An Improved Data Stream Summary: The Count-Min Sketch and its Applications

Graham Comrode, S. Muthukrishnan

This paper by Cormode and Muthukrishnan introduces a new sublinear space data structure which they titled the Count-Min Sketch (CM Sketch). The purpose of this data structure is to summarize data streams extremely fast and while demanding less storage, specifically approximating point queries, range queries, and inner product queries. The paper defines a data stream model as a sequence of updates modifying an implicit vector. CM Sketch is an array where updates are processed using pairwise independent hash functions and where queries return minimum over hash-mapped counters which ensures one-sided error. The paper attributes CM Sketch to the following applications: point queries, range queries, inner products, quantile estimation, and heavy hitters detection. This paper targets the scenario that data is streamed and that computations are done using sublinear space. Additionally, the authors assume bounded error is acceptable, which may not always be the case in practice. CM Sketch also assumes that the hash functions used are pairwise independent, meaning any two elements have uniform probability of collision, but there is limited discussion as to how these hash functions are designed. The applications addressed in the paper are all cases where data is sparse and where accumulations such as frequency counts are of primary interest. The authors argue there are savings in time and space complexity as well as the benefit of one-sided error (CM Sketch never underestimates) making it a better data structure in cases where false negatives are intolerable. Some limitations of CM Sketch are that it is not suitable for norm estimates like other data structures (Tug-of-War Sketch) and its dependence on carefully selected hash functions. Some extensions of CM Sketch could be multi-dimensional data streaming which would be common in applications in neural networks or computing convolutions. Another possible extension would be to make the hashing probabilistic, not uniform, improving accuracy for skewed distributions where not all elements should be treated equally. Overall, my opinion is that CM Sketch is a great improvement in the applications of algorithms in data streaming cases due to its reduced time and space complexity and one-sided error. It is a simple data structure making it practical for deployment and has a wide variety of applications. The design of the data structure is compelling and I judge it to be reproducible making the exploration of the extensions of this paper all the more realistic.

New Directions in Traffic Measurement and Accounting: Focusing on the Elephants, Ignoring the Mice

Cristian Estan, George Varghese

This paper by Estan and Varghese challenges how we should approach traffic measurement in networks. Namely, measurement should focus on large flows and ignore the small ones. The motivations of the paper are the issues with existing approaches' scalability, poor resource allocation, and the circumstance that a small number of flows carry a disproportionate amount of traffic (the elephants). The paper proposes two algorithms and argues that they reduce memory consumption, improve accuracy, and allow for scalable, real-time network traffic measurement. The authors assume SRAM should be prioritized over DRAM, a small number of flows dominate traffic, and that approximate traffic measurements with some error is tolerable. The authors argue that sampling-based approaches are inefficient, such as the existing algorithms like Cisco's Sampled NetFlow which randomly selects packets, leading to high error. They argue threshold accounting is more practical, meaning instead of tracking all flows, only those above a threshold should be counted, improving measurement accuracy and reducing the memory overhead simultaneously. The first algorithm they propose is sample and hold which has the con of false positives as some small flows might get counted if sampled early. Multistage filtering is the second algorithm they propose which reduces that risk and eliminates the risk of false negatives at the compromise of higher computational overhead and running slower than NetFlow. Therefore, depending on the application and the requirements for time complexity, multistage filters could be preferred. The authors analyzed their algorithms on real traffic traces, concluding that sample and hold provides 10 times more accurate measurements than NetFlow, multistage filters reduce false positives by 10 times, and that overall, threshold-based accounting scales much more efficiently than sample-based methods. The limitations of the paper is that both proposed algorithms require SRAM for efficient operation which can be costly or not always available, and sample and hold produces false positives which may not always be tolerated. Some useful extensions include dynamic threshold updating which could more intelligently follow fluctuations in traffic as well as perhaps an AI assistant that can classify traffic as elephants or mice for more effective threshold design. I think that this paper introduces really interesting approaches to solving a problem that I did not realize in a creative way. The motivation is very clear, the paper offers a very thorough theoretical foundation to the algorithms, applications are discussed well, and a real experiment was conducted as opposed to simply proposing the algorithms and promising improvements, making it much more compelling. The only weaknesses of the paper in my opinion are the algorithms' reliance on high-speed memory, and how especially with multistage filters, there is an increase in computational overhead, so these may not always be viable in compute-bound environments.