VAST Challenge of 2020 - Mini challenge 1

Graph analysis and similarity between graphs within large data

Anna Granberg (Anngr950)   Filip Hamrelius (Filha243)

Abstract—
This report presents the analysis approach for identifying patterns within graphs representing connected people. This is one solution to the VAST Challenge of 2020 mini challenge one. The data, according to the challenge description, comes from the Center of Global Strategy (CGSC) and contains structures aligned with specific social groups denoted and gathered by white hat groups, i.e a collective of ethical hackers and cybersecurity experts. The broad aim of the challenge was to identify patterns within the different networks that align with given structures already established to represent the behaviour of these social groups. The analysis approach consisted of using the JavaScript library D3 for visualizations and Python for data processing. The graphs were represented as a directed network with a complementary user interface. This was used together with an interactive horizon activity chart to gain insights and match the candidates to the template graph. The final visualization was a sorted radial bar chart mapping the distribution of edge types for the different candidates. The second part of the challenge involved finding the similar structures within a very large graph, a network derived from a CSV file with 123,892,863 rows. This was solved through extensive preprocessing of the data using visual analytics in-between, specifically identifying and extracting relevant structures from given seeds, i.e starting points within the graph. The analysis of these was then done in the same way as for the first part, using the first visualization tool. In summary it could be determined trough analysis of these visualizations that graph number one is the best match for the template graph. Similarly the first seed leads to the most similar structure compared to the template graph, in the larger graph.

1 Introduction

This report explains the approach towards a solution for the VAST challenge of 2020, mini challenge one. The challenge was stated as follows: ”Center for Global Cyber Strategy (CGCS) researchers have used the data donated by the white hat groups to create anonymized profiles of the groups. One such profile has been identified by CGCS sociopsychologists as most likely to resemble the structure of the group who accidentally caused this internet outage. You have been asked to examine CGCS records and identify those groups who most closely resemble the identified profile.” [1] This report aims to determine the group responsible for the outage with the help of analysing the questions below.

1. Compare the five candidate subgraphs to the provided template. Show where the two graphs agree and disagree. Which subgraph matches the template the best?

   1.1 Which key parts of the best match help discriminate it from the other potential matches?

2. CGCS has a set of “seed” IDs that may be members of other potential networks that could have been involved. Take a look at the very large graph. Can you determine if those IDs lead to other networks that matches the template?

This report also discusses the analysis strategy along with the implemented visualization and data processing techniques relevant to a problem of this nature. Detailed attention is given to the methods used for data preprocessing, the tools employed for visual representation of the data, and the specific strategies adopted to ensure accurate and insightful analysis. By combining these elements, the report provides a comprehensive overview of the approaches taken to address the complexities of the VAST challenge of 2020 - mini challenge one. The challenge description can be accessed in its entirety by reading the references in the end of this document.

2 Analysis Strategy Outline

The nature of the networks demands multiple views of the data to gain any significant information. The analysis strategy, therefore, relies on multiple visualizations to create a comprehensive overview of the data. Where each visualization strengthens the one before it, concluding a solid analysis.

For the first step of the analysis procedure a dual window visualisation was implemented, this consisted of representing the template graph and a potential candidate at the same time. This visualisation had a user interface with the possibility to filter on edges as well as change the length between the nodes, i.e change the gravity to which the graph was rendered with. The graph used for the first visualisation was a directed network were colours was used to map different edge types to the arrow indicating the direction of the edge. Symbols of different sorts were used to differentiate between the different variations of node types. Filtering options was implemented, enabling visualization of specific edge types. The purpose of this visualization was to create a visual overview for each graph, enabling visual analytics of the graph structures and outlining specific and important connections between nodes.

The directed network lacks insight into the distribution over time. To analyse this, a horizon activity chart with shapes was implemented where the x-axis represents time and the y-axis is divided by each node source. Similar to the first approach, a colour legend was used to provide an easier overview and differentiate between the different node types. This visualization was implemented to help differentiate between the candidates derived from visualization two.

The last step was to obtain concrete numbers for an intuitive comparison. For this, a sorted radial bar chart was implemented, which naturally emphasises size proportions of each segment, making it easier to see the size and contribution of each attribute. The radial bar chart is sorted based on the total number of occurrences of attributes, providing a clear visual hierarchy. This sorting enhances the ability to quickly identify the most significant segments and compare them effectively. This was implemented to strengthen the analysis done in the other two visualizations.

