CS579 - Project 1 - Report

Sarvesh Shroff | Akshay Jain

A20488681 | A20502846

Department of Computer Science

Illinois Institute of Technology

February 14, 2022

Abstract

In this project, we were asked to crawl social media data and visualize it using graphs. After which we needed to apply different analysis to the extracted data. We used Twitter as the social media to get the required data.

Project Objective

Creating a user's friendship network that can be represented as a graph where the nodes are the users and the edges show whether there is a friendship relation between them. Example: Users and their follower and followee relationship in Twitter as a directed graph. Implement different types of analysis such as Degree Distribution, Clustering Coefficient, Betweenness, Closeness etc.

Project Outline

1. Create Twitter developers account (*Twitter Developer*, n.d.)
2. Data Collection
3. Data Visualization
4. Network Measures Calculation

Proposed Solution

A. To create Twitter developers account:
    a. Apply for Twitter Developers account and obtain credentials:
        i. Visit https://developer.twitter.com (*Twitter Developer*, n.d.)
        ii. Log in to your Twitter account or Sign up for a new one
        iii. On the top right hand corner click “Apply”. Then click “Apply for a developer account”
        iv. For your primary reason for using Twitter developer tools choose “Student”. Then click “Next”
        v. Add a valid contact and country details to use the Twitter API
        vi. Complete the form on how you intend to use your Twitter Developer Account

B. Data Collection
    a. Install and Import required libraries
        i. Numpy (*NumPy.org*, n.d.)
        ii. Pandas (*Pandas*, n.d.)
        iii. Tweepy (*Tweepy*, n.d.)
        iv. Networkx (*NetworkX*, n.d.)
        v. Matplotlib (*Matplotlib Documentation — Matplotlib 3.5.1 Documentation*, n.d.)
        vi. JSON (*Json — JSON Encoder and Decoder — Python 3.10.2 Documentation*, n.d.)
    b. We used Twitter handle “@sarvesh_shroff” for getting the data
    c. Get the user object for an input screen_name using

        API.get_user(id/user_id/screen_name):

        ```
        user = api.get_user(screen_name='sarvesh_shroff')
        ```

    d. Get the list of followings/friends for an input username using

        API.friends_ids(id/user_id/screen_name):

        ```
        friends = api.friends_ids(user_id)
        ```

    e. Using ‘double for loop’ map the friends list to create a pair of node for an edge

f. We got 188 nodes and 2455 edges for the directed graph and stored the information in node.csv and edge.csv respectively
g. We passed node.csv & edge.csv as an input to Gephi and Networkx

C. Data Visualization

In this project we are using 2 platforms for data visualization:

a. Gephi - The Open Graph Viz Platform (*Gephi - The Open Graph Viz Platform*, n.d.)

Gephi is the leading visualization and exploration software for all kinds of graphs and networks. Gephi is open-source and free.

b. Networkx (*NetworkX*, n.d.)

NetworkX is a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks.

D. Network Measures Calculation

After visualization with Gephi we used different types of network measures

Implementation details

Some of the problems and design issues that I faced are mentioned below:

- The first problem that we face was Twitter API as it always ran into Rate limit exceeded for which we used the wait at rate-limit parameter in tweepy API

- While visualizing the graph in Networkx we observed it worked well for low data nodes but when we applied it for larger data it made the nodes overlap and we were not able to see the edges properly. Due to this, we shifted to Gephi which had better visualization when given bigger data. (*Compare Gephi Vs Networkx | DiscoverSdk*, n.d.)

- As we were new to Gephi we had to install its prerequisite of Java 1.8, initially, we had a different version due to which Gephi did not work properly so we reinstalled java with the proper version

- For implementing Networkx we used an adjacent matrix for plotting but when it came to Gephi we had to change it to defining Source and Target node as Gephi needs this as input for visualization.

