

2017

Trend Filtering in Network Time Series with Applications to Traffic Incident Detection

Pranamesh Chakraborty
Iowa State University

Chinmay Hegde
Iowa State University, chinmay@iastate.edu

Anuj Sharma
Iowa State University, anuj@iastate.edu

Follow this and additional works at: https://lib.dr.iastate.edu/ccee_conf



Part of the [Transportation Engineering Commons](#)

Recommended Citation

Chakraborty, Pranamesh; Hegde, Chinmay; and Sharma, Anuj, "Trend Filtering in Network Time Series with Applications to Traffic Incident Detection" (2017). *Civil, Construction and Environmental Engineering Conference Presentations and Proceedings*. 113.

https://lib.dr.iastate.edu/ccee_conf/113

This Conference Proceeding is brought to you for free and open access by the Civil, Construction and Environmental Engineering at Iowa State University Digital Repository. It has been accepted for inclusion in Civil, Construction and Environmental Engineering Conference Presentations and Proceedings by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.

Trend Filtering in Network Time Series with Applications to Traffic Incident Detection

Abstract

Motivation. Traffic congestion on roadways has been identified by the US Department of Transportation as one of “the largest threats to the economic prosperity of the nation”. [5]. A common contributor to congestion include traffic incidents. Detection congestion events and resolving incidents is of paramount importance: research has shown that early incident detection resulted in reduction of 143.3 million man-hours and savings of \$3.06 million in 2007 [6]. Fortunately, data collected by a variety of sensors positioned along highways can serve as important early indicators of traffic incidents. Each of these sensors continuously acquire and store multiple time series corresponding to (average) vehicle speed, (average) road occupancy, weather conditions, and so on [1, 2]. These sensors can be naturally modeled as nodes of a graph (which reflects the topology of the road network), and the data corpus can be modeled as a multi-dimensional time series. Quickly identifying anomalies in this corpus enables rapid incident detection and timely response.

Disciplines

Transportation Engineering

Comments

This proceeding is published as Chakraborty, Pranamesh, Chinmay Hegde, and Anuj Sharma. "Trend filtering in network time series with applications to traffic incident detection." In *Time Series Workshop, 31st Conference on Neural Information Processing Systems (NIPS)*. 2017. Posted with permission.

Trend Filtering in Network Time Series with Applications to Traffic Incident Detection

Pranamesh Chakraborty
Iowa State University
pranames@iastate.edu

Chinmay Hegde
Iowa State University
chinmay@iastate.edu

Anuj Sharma
Iowa State University
anujs@iastate.edu

1 Introduction

Motivation. Traffic congestion on roadways has been identified by the US Department of Transportation as one of “the largest threats to the economic prosperity of the nation”. [5]. A common contributor to congestion include traffic incidents. Detection congestion events and resolving incidents is of paramount importance: research has shown that early incident detection resulted in reduction of 143.3 million man-hours and savings of \$3.06 million in 2007 [6].

Fortunately, data collected by a variety of sensors positioned along highways can serve as important early indicators of traffic incidents. Each of these sensors continuously acquire and store multiple time series corresponding to (average) vehicle speed, (average) road occupancy, weather conditions, and so on [1, 2]. These sensors can be naturally modeled as nodes of a graph (which reflects the topology of the road network), and the data corpus can be modeled as a multi-dimensional time series. Quickly identifying anomalies in this corpus enables rapid incident detection and timely response.

Unfortunately, however, two major challenges arise:

1. The dynamics of traffic patterns are somewhat hard to model. Certain intersections are naturally congested at certain times of the day (even though no incident has occurred). Therefore, separating false positives (caused by recurring congestion events) from true positives (actual traffic incidents) can be confounding.
2. Moreover, the scale of the data poses a major challenge. Parsing the data *even once* involves a considerable amount of computation, and well beyond the reach of standard methods. Moreover, new data keeps streaming in continuously so need to develop real-time outlier detection strategies.

Our primary aim in this short paper is to propose and implement a new framework for *trend filtering* in time series defined over graphs. Our framework is extendable to very large datasets comprised of time series over networks, and combines existing massively parallelizable methods with a new method of multivariate network denoising.

Our secondary aim is to showcase this application to the machine learning community, and highlight several basic challenges arising in analysis of massive data streams. Exposing the special challenges encountered in such datasets could spur discussion of *real-time* anomaly detection techniques in non-stationary streaming datasets over graphs. We make a *call to action*: how to translate the wealth of knowledge in graph-based analysis into pattern analysis problems involving massive time series?