The second question relies on a larger dataset, and due to the nature of the seeds, it needs to be preprocessed before any visualization can be done. For this part, Python was used. The seeds were extracted from their CSV files and for the first level of communication, a linear search within the large graph was executed. This was to check for any occurrences of the seed ID as a target or source. When all instances were found, it would save the structure to a new file, the result being the seed and its directly connected neighbours. This data enabled further investigations and assumptions, it also enabled a second iteration where weak and irrelevant connections could be filtered out, finally resulting in candidates for the template graph.

3 Data Characterisation and Preparation

This chapter presents the data available to solve the challenge as well as some delimitation's taken trough out this analysis approach.

3.1 Data

The data given for this challenge was given in multiple CSV files. All data has the same structure, one entity can have multiple node types, always representing a person but can also represent something more, e.g a person and a product. With this in mind the node types are as follows: ”Person (used in all channels)”, ”Product category (for the procurement channel, eType = 3)”, ”Document (from the co-authorship channel, eType = 4)”, ”Financial category (from financial demographics channel, eType = 5)” and ”Country (from the travel channel, eType = 6)”

The eType refers to the edge type, there are seven different edge types, these are as follows: ”Email”, ”Phone”, ”Sell (procurement)”, ”Buy (procurement)”, ”Author-of”, ”Financial (income or expenditure, depending on direction)” and ”Travels-to”.

All entities within the graph contains these columns, even if they are not always filled with data. The columns are as follows: ”Source”, ”eType (edge type)”, ”Target”, ”Time”, ”Weight”, ”SourceLocation”, ”TargetLocation”, ”SourceLatitude”, ”SourceLongitude”, ”TargetLatitude” and ”TargetLongitude”.

For the first question there are six files to take into consideration. There is one template graph describing an already established relationship pattern for the groups of interest, this network consists of 1325 rows, i.e connections. Then there are five potential candidate graphs, ranging in size from 400 to 1300 rows.

For the second question a large graph is involved together with three seeds that can be found within the large graph. The large graph consists of 123,892,863 rows.

3.2 Delimitations

All data except edge type five was used. As denoted in the challenge description channels with edge type 5 creates a lot of edges that do not represent person-to-person connections in the same way as the other channels, this analysis focuses on person-to-person interactions and connections. The choice to exclude analysis of edge type five where therefore taken.

4 Results

This chapter presents and showcases the different visualizations and what information could be derived from them.

4.1 Compare the five candidate subgraphs to the provided template. Show where the two graphs agree and disagree. Which subgraph matches the template the best?

The template graph visualized as a directed network can be seen in Figure 1. The blue squares denote clusters of communication, red represents key persons, green indicates a buy and a sell occurrence, the pink squares signify a cluster of travel connections, and yellow denotes co-authorship connections. The ”-99” in the co-authorship area is explained in the challenge description as a placeholder, meaning it can represent any number of co-authorship connections, not specific to amount or ID numbers.

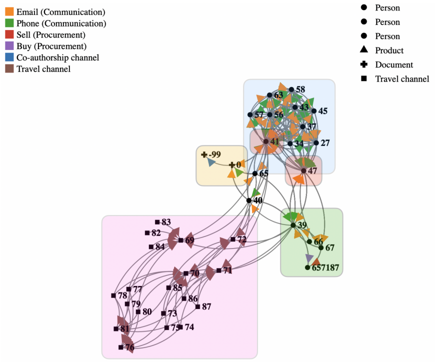

Fig. 1: Directed network - Template data

In Figure 2, candidate number one can be viewed in the same manner as the template graph. At first glance, many similarities can be observed. The most prominent ones are the large and compact communication cluster, the travel cluster, the buy/sell occurrences, and the co-authorship occurrences. It is also important to note that both of them have a smaller cluster of communication in the middle of the graph, these are the nodes which are not coloured in the figures. Candidate one and the template also shares the same product ID: 657187 within the buy/sell occurrence.

