: 5000 Key value pairs / 10 sites = 500 key value pairs.
Each reducer will handle 500 key value pairs.

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.
Answer:


MapReduce Flow Diagram
Input Data
100TB
10TB per site
Mappers
20 per site
0.5TB per mapper
Combiners
1 key-value per key
Shuffle and Sort
Reducers
1 per site
500 key value pairs
Final Output