**PRINCIPLES OF BIG DATA MANAGEMENT**

**PHASE-1**

**Objective:**

• The main aim of this phase is to develop a system to store, analyze, and visualize a social network data.

• Tasks:

1. Collect social network data (e.g. tweets) in JSON format.

2. Extracting the hashtags, URLs from the collected tweets.

3. Store the text content (e.g. tweet’s text) from the data into a file in HDFS.

4. Run a Word Count program in Apache Spark, Apache Hadoop on extracted hashtags, URLs files and

store the output and log files locally.

**Applications/Software’s Used:** Apache Spark, Hadoop, Scala, Cloudera, Twitter Developer Account, Python.

**Tweets collection from Twitter:**

• Firstly, we have created a developer account in Twitter using link: https://apps.twitter.com/

• Below are the variables that contains the user credentials to access Twitter API

➢ **API ACCESS\_TOKEN** = "1089997729219178496-BvhcwdDw6eGPbPJM5wJYqOvsq7rE7G"

➢ **ACCESS\_SECRET** = “xZu9ILsesdPiRi5quCxx94mQTfrj00sMEedGrddvYOmsQ"

➢ **CONSUMER\_KEY** = "f4qQNYSFaGxO4iB6SamtIZarf"

➢ **CONSUMER\_SECRET** = "PTuSFEtZAeRP9nov16hJTWQzFNedKTSDKR9ZSS1Iu65JiSbXvO"

• We have written python program that is used to fetch tweets in JSON format. (Tweetscollection.py)

• The extracted file in JSON format contains all the tweet details such as id, created at, text, URL, hashtags etc.

• From JSON tweets file only the hashtags and URLs content is extracted using python program for hashtags and URLs. Then fetched details are stored in files.

**Run the Word Count program in Apache Spark and Apache Hadoop the extracted hashtags/URLs and collect the output and log files:**

**Word Count Implementation In spark:**

**Step 1**:

On the Hadoop terminal, First I will see my input file is where it is from terminal



