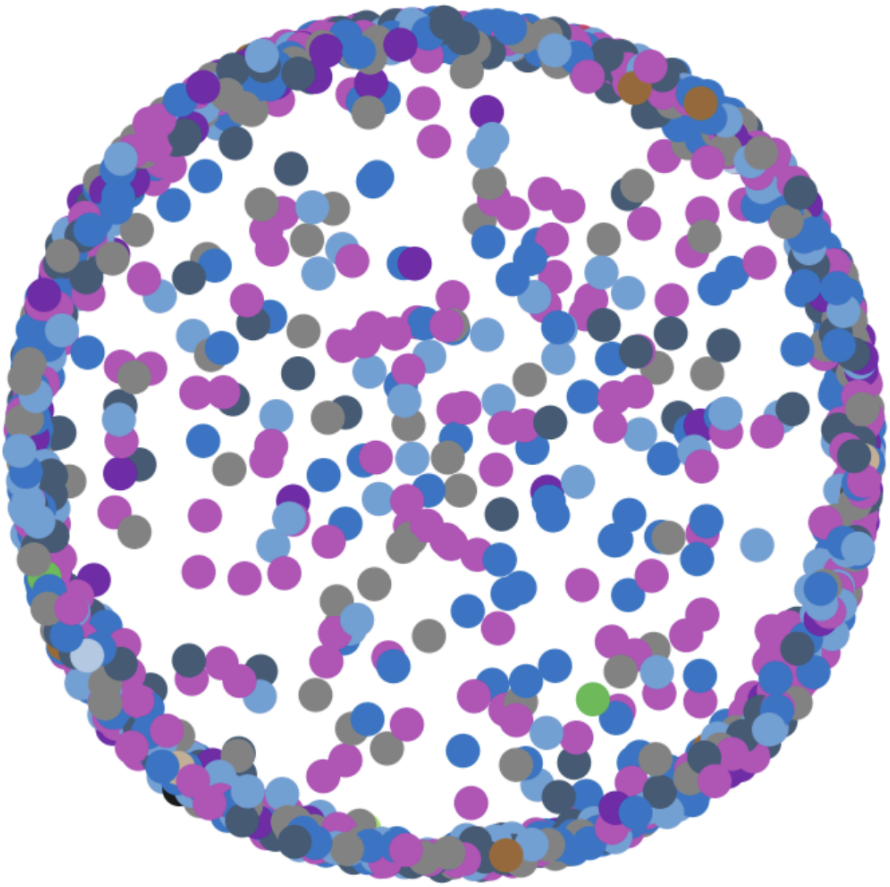
CS176 Project Report: Youtube Social Network Analysis



Shreya Raj and Erica Xue
CS 176 Sect 01
Katerina Potika

Abstract

This report includes an in depth analysis of an undirected graph of the Youtube social network. Youtube allows users to form friendships and build communities of similar interests and hobbies. These connections, or lackthereof, will be examined to better understand the strengths and structure of the network. The project report will cover the following sections: Introduction, Algorithms, Experiments, Conclusion, and References.

Introduction

The project dives into the analysis of the Youtube social network - a platform that forms "a user network where the nodes represent the users and the edges represent the social ties (friendship) between users" [4]. The dataset is from Stanford Network Analysis Project (SNAP) and contains 1,134,890 nodes (representing users) and 2,987,624 edges (representing friendship connections). The network has an average clustering coefficient of 0.081, a diameter of 20, and 8,385 communities [5]. It is important to understand the structure and cohesiveness of the network to analyze and comprehend how information flows. With that information, we can identify which nodes are significant, the roles of different nodes, and how communities form. This can also shed light on how well representative the network is of real world social interactions and behavior.

