Proof-of-concept of a Communication Surveillance technique using Graph theory and Network Analysis

Introduction

I have built a graph-based solution to analyze and visualize chart logs (bilateral and multilateral chats). Again, given the confidentiality of logs, I have generated synthetic data for the purpose of this study.

In terms of prior usage, Palantir technologies has implemented similar solution at JP Morgan. The solution has capabilities to scan through emails, browsing histories, GPS location using company owned smart phones, transcripts of phone conversations and employee badge timings.(https://www.bloomberg.com/features/2018-palantir-peter-thiel). The following is an attempt to implement a proof-of-concept to build a visualization-based model for a graph-based surveillance technique.

There is an active use of network analysis and Graph Theory in areas of social networking, communication, organizational change management and recently in area of market surveillance. Network Analysis helps us in visualizing multiple data points and drawing insights from a complex set of connections. In Graph theory, insights can be drawn in either quantitative measures like centrality (degree, closeness or eigenvector) or network density, community formation, etc. via visualizations and attribute mapping

Data Overview

The synthetically generated data has three columns:

1. Inviter: Person initiating the chat
2. Invitee: Person receiving the chat
3. MsgCount: Number of messaged sent in corresponding conversation

Following is a sample of the dataset.

Num	Inviter	Invitee	MsgCount
1	P1	P63	180
2	P2	P64	135
3	P3	P65	88
4	P4	P66	82
5	P5	P67	67

The data consists of 130 participants, with 91 conversations. The following graph helps us understand get an initial idea of the dataset.

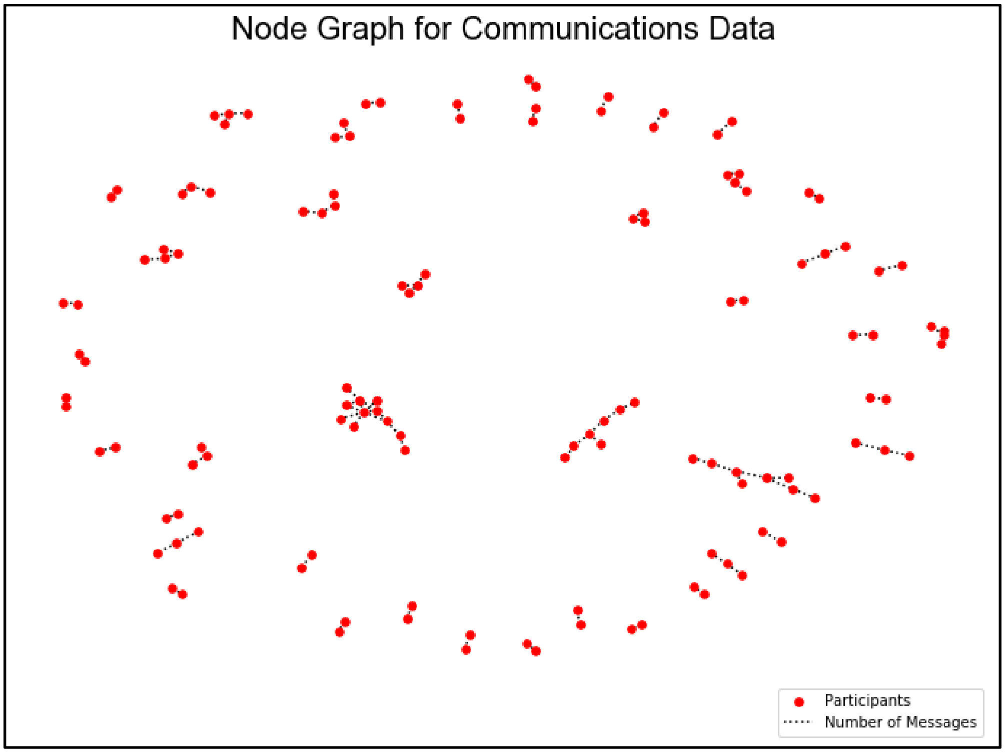

Figure 1: Overview of Communications Data

Graph theory Concepts

Centrality: This paradigm aims to denote the most important vertex/node in a network. The different types of centrality in analyzing the network are given as follows (Reference: https://sctr7.com/2013/06/17/adopting-analytics-culture-6-what-information-is-gained-from-social-network-analysis-6-of-7/). Also, following is the list of various concepts associated with a network in graph theory.

