IE532 Final Report
Nodes Importance Analysis

Shiyuan Wang

12/16/2019

Content

Motivation

In transportation network, failure of important nodes usually results in network paralysis. In order to protect network from cascading damages, we should first identify which nodes are of the most importance, and then take measures to avoid spread of damages when those important nodes are destroyed, that is, to decrease the effect of important nodes on the whole network.

There are many ways to evaluate nodes importance, such as degree, betweenness, closeness and so on. Here I simply introduce those three methods as below.

Degree is defined by the number of nodes connected to the target node. However, degree is not always capable of recognizing some important nodes that are connected to fewer nodes but acting as bridges to connect node clusters.

$$d_i = |N(i)|$$

Betweenness is a global metric compared to degree, measuring the portion of shortest path in the graph passing through the target node.

$$b_i = \frac{\sum_{k,l \in V \setminus \{i\}} g_{kl}^{i}}{\sum_{k,l \in V \setminus \{i\}} g_{kl}}$$

Where g_{kl} is the total number of shortest paths connecting k and l,

g_{kl}^{i} is the total number of shortest paths connecting k and l passing through i.

Closeness is also a global metric, and its formulation is shown as below:

$$c_i = (\sum_j d_{ij})^{-1}$$

Where d_{ij} is from target node i to every other node j.

In this project, I would like to see the connectivity and vulnerability of a network, and thus the betweenness is a good choice to measure the node importance. We want to investigate how the nodes with high betweenness, which is recognized as the important nodes, will affect the whole network, and to see whether we can do something to the remaining nodes to maintain connectivity when those important nodes are destroyed. One extreme example can be shown as below.

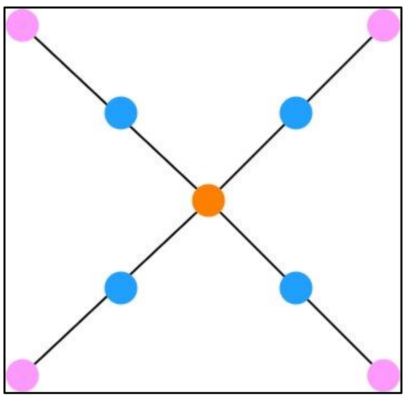

Figure 1

From this graph, we can easily tell that the orange node is with highest betweenness, while the four pink nodes are with zero betweenness. Now if we remove the orange node, this graph will become disconnected and the four remaining parts can't communicate with each other anymore.

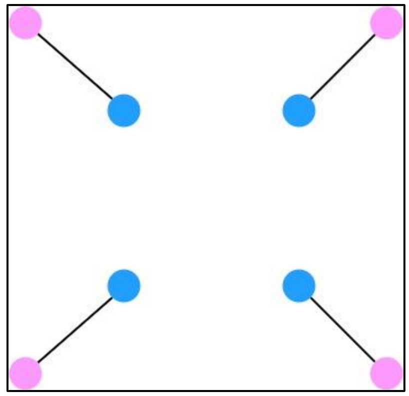

Figure 2

However, if we connect those pink nodes in a proper way, we can retrieve the connection of the network. Below shows one connection option for this situation. The green lines are newly added edges.

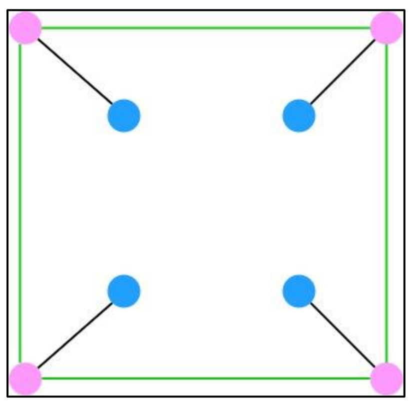

Figure 3

Data Set

I use the network data from https://github.com/bstabler/TransportationNetworks , including the arcs information and nodes' coordinates. Distances among nodes are calculated with their coordinates and I assume the roads are all in Manhattan Matrix.

Then I use networkx package in python to process and analyze those networks.

Analysis of networks

The analysis procedure is shown as below.

There are three stages in my analysis, and for each stage, I calculate the average betweenness, shortest path length and closeness. At the end, I will do some comparison among those three stages.

