Scientific Modeling Computer Laboratory

Project: Time Evolving Networks

Third Bi-weekly Report

Ádám Gergely Szabó

11th of April, 2022

1 Introduction

This project is about exploring MTMT's co-partnership network, which evolves in time as more publications get submitted to the site. MTMT's main goal is to host a site that maintains high quality publications, meaning the submitted works are often checked and rated for their quality. The data stored by the site is publicly available, thus data can be gathered from the site without registration.

This project's goal is to explore the co-partnership network of MTMT. In this work, we will look at this network in different given states or its subsets, see how it develops in time, calculate different central indicators, apply different group searhcing methods and embeddings. The work will be mostly done in Python3 language that will be utilized in the Jupyter Notebook framework.

2 Progress

2.1 Assortativity

Previously, it was shown that the co-authorship network is disassortative by fitting a power-law function on the degree correlation results. It can be seen that it is inaccurate: for lower values, it shows a positive exponent, while for higher degree, it shows negative exponent. This change is a directs as to the problem: the degree correlation was calculated from the degrees given by networkx graph object. This object generates an unweighted and undirected network which introduces structural disassortativity.

In the analysis, the following will be compared:

1. The original network
2. The network that has 2 edges randomly swapped, only 20% of the edges are replace
3. Randomizing the network by deleting 20% of its edges and creating new ones.

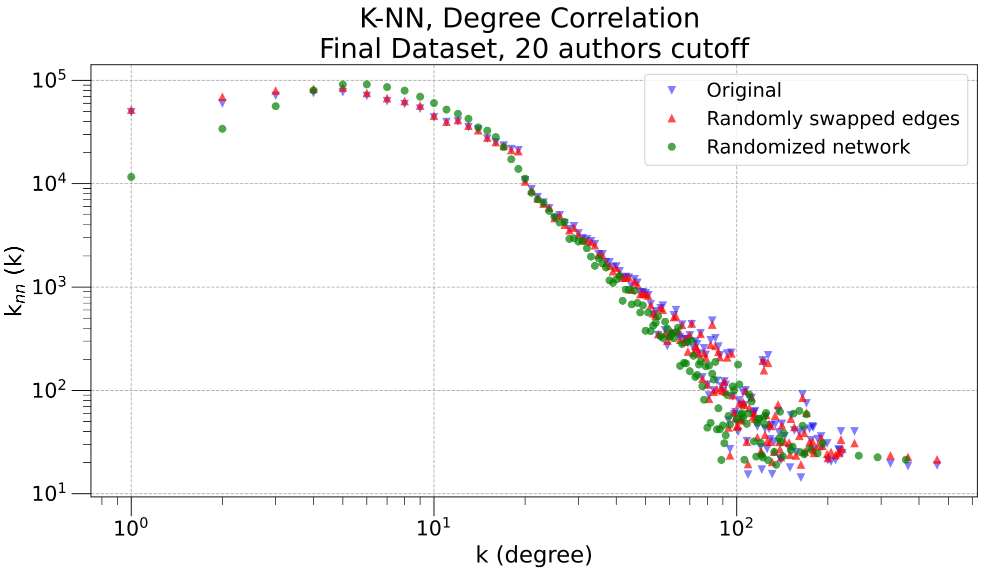

Figure 1: Degree correlations. Comparing the original results and 2 ways of randomizations

It visible now that the disassortativity is introduced by using the results of the simple graph like graph.

2.2 Betwenness

Betweenness is another centrality measure like average shortest path. By definition:

$$g = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{1}$$

where σ_{st} is the amount of shortest paths between nodes s and t, while $\sigma_{st}(v)$ is the amount of shortest paths passing through v, where v is not an endpoint. As it is visible, this is an expensive calculation, as it scale with O($|V| \cdot |N|$) for undirected graphs, where $|V|$ is the amount of vertices and $|N|$ is the amount of nodes in the graph. Luckily, Networkx implementation of betweenness_centrality has an argument, where we can choose how many nodes should be used for calculation. In my case, I choose 2000, as it is a significant part of the network. (4)

