EE 232E Project 2
Social Network Mining

Hengjie Yang, Sheng Chang, Wandi Cui, and Tianyi Liu

May 7, 2018

1 Facebook network

1.1 Structural properties of the facebook network

The facebook network is plotted in Fig. 1.

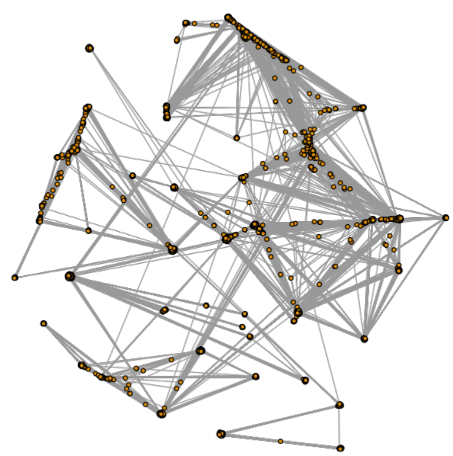

Figure 1: Facebook network

Question 1: Is the facebook network connected? If not, find the giant connected component (GCC) of the network and report the size of the GCC.

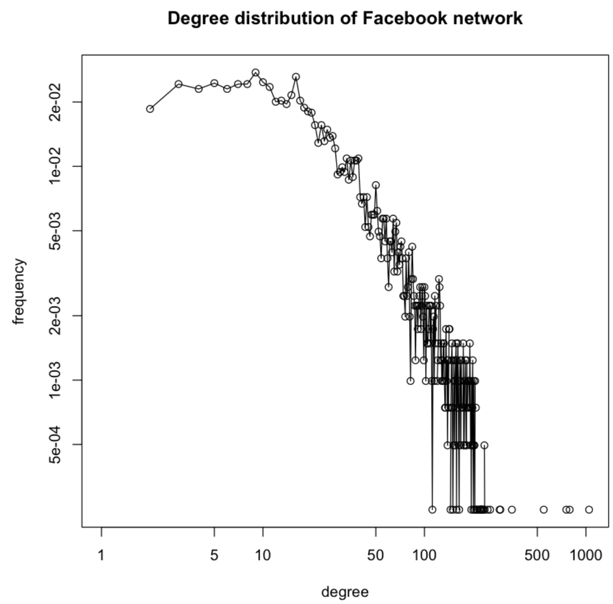

Figure 2: Degree distribution of the Facebook network

The facebook network is connected.

Question 2: Find the diameter of the network. If the network is not connected, then find the diameter of the GCC.

The diameter of the Facebook network is 8.

Question 3: Plot the degree distribution of the facebook network and report the average degree.

The degree distribution of the Facebook network is shown in Fig. 2. The average degree is 43.69101.

Question 4: Plot the degree distribution of question 3 in a log-log scale. Try to fit a line to the plot and estimate the slope of the line.

The degree distribution of Facebook network in log-log scale is shown in Fig. 3. In order to find a line which fits the data, we consider the data starting from 20-th and ends 6 before the end, thus, with linear regression analysis, the line we find is as follows.

$$y = 1.032 - 1.607x \tag{1}$$

where y represents the $\log(frequency)$ and x represents the $\log(degree)$. The estimated slope is -1.607.

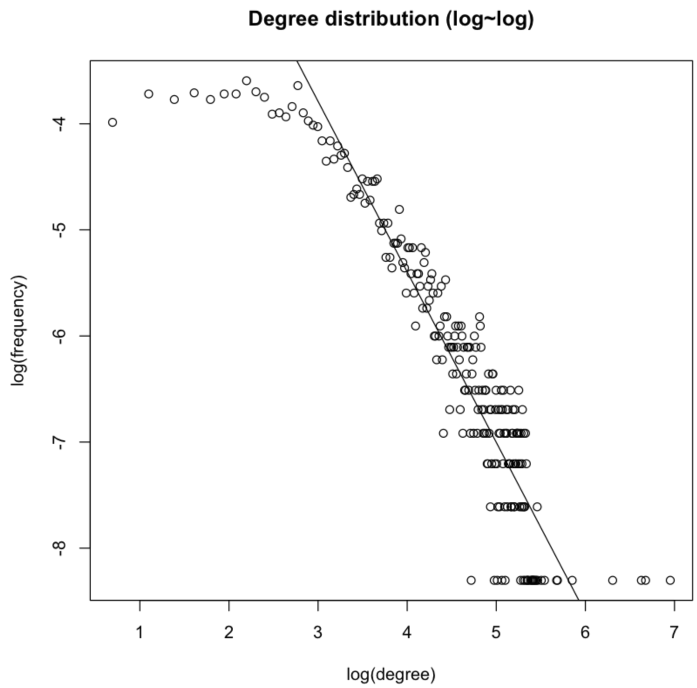

Figure 3: Degree distribution of the Facebook network in log-log scale

1.2 Personalized network

Question 5: Create a personalized network of the user whose ID is 1. How many nodes and edges does this personalized network have?

The personalized network is shown in Fig. 4. The number of nodes is 348, and the number of edges is 2866.

Question 6: What is the diameter of the personalized network? Please state a trivial upper and lower bound for the diameter of the personalized network.

The diameter of the personalized netowrk is 2. A trivial upper bound of the diameter of the personalized network is 2 and the lower bound of the personalized network is 1.

Question 7: In the context of the personalized network, what is the meaning of the diameter of the personalized network to be equal to the upper bound you derived in question 6. What is the meaning of the diameter of the personalized network to be equal to the lower bound you derived in question 6?

The meaning is that: give the core node of the personalized network, when the number of the neighbor nodes is 1, clearly the diameter of this network is 1; If the number of the

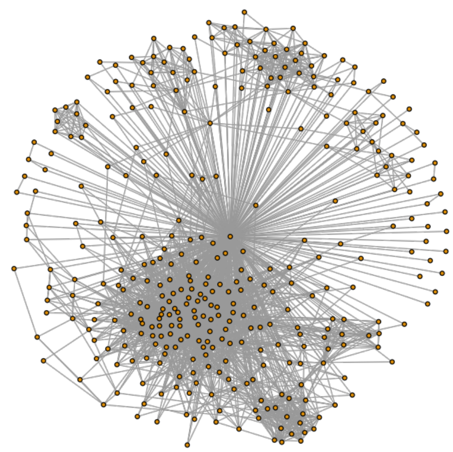

Figure 4: Personalized network of user with ID 1

neighbor nodes is greater than 1, since all neighbor nodes are connected to the core node, thus the diameter of this network is 2.

