INSIGHT: A Distributed Monitoring System for Tracking Continuous Queries

Navendu Jain, Praveen Yalagandula, Mike Dahlin, and Yin Zhang
Department of Computer Sciences
University of Texas at Austin
Austin, TX, 78712
{nav, ypraveen, dahlin, yzhang}@cs.utexas.edu

1. EXTENDED ABSTRACT

A distributed monitoring framework can serve as an important building block for constructing large-scale *data aggregation* and *continuous event monitoring* applications, such as IP traffic monitoring (DDoS attacks), network anomaly detection (Internet worms), accounting and bandwidth provisioning (hot spots, flash crowds), sensor monitoring and control, and grid resource monitoring. At the core of these applications is a distributed query engine that aggregates information and performs continuous tracking of queries over collections of physically-distributed and rapidly-updating data streams. The underlying aim is to provide a *global view* of information in the system at a reasonable cost and within a specified precision bound. To achieve this objective, a distributed monitoring system should (a) scale to a large number of streams and query attributes, (b) incur minimal communication overhead for aggregating query results, (c) be time responsive for quickly identifying anomalies, and (d) be able to bound the inaccuracy of the computed value for the aggregate function.

Consider the example of a continuous query workload in the context of IP traffic monitoring where a network administrator wants to query for *heavy hitters* i.e., *monitor all flows that account for a significant fraction (say* 0.1*%) of the volume of ongoing traffic.* A centralized monitoring approach of logging the entire data at a single coordinator would either generate monitoring traffic whose volume is proportional to the total traffic, or be too late in detecting network anomalies (e.g., DDoS attacks), both of which are clearly unacceptable. A key insight to address these challenges is that the traffic monitoring applications do not require accurate answers; approximate answers bounded by an absolute numerical error suffice. Our solution uses this insight to provide the infrastructure for building large-scale distributed monitoring systems that efficiently aggregate and quickly react to changing network activity.

Previous efforts in the networking and database community have often focused on centralized one-shot queries on data streams and therefore do not address the problems of continuous tracking, scalability, and communication cost in a distributed environment. Some recent works have proposed methods based on exploiting tradeoff between precision and communication cost in a distributed setting [1]. These techniques, however, assume either a single-level communication hierarchy, periodic aggregation of data, or event-triggered response to a one-shot query. Thus, they are not applicable to general communication hierarchies or lack sufficient time responsiveness in a distributed environment.

In this work, we present INSIGHT, a scalable and time responsive monitoring system that tracks continuous queries and efficiently gathers local information about data streams into an aggregate view. Our system is based on three key ideas: First, we provide control over the tradeoff between precision and communication cost in a multi-level communication hierarchy. Specifically, the root of the hierarchical aggregation tree divides the tolerable error bounds of a query answer among all its children using an adaptive strategy based on their *stream update rate* and *load* (and so on down the tree). Second, we exploit *temporal stability* to cull out updates sent by a node that are within error bounds of previously cached values at its parent. Third, we leverage Distributed Hash Tables (DHT) to construct multiple aggregation trees that map different attributes to different trees for load balancing. We are building INSIGHT on top of SDIMS [2], a scalable distributed information management system. SDIMS provides a a hierarchical aggregation framework that allows applications to access detailed views of nearby information and summary views of global information.

We have applied INSIGHT for detecting distributed heavy hitters. Through experiments on two real-world traces, we observe that INSIGHT achieves up to two orders of magnitude reduction in communication costs compared to the algorithm by Manjhi et. al [1], and at the same time achieves near real-time detection of global heavy hitters with modest communication overhead. More information on INSIGHT is available at `http://www.cs.utexas.edu/users/nav/INSIGHT`.

SOSP'05, October 23–26, 2005, Brighton, United Kingdom.

2. REFERENCES

[1] Amit Manjhi, Vladislav Shkapenyuk, Kedar Dhamdhere, and Christopher Olston. Finding (Recently) Frequent Items in Distributed Data Streams. In *ICDE*, pages 767–778. IEEE Computer Society, 2005.

[2] Praveen Yalagandula and Michael Dahlin. SDIMS: A Scalable Distributed Information Management System. In *SIGCOMM*, pages 379–390. ACM, 2004.