Q6 Report: Community Detection

Author: YANG Rongfeng

Date: 2020-05-31

I use networkx to represent the follower-network and use Girvan-Newman (GN) algorithm to detect the communities. I mainly use the follower social network to do this task. The graph below is the visualization of the Twitter social network.

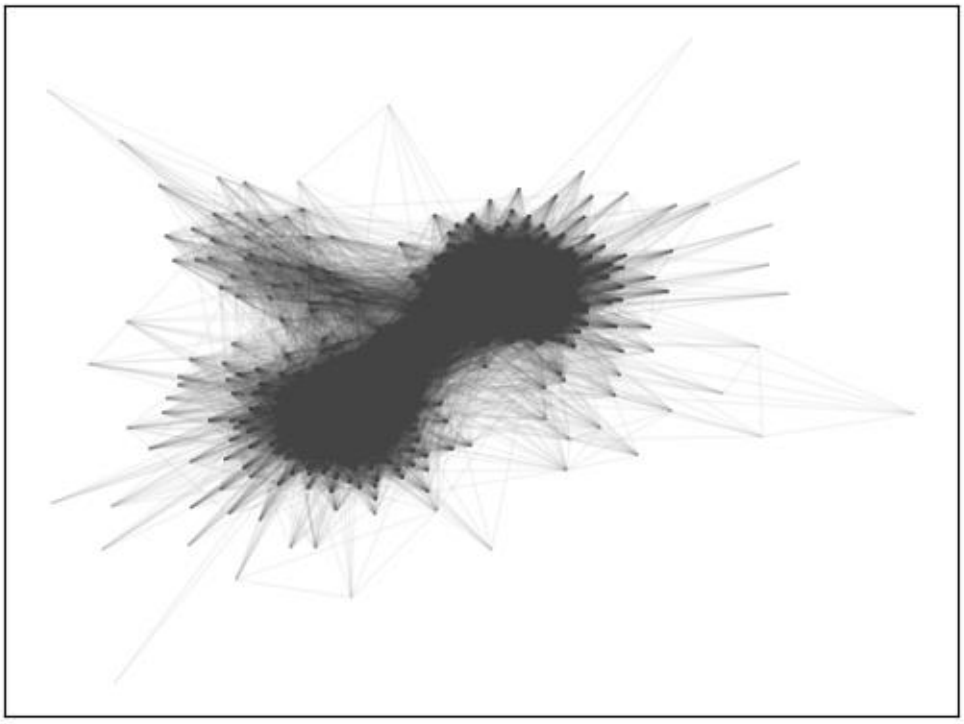


Figure. The visualization the follower social network

Next, I will introduce the details of the GN algorithm.

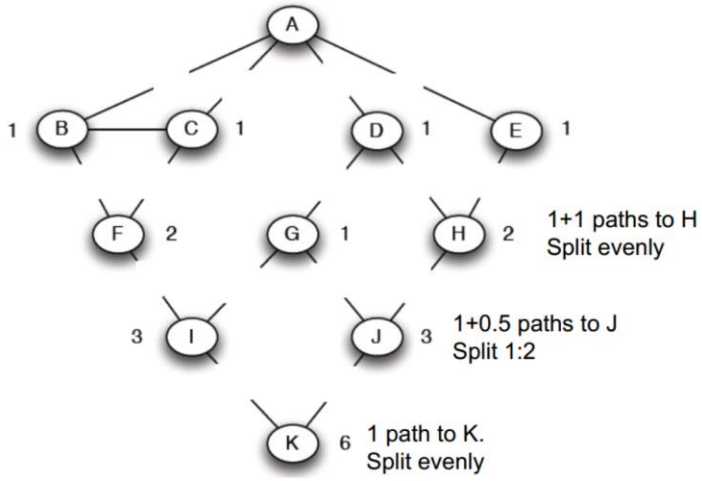
1. The GN algorithm Description

The Girvan–Newman algorithm detects communities by progressively removing edges from the original network. The connected components of the remaining network are the communities. Instead of trying to construct a measure that tells us which edges are the most central to communities, the Girvan–Newman algorithm focuses on edges that are most likely "between" communities.

The algorithm's steps for community detection are summarized below

(1) The betweenness of all existing edges in the network is calculated first.
(2) The edge(s) with the highest betweenness are removed.
(3) The betweenness of all edges affected by the removal is recalculated.
(4) Steps 2 and 3 are repeated until no edges remain.

