Reaction Paper # 1

Mohammad Elkholy  
Computer Science and Engineering  
 The American University in Cairo  
 New Cairo, Cairo, Egypt  
moelkholy@aucegypt.edu

Chart, waterfall chart

Description automatically generated

SUMMARY

The assigned papers this week covered two important concepts in the topology of complex networks. The first paper by Newman and Girvan named *Finding and evaluating community structure in networks* (Phys. Rev. E69, 2004) discusses different methods to find an underlying community structure in a given network. This paper will be referred to as the CS-paper for brevity’s sake. The second paper by Xiang Et al. named *A unified method of detecting core-periphery structure and community structure in networks* proposes a general method to find meso-scale structures inside networks. This paper will be referred to as the CP-paper for brevity as well. Both papers discuss ways to find structures inside a network as well as ways to measure the goodness of fit of the output of the proposed algorithms.

Before we get into the algorithms however, we need to distinguish between community structure and core-periphery structures. A community structured network is defined as one where there are sets of nodes (might be overlapping) that are densely connected internally. A core-periphery structure on the other hand is a set of nodes where some of them are densely connected (core nodes) and the rest are loosely connected to the core nodes or other loosely connected nodes (periphery nodes). They are not mutually exclusive, and some networks can contain a mixture of them, or several of the same type. The adjacency matrix representations of several of them were shown in the CP-paper (Fig.1) and relationships between the different structures were discussed:

* Single CP-structure implies absence of community structure.
* A network with several CP-structures often implies a community structure.
* Presence of community structure does not imply the existence of multiply CP-structures.

With definitions and relationships established, I will now discuss the CS-paper in more detail. The CS-paper proposes a new way of determining the communities present in complex networks. They used divisive methods for the clustering of the different communities in networks instead of agglomerative ones as they are not consistent in their results. The algorithm used in the paper can be summarized as follows:

1. Calculate “betweenness” score using one of the different methods provided for all edges
2. Find edge with highest score and remove it.
3. Recalculate scores for remaining edges
4. Repeat from step 2.

Three methods were discussed to assess the betweenness score of edges. The idea of all the methods is to find a score that favors inter-community edges over intra-community edges. The simplest of them involved finding the shortest path between each pair of vertices and finding the edges with the most visits from all the paths. The other two methods were essentially two different versions of the same thing, one involving random walks across the graph and the other modeling the graph as a resister network and finding the difference in voltages using kirchhoff’s laws. The shortest path method was favored as it has a smaller time complexity compared to the other two. The results between the different methods were very close for the most part, and thus one is inclined to use the shortest path one. The goodness of fit of the results of the algorithms is then assessed using a *modularity* measure *Q* (Fig.2). This quantity measures how dense each community is connected compared to a graph with the same communities with random edges in each community. This quantity is calculated for each split of the network and graphed against the average number of inter-community edges per node, the peak height indicates a satisfactory split. The authors then end the paper by applying their new proposed methods to existing networks with known communities as well as generated ones.

Text

Description automatically generated

FIG. 2. is an entry in the symmetric matrix of size where is the number of communities and is the fraction of edges that connect community to community .

In the CP-paper the authors propose a method to detect both kinds of structures in a given network, as well as identifying sub-CP structures, and finding overlapping nodes between structures. The method can be broken down into the following steps:

1. Re-rank nodes in the graph using a centrality score.
2. Introduce new node to an originally empty set U in decreasing order of rank.
3. Calculate the regional density of the new set and plotting it on a graph against the rank of the nodes.
4. Repeat from step 2 until the originally empty set is full.
5. Generate the Core and periphery sets of nodes.

In the paper, they used the node with the highest centrality value as the first node as it is more likely to be a core node, they also mentioned that the centrality measure did not affect the results that much. More nodes are then added depending on the number of connections they have with the set U. If two or more have the same number of connections, the node with highest degree is chosen. Once the reranking is done, the regional density is calculated for the subsets generated of U. If the regional density of a certain subset is where is a threshold value that signifies the strength of the structure, and the size of the subset is greater than or equal to some value where is the number of elements in the subset, then a core is found. If the regional density is greater than for two consecutive nodes, they are merged into one community. CP structures usually have huge differences between maximum and minimum values in the region density graph as well as presence of many low values for regional density. This is due to the presence of many peripheral nodes compared to core ones. The peripheral nodes, active and overlapping nodes are found by classifying the nodes into different levels. The classification works as follows:

1. Assign core nodes to class 0.
2. Assign periphery nodes that are connected to core nodes as class 1.
3. Expand to periphery nodes connected to previous periphery node class.

