

CSCE 5013 - Graph Theory

010646947

Zack Fravel

1 Fighting Reticulovirus Avarum via Email

Exercise 1

The network analysis concept that best captures the set of infected nodes, given a node v , would be the a Breadth-First Search, $BFS(v)$. This will give us the set of nodes in the out-component of a given node v and thus, describes the set of eventually infected nodes.

Exercise 2

I used networkx to calculate the different component sizes of the simplified bow-tie structure. The process was as follows: (1) Isolate the largest strongly connected component. (2) Pick a node within the SCC at random. (3) Use the random node to calculate the number of successors (IN component) and predecessors (OUT component) from the SCC using a Breadth-First Search. Below is a diagram of the bow-tie structure. In this analysis we ignore tendrils and tubes.

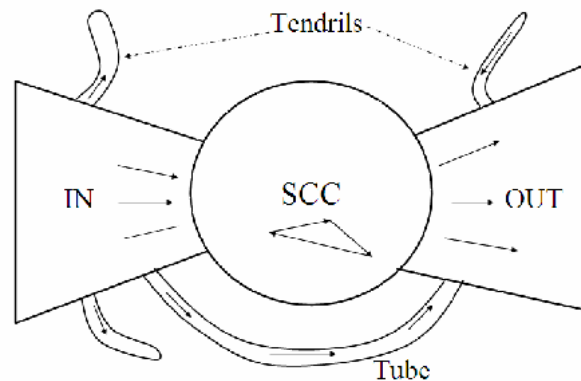


Figure 1: bow-tie network structure

bow-tie network analysis

total nodes	:	85,591
total edges	:	156,217
size of SCC (core)	:	26.72 % - 22,868
size of in-component	:	2.36 % - 2,020
size of out-component	:	41.11 % - 35,187
size of disconnected components	:	29.81 % - 25,516

Exercise 3

Using only the numbers we calculated from exercise 2, it seems to be the case that a large-scale epidemic would break out if the randomly chosen node was either in the central core, or the in-component since then the virus would definitely propagate through the out-component and cause a wide-spread infection. If we infect the out-component FROM the in-component or the SCC, we can guarantee at least 41.11%. (Needs to be at least 30%) If a randomly chosen node was in the out component, it is likely to be the case that the infection wouldn't capture the entire out-network (41.11%). The size alone doesn't capture the infection potential.

$$\mathcal{P}(\text{infection}) = (V(SCC) + V(IN))/V(G)$$

$$\mathcal{P}(\text{infection}) = (22,868 + 2,020)/85,591 = 0.2907$$

$$\mathcal{P}(\text{infection}) = 29.07\%$$

2 Fighting Reticulovirus Avarum via Twitter

Twitter Network Summary

100 Million Nodes

SCC - 40 Million

IN - 20 Million

OUT - 30 Million

DIS - 10 Million

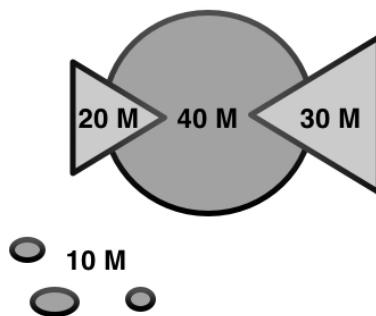


Figure 2: Twitter network structure

Exercise 4

The resulting outbreak magnitude of an SCC node being infected, $|SCC|$, will be $SCC + OUT$ since every node in the SCC is connected to every other node in the SCC and every node in the OUT component.

$$|SCC| = SCC + OUT$$

$$|SCC| = 40M + 30M = 70M$$

M = Millions of nodes

Exercise 5

I created a visual diagram of the twitter network structure (Figure 2) on the previous page. Since 10 Million nodes are disconnected from the majority of the network, the worst possible outbreak size $|W|$ we can get would be the sum of the *SCC*, *IN*, and *OUT* components. Thus...

$$|W| = 20M + 40M + 30M = 90M$$

M = Millions of nodes

Exercise 6

The most simple way to modify the network to reduce the worst-case outbreak by only removing one edge would be to restructure the *IN* component into a tree. A diagram of this idea is shown below.

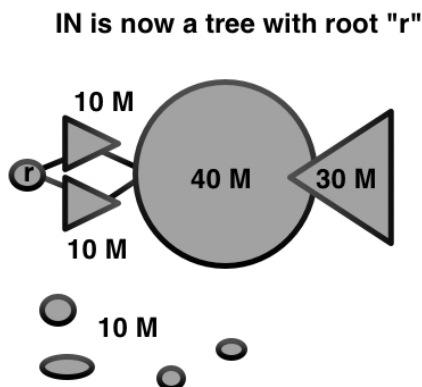


Figure 3: Modified Twitter Network structure

This way, starting from the root node, we can remove one edge that prevents outbreak to 10 Million nodes. If it were possible to restructure the entire network into a series of trees, it would provide an easy way to identify edges that limit access to large portions of the network.

The python script used to compute exercise 2 is included in the assignment package.