1.3 Core node's personalized network

There are 40 core nodes in the Facebook network, which is the nodes that have more than 200 neighbors i.e. the degree of the nodes is greater than 200. And the average degree of the core nodes is 279.

1.3.1 Community structure of core node's personalized network

We aim to find the community structure and compute the modularity scores using Fast-Greedy, Edge-Betweenness, and Infomap community detection algorithms for each of some core nodes' personalized network.

For Node ID 1, the community figures based on different algorithms are shown in Fig 5, Fig 6 and Fig 7.

For Node ID 108, the community figures based on different algorithms are shown in Fig 8, Fig 9 and Fig 10.

For Node ID 349, the community figures based on different algorithms are shown in Fig 11, Fig 12 and Fig 13.

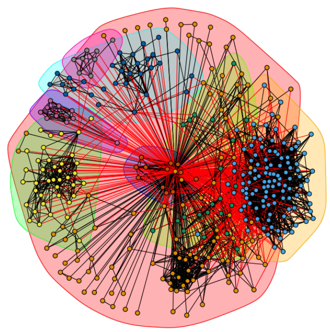

Figure 5: community structure using Fast-Greedy algorithms

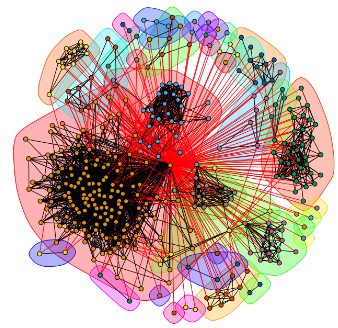

Figure 6: community structure using Edge-Betweenness algorithms

For Node ID 484, the community figures based on different algorithms are shown in Fig 14, Fig 15 and Fig 16.

For Node ID 1087, the community figures based on different algorithms are shown in Fig 17, Fig 18 and Fig 19.

What's more, all the modularity scores from above core nodes' personalized network computed by different algorithms is shown in Table 1.

Table 1: The modularity scores for core nodes' personalized network

Node ID	**Fast-Greedy**	**Edge-Betweenness**	**Infomap**
1	0.41310	0.35330	0.38912
108	0.43592	-	0.50842
349	0.25171	0.13352	0.20375
484	0.50700	0.48909	0.51527
1087	0.14553	0.02762	0.02690

1.3.2 Community structure with the core node removed

In this part, we aim to explore the effect on the community structure of a core node's personalized network when the core node itself is removed from the personalized network.

For Node ID 1, the community figures based on different algorithms are shown in Fig 20, Fig 21 and Fig 22.

For Node ID 108, the community figures based on different algorithms are shown in Fig 23, Fig 24 and Fig 25.

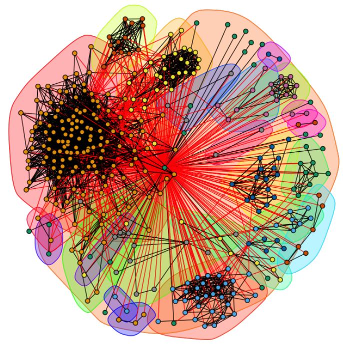

Figure 7: community structure using Infomap algorithms

For Node ID 349, the community figures based on different algorithms are shown in Fig 26, Fig 27 and Fig 28.

For Node ID 484, the community figures based on different algorithms are shown in Fig 29, Fig 30 and Fig 31.

For Node ID 1087, the community figures based on different algorithms are shown in Fig 32, Fig 33 and Fig 34.

What's more, all the modularity scores from above core nodes' personalized network computed by different algorithms is shown in Table 2.

Table 2: The modularity scores for core nodes' personalized network

Node ID	**Fast-Greedy**	**Edge-Betweenness**	**Infomap**
1	0.41310	0.35330	0.38912
108	0.45812	0.52132	0.52068
349	0.24569	0.15056	0.24657
484	0.53421	0.51544	0.54344
1087	0.14819	0.03249	0.02737

1.3.3 Characteristic of nodes in the personalized network

We aim to explore characteristics of nodes in the personalized network using two measures. These are two measures. One is embeddedness of a node that is defined as the number of mutual friends a node shares with the core node. Another that is dispersion of a node is defined as the sum of distances between every pair of the mutual friends the node shares

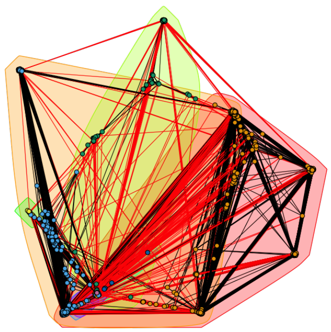

Figure 8: community structure using Fast-Greedy algorithms

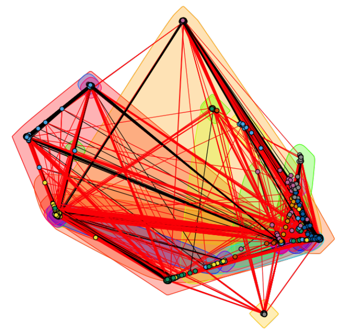

Figure 9: community structure using Edge-Betweenness algorithms

with the core node. The distances should be calculated in a modified graph where the node (whose dispersion is being computed) and the core node are removed.

The expression relating the Embeddedness of a node to it's degree is given as follows.

$$Embedd(i) \leq Degree(i) - 1 \tag{2}$$

For code code 1's personalized network, the distribution of embeddedness and dispersion is shown in Fig 35 and Fig 36.

For code code 108's personalized network, the distribution of embeddedness and dispersion is shown in Fig 37 and Fig 38.

For code code 349's personalized network, the distribution of embeddedness and dispersion is shown in Fig 39 and Fig 40.

For code code 484's personalized network, the distribution of embeddedness and dispersion is shown in Fig 41 and Fig 42.

For code code 1087's personalized network, the distribution of embeddedness and dispersion is shown in Fig 43 and Fig 44.