Algorithms & Experiments

This section will include an overview of the algorithms and measures used in this project such as the sampling process, graph metrics, centrality measures, and community detection. The Youtube social network is a very large dataset so we need to randomly sample. First, we sorted the nodes, got the top 5000 nodes with the highest degrees, and created a subgraph with them. Then we sorted the edges in ascending order, starting with the smallest node ID. This helps ensure consistency. Following that, we created a random seed of 42 to be able to replicate our results. We randomly sampled 15,000 edges from the subgraph then added the sampled nodes and edges into a set, ensuring no duplicates. Lastly, we added the nodes and edges together to create a sampled graph. The randomly sampled network got a sample size of 4,568 nodes and 15,000 edges. We also got an average degree of 6.57, a clustering coefficient of 0.015, and that the graph is not connected. These statistics can be seen in Figure 1 below:

```
Number of Nodes: 4568
Number of Edges: 15000
Average Degree: 6.567425569176883
Clustering Coefficient: 0.015436066577593293
Is Graph Connected: False
```

Figure 1. Sampled Graph Statistics

Next, we found the average degree, clustering coefficient, and whether or not the graph is connected. The average degree is calculated by 2 multiplied by the number of edges then divided by the number of nodes. This refers to the average number of edges for each node in the graph. In the sampled graph, we got the result of 6.56. A higher degree average means that the nodes have more connections and creates a more connected and dense graph. The next statistic is the clustering coefficient which is the likelihood that two neighbors of a selected node will form connections or become friends, and form a triangle. We got 0.015 which is relatively low, meaning that in the sampled graph, most of the neighbors of a node are not connected to each other and do not form a triangle structure. We also checked if the graph is connected, which implies that there is a path between every pair of nodes in the graph. In our graph, it was not connected, meaning that there was at least one pair of nodes that were not connected by an edge, creating disconnected components. These statistics can be seen in Figure 1.

Other graph metrics we looked at are the bridges, articulation points, diameter, and degree distribution. Bridges are important edges that keep the graph connected and if they are removed, it creates disconnected components. A greater number of bridges means that there are many edges that are extremely critical to keep the bridge connected, and there aren't any other alternative paths. Articulation points are similar but they refer to the nodes, and if the node is removed, it creates disconnected parts in the graph. The same logic about the greater number of bridges present can be applied about the greater number of nodes in the network. The graph has 577 bridges and 469 articulation points, which are 3.85% of the total edges and 10.3% of the total nodes respectively. This is shown in Figure 2.

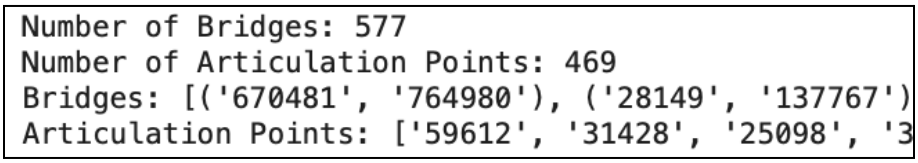


Figure 2. Bridges and Articulation Points

There is a moderately low number of bridges present which means although there are significant edge connections, there are also alternative pathways. There is a moderate number of articulation points since approximately 1 in 10 nodes play a key role in maintaining graph connectivity. Combined, the graph is moderately vulnerable to disconnection and it is not an extremely dense network. Additionally, since the graph is disconnected, we cannot find the entire diameter. Instead, we found the diameter of the largest connected component of our network, which is 14 (seen in Figure 3). To find the largest diameter, we created a components list of the connected components, found their diameters using a for loop, and appended the diameters into a diameter list. Then we found the largest one using the max (diameter).

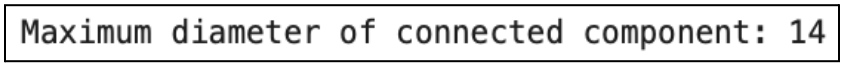


Figure 3. Maximum Diameter of Connected Component

We also drew a degree distribution histogram to get a better understanding and view of the node degrees. The visualization supported the idea of power-law degree distribution where most nodes have a small number of neighbors (resulting in a smaller degree) and other nodes have a very high degree. The histogram is shown in Figure 4.

