Ex No.: 4 **Create UDF in PIG**

Step-by-step installation of Apache Pig on Hadoop cluster on Ubuntu Pre-requisite:

· Ubuntu 16.04 or higher version running (I have installed Ubuntu on Oracle VM (Virtual Machine) VirtualBox),

· Run Hadoop on ubuntu (I have installed Hadoop 3.2.1 on Ubuntu 16.04). You may refer to my blog "How to install Hadoop installation" click here for Hadoop installation).

Pig installation steps

Step 1: Login into Ubuntu

Step 2: Go to https://pig.apache.org/releases.html and copy the path of the latest version of pig that you want to install. Run the following comment to download Apache Pig in Ubuntu:

$ wget https://dlcdn.apache.org/pig/pig-0.16.0/pig-0.16.0.tar.gz

Step 3: To untar pig-0.16.0.tar.gz file run the following command:

$ tar xvzf pig-0.16.0.tar.gz

Step 4: To create a pig folder and move pig-0.16.0 to the pig folder, execute the following command:

$ sudo mv /home/hadoop/pig-0.16.0 /home/hadoop/pig

Step 5: Now open the .bashrc file to edit the path and variables/settings for pig. Run the following command:

$ sudo nano .bashrc

Add the below given to .bashrc file at the end and save the file.

#PIG settingsexport PIG_HOME=/home/hdoop/pigexport PATH=$PATH:$PIG_HOME/binexport PIG_CLASSPATH=$PIG_HOME/conf:$HADOOP_INSTALL/etc/hadoop/export PIG_CONF_DIR=$PIG_HOME/confexport JAVA_HOME=/usr/lib/jvm/java-8-openjdkamd64export PIG_CLASSPATH=$PIG_CONF_DIR:$PATH#PIG setting ends

```
# PIG settings
export PIG_HOME=/usr/local/pig
export PATH=$PATH:$PIG_HOME/bin
export PIG_CLASSPATH=$PIG_HOME/conf:$HADOOP_INSTALL/etc/hadoop/
export PIG_CONF_DIR=$PIG_HOME/conf
export PIG_CLASSPATH=$PIG_CONF_DIR:$PATH

export PYTHONPATH=/home/subhikshaa/pig:$PYTHONPATH

export PATH=$PATH:/usr/bin/jython
export PIG_CLASSPATH=$PIG_CLASSPATH:/usr/bin/jython
```

Step 6: Run the following command to make the changes effective in the .bashrc file:

$ source .bashrc

Step 7: To start all Hadoop daemons, navigate to the hadoop-3.2.1/sbin folder and run the following commands:

$ ./start-dfs.sh$ ./start-yarn$ jps

```
varunesh@varunesh:~$ nano ~/.bashrc
varunesh@varunesh:~$ source ~/.bashrc
varunesh@varunesh:~$ jps
16592 SecondaryNameNode
18675 Jps
16869 ResourceManager
16377 DataNode
17018 NodeManager
```

Step 8: Now you can launch pig by executing the following command: $ pig

```
varunesh@varunesh:~$ pig
2024-09-28 22:46:36,618 INFO pig.ExecTypeProvider: Trying ExecType : LOC
AL
2024-09-28 22:46:36,619 INFO pig.ExecTypeProvider: Trying ExecType : MAP
REDUCE
2024-09-28 22:46:36,619 INFO pig.ExecTypeProvider: Picked MAPREDUCE as t
he ExecType
2024-09-28 22:46:36,656 [main] INFO  org.apache.pig.Main - Apache Pig ve
rsion 0.17.0 (r1797386) compiled Jun 02 2017, 15:41:58
2024-09-28 22:46:36,656 [main] INFO  org.apache.pig.Main - Logging error
 messages to: /home/subhikshaa/pig_1727543796651.log
2024-09-28 22:46:36,674 [main] INFO  org.apache.pig.impl.util.Utils - De
fault bootup file /home/subhikshaa/.pigbootup not found
2024-09-28 22:46:36,868 [main] INFO  org.apache.hadoop.conf.Configuratio
n.deprecation - mapred.job.tracker is deprecated. Instead, use mapreduce
.jobtracker.address
2024-09-28 22:46:36,868 [main] INFO  org.apache.pig.backend.hadoop.execu
tionengine.HExecutionEngine - Connecting to hadoop file system at: hdfs:
//localhost:9000
2024-09-28 22:46:37,240 [main] INFO  org.apache.pig.PigServer - Pig Scri
pt ID for the session: PIG-default-613b5bd0-08b5-46f8-a8a6-17cdd22e8920
2024-09-28 22:46:37,241 [main] WARN  org.apache.pig.PigServer - ATS is d
isabled since yarn.timeline-service.enabled set to false
grunt>
```

Step 9: Now you are in pig and can perform your desired tasks on pig. You can come out of the pig by the quit command:

> quit;

CREATE USER DEFINED FUNCTION(UDF)

Aim :

To create User Define Function in Apache Pig and execute it on map reduce.

