CS 5513 - Spring 2019- Homework 4
Assigned: 4/16/2019
Due: 4/25/2019 by 3:00 PM to the class website
Maximum Points: 75 points

Notes:

- **Homework answers must be typed and submitted by 3:00 PM to the class website.**
- **Homework is individual work; it must be done by you only, no collaboration with anyone else is allowed.**
- **Late homework must be submitted by 11:59 PM to the class website on the date following the due date which will be graded with 5% (of the maximum points) penalty. Late homework submitted after this time will not be graded.**

We have a database that contains three tables, twitter_account, follows_account and stack_overflow_account, with the following schemas:

- **twitter_account**(
  twitter_account_id: int,
  email_address: string,
  phone_number: string,
  user_location: string,
  num_tweets: int)
- **follows_account**(
  follower_twitter_account_id: int,
  subject_twitter_account_id: int)
- **stack_overflow_account**(
  email_address: string,
  reputation: int,
  num_questions: int)

Every twitter account has a unique twitter_account_id and a unique email_address. The data for the three tables are contained in the files `HW4-twitter_account.csv`, `HW4-follows_account.csv`, and `HW4-stack_overflow_account.csv`, respectively, posted on the class website. Using Apache Pig, perform the following tasks:

Tasks:

1) For each of the following queries (1a-1c), create a script in Pig named `query_<letter>.pig` that implements the query:
   a. Given a positive three-digit integer *k* that represents the phone area code *read as a user input to the query*, find the email addresses associated with the twitter accounts that have a phone number starting with that area code *k*. The output of this query must have the format:

      (k, {email_address_1, email_address_2,…})

b. For every user_location, find the number of twitter accounts that have that user_location value. The output of this query must have the format:

   (‘Houston’, 12)
   (‘Oklahoma City’, 34)

c. Given a positive integer *k read as a user input to the query*, find the average reputation of stack_overflow accounts associated with the twitter accounts of people who have published more than *k* tweets. The output of this query must have the format:

   (num_tweets, avg_stack_overflow_reputation)

2) In this task you will implement a **single iteration** of the PageRank algorithm discussed in class to rank twitter accounts where a twitter account X is treated as a webpage X, a twitter account Y that X follows is treated as X’s forward webpage (link), and a twitter account Z that follows X is treated as X’s backward webpage (link). The rank of a twitter account is thus the same as the PageRank of a webpage in the PageRank algorithm. To do this, create a script in Pig named `twitter_account_rank_iteration.pig` that first reads the file `HW4-old_twitter_account_rank.csv` Each line in this file contains the initial twitter_account rank (i.e. PageRank) for a twitter account in the following format:

   (twitter_account_id, twitter_account_rank)

   Then, your Pig script will compute the new rank for every twitter_account using the following PageRank formula:

$$rank(account_i) = \frac{1-d}{n} + d * \sum_{account_j \in followers(account_i)} \frac{oldrank(account_j)}{|subjects(account_j)|}$$

   where *d* is the damping factor with the value of *0.85*, *n* is the total number of twitter_accounts, *followers(*$account_i$*)* is the set of twitter accounts that follow *$account_i$* and |*subjects(*$account_j$*)*| is the number of twitter accounts that *$account_j$* follows. Then, your program must store these computed ranks back into the file `HW4-old_twitter_account_ranks.csv`, **overwriting the previous ranks**.

3) In this task you will implement a Pig script named `find k_percentile_accounts.pig` to find the twitter_account_ids, email_addresses and twitter_account_ranks of the *k-percentile* twitter accounts based on their twitter_account_ranks where *k* is a user input to the query. The output of this query must be in the format:

   {(twitter_account_id$_1$, email_address$_1$, twitter_account_rank$_1$), (twitter_account_id$_2$, email_address$_2$, twitter_account_rank$_2$),…}

Required Implementation:

- Implement the tasks using Pig Latin installed in the oracle18.cs.ou.edu machine. To ssh into that machine use your login and password sent to you by the IT administrators earlier this semester.
- You *must not use any Python/Java packages implementing APIs to Pig*. This means

that you must not use org.apache.pig, or anything similar. **You must write all the code by yourself.**

- Use the file `HW4-follows_account.csv` containing the information of which twitter account follows which twitter account. Each line in this file has the following format:

  follower_twitter_account_id,subject_twitter_account_id

  where follower_twitter_account_id is the id of the twitter account that follows the account of the subject_twitter_account_id. This file is posted on the class website.
- Use the file `HW4-old_twitter_account_rank.csv` containing the initial values for the twitter_account_ranks. Each line in this file has the following format:

  twitter_account_id,initial_twitter_account_rank

  where twitter_account_id is the id of a twitter account and initial_twitter_account_rank is the initial rank of that twitter account. This file is posted on the class website.
- Test your queries in Task 1 as follows: Query 1a: 3 times; Query 1b: 1 time; and Query 1c: 3 times. Test your script of Task 2 one time and test your script of Task 3 two times.
- The results of each query/task must be displayed after the query execution (use the diagnostic operator DUMP).
- The implementation of each query/task must be saved in an individual text file.
- The output of each query/task must be saved in an individual file (use the STORE operator in Pig Latin).
- Every single Pig command *must have a comment preceding* it that indicates the format of the output table of the command. To obtain this format, use the Pig command `describe`. For example,

  ```
  -- Format:
  --      (group:int,
  -- {(follower_account_id:int, subject_account_id:int)})
  SUBJECTS_FOLLOWED = GROUP follows_account BY follower;
  ```
- Submit the soft copy of all query/task files and results to the class website.
- Your soft copy must be a plain text file with the extension `.pig` and it must be directly runnable in Pig. Your soft copy MUST NOT be a WORD or PDF file as we will run your text file on Pig to check its correctness.
- When uploading your files to the class website, you MUST NOT compress your files or put them in a compressed folder of any kind. This implies, among other things, that you MUST NOT upload files with the extensions .zip, .bz, .tar.gz, .tar.bz, .7z, and .rar.

Important notes:

- All of your files must be well documented. Your work will be graded based on correctness as well as documentation.
- Before working on this homework assignment, you should have reviewed the “Lecture Topic 8 - MapReduce and Pig Latin” and “Lecture Topic 8 - Pig Latin Examples and Execution Instructions” slides posted on the class website and the references listed below.
- Keep a soft copy of all your files and results in your directories. To check if your program works correctly, we will have live demos. The exact time and date for the live demos will

be announced later. You MUST NOT modify your files or results after you turn them in. You will get a zero grade for this homework assignment if you violate this rule.

References:

- http://pig.apache.org/docs/r0.16.0/start.html
- http://pig.apache.org/docs/r0.16.0/func.html
- http://pig.apache.org/docs/r0.16.0/test.html
- Langville, Amy N.; Meyer, Carl D. (2003). "Survey: Deeper Inside PageRank". *Internet Mathematics* **1** (3).
- Sergey Brin and Lawrence Page. The Anatomy of a Large-Scale Hypertextual Search Engine. Available from http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf
- Ramez Elmasi and Shamkant Navathe, Chapter 27: Introduction to Information Retrieval and Web Search," Fundamentals of Database Systems, 6th Edition, Addison-Wesley, 2011. This chapter is on reserve for CS 5513 in the OU main library (the Bizzell Memoerial Library).