Task – 3

Aim: Implement EDA -Exploratory data analysis Module using MapReduce. come up with proper problem architecture with algorithms and document well.

Theory:

- EDA stands for Exploratory data analysis. It is an approach to analyze data sets to summarize their main characteristics, often using statistical graphics and other data visualization methods.
- MapReduce is the tool which distributes the data and process it in parallel manner. It mainly consists of two phases Mapper And Reducer. The mapper takes the data generates the data in key-value pair then output of mapper considered as input for reduce which reduces the same key values and generates the output.

Algorithm:

- Start
- I extract the data by web scraping using BeautifulSoup library in python.
- Then I clean the data such as
  a) Covert each word into lowercase
  b) Tokenization
  c) Remove Stopping words
  d) Stemming/ Lemmatization
  e) Remove Special Symbols
- Then I pass the data into map reduce using Hadoop streaming, here I count the frequency of each word.
- For Visualization of data I use Elk Stalk pipeline, I create a index in Kibana and pass the output file using logstash configuration file into the kibana index.
- End.

Program Code:

- **Mapper.py**

```
#!/usr/bin/env python
"""mapper.py"""
import sys
for line in sys.stdin:
line = line.strip()
words = line.split()
# increase counters
for word in words:
        print '%s\t%s' % (word, 1)
```

- **Reducer.py**

```
#!/usr/bin/env python
"""reducer.py"""
from operator import itemgetter
import sys
current_word = None
current_count = 0
word = None
for line in sys.stdin:
    line = line.strip()
word, count = line.split('\t', 1)
try:
count = int(count)
except ValueError:
continue
if current_word == word:
```

current_count += count

else:

if current_word:

print '%s\t%s' % (current_word, current_count)

current_count = count

current_word = word

if current_word == word:

print '%s\t%s' % (current_word, current_count)

Outputs:

```
[cloudera@quickstart Desktop]$ hdfs dfs -put Task3.txt
[cloudera@quickstart Desktop]$ hdfs dfs -ls
Found 24 items
-rw-r--r--   1 cloudera cloudera    4705957 2020-11-27 02:02 Genome.fa
drwxr-xr-x   - cloudera cloudera          0 2020-11-07 02:34 Outpu12
drwxr-xr-x   - cloudera cloudera          0 2020-11-01 22:58 Output12
drwxr-xr-x   - cloudera cloudera          0 2020-11-01 22:47 Output13
-rw-r--r--   1 cloudera cloudera      57833 2021-05-28 01:41 Task3.txt
drwxr-xr-x   - cloudera cloudera          0 2020-12-04 01:12 assignment
drwxr-xr-x   - cloudera cloudera          0 2020-12-04 01:27 assignment1
drwxr-xr-x   - cloudera cloudera          0 2020-11-27 02:10 checkoutput
drwxr-xr-x   - cloudera cloudera          0 2020-10-30 09:09 hadoops1
drwxr-xr-x   - cloudera cloudera          0 2020-10-30 09:16 hadoops2
drwxr-xr-x   - cloudera cloudera          0 2020-10-30 09:44 hadoops3
drwxr-xr-x   - cloudera cloudera          0 2020-10-31 10:10 hadoops4
drwxr-xr-x   - cloudera cloudera          0 2020-10-31 11:29 hadoops5
drwxr-xr-x   - cloudera cloudera          0 2020-10-31 11:33 hadoops8
-rw-r--r--   1 cloudera cloudera         61 2020-10-31 09:58 inputfile
drwxr-xr-x   - cloudera cloudera          0 2020-11-08 22:05 output100
drwxr-xr-x   - cloudera cloudera          0 2020-11-08 22:13 output101
drwxr-xr-x   - cloudera cloudera          0 2020-11-08 22:17 output102
drwxr-xr-x   - cloudera cloudera          0 2020-12-04 08:39 output_assignment
drwxr-xr-x   - cloudera cloudera          0 2020-12-04 06:24 outputass
drwxr-xr-x   - cloudera cloudera          0 2020-10-31 10:03 outputs1
-rw-r--r--   1 cloudera cloudera     485246 2020-11-01 22:23 test_access_log
-rw-r--r--   1 cloudera cloudera         59 2020-11-08 22:02 wordcount
-rw-r--r--   1 cloudera cloudera         59 2020-11-08 22:11 wordcount.txt
[cloudera@quickstart Desktop]$
```

