TGRAPP: Anomaly Detection and Visualization of Large-Scale Call Graphs

Mirela T. Cazzolato[1,2], Saranya Vijayakumar[1], Xinyi Zheng[1], Namyong Park[1], Meng-Chieh Lee[1], Duen Horng Chau[3], Pedro Fidalgo[4,5], Bruno Lages[4], Agma J. M. Traina[2], Christos Faloutsos[1]

[1]Carnegie Mellon University (CMU)
[2]University of São Paulo (ICMC-USP)
[3]Georgia Institute of Technology
[4]Mobileum
[5]University Institute of Lisbon (ISCTE-IUL)
mteixeir@andrew.cmu.edu, polo@gatech.edu, {pedro.fidalgo, bruno.lages}@mobileum.com, agma@icmc.usp.br,
{saranyav, xinyizhe, namyongp, mengchil, christos}@cs.cmu.edu

Abstract

Given a million-scale dataset of who-calls-whom data containing imperfect labels, how can we detect existing and new fraud patterns? We propose TGRAPP, which extracts carefully designed features and provides visualizations to assist analysts in spotting fraudsters and suspicious behavior. Our TGRAPP method has the following properties: (a) *Scalable*, as it is linear on the input size; and (b) *Effective*, as it allows natural interaction with human analysts, and is applicable in both supervised and unsupervised settings.

Introduction

Given millions of phone calls with source and destination numbers, timestamps and duration, how can we locate anomalous activities and fraudsters? How can we assist specialists in detecting, visualizing and understanding different anomalies on large-scale who-calls-whom graphs?

Our goal is to help analysts sift through millions of phone calls to: (a) spot suspicious nodes, quickly and (b) provide explanations, e.g., through visualization. Explainability is vital, as companies must give good reasons for blocking a phone number that is suspected of fraud.

We propose TGRAPP with following properties:

- Scalable: TGRAPP scales linearly with the database size;
- Effective: TGRAPP spots suspicious nodes, while being *Explainable*, with meaningful visualizations; *Automatic*, not requiring parameter tuning; and *Interactive*, allowing drill-down and deep dives for suspicious nodes.

Figure 1(B) shows TGRAPP in action: Figure 1(B-ii) provides a heat-map scatterplot, where some nodes demonstrate anomalous behavior in that they are extremely regular, along the 45-degree line; Figure 1(B-iii) is the result of deep-dive (our Module-iii of TGRAPP), showing the parallel-axis plot, where every one of these nodes have *exactly 1-second* duration.

Figure 1(B-'corroboration') demonstrates domain expert corroboration of our analysis, who confirmed that this is a so-called 'camouflage' attack: fraudulent actors use decoys with automated domestic traffic to obfuscate their fraudulent international traffic. The numbers used are non-fraudulent phone numbers that have been co-opted via Telephonic Denial of Service.

We emphasize that we use *real-world* anonymized call data rather than common benchmark fraud datasets.

Reproducibility: Code and synthetic datasets are open-sourced at *https://github.com/mtcazzolato/tgrapp*.

Related Work

There is a lot of work on *Anomaly detection* (Akoglu, Tong, and Koutra 2015; Liu, Ting, and Zhou 2008; Lee et al. 2021); on *dense sub-graph detection*, which are usually suspicious (Hooi et al. 2016; Shin, Eliassi-Rad, and Faloutsos 2016); on *unsupervised clustering*, that group nearby points and indicates groups and trends in the dataset (Hamerly and Elkan 2003; Ester et al. 1996; Ankerst et al. 1999; Belth et al. 2020; Belth, Zheng, and Koutra 2020); on *(semi-)supervised methods*, when only some of the nodes have labels (Ester et al. 1996; Ankerst et al. 1999; Hamerly and Elkan 2003); on *time-evolving graphs* (Kazemi et al. 2020; Lee et al. 2020); on *graph visualization* (Stolper et al. 2014; Chau et al. 2011; Zheng et al. 2022); and on *call graphs* (de Melo et al. 2010; Akoglu, de Melo, and Faloutsos 2012).

