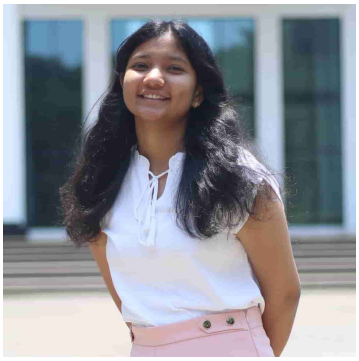
Detecting Most Influential Nodes in a Network

We are analysing a dataset and finding the most influential person using different techniques and also coming to a conclusion, as of why there is formation of communities and how the graph is made.

Team #1: Detecting Most Influential Nodes in a Network

Team Members:

Keshav Garg
SE20UCSE065



Aayushi Singh
SE20UCSE004

Maitreya Guduru
SE20UCSE083

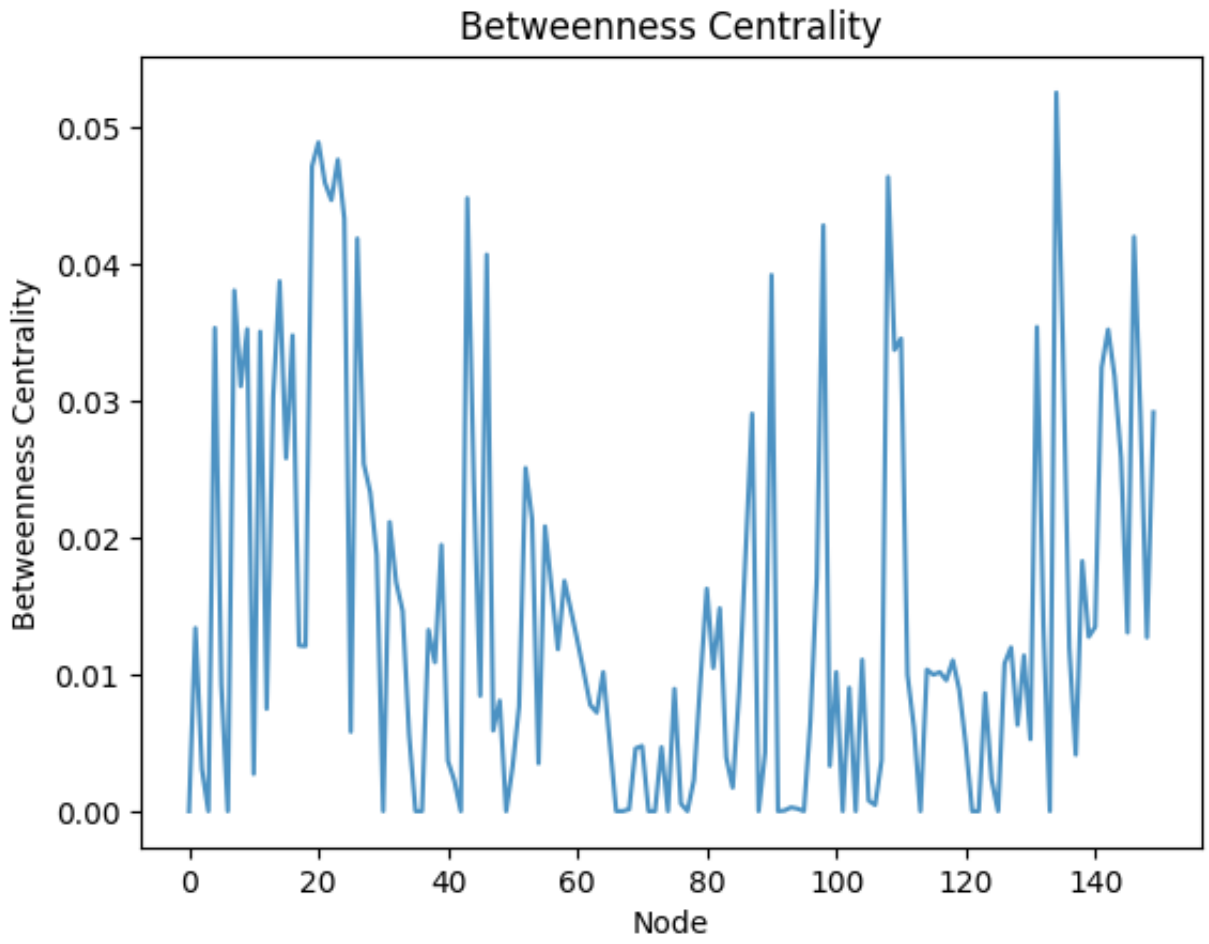
Mallika Asthana
SE20UCSE086

About our Project

We are analyzing the Facebook dataset and finding the most influential node using different techniques like:

1. Degree Centrality
2. Eigenvector Centrality
3. Betweenness Centrality
4. Association(Clustering)
5. Page Rank
6. and different analysis techniques like:
    1. Density
    2. Transitivity
    3. Average Shortest Path

Facebook Network:



Network build using 4000+ Facebook users.