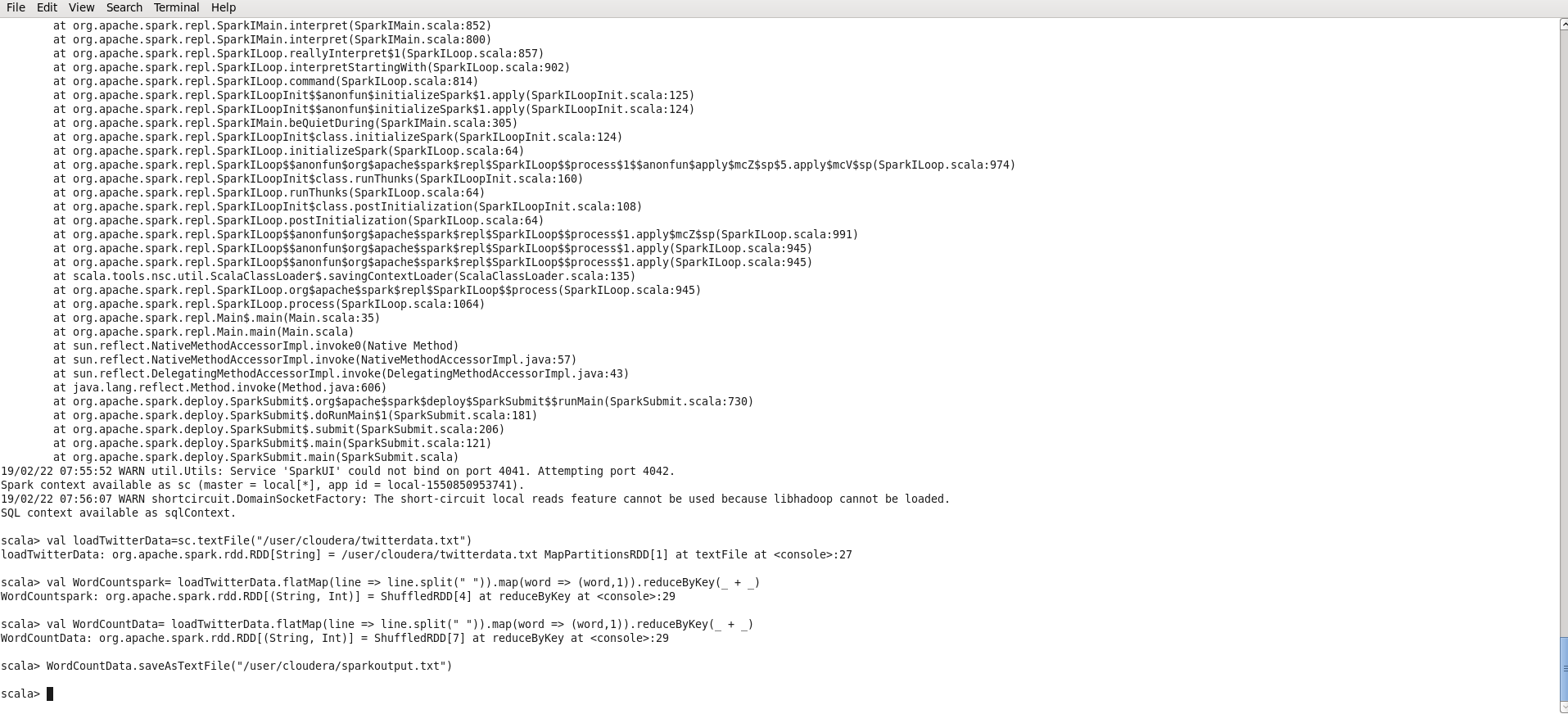
**Step 2:**

As you can see in the step1 screenshot from the ls command out input file is twitterdata.txt. Then, I put

the file in Hadoop file system and change the location according to my convenience. Now, I have open

spark-shell will implement word count program.



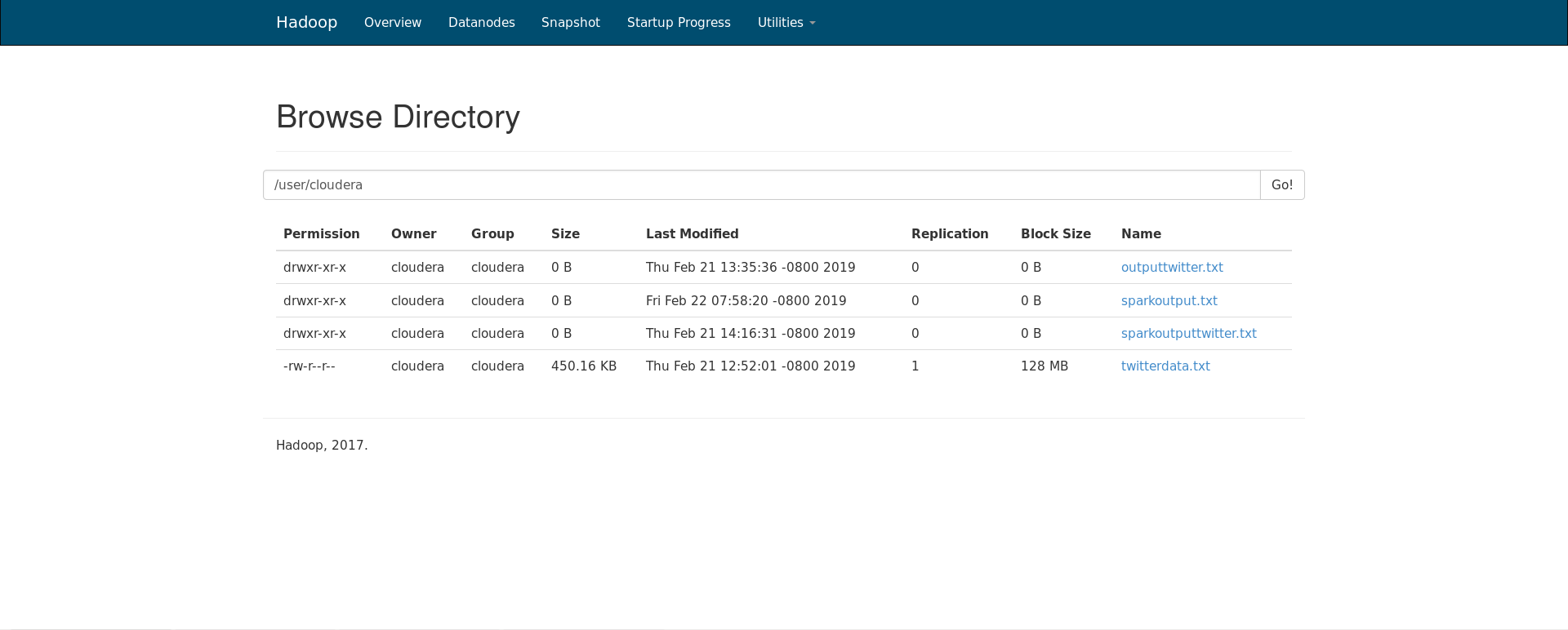


I have load the file data into loadTwitterData variable using sc.textFile function.

Then , I have use flatMap to split my data in the file using space as a delimiter and then use reduceByKey function to do my word count.

