Network Science Project

Pugachev Alexander

April 2020

Contents

1 Network Summary

1.1 Network source and preprocessing

For the Network Science Project I decided to build a graph based on my friends in VK social network[1]. At the beginning of April 2020 I have 536 friends in VK, 86 of them have deactivated or banned profile. 53 of my friends do not know any of my friends (except me), 8 people are partially in friends with each other and are not linked to the largest connected component of my friends' graph. Because of some of the graph characteristics can not be calculated in disconnected graph (for example radius) I will take into account only the largest connected component which consists of 388 nodes. The network visualization where the nodes sizes are presented according to their degrees is shown in Figure 1.

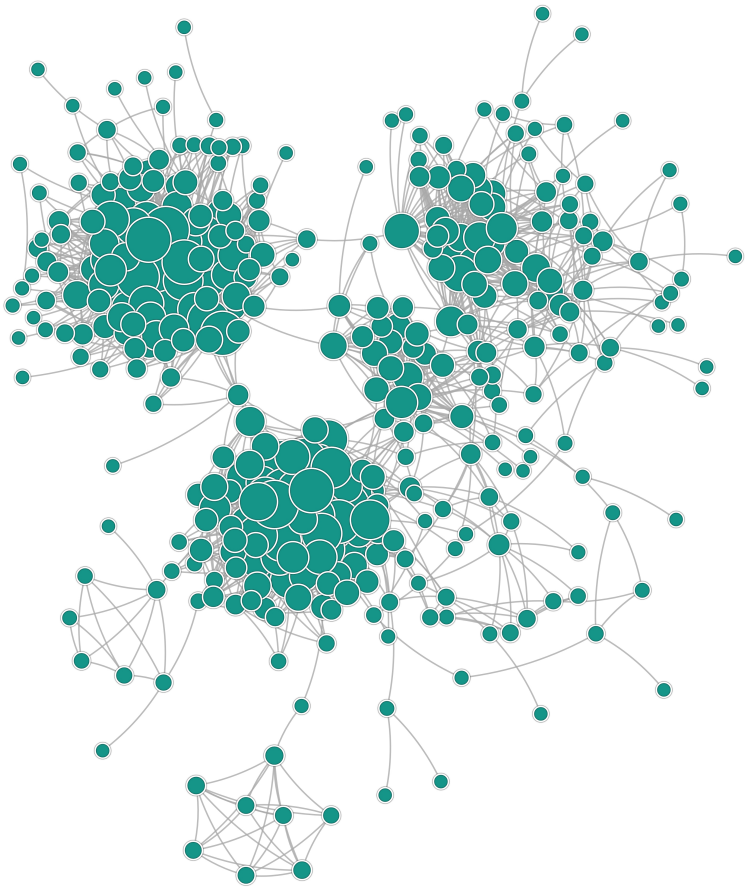


Figure 1: Friends network, node size according to node degree

In my friends network I distinguish these "clusters":

1. People from Kishinev (my home town)
2. People who have studied or are studying at HSE
3. People from Wylsacom VK group

[1]My VK profile: https://www.vk.com/a__pugachev

4. People from HSE camp
5. People from Sberbank Internship
6. People from HSE winter school

The first group are my friends and acquaintances from my home town Kishinev, former classmates and relatives. The second group are people I took bachelor degree with at HSE, my former groupmates. Also there are some people who are currently taking masters degree at HSE.

There exists a famous YouTube channel called Wylsacom[2]. About 8 years ago, when this channel was not so popular, it had a VK group[3] where people discussed technology news, gadgets, software and other topics. Right now this group has "read—only" mode, people can not post there anything, but in spite of this many of my friends are still in this group. So, the third "cluster" contains people who are in Wylsacom VK group.

About 7 years ago Higher School of Economics organized summer camps for pupils in CIS countries. In 2013 I participated in such camp in Moldova. At this camp employees and students from HSE talk about studying at HSE, ways how to go to the HSE, conduct career guidance games. The fourth group consists of people who participated in this camp with me.

The fifth group are people who were with me in the same internship team at Sberbank in 2018. The last one group consists of people who participated in HSE winter school on Computer Science in February, 2019.

