IERG4300 / ESTR4300 Spring 2023
Homework 2

Release date: Feb 22, 2023
Due date: Mar 10, 2021 (Friday) 11:59 pm
No late homework will be accepted!

Every Student MUST include the following statement, together with his/her signature in the submitted homework.

I declare that the assignment submitted on the Elearning system is original except for source material explicitly acknowledged and that the same or related material has not been previously submitted for another course. I also acknowledge that I am aware of University policy and regulations on honesty in academic work, and of the disciplinary guidelines and procedures applicable to breaches of such policy and regulations, as contained in the website http://www.cuhk.edu.hk/policy/academichonesty/.

Signed (Student_________________________) Date:______________________________

Name_________________________________ SID_______________________________

Submission notice:

- Submit your homework via the elearning system

General homework policies:

A student may discuss the problems with others. However, the work a student turns in must be created COMPLETELY by oneself ALONE. A student may not share ANY written work or pictures, nor may one copy answers from any source other than one's own brain.

Each student **MUST LIST** on the homework paper the **name of every person he/she has discussed or worked with**. If the answer includes content from any other source, the student **MUST STATE THE SOURCE**. Failure to do so is cheating and will result in sanctions. Copying answers from someone else is cheating even if one lists their name(s) on the homework.

If there is information you need to solve a problem but the information is not stated in the problem, try to find the data somewhere. If you cannot find it, state what data you need, make a reasonable estimate of its value and justify any assumptions you make. You will be graded not only on whether your answer is correct but also on whether you have done an intelligent analysis.

Question 0 [20 marks]: Frequent Itemsets

Considering running the PCY algorithm to count frequent item pairs on a dataset with ***N*** baskets. Suppose each basket contains ***n*** items and there are ***d*** distinct item pairs amongst all of the baskets. Consider the following setup during the first pass of PCY: after keeping the counters for every singleton itemset observed during the first pass, we can still afford to store in main memory ***M*** integers, each of which will be used as a bucket. Assume further that ***d*** is much larger than the total number of buckets available, *i.e.*, ***d*** >> ***M***.

(a) [10 marks] What is the minimum support threshold ***s*** (in absolute number) we can allow if the average count for a bucket should be no more than 50% of the threshold ***s***? Please detail the steps to derive it.

(b) [10 marks] Suppose that A, B, C, D, E, and F are all the items under consideration. For a particular support threshold, the maximal frequent itemsets are {A, B, C} and {B, C, F}. What are all the other frequent itemsets?

Question 1 [80 marks + 20 Bonus marks]: Finding frequent itemsets

In this problem, we use the Yelp review dataset. The dataset[1] contains many user reviews, which are extracted from a website (Yelp) that publishes crowd-sourced reviews. The original dataset has been pre-processed as follows:

- Apply a sliding window of 40 words on each work. All the 40 words in one window make up a basket.
- Delete duplicate words in one basket, then filter out some common words
- You can download the pre-processed data from the following link: http://mobitec.ie.cuhk.edu.hk/ierg4300Spring2023/static_files/homework/yelp_review.rar
- Each line of this input is a space-separated list of words that corresponds to one basket.

The threshold for a frequent pair is defined as s=0.01. The frequency of a pair = *Occurrence of pair (i, j)* / *Total number of baskets*. For Q1(a), (b), (c), and (d), if the number of frequent pairs is more than 40, please only submit the **Top** 40 pairs (if any). Your results should consist of the frequent pairs and their corresponding count.

- You are allowed to use Linux command *sort* to post-process your results.

[1] Zhang, Xiang, Junbo Zhao, and Yann LeCun. "Character-level convolutional networks for text classification." In Advances in neural information processing systems, pp. 649-657. 2015.

(a) [20 marks] Implement the A-Priori algorithm to find frequent pairs on a single machine

Refer to the lecture slides, Page 30, 31 to implement the **A-Priori** algorithm. **You should not use MapReduce Framework for this sub-question.** You can run this job on one single AWS/ GoogleCloud machine or your PC. Note that dicvmc4.ie.cuhk.edu.hk is only a client for our DIC cluster. Please do NOT run the job on this machine.

(b) [30 marks] Implement the SON algorithm on MapReduce to find frequent pairs

Implement the SON algorithm under the MapReduce framework to find the frequent pairs. **Note that your code should be scalable.** In other words, your code should allow multiple mappers or reducers in both jobs. You need to implement two MapReduce jobs:

- The First MapReduce job should use **A-priori** algorithm to find the candidate pairs, which are frequent in at least one input file.
- The second MapReduce job counts only the candidate frequent pairs.

Tips:

- In the second MapReduce job, each mapper will load all the candidate pairs. You can pass them as a supplementary file.

Streamline and performance comparison:

- Wrap the two MapReduce rounds as a single executable by putting those commands you typed in a shell script.
- Compare the overall execution time of (a) and (b).
- Report the commands you used to submit the Hadoop job.

You can use the IE Data-Intensive Cluster (DIC) or any other Hadoop cluster (e.g., the AWS/GoogleCloud cluster built in HW#0) in various cloud computing platforms of your choice to do this problem.

(c) [30 marks] SON on MapReduce to find frequent triplets

The threshold for frequent triplets is defined as s=0.005. The frequency of a triplet (i, j, k) = *Occurrence of triplet (i, j, k)* / *Total number of baskets*. If the number of frequent triplets is more than 20, please only submit the top 20 triplets (if any).

Tip:

- In case of memory error, you may need to use multiple mappers/ reducers (*e.g.* 20+).

(d) [20 Bonus marks] Use the PCY algorithm to filter the candidate pairs in the SON algorithm

Implement the SON algorithm under the MapReduce framework. Use the PCY algorithm to filter the candidate pairs in the first MapReduce job. You can use the following Python hash function.

HashFunction=hash(word_1 + word_2) mod 100000

For example, the result of the word pair (‘Monday’, ‘Tuesday’) can be implemented as follows:

HashFunction=hash(‘Monday’ + ‘Tuesday’) % 100000

Streamline and performance comparison:

- Wrap the two MapReduce rounds as a single executable by putting those commands you type before in a shell script.
- Compare the overall execution time of (a), (b), and (d).
- Report the commands you used to submit the Hadoop job.

Part (d) is an optional (bonus) part for IERG4300 but is required for ESTR4300.

Submission requirement:

- You need to submit BOTH your code and your result. Please place the relevant code and the result in **a SINGLE PDF** file.