However, none of these methods fulfill all the properties presented in the Introduction that TGRAPP offers.

The Proposed Demo: TGRAPP

Figure 1(A) shows TGRAPP and its modules.

- Module i: Feature extraction,
- Module ii: Static and interactive visualization – (a) heatmaps and (b) scatter matrix of selected features,
- Module iii: Deep dive capabilities – (a) for a single node, cumulative in/out degree and number of calls, and cumulative in/out call duration per hour; (b) for a group of nodes, adjacency matrix, parallel coordinates, graph spring model.
- Module iv: Attention Routing - it highlights the outliers and micro-clusters, in importance order.

Features

We use node-level features: if a given node is a fraudster, we want to capture its behavior, and spot patterns and deviations from the typical behavior of a non-fraudulent subscriber.

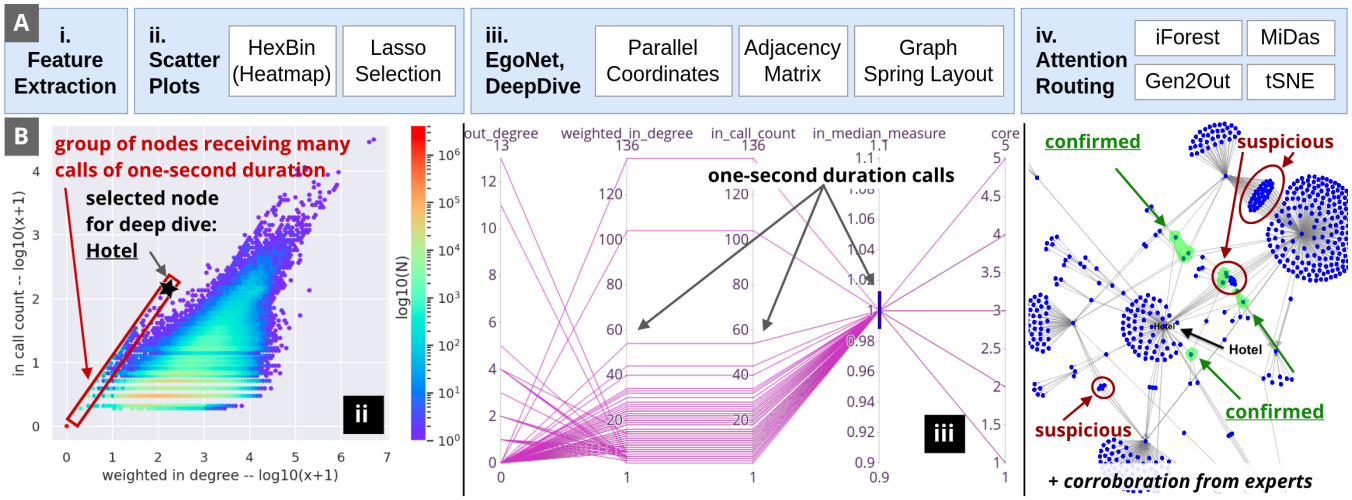

Figure 1: TGRAPP in action, spotting fraudsters. The pipeline in (A) shows the steps followed by TGRAPP, and in (B) we present the resulting visualizations. After (i) feature extraction, TGRAPP provides tools for visualizing (ii) combinations of features in scatter plots. The Lasso Selection allows users to select a set of nodes to dive deep. TGRAPP constructs (iii) an EgoNet with selected nodes, showing the adjacency matrix, parallel coordinates and the EgoNet using the Spring Layout. The (iv) attention routing module provides tools for detecting fraud in the given dataset.

