Facebook pages: TV shows

Charlene Khun and Tiantong Li

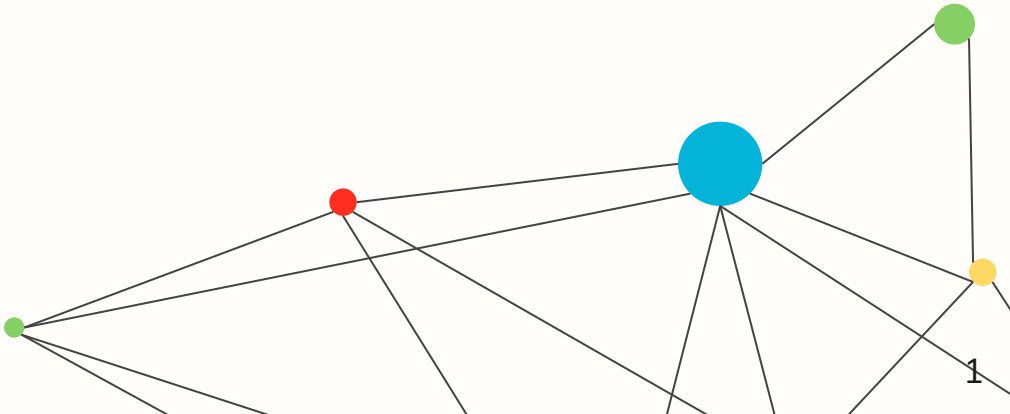


Table of contents

01 Introduction

- In our project, we are analyzing the relationships of tv shows collected from Facebook pages.
- We got our dataset from networkrepository.com under the social network category
- The data was collected on November 2017
- There are 3892 nodes, and 17262 edges
- The nodes represent the individual facebook tv show pages while the undirected edges represent the mutual likes among the TV show Facebook pages.
- The graph is undirected

Screenshot of Dataset

```
1   0,1838
2   0,1744
3   0,14
4   0,2543
5   1,1009
6   1,1171
7   1,1465
8   1,2069
9   1,2080
10  1,1856
11  1,3799
12  1,1033
13  1,2717
14  1,300
15  1,1603
16  1,942
17  1,3678
18  1,952
19  1,815
20  2,3656
21  2,3459
```

```
17246   3868,2298
17247   3762,3827
17248   3767,3816
17249   3775,3823
17250   3777,3777
17251   3783,3788
17252   3786,3867
17253   891,3863
17254   3798,3806
17255   3741,3808
17256   3813,3854
17257   486,3867
17258   3826,3844
17259   3830,3843
17260   1240,1240
17261   3876,3885
17262   3879,3886
17263
```

02 Problem Description

- Social media platforms like Facebook have become significant sources of data for understanding user preferences and behavior. These Facebook pages accumulate vast amounts of data regarding user interactions.
- In our dataset, the interactions are mutual likes. Analyzing these interactions can provide insights into audience preferences and interests.
- We want to find which TV shows are more popular
- Study connections to find the communities in the dataset

03 Graph Analysis Steps

- Step 1: Parsing the data
- Step 2: Obtaining basic graph information (number of nodes and edges)
- Step 3: Finding the number of connected components
- Step 4: Finding the diameter
- Step 5: Finding the top 5 nodes with highest clustering coefficients
- Step 6: Finding the top 5 nodes with highest betweenness centrality
- Step 7: Applying Community Detection Algorithms (Girvan Newman and Louvain)
- Step 8: Visualizing dataset

Parsing the Data

- We used the built-in method read_edgelist method from NetworkX.

```
1   0,1838
2   0,1744
3   0,14
4   0,2543
5   1,1009
6   1,1171
7   1,1465
8   1,2069
9   1,2080
10  1,1856
11  1,3799
12  1,1033
13  1,2717
14  1,300
15  1,1603
16  1,942
17  1,3678
18  1,952
19  1,815
20  2,3656
21  2,3459
```

Obtaining Basic Graph Information

- We created a function get_graph_info to display the number of nodes, number of edges, and a list of the individual nodes and edges.
- All methods inside the function are built-in from NetworkX.
- We found the total number of nodes is 3892 and the total number of edges is 17262

```
Number of nodes: 3892
Number of edges: 17262
Available nodes: ['0', '1838', '1744', '14', '2543', '1', '1009', '1171', '1465', '2069', '2080', '1856', '3799', '1033',
Available edges: [('0', '1838'), ('0', '1744'), ('0', '14'), ('0', '2543'), ('1838', '2013'), ('1838', '2714'), ('1744', '
```

Number of Connected Components

- To obtain the number of connected components, we used a built-in method from Network X and found that the number of connected components is 1.
- This makes sense because all the TV shows are connected together and no nodes are isolated.
- Thus, there is a single giant component.