Also there are people who do not fit to any of the above mentioned clusters.

1.2 Node and Edge attributes

As was said earlier, graph consists of 388 nodes connected with 2885 edges. Each node represents information about one of my friends the node attributes are the following:

- id: (string): Unique profile ID
- name: (string): First and last person's name
- sex: (string): Person's sex
- city: (string): Person's city
- in_hse (binary): Has person studied in HSE or not
- in_wylsa (binary): Is person a part of Wylsacom group or not

For a certain person some of the attributes (values) can be equal to *None*. It means that the person did not publish corresponding information on his VK page. Here is an example of one node:

- id: "85024880"
- name: "Борис Логашенко"
- sex: "Мужской"
- city: "Москва"
- in_hse: 1
- in_wylsa: 0

Two nodes (friends) are connected with an edge if and only if they are mutual friends. So we are dealing with an unweighted non-oriented graph.

[2]Wylsacom YouTube channel https://www.youtube.com/user/Wylsacom
[3]Wylsacom VK group https://vk.com/wylsacom

1.3 Graph properties

The graph **diameter** is the maximum distance between any pair of vertices. In our case the diameter is equal to 10.

The graph **radius** is the minimum among all the maximum distances between a vertex to all other vertices. In our case the radius equals 6.

The **clustering coefficient** for a certain node is the probability that two of its neighbors are connected. The average clustering coefficient is the mean clustering coefficient among all the vertices. For our graph the clustering coefficient is equal to 0.449.

The **average shortest path** is the average value of shortest paths between each pair of vertices. In our case it equals 4.103.

The **degree distribution** is the distribution of the number of nearest neighbours among all the nodes. Degree distribution for my graph is presented on Figure 2.

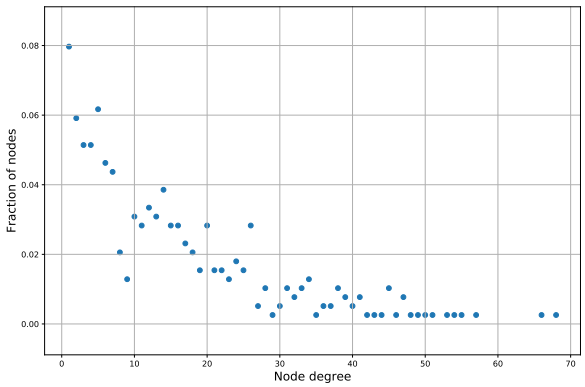


Figure 2: Degree distribution

2 Structural Analysis

2.1 Degree / Closeness / Betweenness centralities

The **degree centrality** measures the number of direct connections for a certain node. For node i degree centrality is calculated in the following way:

$$C_D(i) = \frac{\sum_j A_{ji}}{n-1}$$

where:

A — adjacency matrix,
n — number of nodes in the graph

The **closeness centrality** shows how close a certain node is to all other nodes in the network. For node i this centrality is calculated as follows:

$$C_C(i) = \frac{n-1}{\sum_j d(i,j)}$$

where:

$d(i,j)$ — path length between node i and node j,

n — number of nodes in the graph

The **betweenness centrality** for a certain node shows how many shortest paths go through this node. For node i betweenness centrality is calculated as follows:

$$C_B(i) = \frac{2}{(n-1)(n-2)} \sum_{s \neq t \neq i} \frac{\sigma_{st}(i)}{\sigma_{st}}$$

where:

σ_{st} — total number of shortest paths between node s and node t
n — number of nodes in the graph

The Top — 10 nodes for each of the centralities are shown in Table 1.

