Immersion Day

Pig

September 2016

Rev 2016-01-13

Table of Contents

Overview....3

Writing Pig script interactively through SSH....3

Submit Pig Work Using the Amazon EMR Console....5

Overview

Apache Pig is a platform for analyzing large data sets that consists of a high-level language for expressing data analysis programs, coupled with infrastructure for evaluating these programs. The salient property of Pig programs is that their structure is amenable to substantial parallelization, which in turns enables them to handle very large data sets.

This section demonstrates submitting Pig work to an Amazon EMR cluster. The lab evaluates Apache log files. We'll first write some Pig analytics interactively through a shell and then run a script that generates a report containing the total bytes transferred, a list of the top 50 IP addresses, a list of the top 50 external referrers, and the top 50 search terms using Bing and Google. The output of the script will be saved to your own bucket (but the source script and files are stored in a shared location).

Writing Pig script interactively

1. SSH into the Master node for the interactive EMR cluster.
    - Log into the EMR Web console
    - Go into your BigDataImmersionLab cluster that is in the Waiting state
    - Allow SSH access from your current IP.
        i. Under the Security and Access section, click on the link in the Security groups for Master section.
        ii. Select the Security Group named ElasticMapReduce-master, then click the inbound tab at the bottom.
        iii. Click Edit, then click Add Rule. This adds an additional line at the end of the list of rules.
        iv. Select "SSH" in the first column, then "My IP" from the second drop down box. Then click Save.
    - Next to the Master public DNS entry towards the top, you will see an SSH link describing how to SSH into the master node.
    - More detail can be found here: https://docs.aws.amazon.com/ElasticMapReduce/latest/ManagementGuide/emr-connect-master-node-ssh.html
2. Enter in the following command to start a pig session:

```
pig
```

3. After the session starts, you'll see this:

```
grunt>
```

4. This will allow you to interactively write the pig code (called piglatin) in the session. There are also multiple editors (including Hue, which you'll see in the next lab) that allows you to write the pig analytics.

5. First, DEFINE a few elements in the script:

```
DEFINE CustomFormatToISO org.apache.pig.piggybank.evaluation.datetime.convert.CustomFormatToISO;
DEFINE ISOToHour org.apache.pig.piggybank.evaluation.datetime.truncate.ISOToHour;
```

6. Next, we'll load the raw logs into a variable in pig using the TextLoader. There are many types of loaders to be able to read from various datasources, including S3, HDFS, DynamoDB, etc.

```
raw_logs =
  LOAD 's3://elasticmapreduce/samples/pig-apache/input' USING TextLoader AS (line:chararray);
```

7. This next command will indicate that we want to iterate over each and save them in named variables under a varable named logs_base.

```
logs_base =
FOREACH
    raw_logs
  GENERATE
    FLATTEN (
      REGEX_EXTRACT_ALL(
        line,
        '^(\\S+) (\\S+) (\\S+) \\[([\\w:/]+\\s[+\\-]\\d{4})\\] "(.+?)" (\\S+) (\\S+) "([^"]*)" "([^"]*)"'
      )
    )
    AS (
     remoteAddr: chararray, remoteLogname: chararray, user: chararray,
     time: chararray, request: chararray, status: int, bytes_string: chararray,
     referrer: chararray, browser: chararray
    );
```

8. Now we can indicate how we want to process date/time values rather than the native types

```
logs =
  FOREACH
    logs_base
  GENERATE
    *,
    CustomFormatToISO(time, 'dd/MMM/yyyy:HH:mm:ss Z') as dtime,
    (int)REPLACE(bytes_string, '-', '0') as bytes;
```

9. Lastly, we get the total number of requests and sum of the bytes sent for each hour in the day:

```
by_hour_count =
FOREACH
   (GROUP logs BY GetHour(ToDate(dtime)))
GENERATE
   $0,
   COUNT($1) AS num_requests,
   SUM($1.bytes) AS num_bytes;
```

10. Lastly, we'll dump the results to the screen:

```
dump by_hour_count
```

The dump command will actually trigger the Hadoop MapReduce jobs to start executing. This is because at this point, the grunt shell is asking for the actual results to be displayed on the screen. If we were saving the data to a repository like S3, DynamoDB or HBase, this would also execute the jobs.

```
16/09/22 21:09:49 INFO util.MapRedUtil: Total input paths to process : 1
(0,10074,100605013)
(1,10116,115774819)
(2,10034,109164571)
(3,10229,91581903)
(4,10344,107076376)
(5,10946,115948485)
(6,10233,89334406)
(7,10362,101377187)
(8,10542,96513122)
(9,10513,107319930)
(10,10236,106364286)
(11,10238,93487615)
(12,10346,95011050)
(13,10164,101439337)
(14,9724,98791432)
(15,9846,92495073)
(16,9549,100318941)
(17,9579,103614434)
(18,9328,101105266)
(19,9355,93863828)
(20,9267,94024847)
(21,9319,79561904)
(22,9468,87023113)
(23,9532,97159777)
```

In the results above, you can see the first element in the tuple is the hour of the day, the second is how many requests, and the third is the sum of the bytes.

Exit grunt by typing

```
quit
```

Submit Pig Work Using the Amazon EMR Console

This section of the lab will show how you can execute the pig script through the Amazon EMR console using a Pig step on a cluster.

The script that we are going to run is very similar to the commands we did interactively. This script saves three new outputs to a bucket that you specify (rather than doing the dump command to stdout).

Let's first look at the contents of the file.

In the ssh window, run the following command:

```
hadoop fs -cat s3://immersionday-rkolak/reports.pig | more
```


1. Now let's create a new S3 bucket to write the results to. If you don't already have a bucket created, click the Create Bucket button on the S3 console at: https://console.aws.amazon.com/s3/ Be sure to create a unique bucket name to use.

Open the Amazon EMR console at https://console.aws.amazon.com/elasticmapreduce/.

2. In the Cluster List, click the name of your cluster created in lab 1.
3. Scroll to the Steps section and expand it, then click Add step.
4. In the Add Step dialog:
5. For Step type, choose

   Pig program

6. For Name, type

   PigSearchAnalysis

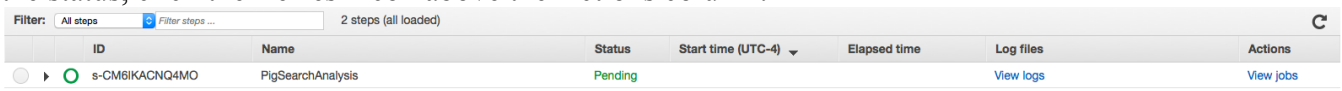
7. For Script S3 location, type the following:

   s3://immersionday-rkolak/reports.pig

8. For Input S3 location, type the following:
   - This is a location that contains the log files we'll analyze

   s3://elasticmapreduce/samples/pig-apache/input

9. For Output S3 location, type or browse to the name of your Amazon S3 output bucket.
   - This is the bucket name from step 1.
10. For Arguments, leave the field blank.
11. For Action on failure, accept the default option (Continue).
12. Click Add. The step appears in the console with a status of Pending.
13. The status of the step changes from Pending to Running to Completed as the step runs. To update the status, click the Refresh icon above the Actions column.



14. Once running, it automatically submitted the job to the cluster:

15. When it completles, chech your S3 bucket for the results:

16. You'll see 4 new folders that contain the results from the pig script:

17. Go through the different folders in S3 and look at the results.

