Massive Data Streams Research: Where to Go

S. Muthukrishnan
Rutgers Univ

March 7, 2010

Summary. Computing power has been growing steadily, just as communication rate and memory size. Simultaneously our ability to create data has been growing phenomenally and therefore the need to analyze it. We now have examples of massive data streams that are

- created in far higher rate than we can capture and store in memory economically,
- gathered in far more quantity than can be transported to central databases without overwhelming the communication infrastructure, and
- arrives far faster than we can compute with them in a sophisticated way.

This phenomenon has challenged how we store, communicate and compute with data. Theories developed over past 50 years have relied on full capture, storage and communication of data. Instead, what we need for managing modern massive data streams are new methods built around working with less. The past 10 years have seen new theories emerge in computing (data stream algorithms), communication (compressed sensing), databases (data stream management systems) and other areas to address the challenges of massive data streams. Still, lot remains open and new applications of massive data streams have emerged recently. We presents an overview of these challenges.

Background. We start with a quintessential application that generates massive data streams.

- *IP Traffic Analysis.* Consider the data generated from IP networks at the level of each packet and its contents: each new packet is seen at a router and forwarded in a few nanoseconds and has 100's of bytes of header and payload information. Capturing all this information will need memory working at nanosecond response which is expensive, transporting them to central warehouses would need lot of communication infrastructure because of duplication in observation, and doing any analysis in the few nanoseconds before another packet arrives will limit computation to simplest tasks.

There are other examples from sensor networks to Homeland Security, scientific discovery and others where one or the other of the computing, communication and storage infrastructure becomes the challenge because of the massiveness of the data.

This phenomenon has challenged how we store, communicate and compute with data. Traditional systems capture all the data they can, communicate all the data between storages, and compute sophisticated functions on the stored data when needed. This in turn saw over half a century of theory:

- Theory of Computing led to the notion of efficiency to be polynomial time algorithms based on the premise that all the data is stored and can be processed multiple times.
- Theory of Communication led to the notion of information content and the minimum number of bits needed to transfer data in entirety.
- Theory of Signal Processing developed under the premise that signal can always be sampled at the Nyquist rate for full reconstruction.
- Theory of Databases led to an algebra to process data that can be applied recursively to intermediate results and are provably correct on the state of the stored input and intermediate data.

The assumption that all data can be captured, moved and processed pervades all of computing, communication, data acquisition and ananlysis we have seen the past half century. Instead, in the new world of massive data streams, we need to ask what is the minimal amount of data that needs to be captured and stored, what is the minimal number of bits that need to be transferred for suitable analyses we need to do, what are algorithms that will support sophisticated analyses on whatever portion of data is available, and finally, what are suitable redesign of applications to work with this new world of storage, communication and computation.

The research community is beginning to address the above fundamental questions. In particular, they have developed *theory of streaming computation* that uses few passes over the data with only polylogarithmic storage and per-item processing time [1, 2]; database community has developed *data stream management systems* for dealing with operators for which data sources never end [3]; there are *specialized applications* for IP network analysis [4], financial stream, etc.; and, the signal processing community has developed *theory of compressed sensing* [5, 6] where signals can be sampled at sub-Nyquist rate for sparse signals. Altogther, these developments are already impacting Telecoms and have potential applications in Medical Imaging, Radar and many others [7]. Still, this world is nascent, comparable to early days of computing. Our ability to deal with massive data is still primitive despite the developments in the last 10 years in directions above. Massive data is real, and the last few years have convinced everyone from Scientists to Engineers, Industries and Governments that analyzing them is needed for security, operations and innovation.

Directions in Massive Data Streams Research. We describe some of the challenges research community needs to address to deal with massive data streams. We will use two new massive data applications to motivate our discussion.

