SNAPSKETCH: Graph Representation Approach for Anomaly Detection in Graph Stream

Ramesh Paudel and Dr. William Eberle
Department of Computer Science

Introduction

- Identify denial of service attacks, port scans, and other cyber-attacks using network graphs.
- Unique approach that identifies anomalous hotspots by tracking sudden increases/decreases edges connecting to a vertex; or the sudden (dis)appearance of edges with high weight.
- SNAPSKETCH is fully unsupervised, has constant memory space usage, and can be used for real-time anomaly detection.

Research Objective

Problem Statement:
Given a graph stream $G_2 = \{G_1, G_2, \ldots, G_d, \ldots\}$, our goal is to learn a graph representation function $f$ for each graph $G_i \in \mathbb{E}^d$ such that $f : G_i \rightarrow \mathbb{E}^d$ and $d \ll |V|^2$ and using $V_{G_2}$ detect whether a graph $G_i$ at any time $t$ contains an anomalous hotspot.

Goals
- Generate a fixed-size feature vector (SNAPSKETCH) to represent a graph in a graph stream.
- Detect DoS attack (a type of anomalous hotspot) in network traffic using SNAPSKETCH.

Experimentation

- Run RRCF [1] anomaly detection algorithm on sketch vector generated by SNAPSKETCH generated, SpotLight [3], and StreamSpot [2] on the following two datasets and compare their performances.

SNAPSKETCH Framework

- Perform node2vec [5] random walk on the graph and construct n-shingles.
- Identify discriminative shingles (shingles with the highest frequency) and randomly project them into a d-dimensional projection $h_d$.
- Sketch graphs using a simplified hashing of projection vector $h_d$ and the cost of shingles $c_2$.
- The sketch converts the graph $G_i$ into a d-dimensional sketch vector $v_{G_i}$.
- SNAPSKETCH has several advantages, fully unsupervised learning, constant memory space usage, entire-graph embedding, and real-time anomaly detection.

SNAPSKETCH Algorithm

Algorithm 1: SNAPSKETCH Algorithm

\begin{align*}
\text{Input: Graph Stream } & G_2 = \{G_1, G_2, \ldots, G_d, \ldots\} \\
\text{Parameters: } & \text{Graph Detection } d, \text{Number of Discriminative Shingles } k, \text{Walk Length } L, \text{Size of Digest } |V| \\
\text{Output: Graph Sketch } & V_{G_2} \\
\text{1. Function Main}(G_i, L, k, \text{Walk length}) \\
\text{2. while not end of stream do} \\
\text{3. } & \text{Generate OverlapHashes}(G_i, L) \\
\text{4. } & S = |\text{Proposed}(|V)| - |\text{Expected}|/c_2 \\
\text{5. } & S = |\text{Expected} - |\text{Proposed}|/c_2 \\
\text{6. } & \text{Detect using }
\end{align*}

Results

<table>
<thead>
<tr>
<th>Dataset</th>
<th># of Graph</th>
<th># of Anomalies</th>
<th>Edges</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smart Homes IoT</td>
<td>9,678</td>
<td>1,007</td>
<td>29,959,737</td>
</tr>
<tr>
<td>DARPA 1998</td>
<td>3,497</td>
<td>361</td>
<td>3,904,797</td>
</tr>
</tbody>
</table>

Fig. 1. An Illustration of SNAPSKETCH framework

Real-time Anomaly Score

Fig. 2. Anomaly score reported on smart home IoT traffic. Blue plot indicates the ground truth anomalies. Spike in red plots indicates the anomaly score reported by the respective approaches over time.

Fig. 3. Anomaly score reported on DARPA dataset.

Conclusion

- SNAPSKETCH can effectively represent the graph into a fixed-sized sketch vector.
- Using RRCF [1] on sketch vector anomalous events like denial-of-service attacks can be detected.
- SNAPSKETCH has better precision and recall than baseline SpotLight [3] and StreamSpot [2] approaches on top-\(m\) anomalous graphs.

Acknowledgements

Funding provided by Tennessee Tech University, College of Engineering for achieving Carnegie classification.

References