School of Computer Science, McGill University

COMP-421/764 Database Systems, Winter 2021

Written Assignment 3: MapReduce and Pig Latin

Due Date April 08, 05:00 pm EST

This is an individual assignment. You are required to work on your own to create the solution.

This assignment is worth 8% of your course grade. The total points in this assignment is 24.

For each day late, 15% of the maximal achievable points will be subtracted from the achieved points. Maximum of 2 late days allowed.

Please read the complete assignment description, including the **Guidelines** section before starting your work.

Ex. 1 — Using MapReduce in Social Media(3 Points)

A common element of many social networking applications is to find a list of friends that you have in common with another person. Many applications store the list of friends associated with a person in a structure similar to what is given below (`Joe`'s friends list).

```
Joe, (Abe,Jane,Ali,Zack)
```

Further, if `Ali` is a friend of `Joe`, with a friend list,
`Ali, (Sheila,Jane,Zack,Mary)`

Their common friends would be `(Jane,Zack)`.

It is often useful to pre-compute such lists in advance so that it is readily available when needed (say when `Ali` visits `Joe`'s profile page.). For example, we can store the above information as

```
(Ali,Joe), (Jane,Zack)
```

Where the first tuple functions as the key (used by the application to lookup), and the second tuple is the value (common friends list).

In our simplified system, we will assume that we need to have this information only when someone visits an immediate friend's profile page. We will also assume that any friendship is mutual. Names are unique.

(a) Design a MapReduce workflow that will take as input the person information data set given above (assume the person's name is the key and the friends list is the value). It should produce a pre-computed list of common friends for each pair of people who are friends. Write some simple comments in the design about your logic.

(b) The application trying to retrieve the common friend list of `X` and `Y` when `X` is visting `Y`'s profile will lookup the list associated with `(X,Y)`.

(c) Does the MapReduce workflow design store information that is practically duplicate? If so what can you do to remove this from your MapReduce workflow and what change can you do to the application's lookup strategy to address this?

(d) Once you have developed your workflow, give a couple of examples of what the input and output of each mapper and reducer in your workflow would look like. Writing this down as you develop your design may help you notice and fix any logical mistakes.

Turn in:- Your solution (typed) in **assignment3.pdf** under a section **Q1**.

Ex. 2 — Using MapReduce and Relational Operations(5 Points)

Consider two data sets that follow the model given below.

`Hospital(Hname, Province)`
`Patient(HInsurNum, age, Hname)` `Hname` references `Hospital`

Design a MapReduce workflow that will read these data sets and produce in its final output, provinces that have more than 100 patients in their hospitals who are above the age of 60 and the total number of such patients in each province. (An example output is given below)

PQ, 352
ON, 284

Make sure to include some comments, example inputs and outputs for each of the mappers and reducers, etc.,
Turn in:- Your solution (typed) in **assignment3.pdf** under a section **Q2**.

Ex. 3 — PigLatin - Warmup(0 Points)
The goal of this exercise is to help you verify that you are able to acess the MapReduce cluster and execute Pig Latin scripts:

The data set used for all the Pig Latin questions are based on a modfied version of the Covid data set provided by stats Canada[1] which is already loaded into the HDFS. The individual records of the data set is basically the covid related summary captured for each province / territory for a day.
This data set contains the following fields:

1. `prname` → Name of the Canadian province / territory.
2. `idate` → the day for which the data is captured.
3. `newcases` → the number of new cases reported on that day.
4. `newdeaths` → the number of deaths reported on that day.
5. `tests` → the number of tests conducted on that day.
6. `recoveries` → the number of recoveries reported on that day.

You are provided with an `example.pig` script, that contains the necessary `LOAD` instructions to load the data from HDFS to a schema described above. You should be able to reuse this for the remaining exercises.

It is important that you read through the supporting **PigLatinInstructions-vvv.pdf** file before starting to write and execute the Pig Latin scripts.

You can either run the script as it is by passing the script as an argument.

```
$ pig example.pig
```

or by starting pig by itself first

```
$ pig
```

and then copy pasting each statement by itself (for interactive programming).

