**National Research University Higher School of Economics (Higher School of Economics/HSE)**

*Faculty of Computer Science*

Bachelor's Programme Data Science and Business Analytics

01.03.02 Applied Mathematics and Computer Science

**Internship report**

Fulfilled by

*Nikitin Bogdan Nikitich*

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*(signature)*

**Checked by**

|  |  |  |
| --- | --- | --- |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_* | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| (*job or academic title*) | (*surname, initials)* | (*signature*) |

**Moscow, 2019**

**Introduction**

**Internship goals:** consolidation, expansion and deepening of the theoretical knowledge and acquirement of initial practical skills in solving specific problems

**Tasks:**

* 1. consolidation and deepening of the theoretical knowledge on the disciplines passed at the university;
  2. acquirement of information competence for the purpose of successful work in professional activities;
  3. obtaining skills of both independent and team work.

**Contents**:

For a given VK community, it is necessary to detect sub-communities inside, investigate them and ﬁnd dependencies.

*Table 1*

**Educational Internship Schedule (Plan)**

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Calendar period** | **Work Plan** | **Internship Supervisor’s signature** |
| 1 | 01.07.2019 | 1. Organizational (induction) meeting |  |
| 2 | 01.07.2019 | 2. Instructing on the requirements of labor protection, safety, fire safety and internal labor regulations |  |
| 3 | 01.07.2019 -13.07.2019 | 3. Fulfillment of Individual Assignment |  |
| 4 | 01.07.2019 -13.07.2019 | 4. Consultation |  |
| 5 | 14.07.2019 | 5. Preparation and submission of the Report |  |

**Contents**

[**Introduction** 2](#_Toc13529981)

[**Educational Internship Schedule (Plan)** 2](#_Toc13529982)

[**Description of the learnt materials** 3](#_Toc13529983)

[**Description of the results** 3](#_Toc13529984)

[**Conclusion** 4](#_Toc13529985)

[**Bibliography** 4](#_Toc13529986)

[**Appendix and supplementary material** 5](#_Toc13529987)

**Description of the learnt materials**

During the Educational Internship Programme I conducted a research on a group in social media service vk.com. The url of the groups I conducted the research are <https://vk.com/csposvyat18>, <https://vk.com/dormitory8hse>. These are two groups with mostly students of HSE subscribed, hence we could speculate and check the nature of community formations. I obtained the data using Python and VK API – open application for developers of vk.com. The data I obtained is represented in two csv files, the first one is the table of matching of id with full name of a person, the second one is the list of edges between each id (matching people to each other).

To analyze the data, I transformed it into a graph, using Python library NetworkX and applied Karger’s algorithm, which I implemented by hand, and K-clique algorithm, which I implemented using NetworkX library. To optimize analysis for a big data collection I also implemented Karger’s algorithm in C++ and created an interface to conveniently process the data. The result I also collected into csv files with the same structure as for obtained data.

To visualize the graphs and make mid-term solutions, I used NetworkX methods with Matplotlib. To analyze the number of communities, modularity and the number of connected components in the graph I used Gephi 0.9.2.

**Description of the results**

In the result I received an application that allows me to analyze any Community in vk.com, process and collect the data, draw graphs and find sub-communities. I also found out the differences between different analytical algorithms.

The figures of the graphs are put in the supplementary materials. On picture 1[1] we can observe the graph that was processed. I used Yifan Hu’s proportional drawing so we can see the shaping of sub-communities in the graph.

Being precise I put the node labels on picture 2[2]. Here we can already notice what nature lies in the sub-group division. It is commonly in accordance with Educational Program on which one studies. By the picture 3[3] we can observe the centers of clusters of sub-communities. If we compare it with the information from the picture 3[3], we can admit that these attraction points are mostly curators of study groups, (Nikita Veselko who is one of the most active curators on faculty in one of the most considerable attractors in the graph).

Transforming these groups using community algorithms we can receive completely different results. From it we can accept that different algorithms are suitable for different cases. For example, on picture 4[4] we obtained the result using Karger’s algorithm. For convenience I used Fruchterman Reingold’s drawing to show the structure. There I implemented Karger’s algorithm to each strongly connected component of the graph and using Gephi visualized all min cuts in each strongly connected component. Hence, we see the proportion in the number of nodes belonging to each community and can judge of different sizes of sub-groups and also see different connectivity components. In addition, this algorithm allows us to get access to one of the edges from the min cut and also collect the data about nodes from different communities.

On the picture 5[5] and 6[6] I implemented K-clique algorithm with different drawing. There we admit 6 modularity classes. The hole set of cliques forms a single connectivity component. In comparison to Karger’s algorithm this is more flexible and allows us to scale the number of modularity classes (i.e. sub-communities) using the k value. This algorithm is mostly aimed to distinguish the most sub-communities, with the big number of persons, while the previous one is random, but mostly ideal to find the smallest sub-groups as the have fewer cut.

**Conclusion**

In conclusion, I would like to repeat that, as I mentioned before, Karger’s algorithm is more suitable to locate smaller subgroups as it results in the min cut, while the K-clique algorithm addresses the need in finding bigger sub-groups. Considering the complexity of both algorithms for Karger’s algorithms it is O(|V|2) and O(|E| log |E|), so it is faster when we have the small number of edges, while the K-clique algorithm complexity is rapidly changing with increase of K and implemented using nodes to count the complexity.

**Bibliography**

1. Fortunato S. Community detection in graphs //Physics reports. – 2010. – Т. 486. – №. 3-5. – С. 75-174.

**Appendix and supplementary material**