PROCEDURE:

Create a sample text file

hadoop@Ubuntu:~/Documents$ nano sample.txt

Paste the below content to sample.txt

1,subhikshaa

2,srilekha

3,s

hadoop@Ubuntu:~/Documents$ hadoop fs -put sample.txt /home/hadoop/piginput/

Create PIG File

hadoop@Ubuntu:~/Documents$ nano demo_pig.pig

paste the below the content to demo_pig.pig

-- Load the data from HDFS

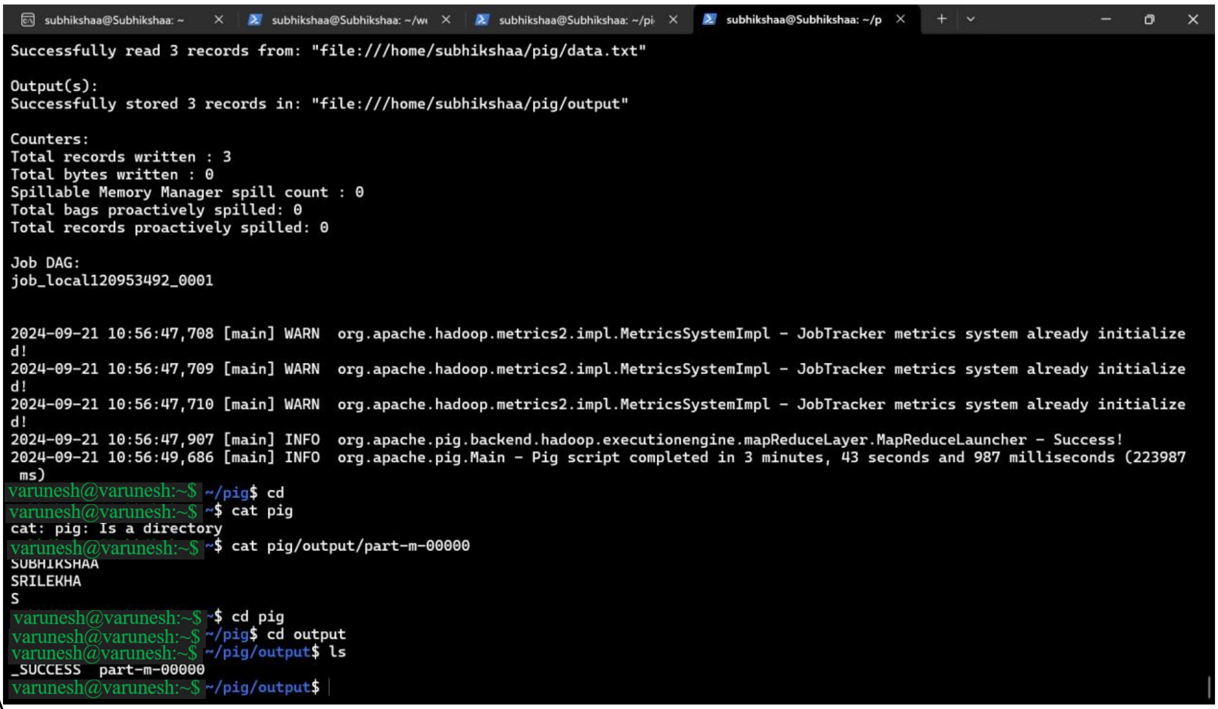
data = LOAD '/home/hadoop/piginput/sample.txt' USING PigStorage(',') AS (id:int>

-- Dump the data to check if it was loaded correctly

DUMP data;