Degree centrality		Closeness centrality		Betweenness centrality	
Node name	Value	Node name	Value	Node name	Value
Александр Ермачков	0.173	Христофор Парфиров	0.337	Христофор Парфиров	0.183
Pavel)))	0.167	Александр Проскуряков	0.336	Стас Рыбин	0.151
Georgiy Demenchuk	0.144	Стас Рыбин	0.333	Степан Куртев	0.140
Валерий Батурин	0.142	Ксения Жилкина	0.326	Cjb Bot	0.128
Денис Тарасов	0.139	Антон Таныгин	0.319	Антон Таныгин	0.093
Антон Наумов	0.136	Михаил Флоринский	0.319	Андрей Парницкий	0.087
Лера Стоева	0.131	Максимилиан Артемьев	0.316	Инга Балан	0.082
Денис Шаклеин	0.126	Алексей Соловьёв	0.310	Pavel)))	0.080
Гера Войнов	0.124	Денис Самохвалов	0.309	Александр Проскуряков	0.074
Максимилиан Артемьев	0.124	Лера Стоева	0.309	Джульета Мурадян	0.054

Table 1: Top — 10 nodes according to different centralities

The Top—3 friends by degree centrality are people from Wylsacom group, the first one was the group administrator. Other people from Top—10 are my acquaintances from HSE. The majority of Top—10 people by closeness centrality are also former students of HSE university.

The visualizations of friends networks where node sizes are depicted according to degree, closeness and betweenness centralities are shown on Figures 3, 4 and 5 respectively. The Top-10 nodes are colored in red. The Top—10 by betweenness centrality is quite interesting, because it practically contains people from each of the clusters which I described in paragraph 1.1.

2.2 PageRank

Another way to calculate nodes' importance is **Pagerank**. The more links lead to a certain node, the more important this node is. The Top — 10 nodes according to PageRank score is shown in Table 2. The corresponding network is presented on Figure 6.

According to PageRank scores, Top—3 people are the participants of Wylsacom group. Other people from Top—10 are my former classmates and former groupmates.

2.3 Assortative mixing according to node attributes

As an assortative mixing metrics I chose assortativity coefficient. It shows how does the number of connections between the nodes with the same attributes differ from the connections in a random graph. For each node attribute except name and id I calculated assortativity coefficients. The results are presented in Table 3

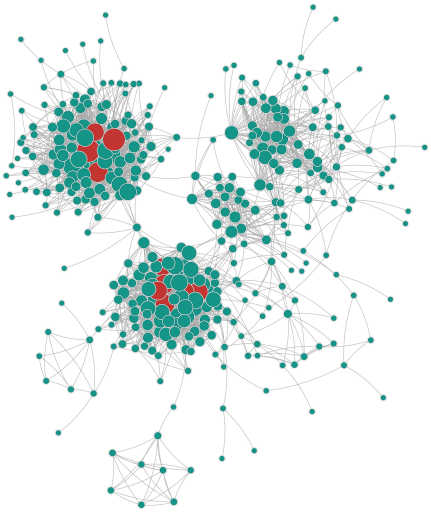


Figure 3: Friends network according to degree centrality.

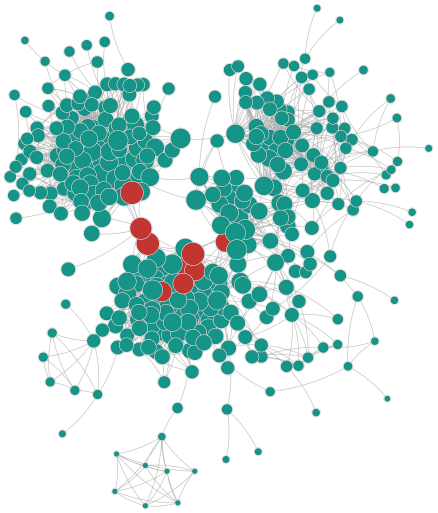


Figure 4: Friends network according to closeness centrality.

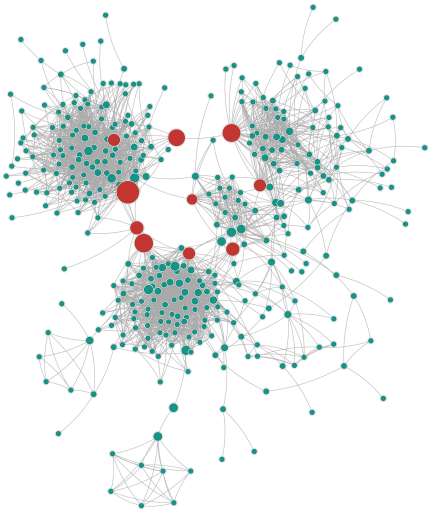


Figure 5: Friends network according to betweenness centrality.

