CS6675/ CS4675

Homework Assignment 4

(Programming Category)

Student Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student Session: cs6675 or CS4675 (circle one)

You are given three types of programming problems in the 4th Programming homework assignment. You only need to choose one of the problems for this assignment. For the problem with multiple options, you only need to choose one of the options within a problem. Feel free to choose any of your favorite programming language Java, C, Perl, Python, …

Post Date: Monday of Week 8 (Feb. 29)

Due Date: Midnight on Friday of Week 9 (March 11)

**Problem 1. Learning BigTable, NoSQL, Key-Value Systems**

This problem has 2 options.

**Option 1.**

This option is designed for students who are familiar with HDFS but are beginner of Big Table/NoQL/Key-Value Store Systems and have no hand-on experience with any existing Big Table systems.

1. Download one of your favorite NoSQL (key-value) store software package, such as HBase, Redis, LevelDB, Mongo, and so forth.
2. Populate the system with example dataset. For example, you can use YCSB to generate dataset or use real world datasets available in public domain. See an example set of datasets in the course resource tab.
3. Run 2-3 types of query workloads and measure the performance of the NoSQL system. Provide your analysis on the experimental results.

4. Data size in MongoDB is typically higher due to e.g. each document has field names stored it.

5. Less flexibity with querying, because there is no such

Deliverable.

1. URL to the NoSQL download and the datasets used
2. screen shots of your execution process.
3. Runtime statistics in excel plots or tabular format.
4. Your analysis

**Option 2.**

This option is designed for students who are familiar with Big Table/NoQL/Key-Value Store Systems.

1. Download two of your favorite NoSQL (key-value) store software packages, such as HBase, Redis, LevelDB, Mongo, and so forth.

2. Populate the two system with the same example datasets. For example, you can use YCSB to generate dataset or use real world datasets available in public domain. See an example set of datasets in the course resource tab.

3. Run 2-3 types of query workloads and measure the performance of the two NoSQL systems. Provide your comparative analysis.

Deliverable.

1. URL to the NoSQL systems and the datasets used
2. screen shots of your execution process.
3. Runtime statistics in excel plots or tabular format.
4. Your comparative analysis.

**Problem 2. Hand-on Experiences with k-means clustering**

**Option 1: Understanding k-means clustering**

1. Install Mahout on your laptop, and run the k-means algorithm (http://mahout.apache.org/users/clustering/k-means-clustering.html) or k-means commandline (<http://mahout.apache.org/users/clustering/k-means-commandline.htmlon>);
2. Select two datasets from the UCI repository (<http://archive.ics.uci.edu/ml/datasets.html>) or use datasets of your own, and
3. Report the runtime for two different datasets and the quality of clustering. Some reference on clustering evaluation can be found at <http://nlp.stanford.edu/IR-book/html/htmledition/evaluation-of-clustering-1.html>.
4. You may use excel file to generate your runtime statistics plot or organize the performance measurement data in a tabular format.
5. You are encouraged to learn by setting different k and varying the initial points and report the quality and runtime performance of Mahout K-means.

Deliverable.

1. Source code (if available)
2. screen shots of your execution process.
3. Runtime statistics in excel plots or tabular format.

**Option 2: Hand-on Experimentation with k-Means Clustering**

You are asked to select and evaluate clustering algorithms using Mahout, R or Weka (http://www.cs.waikato.ac.nz/ml/weka/).  You are also welcome to implement your own k-means clustering algorithm and evaluate it.

1. Select two datasets from the UCI repository (<http://archive.ics.uci.edu/ml/datasets.html>) or use datasets of your own.
2. Determine how you will measure the quality of the clusters produced. Some reference on clustering evaluation can be found at <http://nlp.stanford.edu/IR-book/html/htmledition/evaluation-of-clustering-1.html>.
3. Select two algorithms for each dataset (e.g., K-means, canpopy or your own implementation) and compare their results using your quality metrics. Also you are encouraged to learn the performance impact of setting different k and varying the initial points and reporting the quality and runtime performance of Mahout K-means

4. Write a brief report to:

* Describe the datasets and your quality metrics.
* Describe your experiment setup such as how you preprocessed the data (if any), how you chose the parameters for the selected algorithms (if any), and why.
* Present the experiment results.  They should not be a simple copy-and-paste from Mahour/R/Weka output, but rather presented in a tabular or chart format for easy comparison.
* Discuss the insights and conclusions from your experiments.  For example, do different clustering methods make a difference in terms of quality or performance for the particular datasets you selected?  And why?  How data preprocessing might help?

5. Deliverable.

* Source code (if available)
* screen shots of your execution process.
* Runtime statistics in excel plots or tabular format.
* Report.