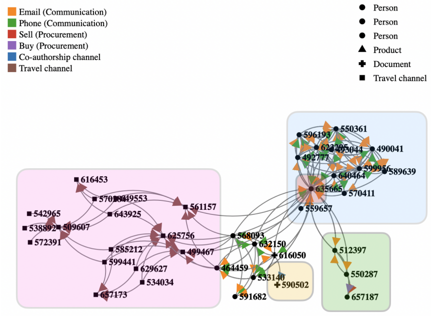

Fig. 2: Directed network Candidate 1

In Figure 3, candidate number two can be viewed as a directed network. While there are many similarities to candidate number one and the template, there are also some differences. In this candidate, the travel cluster is split into two, and there are two key persons connected to two different clusters. In this template the placeholder for co-authorship in the template can be used, in this candidate there are four co-authorship connections, this cluster can still be a direct match to the corresponding single node in the template.

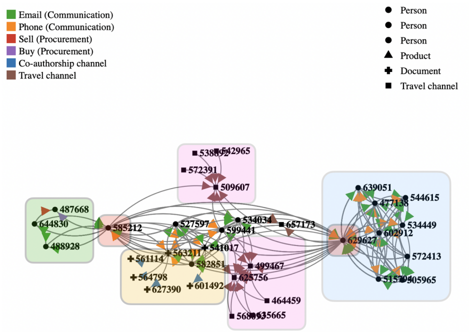

Fig. 3: Directed network - Candidate 2

In Figure 4, candidate number three can be viewed as a directed network. This also shows some similarities to the template graph, especially in the overall clusters. Although the key persons are spread out and the communication cluster is a lot smaller than in the template graph. Neither candidate two or three shares a product ID with the template.

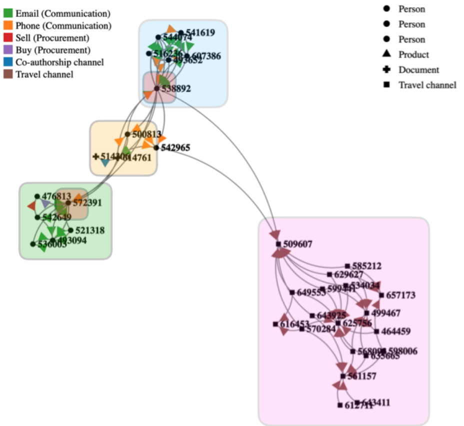

Fig. 4: Directed network - Candidate 3

In Figures 4 and 5, the remaining candidates can be viewed as directed networks. These candidates lack fundamental similarities to the template graph and cannot be considered potential matches.

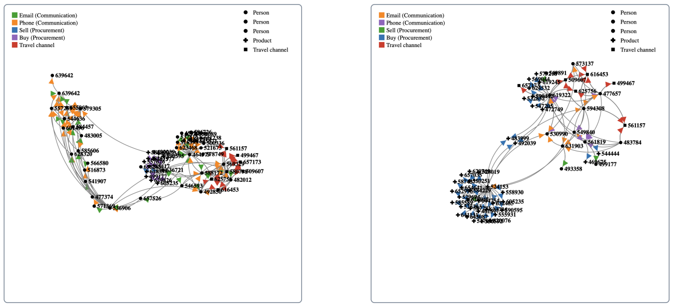

Fig. 5: Directed network - Candidate 4

Fig. 6: Directed network - Candidate 5

Fig. 7: Subplot of other candidates as a directed network

In Figure 8, a horizon activity chart of the template graph can be viewed. This activity chart plots the individual persons over time, with the dots representing occurrences of activities. Within the template, one can identify the same patterns as in the directed network. There are only three colours visible here, representing the clusters that are most prominent in this visualization, making it easy to link them to the directed network.

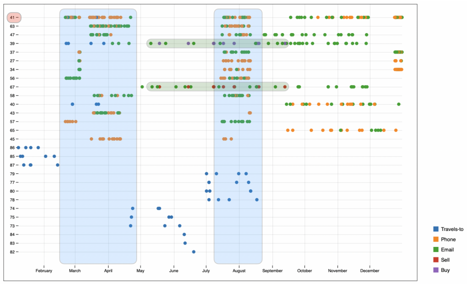

Fig. 8: Horizon activity chart over template graph

In Figure 9, the horizon activity chart of the three best candidates from the previous visualization can be viewed. The third candidate, as the first visualization somewhat showed, shows too many differences from the template and can from this be ruled out. It does have some similarities, but the time period does not correlate with the template. Candidate four and five did not show any resemblance to the template.

