ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Academic year: 2021-2022

Subject: Social Network Analysis (SNA)

Analysis of a Twitter Network

Full name: Vasileios-Ilias Drouzas

AM: p3180051

I. Introduction

Twitter is a major social media platform where users can interact with each other via messages called 'tweets'. A lot of information is shared there on a daily basis. People from all over the world use Twitter as their place to share what is going on with their lives and any ideas coming from their mind. The purpose of this project is to perform network analysis on Twitter, create a friendship network based on its users, and see how these users are connected. The graph created allows us to inspect the structure of the network's relationships.

Why Twitter?

Twitter is a very popular social media platform and a lot of people use this platform as their primary source of sharing information. Most of the user profiles in Twitter are public, so scraping data is fairly easy. All the information on Twitter is publicly available through the Twitter API. So Twitter would be a perfect network to analyze, due to the vast amount of information in it!

Twitter API

In order to access the data from Twitter, we need to use the Twitter API. After setting a developer account on Twitter, we get four authentication keys, which are used in the python script to find friends of specific users.

II. Data Collection

Twitter API and the twitter library are used to write a script in order to collect the data.

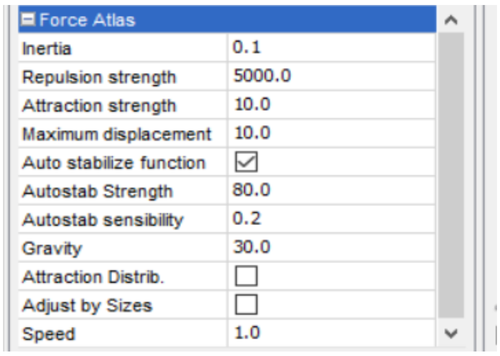
My main focus was on the Indian Bollywood industry to get the users. So, T-series (an Indian music record label company) is a good choice to start with. We can get people TSeries follows. T series follows many celebrities , thus we are taking data from 13 celebrities to form a network.

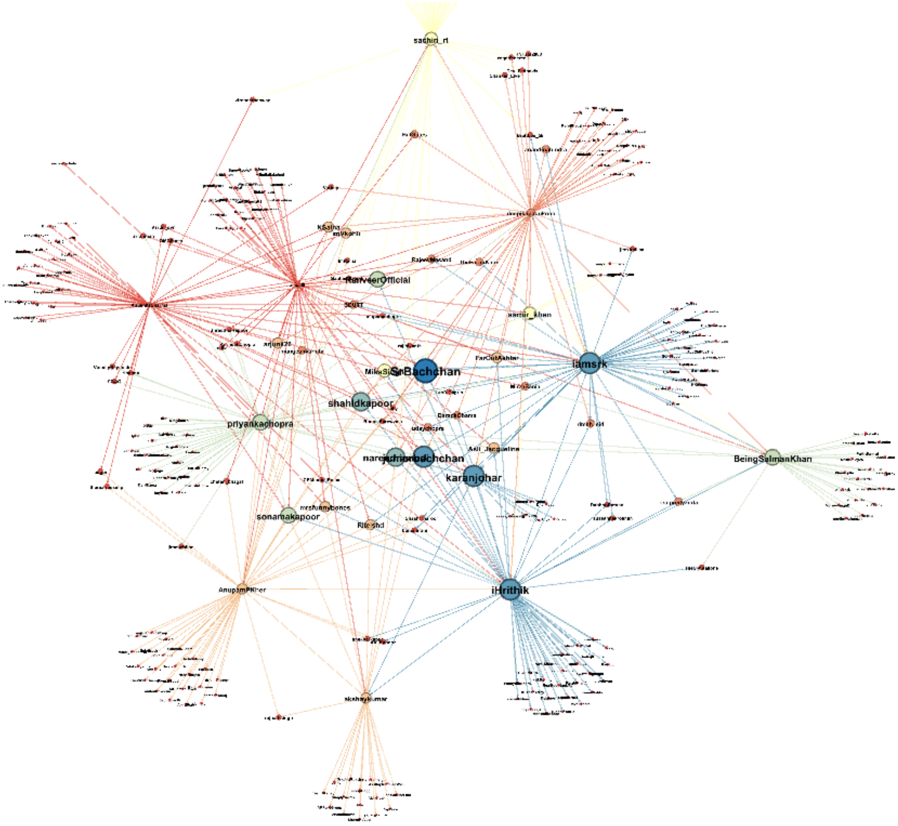
Once we have this data ready, we save it in a .csv file and then we are ready to import this data to Gephi to draw network diagrams and calculate network measures. We import our data in two csv files: nodes.csv and edges.csv and we are ready to start exploring!

III. Data Visualization

Now, it's time to use Gephi to analyze the data we collected in the previous section. Gephi is a very popular visualization and exploration software for graphs and networks, an open-source and free tool. For some tools that are not available via Gephi, I used NetworkX package to cover the topics, as we will see below.

Using *Force Atlas* layout in Gephi, we get the following visualization after setting repulsion strength to 5000:



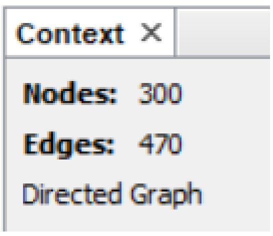


The graph is directed (see below) and the outgoing edge shows that the user node follows the person to which the arrow is pointed.

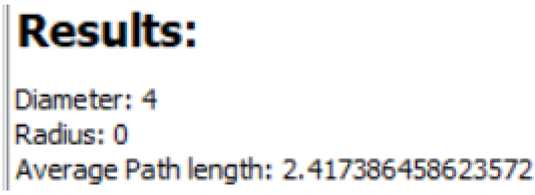
IV. Analyzing the network.

1. Topology.

In general, a graph G is defined by two sets, a set for *nodes* and a set for *edges*. The nodes in our network represent the users and the edges represent the connection between the users. The number of nodes in our graph is 300 , the number of edges is 470 and the graph is directed, as we can see below:



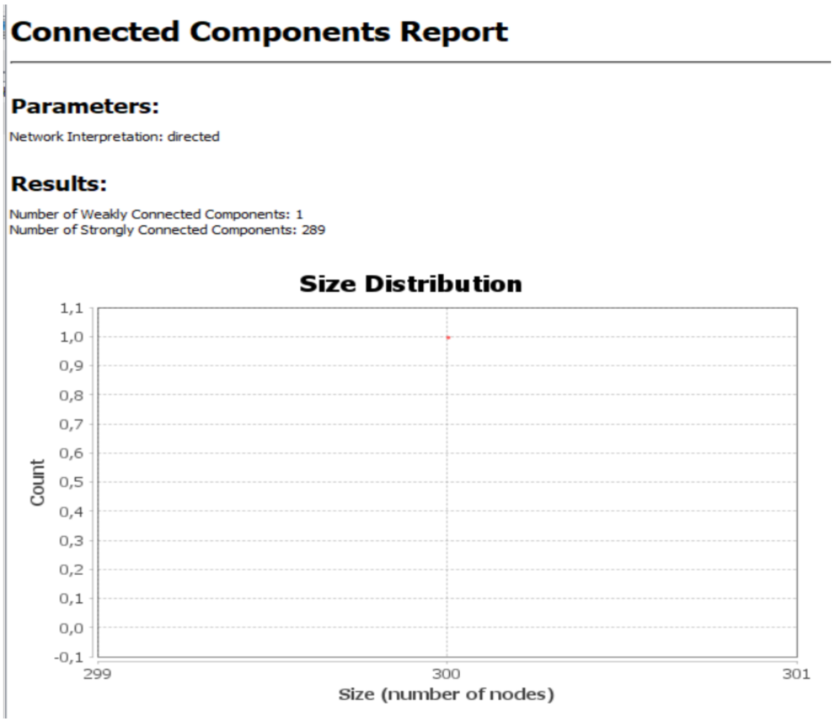
Now we can calculate the average path length and the diameter of the network. The diameter (also known as the longest shortest path) shows us the distance between two users that normally are not friends. Obviously, we can safely expect that the diameter of the network would be certainly bigger than the average path length. Using the statistics field in Gephi, we get the following results:



As expected, Diameter(=4) is bigger than Average path length(=2.42).