Results and Discussions

- Graph (figure 1) using Networkx for large nodes and edges we obtained following graph which was not a good visualization

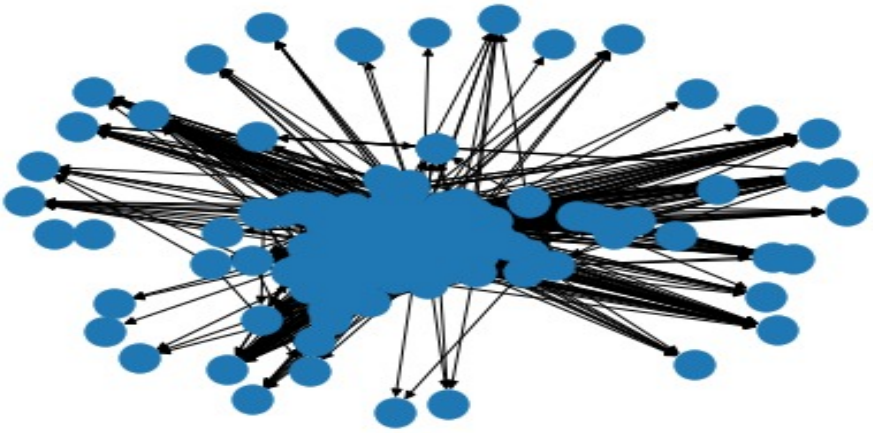


figure 1: connected nodes

- Graph (figure 2) using Gephi for large nodes and edges

  The following graph shows all the nodes in the network

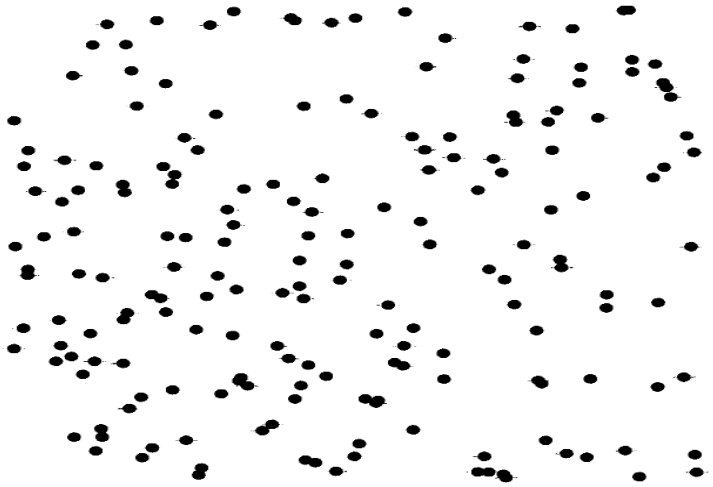


figure 2: Nodes

- Figure 3 shows all the connected edges of the graph in Gephi

