CS6220 BDS 2020 Fall

Homework Assignment 2

(Programming Category)

Student Name:______________________

Student Session: cs6220-A

You are given 3 types of programming problems. You only need to choose one of the 3 problems as your second homework. For the problem with multiple options, you are required to choose only one option. Feel free to choose any of your favorite programming languages, such as Java, C, Perl, Python, and so forth.

You are also allowed to choose a problem from HW1 programming as your HW2 programming assignment, if you wish to do so, as long as you have not done the problem in your HW1.

Post Date: Monday of Week 4
Due Date: midnight on Friday of Week 5

Problem 1. Hand on experience with Crowd Sourcing or Collaborative Filtering CF (Recommender Systems)

Option 1.1 Collaborative Filtering CF (Recommender Systems)

You are asked to download two user-user CF algorithms from the public domain, Here are two popular library for CF.

- **LightFM**: a hybrid recommendation algorithm in Python
- **Python-recsys**: a Python library for implementing a recommender system

You can also find CF algorithms and datasets from Mahout (Hadoop framework), or Spark ML library or Scikit Learn, Kaggle …

You are asked to find a public dataset such as Kaggle 1 Million Song dataset and run these two CF algorithms on three versions of the same dataset with at least three different sizes of the dataset, such as 1000, 10,000, 20,000. Compare the performance of the two CF algorithms on the three versions of the dataset.

Deliverable.
(1) URL of the code (algorithm)

(2) URL of the dataset
(3) 5-10 Examples of the datasets
(4) Describe your similarity algorithm and then present the similarity matrix for the 5-10 examples you give in (3)
(4) Screen shot of your CF runtime execution
(5) Accuracy analysis of the CF based on average over at least 10 queries.
(6) Elaborate on your learning experience.

Option 1.2 Understand Crowd Sourcing

1. You are asked to think of an example application scenario that can benefit from combining human intelligence with machine intelligence.

2. Develop example scenarios and document them in ppt file or word file to illustrate your ideas.

3. You are asked to create detailed workflow of how you would implement such system.

4. You are asked to also create application interface to show how your system would be used by an end user. You can use PPT to explain your idea.

5. The first 4 steps are conceptual learning and document your design in ppt file ideally with illustrative examples.

   Once your first part is done, you are asked to find a crowd sourcing service of your choice, such as Amazon Mechanical Turk (https://www.mturk.com). Study how a crowd sourcing task is submitted and how it is carried out by human as well as by machine in collaboration. You can screenshot your experience with this crowd sourcing service.

   Alternatively, you may also implement a scenario-based case study if you have time and wish to do so.

6. Deliverable: Each of the above five steps should be included in your HW2 package.

Option 1.3 Learning by Analyzing Popular Crowd Sourcing Services

7. You are asked to find at least two popular crowd sourcing services with the goal of making a comparative analysis of them and through this exercise to gain next level understanding of how to develop a novel

application support with the best combination of crowd intelligence with machine intelligence. Provide the URL for each.

8. The second task is to analyze each of the two crowd sourcing services in terms of strength and weakness.

   Hint: consider the trending videos in tiktok.com. Analyze how the service incorporate human intelligence with machine intelligence to provide fun short videos with good theme matching capabilities. Note that for fun dance move, tiktok provides the music for their visitors to upload the funny moves using the same music, minimizing the complex dance move video recognition learning to simpler machine matching. Similarly, each video upload will have hash tag (short text) to allow TikTok to label short video uploads for search and marching by keywords such as funny cats.

9. Develop comparison metrics to compare the two crowd sourcing services you analyzed in (2).

10. Provide a short summary of what you will do if you are asked to design a novel crowd sourcing service to beat the two you have analyzed. What to promote and what to avoid.

11. Deliverable: Each of the above four steps should be included in your HW2 package.

Problem 2. Hand on experience with Advanced Supervised Learning

Option 2.1 You are asked to train K+1 class classifiers, with K classes as regular k-class classification and 1 additional class as “unknown” class.

(1) The easiest is to start with 3-class classifier such as cat and dog and unknown. For any out of distribution images, such as flowers, people, cellphones, airplane, etc., they will be trained to be classified as unknow class.

(2) Make sure you have equal number of unknown class training examples as dog and cat. You may tune your unknow dataset to be more than dog or cat. The goal is to train a 3-classs classifier with high accuracy.

(3) Report your training measurement in terms of training time, accuracy, test time and accuracy and the trained model size.