- *Web Data Analysis.* The web is an information publication network that amasses text to conversations, friendships, video and images, growing in terabytes a day. Portions of the web can be gathered, stored and analyzed via traditional crawl based systems or as updates when information changes. Billions of dollar of Industry as well as telecom and Homeland Security application rely on analyzing such web information [9]. Such analysis can not be carried out easily with single machines, or traditional streaming systems.

- *Continual Security Monitoring.* Homeland Security and other security applications use not only on-demand query analyses for forming hypotheses, discovering evidence and finding connections, but need to monitor many data sources and sensors in a continual way, looking for pre-specified or emergence of patterns, and generate alerts and root causes, so possibly humans can investigate them. This requirement is not met by systems that do periodic analyses of all data because it is difficult to analyze all the data at sufficient time granularity to be useful for real time alerting. On the other hand, for many analyses we do not yet know how to compile them down to monitoring primitives on distributed sensors or sources so they can automatically trigger alerts as early as possible as threatening phenomenon emerges.

Inspired by these applications, we present a series of research directions.

- *Models of Massive Distributed Data.* While MapReduce [8] and its variants are being used for simple analyses of web data with massively distributed systems, often analysts in the field tend to change problems they need to solve into ones they can solve easily using known techniques and compromising on the end analysis. We need suitable new models to design and analyze algorithms for massively distributed systems going beyond streaming and MapReduce and they have to model energy concerns of using 1000's of processors. We need a theory to provide us new algorithmic design techniques as well as provide a framework to understand what are truly hard problems. We need notions of complexity classes beyond the traditional NC that captures the true tradeoff between communication, energy and computation in massively distributed systems, and will guide this area much as polynomial time computations did for 50+ years and polylog one-pass models have done for the past 10+ years. Preliminary work appears in [10, 11].

- *Continual Computation Theory.* For monitoring applications, we need a new framework of algorithm design that rather than quantifying the cost of solving a problem on the whole data in one shot, quantifies the incremental overhead of a computation as data is progressively seen. The crux here is to

also model the cost of communicating the updates from distributed points. Recently, some progress has been made in defining such models [12], but a far deeper understanding and richer theory is needed. In particular, the underlying challenges are in developing a new, continual eversion of the well known Communication Complexity [13], and a time-varying extension of the well-known Slepian-Wolf results to distributed networks for specific analyses of interest.

- *Graph and Matrix Data Analyses Problems.* The applications above need new kinds of analyses. In particular, web data has a variety of rich graph structures in them. The analyses of interest involves computing graph properties, finding substructures, learning properties of nodes and edges, as well as looking at them as suitable matrices and computing fundamental quantities like rank approximations, eigenvalues etc. While many of these problems have been studied in standard settings, scaling these solutions to applications above — massively distributed or continual monitoring — is a great challenge. The past few years have seen some progress on graph algorithms [16] and matrix methods [15], but far more is needed.

- *Stochastic Data Algorithms.* The applications above have uncertain data because of inaccuracies in sensing or malicious data from web. A large class of tasks with massive data is machine learning including classification, clustering, and labeling. Finally, nearly all problems of interest are hard in distributed or continual settings above in standard worst case models. Therefore we need principled methods for learning the stochastics and uncertainties of data and designing algorithms that tune to them, and a new theory of Stochastic Data Algorithms. There is some recent work in machine learning community on incremental machine over stochastic data, recent work in CS and OR community on stochastic online optimization and space-efficient tracking of sufficient statistics, emerging work in database community on processing uncertain data, etc. We need a far richer research program to truly have a suite of algorithms suitable for dealing with stochastics of massive data applications.

- *Cryptography and Privacy of Massive Data.* Data needs encryption, authentication and privacy, no matter what size. But the motivating applications for massive data including web data and security analsys bring many novel challenges to even traditional tasks. For example, well known Interactive proofs have to be expanded in novel ways for applications to stream verification [17]; privacy of data stream analyses needs new stringent notions including pan-privacy [18]; and, continual computation needs new differential privacy [19]. We need a systematic research effort to extend and invent new cryptographic and privacy primitives for data access and computation for massive data applications such as the ones above.