About the Facebook Network

1. We can say its a real-world network
2. Contains different communities, which are easily visible.
3. Contains Giant components.
4. It is an undirected graph.
5. Density is 0.01082 or 1.082%
6. Transitivity is nearly 52%
7. Also, it is a small world as the average path length is nearly 4

Code File

We used NetworkX library in Python to analyze the dataset and build graphs.

Dataset: Facebook

Code file: Colab Notebook

Steps to access the code and see the analysis:

1. Download the dataset from the above-given link.
2. Go to the code file using the collab link given.
3. Then upload the dataset in collab and execute all the cells to see results.

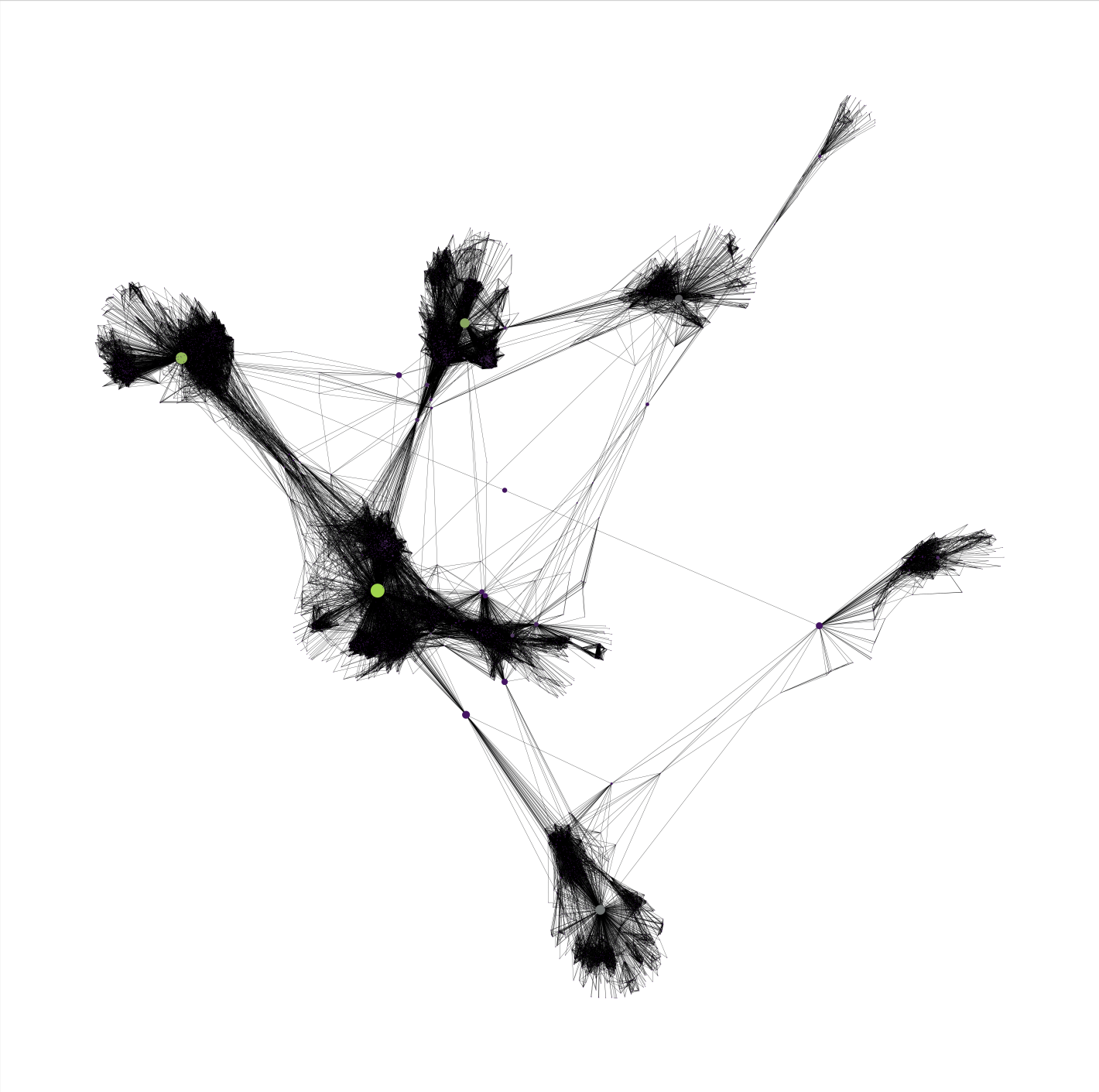
OR

Access the pdf file of the code to see the output: Analysis

Analysis of the Facebook network: Centrality measures & other metrics

We used different metrices for our analysis:

- **Degree Centrality**: We have found out that node number 107 has the highest degree of centrality with 0.258791480931154.
- **Eigen Vector Centrality**: Eigenvector Centrality is an algorithm that measures the transitive influence of nodes. Relationships originating from high-scoring nodes contribute more to the score of a node than connections from low-scoring nodes. A high eigenvector score means that **a node is connected to many nodes** with **high scores**. In our network node, 1912 has the highest Eigen Vector centrality of 0.09540696149067629.
- **Betweenness Centrality:** This measure shows which nodes are 'bridges' between nodes in a network. It does this by identifying all the shortest paths and counting how many times each node falls on one. In our network node, 107 has the highest betweenness of 0.4805180785560152.
- **Closeness Centrality:** Closeness centrality is a useful measure that estimates how fast the flow of information would be through a given node to other nodes. In our model, we have node 107 which is having the highest centrality of 0.45969945355191255.
- Using the PageRank algorithm we saw that node 3437 stood first in the race.
- Using Louvain community detection, we have found that there is a total of 15 communities in our network.
- Using betweenness centrality measures we have found the top 5 nodes in each cluster that plays significant roles in spreading information and connection from one node to another [107, 1684, 3437, 1912, 1085]. Below is the figure:

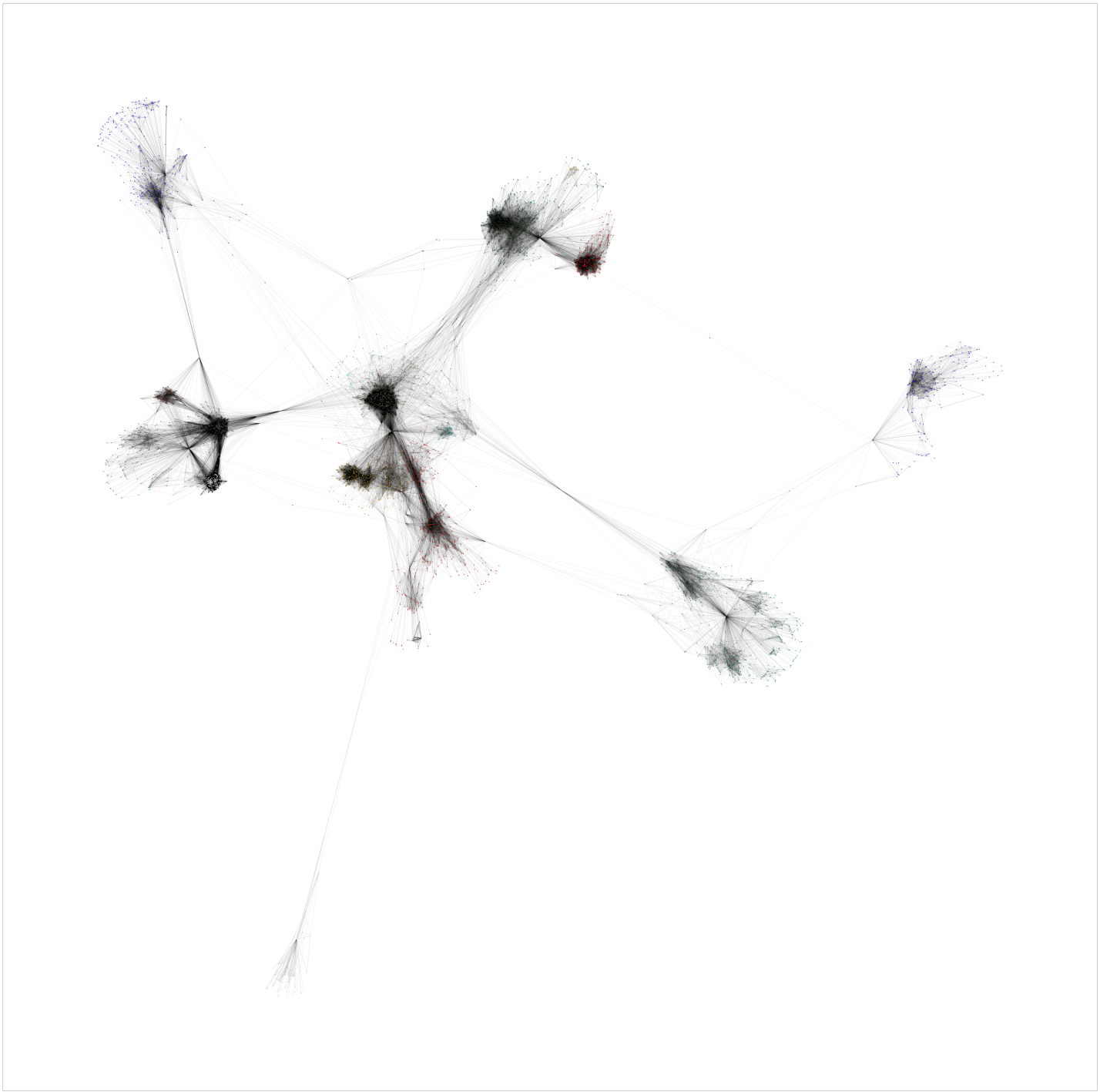


Different Communities in the Facebook Network

Community Detection

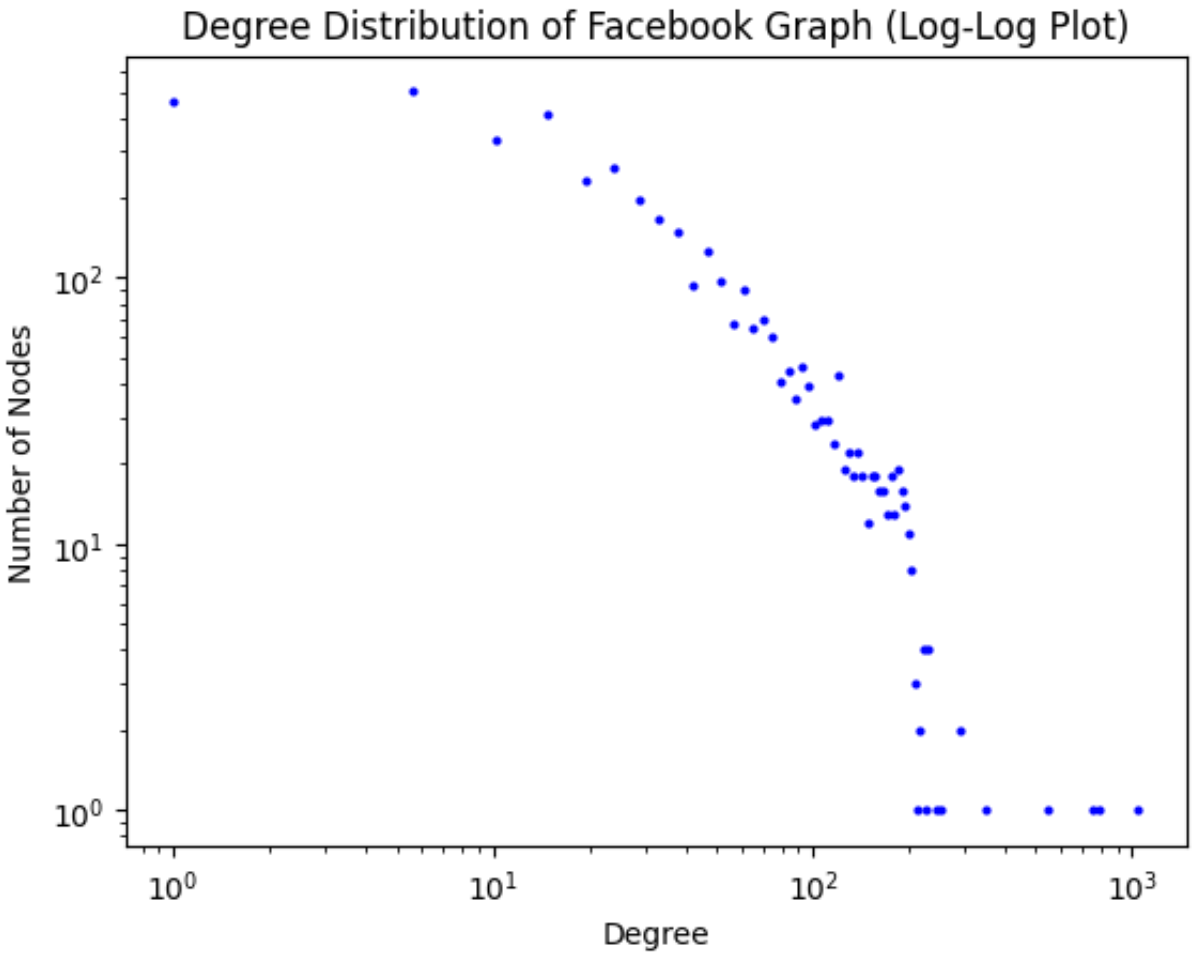
Total communities

Since we have found that there are a total of 15 communities