Our Contributions. We build upon our earlier work (reported in [4]), where we built a parallelizable, *univariate* framework for streaming anomaly detection in network time series. In this paper, we develop and test an approach to extend it to a more refined *multivariate* analysis that leverages the topology of the underlying graph. At a high level, our approach works in two stages:

1. As a quick dimensionality reduction step, we construct a coarse “heat map” of the raw data by computing robust univariate statistics of the time series recorded at each node in the network (including median, maximum absolute deviation, and inter-quartile range). The calculation of these statistics can be massively parallelized using MapReduce.

2. We post-process the heat map using *graph-based trend filtering*. This post-processing step incorporates both the topology of the underlying network as well as temporal correlations in the time series. Jointly incorporating both types of correlations enables us to filter out recurring congestion events from true incidents from the observed time series.

The resultant heat-map gives us a simple criterion which helps us quickly visualize traffic patterns. We validate this method on large-scale streaming datasets acquired from the Iowa DoT.

Techniques. As mentioned above, each time series is exceptionally large-scale (with millions of data points recorded daily per sensor). Therefore, dimensionality reduction is crucial. However, standard moment-based methods do not work very well since they are susceptible to outliers. Therefore, we resort to computing robust *summary* statistics of each observed time series across non-overlapping time windows. However, this preliminary computation ignores spatiotemporal correlations, and the resultant summaries can be rather noisy. Our new trend-filtering approach leverages the fact that the summary statistics values in topologically adjacent regions are likely to be similar, and we pose an optimization problem (basically, a graph-based generalization of anisotropic total variation) to obtain filtered estimates.

It remains to show how to do this efficiently. To this end, we leverage the proximal forward-backward approach of [3]. This enables us to obtain the final post-processed output in time (essentially) linear in the number of nodes in the network times the number of time windows under consideration. Our implementations in Python testing on INRIX sensor data [1] show that we can get very favorable results. See Section 3 below for details.

2 Algorithm Details

Setup. We describe a mathematical abstraction of our observed data corpus. Let $G = (V, E, W)$ denote a (weighted) graph with nodes $V = \{v_i\}_{i=1}^n$ and undirected edges $\{e_i\}_{i=1}^m$. In our application, a node corresponds to a sensor (which measures vehicle speed in a given road segment), and nodes corresponding to consecutive sensors along the highway are connected via appropriately weighted edges. We assume complete knowledge about the connectivities and edge weights. For each node v , on a given day (say d), we measure a (noisy) time series of length- N :

$$X_v^d = (X_v^{t_1,d}, X_v^{t_2,d}, \dots, X_v^{t_N,d}), \quad v \in V.$$

The super-script (t_i, d) denotes the t_i^{th} time instant on the d^{th} day of the week. The observed time series across different nodes are *synchronized*, i.e., the time stamps t_1, t_2, \dots, t_N are the same across sensors for all days. Overall, we get a third-order *time series tensor* $X \in \mathbb{R}_+^{n \times N \times D}$, where D corresponds to different days. Our goal is to reliably identify anomalous local patterns in this tensor.

A major difficulty in this setup is that there exist *two* types of anomalies: one type that arises due to measurement noise, and the second that arises due to recurring events. A suitable anomaly detection algorithm would keep, while retaining any phenomena that corresponds to typical recurring events.

Analogous methods for anomaly detection in structured datasets have been extensively studied in the machine learning literature in the matrix case; cf. robust low-rank matrix decomposition methods. While similar methods exist in the tensor case exist, two distinguishing phenomena (challenges) arise in our case. We highlight and address each challenge as follows.

Preprocessing. The first challenge is that the sampling rate of our sensors is very high, so the length of each observed time series can range in the millions. This necessitates some processing along the second dimension of our tensor.

Following [4], we model the (univariate) statistics of each time series within a given time window as a Laplace distribution with location parameter μ and scale parameter b . Therefore, as a quick dimension reduction step, we can replace each time window with the maximum likelihood estimates of μ and b , which are respectively the sample median and maximum absolute deviation (MAD)¹.