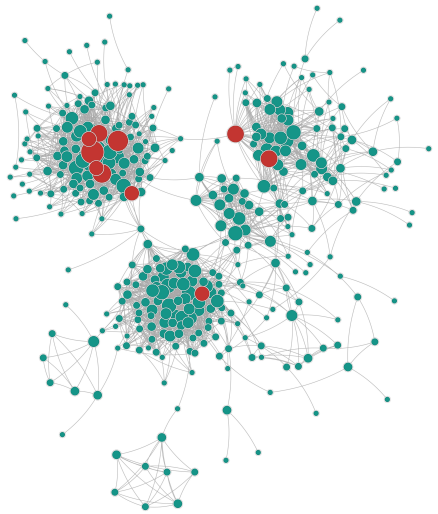


Figure 6: Friends network according to PageRank.

Pagerank					
#	Node name	Value	#	Node name	Value
1	Александр Ермачков	0.009	6	Гера Войнов	0.006
2	Pavel)))	0.009	7	Христофор Парфиров	0.006
3	Georgiy Demenchuk	0.007	8	Алексей Ушаков	0.006
4	Степан Куртев	0.006	9	Кирилл Громыко	0.006
5	Андрей Быстрицкий	0.006	10	Валерий Батурин	0.006

Table 2: Top — 10 nodes according to Pagerank

Based on the achieved results I can claim that nodes with the same "in_hse" attribute connect to each other more often than nodes with the same value of other attributes. It means the fact of studying in HSE (or not) is very important for determining whether two people are friends in VK. It is more likely that two people are friends if they both studied (or studying) in HSE than if they both are from the same city or have the same sex.

Attribute	Sex	City	in_hse	in_wylsa
Assortativity coefficient	0.245	0.107	0.394	0.199

Table 3: Assortativity coefficients

Talking about "in_wylsa" attribute, although Wysacom community which I talked about earlier is quiet specific, for the last several years the Wysacom YouTube channel became more famous and it is far from fact that people who are in Wylsacom VK group now were also there 8 years ago.

According to Table 3 I can affirm that two people of the same gender are more likely in friends than people of different gender. The same story is about city attribute.

2.4 Node structural equivalence / similarity

For the structural similarity research I decided to calculate cosine and Jaccard similarities and Pearson correlation coefficient for nodes in my friends network. Suppose that we have adjacency matrix of our network where in (i, j) cell is written 1 if there is connection between node i and j and 0 otherwise.

The **cosine similarity** between nodes i and j based on the adjacency matrix is being calculated like this:

$$\cos(v_i, v_j) = \frac{v_i^T v_j}{|v_i||v_j|}$$

where:

v_k — k-th column of adjacency matrix,
$|x|$ — norm of vector x

The **Jaccard similarity** between nodes i and j based on the adjacency matrix is being calculated in the following way:

$$Jac(v_i, v_j) = \frac{|N(v_i) \cap N(v_j)|}{|N(v_i) \cup N(v_j)}$$

where:

v_k — k-th column of adjacency matrix,
$N(v_k)$ — set of nodes v_k is connected with,
$|A|$ — number of elements in set A

The **Pearson correlation coefficient** between nodes i and j based on the adjacency matrix is being calculated in the following way:

$$r_{ij} = \frac{\sum\limits_k (A_{ik} - \langle A_i \rangle)(A_{jk} - \langle A_j \rangle)}{\sqrt{\sum\limits_k (A_{ik} - \langle A_i \rangle)^2} \sqrt{\sum\limits_k (A_{jk} - \langle A_j \rangle)^2}}$$

where:

A — adjacency matrix,
$\langle A_k \rangle$ — the proportion of nodes to which the node A_k is connected

The adjacency matrix, cosine similarity, Jaccard similarity and Pearson correlation matrices are shown on Figures 7, 8, 9 and 10 respectively.

Based on the obtained images I can claim that in my friends network there are both highly connected and weakly connected areas. The Top — 10 pair of nodes which are the most similar according to cosine similarity are presented in Table 4. Regarding Jaccard similarity and Pearson correlation, the Top — 10 is the same.

