ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Departemnt of Management Science and Technology

Athens - February 2022

Twitter User Network Analysis

Nikoalaos Kakonas

supervised by
Dimitrios Pournarakis

Abstract

In this project, I address the problem of analyzing a network and its various properties. The network data is presented and a graphical representation is provided. Through this project we explore the basic topological properties, such as component measures, degree measures, and centrality measures, to gain a deeper understanding of the network. By studying these properties, we tackle the issue of understanding the size of connected components, degree distribution, in-degree and out-degree distributions, weighted degree distribution, betweenness centrality, closeness centrality, eigenvector centrality, clustering effects, triangles, triadic closure, bridges, gender, homophily, graph density, and community structure. The approach effectively provides a comprehensive analysis of the network and its various properties.

Contents

1 Introduction **5**

2 Data Presentation **5**

3 Graphical Representation of the Network **7**

4 Basic Topological Properties **9**

5 Component Measures **10**

5.1 Connected Components 10

5.2 Component Size Distribution 10

5.3 Giant Component 10

6 Degree Measures **12**

6.1 Average Degree Distribution 12

6.2 Degree Distribution 12

6.3 In-Degree Distribution 14

6.4 Out-Degree Distribution 15

6.5 In-Degree vs Out-Degree 17

6.6 Weighted Degree Distribution 17

7 Centrality Measures **22**

7.1 Degree Centrality 22

7.2 Betweenness Centrality 26

7.3 Closeness Centrality 29

7.4 Eigenvector Centrality 32

8 Clustering Effects **36**

8.1 Clustering Coefficient 36

8.2 Triangles 37

8.3 Triadic Closure 38

9 Bridges and Local Bridges 41

10 Gender 42

11 Homophily 42

12 Graph Density 44

13 Community Structure (modularity) 45

14 PageRank 47

15 Integrated Graphical Representation of the Network 50

1 Introduction

The widespread availability of news content on the social media platform, Twitter, has made it a crucial platform for many leading news organizations, including the BBC News. The BBC News has established a strong online presence on the platform, attracting a significant number of tweets related to the organization. Given the prominence of this network, it is imperative to conduct a comprehensive analysis of its basic properties to identify the key players who have the most influence, significance, and proximity to the BBC News. This analysis will provide important insights into the workings of the network and the interactions between its constituents

2 Data Presentation

This project utilizes the Twitter Streaming API V2 to analyze the social network related to the BBC News organization. The hashtag #BBCNews was employed to collect data and construct the network graph, which exclusively focuses on user interactions on the platform. The user network was utilized as the network logic, and the relationships between users were categorized into four types: follow, retweet, mention, or reply. This can be observed from the following figure.

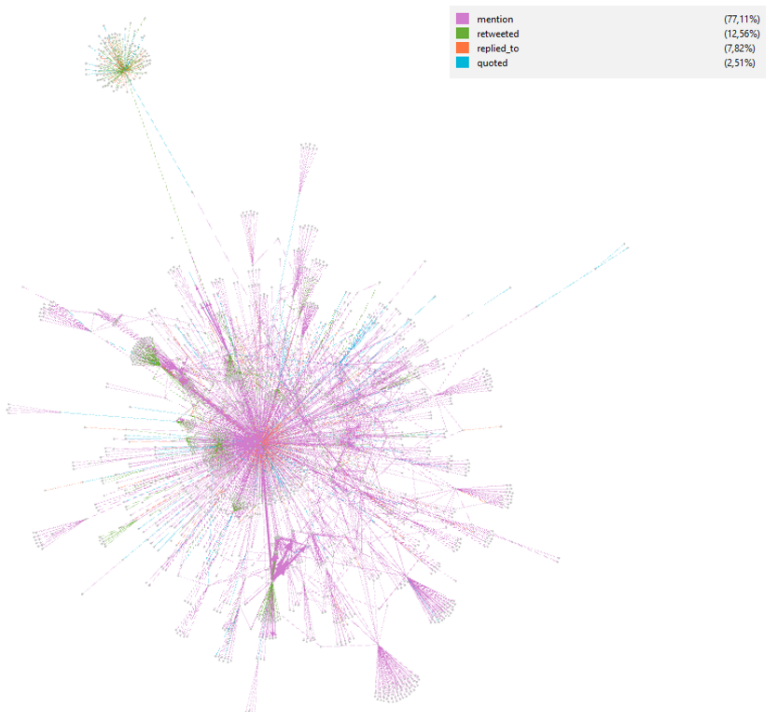


Figure 1: Type of relationship between the nodes

The following table is showing a part of the nodes dataset:

Id	Label	In-Degree	Out-Degree	Degree	Betweenness Centrality	PageRank	Modularity Class
1332485546111692801	@hilina21	0	11	11	0.0	0.000202	85
1491386148308107266	@letina241	78	19	97	588.0	0.002422	85
51241574	@ap	20	0	20	0.0	0.000628	85
14159148	@un	33	0	33	0.0	0.000894	85
612473	@bbcnews	1208	0	1208	0.0	0.095873	3
759251	@cnn	30	0	30	0.0	0.000876	85
1003895325025865730	@josepborrellf	18	0	18	0.0	0.00059	85
189868631	@drtedros	18	0	18	0.0	0.00059	85
2195671183	@msf	18	0	18	0.0	0.00059	85
380648579	@afp	19	0	19	0.0	0.000598	85
1349149096909668363	@potus	25	0	25	0.0	0.000774	85
1171334091637235714	@janezlenarcic	1	0	1	0.0	0.000317	85
1422573763682439168	@rohanmurudkar4	0	4	4	0.0	0.000202	1
19701628	@bbc	44	0	44	0.0	0.002158	28
372754427	@bbchindi	18	0	18	0.0	0.000919	1
1134434846	@gudduk_mandal	9	3	12	5.0	0.000425	1
742143	@bbcworld	50	0	50	0.0	0.002332	1
442782547	@bbcindia	13	0	13	0.0	0.000913	1
2888577619	@shrinivasaraob	0	3	3	0.0	0.000202	2
73364044	@keveeyes	161	2	163	55.5	0.004054	2
18839785	@narendramodi	122	0	122	0.0	0.00802	2
107070831	@unnidev	0	3	3	0.0	0.000202	3
31135086	@ikabirbedi	303	1	304	7.0	0.010575	3
1384473359132491787	@abdulpuncturew5	0	12	12	0.0	0.000202	55
1501772314722983936	@ashwinisahaya	2	0	2	0.0	0.000217	55
22869693	@schmittnyc	1	0	1	0.0	0.000217	55
20545835	@newsmax	3	0	3	0.0	0.000234	55
34228791	@unwomenuk	1	0	1	0.0	0.000217	55
10302222	@unicef_uk	1	0	1	0.0	0.000217	55
17537467	@tarekfatah	1	0	1	0.0	0.000217	55
19985444	@jihadwatchrs	1	0	1	0.0	0.000217	55
1585589059	@unwomenchief	1	0	1	0.0	0.000217	55
950824252562903040	@unicefchief	1	0	1	0.0	0.000217	55
428333	@cnnbrk	11	0	11	0.0	0.000392	1
44207644	@caherciveen	0	6	6	0.0	0.000202	4
1239526746724151296	@jacobbe79601492	58	6	64	123.0	0.001593	4
36083998	@omnisis	29	0	29	0.0	0.001645	4
14476016	@adambienkov	1	0	1	0.0	0.00044	4

Figure 2: Nodes

And lastly the following table shows an example from the edges dataset:

Source	Target	Type	Kind	Id	Label	Weight
1004412396612407296	73364044	Directed	mention	3932		1.0
1004412396612407296	18839785	Directed	mention	3933		1.0
1004412396612407296	73364044	Directed	retweeted	3934		1.0
100496083	31135086	Directed	mention	1472		1.0
100496083	612473	Directed	mention	1473		1.0
100496083	31135086	Directed	retweeted	1474		1.0
100540305	31135086	Directed	mention	2318		1.0
100540305	612473	Directed	mention	2319		1.0
100540305	31135086	Directed	retweeted	2320		1.0
1006897311908126722	742143	Directed	mention	1339		1.0
1006897311908126722	612473	Directed	mention	1340		1.0
1006897311908126722	786764	Directed	mention	1341		1.0
1007209462585872384	1274615546118635520	Directed	mention	4192		1.0
1007209462585872384	612473	Directed	mention	4193		1.0
1007209462585872384	1274615546118635520	Directed	retweeted	4194		1.0
101340404	1239526746724151296	Directed	mention	4668		1.0
101340404	36083998	Directed	mention	4669		1.0
101340404	612473	Directed	mention	4670		1.0
101340404	1239526746724151296	Directed	retweeted	4671		1.0
1013701498214277120	19701628	Directed	mention	4407		1.0
1013701498214277120	21866939	Directed	mention	4408		1.0
1013701498214277120	612473	Directed	mention	4409		1.0
1013701498214277120	14569869	Directed	mention	4410		1.0
1013701498214277120	358204197	Directed	mention	4411		1.0
1013701498214277120	14281853	Directed	mention	4412		1.0
1013701498214277120	14291684	Directed	mention	4413		1.0
1013701498214277120	14700117	Directed	mention	4414		1.0

Figure 3: Edges

3 Graphical Representation of the Network

Below are some visual representations of the network using different layouts:

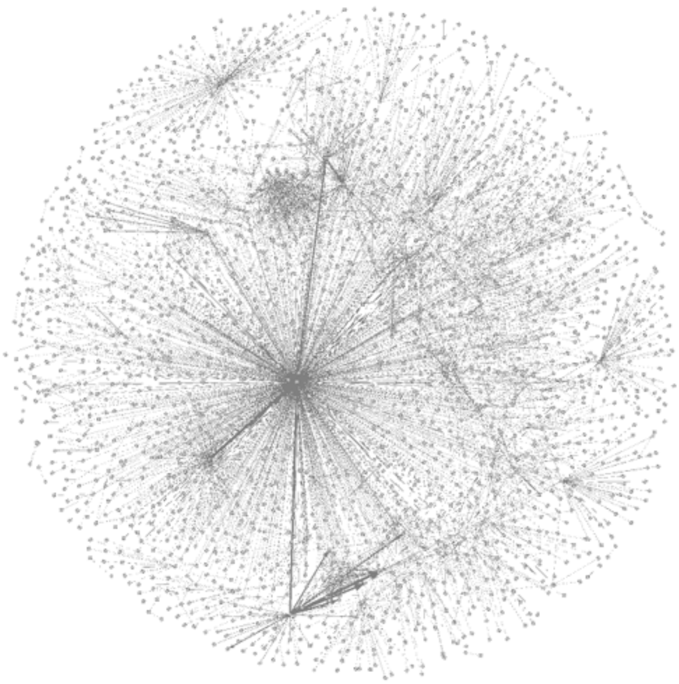


Figure 4: Fruchterman Reingold

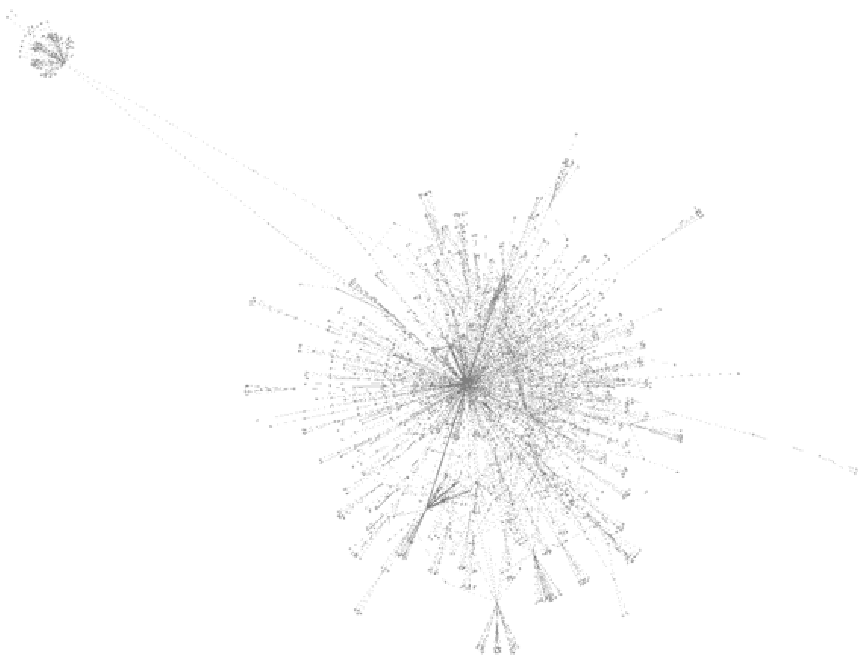


Figure 5: Yifan Hu's

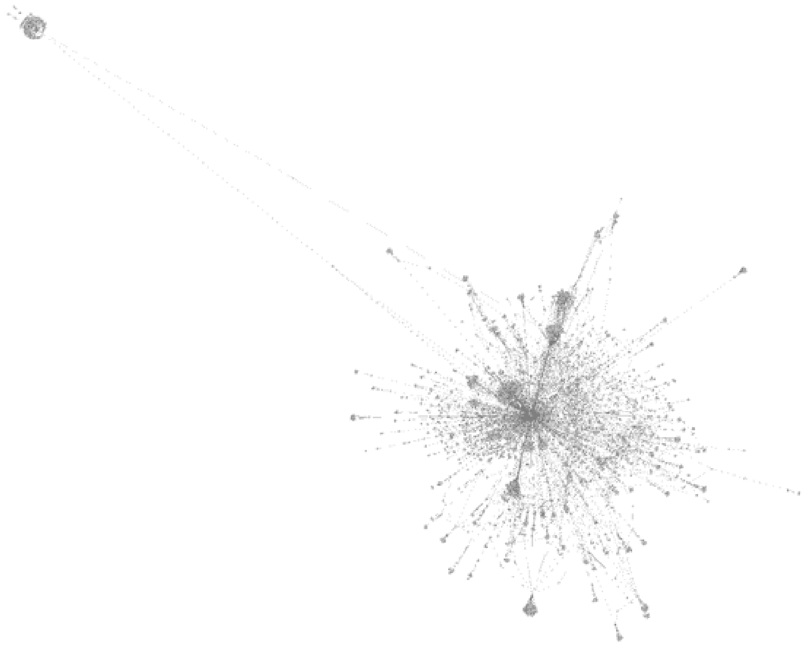


Figure 6: Force Atlas 2

Figure 7: OpenOrd

4 Basic Topological Properties

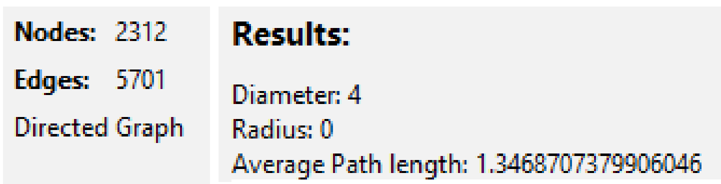
Some important network properties are the number of nodes, the number of edges, the Network Diameter, and the Average Path Length.

- Number of Nodes:
  In total, the network consists of 2312 nodes, each of which represents a Twitter user who has tweeted with the Hashtag BBCNews.

- Number of Edges:
  Accordingly, the network consists of 5701 edges, which show the connections between nodes. The network is a directed graph, where each node has edges that point to it and edges that start from it and point to another node. More specifically, the node that the edge is pointing to is known as the "target" node or "end" node, while the node that the edge originates from is known as the "source" node or "start" node. The size of an edge (thickness) can be used to represent the strength or weight of the relationship between the two nodes. For example, a bigger edge might indicate a stronger relationship, such as a higher frequency of retweets, while a smaller edge might indicate a weaker relationship, such as infrequent retweets. In this user network, the edges represent the retweets between users (nodes of the graph).

- Network Diameter:
  The network diameter is a measure of the size of a network. It is defined as the longest shortest path between any two nodes in the network. In other words, it is the maximum distance between any two nodes in the graph. The diameter represents the maximum separation between the nodes in the network and it is a useful metric for understanding the overall connectivity and structure of a network. The diameter of this network is 4, which is not a lot.

- Average Path Length:
  The average path length is a measure of the average distance between any two nodes in a network. It is calculated by summing up the shortest paths (i.e., the number of edges in the shortest path) between all pairs of nodes in the network, and then dividing by the total number of pairs. It is a measure of how "close" the nodes in a network are to each other, on average. A smaller average path length indicates that the nodes in a network are more closely connected to each other, while a larger average path length indicates that the nodes are more spread out. The average path length of this network is almost 1.35, which means that in this network, the nodes might be closely connected.



5 Component Measures

Component measures refer to metrics that are used to evaluate specific aspects of a network structure, such as the size of the network. These metrics provide a detailed understanding of the network structure and help identify key nodes and relationships within the network.

5.1 Connected Components

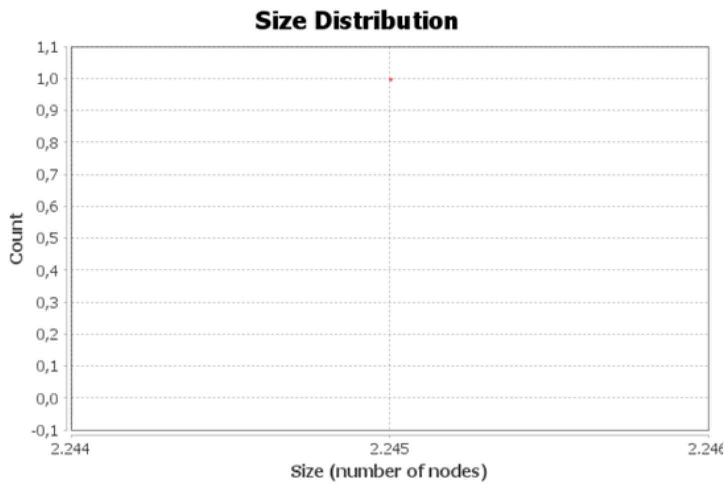
A connected component in a graph is a subgraph in which every two vertices are connected to each other by a path, and which is connected to no other vertices outside the subgraph. In other words, a connected component is a group of nodes in a graph that are all connected to each other, but not connected to any other nodes in the graph. There can be one or multiple connected components in a graph.