Denote the estimates of μ and b within each window to be \bar{x} and $\bar{\zeta}$ respectively. Then, our pre-processed dataset now consists of two (smaller) tensors \bar{X} and $\bar{\zeta}$ respectively, and the rest of our

¹In our earlier work [4], we have also experimented with using the inter-quartile distance (IQD) instead of the MAD, and found that this gives somewhat better results.

computations will solely involve these quantities. We also calculate an auxiliary tensor called the *threshold* (τ), defined as:

$$\tau_v^{p,d} = \bar{X}_v^{p,d} - 3 \times \zeta_v^{p,d}. \quad (1)$$

This threshold represents the lower end of the “three-sigma” band indicating standard operating behavior; any time window in which the estimated median falls below the threshold represents a possible anomaly. However, this represents only a coarse indicator of anomalous behavior, and we show how to refine this estimate next.

Trend Filtering for Network Time Series. The second distinguishing phenomenon is that the coordinates along the first dimension of our tensor correspond to nodes of a graph. Therefore, instead of performing standard tensor decomposition, we instead define a special denoising procedure that leverages the topology of the underlying graph, as well as the temporal coherence of each time series. To the best of our knowledge, this definition is novel, although it follows from a fairly straightforward extension of existing methods for trend filtering over graphs [?].

For a given time series tensor $X := \{X_v^d\} \in \mathbb{R}^{n \times N \times D}$, define the *temporal graph difference operator*, Δ in such a way that its weighted ℓ_1 -norm yields the sum of local spatio-temporal (absolute) differences:

$$\|\Delta X\|_\rho = \sum_{d=2}^D \left(\sum_{(u,v) \in E} w_{u,v}^{-1} |X_u^d - X_u^{d-1}| + \gamma |X_v^d - X_v^{d-1}| \right), \quad (2)$$

where $\gamma > 0$ is a scalar weighting factor. For the special case where G corresponds to the chain graph with unit weights, the above norm $\|\cdot\|_\rho$ is nothing but the (anisotropic) 3D total-variation norm. Using this norm, we can now define a *trend-filtered estimate* of the time series tensor as the solution to the optimization problem:

$$\hat{X} = \underset{Y \in \mathbb{R}_+^{n \times N \times D}}{\operatorname{argmin}} \|Y - X\|_2^2 + \lambda \|\Delta X\|_\rho. \quad (3)$$

We can thus obtain trend-filtered estimates of tensors representing each of the desired statistical quantities of interest: median speed μ , deviation ζ , and threshold parameter τ .

Algorithm. To alleviate computational complexity issues, we adapt the proximal gradient method of [3] to solve regularized problems of the form (3). The high level idea is to use a forward-backward approach which alternately takes a gradient step (with respect to the first term in (3)), followed by a proximity operator applied with respect to the norm $\|\cdot\|_\rho$. The prox-operator can be computed via a “taut string” approach by unpacking the structure of the dual of the optimization problem representing the proximal step. The worst case complexity of each step is quadratic in the total tensor size, but empirically it is known to exhibit linear running time (and we have confirmed this in experiments).

3 Results

We test our method using probe vehicle speed data from 1st August, 2016 to 21st October, 2016 of the Des Moines region. The geographical area comprised Interstates 35, 80, and 235 covering 164 miles and were divided into 254 segments. The length of the segments vary from 0.2 miles to 1.5 miles. For comparison we also include univariate median filtering results previously reported in [4].

Threshold maps (encoding the parameter τ) are created for each road direction, day of the week (d), and week number (w). Each threshold map is generated using the previous 8 weeks as training data, following [4]. Figure 1 shows a sample threshold map (I-235 EB, Thursday, Week number 40) created from raw thresholds, the denoised threshold map created using our trend filtering approach and median filtering. It is clear that the denoising algorithms help in smoothing incorrect thresholds generated due to sensor errors or anomalies such as incidents. The denoising algorithms take advantage of the spatio-temporal aspect of the thresholds generated across adjacent segments.

