CS 579: ONLINE SOCIAL NETWORK ANALYSIS

PROJECT - 1

TEAM MEMBERS:

1. **Hemanth Vennelakanti (A20526563)**
2. **Kevan Dedania (A20522659)**

1. Setup, Challenges & Policies

How we crawled the data

- We have used the **Python Reddit API Wrapper** (PRAW) to crawl the data.

Challenges faced & How we overcame them

- First we decided to go with Twitter (X) and Instagram however with Twitter there was a rate limiter for read requests and thought this would limit my development hence moved to Instagram. Just setting up the Meta developer account took a long time however I had to follow certain steps (make my current account as Creator / Business account first and then have to create a page for our project in facebook, somehow even after doing all the steps it was very difficult to navigate through other steps.

- **Reddit:** Difficulty to setup the app as we have to set up the new app in preferences sections and there's no straightforward way to navigate to that page. I had to do a repeated search on google to find out the URL which leads to that page.

- **Reddit:** Unable to find a proper way of connecting the different nodes. Therefore we skipped the top-level comments and those without an author and recorded only those interactions where a user replies to another user comment on various posts.

How challenges impacted the data that we collect

- Earlier as said, we were not able to set up the edges between the users as we didn't ensure the edge cases coverage. By skipping top-level comments and those comments without an author kind of resolved our issue.

User Privacy Policy:

- **https://www.reddit.com/policies/privacy-policy#:~:text=We%20do%20not%20use%20or,Choices%22%20section%20of%20this%20notice.**

Data Usage Policy:

- **https://www.redditinc.com/policies/content-policy**

2. Data Cleansing & Transformations:

Skipping Comments without Authors

- If a comment has no author (indicative of deleted users or anonymous postings), it is skipped. This is important because interactions involving anonymous or non-existent users do not contribute meaningfully to the network analysis

Avoiding Self Interactions

- We didn't count the interactions where a user replies to their own comments are not counted.

Transformations

- Structuring the data into a uniform interaction key format that represents the direction of interaction.

- Each interaction is formatted as a string **"{parent.author.name} -> {comment.author.name}"**, which standardizes the representation of interactions. This uniformity is crucial for the next steps of analysis, ensuring that all data points follow the same structure for easy processing and interpretation.