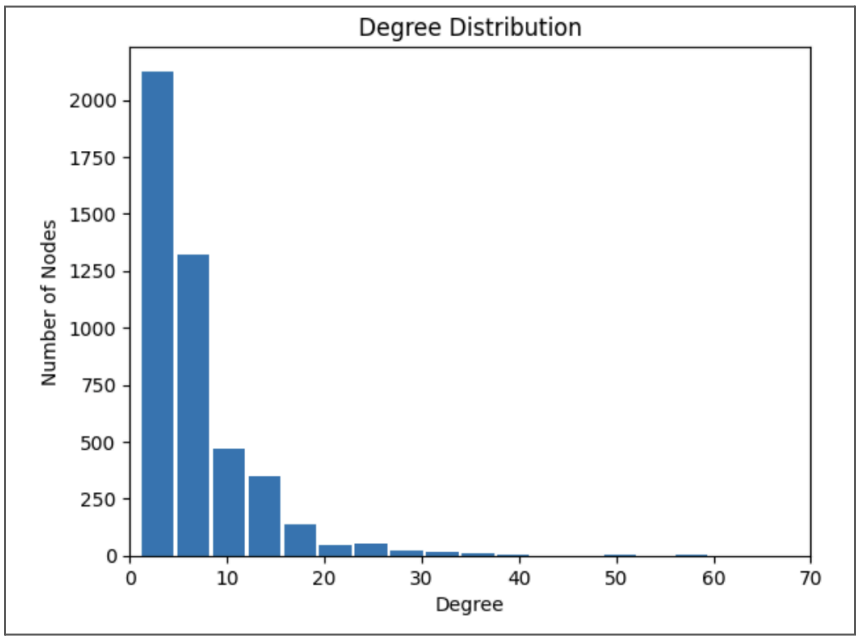


Figure 4. Degree Distribution Histogram

Centrality refers to how the center equates to importance. The three centrality measures we explored were degree, closeness, and betweenness. Degree centrality is how connected each node is. Closeness centrality is how easy a node is able to reach other nodes. Betweenness is how important a node is in regards to connecting other nodes via a path. The higher these values are for each node, the more vital they are to the network. These measures allow us to understand which nodes are most important and the dynamics of the graph. Below are examples of the first 5 degree centrality values (Figure 5), closeness centrality values (Figure 6), and betweenness centrality values (Figure 7).

```
137767    0.000438
5754      0.001971
28233     0.000438
4964      0.001314
30816     0.002409
dtype: float64
```

Figure 5. Degree Centrality

```
137767    0.178253
5754      0.228315
28233     0.222915
4964      0.216543
30816     0.218049
dtype: float64
```

Figure 6. Closeness Centrality

```
137767    0.000218
5754      0.000225
28233     0.000008
4964      0.000491
30816     0.001756
dtype: float64
```

Figure 7. Betweenness Centrality

We also visualized the first and second largest connected components of the graph. We used the components list we created initially when finding the maximum diameter of the connected component in the graph. We sorted the components by length in decreasing order so the largest component would be in the first index. The largest component is shown in Figure 8 below. It has a diameter of 14 as mentioned earlier.

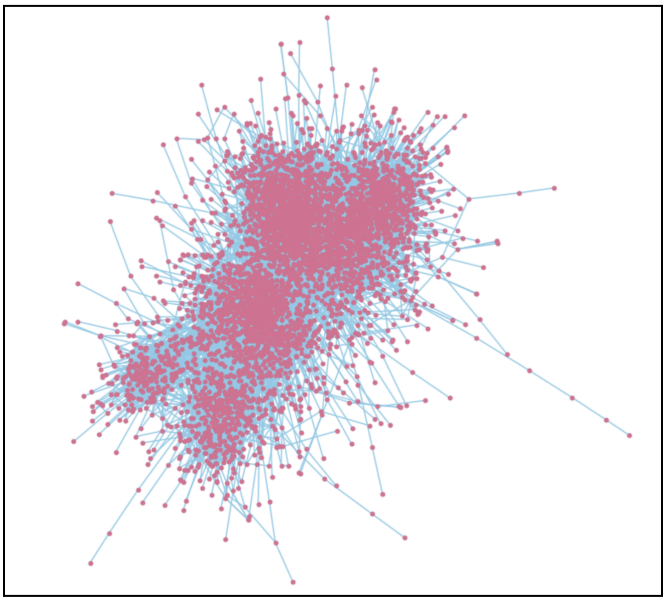


Figure 8. Largest Connected Component

The second connected component is in the first index of the components list. It is just two nodes with an edge connecting them. See Figure 9.

