Discovering Community Structure using Network Sampling

[1] Samuel Chen, [2]Joyati Debnath, [3] Ralucca Gera, [4] Brian Greunke, [3] Nicholas Sharpe, and [3] Scott Warnke
[1]Department of Operations Research, Naval Postgraduate School, Monterey, CA 93943
[2] Department of Mathematics and Statistics, Winona State University, Winona, MN 55987
[3] Department of Applied Mathematics, Naval Postgraduate School, Monterey, CA 93943
[4] Department of Computer Sciences, Naval Postgraduate School, Monterey, CA 93943
[2] jdebnath@winona.edu, [3] rgera@nps.edu

Abstract

This paper explores the strength of communities in different types of networks as they are being discovered through inference algorithms based on the degrees of the discovered nodes. The goal is to explore the correlation between the communities of the partial information network extracted at different steps while mapping the topology of a network. The paper measures and analyzes the community size and modularity, and compares the communities from several partial information networks using Normalized Mutual Index (NMI) on four dierent types of real networks: the West Coast Power Grid, P2P Gnutella, a terrorist network, and a snapshot of the Internet from CAIDA.

Keywords: Louvain community detection, community evolution, social networks, technological networks.

1 Introduction

Mapping and monitoring an entire network that is volatile can be difficult, time-consuming, expensive, or in some cases impossible, depending on the type of network in question as discussed in [12]. The current research deals with discovering community structures as the network changes, using the existing algorithm of [9] implemented in NetworkX, on the four networks described next.

One of the networks considered is West Coast Power Grid, a technological network. These are typically man-made, designed, and exist with physical components. The current research uses data from Watts and Strogatz [13]. The network consists of 4941 nodes representing generators and power substations, and 6594 edges representing high-power transmission lines.

The second network considered is the Noordin Top Terrorist Network, composed from five major terrorist organizations that operate in Indonesia. This is a social dark network, i.e. covert and secretive in nature. This network evolved in a manner that is purposefully inefficient [11] with the goal of hiding relationships. It is expected that this network demonstrate some of the hallmarks of a social network [8]. However, since the Noordin Network is a dark network, it lacks some other traditional distinctions of a social network [11], for instance there should be few hubs, if any. There are 139 vertices in this network, with 1499 edges representing different types of relationships between the terrorist, such as communication, classmates, relatives, etc. Noordin Top has the highest degree, and the degrees range from 0 to 233 counting edge multiplicities from the different layers [6].

The third network, the Gnutella, is studied as the first decentralized Peer-to-Peer file sharing networks. Computers act as both servers and clients and the goal of the network was to be dynamic and scalable as new users entered or left the network, attack tolerant and anonymous. Data came from the Stanford Network Analysis Project snapshot of a Gnutella network from August $5, 2002$ as an undirected network [7]. The network has 8846 nodes and $31,839$ edges, and it consists of one weakly connected component.

The fourth and final network is the interconnection of Border Gateway Protocol (BGP) relationships between organizations. Organizations create logical adjacencies to one another as a way to route traffic throughout the entire Internet. CAIDA, the Center for Applied Internet Data Analysis, collects temporal snapshots of these adjacencies. It is represented as a graph where nodes represent organizations, and edges represent the logical connection between these adjacencies [2]. The snapshots compiled are provided by the Stanford Network Analysis Project from 6 February 2006 [7]. The original graph had $26,475$ nodes and $106,762$ edges. In order to make this network comparable to the others, this research, chose to use the k-core, where $k = 3$, of the original network. This resulted in a graph where $|V(G)| = 1536$ and $|E(G)| = 6135$, and

a diameter of 6 and the degrees range between 1 and 427.

The focus of this research is on the modularity of networks while inferring the graphs. The networks that are considered have a variety of qualities of community partitions. The weakest is Noordin Top with a modularity of 0.30, however, if one combines the very small communities into a misfit community, one gets good 5 communities. Next, the Gnutella network has a modularity of 0.35 followed closely by the CAIDA's network with a modularity of 0.43. These two networks have much larger communities. Finally, the highest modularity is of the Power Grid, of 0.94. Since the Power Grid has many nodes of degree 2, it is much easier to partition the nodes into communities with very few edges between communities, thus increasing the modularity.

