Illinois Institute of Technology

Instructor: Dr. Cindy Hood

CS 579- Online Social Network Analysis

Project 1

GitHub Analysis: Mapping User Relationships

Group

Anushka Manish Chaubal

CWID: A20511568 Email: achaubal@hawk.iit.edu

Rewa Sanchit Deshpande

CWID: A20492328 Email: rdeshpande1@hawk.iit.edu

Abstract

This report presents the findings and methodologies employed in the analysis of social media - ***GitHub*** data, focusing on the utilization of the GitHub API to construct a friendship network among users with significant repository activity. Initially exploring various social media platforms such as Instagram and Twitch, the project encountered limitations with API access. Ultimately, GitHub emerged as a viable platform due to its accessible API. The analysis involved data collection, visualization, and calculation of network measures using Python libraries like ***NetworkX***, ***pandas***, and ***Matplotlib***. The report discusses the challenges faced, methodologies employed, and insights gained from the analysis.

Contents

List of Figures

1 Introduction

Social media platforms offer a wealth of data that can provide valuable insights into user interactions and behaviors. This project focuses on leveraging network analysis techniques to explore these insights, with a specific emphasis on understanding friendship networks.

The project utilizes network analysis to delve into the dynamics of social interactions within digital spaces. By examining friendship networks, we aim to uncover patterns and structures that shed light on how users connect and interact on social media platforms.

GitHub emerges as the primary platform for analysis due to its accessible API, which enables comprehensive data collection and exploration of friendship networks within its community. Through this project, we aim to gain a deeper understanding of social media dynamics and the underlying mechanisms that govern user interactions within online communities.

1.1 Why Git?

We chose to focus on GitHub for our analysis mainly because it has an easy-to-use API that allowed us to collect data without any hassle, unlike other social media platforms we looked at. GitHub is all about people working together on projects, so it naturally creates a lot of connections between users, which makes it perfect for studying friendships. By using GitHub's API, we were able to dig into how people are connected online and learn a lot about how friendships form in digital communities. Overall, GitHub's user-friendly API and its community-focused nature made it the perfect choice for our project, giving us plenty of data to explore and understand.

1.2 Advantages of API over Web Scraping for GitHub

Using the GitHub API is better than web scraping for a few reasons. Firstly, API access is authorized by GitHub, ensuring compliance with platform policies. This minimizes the risk of violating terms of service. Additionally, the API makes it easier and faster to get the data we need compared to web scraping. Plus, it gives us more complete and up-to-date information because it can access the main database directly. Overall, the GitHub API is a better choice for getting user data on GitHub because it's easier, safer, and gives us better information.

2 Analytical Methodology

2.1 Data Collection:

Initially, we utilized the GitHub API to collect data on highly active users, specifically those with numerous repositories. Our approach involved identifying the top 10 users with the highest repository counts and examining their followers. Subsequently, we constructed a friendship network, where each GitHub user was represented by a node, and connections were drawn between users who followed each other. This method enabled us to gain insights into the connections among GitHub users, particularly those who actively share their work. Despite encountering challenges with the APIs of other social media platforms, GitHub's API proved to be user-friendly and provided us with the necessary data to construct our friendship network.

We encountered a significant volume of data during this process. However, due to the extensive nature of the dataset, the resulting network graph became overly complex, making interpretation challenging. Consequently, we made the decision to work with a subset of the collected data to facilitate clearer analysis and interpretation. Thus, we processed and cleaned the data using Excel before further analysis.

2.2 Data Visualization

In order to understand our friendship network more clearly, we utilized Python libraries like NetworkX and Matplotlib for visualization. NetworkX enabled us to create a visual representation illustrating the connections between users in our network. This visualization made it simpler to grasp the relationships between users. Furthermore, with Matplotlib, we generated graphs to display the distribution of connections among users. By leveraging these Python tools, we were able to effectively interpret our data and communicate our findings in a clear and professional manner.

2.3 Network Measures

In our analysis of the friendship network, various network measures were calculated to understand its characteristics. Using Python, we computed measures such as degree distribution, closeness centrality, page rank, and clustering coefficient. These calculations were conducted within a unimodel framework, where users served as nodes and the connections between them represented friendships denoted by edges.

By examining these measures, we gained insights into different aspects of the network, including the centrality of nodes, overall network connectivity, and clustering patterns. This provided us with a deeper understanding of how users interact within the network and the underlying structure of their relationships.