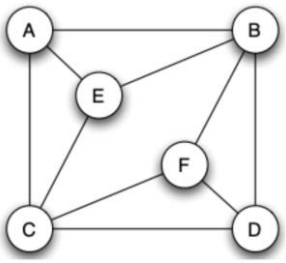
More detailed, the steps of computing betweenness can be described as



1. Add edge flows
   Node flow = 1 + Sum(child edges)
   Split the flow up based on the parent value
2. Repeat eh BFS procedure for each starting node
3. Sum up the flow values to get the betweenness
4. Divide everything by two

Algorithm. The steps of computing betweenness

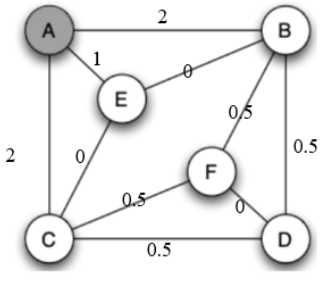
We can see an easy example of implementing the GN algorithm.

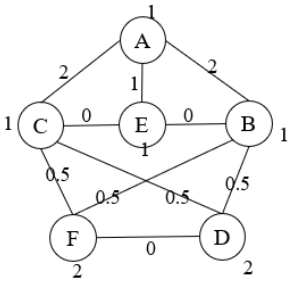


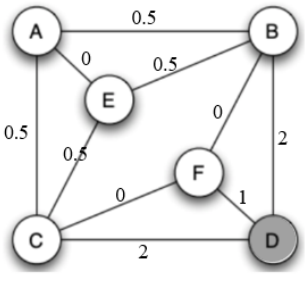
Next, we'll see how to implement the GN algoritm on the simple graph above.

Step 1

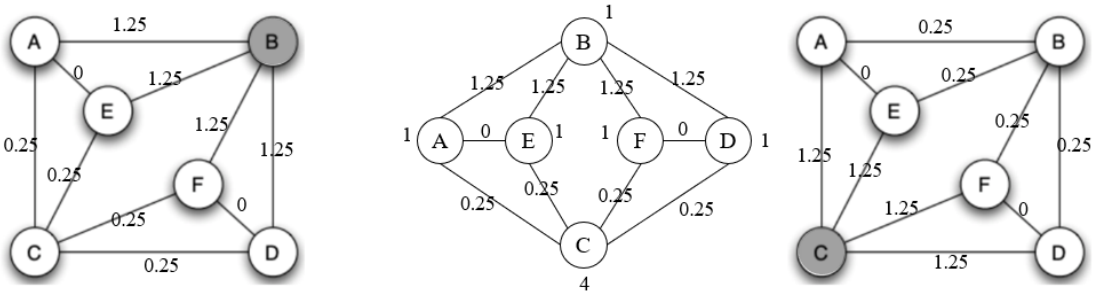
BFS starting from A and use symmetry to get the flows beginning with D