3. Graph Analysis:

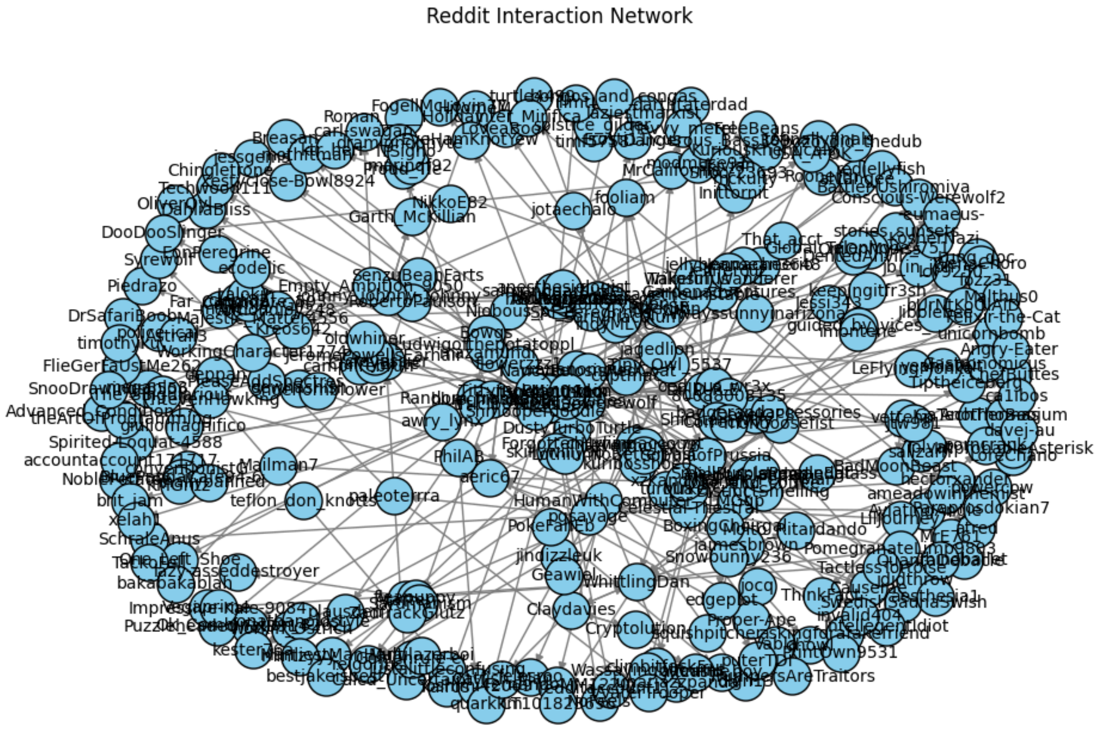
Software Used & Why we choose it

- Software used: **NetworkX**
- NetworkX is a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks.
- Documentation is quite clear and also it has the built in support for most of the network measures particularly that are asked.

Format of the Input Graph Data:

- **Edge List:** For constructing the graph
- Based on the interactions we added edges to the graph.
- An edge list represents each connection between two nodes (users) as a single entry and typically in the form (source, target)

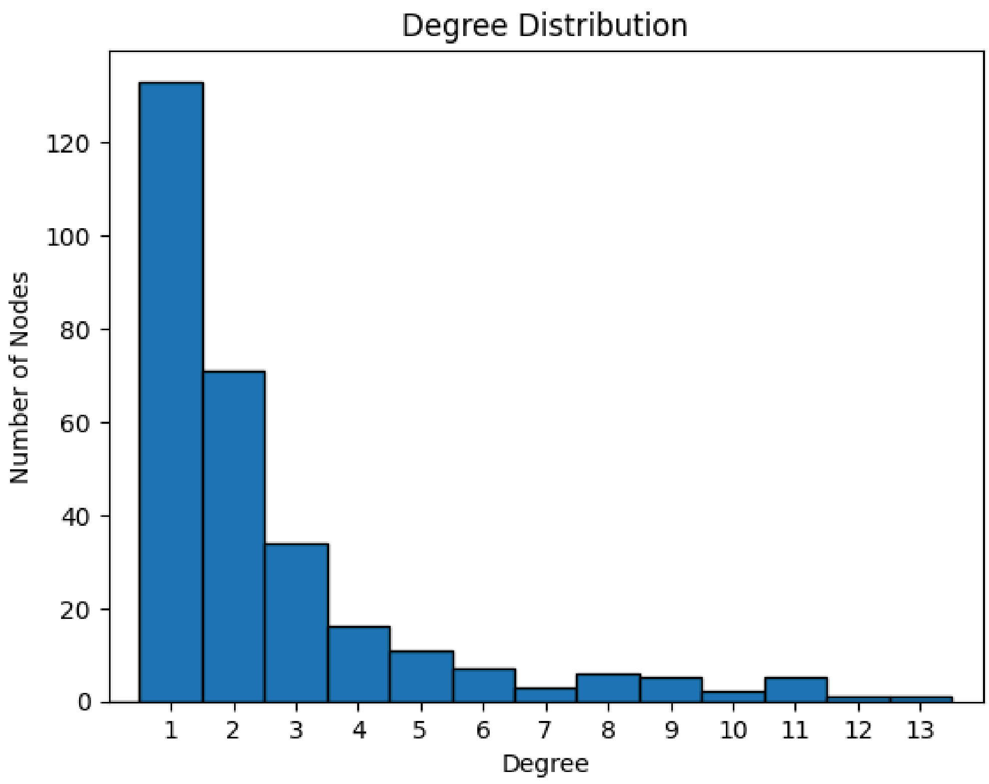
Visualization & How to Interpret



- **Nodes**: Each node in the graph represents a unique user.
- **Edges**: Each edge represents a directed interaction from one user to another. If user A comments on user B's comment, a directed edge is created from B to A.