Diameter

- The diameter is defined as the “maximum distance between any pair of vertices.”
- It is also known as the longest shortest path.
- To find the diameter, we just called the built-in diameter function from NetworkX.
- We found that the diameter for our dataset is 20.
- This means that the nodes are at most 20 hops away from each other

Top 5 Clustering Coefficients

- We used a built-in method from NetworkX
- We then used a sorting method (in reverse) to sort the clustering coefficients from highest to lowest. We extract only the top 5.
- The top 5 clustering coefficients are all 1.0.

```
Node 2972: Clustering Coefficient = 1.0
Node 3283: Clustering Coefficient = 1.0
Node 4: Clustering Coefficient = 1.0
Node 2713: Clustering Coefficient = 1.0
Node 7: Clustering Coefficient = 1.0
```

Top 5 Clustering Coefficients(Cont).

- A high clustering coefficient suggests that these 5 nodes are very connected and all their friends know each other. In other words, users(other TV show pages) engaging with these TV show pages tend to have mutual connections, implying a cohesive community of fans or followers sharing common interests.
- We examined the node list file and discovered which nodes correspond to specific TV shows.

Nodes	Clustering Coefficient
Living Biblically (Node 2972)	1.0
BOOM (Node 3283)	1.0
Jorge Ramos y su Banda (Node 4)	1.0
Return To Amish (Node 2713)	1.0
The Voice Kids (Node 7)	1.0

Top 5 Clustering Coefficients(Cont).

Living Biblically (Node 2972)

BOOM (Node 3283)

Return To Amish (Node 2713)

Top 5 Betweenness Centrality

- Similar to the clustering coefficients, we used a built-in method from NetworkX
- We then used the sorting method (in reverse) to sort the betweenness centrality from highest to lowest. We extract only the top 5.

```
Node 3254: Betweenness Centrality = 0.10544488181477074
Node 2008: Betweenness Centrality = 0.09352541687013526
Node 819: Betweenness Centrality = 0.0804900367587108
Node 2170: Betweenness Centrality = 0.07471499425323284
Node 2751: Betweenness Centrality = 0.07465790776474893
```

Top 5 Betweenness Centrality (Cont).

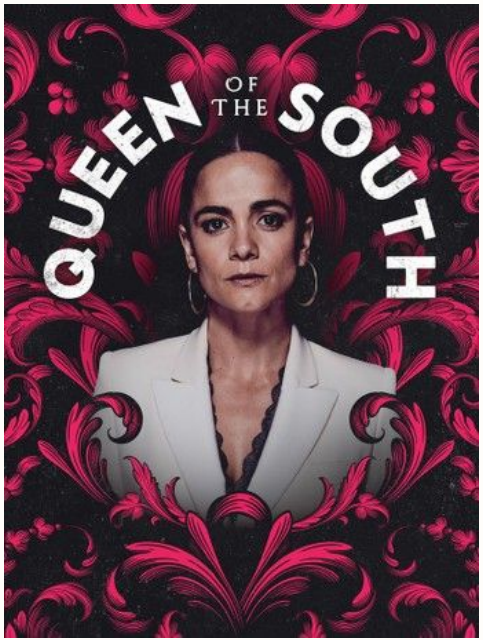
- Nodes with high betweenness centrality suggests that these nodes plays a critical role in connecting different parts of the network together
- We examined the node list file and discovered which nodes correspond to specific TV shows.

Nodes	Betweenness Centrality
Queen of the South (Node 3254)	0.10544488181477074
Home & Family (Node 2008)	0.09352541687013526
The Tonight Show Starring Jimmy Fallon (Node 819)	0.0804900367587108
The Voice Global (Node 2170)	0.07471499425323284
The Voice (Node 2751)	0.07465790776474893

Top 5 Betweenness Centrality (Cont).

The Voice (Node 2751)

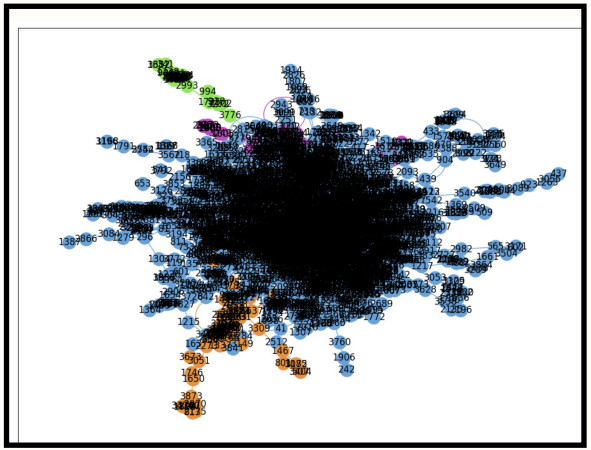
Tonight Show Jimmy Fallon (Node 819)