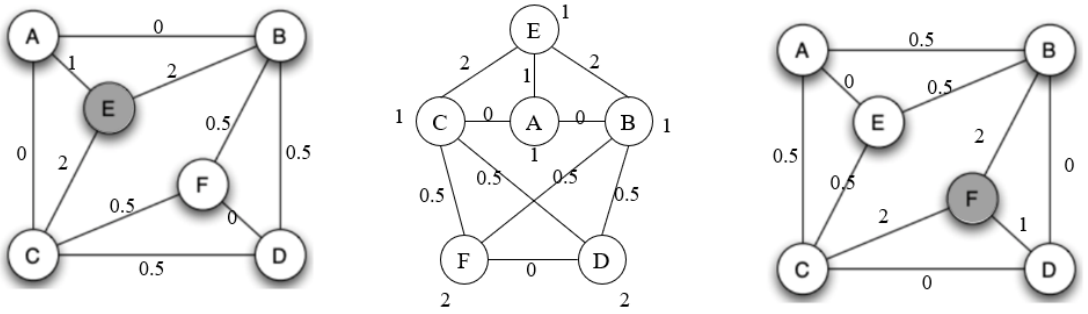




BFS starting from B and C

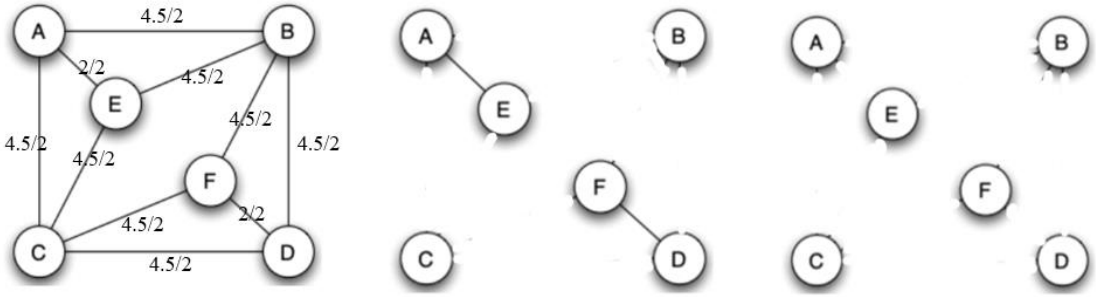


BFS starting from E and F



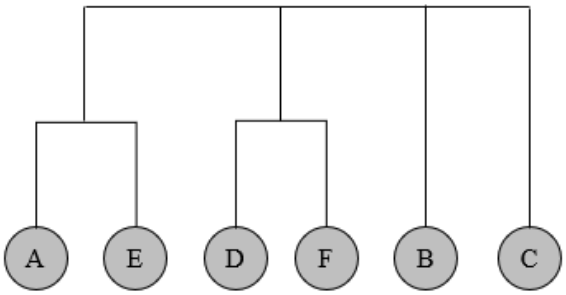
Step 2

Add up the flow values and divided everything by 2. Then we remove all the maximum edges and we only have AE and DF left. It's obvious that we remove these two edges in the last step.



Now all the edges have been removed.

We analyze each step of GN algorithm, we can obtain the dendrogram



2. The Experiment Results

The whole community detiection results are stored in `task1_output.mtx`.

I use GN algorithm to detect five communities. The visualization result is shown below. The internal connections in the community are set to black so we cannot see them the black background. However, the external connection between communities are set to silver.

Four farthest four users form four communities and the middle bundles of users form the fifth community.

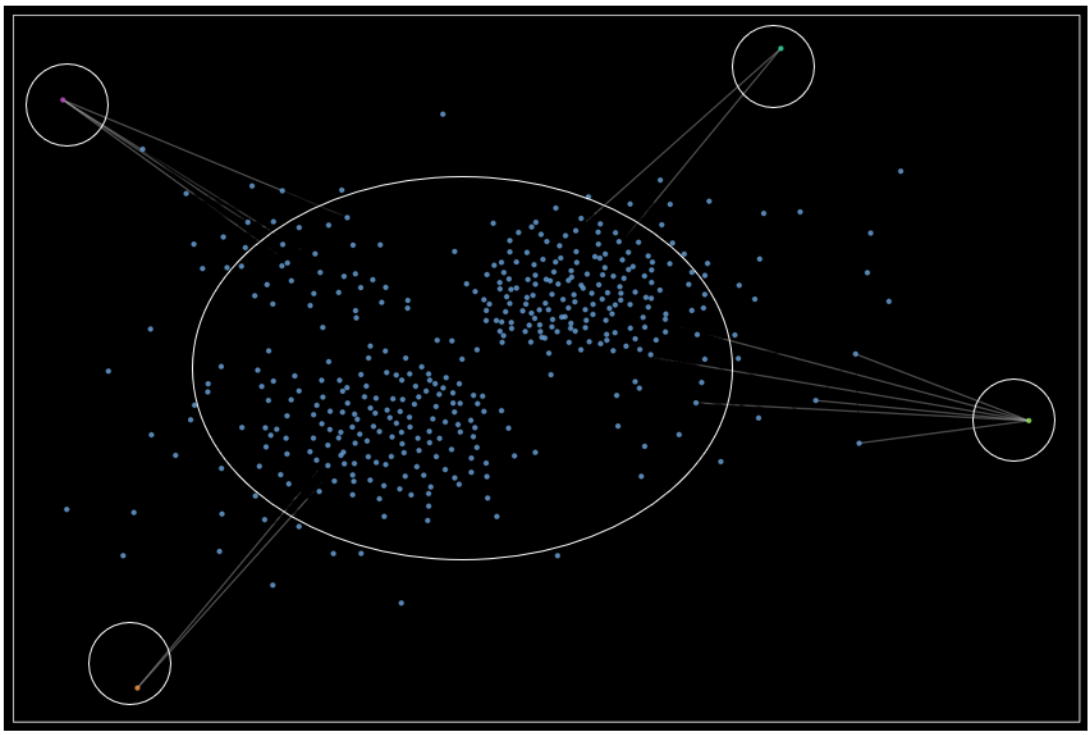


Figure. The Visualization of the community detection

Reference

[1] Node2Vec: https://cs.stanford.edu/~jure/pubs/node2vec-kdd16.pdf

[2] COMMUNITY DETECTION USING NETWORKX: https://orbifold.net/default/community-detection-using-networkx/

[3] https://www.researchgate.net/publication/337469159_Community_detection_in_complex_networks_using_Node2vec_with_spectral_clustering

[4] Generating A Twitter Ego-Network & Detecting Communities: https://towardsdatascience.com/generating-twitter-ego-networks-detecting-ego-communities-93897883d255