Towards High-Performance In-network Computing

Haoyu Li
The University of Texas at Austin

A wide range of network-intensive applications, including distributed large language model (LLM) training, high-frequency trading (HFT), and content delivery networks (CDNs), require a large amount of data movement and/or have a tight time budget. As network bandwidth continues to grow at approximately 70% per year, in-network computing is a promising paradigm for accelerating applications by performing computations directly on heterogeneous programmable hardware such as field-programmable gate arrays (FPGAs) and application-specific integrated circuits (ASICs).

While such programmable devices can be orders of magnitude faster than CPUs, a key challenge is to implement various functions within limited resources and strict hardware constraints. For example, although Intel's Tofino programmable switching ASICs support packet processing speeds of up to 12.8 terabits per second (Tbps), their capabilities are limited by their only several hundred megabytes (MB) of memory and a limited packet processing pipeline that supports only simple calculations. Therefore, designing and analyzing algorithms within such limitations of programmable hardware is a challenging and increasingly important topic in the systems and networking community, and corresponding theories are valuable for addressing database, machine learning, data mining, and security problems. My research focuses on develping algorithms and theory under strict constraints, which become key building blocks for in-network computing systems.

In particular, I have designed in-network computing algorithms for membership queries (ChainedFilter [1], *SIGMOD 2024*), frequency estimation (StingySketch [2], *VLDB 2022*), and collective communication (Homomorphic Compression [3], preprint) problems with solid theoretical guarantees and system evaluations. All of these works address fundamental aspects of network functions - filtering, network measurement, and in-network aggregation. They are based on increasingly complex mathematical definitions: starting from a collection of key-value pairs, where the values are binary, the membership query algorithm seeks to identify these pairs with minimal type I (false positive) error. As the range of values expands to all integers and the values change dynamically, the focus shifts to frequency estimation. As the algorithm continues to demand zero-error performance, it evolves into a homomorphic compression algorithm.

1 ChainedFilter: Compact In-network Membership Query

Given a collection of key-value pairs, where each value is either 0 or 1, a membership algorithm aims to accurately identify these pairs, allowing for Type I (false positive) errors. The algorithm should correctly report a value of one when the key value is one, and can report one with a false positive rate (ϵ) even when the key value is zero. Such a membership algorithm plays an important role in networking, databases, and security. For example, routers and switches use Bloom filters to classify, forward, and drop network packets; LSM tree-based storage engines use learned filters to speed up K-V storage; bitcoin miners use invertible Bloom lookup tables (IBLTs) to reduce the amount of information needed for block propagation and reconciliation. However, although membership algorithms have been studied for over fifty years, the space lower bound for general membership problems was unknown.

To address this, I developed a new data structure and algorithm called ChainedFilter, which provides the space lower bound and gives a surprising theoretical result of membership. Let's first consider a scenario with n mappings to one and λn mappings to zero. We can express the space lower bound for this problem as $nf(\epsilon, \lambda) + o(n)$. It's easy to see that the problem can be decomposed into two sub-problems: First, storing the n one mapping and the λn zero mappings with a false positive rate of $\epsilon' \in [\epsilon, 1]$, and second, storing the n one mappings and the $\epsilon'\lambda n$ remaining false positive zero mappings with a false positive rate of ϵ/ϵ'. This leads to the inequality $f(\epsilon, \lambda) \leq f(\epsilon', \lambda) + f(\epsilon/\epsilon', \epsilon'\lambda)$. The main contribution of ChainedFilter is that the proof of

$$f(\epsilon, \lambda) = f(\epsilon', \lambda) + f(\epsilon/\epsilon', \epsilon'\lambda),$$

via information theory, indicating that the decomposition process described above involves zero information loss. *This discovery allows me to establish for the first time a complete space lower bound for general membership problems, i.e., $f(\epsilon, \lambda) = f(0, \lambda) - f(0, \epsilon\lambda)$, after the expressions $f(0, \lambda)$ and $f(\epsilon, +\infty)$ were known in 1978*. This theory shows that by combining two sub-algorithms, an effective membership algorithm can be developed.

Both theoretical and experimental results show that this technique significantly improves the performance of many fundamental applications, including static dictionaries, lossless data compression, cuckoo hashing, learned filters, and LSM trees in RocksDB.

2 StingySketch: Fast and Accurate Network Measurement

Since membership algorithms only consider binary values, StingySketch extends the value range to all integers and supports dynamic value updating to measure network traffic, which is also the basis for finding top k items in NetCache, joining tables in databases, and multiset queries in data mining. The literature shows that sketches are the most promising probabilistic algorithms in data streams because of their compact space and $O(1)$ time. However, in practice, existing sketches are unable to efficiently balance accuracy and speed for common highly skewed data distributions.

In this work, I delve deeper into optimizing the algorithm by considering two key factors: data distribution and memory access locality with a novel carry-in (overflow) mechanism based on in-order traversal of the binary tree. Experimental results show that my technique achieves up to 50% more accuracy than the state-of-the-art for accuracy-oriented algorithms and up to 33% more throughput than the SOTA for speed-oriented algorithms.

3 Homomorphic Compression: Asymptotic Optimal In-network Aggregation

While StingySketch serves as a fast and accurate method for dimensionality reduction, in its capacity as a data compression algorithm, it introduces a small loss of accuracy that can limit its applications. A case in point is distributed deep neural network (DNN) training, where using sketches to compress gradients in different worker nodes reduces communication overhead but may negatively impact model accuracy.

To address this challenge, my ongoing work on the Homomorphic Compression algorithm represents a significant advance. It ingeniously blends concepts from membership, sketching, and peeling theories to achieve zero loss. Using a strategy similar to ChainedFilter, our work achieves asymptotically optimal lossless compression ratio and computational complexity, high parallelism, and excellent locality. These properties are maintained over arbitrary data types and sparsity levels.

To demonstrate its effectiveness, I integrated my gradient aggregation algorithm into the most popular NVIDIA Collective Communications Library (NCCL) and the ATP in-network aggregation framework in PyTorch and conducted experiments on two distributed systems. I compared the distributed training acceleration for different models: VGG19 with the Cifar10 dataset, LSTM with the GBW dataset, and BERT with the Wikipedia dataset. Our evaluation shows that it improves the aggregation throughput by up to 6.33× and achieves a 3.74× speedup in per-iteration training speed in distributed training tasks.

4 Conclusion and Future Work

My theoretical background allows me to tackle fundamental system problems with new insights and approaches. During my Ph.D., I plan to build high-throughput, low-latency systems for important applications such as LLM training and configurable networks and explore new networking technologies, including SmartNIC, Compute Express Link (CXL) storage and optical networking, and turn them into impactful real-world systems.

References

[1] Haoyu Li, Liuhui Wang, Qizhi Chen, Jianan Ji, Yuhan Wu, Yikai Zhao, Tong Yang, and Aditya Akella. Chainedfilter: Combining membership filters by chain rule. *Proc. ACM Manag. Data*, 1(4), dec 2023.

[2] Haoyu Li, Qizhi Chen, Yixin Zhang, Tong Yang, and Bin Cui. Stingy sketch: A sketch framework for accurate and fast frequency estimation. *Proc. VLDB Endow.*, 15(7):1426–1438, mar 2022.

[3] Haoyu Li, Yuchen Xu, Jiayi Chen, Rohit Dwivedula, Wenfei Wu, Keqiang He, Aditya Akella, and Daehyeok Kim. Accelerating distributed deep learning using lossless homomorphic compression. In submission to ICML 2024.