Fig1: I put the data file into the mapreduce.

```
[cloudera@quickstart Desktop]$ hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar -file mapper.py -mapper mapper.py -file reducer.py -reducer reducer.py -input Task3.txt -output Task3_output
21/05/28 01:46:17 WARN streaming.StreamJob: -file option is deprecated, please use generic option -files instead.
packageJobJar: [mapper.py, reducer.py] [/usr/lib/hadoop-mapreduce/hadoop-streaming-2.6.0-cdh5.13.0.jar] /tmp/streamjob5051405192482561066.jar tmpDir=null
21/05/28 01:46:21 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
21/05/28 01:46:22 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
21/05/28 01:46:23 WARN hdfs.DFSClient: Caught exception
java.lang.InterruptedException
        at java.lang.Object.wait(Native Method)
        at java.lang.Thread.join(Thread.java:1281)
        at java.lang.Thread.join(Thread.java:1355)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.closeResponder(DFSOutputStream.java:967)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.endBlock(DFSOutputStream.java:705)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.run(DFSOutputStream.java:894)
21/05/28 01:46:23 WARN hdfs.DFSClient: Caught exception
java.lang.InterruptedException
        at java.lang.Object.wait(Native Method)
        at java.lang.Thread.join(Thread.java:1281)
        at java.lang.Thread.join(Thread.java:1355)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.closeResponder(DFSOutputStream.java:967)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.endBlock(DFSOutputStream.java:705)
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.run(DFSOutputStream.java:894)
21/05/28 01:46:23 INFO mapred.FileInputFormat: Total input paths to process : 1
21/05/28 01:46:23 INFO mapreduce.JobSubmitter: number of splits:2
21/05/28 01:46:24 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1622188709635_0007
21/05/28 01:46:26 INFO impl.YarnClientImpl: Submitted application application_1622188709635_0007
21/05/28 01:46:27 INFO mapreduce.Job: The url to track the job: http://quickstart.cloudera:8088/proxy/application_1622188709635_0007/
21/05/28 01:46:27 INFO mapreduce.Job: Running job: job_1622188709635_0007
```

Fig2: Executing the Hadoop streaming command.

```
Found 25 items
-rw-r--r--   1 cloudera cloudera    4705957 2020-11-27 02:02 Genome.fa
drwxr-xr-x   - cloudera cloudera          0 2020-11-07 02:34 Outpu12
drwxr-xr-x   - cloudera cloudera          0 2020-11-01 22:58 Output12
drwxr-xr-x   - cloudera cloudera          0 2020-11-01 22:47 Output13
-rw-r--r--   1 cloudera cloudera      57833 2021-05-28 01:41 Task3.txt
drwxr-xr-x   - cloudera cloudera          0 2021-05-28 01:47 Task3_output
drwxr-xr-x   - cloudera cloudera          0 2020-12-04 01:12 assignment
drwxr-xr-x   - cloudera cloudera          0 2020-12-04 01:27 assignment1
drwxr-xr-x   - cloudera cloudera          0 2020-11-27 02:10 checkoutput
drwxr-xr-x   - cloudera cloudera          0 2020-10-30 09:09 hadoops1
drwxr-xr-x   - cloudera cloudera          0 2020-10-30 09:16 hadoops2
drwxr-xr-x   - cloudera cloudera          0 2020-10-30 09:44 hadoops3
drwxr-xr-x   - cloudera cloudera          0 2020-10-31 10:10 hadoops4
drwxr-xr-x   - cloudera cloudera          0 2020-10-31 11:29 hadoops5
drwxr-xr-x   - cloudera cloudera          0 2020-10-31 11:33 hadoops8
-rw-r--r--   1 cloudera cloudera         61 2020-10-31 09:58 inputfile
drwxr-xr-x   - cloudera cloudera          0 2020-11-08 22:05 output100
drwxr-xr-x   - cloudera cloudera          0 2020-11-08 22:13 output101
drwxr-xr-x   - cloudera cloudera          0 2020-11-08 22:17 output102
drwxr-xr-x   - cloudera cloudera          0 2020-12-04 08:39 output_assignment
drwxr-xr-x   - cloudera cloudera          0 2020-12-04 06:24 outputass
drwxr-xr-x   - cloudera cloudera          0 2020-10-31 10:03 outputs1
-rw-r--r--   1 cloudera cloudera     485246 2020-11-01 22:23 test_access_log
-rw-r--r--   1 cloudera cloudera         59 2020-11-08 22:02 wordcount
-rw-r--r--   1 cloudera cloudera         59 2020-11-08 22:11 wordcount.txt
[cloudera@quickstart Desktop]$ hdfs dfs -ls Task3_output
Found 2 items
-rw-r--r--   1 cloudera cloudera          0 2021-05-28 01:47 Task3_output/_SUCCESS
-rw-r--r--   1 cloudera cloudera       2195 2021-05-28 01:47 Task3_output/part-00000
```