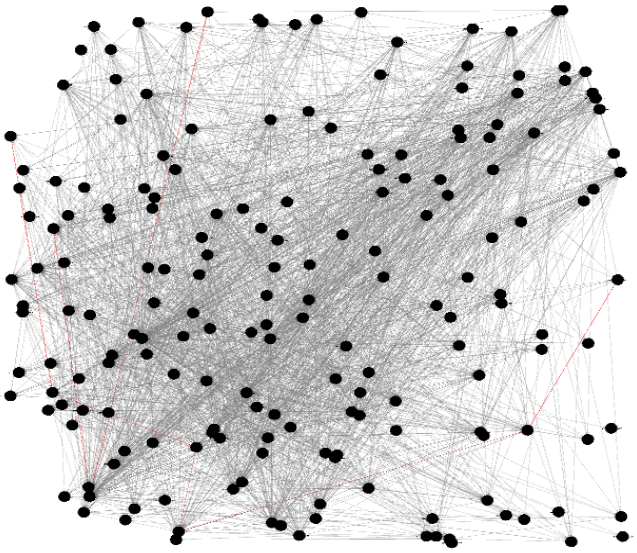


figure 3 : connected nodes

- Figure 4 shows all the followers in the yellow node who follow the red node

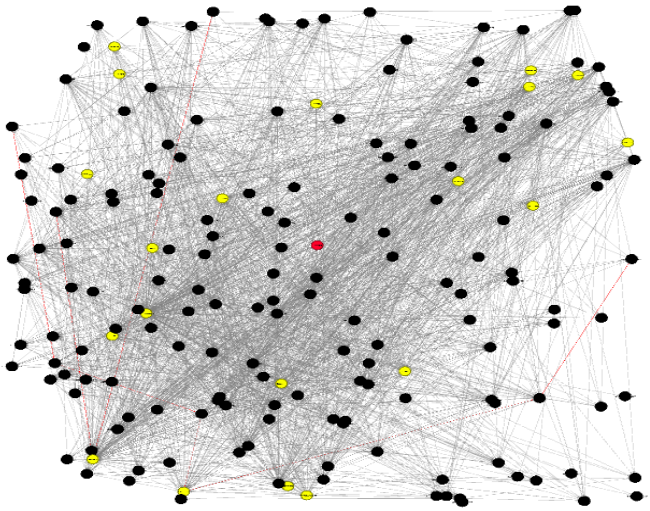
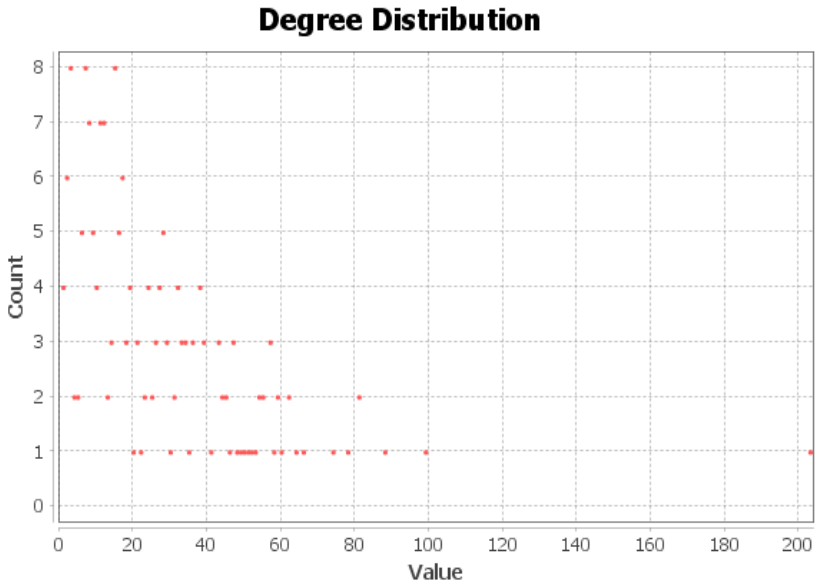


Figure 4

Network Measure Analysis

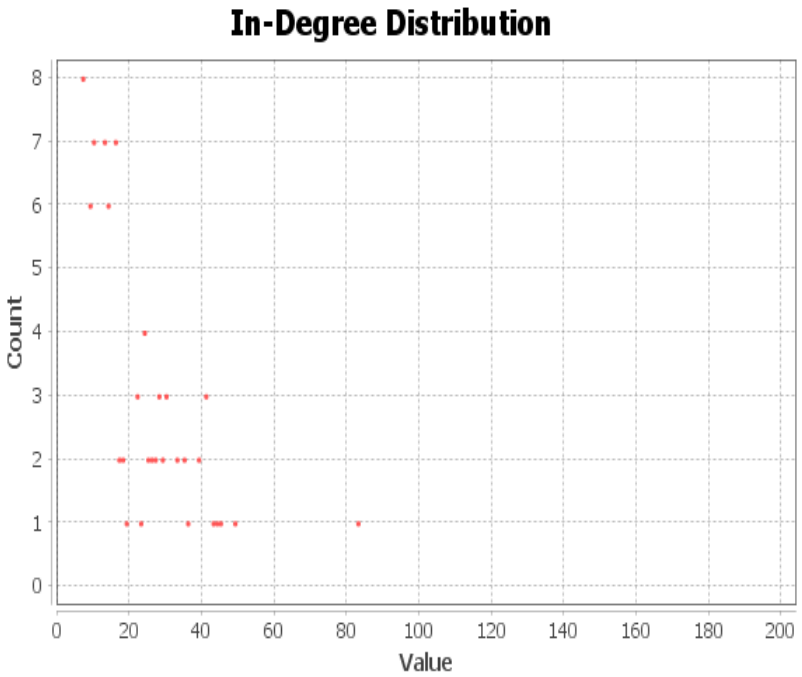
Degree Distribution:

The degree of a node in a network is the number of connections it has to other nodes. The average degree came out to be 13.059