1.1 Definitions and Algorithms

The idea of the *monitor* used in the current research was introduced in the discovery for networks by Davis [3]. The monitors discover the labels of the edges and the nodes that it detects, based on the labels of the true topology inferred. The formal definition is given below.

We say that a *monitor on node i detects a node j* if $d(i,j) \leq 1$. Also, then the monitor on i discovers the label of j and the $\deg j$. A *monitor on node i detects an edge ij* if i and ij are incident, and i detects the label ij of the edge (i.e. that the monitor discovers the label of the other end node of ij).

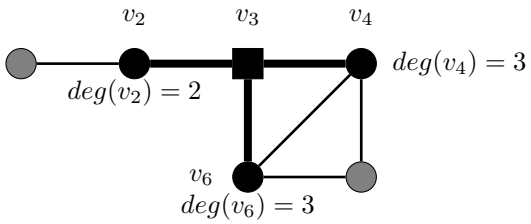

Figure 1: A monitor placed at node v_3, from [3]

Paper [3] thus defined a monitor to be a sensor that is placed at node i that detects its incident edges and its neighbors as shown in Figure 1. That is, for example, if each of two monitors i and j will individually detect node k, they will identify that it is the same k.

The *Undiscovered Neighbor Degree Discovery* (UNDD) algorithm is used to light the networks: http://faculty.nps.edu/rgera/projects.html [4]. The algorithm uses the above described monitors to light up the network as follows: A monitor is placed randomly on an initial vertex x. Then the monitor discovers $N(x)$, which is the set of neighbors of x. Moreover, a list of all discovered nodes in graph G is created and labelled $L(G)$. For each $y \in N(x)$, the algorithm queries y for the count $|N(y) - L(G)|$. This value $|N(y) - L(G)|$ counts the number of undiscovered nodes adjacent to y, which was defined as undiscovered neighbor degree of y and formally define as

$$DoUN(y) = |N(y) - L(G)|, \forall y \in N(x).$$

Recall that the relative complement of a set X with respect to Y is $Y \backslash X = \{x \in Y \wedge x \notin X\}$. This is used in **Algorithm 1** as presented using pseudo code for $V(G) \backslash MonitorList$. For further details, one can refer to [8]

The line labeled ** is to preserve a 'non-complete' information setting. That is, one can't update all the neighbor's (i.e. 2-distant nodes) fake degree, because one may not be aware of them in the inferred graph. This should actually have no impact though on the algorithm operation (except, perhaps under certain fringe cases of restarting) since the algorithm only ever chooses from nodes already in the inferred graph (i.e. seen nodes). It is valuable to point out as the line and condition are removed for the upper-bound algorithm a complete information about the network is still preserved.

2 Methodology

Running the UNDD algorithm through these four networks created partial networks at each monitor placement. The Louvain community detection method is used to determine communities within the network. Modularity is used to quantify the community detection during the inference. It returns a value between -1 and 1 and it shows how well partitioned in communities is the current network, compared to a similar size random network [9]. At each step of network discovery the modularity is also calculated for each network.

Normalized Mutual Information (NMI) that compares two community partitions (as they are discovered) of graphs [1] is used. NMI will return values between 0 and 1. A value closer to 1 means the two partitions are closer to identical. A value closer to 0 signifies two partitions are less similar [10]. Note that NMI values are related to the partitions of a network and hence it would be inappropriate to compare NMI values from the CAIDA network to NMI values from the Power Grid. Rather NMI is better used as a measure of how similar the partitions of the network are to each other.

