Caleb Moore / Michael Castle

Rachael Becker

Discrete Computational Structures

6 December 2021

Community Detection in Networks

The program is an implementation of the Girvan Newman Algorithm for an undirected

graph. The program uses a graphml file to create a graph and execute the algorithm on it to

detect communities. After running the algorithm, an output file is created called "graphed.txt"

and pasting this file into the editor will create a visual output of the communities the program has

created. In addition to this file output, the program posts a single line to the console output to

describe the runtime of the community detection in seconds. The algorithm works off of a

custom implementation of a Girvan Newman Algorithm. It first analyzes the shortest path

between every pair of nodes in the graph. All these shortest paths can be used to analyze the

most commonly traversed path. The path that is traversed the most is then removed. This

continues all the way until there are no edges left. After each edge deletion a new modularity

value is calculated based on how many edges are observed in each community compared to the

network as a whole. This modularity value is noted if it is higher than the current highest

modularity value. In this case, the current structure of the graph is also noted as its communities

will now represent the best modularity observed thus far in removing edges. Storing the highest

modularity and graph corresponding with it, ensures the final output has the greatest modularity

value. A community, with respect to graphs, can be defined as a subset of nodes that are densely

connected to each other and loosely connected to the nodes in the other communities in the same

graph. The project is related to graph theory through the connections between the communities

of Nodes.