FACULTY OF ENGINEERING

Lab 2: Parallel K-Means using Hadoop

Distributed Systems

Presented by

Names	IDs
Aaser Fawzy Zakaria Hassan	**19015403**
Gamal Abdel Hamid Nasef Nowesar	**19015550**
Mohamed Ezzat Saad El-Shazly	**19016441**

9/3/2024

Table of Contents

1 Problem Definition

It is required to understand spark frame work and to be able to build tasks using it, in this lab it is required to install spark on a single node cluster and build the MapReduce wordcount and run it on an input file in the HDFS.

2 Algorithms

The Parallel word count using the MapReduce framework requires to implement 2 methods the mapping method and the reduce method which were already implemented in the provided code.

For the mapping method:

```
mapToPair algorithm (int key, String value){
      Emit (value, 1);
}
```

For the reduction method

```
reduceByKey algorithm (String key, int counts[]){
      sum = 0
      for count in counts
            sum += count
      Emit(key, sum)
}
```

1. The code was built using `mvn clean install package`

2. We then copied the input from the local to the HDFS

```
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/WordCount/Inputs/Gutenbergprojectfiles-20
240404T092813Z-001/Gutenbergprojectfiles$ hdfs dfs -copyFromLocal 1.txt /user/inputs
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/WordCount/Inputs/Gutenbergprojectfiles-20
240404T092813Z-001/Gutenbergprojectfiles$ hdfs dfs -copyFromLocal 2.txt /user/inputs
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/WordCount/Inputs/Gutenbergprojectfiles-20
240404T092813Z-001/Gutenbergprojectfiles$ hdfs dfs -copyFromLocal 3.txt /user/inputs
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/WordCount/Inputs/Gutenbergprojectfiles-20
240404T092813Z-001/Gutenbergprojectfiles$ hdfs dfs -ls /user/inputs
Found 4 items
-rw-r--r--   1 mohamed supergroup     151016 2024-04-04 11:38 /user/inputs/1.txt
-rw-r--r--   1 mohamed supergroup      59210 2024-04-04 11:38 /user/inputs/2.txt
-rw-r--r--   1 mohamed supergroup      39380 2024-04-04 11:38 /user/inputs/3.txt
-rw-r--r--   1 mohamed supergroup       3561 2024-04-03 22:57 /user/inputs/input.txt
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/WordCount/Inputs/Gutenbergprojectfiles-20
240404T092813Z-001/Gutenbergprojectfiles$
```

3. Then the jar was run on the 4 text files

```
mohamed@mohamed-VirtualBox: ~/Desktop/Distributed Lab 3/WordCount/target
File  Edit  View  Search  Terminal  Help
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/WordCount/target$ java -j
ar wordcount-1.0.jar hdfs://localhost:9000/user/inputs/1.txt hdfs://localhost:90
00/user/outputs/1
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
24/04/04 11:50:39 INFO SparkContext: Running Spark version 1.4.0
24/04/04 11:50:40 WARN NativeCodeLoader: Unable to load native-hadoop library fo
r your platform... using builtin-java classes where applicable
24/04/04 11:50:40 WARN Utils: Your hostname, mohamed-VirtualBox resolves to a lo
opback address: 127.0.1.1; using 10.0.2.15 instead (on interface enp0s3)
24/04/04 11:50:40 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another
```

```
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/WordCount/target$ java -j
ar wordcount-1.0.jar hdfs://localhost:9000/user/inputs/2.txt hdfs://localhost:90
00/user/outputs/2
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
24/04/04 11:51:41 INFO SparkContext: Running Spark version 1.4.0
24/04/04 11:51:41 WARN NativeCodeLoader: Unable to load native-hadoop library fo
r your platform... using builtin-java classes where applicable
24/04/04 11:51:41 WARN Utils: Your hostname, mohamed-VirtualBox resolves to a lo
opback address: 127.0.1.1; using 10.0.2.15 instead (on interface enp0s3)
24/04/04 11:51:41 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another
```

```
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/WordCount/target$ java -j
ar wordcount-1.0.jar hdfs://localhost:9000/user/inputs/3.txt hdfs://localhost:90
00/user/outputs/3
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
24/04/04 11:52:10 INFO SparkContext: Running Spark version 1.4.0
24/04/04 11:52:10 WARN NativeCodeLoader: Unable to load native-hadoop library fo
r your platform... using builtin-java classes where applicable
24/04/04 11:52:10 WARN Utils: Your hostname, mohamed-VirtualBox resolves to a lo
opback address: 127.0.1.1; using 10.0.2.15 instead (on interface enp0s3)
24/04/04 11:52:10 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another
```

```
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/WordCount/target$ java -jar wordcount-1.0.jar hdfs://localhost
:9000/user/inputs/input.txt hdfs://localhost:9000/user/outputs/asg3
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
24/04/03 23:05:02 INFO SparkContext: Running Spark version 1.4.0
24/04/03 23:05:03 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java
 classes where applicable
24/04/03 23:05:03 WARN Utils: Your hostname, mohamed-VirtualBox resolves to a loopback address: 127.0.1.1; using 10.0
.2.15 instead (on interface enp0s3)
24/04/03 23:05:03 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
24/04/03 23:05:03 INFO SecurityManager: Changing view acls to: mohamed
24/04/03 23:05:03 INFO SecurityManager: Changing modify acls to: mohamed
24/04/03 23:05:03 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users with view p
ermissions: Set(mohamed); users with modify permissions: Set(mohamed)
24/04/03 23:05:03 INFO Slf4jLogger: Slf4jLogger started
24/04/03 23:05:03 INFO Remoting: Starting remoting
24/04/03 23:05:03 INFO Remoting: Remoting started; listening on addresses :[akka.tcp://sparkDriver@10.0.2.15:43657]
24/04/03 23:05:03 INFO Utils: Successfully started service 'sparkDriver' on port 43657.
24/04/03 23:05:03 INFO SparkEnv: Registering MapOutputTracker
24/04/03 23:05:03 INFO SparkEnv: Registering BlockManagerMaster
24/04/03 23:05:03 INFO DiskBlockManager: Created local directory at /tmp/spark-bbf28471-cef8-494c-b74b-8c5b5b282b06/b
```

