Data & Knowledge Engineering 68 (2009) 509–530

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Privately detecting bursts in streaming, distributed time series data

Lisa Singh [a,*], Mehmet Sayal [b]

[a] Department of Computer Science, Georgetown University, 37th and 'O' Streets, NW, St. Mary's – 3rd Floor, Washington, DC 20057, United States
[b] Hewlett Packard Company, Hewlett-Packard Labs, 1501 Page Mill Road, Palo Alto, CA 94304, United States

ARTICLE INFO

Article history:
Received 10 January 2008
Received in revised form 25 November 2008
Accepted 11 December 2008
Available online 4 January 2009

Keywords:
Privacy preservation
Burst detection
Streaming data

ABSTRACT

Surprisingly, privacy preservation in the context of streaming data has received limited attention from computer scientists. In this paper, we consider privacy preservation in the context of independently owned, distributed data streams. Specifically, we want to protect the privacy of each individual participant's data stream while identifying bursts that exist across participant streams. We define two types of privacy breaches, data breaches and envelope breaches. In order to protect individual data, each participant transforms large subsets of the stream into small vectors that approximate the stream. These small vectors are calculated by summing coefficients of wavelet transforms at different resolutions. The participants share their vectors using bursty, self-eliminating noise. The combined participant vectors can then be used to detect bursts. We find that our approach leads to accurate burst detection results with reduced communication costs. We demonstrate these findings using both real and synthetic data.

1. Introduction

Streaming time series data is prevalent in many domains including financial, medical, network and military applications. We are interested in identifying peaks and valleys (bi-directional bursts) across data streams owned by independent parties. We have two constraints: communication cost and privacy. Because the data is distributed and streaming, it is costly to communicate all the raw data necessary to find bursts. Second, because of privacy concerns, the parties cannot share the raw data with other parties. Given these constraints, our goal is to accurately identify bursts across participant data with as little communication and as much privacy as possible.

Examples of distributed time series applications include customer sales, e.g. credit card transactions across independent entities, trade surveillance for security fraud across independent exchanges, network packet data of different companies, and military tactical surveillance of troops or equipment. These parties may be interested in finding bursts from a combined data set, but have privacy concerns or legal constraints that prevent the sharing of raw data. In order to find these 'aggregate bursts', the data needs to be combined by sharing a perturbed version of the original data. We refer to this general problem as *privacy-preserving aggregate burst detection* [36].

Let's look at a detailed example. Credit card companies maintain a network that their retail partners and banks have access to. Suppose multiple credit card companies wanted to identify peaks and valleys in credit card activity at the customer level, the merchant level, or even in a specific geographical region level. Questions of interest may include: Is there a large spike in customer purchases across different bank cards? Can we verify fraudulent behavior by identifying bursts across the credit, retail and banking network? Because each company maintains its data independently, the data must be merged to detect bursts across these independent entities. Then these companies will have insight into burst behavior of individual

* Corresponding author. Tel.: +1 202 687 9253; fax: +1 202 687 1835.
E-mail address: singh@cs.georgetown.edu (L. Singh).

0169-023X/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.datak.2008.12.003

customers, individual merchants, individual banks, etc., during specific periods of time. It also allows these credit card companies to take the information and see how the burst behavior of their individual data compares to the aggregate burst behavior. However, because of legal privacy regulations, sharing the raw data stream may not be an option. In those cases, bursts need to be detected using an approximation of the raw data.

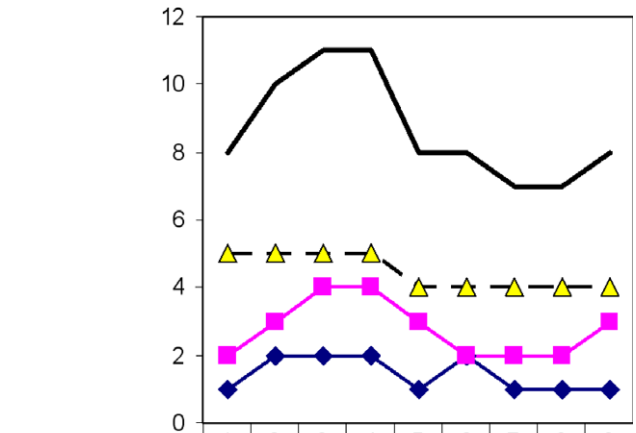
Fig. 1 shows a small example with three participants. Each participant (P1, P2, P3) has a time series containing nine time points of data. Without sharing their raw individual data, the participants are interested in determining whether or not a burst exists in the combined data. In Fig. 1, the top time series is the combined sum or 'aggregated' data. A burst can be seen in the aggregate time series beginning at time point 2 and ending at time point 5. In this figure, we show the data values of each participant and the aggregate time series so the summation of participant data is clear.

Privacy preservation in the context of streaming data has received limited attention [15]. Work by Li et al. uses random perturbations to hide raw data while maintaining correlation information. Our work differs in two critical ways. First, we are focusing on distributed streams that need to be merged. Second, random perturbation as presented in [15] is not an option because of the affect on burst detection accuracy and the associated overhead of communicating the entire perturbed data stream in a distributed environment. Instead, our goal is to find a condensed representation from which participants can identify the aggregate bursts without compromising any participant's original data.

In this paper, we introduce a new approach that involves computing the wavelet transform of each data stream window and then calculating a summation vector that is then combined with data from other participants. The individual participant summation vector is a condensed approximation of a segment of the original stream from which the original stream values are difficult to reconstruct. After data is combined, bursts of varying width within a window are detected directly from an aggregated 'distance' vector. Our online privacy-preserving burst detection algorithm obscures raw data values and the shape or *envelope* of the time series, hides bursts of individual participants, reduces communication costs to logarithmic with respect to the original data size, and detects bursts accurately. This approach is new and has not been introduced in previous literature.

The contributions of this paper are as follows. First, we consider burst detection in the context of a streaming environment with limits on processing time, communication cost, and buffer space. Next, we formalize two types of privacy breaches, including a breach that has not been previously studied in this context (envelope breach), and quantify the discrepancy between the original stream, the adversary's approximate stream, and the participants' individual vectors (stream approximations). We also analyze privacy limits, bounding the level of privacy preserved at different points in the algorithm. Third, we extend the analysis of our previous work for the streaming environment and an envelope breach. Fourth, we introduce a new algorithm that identifies bursts accurately using a small summation vector of potentially large segments of the data stream. Our data aggregation strategy is also a new contribution; here, we use bursty, self-eliminating noise to obscure the bursts of individual participants. Previous work on similar time series problems use a round robin protocol for secure summation – a costly operation in a streaming environment [31,36].

The remainder of this paper is as follows. We begin by reviewing some related literature in Section 2. We then formalize the problem and notation in Section 3. Section 4 details the components of the algorithm for burst detection and studies privacy preservation within its context. In Section 5, we empirically demonstrate the approach using both real and synthetic data sets. Finally, conclusions and future directions are presented in Section 6.



	1	2	3	4	5	6	7	8	9
P1	1	2	2	2	1	2	1	1	1
P2	2	3	4	4	3	2	2	2	3
P3	5	5	5	5	4	4	4	4	4
Aggregate	8	10	11	11	8	8	7	7	8

Fig. 1. Example showing bursts of three participants and the associated aggregate burst.