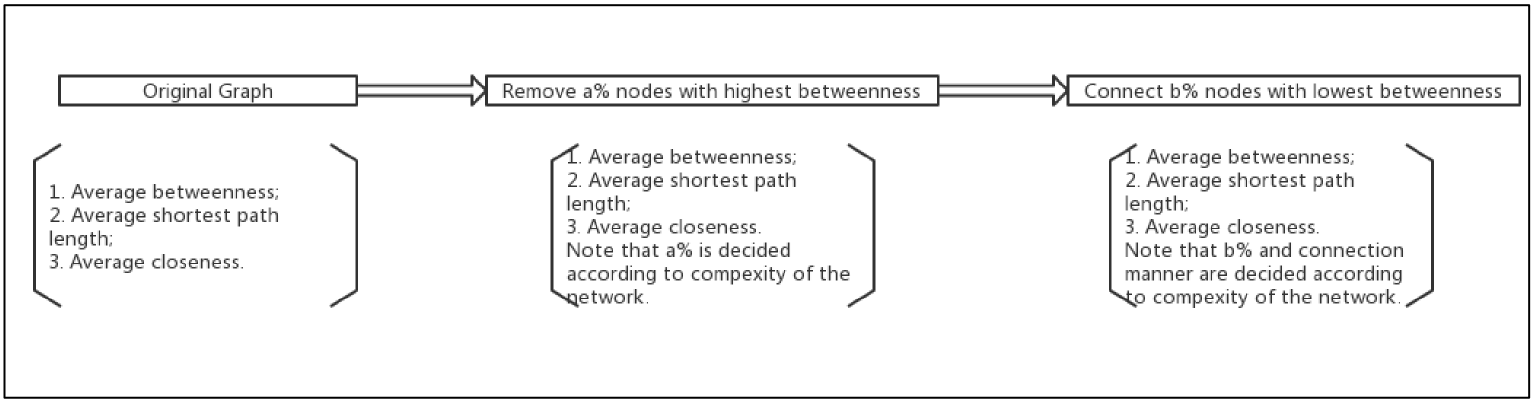

Figure 4 Analysis Procedure

1. Sioux Falls

1.1 Original Network (Network 1)

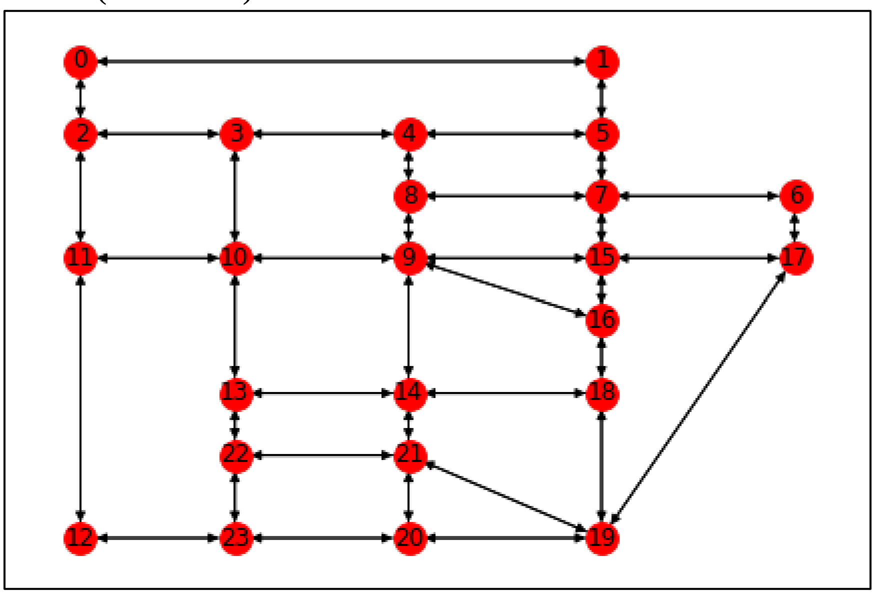

Figure 5 Original Network (Sioux Falls)

	Original network (Network 1)
Average Betweenness	0.0914
Average Length of Shortest Path	61.4783
Average Closeness	0.3363