2. Component measures

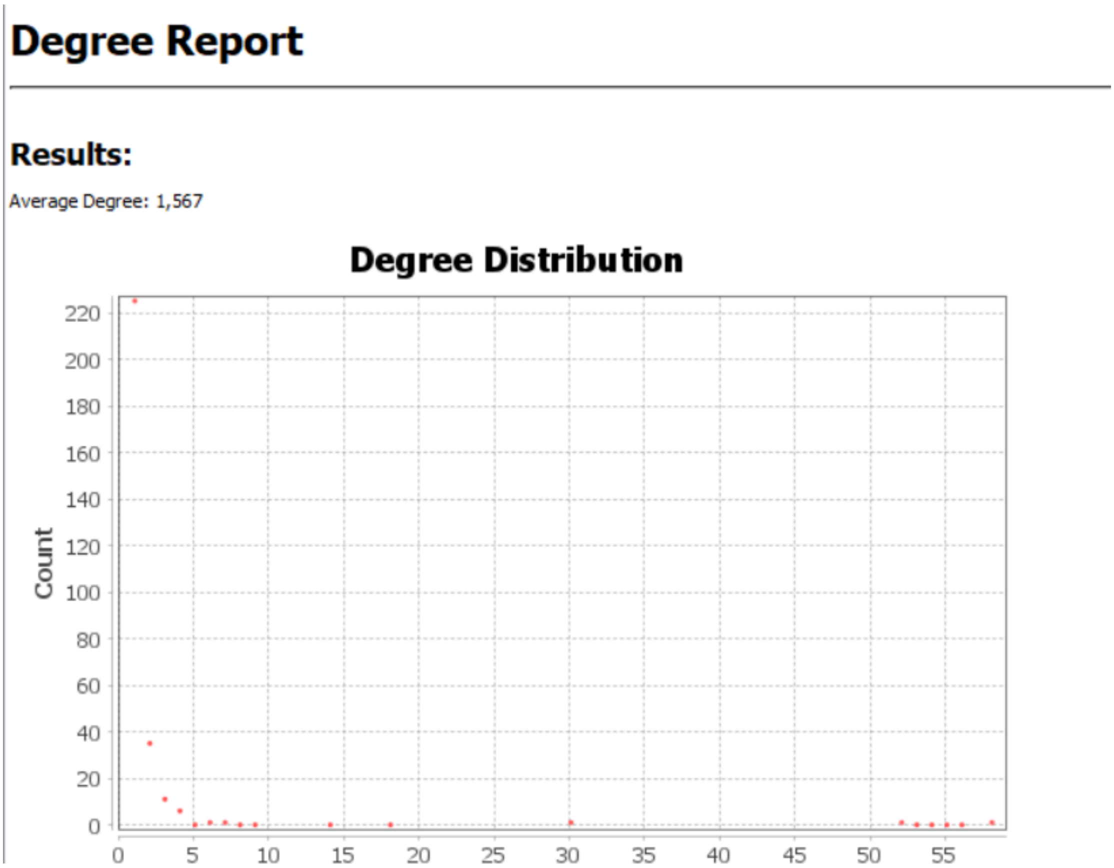
Here we will explore the number of components in the graph.

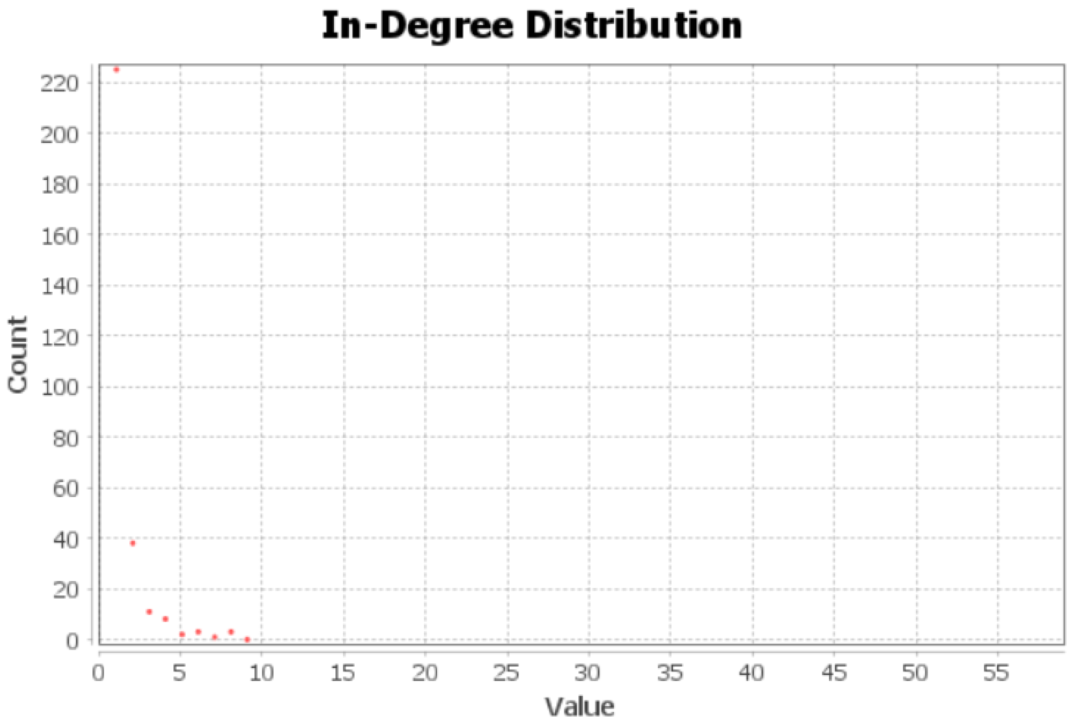


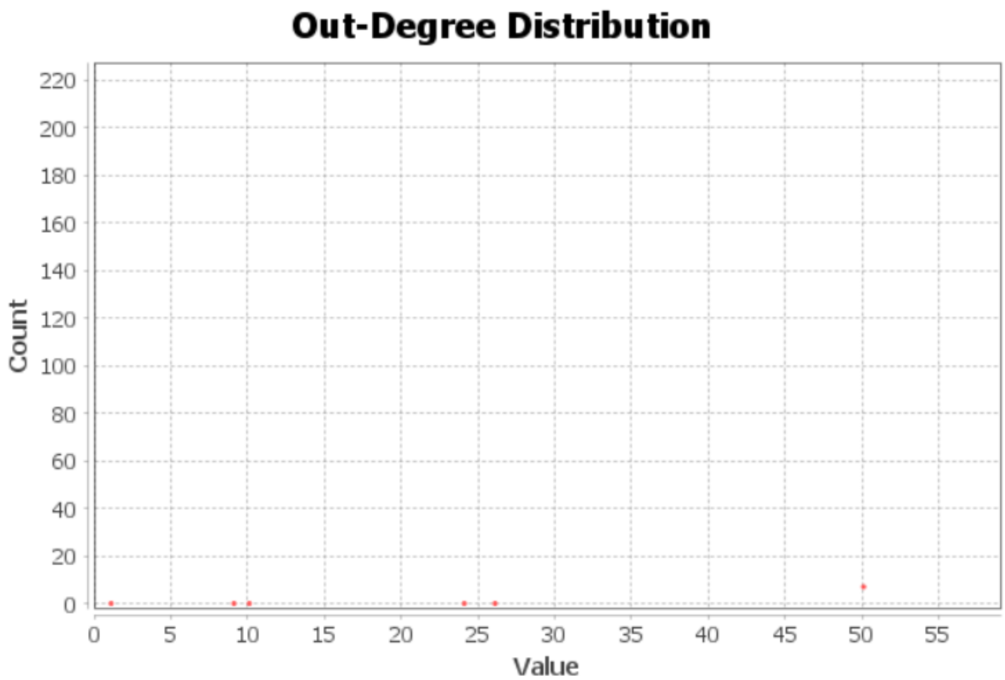
According to the connected components report, there is only one weakly connected component and 289 strongly connected components. The size distribution shows one (Giant) component that consists of 300 nodes. This means that any user can exist in a pattern with any other user.

3. Degree measures

The degree of a node is the number of edges that are adjacent to the node. The number of head ends adjacent to the node is called the *indegree* of the node, while the number of tail ends adjacent to the node is called the *outdegree* of the node. This measure can show us how certain users can be considered for a certain position in a pattern.