(4) Extend K to larger value, such as k=11 (10+1), repeat the above and compare your experience with k=3 …

Deliverable: You are asked to report the accuracy and training/testing time comparison of the two trained models on at least one dataset in tabular format or excel plots. Your deliverable should include the performance measurement results, the experience with the installation, running and measurement, and your analysis and discussion. You can provide your deliverable in ppt or word or pdf.

Option 2.2 Accuracy Comparison of two deep learning frameworks

1. You are asked to download and install two deep learning frameworks: for example, TensorFlow, and PyTorch.

2. You are asked to use some benchmark datasets, such as MINIST, CIFAR10, CIFAR100, ImageNet1000Classes, to generate your training dataset and your testing (validation) dataset. You should choose at least 2000 images with 10 classes from the above datasets or your own preferred dataset with 8:2 or 9:1 ratio for training and testing.

3. You are asked to conduct accuracy comparison study by using at least three of the following six metrics:

   (i) The training dataset size;
   (ii) The neuron size;
   (iii) The number of weight filters;
   (iv) The number of mini-batches over the training dataset;
   (v)The number of training epochs;
   (vi) The # of convolutional layers;
   (vi) The accuracy of classification with varying # of hidden layers but the fixed # of training epochs.

   Note that you are asked to vary each metric (e.g., varying the number of epochs for at least 3~5 different settings on X-axis and measure/display accuracy measure on Y-axis.

4. **Deliverable**: You are asked to report the accuracy and training/testing time comparison of the two trained models on at least one dataset in tabular format or excel plots. Your deliverable should include the performance measurement results, the experience with the installation, running and measurement, and your analysis and discussion. You can provide your deliverable in ppt or word or pdf.

Problem 3. Learning Big Data Computing with Hadoop and/or Spark MapReduce

This problem has 3 options. You only need to choose one option. The last two options are designed for those students who are familiar with Hadoop MapReduce programming and would like to try something more challenging.

Deliverable (for any one of the three options):

(a) Source code and executable with readme. (provide URL if you use open source code packages)

(b) Screen Shots of your execution process/environments

(c) Input and output of your program

(d) Runtime measurements in excel plots or tabular format.

(e) Two scenarios that you choose to run the same MapRedue program to compare the performance or two programs solving the same problem with the same dataset.

(f) Provide your analytical report for your homework 2.

Hint:

a. If you use two or more datasets of different sizes, you report the scaling effect of your program.
b. If you use two different implementations of MapReduce programs running on the same dataset, you report the runtime performance optimization aspects of your two programs.

Option 1.
Suitable for students who are the beginner of Hadoop/Spark MapReduce

- Install HDFS and Hadoop MapReduce on your laptop.
- Run the word count map-reduce program, and report the runtime for two different sizes of datasets.
- Using MapReduce to solve another problem. You may choose one of the following or create a problem yourself.
  - o Consider a dataset of 20 files, print the top 100 words occurring in the most files
  - o Solving K-means clustering problem with a dataset given in Spark ML lib, or R, or Hadoop Mahout ML library, or Scikit-learn.

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

You are encouraged to learn by observing the runtime performance of Hadoop/Spark MapReduce program through different ways of programming the same problem and show their impact on the runtime performance of the MapReduce job.

Option 2:
Comparing Hadoop MapReduce with Spark MapReduce

- Install HDFS, Hadoop MapReduce and Spark MapRedue on your laptop.
- Run the word count map-reduce program on both Hadoop and Spark, and report the runtime comparison.

- Using MapReduce to solve another problem of your choice. You may choose one of the following or create a problem yourself.
  - Consider a dataset of 20 files, print the top 100 words occurring in the most files
  - Solving K-means clustering problem with a dataset given in Spark ML lib, or R, or Hadoop Mahout ML library, or Scikit-learn.

Run your MapReduce program on both Hadoop and Spark to compare the performance results.

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

You are encouraged to learn by observing the runtime performance of Hadoop/Spark MapReduce program through different ways of programming the same problem and show their impact on the runtime performance of the MapReduce job.

Option 3: Learning Graph Mining Algorithms using Spark or Hadoop MapReduce

Write a MapReduce program to solve a graph mining problem of your own interest. For example, you may use GraphX on Spark or Giraph or Hama on Hadoop to run PageRank and report the performance for iteration 1, 3, 5 and analyze your performance results.

Option 2~3 is designed for students who are familiar with or interested in learning both Hadoop MapReduce and Spark and interested in hand-on comparison of them through example data problems and configuration tuning.