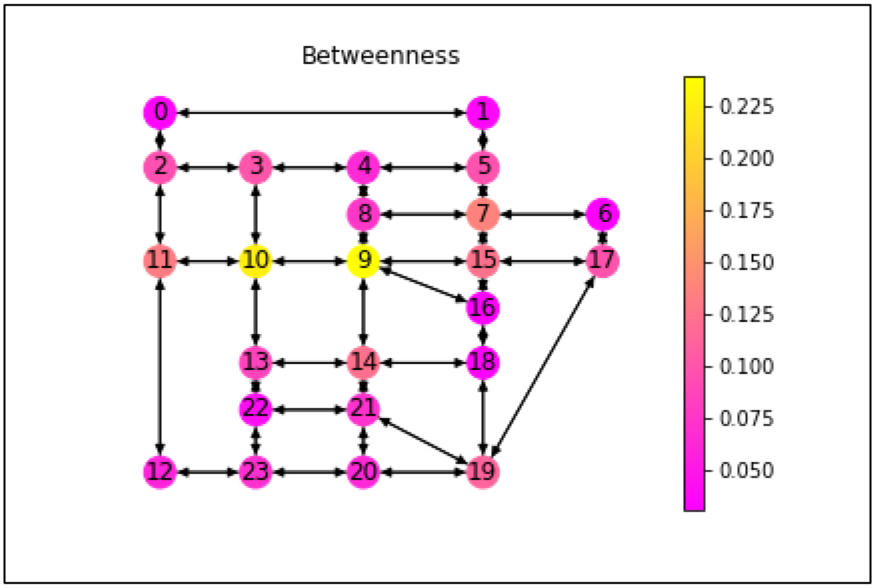

Figure 6 Betweenness of Network 1 (Sioux Falls)

1.2 Remove 10% nodes with highest betweenness (Network 2)

Since the network only has 23 nodes, I decided to take off 10% of the total nodes with the highest betweenness, and name it as Network 2. We can see from the graph that the network is still strongly connected.

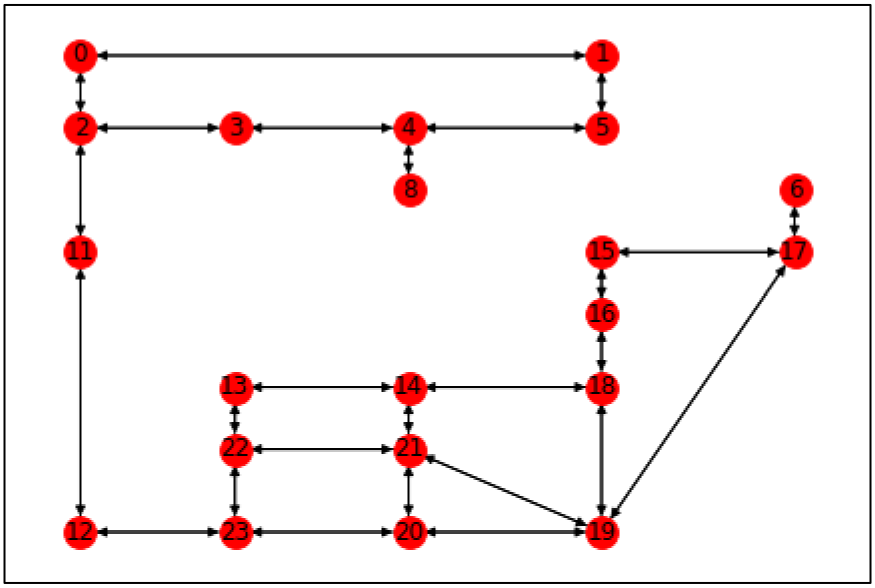

Figure 7 Network 2 (Sioux Falls)

	After Removing Nodes with Highest Betweenness (Network 2)
Average Betweenness	0.1820
Average Length of Shortest Path	109.4286

Average Closeness	0.2325

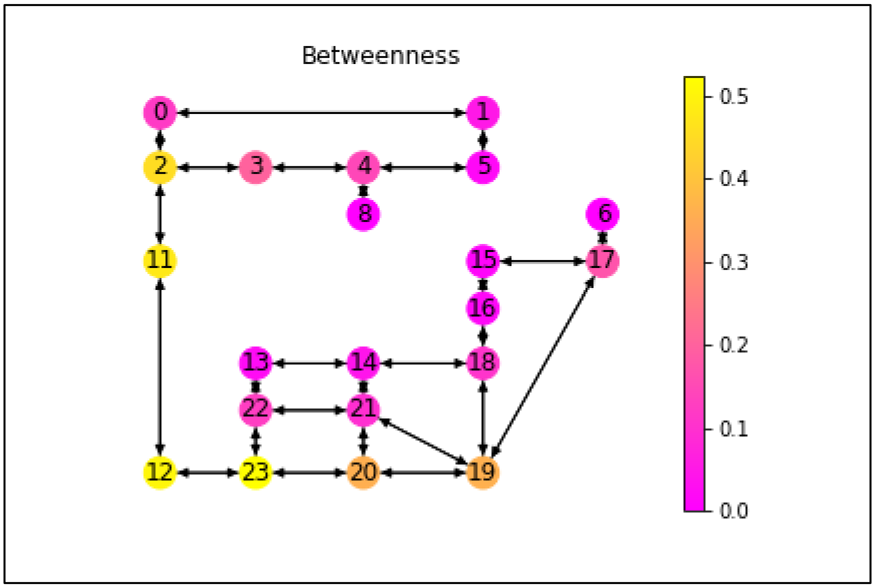

Figure 8 Betweenness of Network 2 (Sioux Falls)