Results:

Number of Weakly Connected Components: 1
Number of Strongly Connected Components: 2201

5.2 Component Size Distribution

Component size distribution refers to the distribution of the sizes of the components within a network. A component is a group of nodes that are connected to each other, but not to any other nodes outside the group. The size of a component is defined as the number of nodes within the component. The number of components in BBC user network is 2245 as the following diagram shows.



5.3 Giant Component

The giant component is the largest connected component of a graph, which means it is a subgraph of a network where every node is connected to at least one other node through a path. The concept of a giant component is important because it represents the largest group of nodes that are all connected to each other, and it is a way to measure the connectivity of a network. A network can have multiple connected components, some of which may be very small. The giant component, on the other hand, is the largest connected component and it typically includes a large portion of the nodes and edges in a network. This means that it is a good representation of the overall structure and connectivity of the network. In the BBC network the giant componetn consists of 2268 nodes which is the 97.1% of the total number of the nodes of the network. The giant component is shown below.

Figure 8: Giant Component

6 Degree Measures

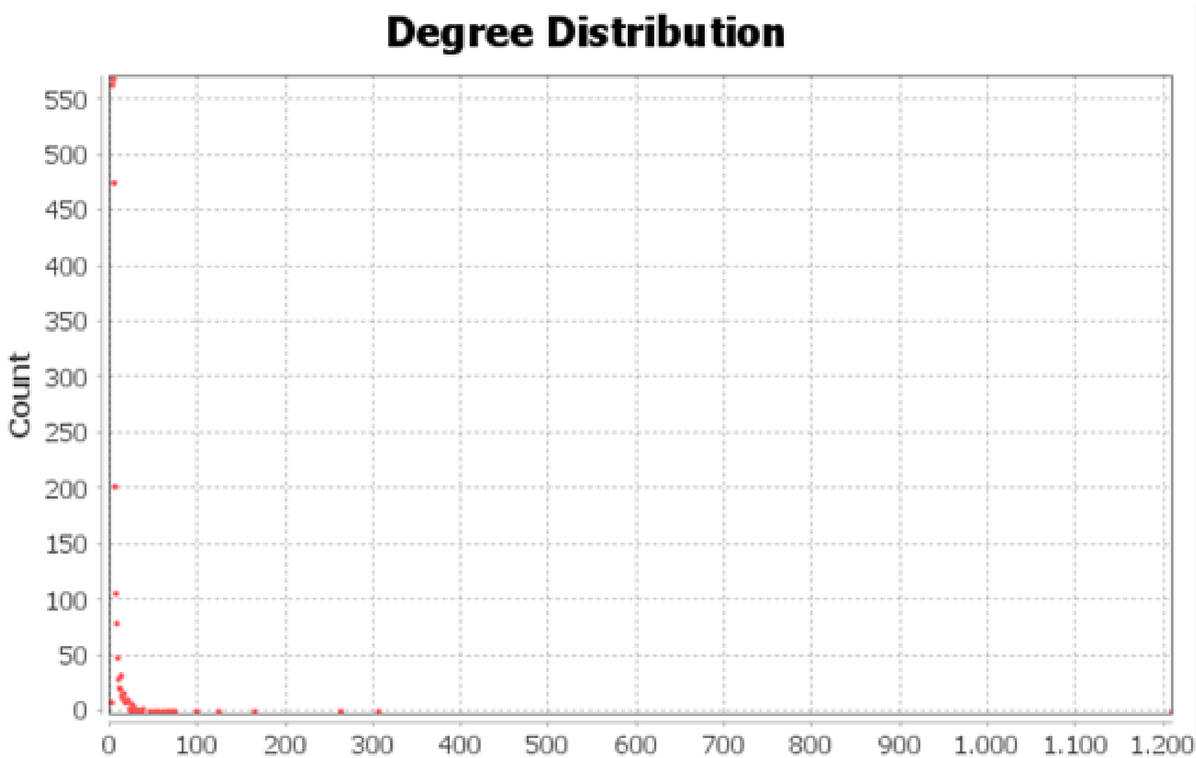
In the field of network analysis, degree measures serve as a means of quantifying the connectivity of nodes within a network. There exist various types of degree measures that aid in analyzing the structure and properties of the network, including but not limited to degree, in-degree, out-degree, weighted degree, and clustering coefficient. These measures provide valuable insight into the relationships between nodes and contribute to a deeper understanding of the underlying structure of the network.

6.1 Average Degree Distribution

The Average Degree of this network is 2.466, which means that every node of the network is connected with 2 other nodes in average. We can see that also in the following snapshot.

6.2 Degree Distribution

Degree distribution refers to the distribution of the degrees of the nodes in a network. The degree of a node is defined as the number of edges connecting to it, and it reflects the node's level of connectivity within the network. The degree distribution describes how many nodes have a certain degree and can be represented graphically as it is shown below.



Degree distribution is a key concept in network analysis, as it provides information about the connectivity of the nodes in a network. For example, a network with a few highly connected nodes and many poorly connected nodes is called a scale-free network and is typically observed in networks with a high degree of heterogeneity in the connectivity of nodes. On the other hand, a network with a more uniform distribution of node degrees is called a random network and is typically observed in networks with a more homogeneous distribution of node connections.

We could say that BBC is a random network, as it seems to have a more homogeneous distribution of node connections.

In fact, we can also represent the network based on the degree of each node, using its degree as the size of each node. We simultaneously use the Yifan Hu layout, and the following representation is obtained:

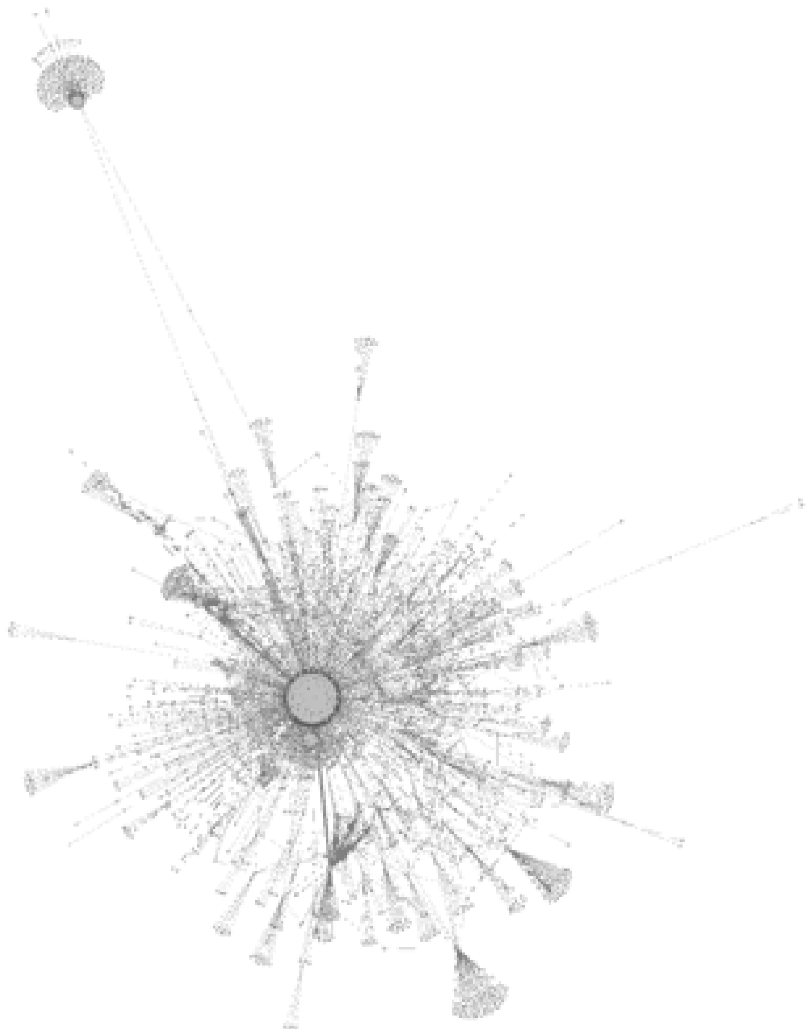


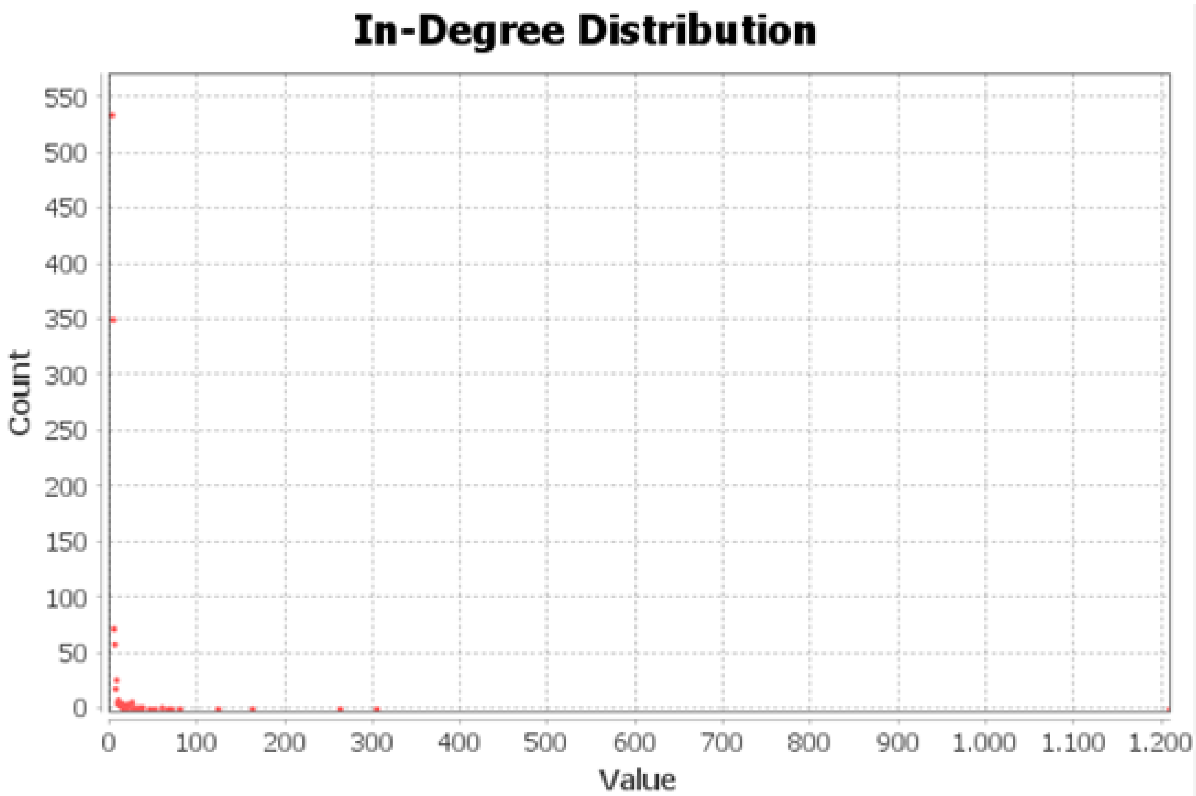
Figure 9: Degree

It is evident that the node with the highest degree is @bbcnews, with a degree of 1208, as depicted in the following illustration.

Id	Label	Degree
612473	@bbcnews	1208
31135086	@ikabirbedi	304
403670053	@timcook32	261
73364044	@keveeyes	163
18839785	@narendramodi	122
1491386148308107266	@letina241	97

6.3 In-Degree Distribution

In-degree distribution refers to the distribution of the in-degree of nodes in a directed network. In-degree is a measure of the number of incoming edges a node has in a directed network. In other words, it is the number of edges pointing towards a node. The in-degree distribution describes how many nodes have a certain in-degree and can be represented graphically as it is shown below.



In-degree distribution provides information about the number of incoming edges to each node in a directed network and can help to understand the overall flow of information or influence within the network. For example, nodes with high in-degree are typically considered to be central or influential in the network, as they receive information or influence from many other nodes. On the other hand, nodes with low in-degree are typically considered to be peripheral or less influential.

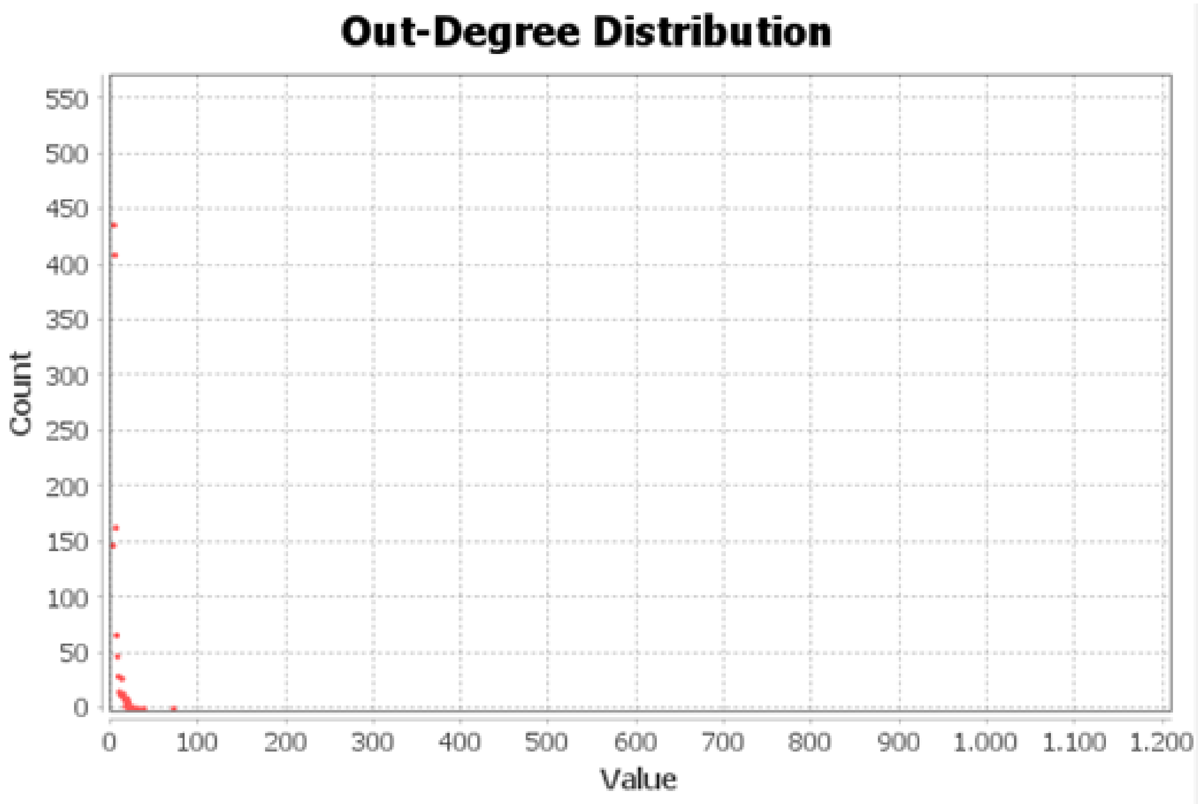
In BBC news network it seems that, there are a lot of nodes that have a degree between 0 and 100 and very little that have a degree of about 350 or 540.

The graphical representation of the network, in which the size of the nodes is based on the in-degree of each node, is similar to the representation in which the size of the nodes is based on the degree of each node, as we saw above.

Id	Label	In-Degree
612473	@bbcnews	1208
31135086	@ikabirbedi	303
403670053	@timcook32	261
73364044	@keveeyes	161
18839785	@narendramodi	122
1491386148308107266	@letina241	78

6.4 Out-Degree Distribution

In network analysis, the out-degree distribution refers to the distribution of the number of outgoing edges or connections from nodes in a network. In other words, it describes the frequency at which nodes in a network have a certain number of outgoing edges. Out-degree is a measure of the connectivity of a node to other nodes in the network, and the out-degree distribution can provide insights into the structure and organization of the network, such as the presence of hubs or central nodes with high out-degree and the presence of isolated nodes with low out-degree. This information can be used to identify important nodes and relationships within the network, as well as to analyze the distribution of influence, power, or other properties of interest in the network.



As we can see there are more nodes with similar and high out-degree value. Perhaps the first node with an out-degree of 70 stands out a bit.

Id	Label	Out-Degree
1600558512739713024	@oobejuan	70
1566141474567331843	@jos_melo2	36
1188916131937034240	@djfury4412	30
734106672570990592	@wacky_woo2	27
896551919870648321	@surmeenegi04	25
378566172	@drakeslayer100	23
1387474726151282689	@sarahlgates1	22
3014045251	@rahul_socialist	22
1514999671168536580	@abhibrijsinghji	21
1425832379554222092	@sunnyyy58001113	20
1933374348	@s97nishu	20
133897396	@sanjivpimple	20
1491386148308107266	@letina241	19

In fact, this can be represented graphically using the out-degree of each node as its size. We simultaneously use the Yifan Hu layout, and the following representation is obtained:

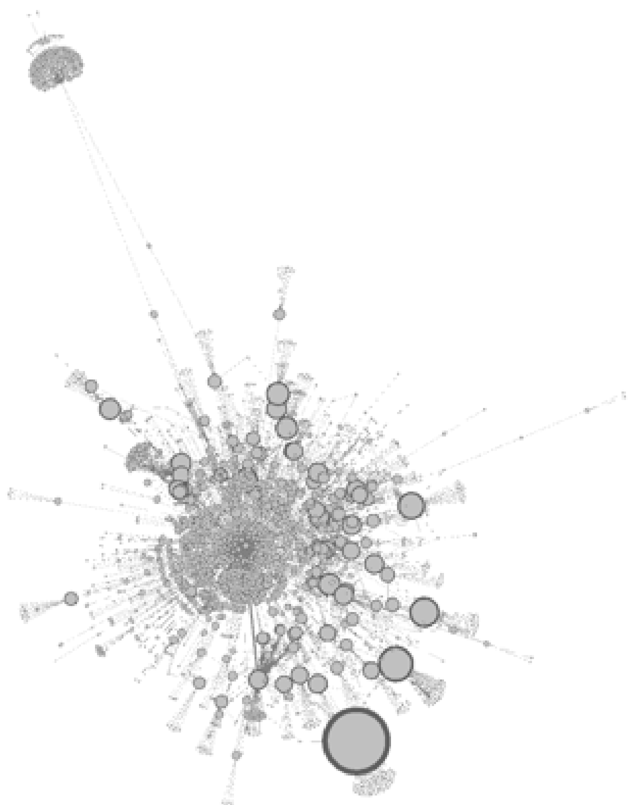


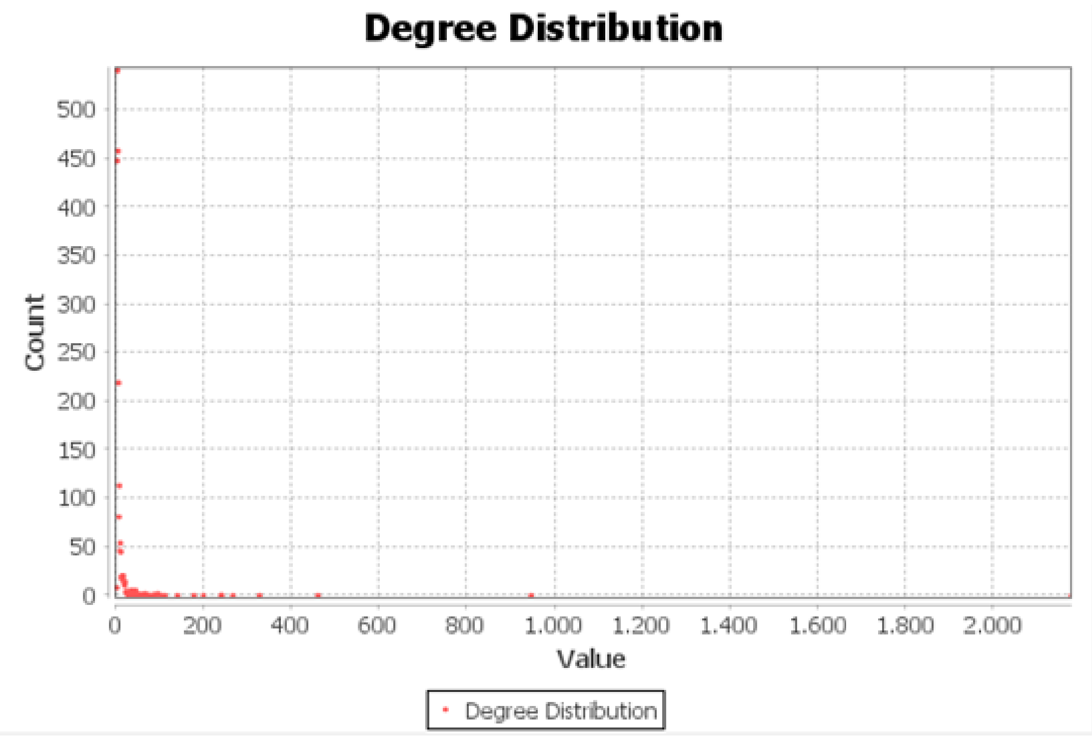
Figure 10: Out-Degree

6.5 In-Degree vs Out-Degree

We note that the two representations of the network, with node sizes based on in-degree and out-degree, exhibit significant differences. The representation based on out-degree features numerous nodes with larger sizes compared to the majority of nodes in the network, while the representation based on in-degree only has a few nodes with larger sizes.
A network with many nodes having a high out-degree and few nodes with a high in-degree often indicates the presence of a highly centralized network structure, with a few central or hub nodes that have a large number of outgoing connections and many other nodes that have fewer incoming connections. In this network, the hub nodes serve as intermediaries that connect many other nodes to one another, and these hub nodes typically have a high degree of influence, power, or control over the network. This type of structure is very logical to exist in this particular network as it is a news network and many nodes act as hubs.

6.6 Weighted Degree Distribution

Weighted degree distribution is a measure used to describe the distribution of the total weight of the edges attached to a node. Unlike the traditional degree distribution, which simply counts the number of edges connected to a node, the weighted degree distribution considers the strength of the connections between nodes. The weight of an edge may reflect the strength of the relationship between the nodes it connects, the frequency of interactions between the nodes, or other factors relevant to the network being analyzed. The weighted degree distribution provides valuable information about the structure of a network and can be used to identify key nodes or clusters within the network.
Below we can see again the weighted average degree, weighted degree, weighted in-degree and weighted out-degree of the network, each graph of these three and a comparison between weighted and non-weighted degrees.



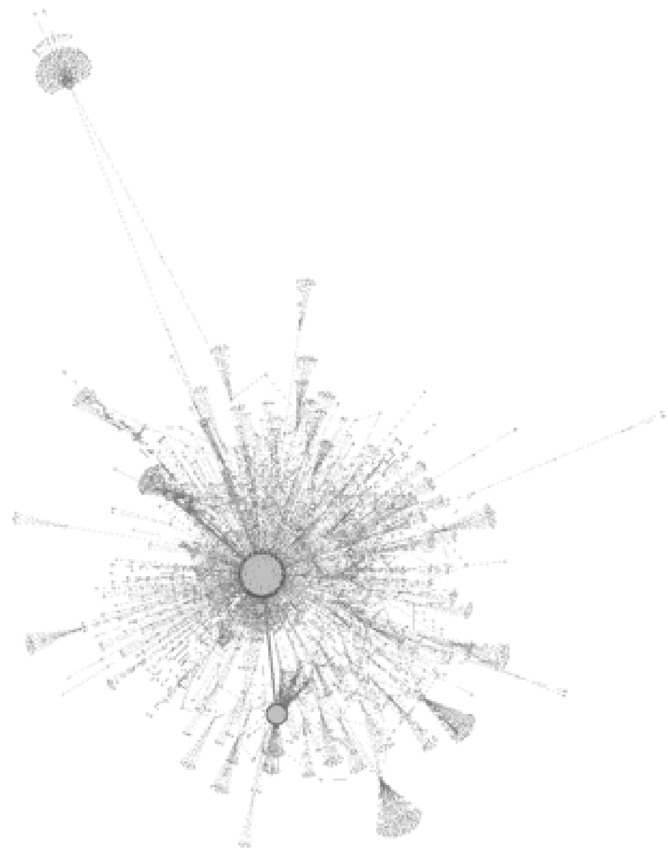
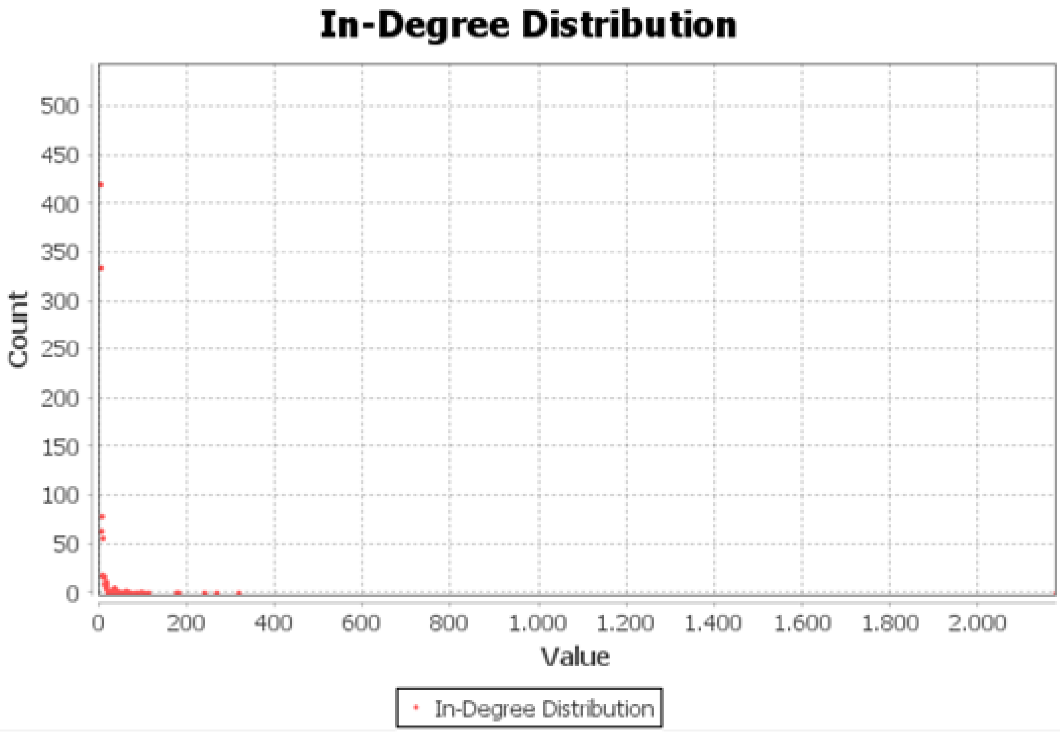


Figure 11: Weighted Degree

Id	Label	Degree	Weighted Degree
612473	@bbcnews	1208	2174.0
1491386148308107266	@letina241	97	944.0
31135086	@ikabirbedi	304	458.0
73364044	@keveeyes	163	325.0
403670053	@timcook32	261	264.0



