CS6675/ CS4675

Homework Assignment 3
(Programming Category)

Student Name:______________________

Student Session: cs6675 or CS4675 (circle one)

You are given three types of programming problems in the second homework assignment. Problems 2 and 3 consists of multiple options. You only need to choose the first problem or one option from the second or third problem as your second homework. Feel free to choose any of your favorite programming language Java, C, Perl, Python, …

Post Date: Monday of Week 6 (Feb. 15)
Due Date: Midnight on Friday of Week 7 (Feb.26)

Problem 1. Learning Optimal Configuration of HDFS / Hadoop MapReduce. *This problem contains two options and suitable for students who have some experience with Hadoop MapReduce Platform already. For both options, you need to do the following 3 steps first:*

(1) Install HDFS and Hadoop MapReduce on your laptop, and select two example map-reduce programs/applications provided with the package, such as word count, sort, grep.

(2) use the default configuration for both HDFS and Hadoop MapReduce and measure and report the runtime performance for each example program using two different sizes of datasets (you can triple the given dataset to generate a larger one). Note the package comes with both code and datasets.

(3) You may use excel file to generate your runtime statistics plot or organize the performance measurement data in a tabular format.

Option 1: Chunk Size optimization
Go through the configuration file to set the map input chunk sizes to small or larger than the default size such that you can have varying map input file size (say 3~5 different chunk sizes). Now measure the performance of the two MapReduce Program and explain the difference observed. Hint: you need to try to select chunk sizes that have larger differences such as 64KB, 128KB, 256KB, 512KB, 1GB, 2GB.

You are asked to analyze your experimental comparison results and provide your intuition and discussion to elaborate what you observe and why.

Option 2: JVM Queue Size Optimization
Go through the configuration file and adjust JVM queue size for Map slot and Reduce slot and run the same two MapReduce programs using the same #mappers and the same #reducers. Compare with the default setting of JVM queue size.

You are asked to analyze your experimental comparison results and provide your intuition and discussion to elaborate what you observe and why.

Deliverable.

(a) URL to the MapReduce codes and the datasets used
(b) screen shots of your execution process.
(c) Runtime statistics in excel plots or tabular format.
(d) Your analysis.

Problem 2. Learning Configuring SPARK Jobs

This problem contains 3 options and suitable for students who have some experience with Spark Platform already. For both options, you need to do the following 3 steps first:

1. Download Spark on your laptop and run one example program of your choice on two example datasets provided in the Spark package.

2. Report the runtime performance for your chosen example program using two different sizes of datasets.

3. You may use excel file to generate your runtime statistics plot or organize the performance measurement data in a tabular format.

Option 1: Chunk Size optimization
Go through the configuration file to vary the input data sizes from small to larger than the default size such that you can measure how different input size of data may impact on Spark performance. You can run MapReduce on Spark and use two MapReduce programs such as wordcount, grep as your test applications. Then measure the performance of the two Spark Programs and explain the difference observed.
Hint: you need to try to select chunk sizes that have larger differences.

You are asked to analyze your experimental comparison results and provide your intuition and discussion to elaborate what you observe and why.

Deliverable.

(a) URL to the Spark code and the datasets used
(b) screen shots of your execution process.
(c) Runtime statistics in excel plots or tabular format.
(d) Your analysis.

Option 2: Configuration Optimization

Go through the configuration file of Spark and the program such as MapReduce you plan to run on Spark, and identify at least one configuration parameter that you want to reset by modifying the default value, such as JVM queue size. Compare with the default setting with the new settings you have used. Ideally you should consider 3-5 different settings compared to the default to gain a better understanding on how to optimize Spark configuration.

You are asked to analyze your experimental comparison results and provide your intuition and discussion to elaborate what you observe and why.

Deliverable.

(e) URL to the Spark code and the datasets used
(f) screen shots of your execution process.
(g) Runtime statistics in excel plots or tabular format.
(h) Your analysis.

Option 3.

This option is designed for students who are familiar with both Hadoop MapReduce and Spark and interested in hand-on comparison of them through example programming problems or big datasets and / or through configuration tuning.

Compare Hadoop MapReduce and SPARK using a common analytic problem: a simple one like sort or word count, and a complex one like Clustering or k nearest neighbor search. You are encouraged to write your own program.

Deliverable.

(a) URL to the HDFS/Spark code, the MapReduce code and the datasets used
(b) screen shots of your execution process.
(c) Runtime statistics in excel plots or tabular format.
(d) Your analysis.