Degree: The number of connections/edges coming into a node
Closeness: This denotes the minimum number of steps required to reach to the next node in a network. In our context, it measures how quickly can one trader connect to others in the network
Eigenvector: Measures a trader's connection to those who are highly connected. A person with a high score will be someone who is influencing multiple players (who in turn are highly connected) and is exercising control behind the scenes.

The following image shows the values for the three types of centrality mentioned above.

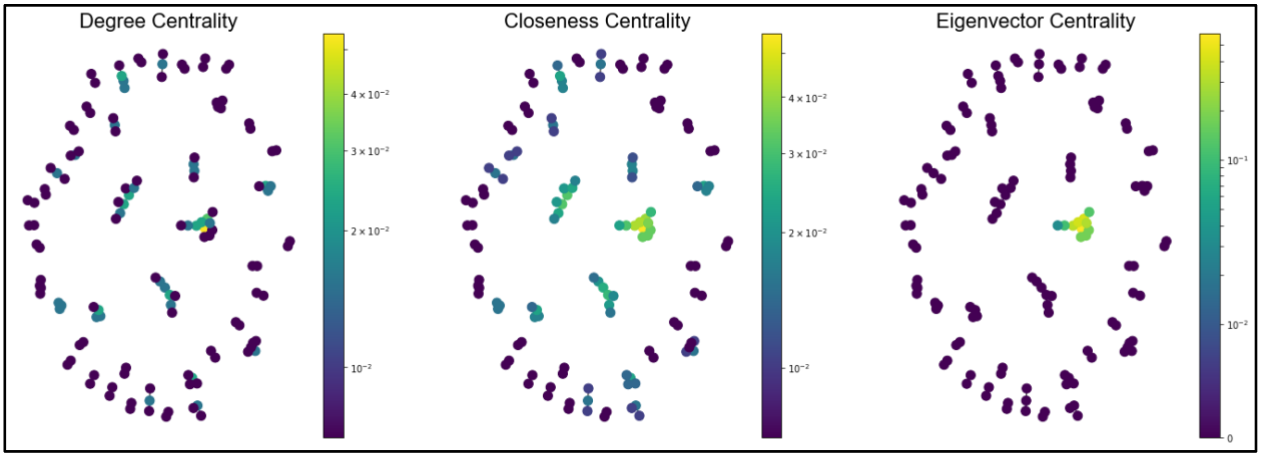

Figure 2: Centrality Metrics for the Dataset

Based on the graphs above, some of the most influential participants are P1, P12, P16, P29, P44 and P63.

Algorithms for Community Detection for the Data

I have focused on the visual aspects of a community in this work. This is a very good way of analyzing network-based dataset as it helps us to correlate various other data sources with the chat logs. Thus, helping compliance analysts arrive at a decision. In addition, these visualizations can be leveraged into reports and available charting tools available in the institution. The communication patterns derived from the above network analysis helps to benchmark an individual or a group of traders against its peer group. For instance, we assign a score to the number of interactions(edges) between two traders (nodes). Based on the score, we create communities based on connectivity between the traders, we will have groups that share more messages among group members as compared to the rest of the network. This exercise of determining closely knit groups of nodes is called as community detection in graph theory and the score is called as Modularity. Community detection algorithms can be of multiple types with varying levels of success. I have used Louvain algorithm for community detection is this case.

Louvain Algorithm for Community Detection

This algorithm works on the principle of partitioning a network into mutually exclusive communities such that the number of edges across different communities is significantly less than expectation, whereas the number of edges within each community is significantly greater than expectation. The Louvain algorithm is one of the most widely used for identifying communities due its speed and high modularity. Modularity values can span from -1 to 1, and the higher the value, the better the community structure that is formed.

I performed the Louvain algorithm on this dataset and observed that we have 46 communities, and a modularity of 0.953, which is a pretty good solution. Also, we see a few communities that have more than 3 members and some of the most influential people are in those communities.

For example, P1, P12, P16 and P44 are all in community 0. These are some of the higher influential participants.