3 Results and Discussion

The analysis of the GitHub friendship network revealed several interesting insights. The PageRank values provided an indication of the importance or influence of individual users within the network. Users with higher PageRank values, such as "esin" and "matiasinsaurralde," likely hold significant influence within the GitHub community. Conversely, users with lower PageRank values may have less influence or activity within the network.

To better understand the network's structure, we examined the friendship graph, which visually represents how users are connected. This graph allows us to see the relationships between users and how they form communities or groups.

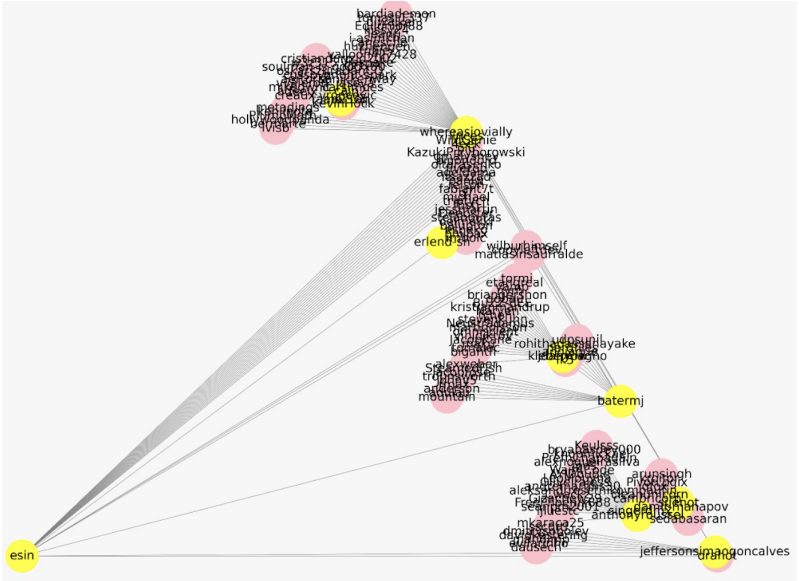

Figure 1: Network Graph

The average clustering coefficient of **0.0129** suggests that the GitHub friendship network exhibits a moderate level of clustering, indicating the presence of tightly-knit groups or communities within the larger network. This observation aligns with the typical behavior seen in social networks, where users tend to form connections with others who share similar interests or work on related projects.

Similarly, the degree distribution graph provides insights into how many connections each user has. This graph helps us understand the distribution of connections across the network and identify users with a large number of connections, often indicating high activity or influence.

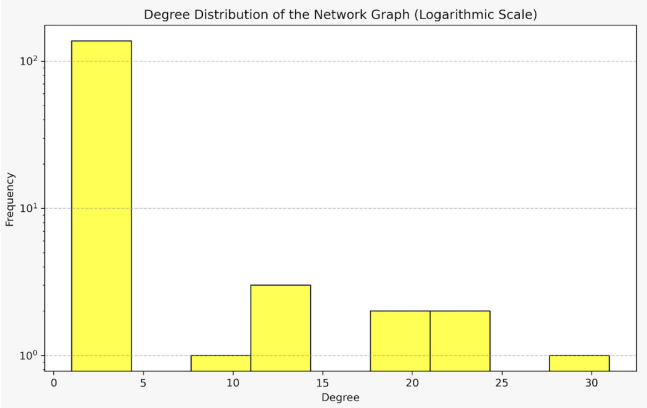

Figure 2: Degree Distribution of the Network Graph

Moving on to closeness centrality, we found that the average value of **0.0091** indicates that users in the GitHub network are, on average, relatively close to one another in terms of their network connections. This implies a high level of interconnectedness and accessibility within the network, allowing for efficient communication and collaboration among users.

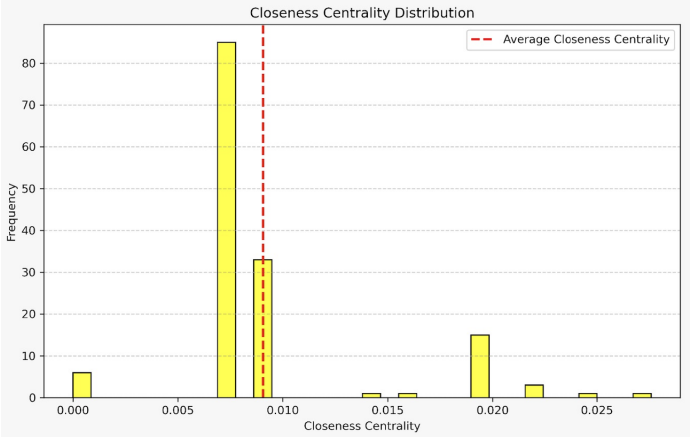

Figure 3: Closeness Centrality

Lastly, we analyzed the PageRank graph, which highlights the influence levels of individual users in the network. This graph visually represents the importance of each user based on their connections and interactions within the GitHub community. Furthermore, the average PageRank value of the network was calculated to be **0.0068493150684931555**, providing an overall measure of the network's structure