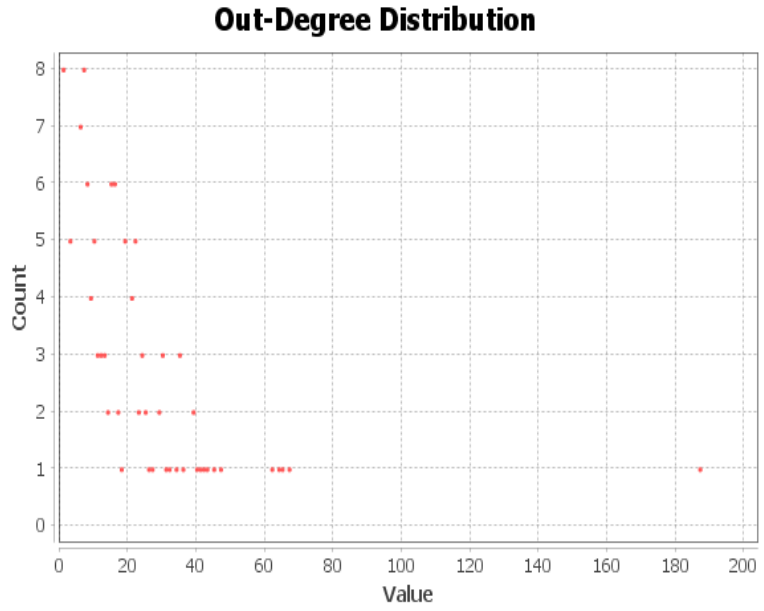
In Degree Distribution:

The in-degree of a node in a network is the number of incoming connections it from other nodes



Out Degree Distribution:

The out-degree of a node in a network is the number of outgoing connections it has to other nodes



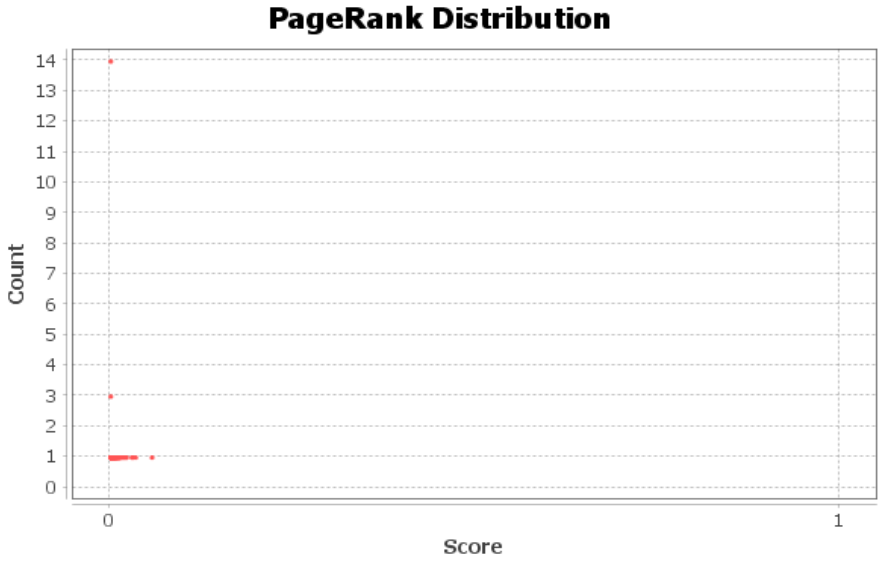
Page Rank Distribution:

PageRank works by counting the number and quality of edges to a node to determine a rough estimate of how important the node is.

Parameters:

Epsilon = 0.001

Probability = 0.85



Eigen Vector Centrality Distribution:

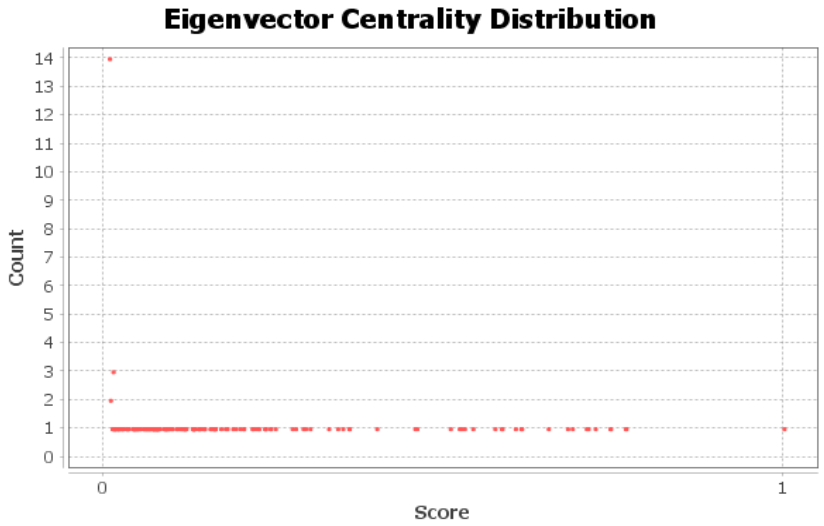
It is a measure of the influence of a node in a network.