Static Case. There are countless features we can extract for each node: PageRank, radius, several betweenness measures, clustering coefficient, linear embeddings (PCA/SVD/Laplacian), non-linear embeddings (GCNs), to name a few. For scalability, and explainability, we chose the in- and out- versions of: (a) the degree of each node (count of distinct sources/destinations), (b) the weight (sum of minutes), and (c) the count (total number of phone calls). Moreover, we used the so-called *core number* of each node, indicating how well-connected that node is (see, e.g., (Shin, Eliassi-Rad, and Faloutsos 2018) for the exact definition).

Dynamic/Time-Evolving Case. The inter-arrival times ('*IAT*') of events often reveal fraudsters: for example, telemarketers will call a new number every few minutes, with a small variance.

The dynamic features we propose for every node are the *robust* versions of mean and standard deviation, and specifically: for *Inter-Arrival Time* (*IAT*): median-IAT, *IQR*-IAT; for incoming/outgoing *call duration*: *median call duration*, IQR *call duration*.

Visualizations

TGRAPP is effective by optimizing for explainability and interactivity through visualization. There are two challenges:

1. **Curse of Dimensionality**: We must visualize a medium-dimensionality space in an effective manner.
2. **Scalability**: We must plot, and interact with, a million of data points.

To address them, for Curse of Dimensionality, we use only a few carefully designed features, as discussed in Section ; for Scalability, we propose two solutions:

1. heatmaps (as in Figure 1(B-ii)) which eliminate duplicate points and over-plotting issues, and
2. filtering of low-activity nodes (say, with less than c phone calls total).

We implemented three main modes of interaction:

- Label Hovering: When analysts hover over a node, a label card will show the node ID (hash), and feature values.
- Labeled Node Highlighting: When there are labeled frauds, the plot will automatically highlight the labeled frauds. Analysts can configure opacity or color for highlighting.
- Brushing and Linking: When analysts select a region in the paired plots by dragging the mouse, all the plots will be updated lively, so that only the selected region is shown on the plots.

Finally, for further explainability, we allow for visualization of a subgraph, e.g., induced subgraph of a set of suspicious nodes. If the subgraph is small, we propose the spring model; otherwise, we plot the adjacency matrix after careful reordering of rows and columns, as in Figure 1(B-iii)

Complexity Analysis

TGRAPP is $O(|E|)$, that is, linear on the number of edges E. Proof omitted for brevity.

Conclusions

TGRAPP aims to help human analysts detect fraud in billion-scale call graphs, by being:

1. Scalable: it scales linearly with the input size
2. Effective: it works on real world data and it is: *Explainable* thanks to our plots; and *Automatic* (no need for parameter tuning).

Reproducibility: TGRAPP is open-sourced on *https://github.com/mtcazzolato/tgrapp*.

Acknowledgments

Funding was provided by the Pennsylvania Infrastructure Technology Alliance - PITA; the São Paulo Research Foundation - FAPESP (grants 2021/11403-5, 2020/11258-2, 2016/17078-0, 2020/07200-9); the National Council for Scientific and Technological Development (CNPq); a fellowship award under contract FA9550-21-F-0003 through the National Defense Science and Engineering Graduate (NDSEG) Fellowship Program, sponsored by the Air Force Research Laboratory (AFRL), the Office of Naval Research (ONR) and the Army Research Office (ARO); the AIDA project - Adaptive, Intelligent and Distributed Assurance Platform (reference POCI-01-0247-FEDER-045907) leading to this work is co-financed by the ERDF - European Regional Development Fund through the Operational Program for Competitiveness and Internationalisation - COMPETE 2020 and by the Portuguese Foundation for Science and Technology - FCT under CMU Portugal. 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, or other funding parties. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

References

Akoglu, L.; de Melo, P. O. S. V.; and Faloutsos, C. 2012. Quantifying Reciprocity in Large Weighted Communication Networks. In *PAKDD (2)*, volume 7302 of *Lecture Notes in Computer Science*, 85–96. Springer.