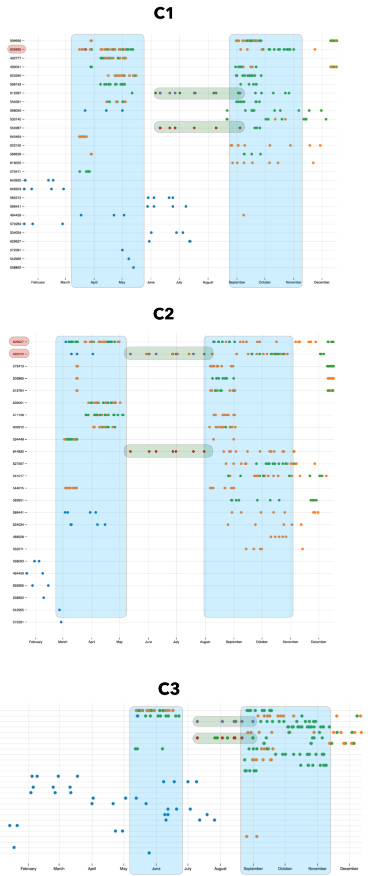

Fig. 9: Horizon activity chart over best potential candidates - C1,C2 and C3

In figure 11 the radial bar chart for all graphs can be viewed, this clearly shows that only candidate number one and two are the only ones of interest, strengthening the conclusion from the other two visualisations.

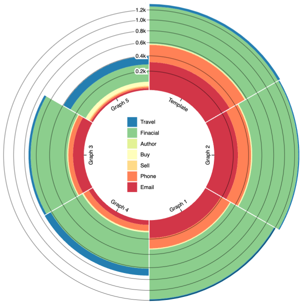

Fig. 10: Sorted, radial bar chart over edge types for all candidates and template

With all these visualizations taken in to consideration, candidate number one is the most similar to the template graph. The horizon activity chart together with the radial bar chart ruled out the third candidate, the activity chart reinforced the similarities of candidates one and two, and finally, the directed network showed that the first candidate was the most similar. This combined with the fact that the template graph and candidate one have the same product ID for the sell/buy occurrence, the first candidate is the best match to the template.

4.1.1 Which key parts of the best match help discriminate it from the other potential matches?

Taking into account the two best candidates, i.e., candidate number one and two, the key aspects are the similarities between clusters and the template graph. What stands out for candidate number one is the coherent structure of the travel cluster, which aligns well with the template and differentiates it from the second candidate, which has two smaller clusters of the same type. This combined with the fact that the template graph and candidate one have the same product ID for the sell/buy occurrence, the first candidate stands out compared to the second one. The horizon chart mostly confirms the assumptions made with the first visualization and does not help discriminate between the two candidates; however, it clearly distinguishes these two from the third candidate, proving once again that these are the best candidates. The same applies to the third visualization, the sorted radial bar chart.

4.2 CGCS has a set of “seed” IDs that may be members of other potential networks that could have been involved. Take a look at the very large graph. Can you determine if those IDs lead to other networks that matches the template?

The first seed leads to a graph that can be matched to the template graph (see Figure 12). The characteristics of the seed structure differ significantly from those found in the candidate graph, but it is still a valid match, especially compared to the other seed structures. The key feature here is the nature of the co-authorship denoted as ’-99’ in the template graph. This allows for a match to any structure of co-authorship, even if it is as large as the one in the seed structure. Another strong resemblance is the clear communication cluster.

The other two seeds led to structures without similarities to the template. Seed number two only has co-authorship connections, nothing else. This makes it clear it cannot be matched to the template graph since the templates co-authorship cluster has communication connections to it. Seed number three only has sell occurrences, lacking the other connections that the template has, and therefore cannot be a match either.

