CINF/CSCI 5931 Big Data Analytics Fall 2017

Assignment 1 MapReduce Programming Using AWS EMR-Part 2

Post Date: Sep 14th,2017 **Due Date**: Sep 28th, 2017 by 11:59pm

In this assignment, you will practice writing MapReduce programs on Amazon Web Services (AWS). You will work in **team of two** for this assignment.

Important Notes:

1. Each of you will get a $40 free credit for AWS when you sign up for Student Starter account. This amount should be sufficient to finish this assignment. It is your responsibility to check your balance frequently. **You will be responsible for any overcharges occur.**
2. **Make sure you debug your program in local mode first before running it on AWS**.
3. In order to cut unnecessary spending, you can use one student's account for exploring and testing. Then use the second one for final work.

Submission Instruction:

You will need to combine all your files into one zip file and name it as lastname1-lastname2-A1. Follow detailed information in this document to properly organize and label all your submitted items.

Make sure you have successfully completed Part 1

1. Connecting to Your Linux Instance

After you launch your instance, you can connect to it and use it the way that you would use a computer sitting in front of you.

Note: After you launch an instance, it can take a few minutes for the instance to be ready so that you can connect to it. Check that your instance has passed its status checks - you can view this information in the Status Checks column on the Instances page.

You could choose to connect to your instance using SSH or PuTTy, depends on your computer. If you want to use SSH, please refer to http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstancesLinux.html. If you want to use PuTTy, please refer to http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html.

2. Getting started with MRJob

Note: The instructions here are for Windows machine. In case you use other operating systems, you could resort to relevant online resources. Some information resources can be found at places such as https://stackoverflow.com/questions/17271319/how-do-i-install-pip-on-macos-or-os-x and https://pip.pypa.io/en/stable/installing/.

mrjob lets you write MapReduce jobs in Python 2.6+/3.3+ and run them on several platforms (https://pythonhosted.org/mrjob/). With mrjob, you will be able to:

- Write multi-step MapReduce jobs in pure Python
- Test on your local machine
- Run on a Hadoop cluster
- Run in the cloud using Amazon EMR
- Easily run Spark jobs on EMR or your own Hadoop cluster

Step 1. Download/install proper versions of Python (please note that OS X and Linux come with Python already installed).

Step 2. Get pip. pip is a package management system used to install and manage software packages written in Python. Python 2.7.9 and later (on the python2 series), and Python 3.4 and later include pip (pip3 for Python 3) by default. But you need to upgrade to the latest version. Please follow instructions on this page (https://pip.pypa.io/en/stable/installing/#do-i-need-to-install-pip) to learn how to use pip to install packages. (Make sure you import pip in python first)

Step 3. Install mrjob. “sudo pip install mrjob" should install mrjob on Mac/Linux flavor OS command line. "python –m pip mrjob" should do it via Windows command line. More information can be found at https://pythonhosted.org/mrjob/guides/quickstart.html#installation.

3. Test Drive on Local Machine

To get you started, we have provided a sample file tweet_sm.txt containing a fairly small number of tweets and a simple program hashtag_count.py that counts the total number of tweets and the total number of uses of hashtags.

Using Windows cmd, ‘python hashtag_count.py tweets_sm.txt’ should run the program. Make sure that the path is set to mrjob folder and both the code and data file are in the folder. You should get the results as 149 hashtags, and 1000 tweets.

Please take some time study the sample data file and mrjob documentation (https://pythonhosted.org/mrjob/) to understand how exactly the map function and reduce function are implemented before you move on to programming them yourself.

As a rule of thumb, you should always test and debug your Map Reduce program locally on smaller datasets, before you attempt it on a big cluster on Amazon—it will cost you money!

4. Setting up mrjob/EMR

Now you should have an AWS account after following instruction in Part 1. In order to run map reduce job, we need use Amazon EMR (Elastic Map Reduce using Hadoop). Amazon EMR makes it easy to spin up a set of Amazon EC2 instances as virtual servers to run a Hadoop cluster.

To interact with AWS, you could do it through the Amazon Console or the Command Line Interface.

Step 1. Create a new file called mrjob.conf and place it into your mrjob folder.

The template for the mrjob.conf file is at https://s3.amazonaws.com/bigalgobucket/mrjob.conf. You need to edit it on instance of EMR.