in our network, here is a visual representation of it:



Log-Log plot for Facebook network

Since our network contains thousands and more nodes and edges, we decide to analyze the log plot for the same. The plot comes out to be an exponentially decreasing plot. Below is the image:



Phase 2 of the project

The Modularity of Our Network

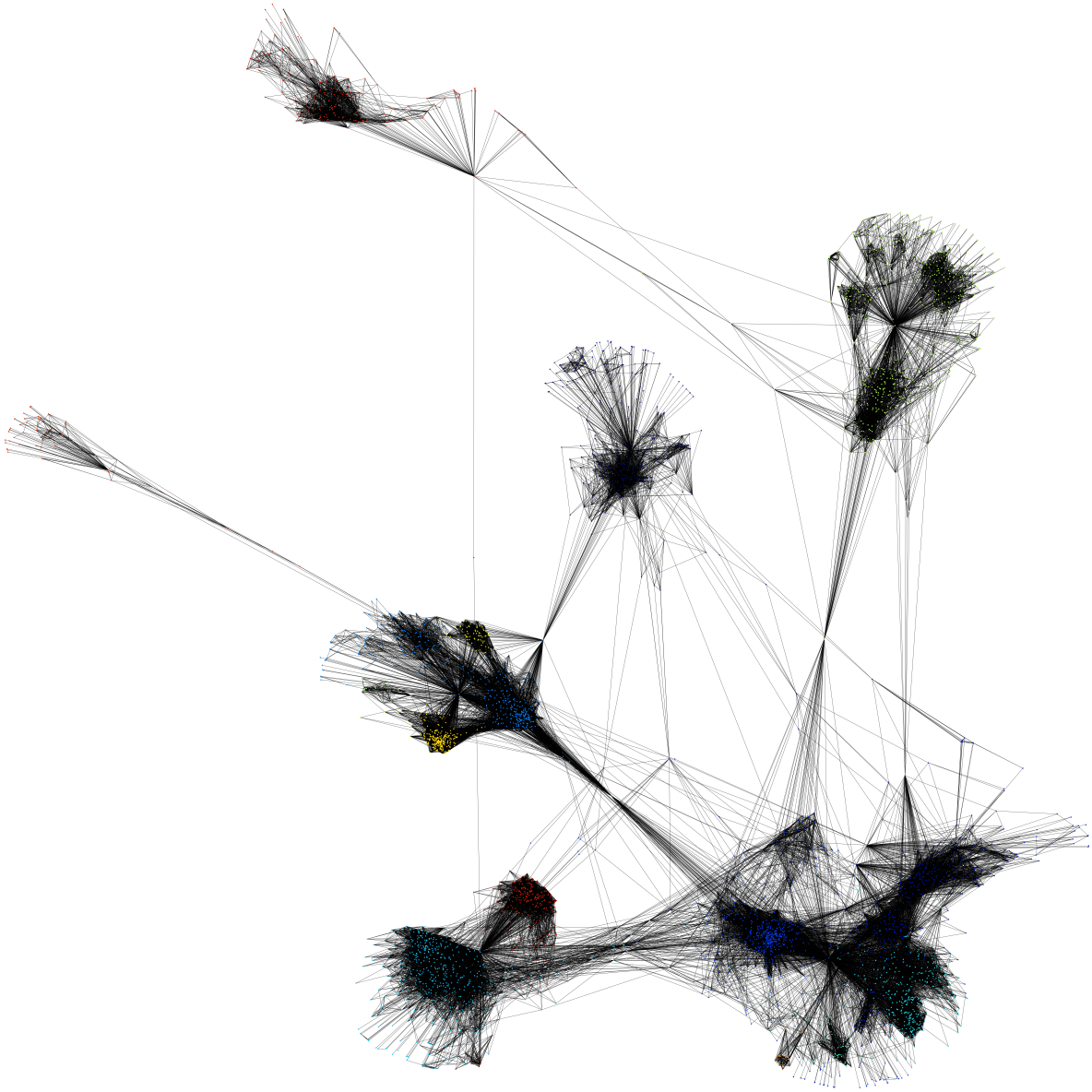
Modularity in social networks indicates the distinct groups within the network.