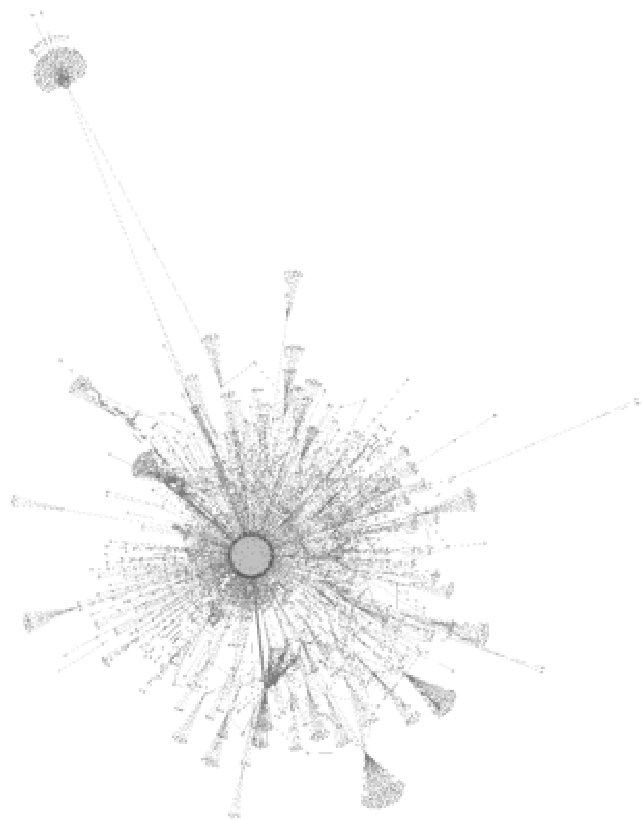
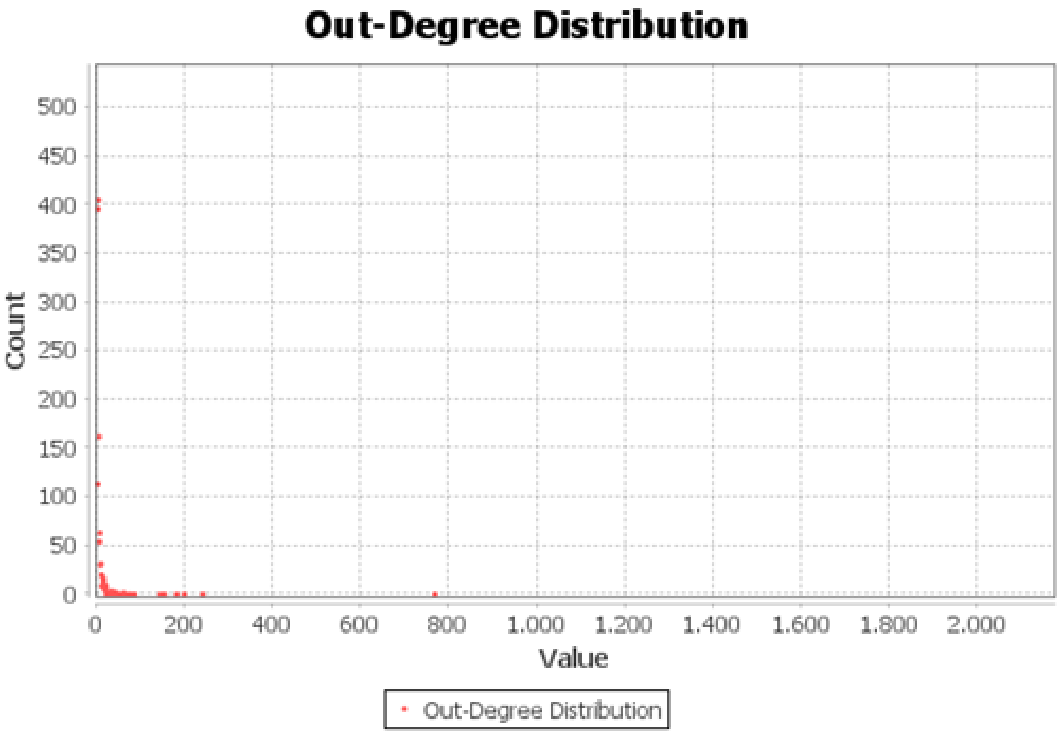


Figure 12: Weighted In-Degree

Id	Label	In-Degree	Weighted In-Degree
612473	@bbcnews	1208	2174.0
31135086	@ikabirbedi	303	315.0
403670053	@timcook32	261	264.0
18839785	@narendramodi	122	237.0
1491386148308107266	@letina241	78	178.0

The analysis of the network reveals that the node with the highest degree and in-degree centrality is the @bbcnews node. This can be attributed to the fact that the hashtag BBCNews was used as a criterion for collecting the data, causing many nodes in the network to have a direct connection to this node. As a result, the @BBCNews node has the largest number of connections or ties to other nodes in the network, making it a central node in the network with high degree and in-degree centrality.



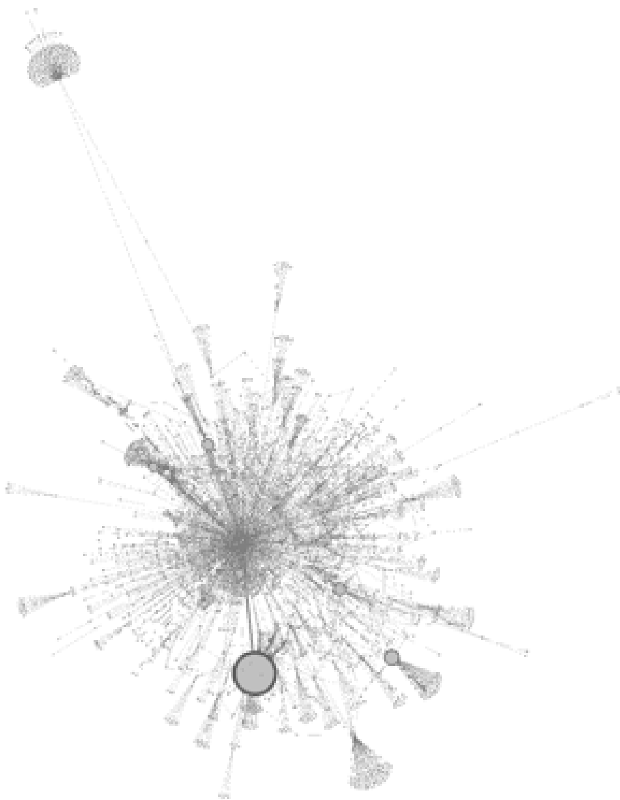


Figure 13: Weighted In-Degree

Id	Label	Out-Degree	Weighted Out-Degree
1491386148308107266	@letina241	19	766.0
1566141474567331843	@jos_melo2	36	239.0
1572608925626671105	@mehraban3356	19	197.0
1239526746724151296	@jacobbe79601492	6	180.0
73364044	@keveeyes	2	152.0
31135086	@ikabirbedi	1	143.0

7 Centrality Measures

Centrality measures are metrics used to quantify the importance or significance of nodes in a network. These measures are used to identify key nodes in a network, as well as to reveal the nodes with big centrality in the network. There are several different types of centrality measures, including: degree centrality, betweenness centrality, closeness centrality, and eigenvector centrality. Each of these measures provides a different perspective on the importance of nodes in a network.

7.1 Degree Centrality

Degree centrality measures the number of connections a node has to other nodes in the network. It is a simple and straightforward metric that reflects the importance of a node in terms of its connections to other nodes. A node with a high degree centrality has many connections to other nodes and is considered to be a highly connected or influential node in the network. Degree centrality is typically used to identify key players, gatekeepers or key nodes in the network that play a central role in connecting and spreading information between other nodes. It is one of the most basic centrality measures, and it provides a simple, intuitive way of quantifying the centrality of a node in a network.

In the previous section, the degree of each node was thoroughly analyzed, i.e. the number of edges it has, whether these are edges that start from the specific node, or whether they are edges that end at the specific node. Furthermore, the in-degree, which represents the number of edges ending at a specific node, and the out-degree, which represents the number of edges originating from a specific node, were also calculated and analyzed. In this section we need to see the importance of each node in terms of its connections to other nodes as already mentioned.

7.1.1 Degree

Starting from the degree we can observe that only the node @bbcnews has a significant number of connections. The following eight nodes also have a relatively high number of connections. As we move down the list of nodes, the degree decreases and the number of nodes with that degree increases. This indicates a concentration of connections around the node @bbcnews, with fewer connections as we move away from that node. A general observation can be made that the first ten nodes in the network hold significant importance in terms of the dissemination of information throughout the network. This is because these nodes, with their high degree, are well connected to the rest of the network and serve as key intermediaries for the flow of information. The high degree of these nodes suggests that they are central to the network structure and play a crucial role in ensuring that information reaches a large number of other nodes in the network.

Id	Label	Degree
612473	@bbcnews	1208
31135086	@ikabirbedi	304
403670053	@timcook32	261
73364044	@keveeyes	163
18839785	@narendramodi	122
1491386148308107266	@letina241	97
3395584851	@profnfenton	72
1600558512739713024	@oobejuan	70
1441683373840224264	@battakashmiri	66
1239526746724151296	@jacobbe79601492	64
1129737385837715456	@mjavinod	59
1387474726151282689	@sarahlgates1	53
742143	@bbcworld	50
19701628	@bbc	44
7587032	@skynews	36
207809313	@bjp4india	36
1566141474567331843	@jos_melo2	36
1616164718	@nickdtrt	35
1425458108017565703	@soswhitstable	35
14159148	@un	33
542029876	@ramiranger	31
759251	@cnn	30
1188916131937034240	@djfury4412	30
36083998	@omnisis	29
1339166129110065152	@gbnews	28
998589355764613121	@fisi_uk	27
21866939	@itvnews	27
734106672570990592	@wacky_woo2	27
1349149096909668363	@potus	25
896551919870648321	@surmeenegi04	25
913777910065983490	@alindamjan	25
378566172	@drakeslayer100	25
2297438462	@bjp4keralam	24
1540218975463673856	@blsanthoshfans	24
906785747180470272	@damodarhegde4	24
93643538	@nrajabpcl	24
2289824455	@pallavict	24

In the following snapshot we can also see that there are nine nodes with zero degree and many of them (564 nodes, 24.39% of the network) have a degree of one.