1.3 Connect 10% nodes with lowest betweenness (Network 3)

For this simple network, the first 10% nodes with lowest betweenness is node 6, 8 and 15. I decided to connect them one by one in both directions, that is, 6 to 8, 8 to 6, 6 to 15, 15 to 6, 8 to 15, and 15 to 8. Network 3 is the network after I added those edges, and it's shown as below. The newly added edges are in the red circle.

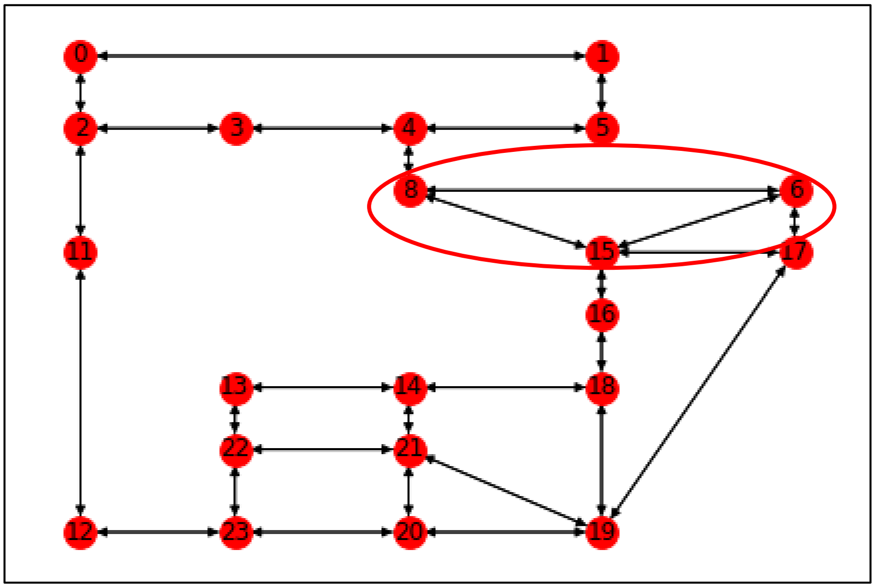

Figure 9 Network 3 (Sioux Falls)

	After Connecting Nodes with lowest Betweenness (Network 3)

Average Betweenness	0.1308
Average Length of Shortest Path	76.7619
Average Closeness	0.2902

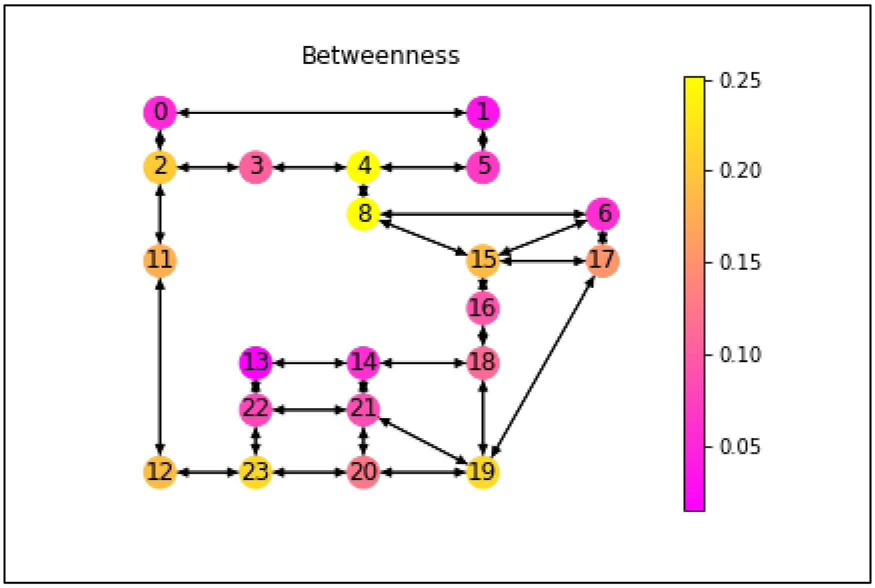

Figure 10 Betweenness of Network 3 (Sioux Falls)

1.4 Comparison

Percentage change is calculated by

$$\frac{value_{after} - value_{before}}{value_{before}} \times 100\%$$

First let's see the difference between the original network (network 1) and the network after removing nodes with highest betweenness (network 2).