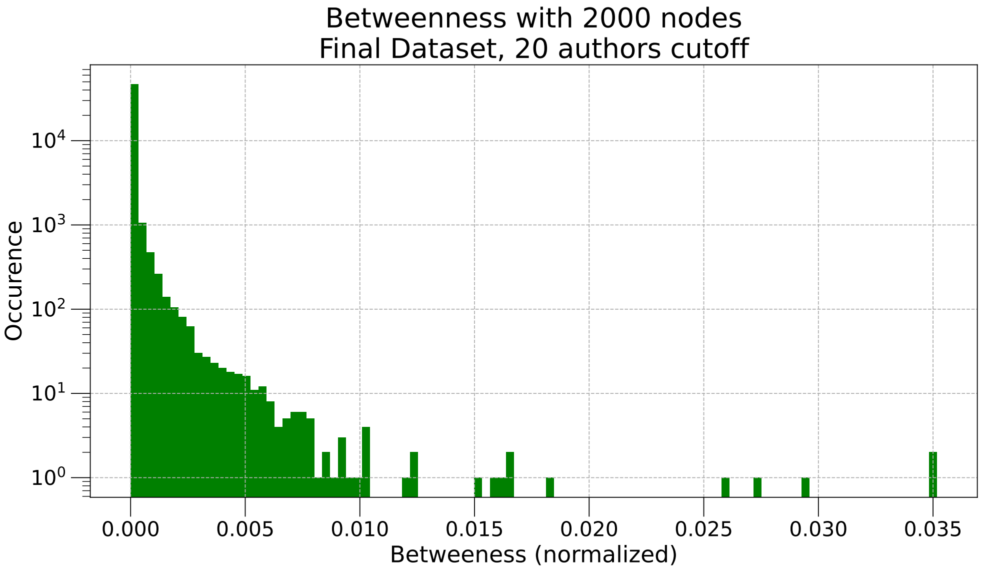

Figure 2: Betweenness calculated for 2000 nodes. It is visible that some nodes are used as 'highways' for shortest paths.

2.3 PageRank

PageRank was developed in 1998 and was used to rank pages according to their significance. Originally was used for ranking webpages, in which case, the hyperlinks on one page are used for to rank other pages by using the quality and number of hyperlinks, but it has to be accounted that some pages are less significant and their vote means less. Generally:

$$PageRank(i) = \frac{1-d}{N} + d \sum_{j \in M(i)} \frac{PageRank(j)}{L(j)} \tag{2}$$

this can be generalized to graphs and undirected graphs. The PageRank calculation was done by using d={0.1,0.6} damping parameters.

