MetricForensics: A Multi-Level Approach for Mining Volatile Graphs

Keith Henderson†
Leman Akoglu‡
B. Aditya Prakash‡
Tina Eliassi-Rad†
Lei Li‡
Hanghang Tong‡
Christos Faloutsos‡
Koji Maruhashi§

† Lawrence Livermore National Laboratory
‡ Carnegie Mellon University
§ Fujitsu Laboratories Ltd.

Work performed under the auspices of U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344
Large volatile graphs present novel analytical problems

- Communication and sensor networks generate rapidly-changing network data

IP Traffic

Call Logs

Social Networking
Requirements for mining large volatile graphs

- **Effectiveness**
  - Compromises in volatile graphs (e.g. IP traffic) often need to be detected immediately

- **Scalability**
  - Graph size and rapid changes require algorithms that can scale (linear on the measures of interest)

- **Flexibility & Generality**
  - The approach should be able to incorporate new tools and modules as they become available

- Found **no** approach that satisfies **all** of these requirements for mining large volatile graphs
Roadmap

- Problem definition
- Overview of MetricForensics
- Models and methods
  - Data model
  - Metrics
  - Analysis techniques
- Experimental results
- Conclusions
Problem definition

- Two tasks
  1. Given a stream of edges in the following form:
     \(<srcNode, dstNode, start\, Time, duration>\)
     detect interesting events in real-time or near real-time
  2. Attribute these events to individual nodes or groups of nodes that are behaving strangely

- Our hypothesis
  - Changes in behavior at the vertex-level can be identified at the graph-level by global metrics

- General idea
  - Calculate computationally burdensome (slow) metrics only when a period of interest is identified by faster metrics
**MetricForensics overview**

- **Multi-level approach provides scalability**
  - Fast global metrics used in real-time as edges come and go
  - Periods of interest are analyzed further offline

- **Metrics and analysis techniques are customizable at each level**
  - Modules can be added or removed as needed
Roadmap

- Problem definition
- Overview of **MetricForensics**
- **Models and methods**
  - Data model
  - Metrics
  - Analysis techniques
- Experimental results
- Conclusions
Dynamic graph model for MetricForensics

- **Nodes**
  - Can be “active” or “inactive” at different times

- **Edges**
  - Can be instantaneous or have duration
  - Can be weighted or unweighted

- **Snapshot graphs**
  - Defined by an instant in time
  - Contains all active nodes and edges at that instant

- **Summary graphs**
  - Summarize a set of consecutive snapshot graphs
  - Various summarization policies are available
    - Sum of adjacency matrices
    - Average of adjacency matrices
    - Element-wise max of adjacency matrices
Three categories of metrics

- Global (graph-level) metrics
  - Based on graph structure and global dynamics alone
  - Fast algorithms
- Regional (community-level) metrics
  - Track group structure of nodes or edges
  - Slower than global metrics
- Local (vertex-level) metrics
  - Vertex- or edge-centric calculations
  - Computationally expensive
    - Cannot run on full graph all the time
  - Can include vertex attributes and deep inspection of available data
## Subset of our metrics

<table>
<thead>
<tr>
<th>Global</th>
<th>Community*</th>
<th>Local</th>
</tr>
</thead>
<tbody>
<tr>
<td># of active vertices</td>
<td># of communities</td>
<td>Betweenness centrality</td>
</tr>
<tr>
<td># of active edges</td>
<td>Community size</td>
<td>RWR scores</td>
</tr>
<tr>
<td>Average vertex degree</td>
<td>Largest community fraction</td>
<td>Spectral measures</td>
</tr>
<tr>
<td>Average edge weight</td>
<td>Community link density</td>
<td>Vertex clustering coefficient</td>
</tr>
<tr>
<td># of CC</td>
<td>Variation of Information</td>
<td>Vertex community-stability</td>
</tr>
<tr>
<td>Fraction of vertices in LCC</td>
<td>[Karrer+, Phys. Rev. E (77), 2008]</td>
<td>OddBall</td>
</tr>
<tr>
<td># of articulation points</td>
<td></td>
<td>[Akoglu+, PAKDD 2010]</td>
</tr>
<tr>
<td>MST weight</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Top-k eigenvalues of A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jaccard($V_T$, $V_{T-1}$)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jaccard($E_T$, $E_{T-1}$)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* Can use any community detection algorithm
Time series analysis

- Each signal considered in isolation
- Can use Fourier analysis, wavelets, ARMA, fractal dimension, …
Metric analysis

• Our analysis suite includes correlation analysis, clustering, …

• Simple scatter plots reveal clusters and outliers

• Colored points by timestamp identify “behaviors” versus “events”
Analysis by visualization