We can also perform trend filtering by stacking together multiple threshold maps corresponding to the same day of the week across different weeks. Figure 2 shows an example of 3D denoising performed using Total Variation and Median Filter. Threshold maps of I-235 EB for four consecutive Thursdays were stacked together and the resultant trend filtered estimates are shown in the figure. As highlighted in the figure, although the evening peak congestion is appearing consistently from Week 39 to 41,

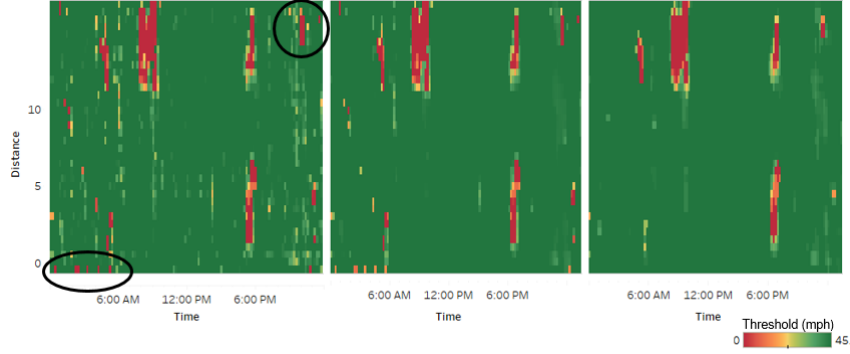


Figure 1: Left: Raw threshold map. Middle: trend filtering $w/\lambda = 5$. Right: median filtering.

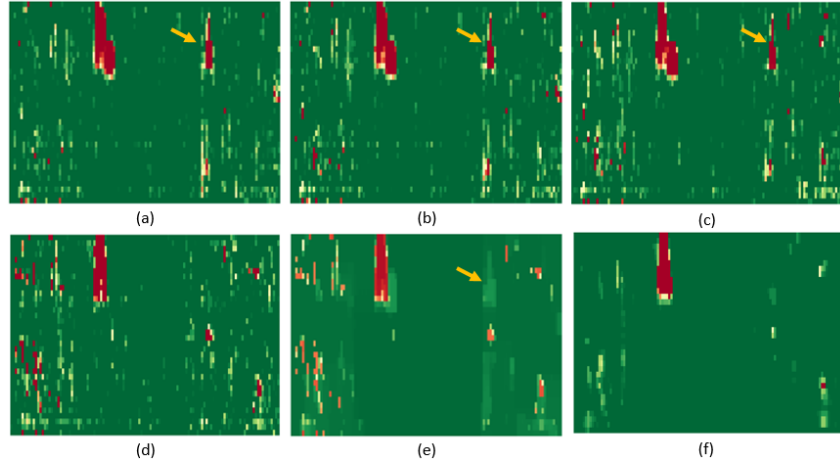


Figure 2: (a) - (d): Week 39 - 42 Original threshold maps. (e): 3D trend filtering for Week 42. (f): 3D median filter denoising of Week 42.

it is missing in Week 42 thresholds (possibly due to sensor errors in the newly added week data). However, our estimate helps retain the feature to some extent in the newest week (i.e., Week 42).

Trend-filtered threshold maps exhibit several benefits. They help identify regions of recurring congestion and explain important recent trends of traffic patterns. For example, in Figure 1, the highlighted congested region in top right (due to a night time workzone) got featured in the denoised threshold. However, the intermittent congestion in the bottom left (plausibly due to sensor errors) were removed in the denoised images. Thus, denoising the threshold maps helps in better visualization, detecting more explainable features, and with higher confidence.

Acknowledgments. Our research results are based upon work jointly supported by the National Science Foundation Partnerships for Innovation: Building Innovation Capacity (PFI: BIC) program under Grant No. 1632116, National Science Foundation under Grant No. CNS-1464279 and Iowa DOT Office of Traffic Operations Support Grant. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

- [1] INRIX. <http://inrix.com/>, accessed July 20, 2016.
- [2] Wavetronix. <http://www.wavetronix.com/en>, accessed July 20, 2016.
- [3] Álvaro Barbero and Suvrit Sra. Modular proximal optimization for multidimensional total-variation regularization. *arXiv preprint arXiv:1411.0589*, 2014.
- [4] Pranamesh Chakraborty, Jacob Robert Hess, Anuj Sharma, and Skylar Knickerbocker. Outlier mining based traffic incident detection using big data analytics. Technical report, 2017.
- [5] N. Owens, A. Armstrong, P. Sullivan, C. Mitchell, D. Newton, R. Brewster, and T. Trego. Traffic Incident Management Handbook. Technical report, U.S. Department of Transportation, 2010.
- [6] David L Schrank and Timothy J Lomax. The 2007 urban mobility report. Technical report, Texas Transportation Institute, The Texas A&M University System, 2007.