Project 2: Big Data Management – Fall 2023

"Advanced Hadoop"

Total Points:	**100**
Given Out:	**Week 6: Monday, Sept 25th, 2023**
Due Date:	**Week 8: Thursday, Oct 12th, 2023**
	Submit the project via CANVAS.
Teams:	**Project is to be done in assigned teams (posted on CANVAS).**

Project Overview

In this project, you will use advanced Hadoop. First, we are going to revisit Project 1 with Apache Pig, then we are going to tackle a clustering problem with the Map-Reduce framework. Finally, a creative part will allow you to explore additional directions that you are interested in.

Project and Deliverables:

1. **One** member of your team will submit a **zip file** containing the programs for creating data files, Java code for your MapReduce analytics via CANVAS for your team. You are welcome to submit the zip file of your IDEA project directory. However, **make sure you don't include any large dataset** (a test dataset small is fine). Please don't submit a .jar file. Use the following submission format: project2_team-number.zip (for example team 5's file name will be project2_5.zip).
2. You will also submit a document (pdf) containing documentation that describes how you accomplished each task. Keep in mind that it is not just important that it "runs", it also should be a scalable solution. Please include screenshots of uploaded input data files and generated output files on HDFS (additional details will follow below).
3. In your project report, please indicate **the relative contributions of each team member explicitly.**[1] This requires you to discuss with each other your expectations and how best you can work together and help each other succeed at this first project. By submitting, all team members confirm the division of labor as indicated in your report.

[1] For instance, if each team member has done the project independently, and then only at the end you pulled the best of the material together, you need to say so. Or, if one team member helped and taught the second team member how to do it, and then the 2nd team member succeeded to do some of the queries (even if with some guidance), please report this. If you have closely collaborated and done the same amount of effort working side by side helping each other, also please state this.

Using Generative-AI (e.g., ChatGPT):

I encourage responsible and sensible use in the assignment. Please report if, how, and what you used to complete the assignment. Explain how you validated the trustworthiness of the solution, which prompt/s you used, and how you used the output of the model (basic code/documentation/etc.).

Project Demonstration

Once completed, one or at most two teams may be asked to provide a brief demonstration of their results to your classmates to review how you solved this project. **If necessary for grading, the instructor/TA will communicate with your team about your project, and also may request a demonstration of your solution.**

Project Description

Please complete the following tasks.

1- Revisiting FaceIn with Pig [20 Points]

Rewrite the 8 Tasks (part 3 a-h) from Project 1 using Apache Pig and test them on the data you created as a part of Project 1. In your report, describe how you chose to implement each task and discuss the trade-off between using Apache Pig and standard Map-Reduce. [2.5 pts for each task]

2- K-means Clustering [40 Points]

K-Means clustering is a data mining algorithm that groups objects together (see, for example, http://en.wikipedia.org/wiki/K-means_clustering#Standard_algorithm for additional details). It starts with an initial seed of K points (randomly chosen) as centers, and then iteratively enhances these centers with the closest data points in the dataset.

In this assignment, your system should terminate if either of these two conditions become true:

a) The K centers did not change over two consecutive iterations

b) The max number of iterations has been reached (parameter R for rounds)

Tast 2.1 (Creation of Dataset):

- Create a **large** dataset of at least 5,000 points consisting of 2-dimenional points, i.e., each point has (x, y) values. x and y values range from 0 to 10,000.
- Create another file containing K initial seed points with each of its values also in the range from 0 to 10,000. Make "K" value as a parameter to your program,

such that your program will generate these K seeds randomly, and then upload the generated file to HDFS.

Step 2 (Clustering the Data):

Develop the K-means clustering strategies described below using java map-reduce jobs. You should document the differences between your solutions, i.e., what changes were made to the mapper, at the reducer, number of mappers/reducers, or in the main control program. These algorithms should include:

a) A single-iteration K-means algorithm (R=1) [5 pts]
b) A (basic) multi-iteration K-means algorithm (remember to set the parameter R to terminate the process, e.g., R=10). [5 pts]
c) An additional (advanced) multi-iteration K-means algorithm that terminates potentially earlier if it converges based on some threshold. [5 pts]
d) An additional (advanced) algorithm that also uses Hadoop Map Reduce optimizations as discussed in class (e.g., a combiner). [5 pts]
e) For your K-means solution in subproblem (d) above, design two output variations:
    i. return only cluster centers along with an indication if convergence has been reached; [5 pts]
    ii. return the final clustered data points along with their cluster centers. [5 pts]
f) Provide a description for each of your above five solutions in your report and conduct experiments over them, for example by choosing different K values and different R values. describe the relative performance, explain, and analyze your findings. [10 pts]

Note: Since the algorithm is iterative, you need your main program that generates the map-reduce jobs to also control whether it should start another iteration.

3- Get Creative! [40 Points]

Choose 2 alternatives to extend the project. [20 pts for each alternative] You can either choose one of the following options or come up with an option of your own. You need to specify in your report which 2 alternatives you have implemented in Project 2.

a. **BYOD (Bring Your Own Data):** Find a relevant dataset online (e.g., using kaggle) and run your K-means algorithm over the new dataset by repeating the steps from above (2a-e). Describe the dataset in your report and repeat the experiments from above (2f) and add to the report.
b. **Offer an extension of k-means (e.g., k-medoids):** Provide an adaptation to the K-means algorithm implemented above. Repeat the steps from above (2a-e) and the respective experiments (2f) and add to your report.

c. **Implement a different clustering algorithm:** Provide an implementation (in Java map reduce code) of an alternative clustering algorithm (e.g., Hierarchical Clustering), repeat the steps from above (2a-e) and the respective experiments (2f) and add to your report.
d. **Visualize the output of the clustering algorithm/s:** Enrich your report by visualizing the clustering results. This implementation should be done using Java code and integrated with Hadoop. You can use, for example, Apache Zeppelin, which has a Java interpreter and integrates with Hadoop. Use the added visualization to support your findings (2f) and update your report accordingly.
e. Evaluate the output of the clustering algorithm/s: Enrich your report by implementing a clustering evaluation measure (e.g., Silhouette). This implementation should be done using Java code with Hadoop map reduce. Use the added evaluation to enrich your findings (2f) and update your report accordingly.

You are welcome to use us (instructor/TA) to browse for ideas.