Queen of the South (Node 3254)

Community Detection: G.N results

- After running the algorithm in NetworkX using 2 steps, we get 4 communities and the modularity score is approximately 0.036.
- Visualization on NetworkX



```
# of communities:  4 , modularity score:  0.03602357841173616
```

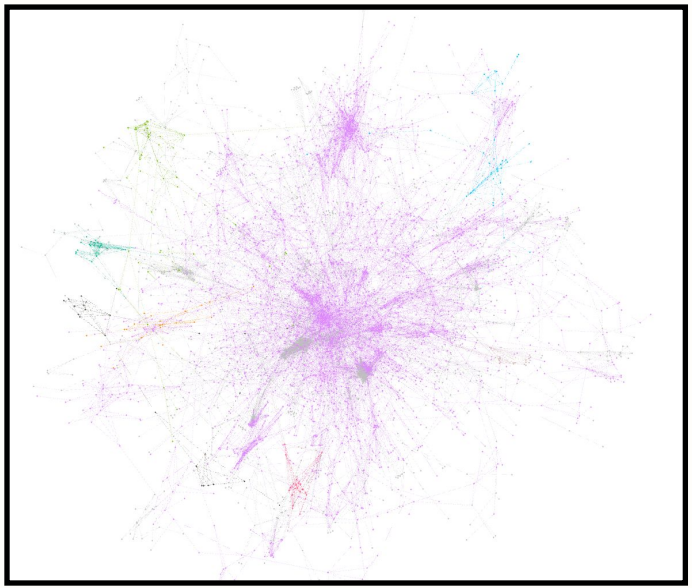
Community Detection: G.N results

- After running the algorithm in Gephi, we get 695 communities and the modularity score is approximately 0.22.

Communities

Number of communities:	695
Maximum found modularity:	0.22318083

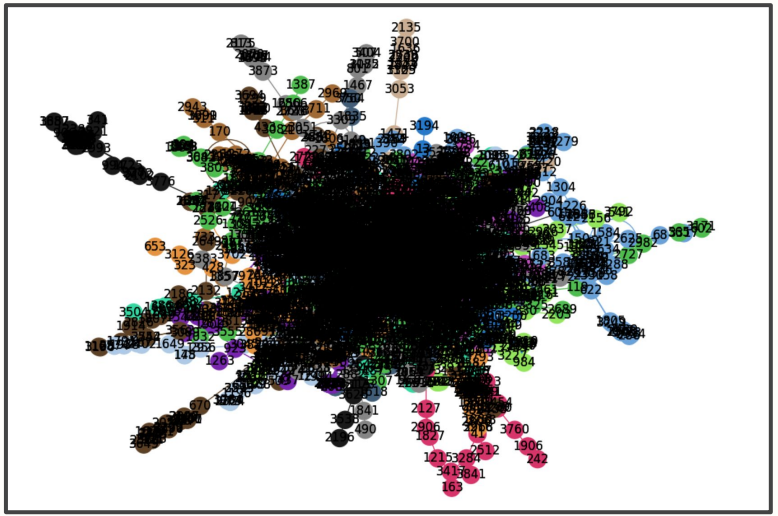
- Visualization on Gephi using Yifan Hu Layout



Community Detection: Louvain Results

- When running the program, we get that the total number of communities is 46 and the modularity score is approximately 0.87.

