CSCI 585 - Database Systems
Fall 2103
Homework Assignment 3
Due: 12/11/2013 11:59 PM

Objective:

With previous assignments, you have designed and implemented a database for a hypothetical social network and developed an interface applications to execute queries on top of this database.

Now, assume you are expected to support a real social network social network with large number of users. At scale, running queries on a single machine is unacceptably slow and impractical.

As an alternative to address the aforementioned efficiency problem, with this assignment you will develop sample queries in *MapReduce* to be executed on top of a cloud computing system (namely, Amazon EC2), which allows parallel computation for efficient execution of queries at large scale (Part I). Once you develop these queries, you will also evaluate and report on their performance as the number of machines allocated to execution of the queries grow (Part II).

Your AWS Account:

You need to sign up for a free AWS account if you do not have already one. You can Sign Up here. You can find a tutorial on this here.

After creating an account, you need to send an email from your USC Email to *shelmi@usc.edu* in the following format:

Subject: *AWS Account*

Body: Body of the email should contain the email address that has been used for AWS account.

Once we receive your email, we will provide you with information that allows you to obtain $40 worth of free resources from Amazon EC2. In particular, you will receive an email from *webservices@amazon.com* with the following subject: “Request to add account to Consolidated Bill”. Once you received this email, you just need to click on the link included in the email body and sign in to your AWS account and accept the consolidate billing request to redeem the credit allocated to you for this assignment.

Important Notes About Your AWS Credit:

- Make sure you do NOT exceed your credit quota ($40)! You will not receive any more credits beyond your quota limit. Make sure to stop or terminate any machines and services once you are not actively working on your assignment; otherwise, your credit will diminish quickly before you get to finish your assignment! If you have any questions, in this regard feel free to consult with the TAs before starting to use your credit.
- When you want to create a MapReduce cluster in the “Elastic Map Reduce Job Flow” make sure to set the “Auto Terminate” to “Yes”.
- You should develop and test your applications locally and only run it on the Amazon clusters once verified locally
- You are only allowed to use EMR (Elastic Map Reduce), EC2 (Elastic Compute Cloud) and S3 (Simple Storage) services. DO NOT USE ANY OTHER AWS SERVICES!
- You can implement your Hadoop MapReduce applications using your desired language and IDE. However you need to follow the MapReduce paradigm.

.

Description

Input Data Files:

You need to use the provided data files as the input and you are not allowed to change them at all. The data is only used as input.
The user.txt contains the information of 40 million users in the following format:

```
userID, age, interest, x coordinate of the user location, y coordination of the user location
```

Note that (x, y) indicates the absolute of the user based on some assumed reference (not latitude and longitude). Given the two points (x_1, y_1) and (x_2, y_2), the distance (d) between these points is given by the formula:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

The friendship.txt contains the users' friendship information in the following format (indicating a friend ship between two identified users):

```
userID, userID.
```

Data files are uploaded on Blackboard along with this assignment.

Part 1: Query Development

In this part of assignment, you are allowed to develop and run your application locally or on the Amazon. If you decide to run your program on the Amazon EMR, run it on a single small machine.

1- Write a Hadoop MR application to find how many friends each user has (don't need to include the users who have no friends as a part of output).
For this query you need to submit (**25 points**):

- **1io.txt** which shows the input and output of your Map and Reduce functions. for example:
  Map: <key 1, value 1> → List <key 2, value 2>
  Reduce: List <key 2, value 2> → <key 3, value 3>
  You need to briefly describe what your Map() and Reduce() functions do.

- **1code.txt** which contains your Hadoop code. You need to put your Map, Reduce and Job (main) functions (or classes according to your code) in a text file. If you would like to use additional classes or functions, put them in this text file as well.

- **1out.txt** which contains the result of question 1 in the following format:
  userid number_of_friends
  userid number_of_friends
  …

2- Assume for each interest, there is a fixed reference location (x*, y*); for example, for football lovers, the reference location can be the location of Los Angeles Memorial Coliseum! You can find the reference location for each interest in the "interest location.txt" uploaded on Blackboard along with this assignment. Write a Hadoop MR application to find the number of all people who share the same interest and are within a 5 mile range from the reference point of that interest. For this query, you need to submit (**25 points**):

- **2io.txt** (for description, see similar description for **1io.txt** above)
- **2code.txt** (for description, see similar description for **1codeio.txt** above)
- **2out.txt** which contains the result of question 2 in the following format:
  interest number_of_users
  interest number_of_users
  …

3- Assume that we have identified the following age groups: **a:** 5-14, **b:** 15-24, **c:** 25-34, **d:** 35-44, **e:** 45-54, f: 55+. We are interested in finding the users' age distribution. Write a Hadoop MR application to find the number of users belong to each group. For this query, you need to submit (**25 points**):

- **3io.txt** (for description, see similar description for **1io.txt** above)
- **3code.txt** (for description, see similar description for **1codeio.txt** above)
- **3out.txt** which contains the result of question 3 in the following format:
  a number_of_users
  b number_of_users
  …

Part 2: Evaluation of Scale-out Effect

In this part of the assignment, you need to run the query no. 3, on 1, 2 and 4 EC2 small machines, respectively. Accordingly, you must prepare and submit the following (**25 points**):

- **4.jpg** which contains a histogram in which the X axis shows the number of machines used to run the query, while the Y axis shows the time it takes to process the query (in seconds) in each case.

Submission Guidelines:

You need to submit your assignment through the Blackboard before the deadline. No deadline extensions will be awarded. Please compress all of your files into one .zip file named “Your_USC_Email_ID.zip”. For example if your USC Email ID is “john”, name the file as “john.zip”. The zip file should contain following items:

- **1io.txt**
- **1code.txt**
- **1out.txt**
- **2io.txt**
- **2code.txt**
- **2out.txt**
- **3io.txt**
- **3code.txt**
- **3out.txt**
- **4.jpg**

Useful Links:

- Hadoop
- AWS SDK For eclipse shows how you can add AWS plugin to your eclipse.
- AWS MapReduce Training. This link contains some video tutorials for AWS MapReduce.
- You can find some good video tutorials regarding MapReduce basics and programming here.
- There is another programming MR programming example here.
- Learn how to install Hadoop on a Windows machine here and here.
- Learb how to install Hadoop on Mac OS here.

  .

As always, please feel free to post your questions on the Blackboard Discussions.