INFSCI 2750 Cloud Computing
Mini Project 1

Jing Pang jip45@pitt.edu
Haoyang Qian haq13@pitt.edu
Tian Xue tix20@pitt.edu

Part 1: Hadoop Setup

We built the Hadoop cluster following the sequence of preparation steps to install and run Hadoop in our three servers. In this case, we setup our master and slaves in following orders.
export MASTER_IP=159.65.253.68
export SLAVE1_IP=68.183.59.111
export SLAVE2_IP=68.183.154.239
The following graphs show that the Hadoop cluster is successfully starting.

```
student@master:~$ jps
23445 Jps
3142 JobHistoryServer
11690 SecondaryNameNode
11228 NameNode
11917 ResourceManager
```

```
student@slave-1:~$ jps
16916 DataNode
17099 NodeManager
21230 Jps
```

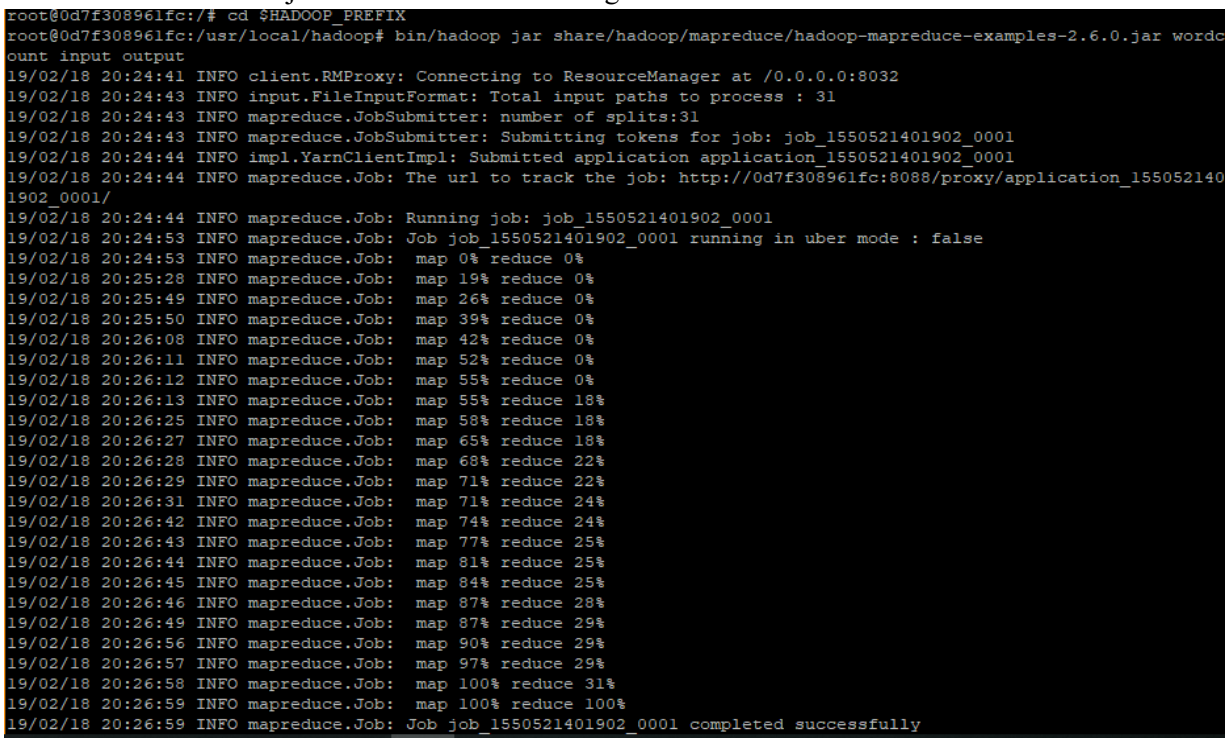
```
student@slave-2:~$ jps
18278 DataNode
21783 Jps
18461 NodeManager
```

We test our Hadoop cluster with simple example presented by Hadoop default wordcount program.

```
student@master:~/hadoop$ bin/hdfs dfs -cat output/wordcount/*
#       1
localhost       1
master  2
slave-1 1
slave-2 1
```