Then I have saved the Count in a text file sparkoutput.txt

Step 3:



We have opened the directory of Hadoop where my output file is save. So you can see my

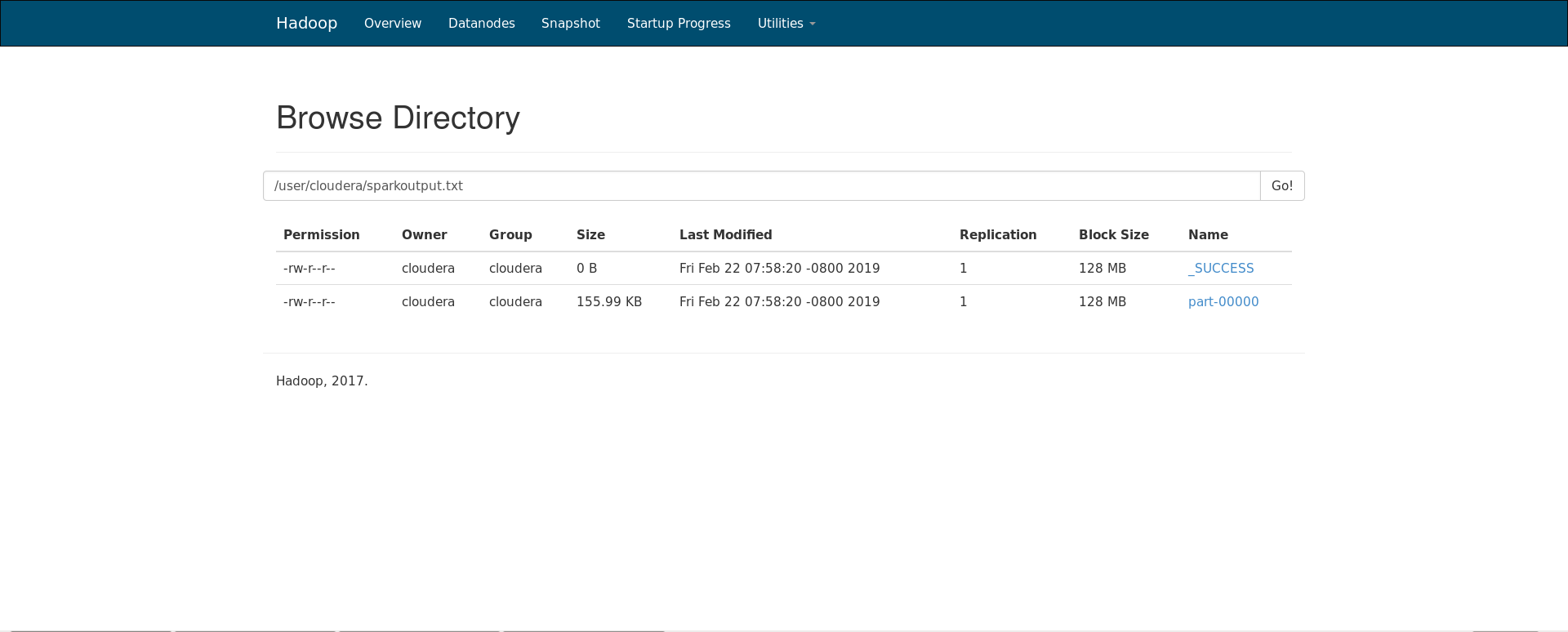
output file name is sparkoutput.txt. It is present there.

**Step 5:**

Further, opening that file you can view two files one name is part-00000 and other is success file.

Part-00000 file contains my output and success file is empty file which just tell me it has been made

successfully.



**Word Count Implementation in Apache Hadoop:**

In order to run the word count program using Apache Hadoop we have written the code in java. In the program TokenizerMapper class is created by extending the Mapper class and the map function is implemented by overriding the map method in the Mapper class. The mapper function takes a key value pair as an input and outputs a key-values pair as an output.

And IntSumReducer class is created by extending the org.apache.hadoop.mapreduce.Reducer class and the reduce method is implemented by overriding the reduce method from the Reducer class. The reduce function collects all the intermediate key-value pairs generated by the multiple map functions and will sum up all the occurrences of each word and output a key-value pair for each word in the text document.

And also the main method is used to setup all necessary configurations and runs the mapreduce job.

Then the .jar file is exported to cloudera to execute the word count for Hashtags and URLs using Apache Hadoop.

**Below are the steps followed:**

1. A folder is created in HDFS and the input file is moved from local to HDFS using below commands.

➢ Hdfs dfs -mkdir /WordCountTutorial

* Hdfs dfs -mkdir /WordCountTutorial/Input

2. Upload the input file into the directory using the command.

* Hdfs dfs -put “/home/cloudera/WordCountTutorial/input\_data/twitterhashtagsurlsdata.txt” /WordcountTutorial/Input

3. Created a sample class named Tutorial\_classes in the local directory.

4.Changing the current directory to local directory.

5.We have written a Mapreduce code in java for wordcount and compiled using the command.

* Javac -classpath ${HADOOP\_CLASSPATH} -d /home/cloudera/WordCountTutorial/Tutorial\_classes

6.Put the output files in one jar file

* Jar -cvf wc.jar -c Tutorial\_classes

7.Now, run the jar file in Hadoop using below command

* Hadoop jar wc.jar WordCount /WordCountTutorial/Input /WordCountTutorial/Out

8. Now , checkout the output using command :

* Hadoop dfs -cat /WordCountTutorial/Out

**Results:**