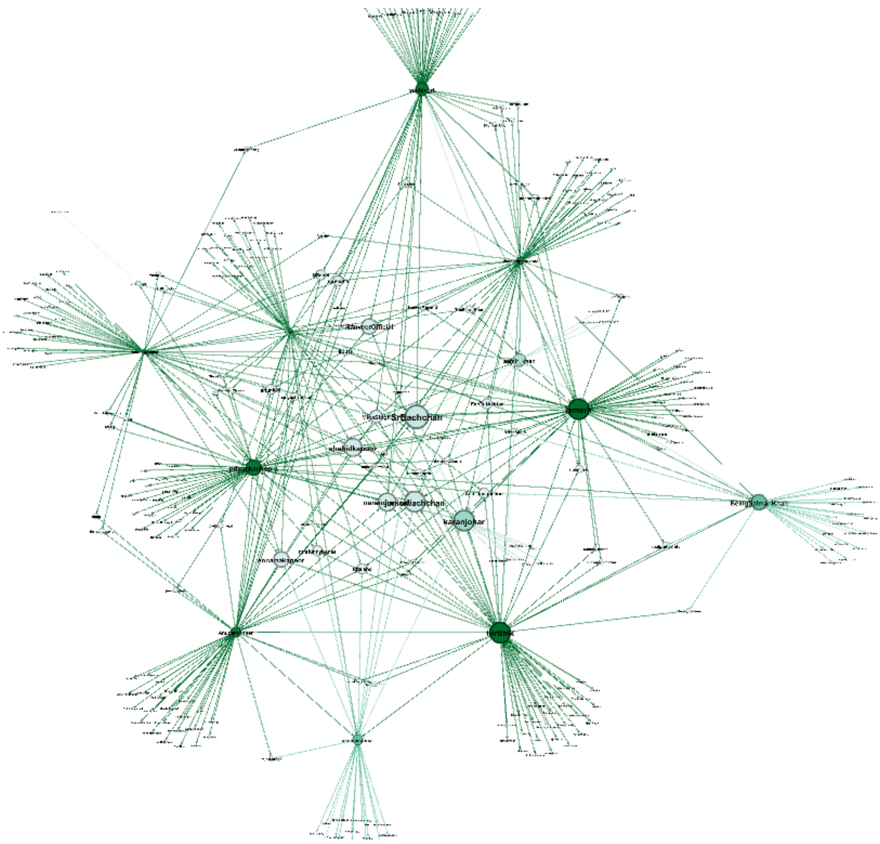




After looking at these graphs, we may conclude that:

● A big part of the nodes has a degree less than 10. Also an other part, a bit smaller have degrees over 50 and a small number of exceptions that have degrees between. (see below to distinguish which graphs have a degree of over 50)

● A ranking based on degree in the color give us the graph:



Needless to say, the darker the green, the bigger the degree. This means that deeper shades of green represent a high degree node, while the most close to white ones represent a low degree node. A list of the nodes with the highest degree follows:

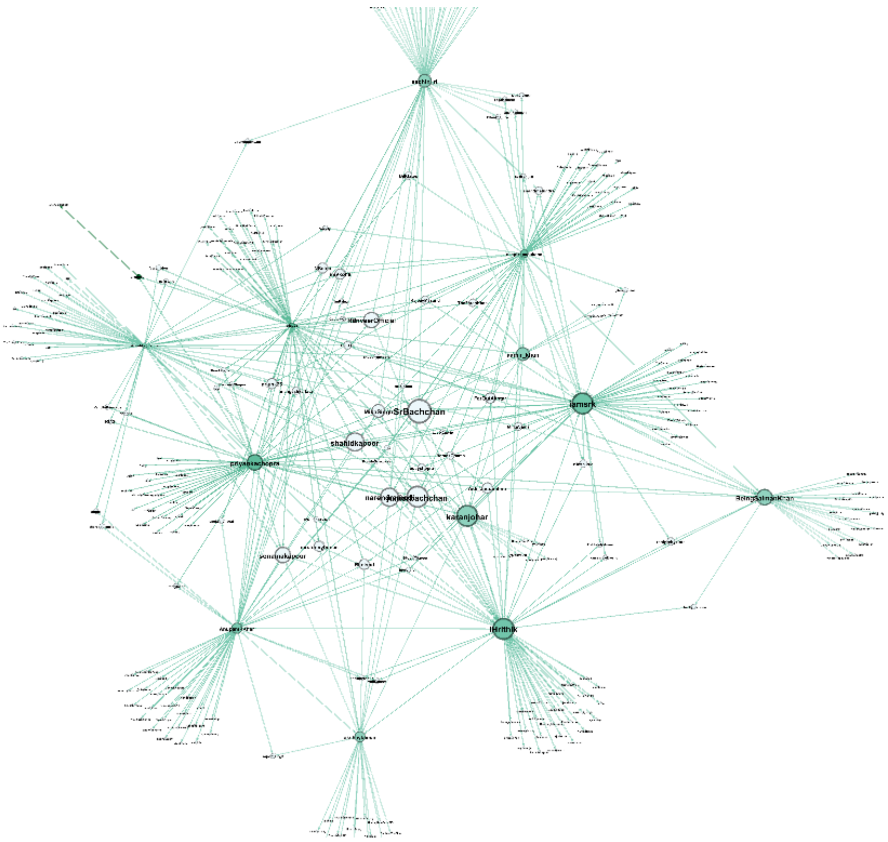
Label	In-Degree	Out-Degree	Degree
iamsrk	8	50	58
iHrithik	8	50	58
priyankachopra	6	50	56
sachin_rt	5	50	55
AnupamPKher	4	50	54
deepikapadukone	3	50	53
AnushkaSharma	2	50	52
aliaa08	2	50	52
BeingSalmanKhan	6	24	30
akshaykumar	4	26	30
karanjohar	8	10	18
aamir_khan	5	9	14
SrBachchan	9	0	9
juniorbachchan	8	0	8
shahidkapoor	7	0	7

The maximum degree in the graph is shared by two nodes, iHrithik and iamsrk , with a size of 58. They both also have an in-degree of 8 and an out-degree of 50. In a way, we could say that the first 8 nodes are the most popular and we would not necessarily be wrong. However, it is wrong to use only the degree to make our decision.

4. Centrality measures

Now, each node will be colored in relation to the measure that is being examined. We will watch the scores from the Data Laboratory of Gephi in each occasion.

a. Closeness Centrality

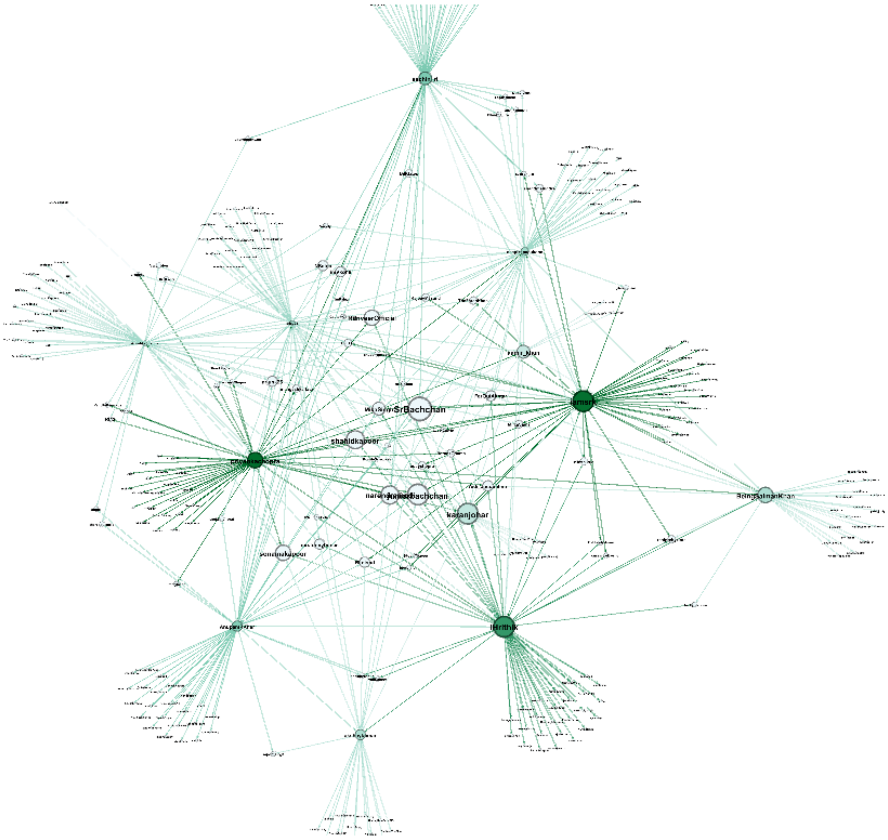


Label	Closeness Centrality
arrahman	1.0
aliaa08	0.54562
priyankachopra	0.531083
deepikapadukone	0.471609
iamsrk	0.469388
iHrithik	0.458589
AnushkaSharma	0.447605
AnupamPKher	0.423513
aamir_khan	0.361111
sachin_rt	0.358084
karanjohar	0.349299
akshaykumar	0.346466
BeingSalmanKhan	0.337853
Asli_Jacqueline	0.0
MirzaSania	0.0

