Communication-Efficient Online Detection of Network-Wide Anomalies

Ling Huang*  XuanLong Nguyen*
Minos Garofalakis§  Joe Hellerstein*
Michael Jordan*  Anthony Joseph*  Nina Taft§

*UC Berkeley  § Intel Research

Towards Decentralized Detection

- Today: Distributed Monitoring & Centralized Computation
  - Stream-based data collection
  - Periodically evaluate detection function over collected data
  - Doesn't scale well in network size or timescale
- Our contribution: Decentralized Detection
  - Continuously evaluate detection function in a decetr. way
  - Low-overhead, rapid response, accurate and scalable
  - Detection accuracy controllable by a “tuning knob”
    - Provable guarantees on detection error (false alarm rate)
    - Flexible tradeoff between overhead and accuracy
Detection Problems in Enterprise Network

For efficient and scalable detection, push data processing to the edge of network!

Detection of Network-wide Anomalies

- A **volume anomaly** is a sudden change in an Origin-Destination flow (i.e., point to point traffic)
- Given link traffic measurements, **detect** the volume anomalies
An Illustration

The Subspace Method (Lakhina’04)

- An approach to separate normal from anomalous traffic based on Principal Component Analysis (PCA)
- **Normal Subspace** \( S \): space spanned by the top \( k \) principal components
- **Anomalous Subspace** \( \tilde{S} \): space spanned by the remaining components
- Then, decompose traffic on all links by projecting onto \( S \) and \( \tilde{S} \) to obtain:

\[ y = y_{no} + y_{ab} \]

Traffic vector of all links at a particular point in time

Normal traffic vector

Residual traffic vector
Detection Illustration

Value of $\|y\|^2$ over time (all traffic)

Value of $\|C_{ob}y\|^2$ over time

Red dots: anomalies  Blue curve: traffic data

The Centralized Algorithm

- Data matrix $Dat$
  1) Each link produces a column of $m$ data over time.
- $Dat_{(i)}$ at each
- $Q_\alpha$
- Threshold
- $Q_\alpha$
- PCA on $Dat$

- Doesn’t scale well to large network or to smaller timescales
  - The number of monitoring devices may grow to thousands
  - The anomalies may occur on second or sub-second time scales

The Network

Operation center

Eigen vectors

Eigen values

$Dat = \begin{bmatrix} 1 & 3 & 5 \\ n (nodeID) & & \end{bmatrix}$
Our In-Network Detection Framework

**Distr. Monitors**
- Original monitored time series
- Filtered data

**Coordinator**
- PCA-Based Detection
- Adjust Filter Parameters
- Perturbation Analysis
- User inputs: detection error

**The Protocol At Monitors**
- Monito
- \(\delta_1, \cdots, \delta_n\)
- Mod. \(t^*\) built or
- e.g., t locally
- Simple but enough to achieve 10x data reduction
The Communication and Error Tradeoff

\[ \| \hat{C}_{ab} \hat{y} \|^2 > \hat{Q}_\alpha \]

Full Info.

\[ \| C_{ab} y \|^2 > Q_\alpha \]

Approximate Info.

\[ \text{Filtered data}(t) \]

The coordinator computes a set of good \( \delta_1, \ldots, \delta_n \) to manage this difference.

Parameter Design and Error Control (I)

- Users specify an upper bound on false alarm rate, then we determine the filtering parameters \( \delta \)'s

\[ \| C_{ab} y \|^2 > Q_\alpha \text{ vs. } \| \hat{C}_{ab} \hat{y} \|^2 > \hat{Q}_\alpha \]

\[ \text{Data vs. Model} \]

Detection error

\[ \epsilon \]

Monte Carlo and fast binary search

Stochastic Matrix Perturbation Theory

Error propagation

Parameter design

Eigen error: \( L_2 \) norm of the difference between the approximate eigenvalues and the actual ones
Parameter Design and Error Control (II)

- Detection Error $\mu \rightarrow$ Eigen-Error $\epsilon$
  - Mont Caro simulation to find the mapping from $\epsilon$ to $\mu$
  - For the given $\mu$, using fast binary search to find an $\epsilon$
- Eigen-Error $\epsilon \rightarrow$ Filtering parameters $\delta$’s
  $$2\sqrt{\frac{\lambda}{m} \sum_{i=1}^{n} \frac{\delta_i^2}{3} + \left(\frac{1}{m} + \frac{1}{n}\right) \sum_{i=1}^{n} \frac{\delta_i^4}{9}} = \epsilon$$

Evaluation

- Given user-specified false alarm rate, evaluate the actual detection accuracy and communication overhead
- Experiment setup
  - Abilene backbone network data
  - Traffic matrices of size 1008 X 41
  - Set uniform slack $\delta_i = \delta$ for all monitors
Performance

<table>
<thead>
<tr>
<th>μ</th>
<th>Missed Detections</th>
<th>False Alarms</th>
<th>Data Reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Week 1</td>
<td>Week 2</td>
<td>Week 1</td>
</tr>
<tr>
<td>0.01</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.03</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0.06</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

error tolerance = upper bound on error

Data Used: Abilene traffic matrix, 2 weeks, 41 links.

Summary

- A communication-efficient framework that
  - detects anomalies at desired accuracy level
  - with minimal communication cost
- A distributed protocol for data processing
  - local monitors decide when to update data to coordinator
  - coordinator makes global decision and feedback to monitors
- An algorithmic framework to guide the tradeoff between communication overhead and detection accuracy
Questions

Reference

Backup Slides
Traditional Distributed Monitoring

- Large-scale network monitoring and detection systems
  - Distributed and collaborative monitoring boxes
  - Continuously generating time series data
- Existing research focuses on data streaming
  - *Centrally* collect, store and aggregate network state
  - Well suited to answering approximate queries and continuously recording system state
  - Incur high overhead!

Our Distributed Processing Approach

- A coordinator
  - Is aggregation, correlation and detection center
- A set of distributed monitors
  - Each produces a time series signals
  - Processes data locally, only sends needed info. to coordinator
  - No communication among monitors
  - *Coordinator tells monitors the level of accuracy for signal updates*
Principal Component Analysis (PCA)

Anomalous traffic usually results in a large value of $y_{ab}$

Principal components are top eigenvectors of covariance matrix. They form the subspace projection matrices $C_{no}$ and $C_{ab}$

$y_{no} = C_{no}y$

$y_{ab} = C_{ab}y$