Mining Data Streams

Tamraparni Dasu
AT&T Labs, USA

Gary Weiss
Fordham University, USA

INTRODUCTION

When a space shuttle takes off, tiny sensors measure thousands of data points every fraction of a second, pertaining to a variety of attributes like temperature, acceleration, pressure and velocity. A data gathering server at a networking company receives terabytes of data a day from various network elements like routers, reporting on traffic throughput, CPU usage, machine loads and performance. Each of these is an example of a data stream. Many applications of data streams arise naturally in industry (networking, e-commerce) and scientific fields (meteorology, rocketry).

Data streams pose three unique challenges that make them interesting from a data mining perspective.

1. **Size:** The number of measurements as well as the number of attributes (variables) is very large. For instance, an IP network has thousands of elements each of which collects data every few seconds on multiple attributes like traffic, load, resource availability, topography, configuration and connections.
2. **Rate of accumulation:** The data arrives very rapidly, like "water from a fire hydrant". Data storage and analysis techniques need to keep up with the data to avoid insurmountable backlogs.
3. **Data transience:** We get to see the raw data points at most once since the volumes of the raw data are too high to store or access.

BACKGROUND

Data streams are a predominant form of information today, arising in areas and applications ranging from telecommunications, meteorology and sensor networks, to the monitoring and support of e-commerce sites. Data streams pose unique analytical, statistical and computing challenges that are just beginning to be addressed. In this chapter we give an overview of the analysis and monitoring of data streams and discuss the analytical and computing challenges posed by the unique constraints associated with data streams.

There are a wide variety of analytical problems associated with mining and monitoring data streams, such as:

1. Data reduction,
2. Characterizing constantly changing distributions and detecting changes in these distributions,
3. Identifying outliers, tracking rare events and anomalies,
4. "Correlating" multiple data streams,
5. Building predictive models,
6. Clustering and classifying data streams, and
7. Visualization.

As innovative applications in on-demand entertainment, gaming and other areas evolve, new forms of data streams emerge, each posing new and complex challenges.

MAIN FOCUS

The data mining community has been active in developing a framework for the analysis of data streams. Research is focused primarily in the field of computer science, with an emphasis on computational and database issues. Henzinger, Raghavan & Rajagopalan (1998) discuss the computing framework for maintaining aggregates from data using a limited number of passes. Domingos & Hulten (2001) formalize the challenges, desiderata and research issues for mining data streams. Collection of rudimentary statistics for data streams is addressed in Zhu & Sasha (2002) and Babcock, Datar, Matwani & O'Callaghan (2003). Clustering (Aggarwal, Han, Wang & Yu, 2003), classification, association rules (Charikar, Chen & Farach-

Colton, 2002) and other data mining algorithms have been considered and adapted for data streams.

Correlating multiple data streams is an important aspect of mining data streams. Guha, Gunopulous & Koudas (2003) have proposed the use of singular value decomposition (SVD) approaches (suitably modified to scale to the data) for computing correlations between multiple data streams.

A good overview and introduction to data stream algorithms and applications from a database perspective is found in Muthukrishnan (2003). Aggarwal (2007) has a comprehensive collection of work in the computer science field on data streams. In a similar vein, Gaber (2006) maintains a frequently updated website of research literature and researchers in data streams.

However, there is not much work in the statistical analysis of data streams. Statistical comparison of signatures of telecommunication users was used by Cortes & Pregibon (2001) to mine large streams of call detail data for fraud detection and identifying social communities in a telephone network. Papers on change detection in data streams (Ben-David, Gehrke & Kifer, 2004; Dasu, Krishnan, Venkatasubramanian & Yi, 2006) use statistical approaches of varying sophistication. An important underpinning of statistical approaches to data mining is density estimation, particularly histogram based approaches. Scott (1992) provides a comprehensive statistical approach to density estimation, with recent updates included in Scott & Sain (2004). A tutorial by Urbanek & Dasu (2007) sets down a statistical framework for the rigorous analysis of data streams with emphasis on case studies and applications. Dasu, Koutsofios & Wright (2007) discuss application of statistical analysis to an e-commerce data stream. Gao, Fan, Han & Yu (2007) address the issue of estimating posterior probabilities in data streams with skewed distributions.

Visualization of data streams is particularly challenging, from the three perspectives dimensionality, scale and time. Wong, Foote, Adams, Cowley & Thomas (2003) present methods based on multi dimensional scaling. Urbanek & Dasu (2007) present a discussion of viable visualization techniques for data streams in their tutorial.

Data Quality and Data Streams

Data streams tend to be dynamic and inherently noisy due to the fast changing conditions.

An important but little discussed concern with data streams is the quality of the data. Problems could and do arise at every stage of the process.

Data Gathering: Most data streams are generated automatically. For instance, a router sends information about packets at varying levels of detail. Similarly an intrusion detection system (IDS) automatically generates an alarm on a network when a predefined rule or condition is met. The data streams change when the rule settings are changed either intentionally by an operator or due to some software glitch. In either case, there is no documentation of the change to alert the analyst that the data stream is no longer *consistent* and *can not be interpreted* using the existing data definitions. Software and hardware components fail on occasion leading to gaps in the data streams (*missing or incomplete data*).

Data Summarization: Due to the huge size and rapid accumulation, data streams are usually summarized for storage -- for instance using 5-minute aggregates of number of packets, average CPU usage, and number of events of a certain type in the system logs. However, the average CPU usage might not reflect abnormal spikes. Or, a rare but catastrophic event might be unnoticed among all the other types of alarms. The trade-off between data granularity and aggregation is an important one. There has been much interest in representing data streams using histograms and other distributional summaries (Guha, Koudas & Shim, 2001) but largely for univariate data streams. Options for multivariate data streams and the use of sufficient statistics (Moore, 2006) for building regression type models for data streams are explored in Dasu, Koutsofios & Wright (2007).

Data Integration: Creating a comprehensive data set from multiple data sources always poses challenges. Sometimes there are no well defined join keys – only soft keys like names and addresses that can be represented in many different ways. For example, "J. Smith", "John Smith" and "John F. Smith" might be different variations of the same entity. Disambiguation is not easy. One data source might contain only a fraction of the entities contained in the other data sources, leading to gaps in the data matrix. Data streams pose additional complexities such as synchronization of multiple streams. There are two ways the temporal aspect could be a problem. First, if the clocks that timestamp the data streams are out of step and second, if the aggregation granularity does not allow the two data streams to be synchronized in any meaningful fashion.