Percent Change	From Network 1 to Network 2
Average Betweenness	99.07%
Average Length of Shortest Path	78.00%
Average Closeness	-30.86%

After removing nodes, we can see that the average betweenness doubles, and the average length of shortest path also increases by 78%, while the average closeness decreases by 30%. Since the nodes with the highest betweenness are removed, many shortest paths in the network have to be rearranged, and lengthened, resulting in the increase in average shortest path length and decrease in average closeness. For betweenness, the removal of highest betweenness nodes let the shortest paths pass other nodes more frequently, and thus the other nodes take more responsibility in the network than before. As a result, the average betweenness in the whole network will face an increase.

Let us see a small example. We can easily tell that node 3 has the highest betweenness. And the shortest paths from node S to node1 and node 2 is shown by two orange dashed arrow lines respectively.

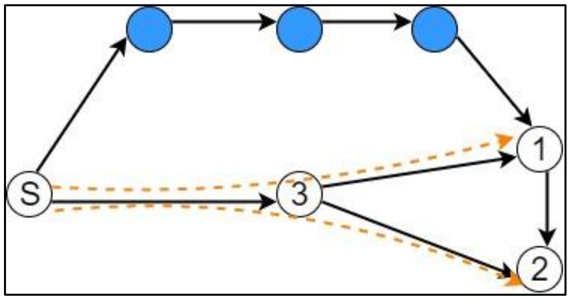

Figure 11 Example

Now we remove node 3, then the shortest paths from node S to node 1 and node 2 have to pass the three blue nodes, and as a result, the betweenness of those blue nodes will all increase, so do the average shortest path. Also, the average closeness will decrease.

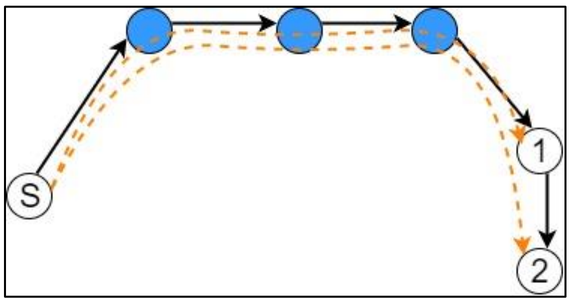

Figure 12 Example

Then let's see the difference between the network after removing nodes with highest betweenness (network 2), and the network after adding edges between nodes with lowest betweenness (network 3).

Percent Change	From Network 2 to Network 3
Average Betweenness	-28.10%
Average Length of Shortest Path	-29.85%
Average Closeness	24.80%

The transition from network 2 to network 3 is a reverse of the transition from network 1 to network 2, but with smaller change in the three criteria. This makes sense since more edges in the network result in a higher connectivity. However, we can't restore the network to the original state, because we may lose many edges and connections by removing only one node with highest betweenness, but in this simple case, I just add two more edges for each node with lowest betweenness.

2. Chicago Sketch

2.1 Original Network (Network 1)

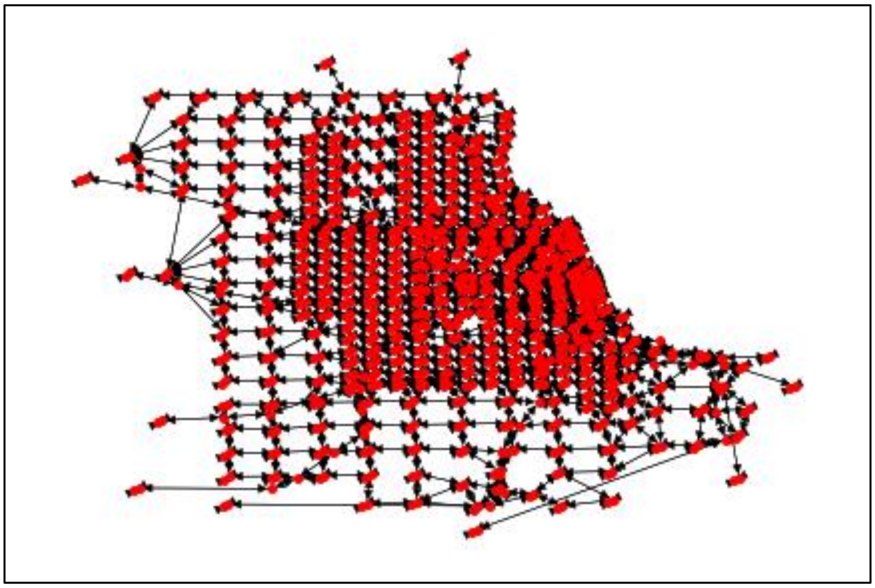

Figure 13 Original Network (Chicago)