Fig3: Generated Output Directory.

```
[cloudera@quickstart Desktop]$ hdfs dfs -cat Task3_output/part-00000
1       44
10specifies     21
11      22
23      9
4       16
a1      55
agarwal 45
aggarwal        45
agrawal 45
akshay  29
along   28
also    29
anything        30
appearing       29
arrowdropupsave 4
arrowdropupsave1        1
```

Fig4: Frequency of each word

```
filename        4
filter  1
find    38
found   40
free    60
geek    29
geekfiletxt     130
geekfiletxt12   24
geekfiletxt4    11
geekfiletxtoutput       33
geeksforgeeks   58
geeksforgeeksorg        29
generate        1
give    6
given   70
globally        1
great   60
grep    389
h       2
help    29
idegeeksforgeeksorg     1
ignores 2
incorrect       30
information     30
input   5
insensitive     6
insensitively   6
inverting       17
l       22
learn   78
learnunix       124
```

SnapShots of Visualizing the data using Kibana.

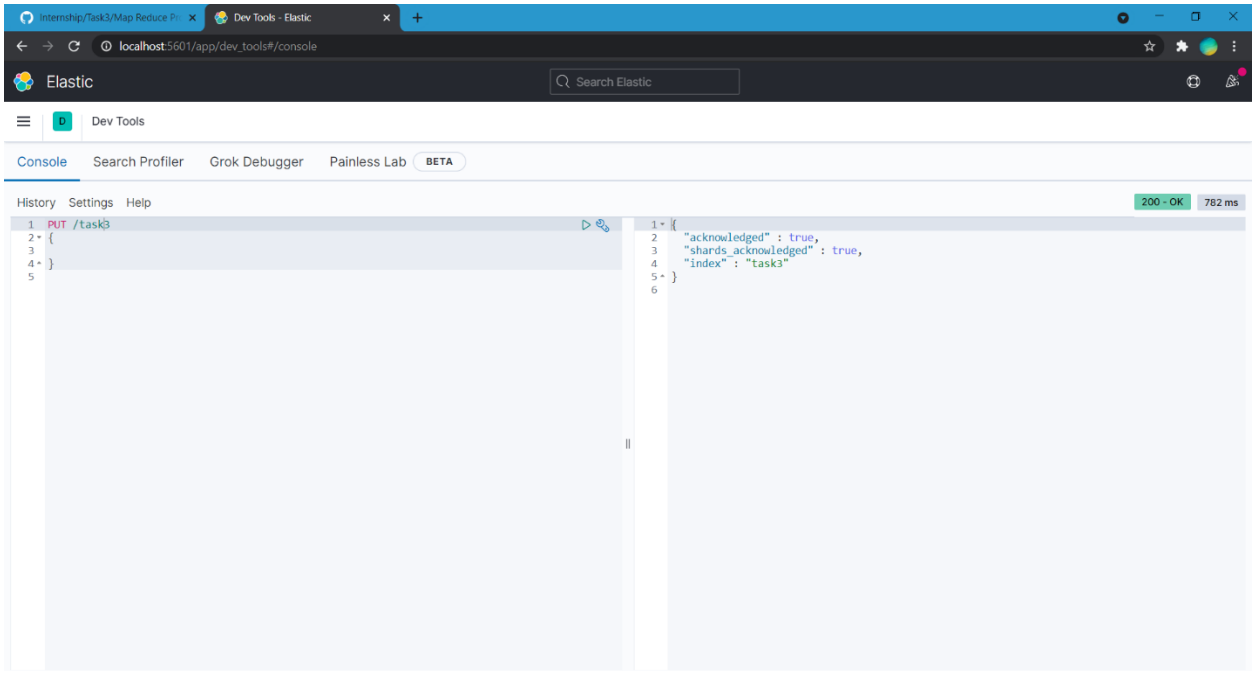
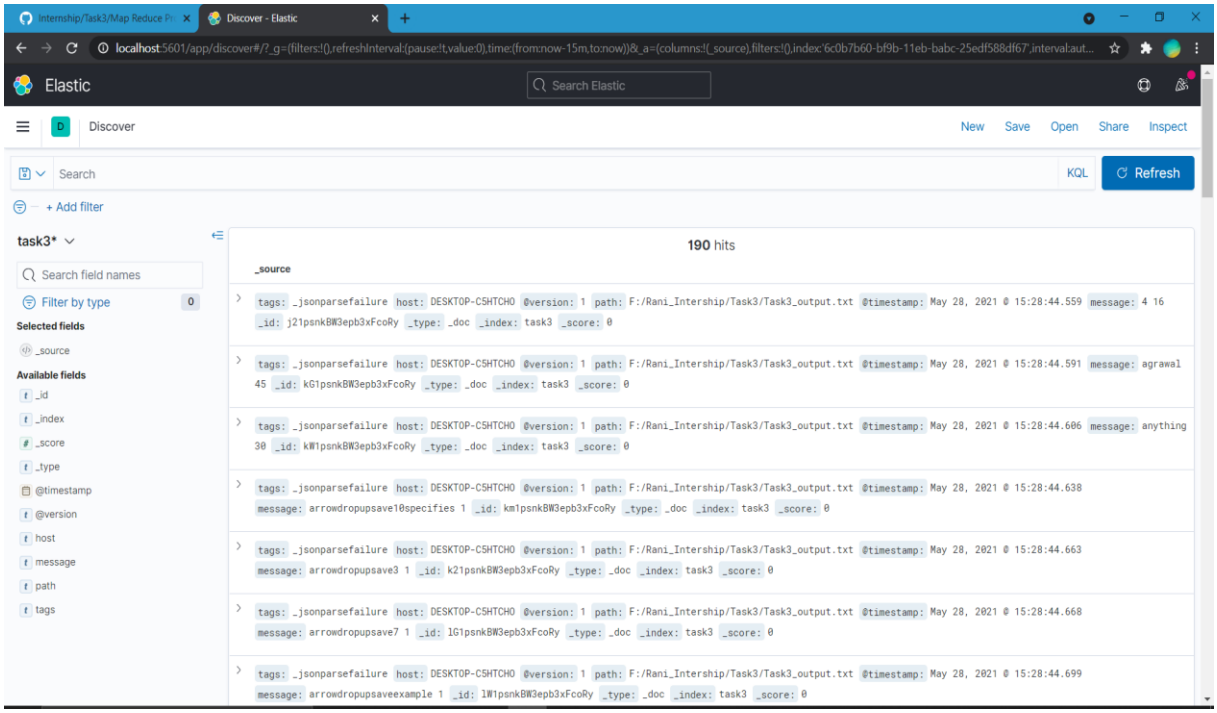


Fig 5: Creating index for storing the data.



