COMPSCI 753

Algorithms for Massive Data

Semester 2, 2020

Assignment 2: Data Stream Algorithms

Ninh Pham

Submission:

Please submit a single pdf file & the source code on CANVAS by **11:59pm, Sunday 13 September 2020**. The answer file must contain your studentID, UPI and name.

Penalty Dates:

The assignment will not be accepted after the last penalty date unless there are special circumstances (e.g., sickness with certificate). Penalties will be calculated as follows as a percentage of the mark for the assignment.

- **By 11:59pm, Sunday 13 September 2020 – No penalty**
- **By 11:59pm, Monday 14 September 2020 – 25% penalty**
- **By 11:59pm, Tuesday 15 September 2020 – 50% penalty**

1 Assignment problem (50 pts)

The assignment aims at investigating data stream algorithms, including Reservoir Sampling, Misra-Gries Summary, and CountMin Sketch on real-world data sets.

In the assignment, you write a program[1] to find *frequent items* on a stream. We use the same data set from Assignment 1, i.e. the ***KOS blog entries*** data set for this assignment.

The file has the format: $docID$ $wordID$ $count$, where $docID$ is the document ID, $wordID$ is the word ID in the vocabulary, and $count$ is the word frequency. Ignoring the first 3 lines, we consider each line as a stream tuple $(docID, wordID, count)$. In the assignment, we do not use the information of $count$. This means that you can think of $count = 1$ for each line. We want to find the most frequent words in our data set by our data stream algorithms.

Students are encouraged to run your implementations on larger data sets, such as *NYTimes news article* and *PubMed abstracts*.

The assignment tasks and its point are as follows.

1. **Execute bruteforce computation (10 pts):** Compute the frequency vector of all words, descendingly sort the words by their frequencies, and save the result into file (since you might use the bruteforce result for the next tasks). You need to report:
   (a) **The average frequency of the words in stream (5 pts).**
   (b) **Plot the sorted frequency of words to observe the skewed distribution (5 pts).**
2. **Reservoir Summary (10 pts):** Implement Reservoir Sampling to see the skewed distribution of our frequency vector. Fix the summary size $S = 10,000$, you need to:
   (a) **Estimate the frequency vector from our Reservoir Summary, and plot this estimate vector to see the approximation skewness (5 pts).**
   (b) **Run your Reservoir Sampling 5 times and report the average number of times the summary has been updated over these 5 runs (5 pts)**.
3. **Misra-Gries Summary (15 pts):** Implement Misra-Gries summary to find the most frequent words whose frequency is larger than 1,000. You need to:

[1] no restriction on programming languages used but preferred Python.

(a) **Explain the size of summary you choose such that you can find these frequent words (5 pts).**

(b) **Run your Misra-Gries summary and report the number of decrement steps with your chosen parameter (10 pts).**

4. **CountMin Sketch (15 pts):** Implement CountMin sketch to estimate the frequency of words. You need to:

   (a) **Explain the size of summary you choose such that the estimate error is at most 100 (5 pts)**.

   (b) **Run your CountMin Sketch with your chosen parameters, and report the estimate of the frequency of the words, whose frequency is larger than 1,000 found in the bruteforce algorithm (10 pts).**

2 What to submit?

An `answer.pdf` file reports the requested values and explanation of each task.

A `source code` file contains detailed comments.

Note: When taking the screenshots, make sure that you do not reveal any additional content you do not wish to share with us ;-).