

Analysis on Friend Relationships among Popular and High Review Users on Yelp.com

I. Formation of subgraphs

There are a large amount of data in yelp_academic_dataset_user.json. To make our analysis clearer, we form two subgraphs, one from the popular users, and the other from the high review users. The graphs are as follows:

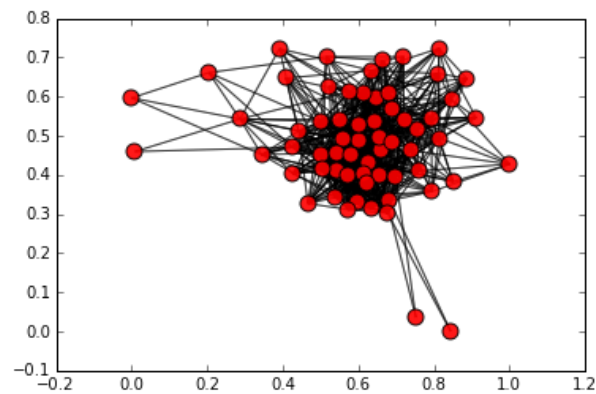


Figure 1: Subgraph for popular users' friends

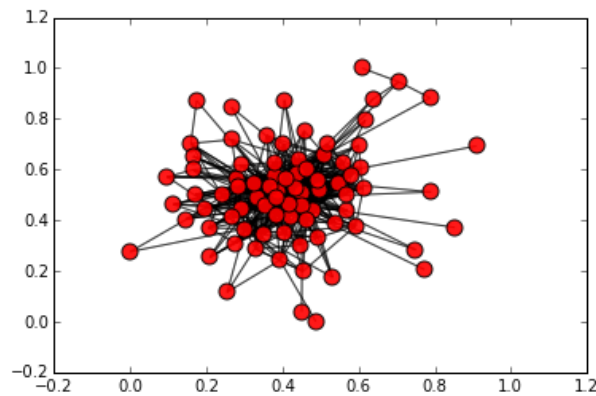


Figure 2: Subgraph for high review users' friends

Figure 1 is more concentrated than Figure 2. Also, there are several nodes in Figure 2 that has only one edge, which means they only has one friend in the 'high review users circle'. That phenomenon is easy to interpret, since high review users tend to be more isolated than popular users, i.e. popular users are more connected to each other.

II. Analyzing Graphs

The maximum degrees among popular users is 40, while the number is 50 for high review users. That means for all the popular users, the user who has the most friends inside the 'popular users circle' has 40 friends. For all the high review users, the user who has the most inside-circle friends has 50 in total.

The average clustering coefficient for popular users is 0.59, while it is 0.51 for high review users. So high review users is less clustered than popular users.

The maximum and minimum closeness centrality values among popular users are 0.74 and 0.38 respectively, and the values are 0.71 and 0.27 in the other case. Thus, the most central node among popular users is more central than the most central one from high review users.

Finally, we find out the most central nodes among those two kinds of users. The size of most central popular users set is 5. That means 7.81% of popular users are most central ones. While in high review users, that number climbs to 28.05%. The difference indicates that the friend relationships are more averaged for high review users.

III. Python Codes

Please check this GitHub page for the codes:

https://github.com/g001234/Python_assignment/blob/master/Data_analysis_assignment.ipynb