For each of the core node's personalized network, apply Fast-Greedy algorithm to detect the community structure of the personalized network and use colors and highlight the node with maximum dispersion and the edges incident to this node to plot this community.

For code code 1's personalized network, the community structure is shown in Fig 45. For code code 108's personalized network, the community structure is shown in Fig 46. For code code 349's personalized network, the community structure is shown in Fig 47. For code code 484's personalized network, the community structure is shown in Fig 48. For code code 1087's personalized network, the community structure is shown in Fig 49.

Similarly, this time we highlight the node with maximum embeddedness and the node with maximum ratio of dispersion to embeddedness.

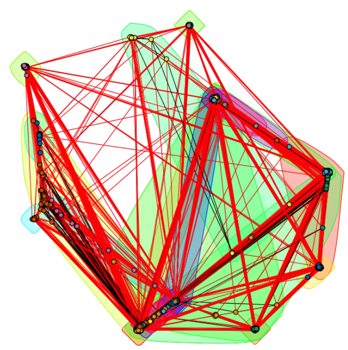

Figure 10: community structure using Infomap algorithms

For code code 1's personalized network, the community structure is shown in Fig 50. For code code 108's personalized network, the community structure is shown in Fig 51. For code code 349's personalized network, the community structure is shown in Fig 52. For code code 484's personalized network, the community structure is shown in Fig 53. For code code 1087's personalized network, the community structure is shown in Fig 54.

1.4 Friend recommendation in personalized networks

In this section, the network that we used is the personalized network of node with ID 415.

Question 16:

We find the list of all nodes with degree 24 within the personalized network:

Nr = [31 53 75 90 93 102 118 133 134 136 137]

The node ID is based on the personalized network. Thus, |Nr| is equal to 11.

Question 17:

For observing the difference of three recommendation algorithms, we ran our program many times and get an interesting result. The accuracy of these three algorithms we got are as follows: (order is: "Common Neighbors measure", "Jaccard measure", and "Adamic Adar measure")

0.8429869 0.8043000 0.8461796
0.8327922 0.7958622 0.8307468
0.8152561 0.7930083 0.8220310
0.8377378 0.8017198 0.8401728
0.8320815 0.7953243 0.8268939
0.8271593 0.8162239 0.8205897

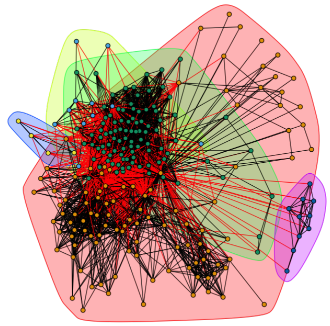

Figure 11: community structure using Fast-Greedy algorithms

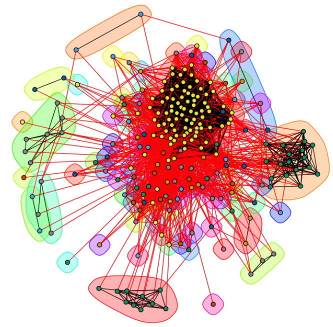

Figure 12: community structure using Edge-Betweenness algorithms

0.8460757 0.8111159 0.8497986

As we can see, both "Common Neighbors measure" algorithm and "Adamic Adar measure" can get best results according to our experiments, but the accuracy of these two algorithm is really similar each time (sometimes "Common Neighbors measure" performs better, sometimes "Adamic Adar measure" performs better). However, "Jaccard measure" get the worst accuracy among these three algorithms.

The reason of this might be:

(1) "Common Neighbors measure" tend to recommend high-degree nodes to objection nodes according to its definition, since high-degree nodes in network probably connect to most of other nodes within the network. This probabiliy may not seem so meaningful in real world since companies are more likely to discover potential relationships in network and make "surprising" recommendations to users. In other words, for example, if a guy is a famous people (a high-degree node in network), people probably connect with him before the recommender system recommends this famous guy to the user; so that in this way, recommend a person who is "popular" in the network may not be an impressive thing for a recommender system. So some ohter measures like "Jaccard measure" and "Adamic Adar measure" are designed to avoid this influence. However, there is no doubt that "Common Neighbors measure" will help us get good recommendations; thus it is not surprise that using "Common Neighbors measure" can get high accuracy in our problem;

(2) "Jaccard measure" is an improvement of "Common Neighbors measure", which divides the length of the union of two nodes to weaken the influence of high-degree nodes. However, during experiments I found that there are many high-degree nodes connected to our objection nodes, so this can perfectly explans why we get worse results by using "Jaccard measure", some high-degree nodes which are true neighbors before we deleted

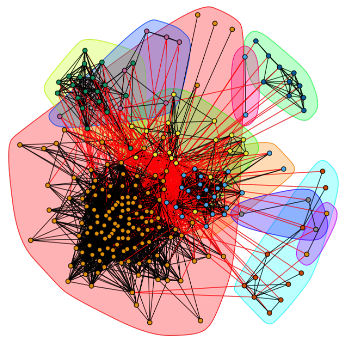

Figure 13: community structure using Infomap algorithms

edges may have low "Jaccard measure score" because of its high-degree property.

(3) "Adamic Adar measure" is also an impropment of "Common Neighbors measure", which intuitively shows that common friends who have very few friends can make more contribution to recommendation. It is easy to understand that if both people are connected to a person who may not like to connect with others, these two people must have high probability to be friends. This algorithm shows nearly equal performence with "Common Neighbors measure" in our problem. This is reasonable because when only considering the definition of the algorithm, "Adamic Adar measure score" won't be influenced by some properties of nodes in network like "Jaccard measure".

2 Google+ Network

In this section, we explored a directed graph: Google+ network.

Question 18:

There are 132 personal networks in total.

There are 57 personal networks for users who have more than 2 circles.

Question 19:

As we can see in the following plots, the in-degree distributions of the three personal networks are different, and the out-degree distributions of the three personal networks are relatively similar (slightly different).

(1) For Nodeid: 109327480479767108490

In-degree distribution is shown in Fig 55.

Out-degree distribution is shown in Fig 56.