-- **Run the above file**

hadoop@Ubuntu:~/Documents$ pig demo_pig.pig



```
Successfully read 3 records from: "file:///home/subhikshaa/pig/data.txt"

Output(s):
Successfully stored 3 records in: "file:///home/subhikshaa/pig/output"

Counters:
Total records written : 3
Total bytes written : 0
Spillable Memory Manager spill count : 0
Total bags proactively spilled: 0
Total records proactively spilled: 0

Job DAG:
job_local120953492_0001


2024-09-21 10:56:47,708 [main] WARN  org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialize
d!
2024-09-21 10:56:47,709 [main] WARN  org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialize
d!
2024-09-21 10:56:47,710 [main] WARN  org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialize
d!
2024-09-21 10:56:47,907 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
2024-09-21 10:56:49,686 [main] INFO  org.apache.pig.Main - Pig script completed in 3 minutes, 43 seconds and 987 milliseconds (223987
 ms)
varunesh@varunesh:~$ ~/pig$ cd
varunesh@varunesh:~$ ~$ cat pig
cat: pig: Is a directory
varunesh@varunesh:~$ ~$ cat pig/output/part-m-00000
SUBHIKSHAA
SRILEKHA
S
varunesh@varunesh:~$ ~$ cd pig
varunesh@varunesh:~$ ~/pig$ cd output
varunesh@varunesh:~$ ~/pig/output$ ls
_SUCCESS  part-m-00000
varunesh@varunesh:~$ ~/pig/output$
```

\

Create udf file an save as uppercase_udf.py

uppercase_udf.py

```
def uppercase(text): return text.upper()

if __name___== "__main__":

import sys for line in
sys.stdin:
        line = line.strip() result =
        uppercase(line)
        print(result)
```

Create the udfs folder on hadoop

hadoop@Ubuntu:~/Documents$ hadoop fs -mkdir /home/hadoop/udfs

put the upppercase_udf.py in to the abv folder

hadoop@Ubuntu:~/Documents$ hdfs dfs -put uppercase_udf.py /home/hadoop/udfs/

hadoop@Ubuntu:~/Documents$ nano udf_example.pig copy and paste the below content on udf_example.pig

```
-- Register the Python UDF script

REGISTER 'hdfs:///home/hadoop/udfs/uppercase_udf.py' USING jython AS udf;


-- Load some data

data = LOAD 'hdfs:///home/hadoop/sample.txt' AS (text:chararray);


-- Use the Python UDF

uppercased_data = FOREACH data GENERATE udf.uppercase(text) AS uppercase_text;


-- Store the result

STORE uppercased_data INTO 'hdfs:///home/hadoop/pig_output_data';
```

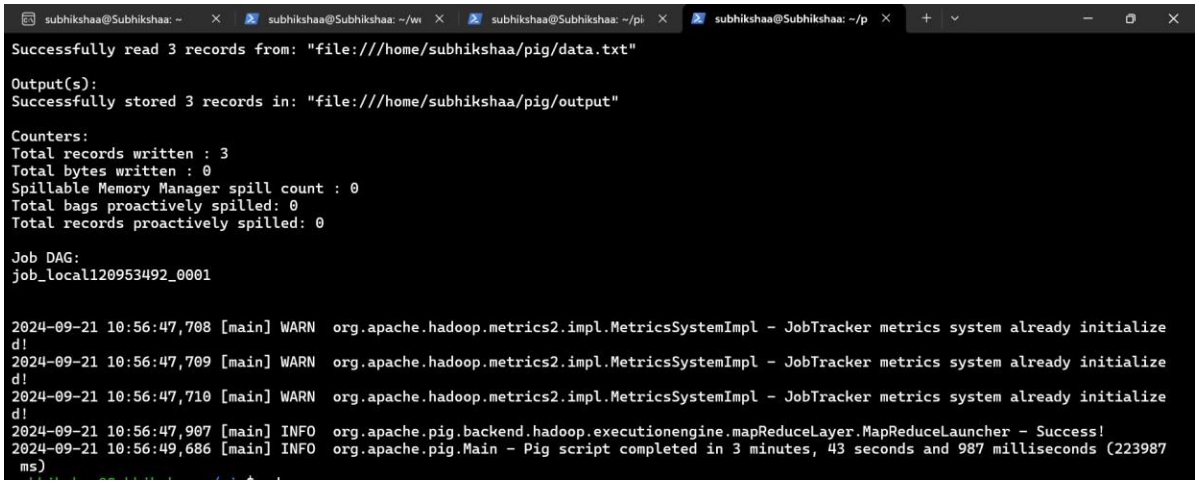
---

place sample.txt file on hadoop

hadoop@Ubuntu:~/Documents$ hadoop fs -put sample.txt /home/hadoop/

To Run the pig file

hadoop@Ubuntu:~/Documents$ pig -f udf_example.pig



To check the output file is created

hadoop@Ubuntu:~/Documents$ hdfs dfs -ls /home/hadoop/pig_output_data

Found 2 items

If you need to examine the files in the output folder, use:

To view the output

hadoop@Ubuntu:~/Documents$ hdfs dfs -cat /home/hadoop/pig_output_data/part-m00000

Result:

Thus the program to create User Define Function in Apache Pig and execute it on map reduce has been done successfully.