Parameters:

Network Interpretation: directed

Number of iterations: 100

Sum change: 0.02022358330946175

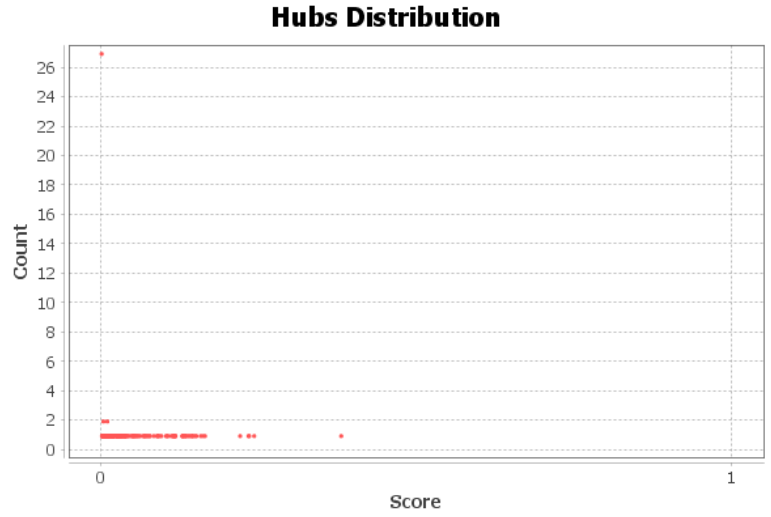


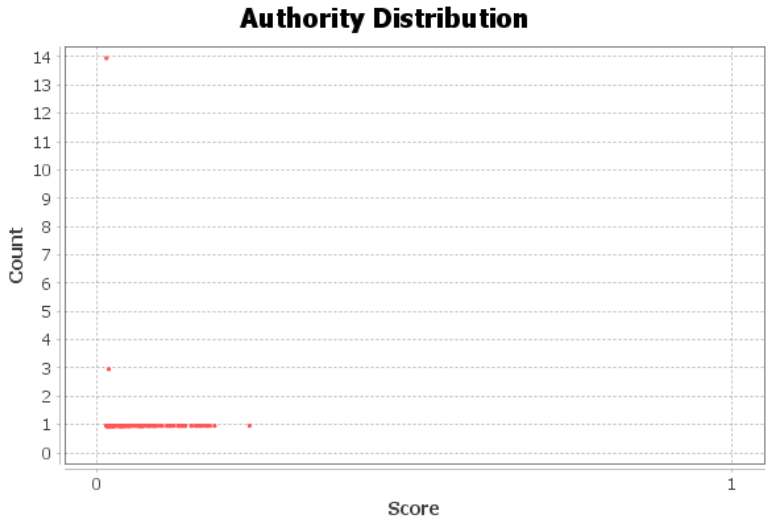
HITS Metric Report:

Hyperlink Induced Topic Search is an algorithm used in link analysis.

Parameters:

E = 1.0E-4





Team Efforts

We worked on the project together so the below table is just a rough estimate that we could come up with as we always sat together and worked on the project

Task	Akshay Jain	Sarvesh Shroff
Create Twitter developers account	60	40
Data Collection	40	60
Data Visualization	50	50
Network Measures Calculation	50	50

References

Compare gephi vs networkx | DiscoverSdk. (n.d.). DiscoverSDK. Retrieved February 14, 2022, from http://www.discoversdk.com/compare/gephi-vs-networkx

Gephi - The Open Graph Viz Platform. (n.d.). Gephi - The Open Graph Viz Platform. Retrieved February 14, 2022, from https://gephi.org/

json — JSON encoder and decoder — Python 3.10.2 documentation. (n.d.). Python Docs. Retrieved February 14, 2022, from https://docs.python.org/3/library/json.html

Matplotlib documentation — Matplotlib 3.5.1 documentation. (n.d.). Matplotlib. Retrieved February 14, 2022, from https://matplotlib.org/stable/index.html#

NetworkX. (n.d.). NetworkX — NetworkX documentation. Retrieved February 14, 2022, from https://networkx.org/

NumPy.org. (n.d.). NumPy.org. Retrieved February 14, 2022, from https://numpy.org/

Pandas. (n.d.). pandas - Python Data Analysis Library. Retrieved February 14, 2022, from https://pandas.pydata.org/

Tweepy. (n.d.). Tweepy. Retrieved February 14, 2022, from https://www.tweepy.org/

Twitter Developer. (n.d.). Use Cases, Tutorials, & Documentation. Retrieved February 14, 2022, from https://developer.twitter.com/en