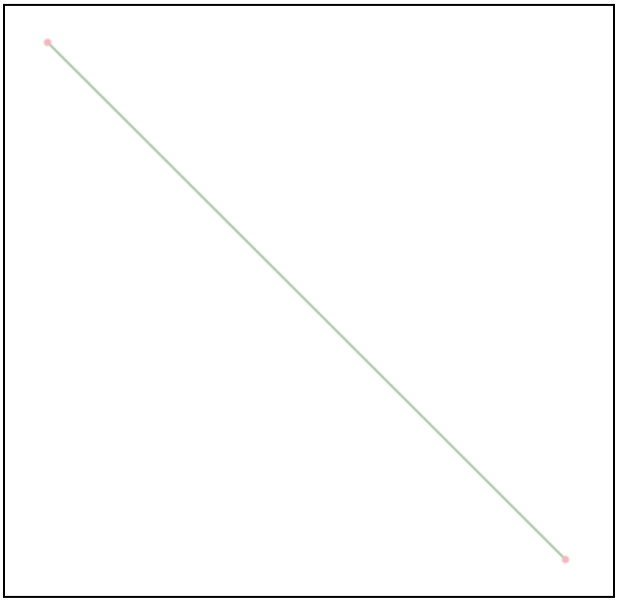


Figure 9. Second Largest Connected Component

Lastly, for community detection we used Louvain's algorithm and Girvan Newman's detection algorithms. Before implementing these two algorithms, we created three functions: set_node_community, set_edge_community, and get_color. The first one assigns community labels to the nodes, the second one assigns community labels to the edges, and the third one generates different colors for visualization of the communities. It is a greedy bottom up optimization approach with a time complexity of O(nlog(n)). Each round is made up of two phases. The first round is when each node begins as its own "group," and then the highest change in modularity is calculated and the specific nodes are joined. The distinct communities are combined into super nodes to build a new network and the two phases repeat. Louvain is very fast but it is only able to find big communities as it suffers from a resolution limit; by optimizing modularity, communities that are smaller will be combined with bigger groups even if they are meaningful [3].

We started by performing Louvain's algorithm. The results were 24 communities and a modularity score of 0.603, which means the graph has a pretty strong community structure. The first visualization includes 500 nodes with edges and their distinct community labels as seen in Figure 10.

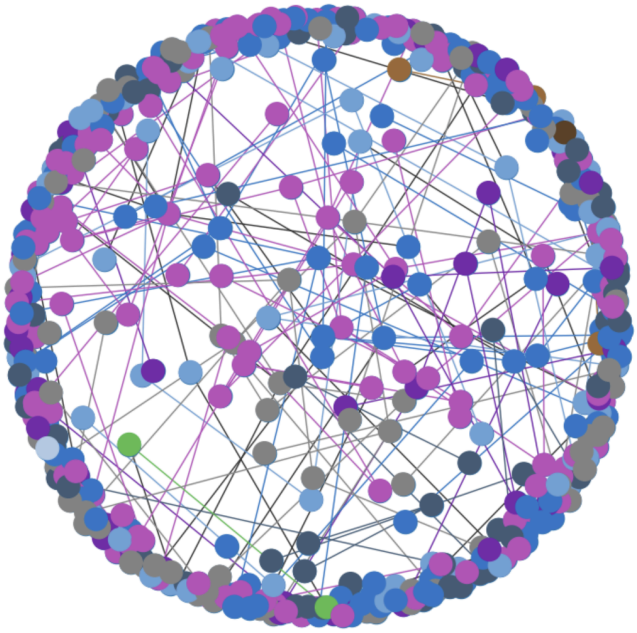


Figure 10. Louvain Algorithm

Figure 11 illustrates the second visualization, which has 1,250 nodes but no edges to make it more clear and readable. In both images, we can see a couple of the different communities represented by different colors. Some examples are the dark blue, blue, purple, dark purple, gray, brown, and green communities. It makes sense that there are various communities because there is so much to watch and explore on Youtube - allowing different interest groups to

form and grow. We also visualized the largest community found by Louvain's algorithm in Figure 12.