The example script is attempting to answer the question as to when was the first case reported in Quebec.
The script starts by first selecting only those records from the data set that belongs to Quebec and has new cases above zero. In the next step, it then projects only the date associated with the data set (as that is the only field required in the output). The script then sorts this data by the date. Finally, the script limits the output to the first record (which is the date on which the first case was reported as per the information in the data set).

We can see that in many ways these individual steps are similar to those performed during query evaluation, except that the onus is on the programmer to figure out the order of execution of the steps instead of the data management

[1] https://open.canada.ca/data/en/dataset/b8d1d622-1ceb-4c1c-96e9-a0b38939080b

system performing an optimized order.

The script will take a couple of minutes to run and produce a lot of messages. At the end, you should be able to see an output like this (truncated for brevity).

```
...
(2020-03-01)
...
```

Turn in:- Nothing.

Ex. 4 — PigLatin - Covid Recovery Info (3 Points)

In this task, we will try to find the days in quebec when the number of recoveries were at least twice as much as the number of cases reported on that day. However, for this purpose, we will only consider those days where there were at the least 50 new cases as well as at the least 100 recoveries.

- Start by making a copy of **example.pig** to **Q4.pig**.
- Modify the selection condition to select only those records with at the least 50 new cases as well as at the least50 recoveries.
- Modify the projecton condition to include the date, the number of cases, recoveries and the ratio of recoveries to newcases (**not percentage**) in the output. (You may have to do some programming language tricks like multiplication with 1.0 if you want to force Pig Latin to perform floating point division).
- Filter the above to select only those with a ratio of 2.0 or above.
- Order this by the date and print the results to the screen.
- Once you have the script completed and is satisfied with its output, execute it the following way (for submission purposes).

```
$ pig Q4.pig > Q4.log 2>&1
$
```

- The output of your script should contain the date, number of new cases, number of recoveries and their ratio in that order of attributes. The result part of your script's output should follow the example format below as-is (truncated for brevity).

```
...
(2020-06-10,157,643,4.095541401273885)
(2020-06-15,130,478,3.676923076923076)
...
```

Make sure that the log file also contains the various information that pig has been producing and not just the final results. If those log information from pig is missing, points will be deducted.

Turn in:- Q4.pig and **Q4.log**

Ex. 5 — PigLatin - Covid Mortality (5 Points)

(a) Write a Pig Latin script **Q5.pig** that will compute the total number of deaths per province. Include only those provinces with 100 or more total deaths. Order the output with the provinces with the highest death tally on the top.

(b) Similar to previous exercise, also produce **Q5.log**.

(c) What does the schema look like immediately after you perform the `GROUP` operation step? Include this under a section **Q5** in your **assignment3.pdf**.

(d) The output of your script should contain the name of the province, and the total mortality in that order of attributes. The result part of your script's output should follow the example format below as-is (truncated for brevity).

```
...
(Some Province,2966)
(Another Province,1042)
...
```

Turn in:- Q5.pig and **Q5.log** (and the contents in **assignment3.pdf**).

Ex. 6 — PigLatin - Covid Mortality Rate in Quebec (8 Points)

(a) Write a Pig Latin script that produces a list of highest mortality days of Quebec. We will define such days as those days with the number of deaths being at 1% or more of the total number of deaths reported from Quebec across the entire data set. You can make use most of the logic from the previous question to start with. At some point, you will have to perform a join of that aggreggated data set with the Quebec data for individual days to compute deaths/totaldeaths.

(b) Similar to previous exercise, also produce **Q6.log**.

(c) What does the schema look like immediately after you perform the `JOIN` operation step? Include this under a section **Q6** in your **assignment3.pdf**.

(d) The output of your script should contain the date, number of reported deaths on that day and what percent it composes of the total deaths from the province. The result part of your script's output should follow the example format below as-is (truncated for brevity).

```
...
(2020-05-03,130,1.210202941724073)
(2020-05-06,115,1.070564140755911)
...
```

Turn in:- Q6.pig and **Q6.log** (and the contents in **assignment3.pdf**).

