[BsC Thesis | Semester Project]
Implementing Bayesian Data-Stream Analysis Algorithm

francesco.dadalt@inf.ethz.ch

February 2024

1 Context

Data-streams are ubiquitous in today's world. Nearly every information transfer from one location to a remote one can be described as a data-stream. In the case of computer networks in particular, we are interested in the dynamics of packets sent over a network. A security-related issue that arises in this context is the monitoring of data-rates of users in the network in order to ensure that no single user behaves unfairly and consumes more bandwidth than what it is allowed to. Thus, efficient algorithms that estimate the total volume sent by any user in a data-stream are an important building-block of any robust network system.

2 Problem

The primary reason as to why this problem is not trivial is that data-streams may contain millions of packets per seconds and thus the processing time of each packet is vanishingly small. Hence, any algorithm that estimates the data volumes of different users in the data-stream must be light weight both in memory footprint (due to fast memory being expensive) and processing time (in order to not cause packet delays and to keep up with the data-stream throughput). Enter sketches; Sketches are a class of algorithms that compress data-stream information down onto a small amount of memory and provide a lossy reconstruction method which returns estimates about some statistic of interest, which in our case it would be the total data volume sent over the stream by some user x. Various algorithms have been proposed in this regard, each with their own pros and cons. We developed our own [1] which is inspired by Bayesian probability theory. In the paper you will also find references to other sketching algorithms. As can be inferred from respective papers, these types of algorithms are of relevance to data-streams in general, not only network packet streams.

3 Your Task

Your task will be to create a high-performance implementation of [1] and to wrap it into a user-friendly library. The algorithm per se is simple so the primary focus lies on an efficient, documented, and clean implementation. One aspect of the algorithm, namely the computation of the prior, is not strictly specified by the algorithm, so you will be able to try out your own ideas. The first choice of language is C++. If that task is complete, we can continue on various avenues: In spirit of the first task, we can talk about implementations in further languages which would perhaps be Rust, Java or P4. Rust for safety reasons, Java for compatibility with database-oriented software such as Apache Flink, and P4 for deployment on network switches. The other option would be to use the library you created to develop more sophisticated systems and benchmark them (if done properly, we can work on wrapping it up in a paper).

4 Requirements

- Experience in C++ and multi-threading.
- Basic Understanding of numerical stability.
- Basic Understanding of probability theory.
- Opt: Rust or P4.

[1] Francesco Da Dalt, Simon Scherrer, and Adrian Perrig. "Bayesian Sketching for Volume Estimation in Data Streams". In: *Proceedings of the International Conference on Very Large Databases (VLDB)*. 2023. URL: `/publications/papers/dadalt_bayesiansketching_2022.pdf`.