A limitation of NMI, specific to this research, is comparing partitions of networks with different size

Algorithm 1: Undiscovered Neighbor Degree Discovery with Restart [5]

Input: G original graph
Output: G_{inf}, a set of nodes and edges that form the inferred graph

$NextNode \leftarrow$ a random node from G

repeat

  $ListOfMonitors$.add($NextNode$)
  G_{inf}.add($NextNode$)
  $UpdateList$.add($NextNode.neighbors$)
  foreach $NextNode.neighbor$ **do**
    $UpdateList$.add(($NextNode.neighbor$).$neighbors$)
    if $NextNode.neighbor \notin G_{inf}$ **then**
      G_{inf}.add($NextNode.neighbor$)
1     $RestartCounter = 0$
    else
      $SeenCount(NextNode.neighbor) += 1$

  foreach $NextNode.edge$ **do**
    if $NextNode.edge \notin G_{inf}$ **then**
      G_{inf}.add($NextNode.edge$)

  foreach $UpdateList.node$ **do**
**   **if** $node \in G_{inf}$ **then**
        $node.FakeDegree =$
          length($node.neighbors \backslash G_{inf}$)
        (i.e. # of undiscovered neighbors)

2   **if** $RestartCounter == 2$ **then**
3     $NextNode \leftarrow$ a random node from $V(G) \backslash ListOfMonitors$
4   **else**
      $NextNodes \leftarrow$
        MaxFakeDegreeNodes($G_{inf} \backslash ListOfMonitors$)
      if length($NextNodes$) > 1 **then**
        $NextNode$=node with minimum of $SeenCount(NextNodes)$
      else
        $NextNode = NextNodes$
5     $RestartCounter += 1$

until 50% of nodes have monitors

node list. If there is a great disparity between the size of the node lists, the NMI value might be misleading. This is due to the construction of the confusion matrix. The confusion matrix will only account for nodes common to the two graphs being compared. Thus, any nodes in the ground truth, but not in the inferred region, will not affect NMI. An example of this shortcoming is in the early stages of discovery using UNDD. The comparison partition might only contain a few dozen nodes, while the ground truth node list has thousands of nodes. The NMI calculations for this early step will only account for the few dozen in the comparison partition, and ignore the thousands in the ground truth that have not been discovered. Therefore the NMI value from those early steps is less informative, has the potential to be misleading.

For this reason, consecutive discovery steps using NMI are evaluated. This is named as the NMI derivative. The NMI derivative is calculated by comparing the current step of discovery to the subsequent step of discovery. In this methodology, the comparison partition can be thought of as the current step of discovery. The ground truth partition is the subsequent step of discovery. This will minimize the impact of different node list sizes on the NMI value. Obviously, once the rate of discovery of nodes decreases, the necessity to calculate the NMI derivative is minimized. However there is an aspect of UNDD that makes the NMI derivative useful.

In UNDD, there is a potential that not all nodes (and therefore edges), will be discovered. The most common reason for this is having multiple disconnected components in the network. If the network is not connected, UNDD will never discover the components outside of the component the algorithm originates in since no restarts in UNDD algorithm is chosen to use. Additionally, there are edges between nodes that never have a high enough Undiscovered Neighborhood Degree to have a monitor placed on them, and therefore are never discovered. All of these possibilities contribute to potential for information in the network that can not be found in UNDD. This means that UNDD might never truly discover all of the network. There are no such limitations in the calculations of modularity when partitioning the network. While this potential difference between modularity and full discovery in UNDD represents a small percentage of the overall network, networks with low modularity, can impact the NMI values. To avoid this, the final step of discovery in UNDD as the ground truth is used. Using this technique of calculating NMI Derivative, disconnected components were prevented from impacting the calculations of NMI.

3 Results and Analysis

The values described above for each of the snapshots are graphed and compared to each other with respect to five major characteristics: percent nodes discovered, percent edges discovered, modularity of the snapshot, derivative NMI comparison, and NMI comparison to the ground truth network.

There is a general consistency: the vertices of the

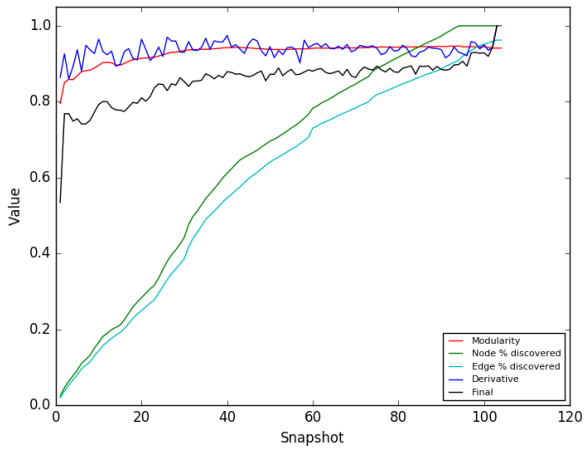

Figure 2: Comparing Snapshots for the Power Grid. Black - NMI compared to ground truth. Blue - Derivative NMI. Red - Modularity. Green - % Nodes Discovered. Cyan - % Edges Discovered.