In steps 2 and 3, peripheral nodes could have the same number of connections to the core nodes of different structures, those are called active nodes. Active nodes are then reallocated to the structure with which is has the most connections to. The active nodes that have the same number of connections to different structures are the overlapping nodes. The authors then test this method on different known datasets while varying values for and to explore different substructures in the datasets. The authors then end the paper by comparing their algorithm’s performance with other algorithms for two metrics:

* Detection of CP structure
* Detection of overlapping communities

Their algorithm performed better across all types of networks compared to other algorithms in the detection of structures, only losing to the Two-step method (which is not good at detecting single CP structure in networks as well as needing much more computational power). Their algorithm outperformed all other algorithms in detecting overlapping structures.

**COMMENTS**

**CS-paper**

The CS-paper was particularly interesting to read, particularly as it introduced me to a new way of thinking about the importance of certain edges in connecting different communities together. While the paper mainly worked with undirected and unweighted graphs of networks of a single nodes type, it is very easy to see generalizations to more complex types of networks. The directed or undirected nature was discussed in the paper and was deemed not significant to the calculations. Using the shortest path method allows for accommodation of weighted graphs where known shortest path algorithms can be used. One can also note that shortest paths in a weighted graph will most likely be unique. It will be very unlikely for the sum of weights across two different paths between a pair of nodes to be the same. This makes it easier to compute the betweenness scores in the shortest path “tree” as it will no longer contain as many cyclic components compared to an undirected one. A method that takes into account the type of node could probably be used, although I have not been able to think of one yet.

The results of the algorithm being used with different betweenness scores were interesting. Random walk results were very comparable to the shortest paths one. Random walks however were not tested on larger datasets which is something worth considering as the result will most likely differ a significant amount as nodes increase in number. It is however worth noting that no results were shown to compare this method with agglomerative clustering ones.

The modularity score proved to be a good measure of goodness of fit as peaks coincided with known divisions of communities across different datasets. The algorithm would be very hard to adapt to networks with values of n larger than 10000 due to its time complexity.

**CP-paper**

The CP-paper was also very interesting as it took a completely different approach to this problem and even attempted to find a general model that is applicable to different types of structures. The authors attempted to construct the different structures from scratch by using the properties they know about the graph. Their method was very straightforward to apply and computationally similar to the one discussed in the CS-paper with the added benefit of finding overlapping nodes.

This method, however, was only discussed and applied for undirected, unweighted graphs of a single node type. Thus, it is not as generalizable as one might first think. Hypothetically, a node with multiple weighted edges might be more important than one which connects more nodes. A different centrality measure would have to be used to account for this and was not explored. The directed nature of the graph would also change the way we want to think about and calculate the regional density and is thus worth exploring.

Its performance on the given datasets however was very interesting. Varying the parameters and could reveal new structures and substructures in a network, even those with known truth values, which might be worth exploring as new findings are possible while studying these substructures. It is also a downside, as we would need to find the appropriate values of and that best describe the network. The comparisons of the results with other algorithms were unfair as the best and values were intentionally used.

The results for the detection of overlapping nodes were interestingly weird for generated networks. Other algorithms faired just as well on generated networks, while failing to do better on real networks, which is worth investigating.

**CONCLUSIONS**

Each method proves to be useful for different reasons. The first method made no attempt to figure out the overlapping nodes problem whereas the second method shines in that problem (as evident in the karate club network where node 3 was labelled incorrectly in the CS-paper but detected to be overlapping correctly in the CP-paper). Both methods are very useful for exploring datasets and trying to make meaning out of complex graphs. They both provide great insight into the types of structures, their size and number. One is generalizable to all kinds of networks and the other is good at solving certain problems for certain graphs, and each shine in their own way

REFERENCES

[1] Newman, M. E., & Girvan, M. (2003). Finding and evaluating community structure in networks. *arXiv*. https://doi.org/10.1103/PhysRevE.69.026113

[2] Xiang, Bing, Bao, Zhong, Ma, Chuang, Zhang, Xingyi, Chen, Han, and Hai Zhang. "A Unified Method of Detecting Core-Periphery Structure and Community Structure in Networks." *arXiv*, (2016). https://doi.org/10.1063/1.4990734.

Conference Name:ACM Woodstock conference

Conference Short Name:WOODSTOCK’18

Conference Location:El Paso, Texas USA

ISBN:978-1-4503-0000-0/18/06

Year:2018

Date:June

Copyright Year:2018

Copyright Statement:rightsretained

DOI:10.1145/1234567890

RRH: F. Surname et al.

Price:$15.00