Id	Label	Degree
760872060	@keithwarburton1	0
105812663	@txbabexx	0
41438290	@maggoo0	0
1072730010501341184	@mathajyoti	0
3382960853	@rumouredrebecca	0
1107293665687293953	@hbahrox4b7pt5nz	0
35187172	@derrymac	0
933754477244493824	@james85fletcher	0
196943559	@mrdavidgray	0
1171334091637235714	@janezlenarcic	1
22869693	@schmittnyc	1
34228791	@unwomenuk	1
10302222	@unicef_uk	1
17537467	@tarekfatah	1
19985444	@jihadwatchrs	1
1585589059	@unwomenchief	1
950824252562903040	@unicefchief	1
14476016	@adambienkov	1

7.1.2 In-Degree

An examination of the in-degree of nodes in the network reveals a close correlation with the degree of the nodes. As depicted in the screenshot below, a majority of the nodes, particularly the top-ranked ones, have similar values for both their degree and in-degree. However, there are instances where nodes with a relatively high degree compared to the other nodes in the network have an in-degree of zero. This suggests that these nodes are highly connected in terms of outgoing connections but receive few or no incoming connections from other nodes in the network.

Id	Label	Degree	In-Degree
612473	@bbcnews	1208	1208
31135086	@ikabirbedi	304	303
403670053	@timcook32	261	261
73364044	@keveeyes	163	161
18839785	@narendramodi	122	122
1491386148308107266	@letina241	97	78
3395584851	@profnfenton	72	69
1600558512739713024	@oobejuan	70	0
1441683373840224264	@battakashmiri	66	65
1239526746724151296	@jacobbe79601492	64	58
1129737385837715456	@mjavinod	59	58
1387474726151282689	@sarahlgates1	53	31
742143	@bbcworld	50	50
19701628	@bbc	44	44
7587032	@skynews	36	36
207809313	@bjp4india	36	36
1566141474567331843	@jos_melo2	36	0

As expected, the same thing happens also with some other nodes, which have zero in-degree value and a much higher degree value, as we see below.

Id	Label	Degree	In-Degree
1600558512739713024	@oobejuan	70	0
1566141474567331843	@jos_melo2	36	0
1188916131937034240	@djfury4412	30	0
734106672570990592	@wacky_woo2	27	0
896551919870648321	@surmeenegi04	25	0
3014045251	@rahul_socialist	22	0
1425832379554222092	@sunnyyy58001113	20	0
1933374348	@s97nishu	20	0
133897396	@sanjivpimple	20	0
20479891	@chrispduck	19	0
1402478042337927168	@cesaralb22	19	0
282652468	@anotherjondough	19	0
1613918751824420865	@bobstercat2000	19	0
1572608925626671105	@mehraban3356	19	0
115502965	@sakura509	19	0
1498267518250610689	@rohit333999	18	0

7.1.3 Out-Degree

We can observe that, only one node has a degree-value of 70, which is large compared to the other nodes and all the others have much smaller out-degrees.

Id	Label	Out-Degree
1600558512739713024	@oobejuan	70
1566141474567331843	@jos_melo2	36
1188916131937034240	@djfury4412	30
734106672570990592	@wacky_woo2	27
896551919870648321	@surmeenegi04	25
378566172	@drakeslayer100	23
1387474726151282689	@sarahlgates1	22
3014045251	@rahul_socialist	22
1514999671168536580	@abhibrijsinghji	21
1425832379554222092	@sunnyyy58001113	20
1933374348	@s97nishu	20
133897396	@sanjivpimple	20
1491386148308107266	@letina241	19
20479891	@chrispduck	19
1402478042337927168	@cesaralb22	19
282652468	@anotherjondough	19

Also, as expected, there are many nodes with zero value of out-degree. These nodes are 862, which is the 37% of the network. This is normal, as the network have been created with a specific focus on @BBCNews, and therefore includes many connections relevant to this node, resulting in a large number of nodes with zero out-degree.

It is also interesting to see that, nodes with high out-degree value (maximum value is 70) have the same degree but zero in-degree value.

Id	Label	Out-Degree	Degree	In-Degree
1600558512739713024	@oobejuan	70	70	0
1566141474567331843	@jos_melo2	36	36	0
1188916131937034240	@djfury4412	30	30	0
734106672570990592	@wacky_woo2	27	27	0
896551919870648321	@surmeenegi04	25	25	0
378566172	@drakeslayer100	23	25	2
1387474726151282689	@sarahlgates1	22	53	31
3014045251	@rahul_socialist	22	22	0
1514999671168536580	@abhibrijsinghji	21	23	2
1425832379554222092	@sunnyyy58001113	20	20	0

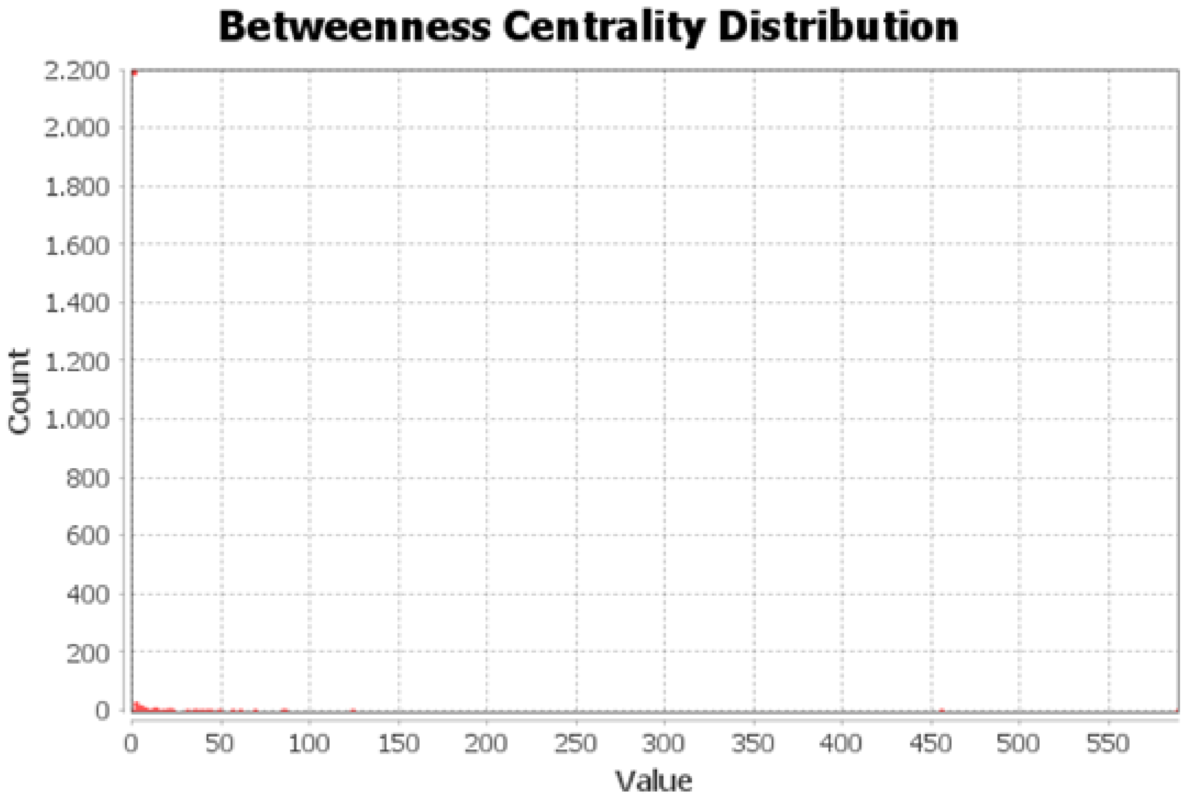
And the nodes with zero out-degree value have much bigger degree and in-degree value. More specifically, in these nodes, exists the @bbcnews node with the highest degree and in-degree value.

Id	Label	Out-Degree	Degree	In-Degree
612473	@bbcnews	0	1208	1208
403670053	@timcook32	0	261	261
18839785	@narendramodi	0	122	122
742143	@bbcworld	0	50	50
19701628	@bbc	0	44	44
7587032	@skynews	0	36	36
207809313	@bjp4india	0	36	36
14159148	@un	0	33	33
759251	@cnn	0	30	30
36083998	@omnisis	0	29	29

7.2 Betweenness Centrality

Betweenness centrality quantifies the extent to which a node lies on the shortest path between other nodes. In other words, it measures how often a node acts as a bridge or mediator between other nodes in the network. Nodes with high betweenness centrality play a critical role in the flow of information and the structural stability of the network. If these nodes were to be removed, the network would become more fragmented, and it would take more steps for information to flow between nodes that were previously directly connected.