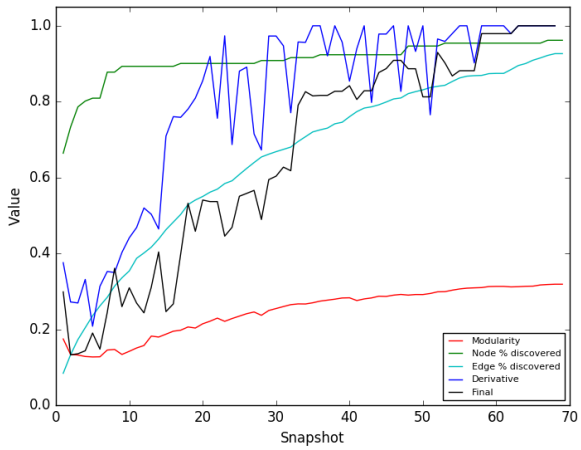

Figure 3: Comparing Snapshots for Noordin Top. Black - NMI compared to ground truth. Blue - Derivative NMI. Red - Modularity. Green - % Nodes Discovered. Cyan - % Edges Discovered.

networks are discovered quickly, and this is expected from the design of UNDD that discovers the hubs quickly. This results in fast discovery of the nodes and edges. This does not equate to an equally fast and accurate picture of the community structure. As more of the network's edges are discovered, our picture of the community structure generally evolved.

The Power Grid had the highest modularity, and therefore the community structure is discovered and remains stable early in the discovery process (Figure 2).

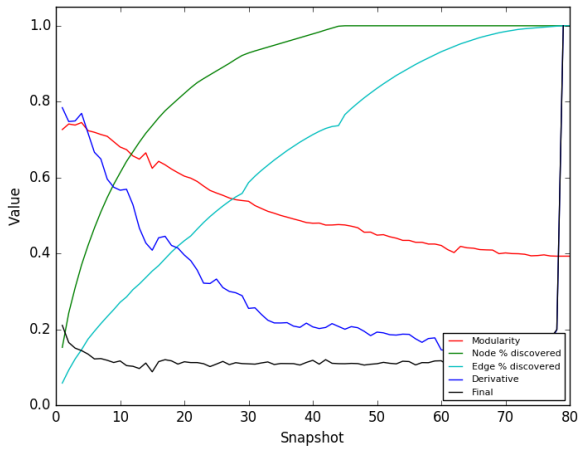

Figure 4: Comparing Snapshots for Gnutella. Black - NMI compared to ground truth. Blue - Derivative NMI. Red - Modularity. Green - % Nodes Discovered. Cyan - % Edges Discovered.

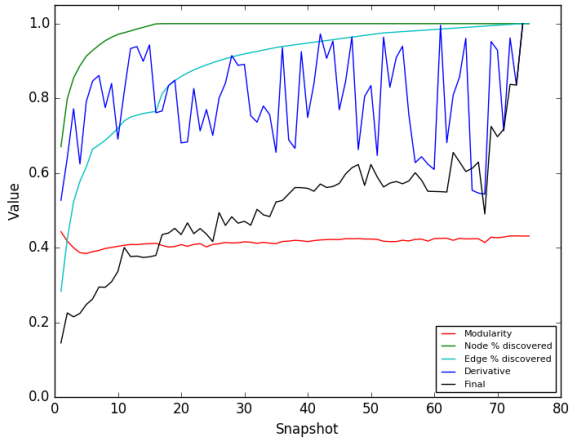

Figure 5: Comparing Snapshots for CAIDA. Black - NMI compared to ground truth. Blue - Derivative NMI. Red - Modularity. Green - % Nodes Discovered. Cyan - % Edges Discovered.