Part 2: Hadoop Docker Image

In this part, we can build our docker images based on the previous part which can quickly deploy hadoop as we did in the part 1. The support files for the docker image are included in the folder “docker”. We also tested a wordcount job on the built docker image. The result is shown as below.



```
root@0d7f30896lfc:/# cd $HADOOP_PREFIX
root@0d7f30896lfc:/usr/local/hadoop# bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.6.0.jar wordc
ount input output
19/02/18 20:24:41 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
19/02/18 20:24:43 INFO input.FileInputFormat: Total input paths to process : 31
19/02/18 20:24:43 INFO mapreduce.JobSubmitter: number of splits:31
19/02/18 20:24:43 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1550521401902_0001
19/02/18 20:24:44 INFO impl.YarnClientImpl: Submitted application application_1550521401902_0001
19/02/18 20:24:44 INFO mapreduce.Job: The url to track the job: http://0d7f30896lfc:8088/proxy/application_155052140
1902_0001/
19/02/18 20:24:44 INFO mapreduce.Job: Running job: job_1550521401902_0001
19/02/18 20:24:53 INFO mapreduce.Job: Job job_1550521401902_0001 running in uber mode : false
19/02/18 20:24:53 INFO mapreduce.Job:  map 0% reduce 0%
19/02/18 20:25:28 INFO mapreduce.Job:  map 19% reduce 0%
19/02/18 20:25:49 INFO mapreduce.Job:  map 26% reduce 0%
19/02/18 20:25:50 INFO mapreduce.Job:  map 39% reduce 0%
19/02/18 20:26:08 INFO mapreduce.Job:  map 42% reduce 0%
19/02/18 20:26:11 INFO mapreduce.Job:  map 52% reduce 0%
19/02/18 20:26:12 INFO mapreduce.Job:  map 55% reduce 0%
19/02/18 20:26:13 INFO mapreduce.Job:  map 55% reduce 18%
19/02/18 20:26:25 INFO mapreduce.Job:  map 58% reduce 18%
19/02/18 20:26:27 INFO mapreduce.Job:  map 65% reduce 18%
19/02/18 20:26:28 INFO mapreduce.Job:  map 68% reduce 22%
19/02/18 20:26:29 INFO mapreduce.Job:  map 71% reduce 22%
19/02/18 20:26:31 INFO mapreduce.Job:  map 71% reduce 24%
19/02/18 20:26:42 INFO mapreduce.Job:  map 74% reduce 24%
19/02/18 20:26:43 INFO mapreduce.Job:  map 77% reduce 25%
19/02/18 20:26:44 INFO mapreduce.Job:  map 81% reduce 25%
19/02/18 20:26:45 INFO mapreduce.Job:  map 84% reduce 25%
19/02/18 20:26:46 INFO mapreduce.Job:  map 87% reduce 28%
19/02/18 20:26:49 INFO mapreduce.Job:  map 87% reduce 29%
19/02/18 20:26:56 INFO mapreduce.Job:  map 90% reduce 29%
19/02/18 20:26:57 INFO mapreduce.Job:  map 97% reduce 29%
19/02/18 20:26:58 INFO mapreduce.Job:  map 100% reduce 31%
19/02/18 20:26:59 INFO mapreduce.Job:  map 100% reduce 100%
19/02/18 20:26:59 INFO mapreduce.Job: Job job_1550521401902_0001 completed successfully
```

```
19/02/18 20:26:59 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=72789
                FILE: Number of bytes written=3550268
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=80588
                HDFS: Number of bytes written=37448
                HDFS: Number of read operations=96
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=31
                Launched reduce tasks=1
                Data-local map tasks=31
                Total time spent by all maps in occupied slots (ms)=611982
                Total time spent by all reduces in occupied slots (ms)=67724
                Total time spent by all map tasks (ms)=611982
                Total time spent by all reduce tasks (ms)=67724
                Total vcore-seconds taken by all map tasks=611982
                Total vcore-seconds taken by all reduce tasks=67724
                Total megabyte-seconds taken by all map tasks=626669568
                Total megabyte-seconds taken by all reduce tasks=69349376
        Map-Reduce Framework
                Map input records=2065
                Map output records=7719
                Map output bytes=103873
                Map output materialized bytes=72969
                Input split bytes=3812
                Combine input records=7719
                Combine output records=3801
                Reduce input groups=1616
                Reduce shuffle bytes=72969
                Reduce input records=3801
                Reduce output records=1616
                Spilled Records=7602
                Shuffled Maps =31
                Failed Shuffles=0
                Merged Map outputs=31
                GC time elapsed (ms)=3694
                CPU time spent (ms)=18100
                Physical memory (bytes) snapshot=7534071808
                Virtual memory (bytes) snapshot=23520075776
                Total committed heap usage (bytes)=5725224960
        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=76776
        File Output Format Counters
                Bytes Written=37448
root@0d7f30896lfc:/usr/local/hadoop# bin/hdfs dfs -cat output/*
!=      3
""      6
".      4
"$HADOOP_CLASSPATH"     1
```

```
        File Input Format Counters
                Bytes Read=76776
        File Output Format Counters
                Bytes Written=37448
root@0d7f30896lfc:/usr/local/hadoop# bin/hdfs dfs -cat output/*
!=      3
""      6
".      4
"$HADOOP_CLASSPATH"     1
"$JAVA_HOME"    2
"$YARN_HEAPSIZE"        1
"$YARN_LOGFILE" 1
"$YARN_LOG_DIR" 1
"$YARN_POLICYFILE"      1
"*"     18
"AS     21
"Error: 1
"License");     21
"alice,bob      18
"console"       1
"dfs"   3
"hadoop.root.logger".   1
"jks".  4
"jvm"   3
"mapred"        3
"rpc"   3
"run    1
"ugi"   3
"x"     1
"x$JAVA_LIBRARY_PATH"   1
#       380
#!/bin/bash     2
###     4
#*.sink.ganglia.dmax=jvm.metrics.threadsBlocked=70,jvm.metrics.memHeapUsedM=40  1
#*.sink.ganglia.slope=jvm.metrics.gcCount=zero,jvm.metrics.memHeapUsedM=both    1
#*.sink.ganglia.tagsForPrefix.dfs=       1
#*.sink.ganglia.tagsForPrefix.jvm=ProcesName     1
#*.sink.ganglia.tagsForPrefix.mapred=    1
#*.sink.ganglia.tagsForPrefix.rpc=       1
#A       1
#Default         1
#HADOOP_JAVA_PLATFORM_OPTS="-XX:-UsePerfData     1
#Security        1
#The     1
#datanode.sink.file.filename=datanode-metrics.out        1
#datanode.sink.ganglia.servers=yourgangliahost_1:8649,yourgangliahost_2:8649     1
#dfs.class=org.apache.hadoop.metrics.file.FileContext    1
#dfs.fileName=/tmp/dfsmetrics.log        1
#dfs.period=10   1
#echo    1
#export 15
#jobhistoryserver.sink.file.filename=jobhistoryserver-metrics.out        1
#jobhistoryserver.sink.ganglia.servers=yourgangliahost_1:8649,yourgangliahost_2:8649     1
#jvm.class=org.apache.hadoop.metrics.file.FileContext    1
#jvm.class=org.apache.hadoop.metrics.spi.NullContext     1
#jvm.fileName=/tmp/jvmmetrics.log        1
#jvm.period=10   1
#log4j.additivity.org.apache.hadoop.mapreduce.v2.hs.HSAuditLogger=false 1
#log4j.appender.DRFA.MaxBackupIndex=30   1
#log4j.appender.DRFA.layout.ConversionPattern=%d{ISO8601}        1
#log4j.appender.HSAUDIT.DatePattern=.yyyy-MM-dd 1
#log4j.appender.HSAUDIT.File=${hadoop.log.dir}/hs-audit.log      1
#log4j.appender.HSAUDIT.layout.ConversionPattern=%d{ISO8601}     1
```