- Many tools exist for static graph visualization
  - Computationally expensive
  - Less useful for large graphs
- We developed a tool for dynamic viewing
  - Highlights more active vertices
  - Differentiates sources from sinks
Roadmap

- Problem definition
- Overview of MetricForensics
- Models and methods
  - Data model
  - Metrics
  - Analysis techniques
- Experimental results
- Conclusions
Data used in experiments

- **ENTP**: IP communication over one week of a conference
  - Collection points inside an enterprise network
- **RMBT**: Reality Mining device proximity data
- **LBNL**: IP traffic from a single port at LBNL
  - Contains some scanning activity

<table>
<thead>
<tr>
<th>Data Graph</th>
<th># Total Vertices</th>
<th># Unique Edges</th>
<th># Total Edges</th>
<th>Observation Time (min)</th>
<th>Window Size* (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENTP</td>
<td>2.9M</td>
<td>6.6M</td>
<td>31.9M</td>
<td>6.5K</td>
<td>0.5</td>
</tr>
<tr>
<td>RMBT</td>
<td>25.5K</td>
<td>55.9K</td>
<td>2.0M</td>
<td>526K</td>
<td>30</td>
</tr>
<tr>
<td>LBNL</td>
<td>3.3K</td>
<td>15.6K</td>
<td>9.3M</td>
<td>60</td>
<td>0.0083</td>
</tr>
</tbody>
</table>

* Determined based on activity rate and expected reaction time to events
Time series view of ENTP at the global level
Eigenvalue elbows

- An elbow corresponds to the two dominant phenomena switching roles
  - Shown by plotting $\lambda_1$ vs. $\lambda_2$
- Examples of phenomena
  - Heavy edge
  - Heavy component
  - High-degree vertex
Eigenvalue elbows on ENTP

- Eigenvalue elbows are insensitive to policy that generates summary graphs

X-axis: $\lambda_1$ & Y-axis: $\lambda_2$
Broken correlations on ENTP’s Region 1

- In Region 1, $\lambda_1$ is constant while $\lambda_2$ is changing

**X-axis:** various global metrics & **Y-axis:** Correlation coefficient with $\lambda_1$
Fractal dimension on ENTP

- Fractal dimension measures "burstiness"

- Observed drop corresponds to a prolonged, low-volume spike in the number of new communications
Community analysis on ENTP’s Region 1

- Recall that ENTP’s Region 1, $\lambda_1$ is constant while $\lambda_2$ is changing.
- Circled clusters are from $k$-means:
  - Large cluster: “normal” behavior
  - Isolated points: “anomalies”
  - Small cluster: “strange” behavior

$X$-axis: $\lambda_1$ & $Y$-axis: fraction of vertices in the largest community.
Local analysis on RMBT with OddBall

- Each point is a device
- Devices on the blue line have egonets that are cliques
- Outliers (circled) are devices whose egonets are stars
- Middle cluster has devices that are neither cliques nor stars
  - Similar to what we expect normally

*Lightweight stars* have many low-weight edges in their star formation
MetricForensics’ wall-clock time

- MetricForensics’ runtime is near real-time
- Runtime as a percentage of observation time
  - ENTP = 1.66%
  - RMBT = 0.001%
  - LBNL = 11.42%

<table>
<thead>
<tr>
<th>Data Graph</th>
<th># Total Vertices</th>
<th># Unique Edges</th>
<th># Total Edges</th>
<th>Obs. Time (min)</th>
<th>Window Size* (min)</th>
<th>Run Time (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENTP</td>
<td>2.9M</td>
<td>6.6M</td>
<td>31.9M</td>
<td>6.5K</td>
<td>0.5</td>
<td>107.75</td>
</tr>
<tr>
<td>RMBT</td>
<td>25.5K</td>
<td>55.9K</td>
<td>2.0M</td>
<td>526K</td>
<td>30</td>
<td>5.47</td>
</tr>
<tr>
<td>LBNL</td>
<td>3.3K</td>
<td>15.6K</td>
<td>9.3M</td>
<td>60</td>
<td>0.0083</td>
<td>6.85</td>
</tr>
</tbody>
</table>
Conclusions

- MetricForensics’ **novel multi-level approach** allows for fast analysis of large volatile graphs
  - Lightweight, automated techniques monitor most of the data
  - Identified events are analyzed by more costly methods
- MetricForensics’ **satisfies the three requirements of effectiveness, scalability, and flexibility/generality**
- Real volatile graph have many **identifiable oddities**: elbows, broken correlations, lightweight stars