#	Node name 1	Node name 2	Cosine similarity
1	Лена Мимоглядова	Наталия Селюто	1.0
2	Ирина Белоусова	Владислав Бадеха	1.0
3	Алла Б	Владислав Бадеха	1.0
4	Алла Б	Ирина Белоусова	1.0
5	Александр Поляков	Группа Alter_ego	1.0
6	Анастасия Слободян	Anastasia Korol	1.0
7	Настя Лебедева	Николай Цалко	1.0
8	Дмитрий Карфидов	Николай Герасименко	0.912
9	Дина Нагибина	Дмитрий Карфидов	0.912
10	Георгий Плотников	Андрей Поляков	0.894

Table 4: Top—10 node pairs by cosine similarity

Node pairs #1, 8, 9 are people from Sberbank Internship, we are almost all VK friends for each other, that is why similarities are so high. Nodes from pairs #2, 3 and 4 have the highest possible cosine similarity because they have only one common friend - my fake VK page which was created and totally controlled by my friend. In truth, I do not know who are people from pair #5, but I know why their similarity is so high, it is because they have only one common friend (besides me). Pair #6 are my mom's friends and they also have only two common friends — me and my mother. Pair #7 are people who I played Clash of Clans with several years ago. We had a clan in this game, and people from pair #7 are friends with me and the head of the clan. The 10—th pair are people from HSE university, who have 4 out of 5 same friends.

2.5 The closest random graph model similar to my network

To determine what random graph model is the most similar to my graph I chose 3 graph models: Erdos—Renyi, Watts—Strogatz and Barabasi—Albert models. I built several random models, calculated properties for each of the model, the results are presented in 5.

In my opinion, the most similar to my friends network is Watts—Strogatz model. Despite its average node degree and number of edges differs from my graph, it has the closest average shortest path and clustering coefficient to my graph among all other models. Also, its diameter is closer to the diameter of my graph.