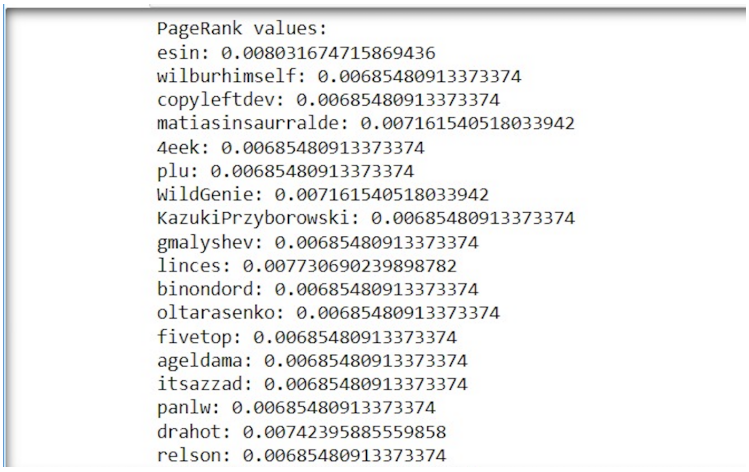

Figure 4: Page Rank Values

Overall, the results suggest that the GitHub friendship network is characterized by a diverse range of users with varying levels of influence and connectivity. The presence of tightly-knit communities and high interconnectedness highlights the collaborative nature of the GitHub platform, where users come together to share knowledge, collaborate on projects, and contribute to open-source development.

4 Conclusions

Through significant obstacles and thorough investigation, our study of the GitHub friendship network has produced interesting results. PageRank analysis identified important network influencers, while network metrics such as clustering coefficient and closeness centrality demonstrated the integrated and interconnected structure of the network.

Through determination regardless of challenges like technological difficulties and limitations on the APIs, we were able to make efficient use of the GitHub API and conduct thorough data collecting and analysis. We have experienced how network analysis can be useful in understanding the complex structure of digital ecosystems on our journey.

To sum up, this study contributes to a better understanding of the social dynamics underlying online collaborative platforms like GitHub and underscores the importance of network analysis in uncovering hidden insights within complex social networks.

5 Challenges

5.1 API Limitations

One of the primary challenges encountered during the project was the limitations imposed by the public APIs of various social media platforms, such as Instagram and Twitch. Despite efforts to collect user data for analysis, these platforms restricted access to user information, hindering the data collection process.

5.2 Insufficient Insights from YouTube Data

While attempting to analyze data from YouTube, it was observed that the obtained graph did not provide significant insights or valuable information for analysis. Despite successfully fetching data related to channels using similar hashtags, the resulting graph lacked depth and meaningful patterns, prompting the decision to discontinue the exploration of YouTube data.

5.3 Data Interpretation Issues

Although data collection and visualization were carried out effectively using tools like NetworkX and Matplotlib, there were challenges in interpreting the generated graphs and deriving meaningful insights from them. Understanding the complex network structures and identifying significant patterns within the friendship networks posed difficulties during the analysis process. Additionally, attempts to utilize Gephi for data interpretation were hampered by compatibility issues, as the application did not function properly with Mac operating systems

6 References

1. **Managing your personal access tokens**
   https://docs.github.com/en/authentication/keeping-your-account-and-data-secure/managing-your-personal-access-tokens#: :text=been%20verified%20yet.-,In%20the%20upper%2Dright%20corner%20of%20any%20page%2C%20click%20your,Generate%20new%20token%20(classic)

2. **How to use GitHub API to extract data with Python?**
   https://melaniesoek0120.medium.com/how-to-use-github-api-to-extract-data-with-python-bdc61106a501

3. **Create GitHub API to fetch user profile image and number of repositories using Python and Flask**
   https://www.geeksforgeeks.org/create-github-api-to-fetch-user-profile-image-and-number-of-repositories-using-python-and-flask/

4. **Use the REST API to get information about followers of authenticated users.**
   https://docs.github.com/en/rest/users/followers?apiVersion=2022-11-28