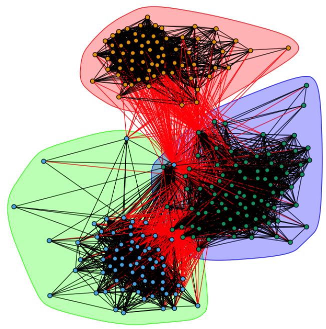

Figure 14: community structure using Fast-Greedy algorithms

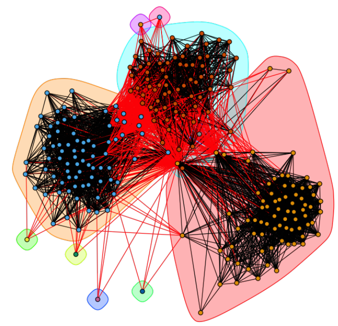

Figure 15: community structure using Edge-Betweenness algorithms

(2) For Nodeid: 115625564993990145546
In-degree distribution is shown in Fig 57.
Out-degree distribution is shown in Fig 58.
(3) For Nodeid: 101373961279443806744
In-degree distribution is shown in Fig 59.
Out-degree distribution is shown in Fig 60.
Question 20:
1. Modularity Scores

They have different modularity scores, which means that they have different strength of division of a network into modules.

(1) for Nodeid: 109327480479767108490 Modularity Score is: 0.252765.
(2) for Nodeid: 115625564993990145546 Modularity Score is: 0.319473.
(3) for Nodeid 101373961279443806744 Modularity Score is: 0.191090.
2. The plots of the communities using colors are as below:
(1) for Nodeid: 109327480479767108490, the plot is shown in Fig 61.
(2) for Nodeid 115625564993990145546, the plot is shown in Fig 62.
(3) for Nodeid 101373961279443806744, the plot is shown in Fig 63.
Question 21:

Homogeneity: A community structure is of higher homogeneity if its clusters more likely contain only data points which are members of a single class. To some extent, it describes the class purity of clusters. The expression of Homogeneity is $1 - H(C|K)/H(C)$. $H(C|K)$ here describes the extent that the class distribution within each cluster belong to a single class. $H(C)$ is the maximum reduction in entropy the clustering information could provide, which is used for normalization purpose since homogeneity also depends on the size of the

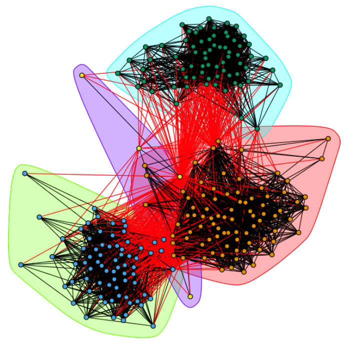

Figure 16: community structure using Infomap algorithms

dataset and the distribution of class sizes. For instance, assigning each data point to a distinct cluster guarantees perfect homogeneity. In that case, each cluster trivially contains only members of a single class. In the perfectly homogeneous case (h = 0), $H(C|K)$ is zero because each cluster contains only members of a single class.

Completeness: A community structure is of higher completeness if all the data points that are members of a given class are more likely assigned to the same cluster. Completeness roughly run in the opposition to homogeneity. The expression of Completeness is $1 - H(K|C)/H(K)$. $H(K|C)$ here describes the extent that the distribution of cluster assignments within each class will be completely adjusted to just one single cluster. For instance, in the perfectly complete case (c = 0), $H(K|C)$ is zero. In this case, the class distribution within each cluster is adjusted to a single class.

Question 22:

(1) for Nodeid 109327480479767108490

$H(C)$: 1.050779

$H(K)$: 1.005208

$H(C|K)$: 0.155636

$H(K|C)$: 0.673616

Homogeneity: 0.851885

Completeness: 0.329874

(2) for Nodeid 115625564993990145546

$H(C)$: 8.465147

$H(K)$: 1.081191

$H(C|K)$: 4.639829

$H(K|C)$: 4.783148

Homogeneity: 0.451890

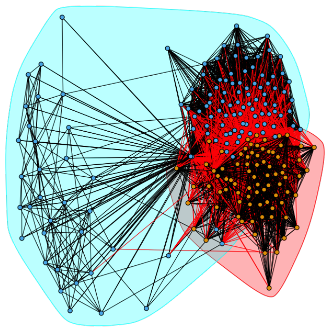

Figure 17: community structure using Fast-Greedy algorithms

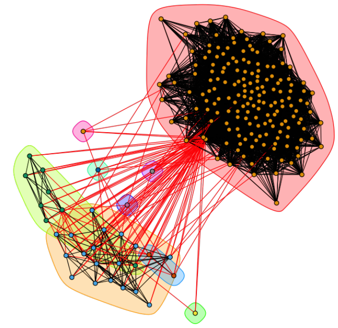

Figure 18: community structure using Edge-Betweenness algorithms

Completeness: -3.423962
(3) for Nodeid 101373961279443806744
$H(C)$: 0.384320
$H(K)$: 0.493331
$H(C|K)$: 0.382834
$H(K|C)$: 1.235417
Homogeneity: 0.003867
Completeness: -1.504238

As we can see from the results above, these three personal networks' homogeneity score is higher than completeness score.

Specifically, in the first personal network for userid=109327480479767108490, it both has highest homogeneity score and completeness score. There are three circles and their size $a[i]$ are 419, 346, 330 respectively. And there are four communities extracted by walk trap community detection algorithm, and their size $b[i]$ are 397, 288, 76, 13 respectively. The total number of unique nodes in three circles is $N = 764$ and the sum number of three circles is 1095, which means there is not much overlapping between circles. This can also be shown in the corresponding plot, the class distribution within each cluster is likely to belong to a single class, and the class distribution within each cluster is adjusted to a single class.

In the second personal network for userid = 115625564993990145546, it both has relatively low homogeneity score and completeness score compared to the first personal network. There are 31 circles and their size $a[i]$ are 489, 485, 373, 362... respectively. And there are 10 communities extracted using walk trap community detection algorithm, and their size $b[i]$ are 350, 256, 233, 40, 37, 3, 2, 1, ... respectively. The total number of unique nodes

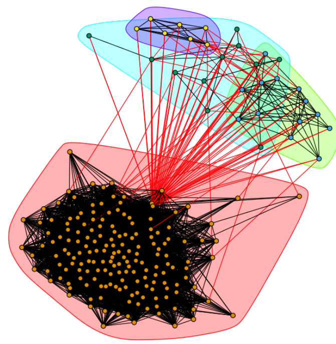

Figure 19: community structure using Infomap algorithms

in three circles is $N = 727$ and the sum number of all circles is 6474, which means there is relatively more overlapping between circles. So the probability that a community just contains nodes belonging to a single circle is low. In terms of the completeness, the score is negative, which could be explained by considering that the size of individual communities $b[j]$ is smaller than the size of most of the circles $a[i]$, which reflects the low probability that a circle only contains nodes that belong to a single community.