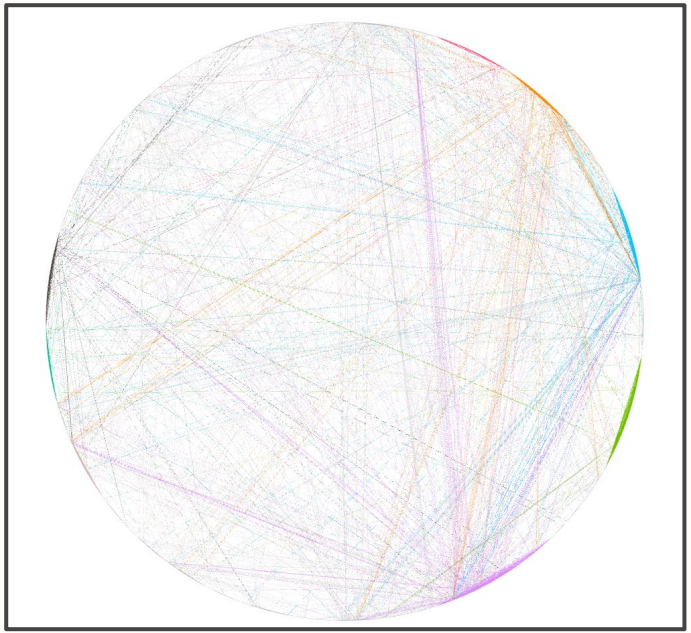
- Visualization on NetworkX

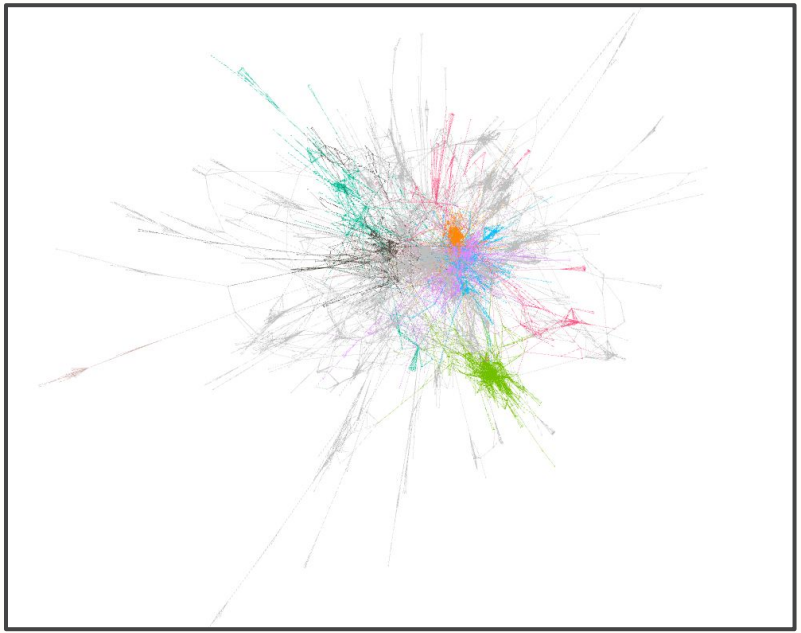


```
# of communities:  46 , modularity score:  0.8721668853348115
```

Community Detection: Louvain Graph

- Visualization on Gephi (Circular and Yifan Hu)



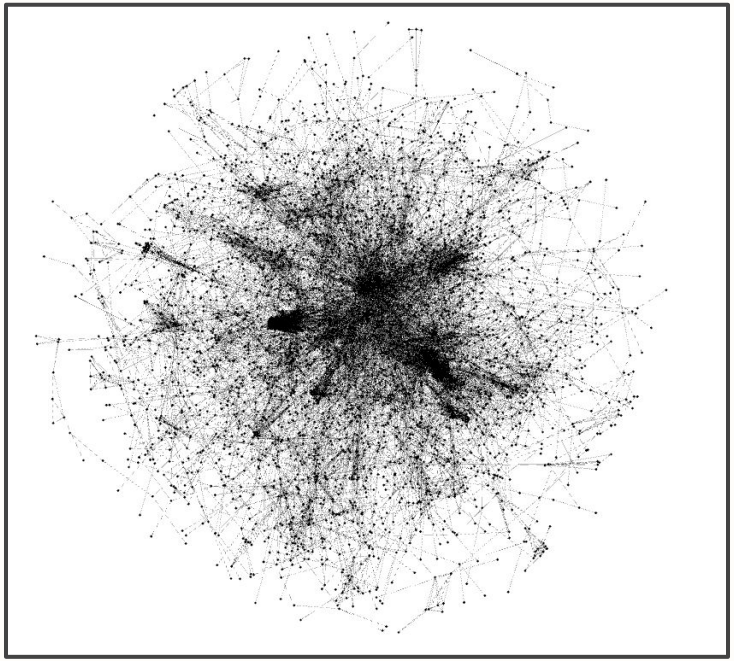


04 Conclusion

- High Clustering Coefficient **doesn't mean it's popular** in the graph
- High Betweenness Centrality **means you're important** in the graph
- Girvan Newman finds **4** communities in NetworkX and **695** in Gephi. Both results have **one big community** that takes **more than half** of the nodes. **Possible reasons** for the difference are
    - The graph may have **distinct characteristics** that **favor** one algorithm over the other
    - Louvain algorithm might be more sensitive to detecting **smaller, more internally cohesive** groups. Girvan-Newman may identify **larger, more interconnected communities**.
    - Both results have shown that the graph has a **homogeneous** structure and the graph represents a **single, large community without clear subdivisions**. These results can also be deduced from the visualization of the dataset.

Visualization

- Visualization of the dataset on Gephi (Yifan Hu Proportional Layout)



Resources and Tools

- Python, Network X
- Gephi
- Google Colab
- networkrepository.com
    - https://networkrepository.com/fb-pages-tvshow.php

Thanks!

CREDITS: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon**, and infographics & images by **Freepik**

Please keep this slide for attribution