(the rest of nodes have closeness centrality of 0, so I skipped them)

Closeness centrality tries to find nodes that spread information in the graph. Nodes with a high closeness score have the shortest distance to all other nodes. In our case, arrahman scores very high (1), while most nodes have a closeness centrality score of 0 and some between. One conclusion here is that arrahman has a very considerable influence within the network due to the control over information passing to the other users.

b. Betweenness Centrality

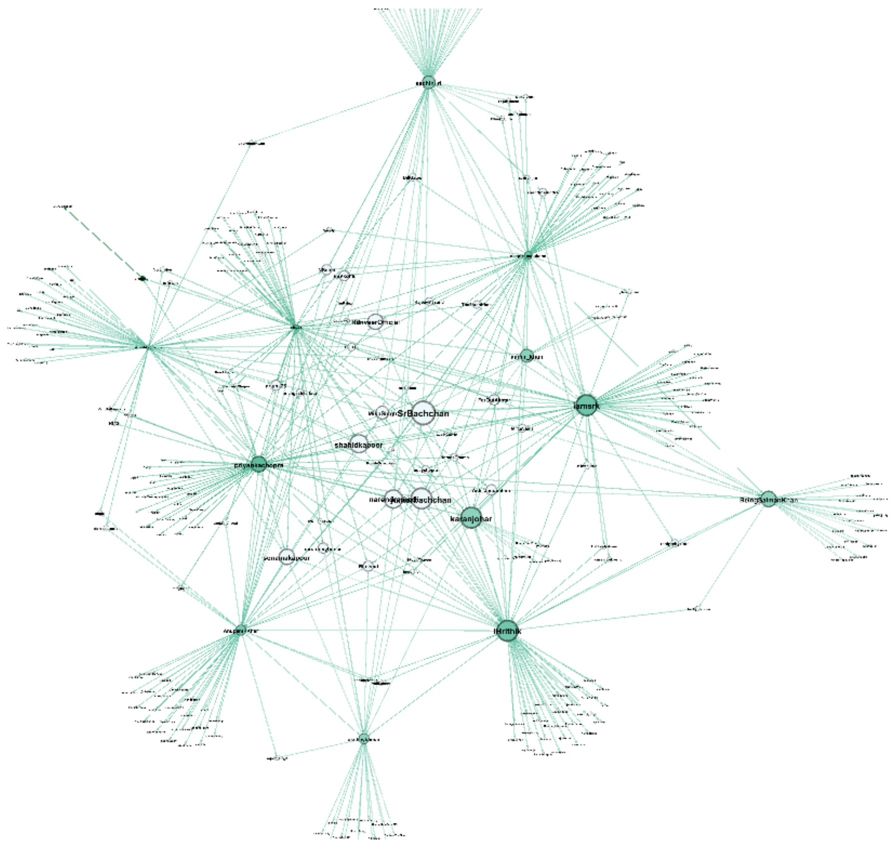


Label	Betweenness Centrality
priyankachopra	991.720238
iamsrk	971.905952
iHrithik	741.530952
sachin_rt	414.269048
AnupamPKher	383.442857
aliaa08	275.483333
deepikapadukone	274.05119
AnushkaSharma	269.683333
akshaykumar	227.971429
BeingSalmanKhan	225.492857
karanjohar	161.15
aamir_khan	138.29881
arrahman	12.0
Asli_Jacqueline	0.0
MirzaSania	0.0
realpreityzinta	0.0

Betweenness centrality is a way of detecting the amount of influence a node has over the flow of information in the graph. We can use it to find nodes that serve as a bridge from one part of the graph to another. This graph is something in the middle between the Degree graph and the Closeness Centrality graph. We can find some weak nodes but there is also a big amount of strong nodes, in green color.

Here the top nodes are *priyankachopra*, *iamsrk* and *iHrithik* , all of which are the top 3 of the graph that is sorted by degree.

c. Harmonic closeness Centrality

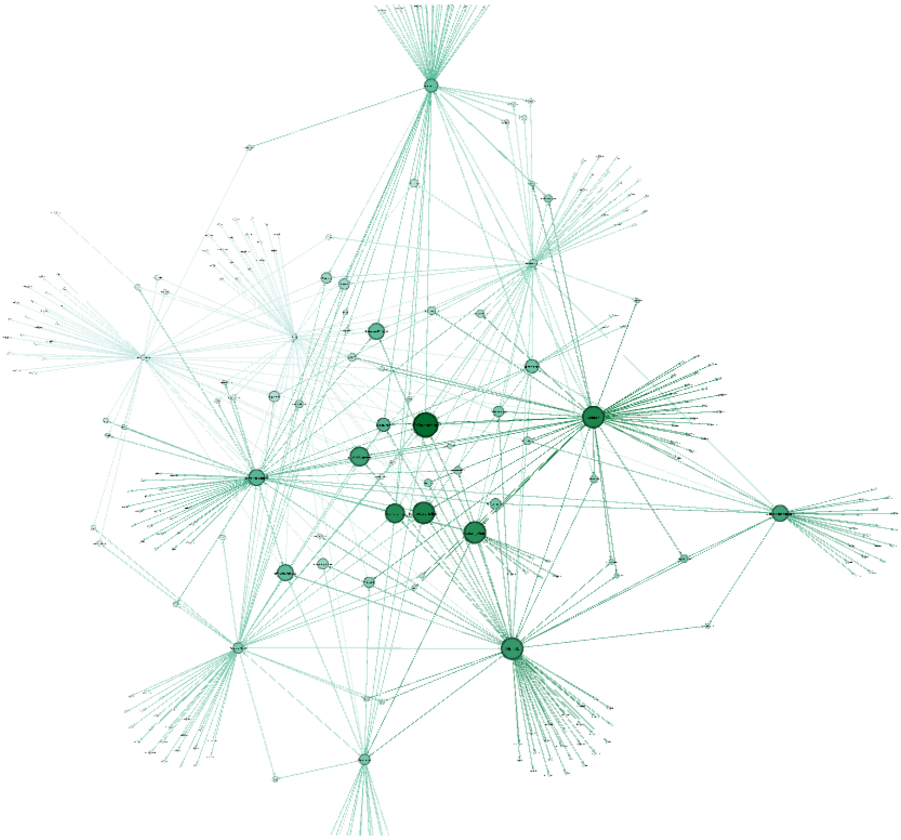


<u>Head to head comparison vs Closeness centrality:</u>

Label	Closeness Centrality	Harmonic Closeness Centrality
arrahman	1.0	1.0
aliaa08	0.54562	0.583612
priyankachopra	0.531083	0.575251
deepikapadukone	0.471609	0.535674
iamsrk	0.469388	0.534002
iHrithik	0.458589	0.525641
AnushkaSharma	0.447605	0.516722
AnupamPKher	0.423513	0.495541
sachin_rt	0.358084	0.447603
akshaykumar	0.346466	0.410256
aamir_khan	0.361111	0.399944
BeingSalmanKhan	0.337853	0.390747
karanjohar	0.349299	0.380435
Asli_Jacqueline	0.0	0.0
MirzaSania	0.0	0.0
realpreityzinta	0.0	0.0

Harmonic closeness centrality is a variant of closeness centrality, so the results are expected to be close. Both metrics normalize the score in [0,1] , but what we can consider is that Harmonic closeness centrality scores every node higher than closeness centrality.