A high modularity value in a social network indicates that the network is organized into cohesive subgroups. Nodes within the same subgroup have a higher density of connection among themselves compared to nodes in different subgroups.

The modularity of our Facebook data is :

0.7368407345348218 ~ 74 %

Which implies that the network is highly modular/strong community structure.



Modular Network

Assortativity

Assortativity is a measure of the tendency of nodes in a network to be connected to other nodes that are similar in some way.

For this network, we have calculated assortativity in such a way that a positive assortativity coefficient indicates that nodes tend to be connected to other nodes with similar degrees, while a negative coefficient indicates the opposite.

Node with maximum assortativity: 1954 (0.34647347040623705)

Node with minimum assortativity: 860 (-1.0)

Overall assortavity of the network: 0.06357722918564943

An Assortativity coefficient of -1.0 indicates that nodes tend to be connected to other nodes with dissimilar degrees. In other words, nodes with high degrees tend to be connected to nodes with low degrees, and vice versa. This is known as disassortative mixing.

The degree assortativity coefficient for the entire network is positive (0.06357722918564943), indicating a weak assortative mixing pattern at the global level. This means that, on average, nodes with similar degrees are more likely to be connected to each other compared to what would be expected by chance. However, **at the individual node level, most of the assortativities are negative.** This suggests a **disassortative mixing pattern for individual nodes.**

Google Matrix

Since, we studied a concept of Google Matrix in class, so we tried to implement the same for our dataset. The matrix tells how the nodes are connected to each other and what is there importance. And, to calculate the importance of each node in sometime in future, then we can use this matrix.

Google Matrix:

The matrix is calculated by first creating an adjacency matrix, which is a matrix that shows which nodes are connected to each other. The Google matrix is then calculated by multiplying the adjacency matrix by a damping factor and a matrix of ones. The damping factor is a number that controls how much weight is given to the incoming links to a node. The matrix of ones is used to ensure that each node has a PageRank of at least 1. The Google matrix is then normalized so that the sum of each row is 1. The rows of the Google matrix can then be used to rank the importance of the nodes in the network.

Here is the matrix:

```
array([[2.47586036e-04, 2.47586036e-04, 2.47586036e-04, ...,
        2.47586036e-04, 2.47586036e-04, 2.47586036e-04],
       [9.98599325e-01, 3.46873464e-07, 3.46873464e-07, ...,
        3.46873464e-07, 3.46873464e-07, 3.46873464e-07],
       [9.98599325e-01, 3.46873464e-07, 3.46873464e-07, ...,
        3.46873464e-07, 3.46873464e-07, 3.46873464e-07],
       ...,
       [1.73558312e-07, 1.73558312e-07, 1.73558312e-07, ...,
        1.73558312e-07, 1.73558312e-07, 1.73558312e-07],
       [8.68095826e-08, 8.68095826e-08, 8.68095826e-08, ...,
        8.68095826e-08, 8.68095826e-08, 8.68095826e-08],
       [3.85895536e-08, 3.85895536e-08, 3.85895536e-08, ...,
        3.85895536e-08, 3.85895536e-08, 3.85895536e-08]])
```

Text file to see the full matrix: Google Matrix

Additional Information related to our project

- Since It is a real-world network it satisfies the properties like low density, shortest average path, and exponentially decreasing plot.
- As the size of the network increases the number of communities will also increase mostly but in the rarest of cases, it may decrease.
- The network is highly modular.
- 107 node is the most important node.
- The mathematical formulation of centrality measures: Pdf

Conclusion

-Here is the table for the Centrality measures →

Centrality Measures	Node Number	Value
Degree	107	0.258
Betweenness	107	0.480
Closeness	107	0.459
Eigen Vector	1912	0.095

Analysis techniques:

Density	1.082%
Transitivity	52%
Average shortest path	4

Modularity	74%
Overall Assortativity	0.06357

Content is available under the Creative Commons Attribution License, by SocialWiki. | Powered by Wiki.js