- *Compressed Functional Sensing.* Streaming and compressed sensing brought two groups of researchers (CS and signal processing) together on common problems of what is the minimal amount of data to be sensed or captured or stored, so data sources can be reconstructed, at least approximately. This has been a productive development for research with fundamental insights into geometry of high dimensional spaces as well as the Uncertainty Principle. In addition, Engineering and Industry has been impacted significantly with analog to information paradigm. This is however just the beginning. We need to extend compressed sensing to functional sensing, where we sense only what is appropriate to compute different function (rather than simply reconstructing) and furthermore, extend the theory to massively distributed and continual framework to be truly useful for new massive data applications above.

Conclusions. The directions above are substantial and require research across not only theoretical and applied areas of Computer Science, but also in Applied Mathematics (eg signal analysis) and beyond in Engineering (eg from telecoms to medical imaging, astronomy, Radar and more). The fledging research of the past 10 years on massive data streams has already generated new insights in Mathematics, new theories of Computing and Communication, as well as new Systems for data analysis; it has also impacted industries of size in billions of dollars. New focused research agenda bringing these communities together on directions above will impact much more, and define fundamentally new ways software, hardware, analog and digital systems will be built in the future.

References

[1] S. Muthukrishnan. *Data Streams: Algorithms and Applications*. In Foundations and Trends in Theoretical Computer Science, 2005.

[2] P. Indyk A tutorial on Streaming, Sketching and Sub-linear Space Algorithms. 2009 Information Theory and Applications Workshop, San Diego, 2009. `http://people.csail.mit.edu/indyk/ita-web.pdf`

[3] M. Garofalakis, J. Gehrke and R. Rastogi. *Data Stream Management: Processing High-Speed Data Streams*, 2007.

[4] C. Cranor, T. Johnson and O. Spatscheck. Gigascope: How to monitor network traffic at 5Gbit/sec at a time. `http://www2.research.att.com/~divesh/meetings/mpds2003/schedule/spatscheck.pdf`.

[5] David Donoho. Compressed sensing. *Technical Report*, 2004.

[6] E. Candes and T. Tao. Near-optimal signal recovery from random projections and universal encoding strategies. 2004.

[7] `http://dsp.rice.edu/cs`.

[8] J. Dean and S. Ghemawat. Mapreduce: Simplified data processing on large clusters. *Proc. OSDI*, 2004.

[9] `http://en.wikipedia.org/wiki/XLDB`.

[10] Jon Feldman, S. Muthukrishnan, Anastasios Sidiropoulos, Clifford Stein, Zoya Svitkina. On distributing symmetric streaming computations. em Proc. SODA 2008: 710-719.

[11] H. Karloff, S. Suri, S. and S. Vassilvitskii. A Model of Computation for MapReduce. Proc. ACM-SIAM SODA 2010.

[12] Graham Cormode, S. Muthukrishnan, Ke Yi. Algorithms for distributed functional monitoring. *Proc. SODA* 2008: 1076-1085

[13] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*, 1997.

[14] `http://www.scholarpedia.org/article/Slepian-Wolf_coding`

[15] Kenneth L. Clarkson, David P. Woodruff. Numerical linear algebra in the streaming model. *Proc STOC.* 2009: 205-214.

[16] Kook Jin Ahn, Sudipto Guha. Graph Sparsification in the Semi-streaming Model. *ICALP* (2) 2009: 328-338.

[17] A. Chakrabarti, G. Cormode, and A. McGregor. Annotations in data streams. In International Colloquium on Automata, Languages and Programming (ICALP), 2009.

[18] C. Dwork, M. Naor, T. Pitassi, G. Rothblum, and S. Yekhanin. Pan-Private Streaming Algorithms. ICS, 2010.

[19] Cynthia Dwork. Differential Privacy in New Settings. *SODA*, 2010.