The file looks like this:

```
runners:
  emr:
    aws_access_key_id: <>
    aws_secret_access_key: <>
    ec2_key_pair: compsci590.4_fall14_Keypair
    ec2_key_pair_file: compsci590.4_fall14_Keypair.pem
    ssh_tunnel_to_job_tracker: true
    aws_region: us-east-1
    num_ec2_instances: 10
    ec2_instance_type: m1.small
# Edit/uncomment the following two lines if you want to tweak
# the maximum number of concurrent map/reduce tasks per node
# (default will be what Amazon pre-configured for the instance type):
#     bootstrap_actions:
#     - s3://elasticmapreduce/bootstrap-actions/configure-hadoop -m
mapred.tasktracker.map.tasks.maximum=2 -m mapred.tasktracker.reduce.tasks.maximum=2
# Edit/uncomment the following three lines if you want to tweak
# the total number of map/reduce tasks per job (default will be
# determined automatically based on how large the input is):
#     jobconf:
#       mapred.map.tasks: 20
#       mapred.reduce.tasks: 20
```

You need to replace the value in the fields with your own instance data. Make sure there is a space between each colon and its ensuing value in mrjob.conf.

- You could get your aws access key information from My Account>Security Credentials.
- Use the .pem file you saved.
- Use this link http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html to make sure you have the right value for aws_region.
- Make sure the num_ec2_instances value does not exceed the quota for the instance type you choose.

Step 2. Run the job using the following command:

```
python hashtag_count.py -c mrjob.conf -r emr tweets_sm.txt
```

You should see the same results as you got from running the program locally.

NOTE: Running this script may take a while. Be patient. Meanwhile, you can check on your AWS account under EMR to see if this mapreduce job is working. Notice that because the job is really small, the overhead of running it on AWS is substantially high. Make sure you check your account to make sure the cluster is terminated and instance(s) are stopped.

Up to this point, you should be familiar with AWS EMR and the basics of running a MapReduce job using mrjob.

Submission Document 1 (50%)

In this document, please clearly label your group members' names and name your file lastname1_lastname2_A1_doc1. You also need to include the following:

1) A screenshot of your AWS Management Console, make sure you capture something denoting your identity.
2) A simple data dictionary—describe the structure and content of the data in the tweets_sm.txt file.
3) Screenshot of results from running hashtag_count locally.
4) Your modified mrjob.conf file.
5) Screenshot of screen output from running hashtag_count using EMR.
6) Screenshot of results from running hashtag_count using EMR.
7) How much time elapsed from the starting of your cluster till you terminated it after the job is done? How much time did it take to actually run the job? What are the conclusions you draw from the difference between the two time?
8) Screenshot of your account balance before and after you run the job.

5. Analyzing White House Visitor Records

The White House publish its visitor records at https://obamawhitehouse.archives.gov/goodgovernment/tools/visitor-records.

The explanation of the data file can be downloaded from Blackboard.

Please download the dataset released in 2012 (28.4MB) and use it for the assignment. You may want to create a subset for testing.

In this part of the assignment, your job is to write MapReduce program and run it using the environment/tools on AWS.

Your MapReduce programs need to provide answers to the following questions:

1) Who are the top 10 most frequent visitors (NAMELAST, NAMEFIRST, NAMEMID) to the White House in the year 2012.
2) Who are the top 10 most frequently visited people (visitee_namelast, visitee_namefirst) in the White House in the year of 2012.
3) (Optional) And some other interesting and non-trivial statistics that you can think of (**Extra credit 10%**).

To run your mrjob on EMR in AWS, you can either do it from your own machine or from a AWS virtual machine. If you want to do it from an AWS virtual machine, you need to have the python class file, mrjob.conf, dataset file on the virtual machine. In order to do that, you may find the following documentations relevant:

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html#Transfer_WinSCP http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstancesLinux.html

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AmazonS3.html

If possible (that is your credit allows), try to see how your EMR setting such as instance type and configuration will affect the performance of the programs.

Submission Document 2 (50%, maybe 60%)

In this document, please clearly label your group members' names and name it lastname1_lastname2_A1_doc2. You also need to include the following:

1) Description of your workflow and program-running environment (try to be as detailed as possible), use screenshots if possible.
2) Your source code.
3) Your analysis results.
4) Summary of job running on AWS.