	Original network (Network 1)
Average Betweenness	0.0125
Average Length of Shortest Path	100.1419
Average Closeness	0.0800

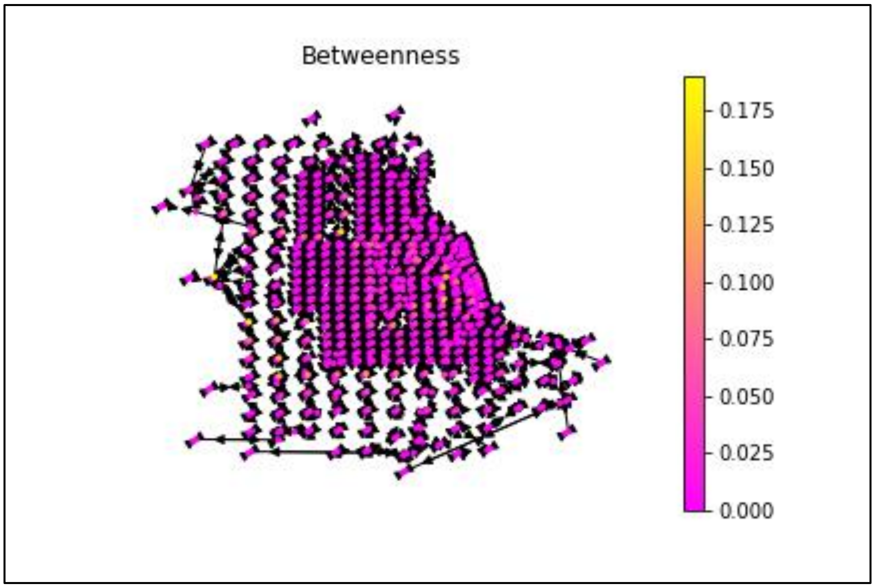

Figure 14 Betweenness of Network 1 (Chicago)

2.2 Remove 5% nodes with highest betweenness (Network 2)

This network originally has 933 nodes, so I decided to remove 5% nodes with the highest betweenness.

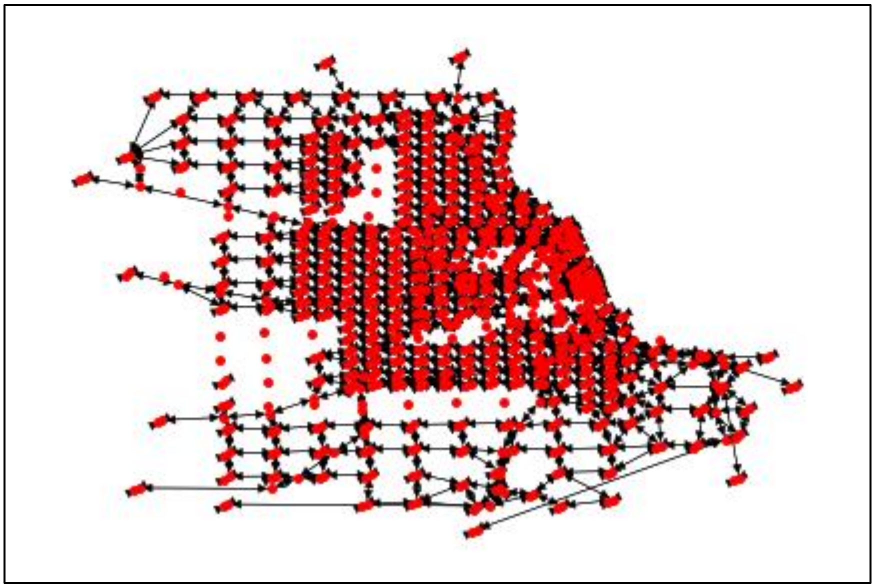

Figure 15 Network 2 (Chicago)

After I removed those nodes, I found that the network is not strongly connected any more. there are two kinds of isolated nodes: the first ones are totally isolated from the major cluster (the largest strongly connected component), and it can't reach or be reached by any other nodes; the second kind of isolated nodes can connect the major cluster just in one direction, either from the cluster to the nodes, or from the nodes to the cluster.