Akoglu, L.; Tong, H.; and Koutra, D. 2015. Graph based anomaly detection and description: a survey. *Data Min. Knowl. Discov.*, 29(3): 626–688.

Ankerst, M.; Breunig, M. M.; Kriegel, H.; and Sander, J. 1999. OPTICS: Ordering Points To Identify the Clustering Structure. In *SIGMOD Conference*, 49–60. ACM Press.

Belth, C.; Zheng, X.; and Koutra, D. 2020. Mining Persistent Activity in Continually Evolving Networks. In *KDD*, 934–944. ACM.

Belth, C.; Zheng, X.; Vreeken, J.; and Koutra, D. 2020. What is Normal, What is Strange, and What is Missing in a Knowledge Graph: Unified Characterization via Inductive Summarization. In *WWW*, 1115–1126. ACM / IW3C2.

Chau, D. H.; Kittur, A.; Hong, J. I.; and Faloutsos, C. 2011. Apolo: making sense of large network data by combining rich user interaction and machine learning. In *CHI*, 167–176. ACM.

de Melo, P. O. S. V.; Akoglu, L.; Faloutsos, C.; and Loureiro, A. A. F. 2010. Surprising Patterns for the Call Duration Distribution of Mobile Phone Users. In *ECML/PKDD (3)*, volume 6323 of *Lecture Notes in Computer Science*, 354–369. Springer.

Ester, M.; Kriegel, H.; Sander, J.; and Xu, X. 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Simoudis, E.; Han, J.; and Fayyad, U. M., eds., *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA*, 226–231. AAAI Press.

Hamerly, G.; and Elkan, C. 2003. Learning the k in k-means. In *NIPS*, 281–288. MIT Press.

Hooi, B.; Song, H. A.; Beutel, A.; Shah, N.; Shin, K.; and Faloutsos, C. 2016. FRAUDAR: Bounding Graph Fraud in the Face of Camouflage. In *KDD*, 895–904. ACM.

Kazemi, S. M.; Goel, R.; Jain, K.; Kobyzev, I.; Sethi, A.; Forsyth, P.; and Poupart, P. 2020. Representation Learning for Dynamic Graphs: A Survey. *J. Mach. Learn. Res.*, 21: 70:1–70:73.

Lee, M.; Shekhar, S.; Faloutsos, C.; Hutson, T. N.; and Iasemidis, L. D. 2021. Gen^{2}Out: Detecting and Ranking Generalized Anomalies. In *IEEE BigData*, 801–811. IEEE.

Lee, M.; Zhao, Y.; Wang, A.; Liang, P. J.; Akoglu, L.; Tseng, V. S.; and Faloutsos, C. 2020. AutoAudit: Mining Accounting and Time-Evolving Graphs. In *IEEE BigData*, 950–956. IEEE.

Liu, F. T.; Ting, K. M.; and Zhou, Z. 2008. Isolation Forest. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy*, 413–422. IEEE Computer Society.

Shin, K.; Eliassi-Rad, T.; and Faloutsos, C. 2016. CoreScope: Graph Mining Using k-Core Analysis - Patterns, Anomalies and Algorithms. In *ICDM*, 469–478. IEEE Computer Society.

Shin, K.; Eliassi-Rad, T.; and Faloutsos, C. 2018. Patterns and anomalies in k-cores of real-world graphs with applications. *Knowl. Inf. Syst.*, 54(3): 677–710.

Stolper, C. D.; Kahng, M.; Lin, Z.; Foerster, F.; Goel, A.; Stasko, J. T.; and Chau, D. H. 2014. GLO-STIX: Graph-Level Operations for Specifying Techniques and Interactive eXploration. *IEEE Trans. Vis. Comput. Graph.*, 20(12): 2320–2328.

Zheng, X.; Rossi, R. A.; Ahmed, N. K.; and Moritz, D. 2022. Network Report: A Structured Description for Network Datasets. In Hasan, M. A.; and Xiong, L., eds., *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, 3694–3704. ACM.