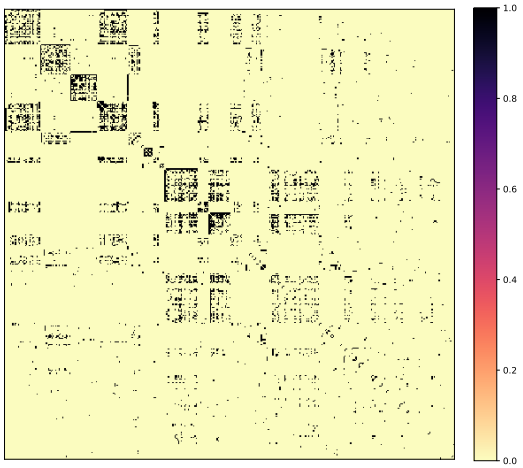


Figure 7: Adjacency matrix

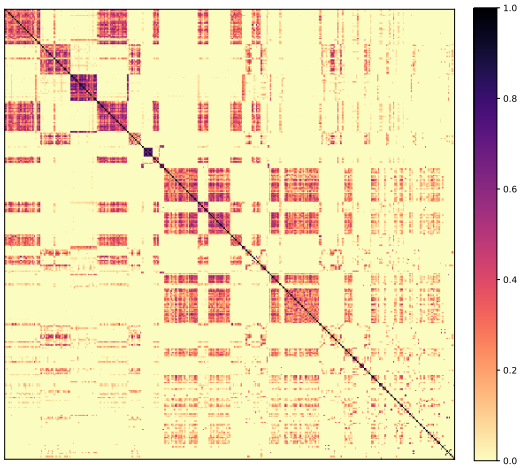


Figure 8: Cosine similarity matrix

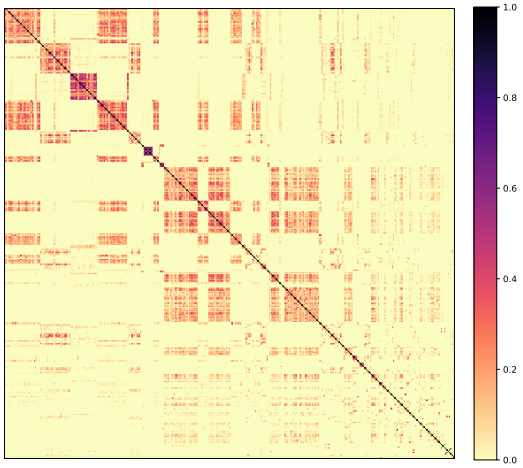


Figure 9: Jaccard similarity matrix

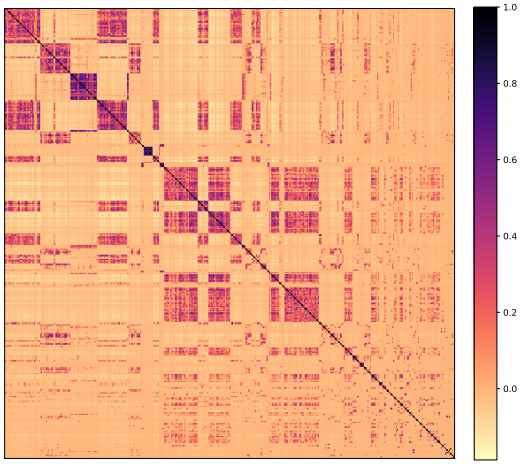


Figure 10: Pearson correlation matrix

	Friends Network	Erdos—Renyi	Watts—Strogatz	Barabasi—Albert
Parameters		$N = 388$ $P = 0.04$	$N = 388$ $K = 15$ $P = 0.05$	$N = 388$ $M = 8$
Number of nodes	388	388	388	388
Number of edges	2885	2903	2716	3040
Average node degree	14.871	14.963	14.000	15.670
Average shortest path	4.091	2.500	3.109	2.400
Average clustering coefficient	0.449	0.038	0.489	0.104
Diameter	10	4	5	4

Table 5: Random graph models comparison

3 Community Detection

3.1 Clique Search

The graph **clique** is an included complete subgraph. There are 2218 cliques in my VK friends graph. Clique size distribution is given on Figure 11.

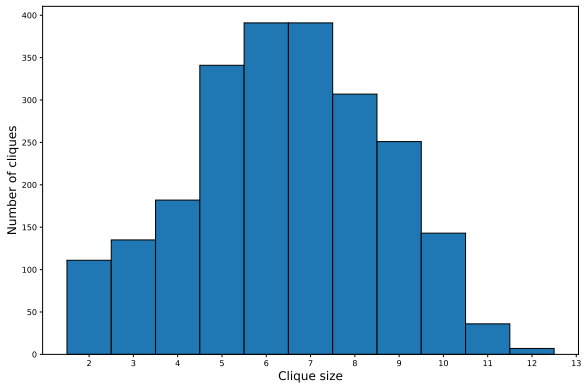


Figure 11: Clique size distribution

There are 7 cliques with size 12 which is the maximum for my graph. These 7 cliques consist of 20 unique nodes, the names of these nodes are given in Table 6. The visualizations of 2 of 7 biggest cliques are presented on Figures 12 and 13.

Андрей Соколов	Иван Толкушкин	Дмитрий Кириллов	Валерий Батурин	Илья Шубаев
Ксения Жилкина	Даша Чижова	Александр Цой	Сергей Павлов	Гектор Вереск
Денис Тарасов	Дарья Корепанова	Екатерина Докучаева	Астра Никитина	Антон Наумов
Денис Шаклеин	Юлия Ренёва	Оля Антонова	Pavel Хрушков	Виктория Тороп