The Noordin and CAIDA networks follow very similar trends, and they are both social networks (one being a dark network). As more of the edges are discovered, the accuracy of the community structure improves as seen in Figure 3 and Figure 5.

One difference between the Power Grid and Noordin or CAIDA is in the construction of the network. The Power Grid is man made, and lacks redundancy or hubs. The network has many vertices with few edges, essentially creating many paths with in the network.

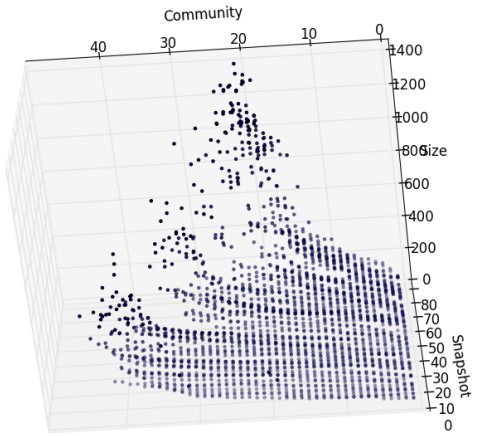

Figure 6: Comparing Community Sizes for the Gnutella network

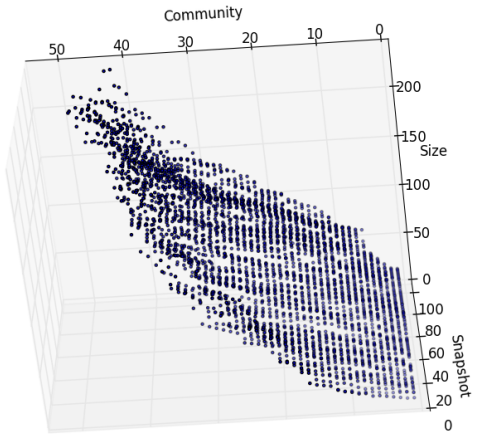

Figure 7: Comparing Community Sizes for the Power Grid network

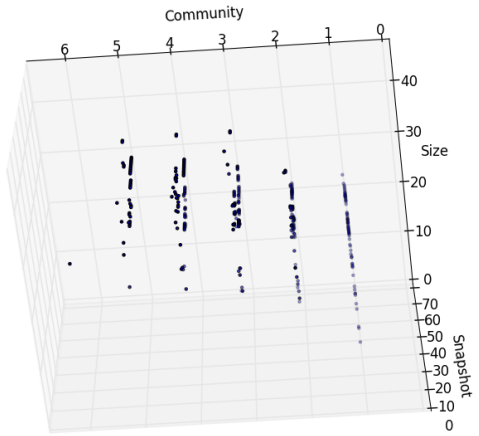

Figure 8: Comparing Community Sizes for the Noordin Top network

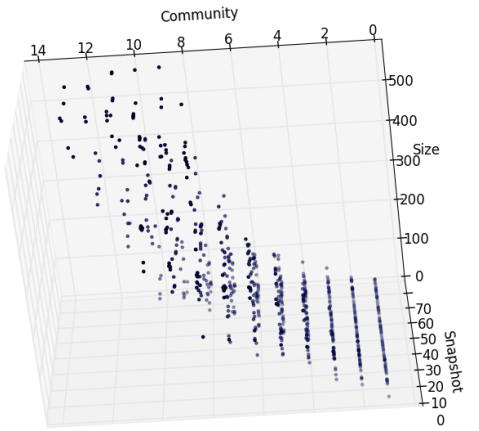

Figure 9: Comparing Community Sizes for the CAIDA network

This network has a very stable community structure, and communities are discovered as soon as they are encountered. The CAIDA and Noordin networks are a technical and dark social network, respectively. Their modularity is not as high as the Power Grid, so more of the network must be discovered in order to determine the community structure. The CAIDA network has a better community structure earlier in discovery than the Noordin network, and this is due to the fact that the Noordin network is a dark network. As stated earlier, a dark network will develop covertly, purposely making community detection more difficult.