In the third personal network for userid = 101373961279443806744, it also has relatively low homogeneity score and completeness score compared to the others. There are 3 circles and their size $a[i]$ are 471, 445, 430 respectively. And there are 31 communities extracted by walk trap community detection algorithm, and their size $b[i]$ are 442, 66, 13, 0, ... respectively, and 27 communities only contain one single node so most of the C_{ji} which is the number of people belonging to community j and circle i are zero since most of the intersection of such communities and circles are empty set. As a result, there is low probability that a community just contains nodes that belongs to a single circle. In terms of the completeness, the completeness score is a negative score considering that the size of individual communities $b[j]$ is also smaller than the size of most of the circles $a[i]$, which reflects the low probability that a circle only contains nodes that belong to a single community.

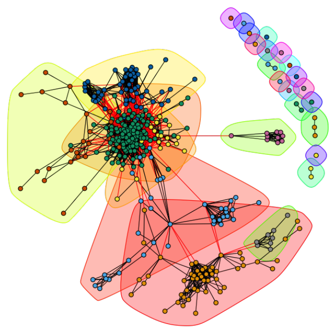

Figure 20: community structure using Fast-Greedy algorithms

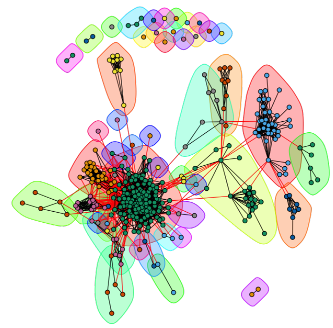

Figure 21: community structure using Edge-Betweenness algorithms

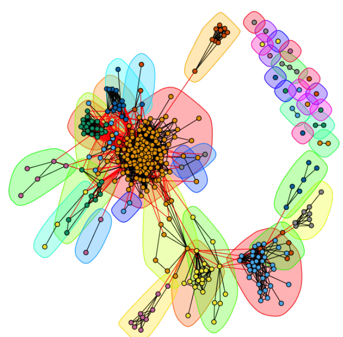

Figure 22: community structure using Infomap algorithms