Table 6: Nodes which are in biggest cliques

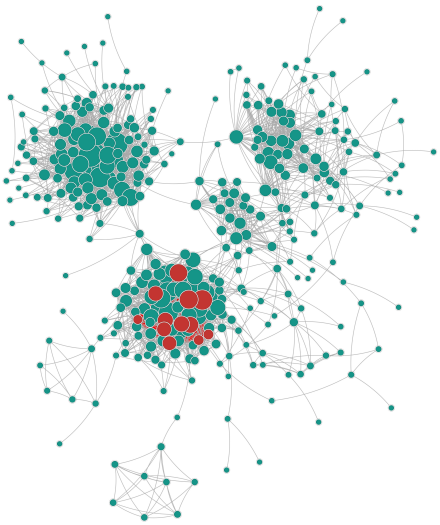


Figure 12: Clique #1

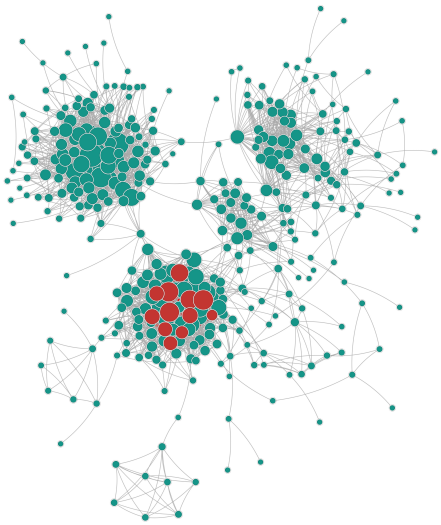


Figure 13: Clique #2

Each of the 20 nodes which form 7 biggest cliques represent people whom I studied with at HSE. 19 of these people are my former group mates. Even after graduation we did not remove from friends each other and sometimes resume communication.

3.2 Community detection

For the community detection research I decided to choose the following list of community detection methods:

- FastGreedy
- Edge Betweenness
- WalkTrap

The FastGreedy method is based on Louvain modularity optimization which is:

$$Q = \frac{1}{2m} \sum_{i,j} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j)$$

where:

A — adjacency matrix,
k_r — node degree of node r,
m — total number of edges,
c_r — community to which node r belongs,
δ — Kronecker function

The **FastGreedy** method increases the modularity in a greedy way. Initially all the nodes belongs to its own community, communities are merged iteratively such that each merge increases modularity score. The algorithm stops when it is not possible to increase the modularity.

The **Edge Betweenness** algorithm represents a decomposition process where edges are removed in the decreasing order based on their betweenness score. It is supposed that the betweenness of the edges connecting two communities is typically high, because many of the shortest paths between nodes in separate communities go through these edges.

The **WalkTrap** community detection method is based on random walks. The idea is that if you perform random walks on the graph, then the walks are more likely to stay within the same community because there are only a few edges that lead outside a given community.

The visualization of communities which were received by FastGreedy, Edge Betweenness and WalkTrap community detection methods are presented on Figures 14a, 14b and 14c respectively.

According to the visualizations that I obtained I can make the following conclusions. Firstly, each of the method was able to approximately identify and separate clusters which I described in Paragraph 1.1. For example, on Figure 14a clusters are labeled like this:

- #1 — HSE students cluster
- #2 — HSE camp cluster
- #3 — Kishinev cluster
- #4 — HSE winter school cluster
- #5 — Sberbank Internship cluster
- #6 — Wylsacom cluster

The Edge Betweenness method unlike other methods has number of communities as a parameter, I ran this method with 7 clusters because I suppose that there are nodes that do not belong to any of the above mentioned clusters. This 7—th cluster are only 3 people whom I played Clash of Clans with and they really do not intersect with any of the 6 "base" clusters.

The FastGreedy approach besides 6 "base" clusters and Clash of Clans community detected 2 more (communities #7 and #8 on Figure 14c). Cluster #8 consists of 2 people which are connected to the one of the biggest (in terms of node degree) nodes from Wylsacom cluster. The community #7 can be clearly seen when we take Figures 14b and 14c) and compare its right parts. We can see that there is a long round path between community #1 and #2 whose nodes belong to different communities. Despite the fact that these nodes are quiet similar to a separate community, all of them represent people who studied at HSE and belong to the community # 1.

The WalkTrip algorithm detected 10 communities in my friends network. The 6 "base" communities were successfully identified as well as Clash of Clans community. Comparing to the FastGreedy algorithm, the WalkTrip made significant changes in community #1 which represents HSE students. He separated several group of nodes and divided them into two clusters, which is unfortunately mistakenly because almost all of the nodes from these 2 new clusters must belong to the community of people who studied at HSE. The 10—th cluster consists of three people - my mom and two her friends, who were mentioned in Paragraph 2.4.

The modularity score for each method is presented in Table 7.