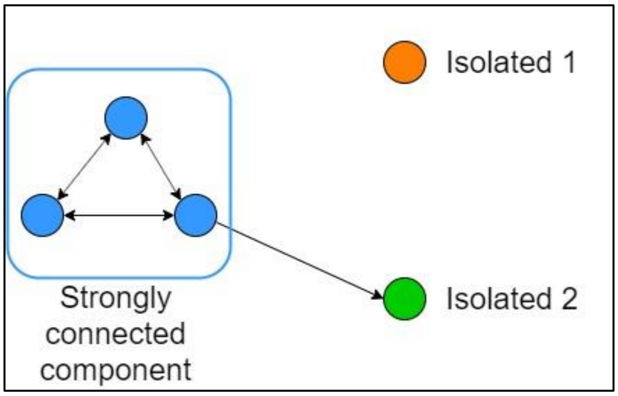

Figure 16 Illustration of Strong Connectivity

	After Removing Nodes with Highest Betweenness (Network 2)
Average Betweenness	0.0147
Average Length of Shortest Path	108.0670
Average Closeness	0.0598

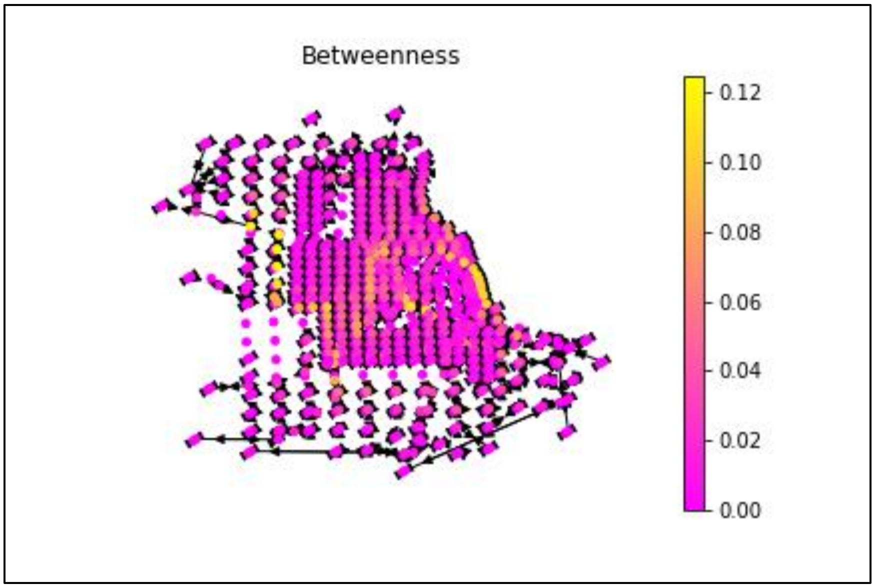

Figure 17 Betweenness of Network 2 (Chicago)

2.3 Connect nodes with zero betweenness (Network 3)

The removal of nodes produced 394 nodes with zero betweenness. Therefore, in this case, I want to make more connections among those nodes, I have to come up with more efficient connection method, since if I just simply connect those nodes one with each other in both directions, it could increase the computational complexity drastically, and building roads everywhere is also unrealistic in practice. As a result, I decide to add edges in the following way. For example, node 1, 2, 3, 4, and 5 are five nodes with zero betweenness. Node 3 is the closest node with node 1, and thus I add the bidirected connections between node 3 and node1. Node 4 is closest to node 2 and node 5, and the following 4 connections are added: 4 to 2, 2 to 4, 4 to 5, and 5 to 4.

A pseudocode is shown as below:

```
count, a, b = 0, 0, 0
for node 1 in Set_of_zero_betweenness:
    count += 1
    shortest_path = 100000
    for node 2 in Set_of_zero_betweenness[count:]:
        if node 1 != node 2:
            if distance(node 1,node 2) <= shortest_path:
                shortest_path = distance(node 1,node 2)
                a = node 1
                b = node 2
    G.add_edge(a,b,length = shortest_path)
```

G.add_edge(b,a,length = shortest_path)