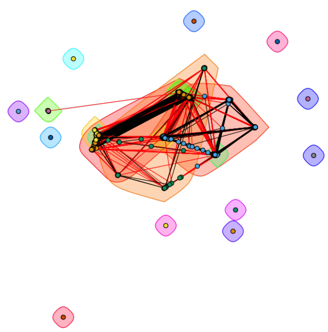

Figure 23: community structure using Fast-Greedy algorithms

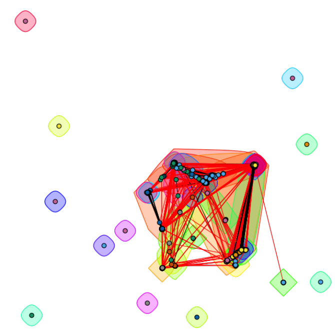

Figure 24: community structure using Edge-Betweenness algorithms

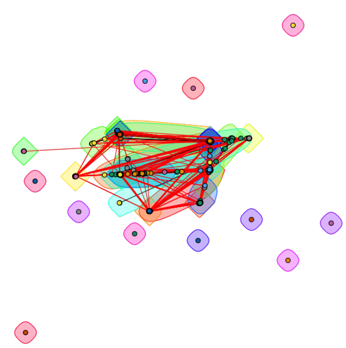

Figure 25: community structure using Infomap algorithms

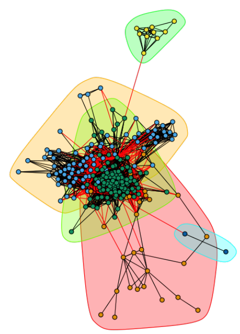

Figure 26: community structure using Fast-Greedy algorithms

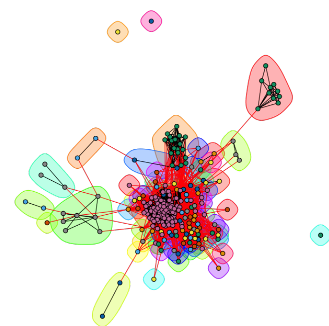

Figure 27: community structure using Edge-Betweenness algorithms

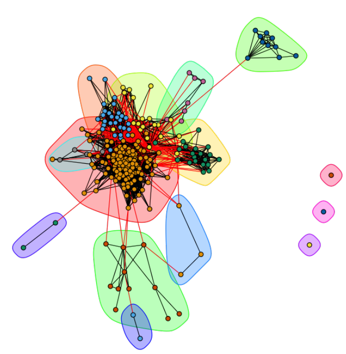

Figure 28: community structure using Infomap algorithms

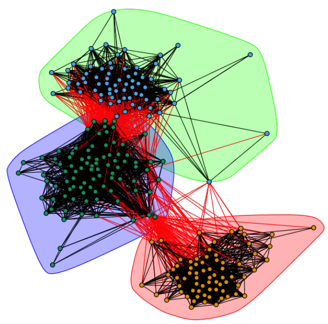

Figure 29: community structure using Fast-Greedy algorithms

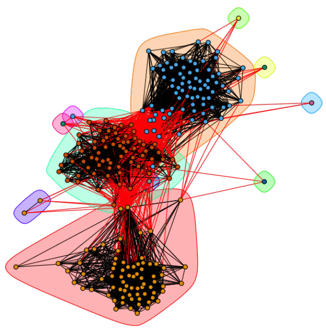

Figure 30: community structure using Edge-Betweenness algorithms

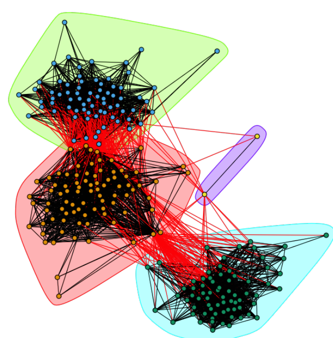

Figure 31: community structure using Infomap algorithms

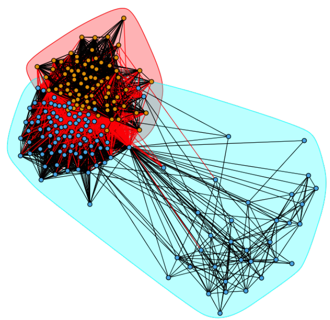

Figure 32: community structure using Fast-Greedy algorithms

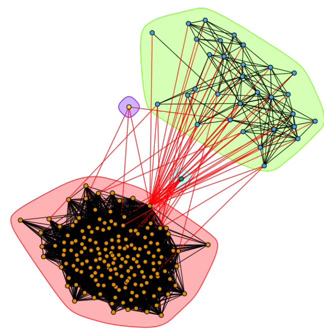

Figure 33: community structure using Edge-Betweenness algorithms

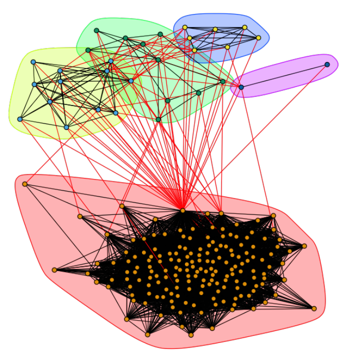

Figure 34: community structure using Infomap algorithms

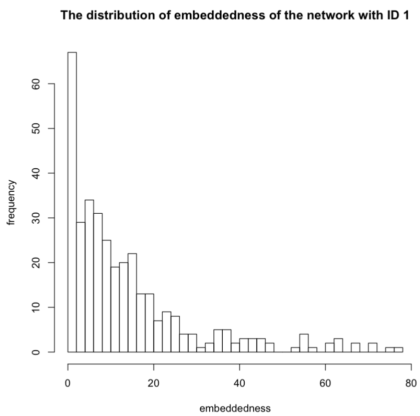