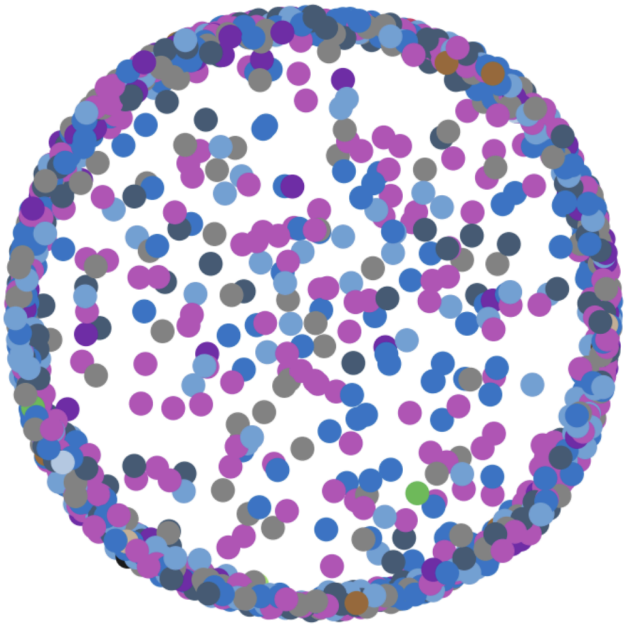


Figure 11. Louvain Algorithm Pt.2

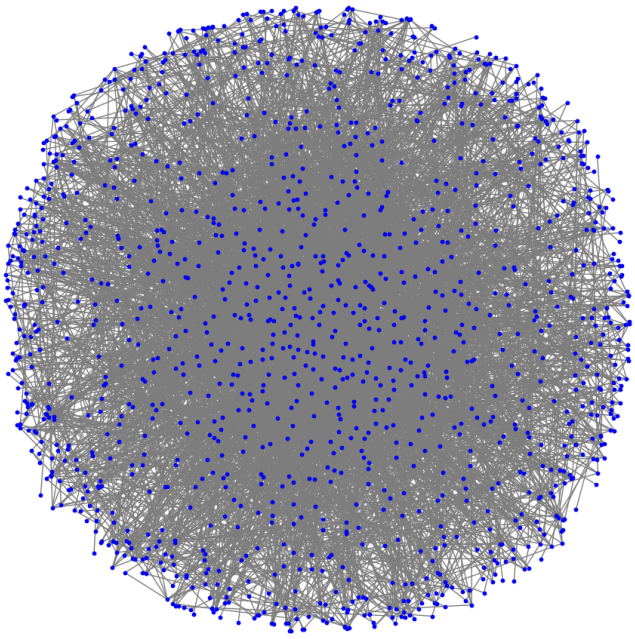


Figure 12. Louvain - Largest Community

The next community detection algorithm we used is Girvan-Newman. It is a divisive (top-down) hierarchical clustering based on the calculated edge betweenness of each node. It

works for indirect and unweighted networks. First, you need to use breadth first search to calculate the betweenness of edges and remove the edges with the highest betweenness centrality value. You repeat this until there are no edges left. The connected components left are communities and this algorithm helps give a hierarchical decomposition of the network [3].

In our results, we did two rounds of this algorithm. For the first round, we got 318 communities and a modularity score of 0.874. For the second round we got 319 communities and a modularity score of 0.896. These values indicate a fairly strong and significant community structure. We visualized the top 20 communities since Girvan Newman is not as fast as Louvain, and it would take an extremely long time to compute. The visualization can be seen in Figure 13. You can see that the "nodes explicitly join various interest based social groups" since the dataset revolves around the Youtube Social Network [1].