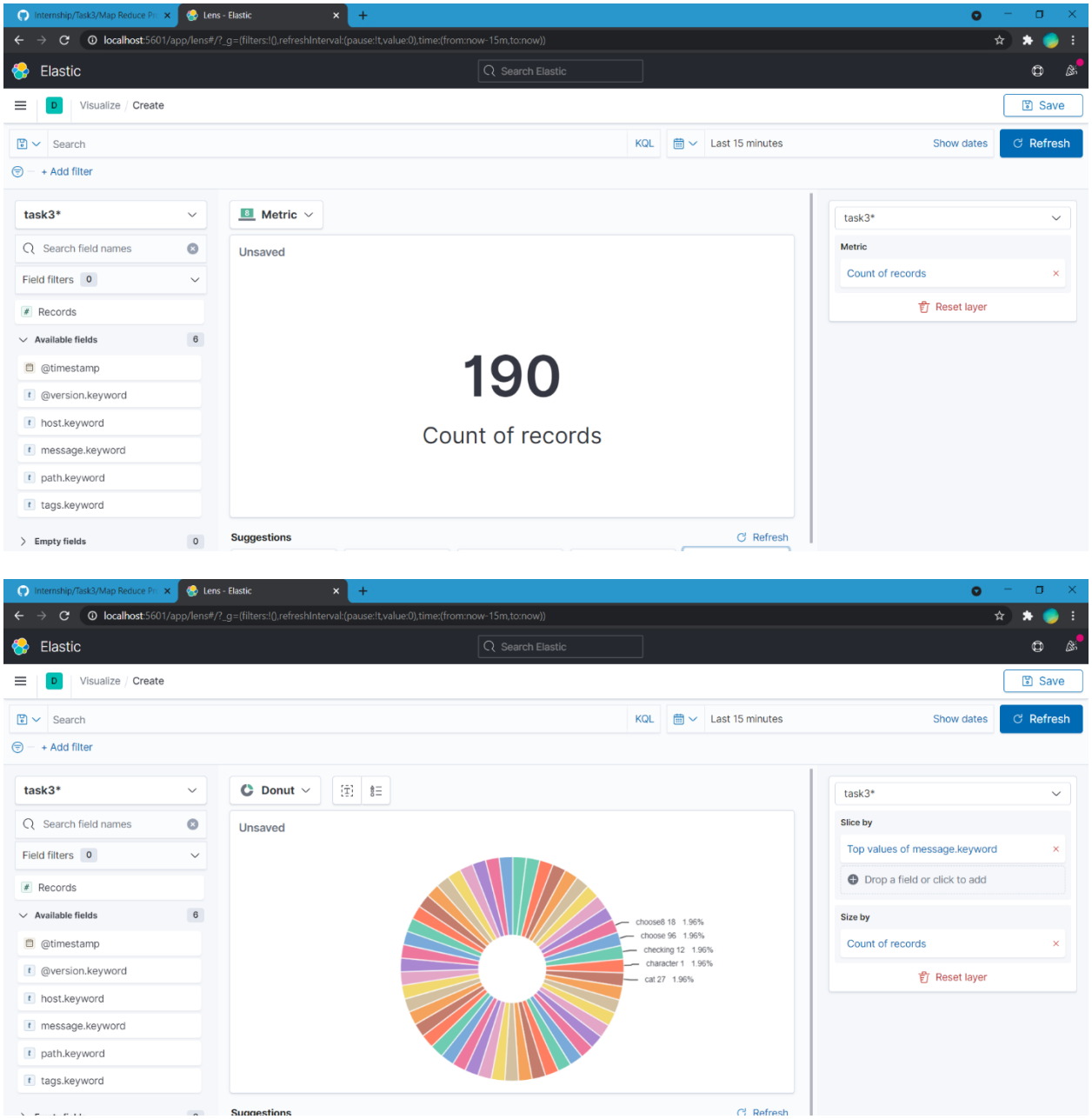


Fig: Represents the total count of word.