d. Eigenvector centrality



Head to head vs all the other centrality measures

Label	Closeness Centrality	Harmonic Closeness Centrality	Betweenness Centrality	Eigenvector Centrality
SrBachchan	0.0	0.0	0.0	1.0
juniorbachchan	0.0	0.0	0.0	0.874242
iamsrk	0.469388	0.534002	971.905952	0.864608
narendramodi	0.0	0.0	0.0	0.835027
karanjohar	0.349299	0.380435	161.15	0.820629
iHrithik	0.458589	0.525641	741.530952	0.750781
shahidkapoor	0.0	0.0	0.0	0.707052
BeingSalmanKhan	0.337853	0.390747	225.492857	0.684861
MikaSingh	0.0	0.0	0.0	0.582966
priyankachopra	0.531083	0.575251	991.720238	0.582155
sonamakapoor	0.0	0.0	0.0	0.548814
RanveerOfficial	0.0	0.0	0.0	0.545309
sachin_rt	0.358084	0.447603	414.269048	0.524375
aamir_khan	0.361111	0.399944	138.29881	0.492708
realpreityzinta	0.0	0.0	0.0	0.488357
Asli_Jacqueline	0.0	0.0	0.0	0.467318
akshaykumar	0.346466	0.410256	227.971429	0.462538
FarOutAkhtar	0.0	0.0	0.0	0.454492
AnupamPKher	0.423513	0.495541	383.442857	0.42115
ritesh_sid	0.0	0.0	0.0	0.417466
Riteishd	0.0	0.0	0.0	0.409366
udaychopra	0.0	0.0	0.0	0.380472
anandmahindra	0.0	0.0	0.0	0.368836
MirzaSania	0.0	0.0	0.0	0.364896
BarackObama	0.0	0.0	0.0	0.359402
deepikapadukone	0.471609	0.535674	274.05119	0.344883
DabbooRatnani	0.0	0.0	0.0	0.343447
sussannekroshan	0.0	0.0	0.0	0.343447
mrsfunnybones	0.0	0.0	0.0	0.333279

Eigenvector centrality is also a measure of the influence of a node in a network. It differs from degree centrality because a node receiving many links does not have a high eigenvector centrality. Also, eigenvector centrality takes into consideration not only how many connections a vertex has, but also the centrality of the vertices it is connected to.

What we can notice here is that eigenvector centrality has little changes in the values from one node to the next one, when sorted, whereas the other centrality measures had some nodes with varying values and most nodes with 0 value. Also, an impressive statistic is that *SrBachchan* scored 0 on all other closeness centrality measures, and 1(the biggest possible value) in Eigenvector centrality.

5. Clustering effects

The first clustering measure we will examine is the Average Clustering Coefficient. In our case, it results in 0.12, which alone is indicative of a network comprised of numerous weak ties. The nodes tend to avoid clustering.

Label	Clustering Coefficient
LeoDiCaprio	1.0
BrettLee_58	1.0
DabbooRatnani	1.0
jiteshpillaai	1.0
sussannekroshan	1.0
rajnathsingh	1.0
AnilKapoor	1.0
shekharkapur	1.0
rihanna	1.0
jimmyfallon	1.0
9GAG	1.0
VancityReynolds	1.0
ParineetiChopra	1.0
ndtv	1.0
ShraddhaKapoor	1.0
Varun_dvn	1.0
OMGFacts	1.0
realpreityzinta	0.833333
anandmahindra	0.833333
ritesh_sid	0.833333
BarackObama	0.833333
Asli_Jacqueline	0.666667
FarOutAkhtar	0.666667
BDUTT	0.666667
TheFarahKhan	0.666667
udaychopra	0.666667
sonamakapoor	0.666667
shahidkapoor	0.642857
mrsfunnybones	0.583333

Label	Clustering Coefficient
shahidkapoor	0.642857
mrsfunnybones	0.583333
RanveerOfficial	0.566667
juniorbachchan	0.535714
MirzaSania	0.5
TheSlyStallone	0.5
anupamachopra	0.5
RajeevMasand	0.5
BillGates	0.5
mangeshkarlata	0.5
rogerfederer	0.5
RNTata2000	0.5
Dev_Fadnavis	0.5
Shankar_Live	0.5
CPMumbaiPolice	0.5
chetan_bhagat	0.5
narendramodi	0.47619
SrBachchan	0.472222
Riteishd	0.416667
MikaSingh	0.4
arjunk26	0.333333
arrahman	0.333333
aamir_khan	0.219697
karanjohar	0.1625
imVkohli	0.083333
NSaina	0.083333
aliaa08	0.043673
BeingSalmanKhan	0.035714
priyankachopra	0.03551

Label	Clustering Coefficient
BeingSalmanKhan	0.035714
priyankachopra	0.03551
akshaykumar	0.032764
iHrithik	0.032495
deepikapadukone	0.030196
iamsrk	0.026018
AnushkaSharma	0.017551
AnupamPKher	0.013725
sachin_rt	0.013062
theathiyashetty	0.0
arbaazSkhan	0.0
khanarpita	0.0
ShahDaisy25	0.0
PulkitSamrat	0.0
iamsnehaullal	0.0
tweetbeinghuman	0.0
luvsalimkhan	0.0
IuliaVantur	0.0
SohailKhan	0.0
binaakak	0.0
impoornapatel	0.0
Vinayvirmani24	0.0
atulreellife	0.0
NirvanKhan15	0.0
arhaankhan0798	0.0
Iamwaluscha	0.0

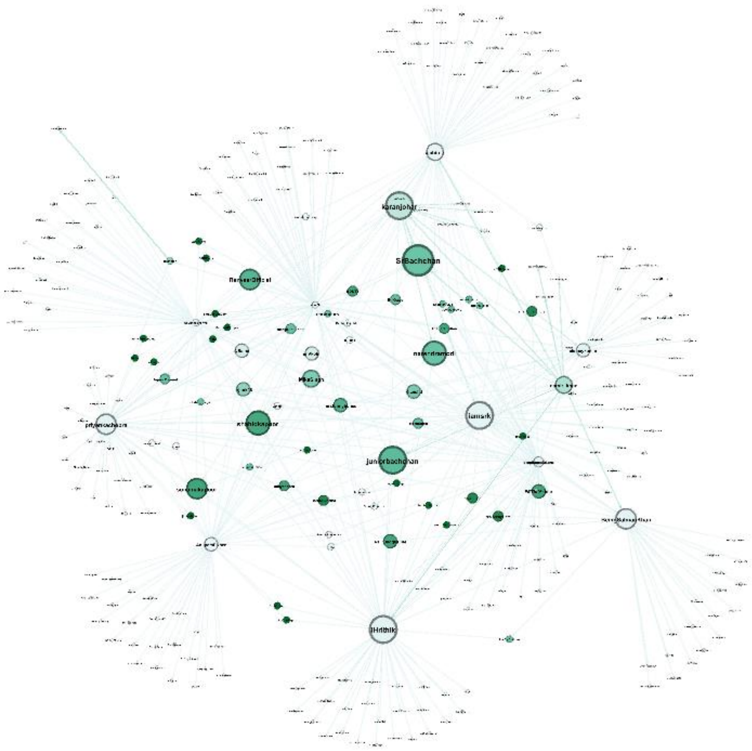
However, the bottom nodes which we see here and all the rest of them which are not in the screenshot above (and there are many of them) , they have a coefficient of 0, so they can be misleading. Still there are a lot of nodes in the graph here that attempt to form clusters (at least all the nodes in the first column above).

Now, let’s talk about triadic closure. Triadic closure supposes that if two people know the same person, they are likely to know each other.

The number of these triangles in the network we found before (256) can be used to find clusters in the network. Triadic closure can be measured by using the clustering coefficient, as we will see below.

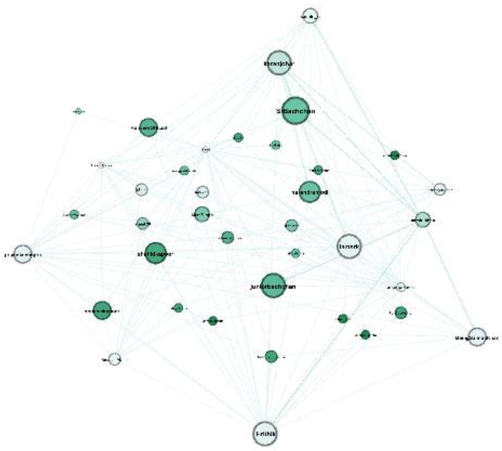
There is also an other plugin, slightly different to average clustering coefficient, called Clustering Coefficient, which also calculates the number of triangles present in the graph. Using the triangle method, I found the value of clustering coefficient at 0.07 (even lower than the average clustering coefficient) , while the number of triangles was 256 and the number of paths was 11.535.

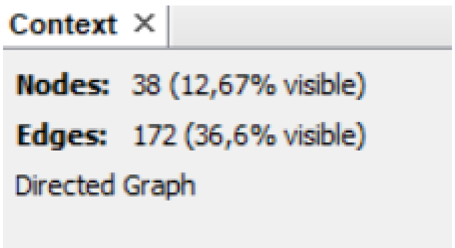
Graphically:



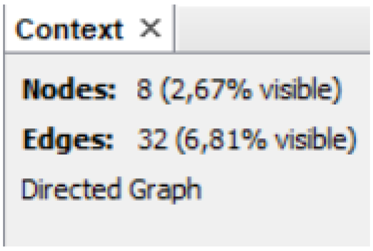
We see here that the most central nodes have the tendency to form clusters (bold green color), while the most remote clusters do not (closer to white color).