Part 3: Developing Hadoop program (n-gram)

We implemented a mapreduce program to realize the function of n-gram, which is to calculate the n-gram frequencies of the input text file. To run this program, we need to upload a text file to the service, and then input an “n” as a parameter (here we tried 3). Finally, we got part of the output like below:

```
student@master:~/hadoop$ bin/hdfs dfs -tail output/ngram/part-r-00000
zi.     1
zi/     4
ziY     1
zic     11
zie     17
zig     5
zik     1
zim     3
zin     36
zio     11
zir     2
zis     3
zit     1
zjk     1
```

Part 4: Analyzing real logs using Hadoop program

1. How many hits were made to the website item “/assets/img/home-logo.png”?

Answer: 98776

```
student@master:~/hadoop$ bin/hdfs dfs -cat output/resQ1/*
        98776
```

2. How many hits were made from the IP: 10.153.239.5

Answer: 547

```
student@master:~/hadoop$ bin/hdfs dfs -cat output/resQ2/*
        547
```

3. Which path in the website has been hit most? How many hits were made to the path?

Answer:

Most hit: “/assets/css/combined.css”

Hits: 117348

```
student@master:~/hadoop$ bin/hdfs dfs -cat output/resQ3/top1/*
/assets/css/combined.css        117348
```

4. Which IP access the website most? How many accesses were made by it?

Most IP access the website: 10.216.113.172

# of accesses were made: 158614

```
student@master:~/hadoop$ bin/hdfs dfs -cat output/resQ4/top1/*
10.216.113.172  158614
student@master:~/hadoop$
```