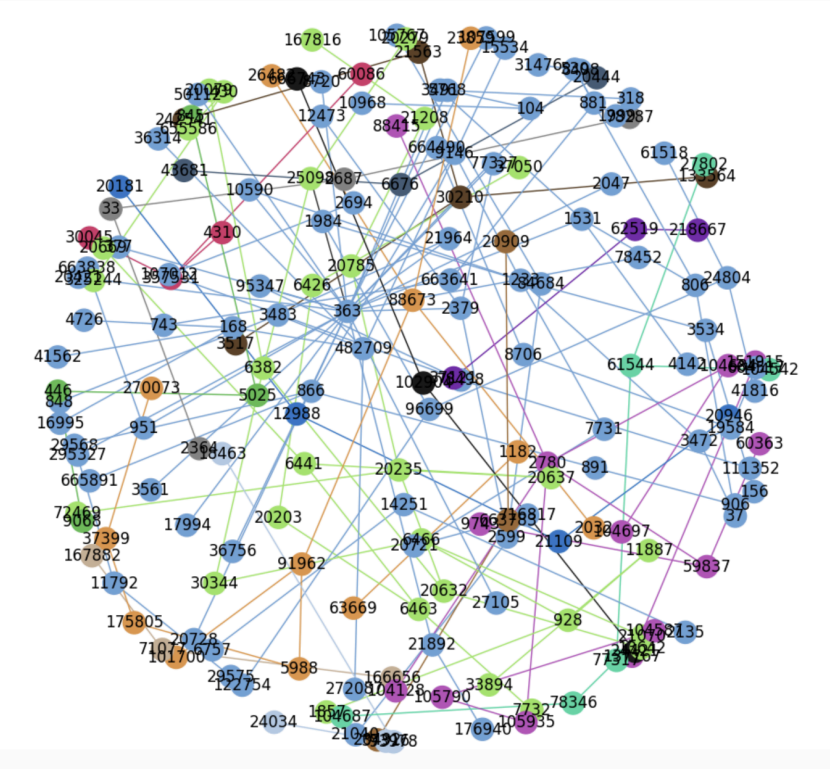


Figure 13. Girvan Newman - Top 20 Communities

Conclusions

Overall, the Youtube Social Network platform demonstrates a strong community and small-world structure. Louvain's algorithm produced a modularity score of 0.603 and Girvan Newman's algorithm resulted in an average modularity score of 0.885. These values indicate a fairly strong and decent community structure. The small world phenomenon refers to how most nodes are a few edges away on average [2]. We can gather this from the diameter of the graph, the number of communities, and low clustering coefficient. The graph is also relatively

vulnerable to disconnectivity due to the high number of bridges and articulation points compared to the total number of nodes and edges. The power-law degree distribution supports the same expected patterns of real world social networks since it highlights the large number of users with few connections (everyday users/Youtube watchers) and the smaller number of influential users (content makers with connections and a big following). The Youtube social network demonstrates typical social network structure and behavior and can be generalized to other platforms as well.

References

[1] J. Yang and J. Leskovec, “Defining and Evaluating Network Communities based on Ground-truth,” *arXiv (Cornell University)*, May 2012, doi: https://doi.org/10.48550/arxiv.1205.6233.

[2] K. Potika. CS 176. Class Lecture, Topic: “Lecture 1 Graphs.” San Jose State University, San Jose, CA, September 18, 2024.

[3] K. Potika. CS 176. Class Lecture, Topic: “Lecture 3 Ties and Communities.” San Jose State University, San Jose, CA, September 18, 2024.

[4] M. Jebabli, H. Cherifi, C. Cherifi, and A. Hamouda, “User and group networks on YouTube: A comparative analysis,” *IEEE Xplore*, Nov. 01, 2015. https://ieeexplore.ieee.org/abstract/document/7507126?casa_token=Y7NdcAYhbqQAAAAA:Vm0DGsYvTTP-rzug5lzhcuqEBou08mVkIAkqiLL1n1OF6aQX-R2rJ4j5VuklB86crwdZcPiX (accessed May 09, 2022).

[5] “SNAP: Network datasets: Youtube social network,” *Stanford.edu*, 2024. http://snap.stanford.edu/data/com-Youtube.html (accessed Dec. 08, 2024).