1. Finally, we retrieved the outputs

```
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3$ hdfs dfs -copyToLocal /u
ser/outputs
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3$
```

Some of the challenges we faced during our implementation:

- Building the jar on Cloudera
    - The spark, maven and Scala on the virtual machine were outdated and could not install dependencies necessary to build the jar.
- Solution
    - We setup Hadoop on another Ubuntu VM and installed spark using the lab's instructions and the code was built normally

4 Implementation

- Environment:
  - Local machine setup using a personal PC in a virtualized environment using Vbox
  - Machine specification (virtual machine)
    - Operating System: Ubuntu
    - CPU: 4 cores @ 2.5 GHz
    - Memory: 8 GB RAM
    - Storage: 50 GB Disk space HDD

- Test Dataset (test cases)
  - Gutenberg project files provided in the first lab were used.
  - A dummy input file also was used for testing.

5 Results

The results obtained for each of the Gutenberg files

```
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/WordCount/target$ hdfs df
s -cat /user/outputs/2/part-00000
(Blanche,1)
(Let,1)
(Hyne,1)
(Hicks,1)
(67043,1)
(Diary,1)
(end,1)
(Script,,1)
(67013,1)
(unkarilainen,1)
(Joseph,4)
(92d,1)
(66828,1)
(Lampérth,1)
(Bourget,1)
(Guglielmo,1)
(1638,,1)
(Vedette,,1)
(Nathan,2)
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/WordCount/target$ hdfs df
s -cat /user/outputs/1/part-00000
(69090,1)
(68393,1)
(House,2)
(Silliman,1)
(park,,1)
(68494,1)
(making,,2)
(Lancers,1)
(68813,1)
(68465,1)
(end,4)
(been,1)
(68009,1)
(pastry,1)
(water-supplies,,1)
(Judd,2)
(apiculture,,1)
```

```
mohamed@mohamed-VirtualBox:~/Desktop/Distributed Lab 3/outputs$ hdfs dfs -cat /u
ser/outputs/3/part-00000
(being,2)
(Neighbors,,1)
(Geographie,1)
(63893,1)
(Elizabethan,1)
(House,1)
(Jeremiah,1)
(5,,1)
(Miguel,1)
(Rider,1)
(Board,1)
(Conrad,1)
(Saavedra,1)
```

6 Conclusion

- In this lab we were able to learn more about the map reduce framework for Hadoop.
- We were able to interact directly more with the HDFS.
- We were able to build spark applications and use it in Hadoop clusters.