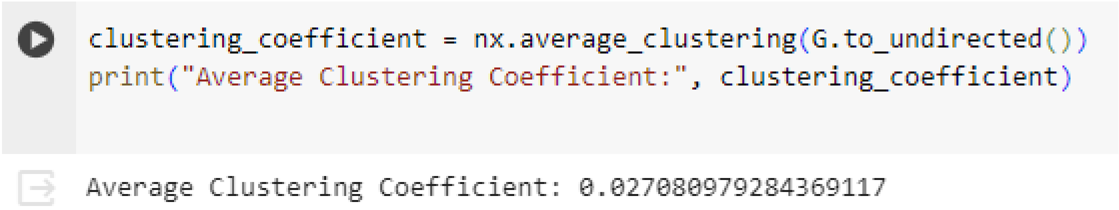
4. Network Measures

Degree distribution histogram



- **Most Users Have Few Connections:** The highest bar is at degree 1, which means that the largest number of users have interacted with only one other user.
- **Fewer Users Have More Connections:** As the degree increases, the number of users with that many connections decreases. This suggests that fewer users are highly interactive & engaging with many other users.
- **How we plotted:** The histogram is plotted by counting the frequency of each unique degree (number of connections) present among the nodes in the graph, and then displaying these counts as bars, with the degree values on the x-axis and the number of nodes on the y-axis

Clustering Coefficient:

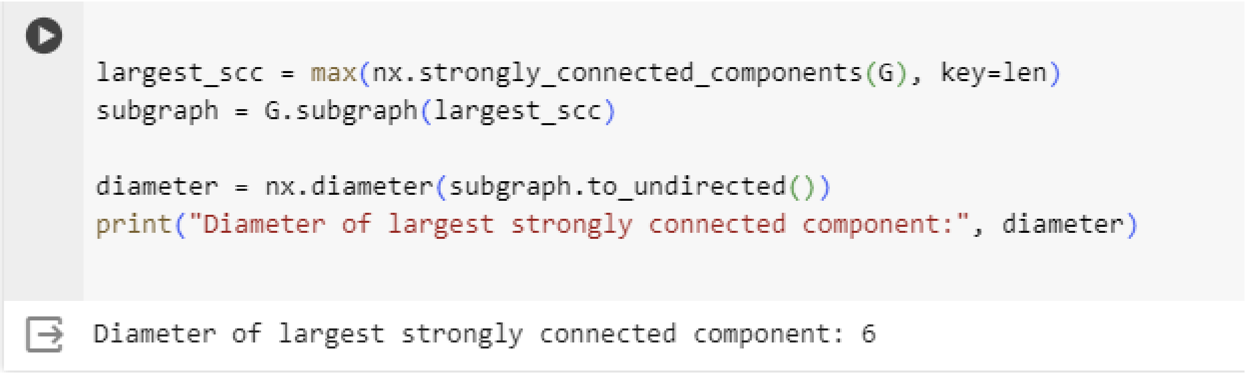


```
clustering_coefficient = nx.average_clustering(G.to_undirected())
print("Average Clustering Coefficient:", clustering_coefficient)
```

```
Average Clustering Coefficient: 0.027080979284369117
```

- **Average Clustering Coefficient:** 0.027080979284369117
- This suggests that, on average, the likelihood of a user's connections interacting with each other is low.
- We used built in method (average_clustering) in NetworkX

Diameter:



```
largest_scc = max(nx.strongly_connected_components(G), key=len)
subgraph = G.subgraph(largest_scc)

diameter = nx.diameter(subgraph.to_undirected())
print("Diameter of largest strongly connected component:", diameter)
```

```
Diameter of largest strongly connected component: 6
```

- The greatest distance from one node to another (measured by the shortest path that connects them) is **6 edges**.
- We used built in method (diameter) in NetworkX

5. Further questions & Investigations:

What further questions do these results raise

- Why do most users only interact once?
- Low clustering coefficient indicates lack of broader interaction on posts. Is this common among several subreddits ?

Our next step to investigate further be

- Investigate across different subreddits to identify the common pattern or most of the interactions are just acknowledgements.
- Investigate patterns of interactions to see if clustering changes during specific events.

6. References:

[1] https://networkx.org/

[2] https://github.com/praw-dev/praw

[3] https://praw.readthedocs.io/en/stable/