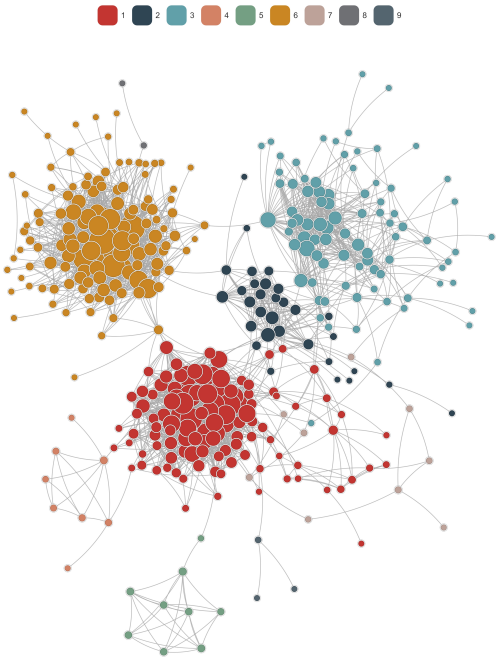
Method	Edge Betweenness	FastGreedy	WalkTrap
Modularity	0.654	0.655	0.6622

Table 7: Nodes which are in biggest cliques

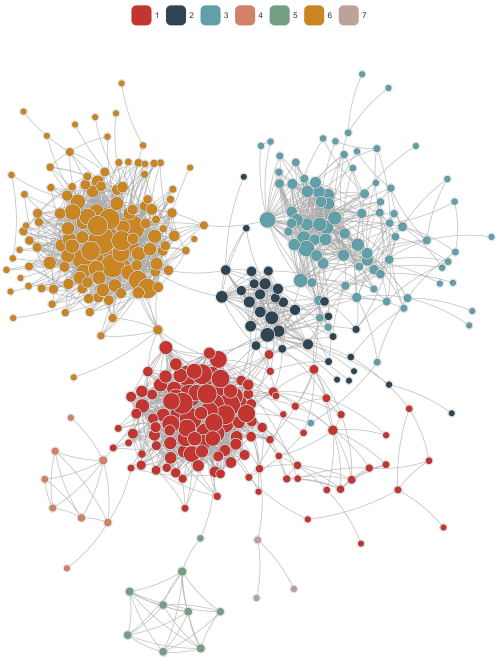
4 Technical details

The code for all tasks was written on Python programming language version 3.6.8. I used the following Python libraries:

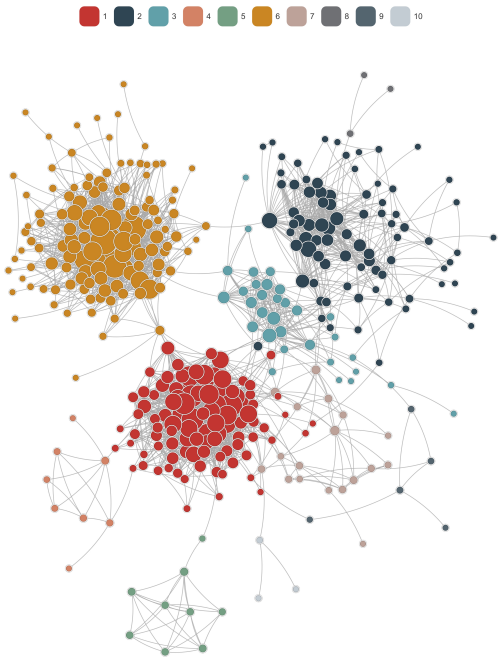
- python—vk 2.0.2
- networkx 2.4



(a) FastGreedy



(b) Edge Betweennes



(c) WalkTrip

Figure 14: Communities visualization

- progressbar2 3.47.0
- matplotlib 3.1.3
- pyecharts 1.7.1
- python-igraph 0.8.0
- scikit-learn 0.22.1

The code for this project is fully reproducible and freely available on Github[4]. The interactive visualizations are also available on GitHub

5 References

[1] Zhukov L. Network Science lectures, 2020.

[4]Alexander Pugachev Github https://github.com/apugachev/network-science-project