Maniell Workman

Social Network Analysis

Betweenness Centrality

**Tools needed**

This implementation of network analysis was created with python 3.7. The packages imported were networkx, Matplotlib.pyplot, and csv are necessary for the operation of the simulation. The packages scipy, pandas, and sys are included for supplementary purposes such as debugging and improvements. IDE in which this was developed is Visual studio code. Other environments such as Jupyter, Python terminal, and any with Python 2 or 3 capabilities are sufficient.

**Abstract**

Social network analysis is a focus area of Network analysis dealing with interpersonal connections. In network analysis, the typical topography is a hypergraph structure comprised of points connected by edges. Attributes such as node degree, edge density, and edge weight are aggregated to reveal network structure both locally and overall. In social networks, nodes can be comprised of people, hashtags, discussion topics or any quantifiable point in the structure. Edges are connections between these nodes. These connections take form as data transmission, shared people between the groups, or a tangible affect the nodes have between each other. Community detection is a challenge of social network analysis. One of the most popular and powerful algorithms for community detection is the Newman Garvin algorithm. This algorithm removes edges which are determined to be insignificant to the structure of the network. In this analysis, Betweenness Centrality is used as the basis of edge elimination. Betweenness centrality is the measure of how often a node is included as a part of a shortest path in the network.

**Implementation**

The function Edge\_to\_remove is designed to create a data structure which organizes edges and their betweenness centrality together. The initial call to edge\_betweenness\_centrality stores these values in a dictionary. Because a dictionary is not a suitable data structure to be iterated over, the dictionary is initially converted into list. Using a for loop, the elements are appended into a new list creating a two dimensional array that can be processed easily. The function btw\_centrality calls the edge\_to\_remove function to get the analytics of the graph. Using a cutoff value of the user’s choice, this function eliminates edges that have a higher betweenness value then the cut off point. The main function builds the graph based on the data list stored in a CSV file and graphs it.