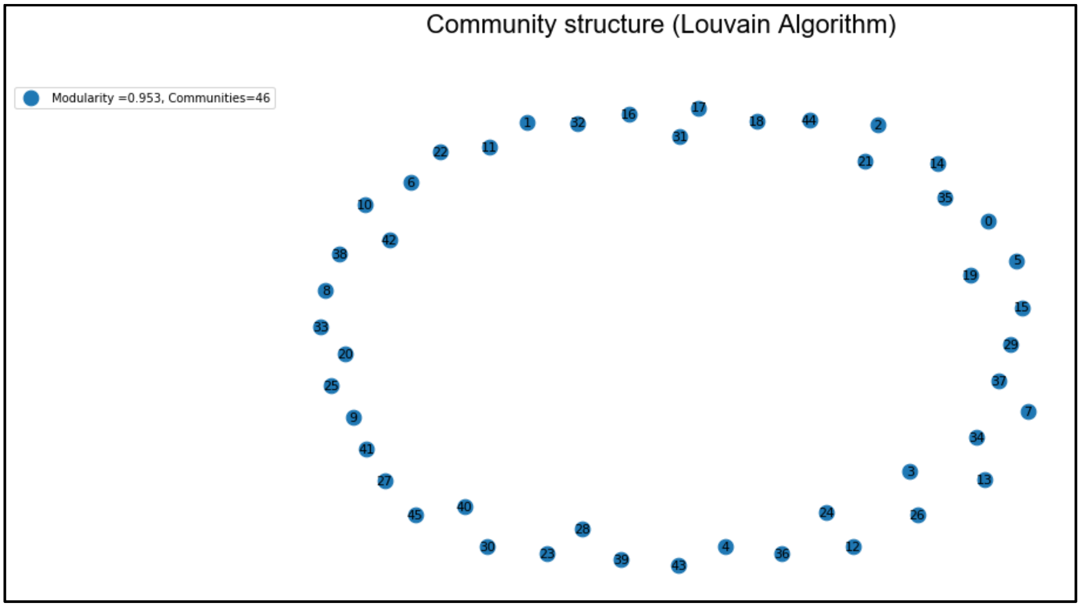

Figure 3: Community detection using the Louvain Algorithm

If we were to visualize all the non-overlapping communities in different colors, we would get the following figure.

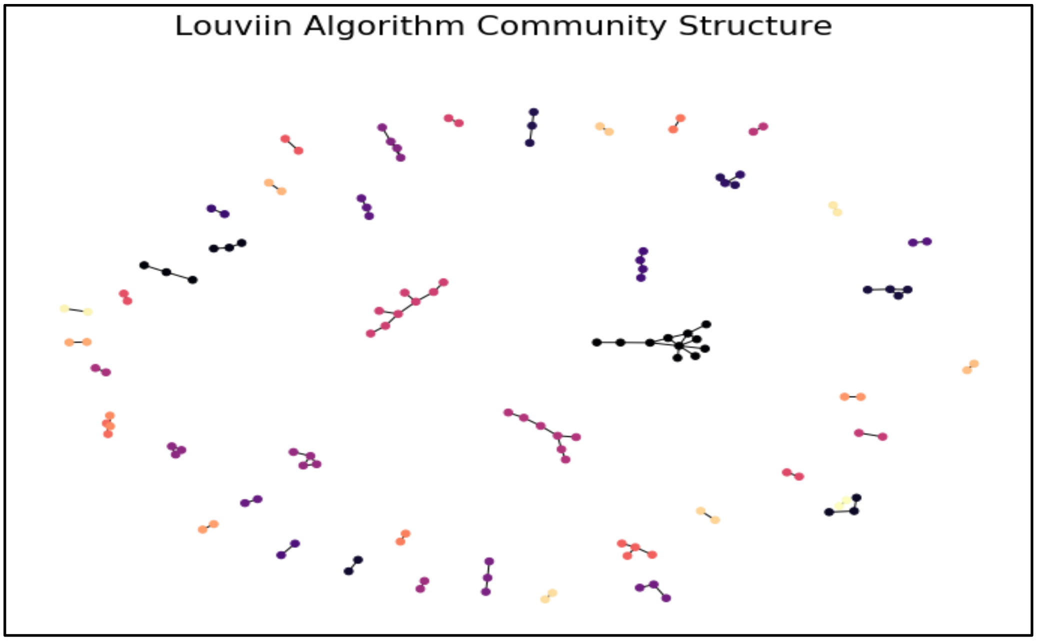

Figure 4: Graphical representation of communities using the Louvain Algorithm

Conclusion

Using community detection, we can analyze communication data and correlate derived analytics with trade data, to detect anomalous behavior. Thus, having a graph-based approach can supplement an analyst’s decision-making process. Further, we can impute the output from the above work into anomaly detection algorithms and detect communities or individuals exhibiting anomalous trading behavior.