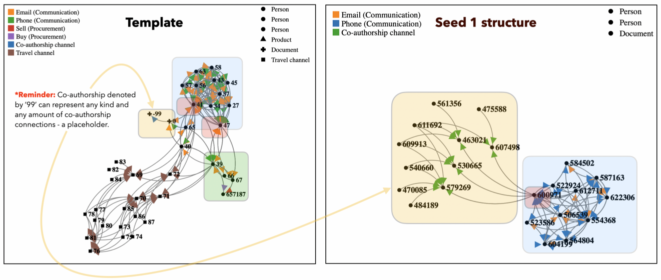

Fig. 11: Final seed structure compared to template graph

4.3 Correct result

Since this is an old challenge there are resources with the correct answers. According to VAST challenge 2020 solution guidelines the graph found most similar in question one is correct, as well as the first seed in question two. [2]

5 Design & implementation of VA solution

The three visualizations were made with JavaScript and the D3 library. To run and style the interface, HTML and CSS were used. For data processing, Python was used. The initial thought was to use JavaScript for everything, but Python’s simplicity in handling files and data, along with greater experience with Python in that area, led to its use. Python was also considered for the visualizations for the same reasons, but this approach lacked the simple implementation of styling and interaction that JavaScript offers and was quickly abandoned.

Data processing within Python only used the CSV library to easily read and write CSV files. The library TQDM was used to track the progress, allowing a simple progress bar and counter to be implemented to give the user an indication of where in the program and the large file it currently is. This was especially useful when working this the large data since computations during trails could take up to an hour.

5.1 Finding seed structures

Extracting the seed structure required a specific pipeline for each seed. The process for the first seed entailed a linear search through a large graph to locate seed one, focusing on immediate neighbours. Relevant edge types (0, 1, and 4, representing Email, Phone, and Co-Authorship connections) were extracted, retaining those occurring more than twice. Connections between these extracted nodes were then identified within the graph using a linear search. The resulting connections were concatenated to form the seed structure, from which one candidate for the final graph structure of seed one was selected.

The process for the second seed followed the first two steps as for the first seed. However, it was determined that the seed structure did not lead to any potential matches since it only contained co-authorship edges, making it impossible to create a structure similar to the template graph. Therefore, the process was stopped at this point.

For the last seed structure, the same process was followed, but different edges were examined. The neighbours contained a significant number of ”sell” edges, which prompted the extraction of connecting ”buy” edges. However, this also led to a structure that did not match the template graph.

6 Discussion

This chapter discusses relevant aspects of the analysis strategy as well as the implemented solution. It also discusses some approaches that were tried but did not contribute to the final analysis.

6.1 Visualizations

6.1.1 Directed Network

Representing multiple edges and nodes is a challenging task, as there is a big risk for overplotting, in turn making it hard to differentiate between the various entities. The approach in this solution relies on using colours for edges in conjunction with symbols. The decision to apply colours solely to the arrowheads of edges was made instead of colouring the entire edge. Applying colours to the entire edge proved strenuous on the eyes and made it difficult to distinguish the paths of different edges. Although using the same colour for all edges did not remove this issue entirely, it contributed to a more visually manageable representation. A slider to control the gravity on the graph was also implemented, this in practise means that the user can change how tightly the graph i grouped together. Together with this, a drag functionality was also added, making it possible to pick a node and drag it to a specific location. Further a filtering tool was also implemented, allowing the user to only visualize certain edges. These tools were added to help the user gain insight towards the graph, giving the opportunity to see the graph from different angles. The interactions was valuable during the investigation and implementation but does not contribute that much to the final result.

Utilizing this visualization for more complex structures with numerous nodes proved to be less efficient, especially when the structure presented symmetry, as in the case of the unfiltered seed structure. In symmetrical structures the gravity used to implement the graph does not provide assistance, its purpose is to create groupings of nodes. When all nodes have edges leading to the same node, they form a single group, even though there may be thousands of nodes. Therefore, this specific approach is not applicable to such types of graphs. This could be solved by looking deeper for more connections, not solely to the seed, but this is a tedious and time consuming task.

6.1.2 Horizon activity chart

The horizon activity chart is an excellent tool for both obtaining an overview and conducting more in-depth analysis, as was the case in this study. While the directed graph highlighted important key features, the horizon chart extended this analysis by helping to identify patterns and trends and allowing for the examination of specific individuals over time.

Despite its strengths, interpreting the chart requires understanding of the characteristics of the data. One limitation noted in this analysis is that the chart does not differentiate based on the size of the graphs or the complexity of the interactions within each node. As a result, nodes with fewer connections might appear less significant, potentially leading to misinterpretation if not cross-referenced with other visualizations.