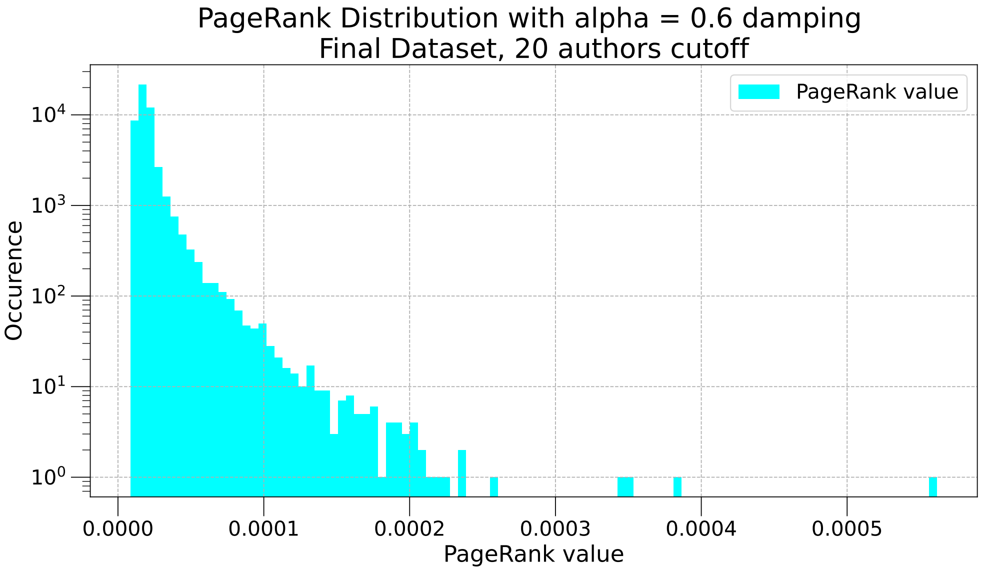

Figure 3: PageRank calculation. d=0.6

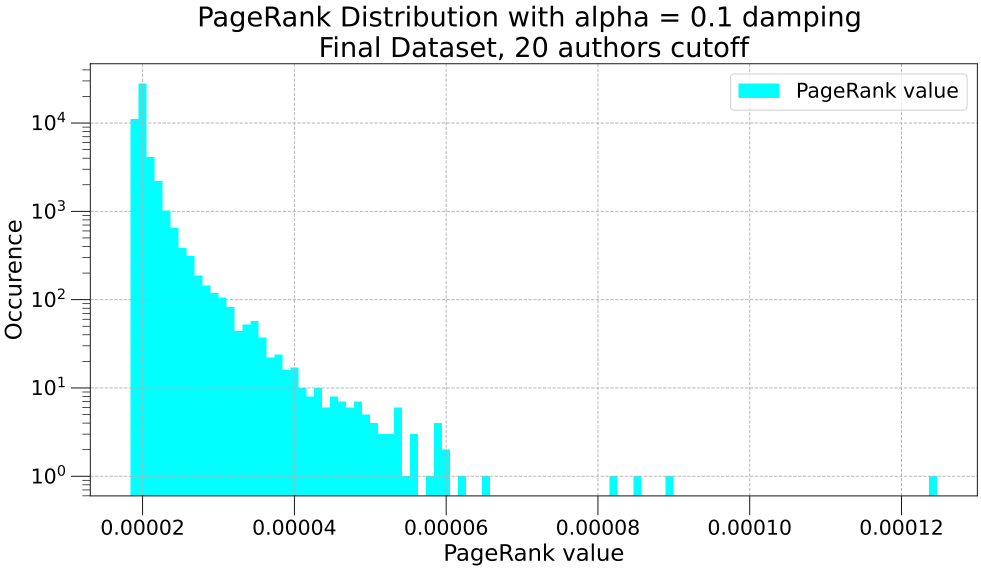

Figure 4: PageRank calculation. d=0.1

As it seems, they look somewhat similar: due to the high number of nodes and a somewhat low maximum degree, no node gets high pagerank. Most of the nodes are not that significant, according to the PageRank value.

2.4 Motifs

Finding motifs in a graph is an entrance level group searching method. Motifs are defined graphs, that we need to define, but these are graphs that have low amount of nodes and given amount of edges. In search the for motifs, we not only need to check how many we find in a given graph, but it is advised to use a random graph to compare our results to. With this, the z-score can be defined:

$$z = \frac{\langle m_i \rangle_g - \langle m_i \rangle_{rand}}{\sigma_{rand}} \tag{3}$$

where $\langle m_i \rangle_g$ is the number of times that motif$iwerefoundingivengraph, \langle m_i \rangle_{rand}$ the average amount of motif$iwerefoundinrandomnetworksand\sigma_{rand}$ is the standard deviation of finding motif i in random networks.

2.4.1 Problems with motifs

Smaller motifs are easy to find (the 2 motif that consists of 3 nodes), but larger ones require much more time to be found. This makes it quite time consuming to find further motifs and brings light to networkx's capabilities.

3 Discussion

In this report, I have shown that the previously calculated degree correlation was incorrect and given reason to why such thing can happen. Using simple graphs are most of the time are sufficient enough to show different attributes of a network, while with the degree correlation, a limit has been reached. This limit can be surpassed by allowing multi-connected nodes, which would result to the correct degree correlation and the correct exponent of which the correct assortativity can be assumed.

The problem with motifs are the following: the more node a motif has, the more variation does exist and the more mappings are found for the same motif. For example the same open triangle can be found twice, while the same closed triangle can be found six times.

References

[1] Albert-László Barabási. Network science. http://networksciencebook.com, 2012.

[2] Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. ”exploring network structure, dynamics, and function using networkx, in proceedings of the 7th python in science conference (scipy 2008)”, 2008.