The obvious outlier in these four networks is the Gnutella network whose results are shown in Figure 4. One might assume that since Gnutella is both a technological and somewhat social network, it would be have similarly to the CAIDA or Noordin networks, but instead, its propensity to keep users anonymous also prevents reliable community detection. Additionally, it is designed in such a way that strong communities are not created, by distributing the connections between users in an attempt to lower the load on servers. Instead of an increasing NMI as in the other three networks, NMI decreases and remains low as monitors are placed. Part of the reason the NMI remains low for so long is that the communities change drastically as more of the network is discovered, see Figure 6. This pattern of community changing as each monitor is being placed, matches the goal of the users in the network to not be obviously identified and clustered. The other three

networks have relatively little change in the number of communities, and the sizes increase in a relatively linear fashion.

The Gnutella network is drastically different. Both the size and number of communities change significantly as monitors are placed. The differences are evident when viewed in a graph, as in Figure 6 compared tot the other three networks in Figures 7, 8 and 9. This indicates there are not clearly defined partitions of communities in the Gnutella network. The vertices are closely related, and with every edge discovered there is significant alteration of the community partitions. The variation on the size and count of community in consecutive snapshots is what keeps the NMI low, and supports the good blending of the users in order to protect their identity.

4 Conclusion

Different network structures lend themselves differently to the method of discovery. Modularity alone is not sufficient to determine whether a step in the inference provides good community detection comparable to the final inferred network. Prior knowledge of the structure of the network that is being exploited or discovered is critical to knowing how accurate the partial representation is at any given step of discovery. For example, the inferred structure of the Power Grid was very accurate at all levels of discovery, whereas the inferred structure of the Gnutella network was inaccurate until the entire network was discovered. This leads to believe that by knowing what type of network is being discovered, one can make a more educated assessment of how many monitors are needed. However, more networks should be analyzed to answer this question.

Also of importance is the fact that the inferences are limited to how well the current discovery partitions match the full network partitions. A high NMI value simply means that the partitions in the first network match the partitions in the second network. But since any nodes not common to both networks are ignored completely in the calculation, these values may not be accurate. An extension that takes into account the number of nodes not used in the computation would be beneficial. Therefore, when the optimal amount of monitors to be placed in a undiscovered network is determined, one must understand that it can only be determined into which communities the discovered nodes fall, and not the structure of the entire network.

Acknowledgement The authors of this paper want to thank the DoD for partially sponsoring the current research.

References

[1] LNF Ana and Anil K Jain. Robust data clustering. In Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Soc. Conf., volume 2, pages II–128. IEEE, 2003.

[2] Center for applied internet data analysis. http://www.caida.org/home/, January 2016.

[3] Benjamin Davis, Ralucca Gera, Gary Lazzaro, Bing Yong Lim, and Erik C Rye. The marginal benefit of monitor placement on networks. In Complex Networks VII, pages 93–104. Springer, 2016.

[4] Ralucca Gera. Network Discovery Visualization Project: Naval Postgraduate School network discovery visualization project. http://faculty.nps.edu/dl/networkVisualization/, July 2015.

[5] Ralucca Gera, Nicholas Juliano, Brittany Raynolds, and Karl R. B. Schmitt. A comparison of inference algorithms in lighting up networks. Preprint, 2016.

[6] Ralucca Gera, Ryan Miller, and Scott Warnke. Multilayered terorist networks. 2016.

[7] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

[8] Mark Newman. Networks: An Introduction. Oxford University Press, Inc., New York, NY, USA, 2010.

[9] Mark EJ Newman. Modularity and community structure in networks. Proceedings of the national academy of sciences, 103(23):8577–8582, 2006.

[10] Günce Keziban Orman, Vincent Labatut, and Hocine Cherifi. On accuracy of community structure discovery algorithms. arXiv preprint arXiv:1112.4134, 2011.

[11] Jörg Raab and H Brinton Milward. Dark networks as problems. Journal of public administration research and theory, 13(4):413–439, 2003.

[12] Yizhou Sun, Jie Tang, Jiawei Han, Manish Gupta, and Bo Zhao. Community evolution detection in dynamic heterogeneous information networks. In Proceed. of the Eighth Workshop on Mining and Learning with Graphs, pages 137–146. ACM, 2010.

[13] Duncan J Watts and Steven H Strogatz. Collective dynamics of small-worldnetworks. Nature, 393(6684):440–442, 1998.