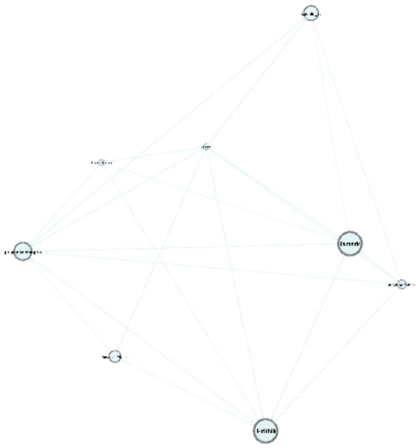
Next, we will use a Degree filter in Gephi to visualize the most node that are most likely to form a cluster. This subgraph consists of only 38 nodes, and these are suitable to form clusters. Calculating again the average clustering coefficient, we get a value of 0.464 which is bigger than our previous one!





But what if we do this again one time? This subgraph consists of only 8 nodes, and the average clustering coefficient has a value of 0.671 this time! We can safely conclude that these 8 nodes certainly have a clustering tendency.





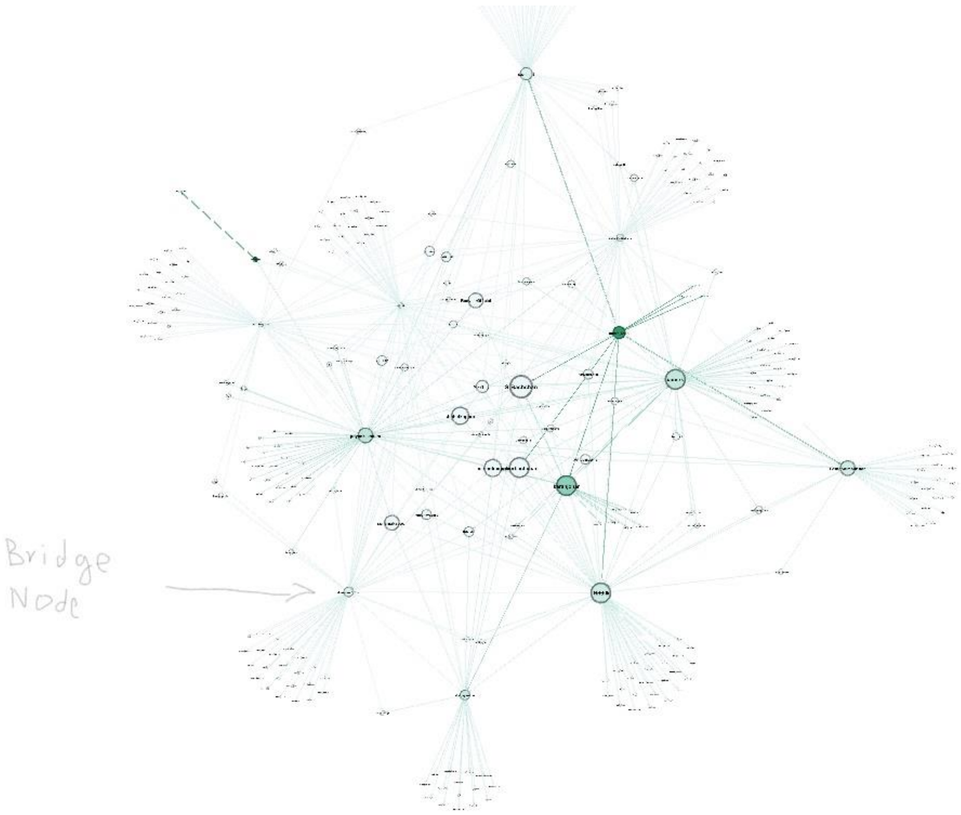
So, what we can conclude is that clustering is present in the graph, mainly on smaller datasets. The initial average clustering coefficient gives us the impression that the whole network would not have a clustering tendency, but if we separate some nodes altogether, the clustering feature can be present.

6. Bridges

A bridge is a direct tie between two nodes that would otherwise be in disconnected components in the graph. Here, we will examine the Bridging Centrality feature. The graph is on the next page.

In social networks, like our case, one person could be close friends with two other people who hardly know each other.

Taking into consideration the in-degree and out-degree values, we can spot that all nodes have at least one edge incoming but not all nodes have at least one edge leaving. Since this is a directed graph, an edge is one-way, so we can say that edges can be considered as bridges that connect nodes. Of course, some combinations of nodes make much more sense than others, but this is something that can be concluded by someone's judgement.

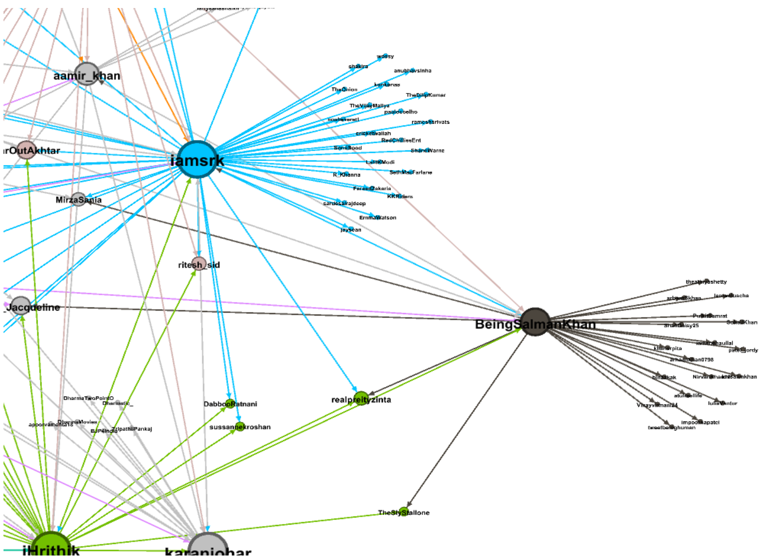


An example of a bridge node is shown in the graph above. Here we can safely conclude that the remote nodes (those that have an out-degree of 0) are being connected by bridges to nodes that have an out-degree of at least 1 (enough to connect them with the remote ones). Otherwise, there would be no way to connect these nodes.

Running on networkx, we find that there are 226 bridges in the graph. Let's take a look at a part of them:

```
('BeingSalmanKhan', 'arbaazSkhan')
('BeingSalmanKhan', 'khanarpita')
('BeingSalmanKhan', 'ShahDaisy25')
('BeingSalmanKhan', 'PulkitSamrat')
('BeingSalmanKhan', 'iamsnehaullal')
('BeingSalmanKhan', 'tweetbeinghuman')
('BeingSalmanKhan', 'luvsalimkhan')
('BeingSalmanKhan', 'IuliaVantur')
('BeingSalmanKhan', 'SohailKhan')
('BeingSalmanKhan', 'binaakak')
('BeingSalmanKhan', 'impoornapatel')
('BeingSalmanKhan', 'Vinayvirmani24')
('BeingSalmanKhan', 'atulreellife')
('BeingSalmanKhan', 'NirvanKhan15')
('BeingSalmanKhan', 'arhaankhan0798')
('BeingSalmanKhan', 'Iamwaluscha')
('BeingSalmanKhan', 'patel_jordy')
('iamsrk', 'shakira')
('iamsrk', 'EmmaWatson')
('iamsrk', 'paulocoelho')
('iamsrk', 'SethMacFarlane')
('iamsrk', 'TheOnion')
('iamsrk', 'sardesairajdeep')
('iamsrk', 'TheVijayMallya')
('iamsrk', 'wossy')
('iamsrk', 'rameshsrivats')
('iamsrk', 'SonuSood')
('iamsrk', 'KKRiders')
('iamsrk', 'LalitKModi')
('iamsrk', 'ShaneWarne')
('iamsrk', 'noaheverett')
('iamsrk', 'konkonas')
('iamsrk', 'jaysean')
('iamsrk', 'FareedZakaria')
('iamsrk', 'R_Khanna')
('iamsrk', 'cricketwallah')
('iamsrk', 'TheDilipKumar')
('iamsrk', 'anubhavsinha')
('iamsrk', 'RedChilliesEnt')
```

Needless to say, BeingSalmanKhan and iamsrk are 2 nodes of the type we mentioned earlier and we can see it in the graph too:



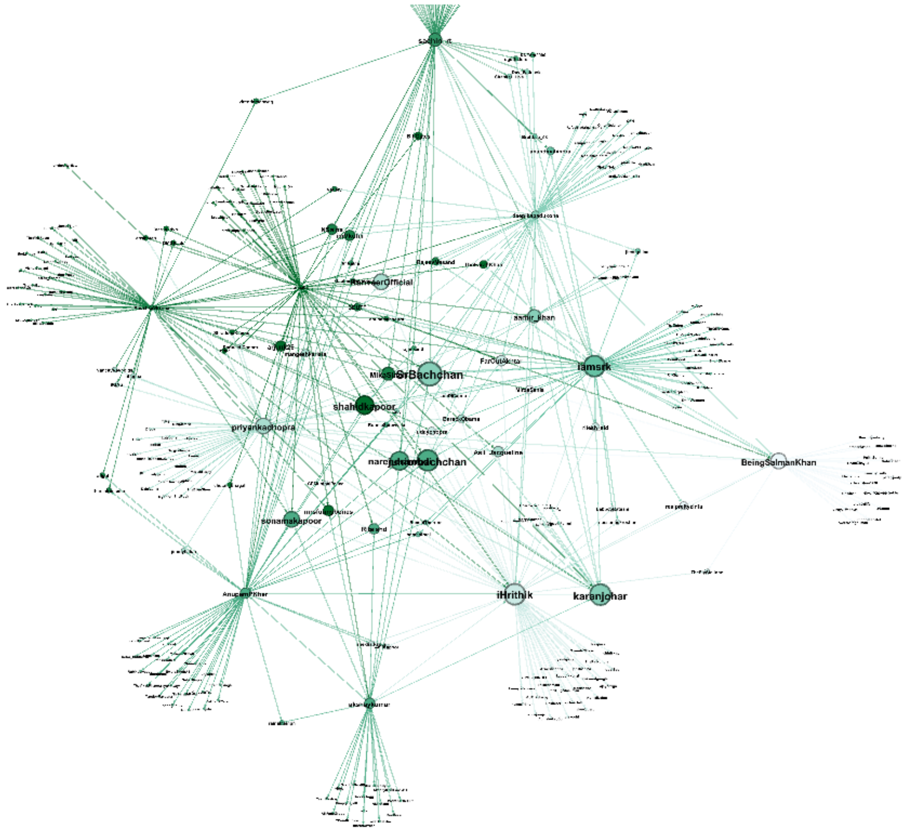
Of course there are more nodes that serve as bridge nodes, following the pattern we discussed here.

About local bridges, networkx found 252 local bridges. See a part of them below:

```
('BeingSalmanKhan', 'theathiyashetty', inf)
('BeingSalmanKhan', 'arbaazSkhan', inf)
('BeingSalmanKhan', 'khanarpita', inf)
('BeingSalmanKhan', 'ShahDaisy25', inf)
('BeingSalmanKhan', 'PulkitSamrat', inf)
('BeingSalmanKhan', 'iamsnehaullal', inf)
('BeingSalmanKhan', 'tweetbeinghuman', inf)
('BeingSalmanKhan', 'luvsalimkhan', inf)
('BeingSalmanKhan', 'IuliaVantur', inf)
('BeingSalmanKhan', 'SohailKhan', inf)
('BeingSalmanKhan', 'binaakak', inf)
('BeingSalmanKhan', 'impoornapatel', inf)
('BeingSalmanKhan', 'Vinayvirmani24', inf)
('BeingSalmanKhan', 'atulreellife', inf)
('BeingSalmanKhan', 'NirvanKhan15', inf)
('BeingSalmanKhan', 'arhaankhan0798', inf)
('BeingSalmanKhan', 'Iamwaluscha', inf)
('BeingSalmanKhan', 'patel_jordy', inf)
('iamsrk', 'shakira', inf)
('iamsrk', 'EmmaWatson', inf)
('iamsrk', 'paulocoelho', inf)
('iamsrk', 'SethMacFarlane', inf)
('iamsrk', 'TheOnion', inf)
('iamsrk', 'sardesairajdeep', inf)
('iamsrk', 'ShashiTharoor', 3)
('iamsrk', 'TheVijayMallya', inf)
('iamsrk', 'bomanirani', 3)
('iamsrk', 'wossy', inf)
('iamsrk', 'rameshsrivats', inf)
('iamsrk', 'SonuSood', inf)
('iamsrk', 'KKRiders', inf)
('iamsrk', 'LalitKModi', inf)
('iamsrk', 'ShaneWarne', inf)
('iamsrk', 'noaheverett', inf)
('iamsrk', 'konkonas', inf)
('iamsrk', 'jaysean', inf)
('iamsrk', 'FareedZakaria', inf)
('iamsrk', 'R_Khanna', inf)
('iamsrk', 'cricketwallah', inf)
```

7. Homophily

Homophily is a feature similar to clustering. It is the tendency of individuals to choose friends that share similar properties .



The assortativity coefficient is used to determine the extent to which certain connected nodes have similar characteristics.

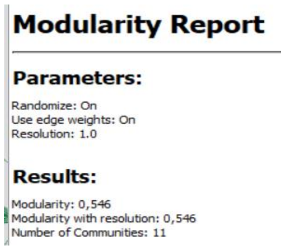
The graph above is generated by coloring the nodes according to the modularity class. From the graph we can see that a major number of edges lies between nodes that are dark green. Also it is safe to conclude that nodes that are at the edges have the least assortavity coefficient than others that are mainly towards the center.

Networkx gives us a value of -0.789 for the assortativity coefficient in the graph G, which implies disassortavity.

8. Modularity

Modularity is a statistic that we can find in Gephi's menu. It tries to find the extent to which the graph tends to create communities (clusters). Running a modularity report in Gephi, with a resolution of

1.0, 11 communities are created and the modularity score is 0.546, which means that there is a tendency in trying to create communities.



Modularity Report

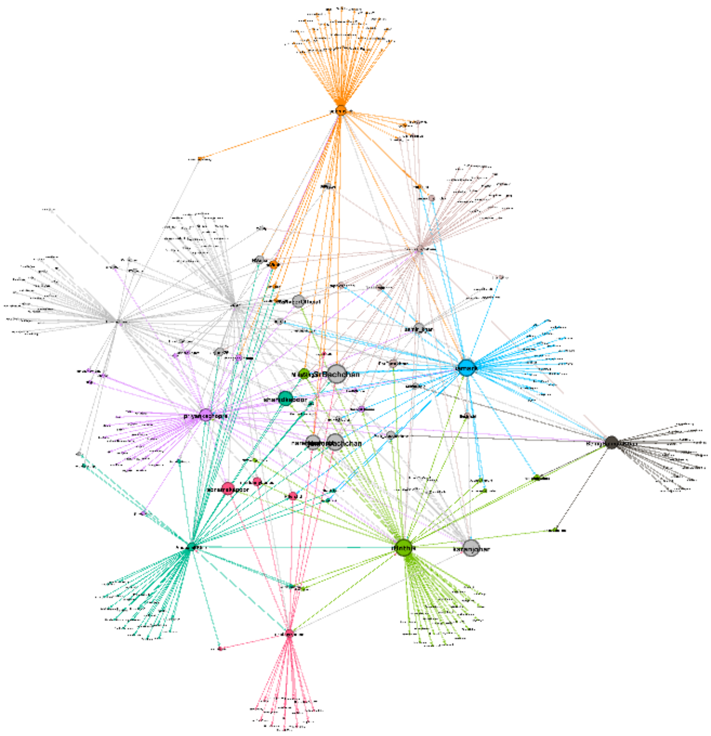
Parameters:

Randomize: On
Use edge weights: On
Resolution: 1.0

Results:

Modularity: 0,546
Modularity with resolution: 0,546
Number of Communities: 11

The 11 communities are shown below, each one having a different color:



The most popular nodes within a community are the ones that are the high-degree nodes, and they mainly are situated at the center of the cluster.

Let's now take a look at the closeness centrality measures of the top nodes:

Label	Closeness Centrality
arrahman	1.0
aliaa08	0.54562
priyankachopra	0.531083
deepikapadukone	0.471609
iamsrk	0.469388
iHrithik	0.458589
AnushkaSharma	0.447605
AnupamPKher	0.423513
aamir_khan	0.361111
sachin_rt	0.358084
karanjohar	0.349299
akshaykumar	0.346466
BeingSalmanKhan	0.337853

The results actually make sense, because these are all high degree nodes. Also, they have a big amount of edges. An impressive stat is that certain communities in the graph share a decent number of the nodes above, which practically means that some popular users belong to the same community.