Guidelines

NO Handwritten / scanned submissions are accepted for this assignment.

MapReduce

This discussion is pertaining to Exercises Ex.1 and Ex.2.

- Your solutions should have only `Mapper` and `Reducer` functions. **DO NOT** use the `Combine` functionality.
- When implementing a solution, remember that in some cases you may need more than one MapReduce job to accomplish a task (Output of one MapReduce's `Reducer` forms the input of another's `Mapper`).
- Each MapReduce step goes through the disk in order to pass data to the next step in the process. Therefore, come up with a solution that will reduce the number of MapReduce jobs required as well as reduce the amount of data that will have to flow from one part of the MapReduce process to the next one (think of some of the simple concepts we applied for query evaluation).
- You can follow the pseudo-code syntax as was shown in class. Our primary interest is to see if you know how to design the workflows to pick the right key/value for the input/output of mappers and reducers and have an understanding of the internal logic that you should put inside these functions. Please do not write Java code, etc.

Pig Latin

This discussion is pertaining to Exercises Ex.3 through Ex.6.

- Watch the tutorial and try to do the statements along side (type it by yourself). This is a good way to get warmed up to the Pig Latin statements before you tackle the assignment problems.
- Your code does not have to be optimized.

- Remember, the `DESCRIBE` command can be very handy to figure out if the schema of your data set is changing as it goes through some of the complex steps (such as `JOIN` and `GROUP`). Trying to access attributes incorrectly may result in pig throwing errors that can be confusing and frustrating. So use this command to investigate it. You can leave the `DESCRIBE` commands in your submission scripts if you would like to or comment them off, once their purpose is served. They are locally interpreted by the pig client and therefore has no significant overhead.

- If you are debugging, it is recommended to run pig interactively rather than using a script. That way, you can adjust the logic as required, run `DESCRIBE` commands, etc., without having to start the execution from scratch as executing a workflow end-to-end can take a few minutes.

- Use the `DUMP` command in the intermediate steps to help you debug to see if you got the logic correct up to there. You can comment it out once your are done with your work. Your final submission should have only one `DUMP` command for your final result. Remember! every `DUMP` command triggers the execution of the MapReduce framework. Verify the log files that you are submitting to ensure that it clearly displays the final (expected) results.

- You must not use the `STORE` command to store anything into the HDFS. Your scripts can also run into problems if it encounters results already stored in the HDFS.

- The example outputs given above are not based on actual solution outputs. So do not try to reproduce those values.The examples are there to show the expected formats. Operations such as `GROUP` can have impact on the data format. It is important that you learn what happens in those circumstances and how you can transform the data back to the required format.

- Your final output (results) format must not be nested, but flat (as shown in the examples.). Below is an example of a format that is not acceptable.
  `(A,(B,C))`
  Instead, it should be of the following format.
  `(A,B,C)`
  Explore the `FLATTEN` command if you have trouble addressing this.

What to turn in

assignment3.pdf that contains the two MapReduce pseudo-code solutions as well as the extra information requested for the Pig Latin questions. The Pig Latin scripts and log files themselves: **Q4.pig**, **Q4.log**, **Q5.pig**, **Q5.log**, **Q6.pig** and **Q6.log**. Ensure that your log files have any information that pig has been producing (not just the results) and most importantly, also the intended final result.

You may `tar` or `zip` your submission if you have to, but make sure that you verify your submission contains all the files and they are correct. There will be no accommodation for submitting incorrect files.

Questions ?

Please use Piazza for any clarifications you need. Do not email the instructor or TAs as this leads to a lot of duplicate questions and responses (not an efficient system). Please check the pinned post "A3 general clarifications" before you post a new question. It might have been already addressed there, in which case we will not address it again.

Remember, you have access to the dataset in the HDFS. So any questions of the form "is Y a possible value for attribute X" can be answered by yourself by looking (analyzing) that data.

There will be specific TA office hours for the assignment that will be announced closer to the due date.