6.1.3 Radial bar chart

The idea of the radial bar chart is good, and it served a purpose in the later stage of the analysis. However, it does not account for time or the size of each graph, which can result in uneven and misinterpreted results. In this case, our template and Graphs 1, 2, and 3 have a similar number of connections, but graphs 4, 5, and 6 have fewer. The structure of Graphs 4, 5, and 6 might resemble a small part of our template, which the radial bar chart would not have shown. However, in our case, those graphs were already ruled out in previous steps of the analysis, but it is something to keep in mind when using a radial bar chart. With this visualization, some data processing needed to be done, specifically counting occurrences. In hindsight, a step to normalize the data in relation to the size of the network could have been implemented and is an interesting aspect to take into consideration.

6.2 Graph similarity and seeds

The initial approach to identify potential graphs from the large graph data involved locating all neighbours directly connected to the seed. This resulted for seed one in a circular graph consisting of approximately 1500 nodes directly linked to the seed. This approach did not significantly contribute to graph similarity with the template graph. Although examining specific edges individually provided some insights, no significant findings emerged. The second approach entailed extracting specific edges deemed relevant towards a structure similar to the template and constructing a structure from them.

Additionally, there was a discussion regarding the possibility of identifying seed structures using pre-defined graph similarity algorithms. This method may prove relevant when dealing with more extensive datasets, multiple templates, or numerous seeds. Utilizing visual and manual analytics for large datasets and multiple matches might not be efficient and could be addressed by employing graph similarity algorithms.

6.3 Large data

Since the large data contained 123,892,863 rows, finding the seeds and their structure took a long time. When extracting the seed structure a recursive approach was tested, this was implemented to try finding a structure, two levels deep, meaning all nodes connected to the seed, and all nodes connected to those nodes. This quickly became problematic since the structure on level one for seed one had 1501 nodes. Making this one level deeper, assuming from this data that each node has 1,500 nodes connected to it, would create a dataset with 2,250,000 nodes. That is just an assumption, there is a possibility that all the nodes are connected at two levels, if that were the case the algorithm would add all 123,892,863 nodes. This resulted in the assumption that the data at this level is worthless for any kind of visualization or analysis. This lead to the steps of filtering this data using visual analytics, specifically looking at heavily used connections and specific edge types. This made the data manageable and it could finally be compared and matched to the template graph. The last approach is beneficial either way, using brute force approaches to extract data is not a valid method and will for the most part not result in any valuable insights.

Even when only looking at specific edges and subsets of the complete dataset, the calculations took a long time. On multiple occasions, the algorithms had to run for 30 minutes or more. Working with large data is a time-consuming and tedious procedure. It helps to have a modern CPU and enough RAM, although this can be streamlined by not loading all data into active memory.

7 Conclusions

In summary, it can be concluded that a directed network, together with a horizon activity chart and a radial bar chart, can be sufficient for identifying patterns in networks with a lot of different features. However, dealing with more advanced networks would require additional visualizations or as in this case - extensive data processing. It can also be concluded that merely looking at neighbours of seeds is not a sufficient method for identifying graph structures emerging from a specific seed, but it does open up possibilities for a starting point. Using the brute force approach together with visual analytics to manually filter down data did provide reliable results. In conclusion, this visualization has successfully completed the VAST challenge of 2020. According to their solution guide, the graph found most similar in question one is correct, as well as the chosen seed in question two.

References

[1] VAST Challenge 2020 MC1, *The 2020 VAST Challenge: Mini-Challenge 1*, `https://vast-challenge.github.io/2020/MC1.html`, accessed on may 2, 2024.

[2] VAST Challenge 2020 - Solution Guide, *VAST Challenge 2020 - Solution Guide*, `https://visualdata.wustl.edu/varepository/VAST%20Challenge%202020/challenges/Mini-Challenge%201/`, accessed on May 24, 2024.

Contributions from Each Project Member

The project was divided very equally where the first visualization was implemented together. The more time consuming part of the project, the seed finding in part two was also done together most of the time. The horizon chart and radial chart were mostly done by Anna, where Filip spent more time on presentation and report. In summary we did most things together and divided some small tasks to make the project a bit more efficient.