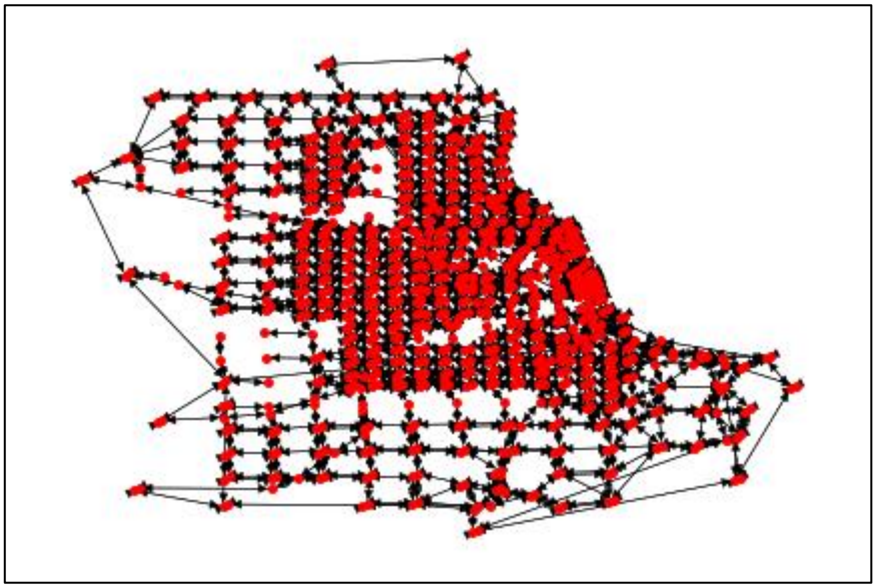

Figure 18 Network 3 (Chicago)

	After Connecting Nodes with lowest Betweenness (Network 3)
Average Betweenness	0.0115
Average Length of Shortest Path	105.4544
Average Closeness	0.0912

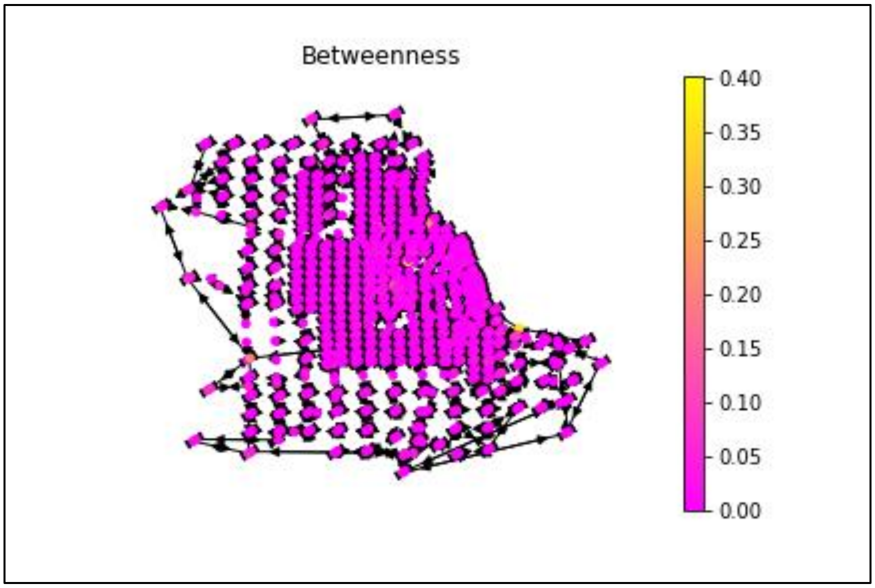

Figure 19 Betweenness of Network 3 (Chicago)

2.4 Comparison

First let's see the difference between the original network (network 1) and the network after removing nodes with highest betweenness (network 2).

Percent Change	From Network 1 to Network 2
Average Betweenness	16.98%
Average Length of Shortest Path	7.91%
Average Closeness	-25.28%

Then let's see the difference between the network after removing nodes with highest betweenness (network 2), and the network after adding edges between nodes with lowest betweenness (network 3).

Percent Change	From Network 2 to Network 3
Average Betweenness	-21.93%
Average Length of Shortest Path	-2.42%
Average Closeness	52.52%

From the tables we can see that the change in average shortest path is very small compared to the other two criteria. Also, the changes in all three criteria are relatively small compared to Sioux Falls. I think it's because the nodes and edges in Chicago Sketch are far more than those in Sioux Falls, and small portion of nodes will have less effect on the whole network.

Future Improvement

1. Identify nodes importance in a more comprehensive way, not just by betweenness.

2. Investigate into more complex network.

3. Produce better ways to decrease the effect of important nodes on the whole network. (i.e. Add connections among less important nodes)