Cliques

Cliques are defined as a maximal complete subgraph of a graph where each node is connected to all the other nodes. In our case, according to Wikipedia, 'cliques would define a group of individuals who interact with one another and share similar interests'.

Running on networkx, we found 337 cliques. A part of them is shown below:

```
['GautamGambhir', 'AnupamPKher']
['GautamGambhir', 'sachin_rt']
['anupamachopra', 'AnushkaSharma', 'iamsrk']
['therealrussellp', 'AnushkaSharma']
['therealrussellp', 'AnupamPKher']
['Akkistaan', 'akshaykumar']
['RedChilliesEnt', 'iamsrk']
['Arsenal', 'aliaa08']
['harbhajan_singh', 'sachin_rt']
['RajeevMasand', 'iamsrk', 'AnushkaSharma']
['RajeevMasand', 'iamsrk', 'deepikapadukone']
['nickjonas', 'priyankachopra']
['shiekhspear', 'deepikapadukone']
['kritisanon', 'aliaa08']
['cricketwallah', 'iamsrk']
['OMGFacts', 'aliaa08', 'AnushkaSharma']
['BuzzFeedNews', 'AnushkaSharma']
['aplusk', 'priyankachopra']
['Fact', 'AnushkaSharma']
['ImRaina', 'AnupamPKher']
['ImRaina', 'sachin_rt']
['taran_adarsh', 'AnushkaSharma']
['BillGates', 'aliaa08', 'sachin_rt', 'deepikapadukone']
['luvsalimkhan', 'BeingSalmanKhan']
['bhogleharsha', 'sachin_rt']
['itsSSR', 'AnushkaSharma']
['BCCI', 'sachin_rt']
['satyanadella', 'sachin_rt']
['VishalDadlani', 'AnushkaSharma']
['RafaelNadal', 'deepikapadukone']
['chetan_bhagat', 'aliaa08', 'AnupamPKher']
['ImZaheer', 'sachin_rt']
['cnnbrk', 'priyankachopra']
['anandmahindra', 'iamsrk', 'sachin_rt', 'deepikapadukone']
['tweetbeinghuman', 'BeingSalmanKhan']
['kiranshaw', 'deepikapadukone']
['KP24', 'deepikapadukone']
['ANI', 'AnushkaSharma']
['shoaib100mph', 'sachin_rt']
['Oprah', 'priyankachopra']
['thekiranbedi', 'AnupamPKher']
```

9. Graph density

The density of a graph is a measure of how many ties between nodes exist compared to how many ties are possible between nodes. In other words, it is a measure of how close the graph is to a complete graph with the same number of nodes. Running the graph density feature in Gephi, we get the following results:

Graph Density Report

Parameters:

Network Interpretation: directed

Results:

Density: 0,005

The graph scores very low in density measures. So what we see is a sparse network, knowing that 0 means there are no connections at all, and 1 meaning that all possible edges are present in the graph. Given that the density is the total number of edges to the graph divided by the total number of possible edges, we see that the number of edges is pretty low.

10. PageRank

The last part of this assignment refers to the PageRank of each node. Historically, PageRank has been used to measure the importance of web pages and is used as a classification technique when ranking websites.

Using the data laboratory in Gephi, we get the following results regarding Pagerank of each node:

Label	PageRank
oneheartartists	0.006018
SrBachchan	0.004368
iamsrk	0.004041
juniorbachchan	0.004029
karanjohar	0.004026
iHrithik	0.004018
narendramodi	0.003973
BeingSalmanKhan	0.003848
sachin_rt	0.00378
akshaykumar	0.003725
shahidkapoor	0.003626
sonamakapoor	0.003622
aamir_khan	0.003574
priyankachopra	0.003566
RanveerOfficial	0.003562
BJP4India	0.003541
DharmaMovies	0.003541
TripathiiPankaj	0.003541
apoorvamehta18	0.003541
DharmaTwoPointO	0.003541
Dharmatic_	0.003541
satyamevjayate	0.003537
fattysanashaikh	0.003537
sanyamalhotra07	0.003537
Asli_Jacqueline	0.003521
MikaSingh	0.003516
AnupamPKher	0.003506
Riteishd	0.003505
mrsfunnybones	0.003502

What we can conclude from the results is that, except *oneheartartists*, the other nodes are pretty close on the measured values. As we can see in the figure below, there is a tendency for pagerank to score high nodes with relatively high degree values. However, the impressive stat is that *oneheartartists* , being the top node by PageRank, has a degree of 1!(pretty low) Exceptions can always happen, we will keep this in mind.

Label	Degree	PageRank
oneheartartists	1	0.006018
SrBachchan	9	0.004368
iamsrk	58	0.004041
juniorbachchan	8	0.004029
karanjohar	18	0.004026
iHrithik	58	0.004018
narendramodi	7	0.003973
BeingSalmanKhan	30	0.003848
sachin_rt	55	0.00378
akshaykumar	30	0.003725
shahidkapoor	7	0.003626
sonamakapoor	6	0.003622
aamir_khan	14	0.003574
priyankachopra	56	0.003566
RanveerOfficial	6	0.003562
BJP4India	1	0.003541
DharmaMovies	1	0.003541
TripathiiPankaj	1	0.003541
apoorvamehta18	1	0.003541
DharmaTwoPointO	1	0.003541
Dharmatic_	1	0.003541
satyamevjayate	1	0.003537
fattysanashaikh	1	0.003537

Lastly, I would like to highlight the similarity of PageRank with the HITS algorithm. The difference between the two algorithms is that HITS only operates on a subgraph of the graph. Let's compare the two algorithm results in each node:

Label	PageRank	Authority
oneheartartists	0.006018	0.0
SrBachchan	0.004368	0.211415
iamsrk	0.004041	0.194065
juniorbachchan	0.004029	0.216232
karanjohar	0.004026	0.217451
iHrithik	0.004018	0.214838
narendramodi	0.003973	0.164974
BeingSalmanKhan	0.003848	0.174464
sachin_rt	0.00378	0.144272
akshaykumar	0.003725	0.099917
shahidkapoor	0.003626	0.221409
sonamakapoor	0.003622	0.169989
aamir_khan	0.003574	0.12958
priyankachopra	0.003566	0.188829
RanveerOfficial	0.003562	0.18316
BJP4India	0.003541	0.005496
DharmaMovies	0.003541	0.005496
TripathiiPankaj	0.003541	0.005496
apoorvamehta18	0.003541	0.005496
DharmaTwoPointO	0.003541	0.005496
Dharmatic_	0.003541	0.005496
satyamevjayate	0.003537	0.009431
fattysanashaikh	0.003537	0.009431
sanyamalhotra07	0.003537	0.009431
Asli_Jacqueline	0.003521	0.104833
MikaSingh	0.003516	0.136997
AnupamPKher	0.003506	0.113613
Riteishd	0.003505	0.10897
mrsfunnybones	0.003502	0.10983

Although the algorithms seem to agree in the ranking of the nodes (top nodes according to PageRank also score high using the HITS algorithm) it is pretty surprising the fact that the top ranked node for pagerank, *oneheartartists*, scores 0(!!) in the HITS algorithm. Wow, occasions like this highlight the need for more than one algorithms that do similar jobs...

V. Conclusion

After the analysis part, some information is obtained:

- Certain users are followed more by most of the Bollywood celebrities.
- These celebrities are following more people and pages of their individual interests.
- Each of the celebrities follow very few people from Bollywood.

These are just sum-ups and all conclusions have a suggestive character. Another conclusion to draw is that major users are more important than minor users in the network, due to the variety of connections they offer, but the minor users still have to exist in the network, maintaining supplementary connections.

VI. References

[1] Gephi Wiki, URL: https://github.com/gephi/gephi/wiki

[2] Gephi-Tutorial-Layouts, URL: https://gephi.org/tutorials/gephi-tutorial-layouts.pdf

[3] Jatin Bhasin, Graph Analytics-Introduction and concepts of centrality, URL: https://towardsdatascience.com/graph-analytics-introduction-and-concepts-of-centrality-8f5543b55de3

[4] Neighbourhood-based bridge node centrality tuple for complex network analysis, URL: https://appliednetsci.springeropen.com/articles/10.1007/s41109-021-00388-1

[5] Gephi tutorial, URL: https://gephi.org/tutorials/gephi-tutorial-quick_start.pdf

[6] The slides in SNA subject, DMST AUEB, URL: https://edu.dmst.aueb.gr/course/view.php?id=24