Figure 35: The distribution of embeddedness

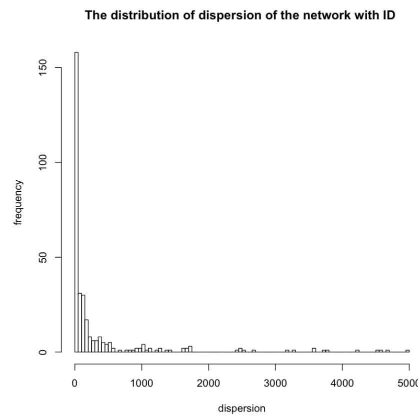

Figure 36: The distribution of dispersion

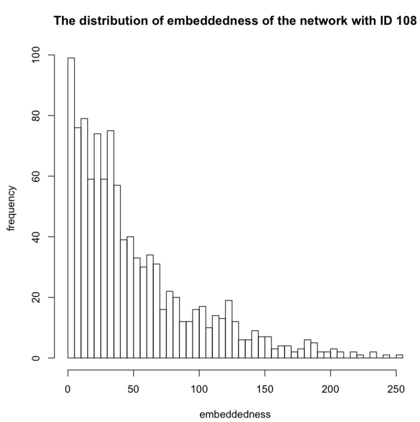

Figure 37: The distribution of embeddedness

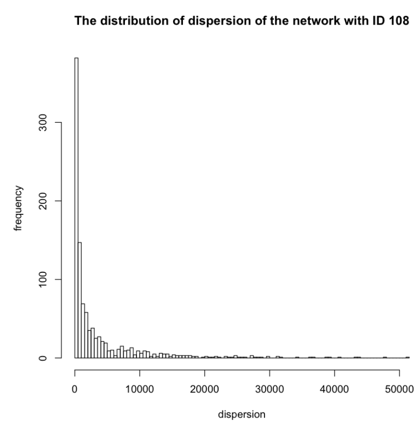

Figure 38: The distribution of dispersion

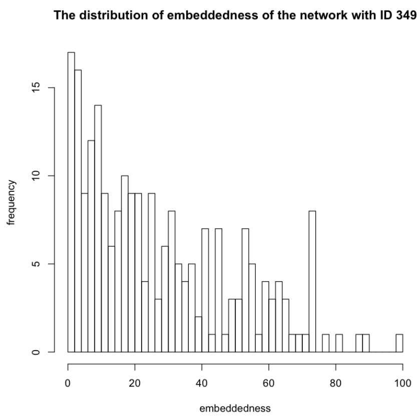

Figure 39: The distribution of embeddedness

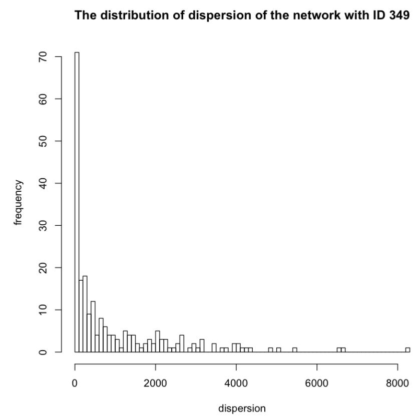

Figure 40: The distribution of dispersion

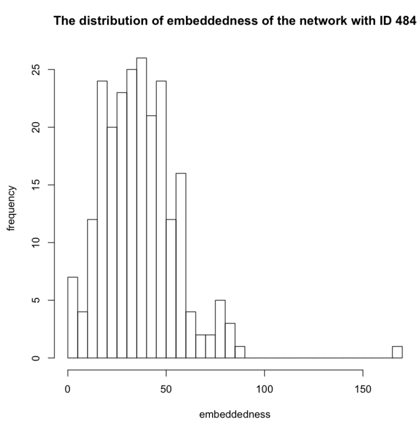

Figure 41: The distribution of embeddedness

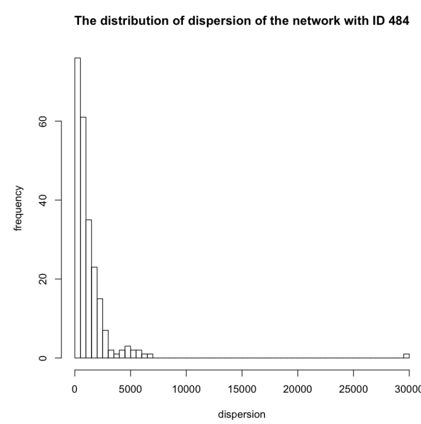

Figure 42: The distribution of dispersion

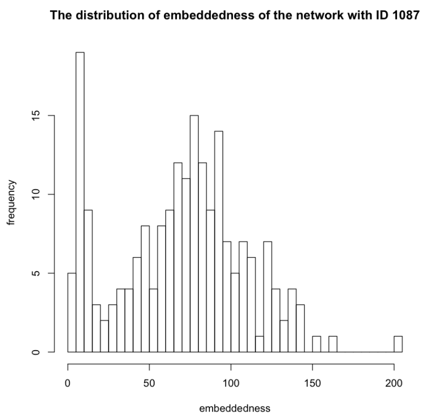

Figure 43: The distribution of embeddedness

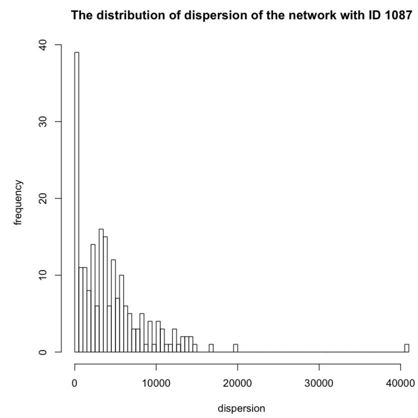

Figure 44: The distribution of dispersion

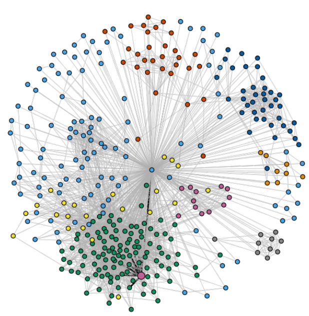

Figure 45: The community structure of No. 1

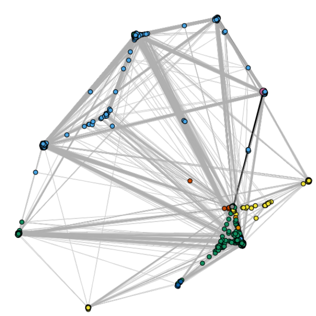

Figure 46: TThe community structure of No. 108

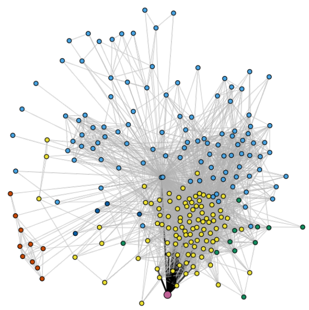

Figure 47: The community structure of No. 349

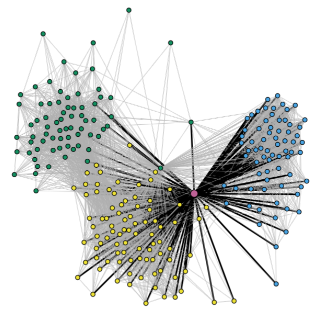

Figure 48: The community structure of No. 484

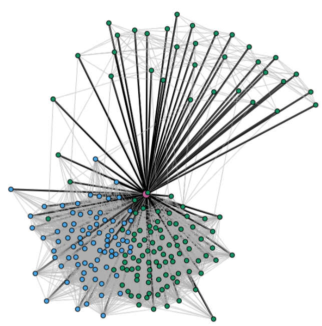

Figure 49: The community structure of No. 1087

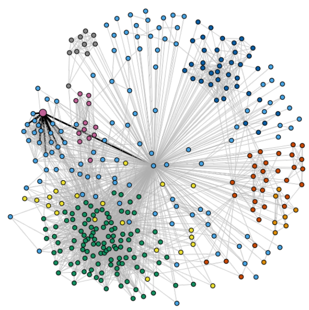

Figure 50: The community structure of No. 1

