Personal Facebook Graph Visualization using NodeXL
CMSC 734 Homework 2
Hao Li
haoli@cs.umd.edu

Introduction
Facebook allows us to discover our 1.5-degree network, from which I can learn who my friends are and which of my friends are friends with each other. In this project, I analyzed my Facebook network with NodeXL, which helps me to know more about my friends.

Data
The data is drawn from the NameGenWeb (http://apps.facebook.com/namegenweb), a Facebook app that can download your Facebook connections in a GraphML file. I download my network including following attributes: Facebook User ID, Full Name, Gender, Profile Picture URL, Locale and Mutual Friend Count. The network has 224 friends and 1982 undirected connections. The density of the overall graph is $2 \frac{|E|}{|V|(|V|-1)} = 0.08$.

Figure 1 shows the network using the Harel-Koren-Fast Multiscale layout. Givan-Newman method is used to identify clusters. The vertex color is based on different clusters. Size is represented by eigenvector centrality value. Thus someone who knows many people will show up bigger.
Headline 1: There are mainly five groups in my network. Most of my Facebook friends are known according to the research lab.

As shown in Figure 2, there are mainly five groups in my network:

1) Friends known at an industry research lab where I interned twice. The cluster consists of a lot of interns and researchers. I spend a year here working with two research groups. (black)
2) Friends at University of Maryland and friends studying at other U.S. universities (blue).
3) Undergraduate schoolmates in China (dark green).
4) Computer vision research group at the graduate school in China (light green).
5) Bioinformatics research group at the graduate school in China (red).

I am pretty glad that NodeXL identified meaningful groups in my networks, which reminds me the time and place where I have studied or worked before. It is a little supervised to see that most of my Facebook friends are known through the internship at the research lab. The reasons might be following:

1) The company is an international software company and people inside the company can easily visit Facebook since there is VPN in the company, while it is not so easy for my friends in the graduate school and undergraduate school to visit Facebook since it is blocked in China.
2) The research lab has been established for 15 years and a lot of students have interned there. The research lab is quite successful and has close relationship with both academy and industry.
The corresponding graph metrics for each group is shown in Table 1. We can see that the small groups (G4 and G5) has large graph density since person in the same research group know each other well. While my friends known from the industry research lab has smaller density comparing to small research groups. The reason behind could be that the research lab has a lot of groups and it is quite difficulty for people in a large company to be fully connected.

**Table 1 Graph metrics for each sub group.**

<table>
<thead>
<tr>
<th>Group</th>
<th>Vertices</th>
<th>Total Edges</th>
<th>Maximum Geodesic Distance</th>
<th>Average Geodesic Distance</th>
<th>Graph Density</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>111</td>
<td>1360</td>
<td>4</td>
<td>1.850</td>
<td>0.223</td>
</tr>
<tr>
<td>G2</td>
<td>34</td>
<td>212</td>
<td>4</td>
<td>1.647</td>
<td>0.378</td>
</tr>
<tr>
<td>G3</td>
<td>32</td>
<td>119</td>
<td>5</td>
<td>2.113</td>
<td>0.240</td>
</tr>
<tr>
<td>G4</td>
<td>16</td>
<td>78</td>
<td>3</td>
<td>1.273</td>
<td>0.650</td>
</tr>
<tr>
<td>G5</td>
<td>12</td>
<td>31</td>
<td>3</td>
<td>1.458</td>
<td>0.470</td>
</tr>
</tbody>
</table>

*Figure 3 Radial layout for each sub group.*
Headline 2: Identify the most influential friend in the research lab.

It is interesting to know who is the most influential or the “star” in a community. I looked into the group of friends in the industry research lab. I removed nodes and edges outside the group and re-calculated their graph metrics. In Figure 4, the size of each vertex is based on its Eigenvector Centrality score, the opacity is based on its PageRank value, whereas its position is determined by its Degree and Between Centrality score. From this figure, we can see the influence of each person. The person on the right upper corner is a student who spends almost 5 years at the research lab, he has a lot of connections (70) there and almost 2/3 of friends I know there knows him. He ends up become the employee of that company. His connection graph can be seen in Figure 5.
People are always in groups. I clustered the connections in the research lab by clique motifs to see close connection. In Figure 6, the clique of connections is represented by star. There are 1 cliques with 10 nodes, 1 cliques with 5 nodes and 12 cliques with 4 nodes. It is interesting that the largest cliques in composed by the persons who have the highest Eigenvector Centrality scores, which indicates that influential people are most friends with each other. The other cliques are also quite meaningful and reasonable. For example, people in one clique are mostly interns at the same research group or sit in a square (a square usually have 4 seats).

Critique of NodeXL

- **Pros**
  1. It is easy to get start based on the popularity of Windows and Excel.
  2. The import function make it easy to access the account of Twitter and Youtube. It would be better if we can access the Facebook account directly instead of importing a GraphML file.

- **Cons**
  1. I often drag the graph window outside the main window of Excel to view it on a larger screen. The graph window lacks of Maximize/Minimize button, so if you want to resize the window, you have to drag it by hand. Also it is a little hard to put it back. Later I figured out that double clicking the top of the graph window will automatically set it back. However, it is not quite intuitive since double click usually means maximizing the window.
  2. When a sub graph (cluster) needs to be analyzed, I would expect to select those data and visualize it separately. However, it seems to me that I have to delete the data of other clusters manually and making it hard to get back to the original data later unless create another copy of the data.
  3. For a large graph (~5000 nodes and ~150K edges), the processing time would be much slower and sometimes crash happens.
  4. When a graph with axis is saved as image to the disk, the axis is not included in the image.