Science | DOI:10.1145/1721654.1721661

Alex Wright

Data Streaming 2.0

In today's real-time Web, data streaming applications no longer have the luxury of making multiple passes over a recorded data set.

As the rise of the real-time Web continues to fuel dizzying growth in the volume of packets moving around the global network, computer scientists are taking a fresh look at an old research problem: data streaming.

Traditionally, data streaming applications have worked by capturing data as it travels by and then storing it for later analysis, allowing developers to create algorithms that make multiple passes through a data set, using tried and tested methods like clustering, classification, and association mining.

With today's Internet, however, it is all but impossible to capture and store every passing bit. "Moore's law is not keeping up with our desire to record anything that can be recorded," notes Sudipto Guha, an associate professor of computer science at the University of Pennsylvania. Modern Web-focused applications like network monitoring, click log analysis, and fraud detection have created massive streams of live data where analysis must be carried out immediately.

In this real-time Web, data streaming applications no longer have the luxury of making multiple passes over a recorded data set. Instead, researchers have developed new, more efficient single-pass algorithms that are starting to approach the sophistication of their multiple-pass predecessors. As the research landscape evolves, some developers are also starting to explore whether these algorithms have applications beyond the traditional realm of data streaming.

PHOTOGRAPH BY OHAD BEN-YOSEPH

Research into data streaming algorithms received an important boost at the turn of the millennium, when the sudden increase of denial-of-service attacks in the wake of the Year 2000 problem switchover prompted network software developers to investigate new approaches to ensure the integrity of data on the network. At about the same time, the research community became interested in the problem as well, inspiring pioneering efforts like AT&T's Gigascope Internet monitoring project.

"In high-speed networks, you have only nanoseconds to handle a packet before the next one arrives," says Graham Cormode, a researcher at AT&T Labs. When a single Internet service provider (ISP) router may handle gigabytes of headers every hour, it's simply not practical to store data—even metadata—for later analysis. For example, if an ISP wants to determine the 90th percentile of Internet Protocol (IP) packet sizes over an IP packet stream, then traditional multiple-pass algorithms will inevitably fall short.

The huge volume of incoming data, coupled with a growing emphasis on real-time computation, has given researchers the impetus to explore new computation models that involve taking a single pass over the data. In this model, algorithms must make smart use of system resources to monitor a passing data stream.

Limited Memory Allocation

Perhaps the greatest challenge for single-pass algorithms involves making smart use of memory allocation. "There's always a limited amount of memory," says Nick Koudas, a faculty member in the computer science department at the University of Toronto, "and you can only look once." Further complications arise from the inevitable memory limitations involved in storing intermediate results, summaries, samples, synopses, or sketches that can be used to provide the final query answers.

To address these constraints, recent data-streaming research has focused on developing techniques based on sampling, sketches, and other methods to provide approximate answers. Many of these approaches can now give bounds on the approximation error, at times deterministic, and many times probabilistic. These bounds are typically a function of the memory space available to the algorithms. Thus, error analysis can offer developers an elegant way of making tradeoffs between accuracy and conserving additional memory.

In recent years, researchers have developed a number of algorithms for approximately answering queries and performing analysis like top-k elements, number of distinct elements, and quantiles. These algorithms are able to provide guaranteed error bounds on the returned answers, letting developers make intelligent tradeoffs between the amount of memory used and accuracy within the single-pass framework.

Modern single-pass algorithms are becoming sufficiently sophisticated so that they can now, in some cases, approach the performance of multiple pass algorithms. "Perhaps the big surprise of streaming is how much is possible with only one look at the data," says Cormode. "We can approximate the entropy, find how many distinct items there are, or identify the most frequently occurring items." These tasks are relatively straightforward with a traditional multiple-pass approach, but accomplishing the same objectives with a single-pass architecture—while managing system costs—calls for more sophisticated techniques. "The cost of these algorithms depends on how accurately the answer is approximated," Cormode says. "It turns out that we can get accurate answers with only kilobytes of storage."

The Best of Both Approaches

While developers have traditionally faced a binary choice in deciding whether to implement single-pass vs. multiple-pass algorithms, researchers have recently started to explore techniques that mix the best of both approaches. For example, Cormode is interested in exploring how to mix linear passes with a partial indexing or reordering of data. "I don't think the debate over how best to model this kind of computation is settled yet," he says.

Single-pass algorithms might have applications beyond the traditional realm of data streaming.

As the networking community continues to push the boundaries of these algorithms for traditional data-streaming applications, some researchers are also starting to explore how these approaches could be leveraged into other emerging research arenas. The University of Pennsylvania's Guha sees potential applications in the realm of specialized hardware, such as like IBM's cell processor and streaming graphics processing unit computation and ternary content-addressable memory systems. He is also interested in exploring the communication between different parts of a stream in a distributed network.

Some researchers are also starting to explore whether the principles of data streaming algorithms could shed light on the world of social networking, where users are collectively generating a rapidly expanding data stream of text updates and other social graph activity.

While online social networks seem to bear a surface-level resemblance to data networks, those similarities only stretch so far. Networking relies on highly structured data, whereas the social Web consists largely of unstructured text and an unpredictable hodgepodge of media objects like photos, videos, and audio streams. Working with such a diverse data set presents conceptual challenges for researchers accustomed to addressing the relatively cut-and-dried problems of data streaming. "With streams of unstructured text, friend links, and media postings, the problems are much less clean to define," says Cormode. "Indeed, defining what to compute is a big part of the problem!"

To mine the social Web effectively, algorithms may need to evolve to incorporate techniques from natural language processing and statistical modeling of temporal trends. Koudas and his team at the University of Toronto are currently exploring approaches to reconstructing chains of events from multiple sources across the Web by sorting out threads, subtopics, and other data points to analyze the evolution of stories in the news using data streaming principles. "The real-time Web is becoming a reality," says Koudas. "Now the data point is a stream of text rather than a packet."

As data streaming algorithms start to expand beyond the network layer and into the world of social media, will they evolve into new applications that help map the unpredictable terrain of online friendships, assorted jokes, and cute kitten photos? Will new streaming algorithms take shape to help make sense of the messy, unstructured facts of daily life? Perhaps if there's one thing that data streaming algorithms can teach us, it is this: Data comes and goes, but the stream never ends. C

Further Reading

Zhang, J.
A survey on streaming algorithms for massive graphs. *Managing and Mining Graph Data*, Aggarwal, C.C. and Wang, H. (eds.), Springer, New York, 2010.

Aggarwal, C.C. (Ed.)
***Data Streams: Models and Algorithms.* Springer, New York, 2007.**

Mathioudakis, M., Koudas, N., and Marbach, P.
Early online identification of attention gathering items in social media. *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, New York, Feb. 2010.

Cormode, G. and Hadjieleftheriou, M.
Finding the frequent items in streams of data. *Comm. of the ACM 52*, 10, October 2009.

Alex Wright is a writer and information architect who lives and works in Brooklyn, NY. Rajeev Rastogi, vice president and head of Yahoo! Labs Bangalore, contributed to the development of this article.