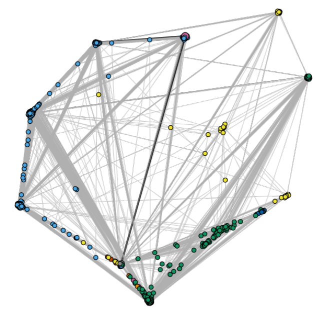

Figure 51: The community structure of No. 108

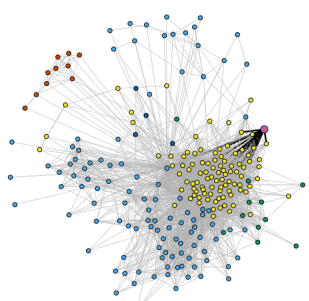

Figure 52: The community structure of No. 349

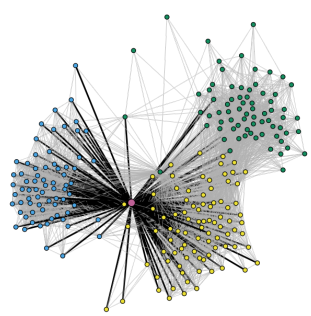

Figure 53: The community structure of No. 484

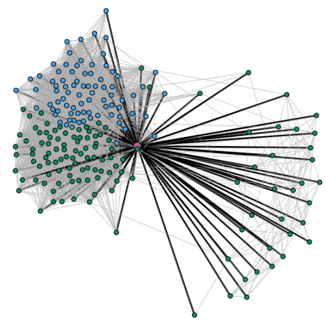

Figure 54: The community structure of No. 1087

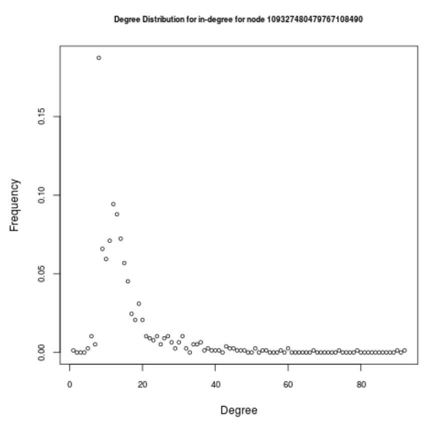

Figure 55: In Degree Distribution of ID:109327480479767108490

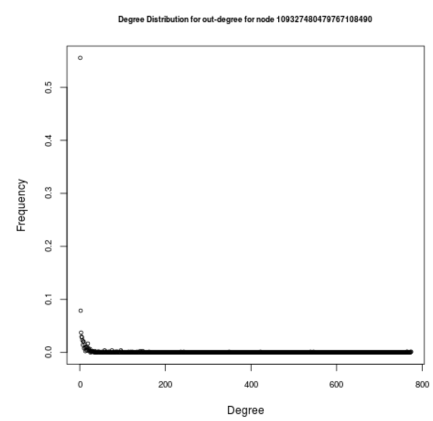

Figure 56: Out Degree Distribution of ID:109327480479767108490

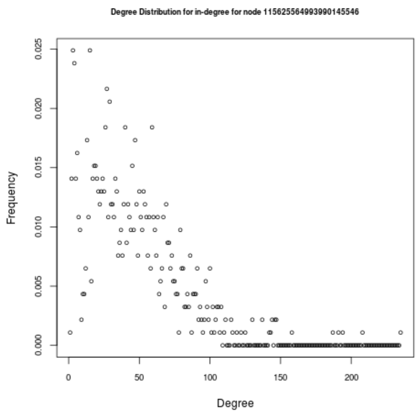

Figure 57: In Degree Distribution of ID:115625564993990145546

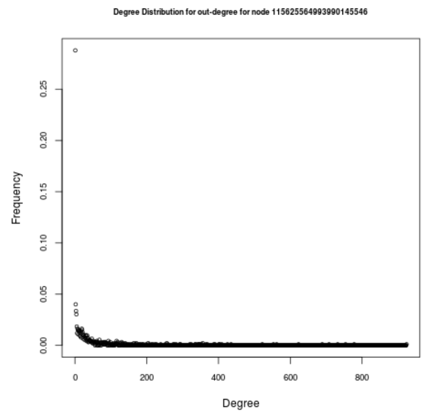

Figure 58: Out Degree Distribution of ID:115625564993990145546

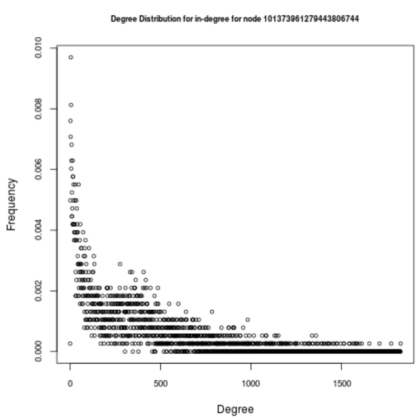

Figure 59: In Degree Distribution of ID:101373961279443806744

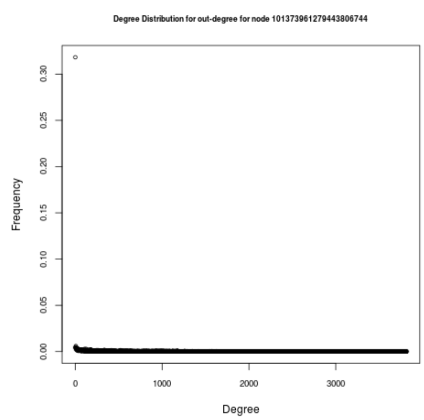

Figure 60: Out Degree Distribution of ID:101373961279443806744

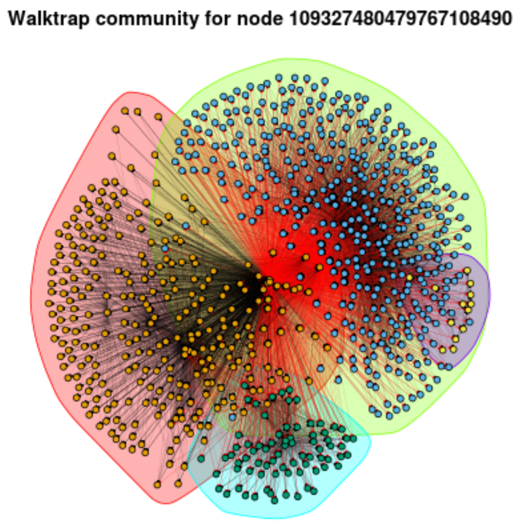

Figure 61: Walktrap Community of ID 1093274804797671 08490

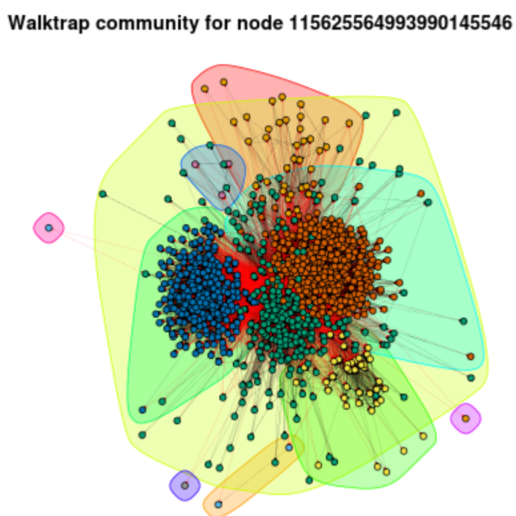

Figure 62: Walktrap Community of ID 1156255649939901 45546

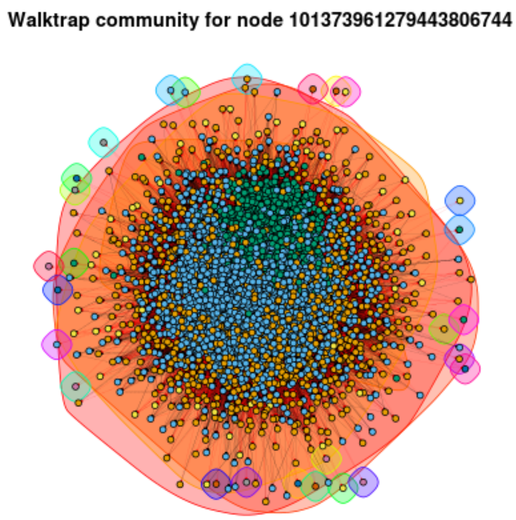

Figure 63: Walktrap Community of ID 101373961279443806744