In BBC News network we observe that the most nodes have zero betweenness centrality, which shows that many nodes are not intermediates of node pairs.



The network also reveals that @letina241 has the highest betweenness centrality, with a value of 588, followed closely by @sarahlgates1 with a value of 455. This indicates that these two nodes play a central role in the flow of information in the network, acting as key intermediaries between other nodes. It is worth noting that only 32 nodes in the network have a betweenness centrality value greater than 10, suggesting that the majority of the nodes in the network have a relatively low level of influence over the flow of information. More specifically, there are 2196 nodes that have zero betweenness centrality. This means that all these nodes do not facilitate the flow of information or influence between other nodes. This observation can be seen in the screenshot, which highlights a selection of nodes with higher and lower betweenness centrality values.

Label	Betweenness Centrality
@letina241	588.0
@sarahlgates1	455.0
@jacobbe79601492	123.0
@nickdtrt	85.0
@inder1158	84.0
@ramiranger	68.0
@profnfenton	60.0
@keveeyes	55.5
@danneidle	48.0
@anil_7118	43.0
@carolvorders	41.0
@dival1000	38.0
@bbclysedoucet	35.0
@insightuk2	34.0
@houseofchanges	30.0
@cell111right	21.333333
@rosieb2019	21.0
@hibbsv1973	20.0

Label	Betweenness Centrality
@unnidev	0.0
@abdulpuncturew5	0.0
@mangala58577842	0.0
@dipakshi_choksi	0.0
@ibharatwasi	0.0
@uk_republic	0.0
@officialpremjit	0.0
@eagle_sonja	0.0
@confuseforever	0.0
@saltydesouffle	0.0
@bilalkhalid1984	0.0
@kumarsdilip	0.0
@nitinkjain	0.0
@urs_ethically	0.0
@imanisheba	0.0
@clarabo20727319	0.0
@jcbing	0.0
@dalexviii	0.0

The mean of betweenness centrality of the network is about 0.998.

```
In [87]: bcm = df['betweenness centrality'].describe()
         bcm

Out[87]: count    2312.000000
         mean        0.998270
         std        16.213579
         min         0.000000
         25%         0.000000
         50%         0.000000
         75%         0.000000
         max       588.000000
         Name: betweenness centrality, dtype: float64
```

The visual representation of the network also provides a useful illustration of the betweenness centrality values of the nodes. The size of each node is proportional to its betweenness centrality value, with larger nodes indicating higher betweenness centrality and smaller nodes indicating lower betweenness centrality. This visualization makes it easy to identify the nodes with the greatest influence on the flow of information in the network, as well as the nodes with less impact.

Figure 14: Betweenness Centrality

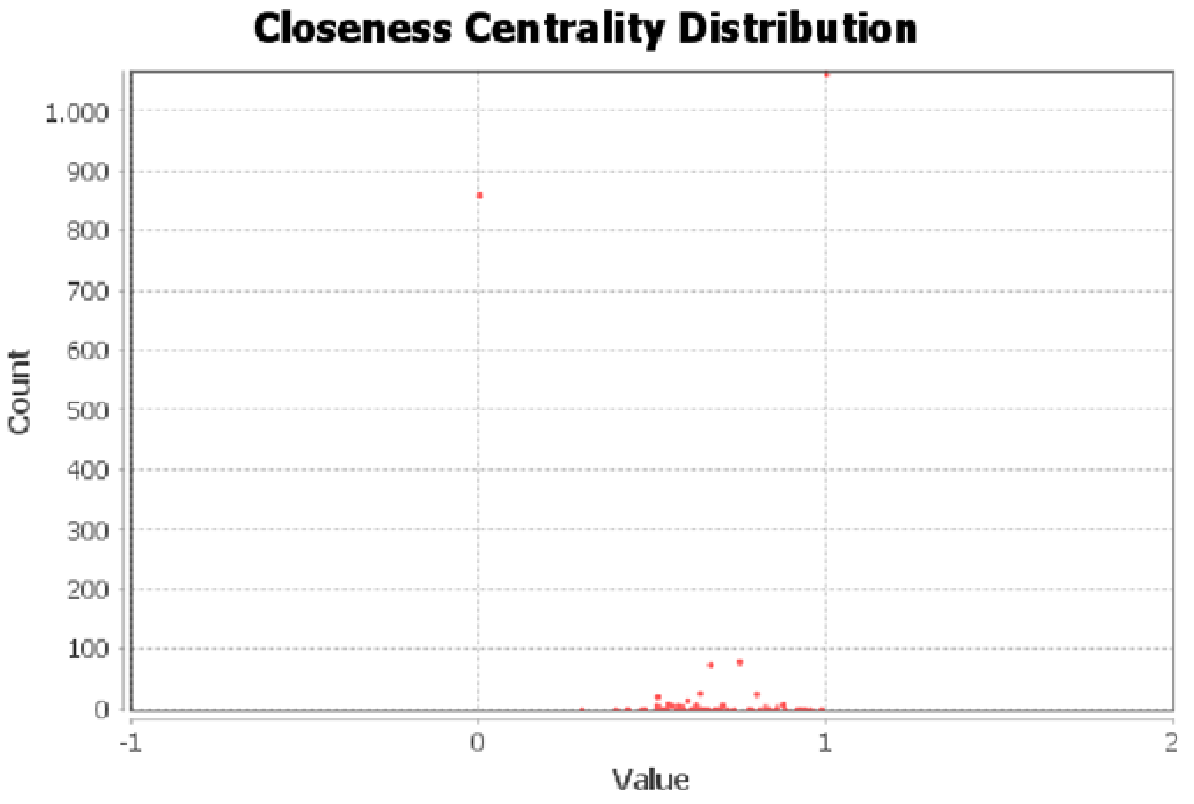
7.2.1 High betweenness centrality - @letina241

Nodes with both high betweenness centrality and high out-degree, like @letina241 can be considered as central and influential nodes in the network, as they play a crucial role in the flow of information and resources. After a quick search on Twitter for this node I found that it has to do with Tigray Defense Forces (TDF). TDF is the military force of the Tigray Regional State in Ethiopia. It was established to protect the rights, interests and the territory of the Tigray region. The owner of this account creates many tweets every day, most of the times mentions as a hashtag BBCNews and shares a lot of content. Below there is an example. These are the reasons why this node has both high betweenness centrality and high out-degree.

7.3 Closeness Centrality

Closeness centrality is a measure of a node's proximity to all other nodes in a network. It reflects the average distance between a node and all other nodes in the network, where distance is typically defined as the number of edges between two nodes. In other words, closeness centrality measures how quickly a node can reach all other nodes in the network. Nodes with high closeness centrality have a relatively small average distance to all other nodes, meaning that they are well-connected and well-positioned to transfer information or resources quickly.

Below we can see the scatter plot that Gephi creates for the closeness centrality distribution of the network.



Here we can see the top closeness centrality nodes of the list and the bottom closeness centrality nodes in the list.

Label	Closeness Centrality
@letina241	1.0
@sarahlgates1	1.0
@jacobbe79601492	1.0
@ramiranger	1.0
@profnfenton	1.0
@keveeyes	1.0
@danneidle	1.0
@bbclysedoucet	1.0
@insightuk2	1.0
@houseofchanges	1.0
@rosieb2019	1.0
@hibbsy1973	1.0
@aartitikoo	1.0
@ashutoshkashi95	1.0
@fisi_uk	1.0
@stellaogbaro	1.0
@elonmusk	1.0
@farax4life	1.0

Label	Closeness Centrality
@ap	0.0
@un	0.0
@bbcnews	0.0
@cnn	0.0
@josepborrellf	0.0
@drtedros	0.0
@msf	0.0
@afp	0.0
@potus	0.0
@janezlenarcic	0.0
@bbc	0.0
@bbchindi	0.0
@bbcworld	0.0
@bbcindia	0.0
@narendramodi	0.0
@ashwinisahaya	0.0
@schmittnyc	0.0
@newsmax	0.0

More specifically, the min of closeness centrality is 0, the max is 1 and the mean is about 0.57.

```
In [86]: ccm = df['closeness centrality'].describe()
         ccm

Out[86]: count    2312.000000
         mean        0.572793
         std         0.458253
         min         0.000000
         25%         0.000000
         50%         0.750000
         75%         1.000000
         max         1.000000
         Name: closeness centrality, dtype: float64
```

In the context of a BBC News network, a large number of nodes with a minimum closeness centrality of zero and many with a maximum closeness centrality of one can be indicative of a star-like network structure. This structure is characterized by the presence of a central node that is connected to many peripheral nodes, but the peripheral nodes are not directly connected to each other.

This network structure can arise due to the presence of a dominant source of information or influence that is connected to many other nodes in the network. The peripheral nodes may be articles, reporters, or topics that are connected to the central node, but not directly connected to each other.

The observation of a star-like structure in the BBC News network may have implications for the flow of information and influence in the network. The central node may serve as a hub for information and play a crucial role in shaping the discourse in the network. On the other hand, the peripheral nodes may be more susceptible to the influence of the central node but may have limited influence over each other. We can understand that from the screenshots below.

Also, we can see below a visual representation of the network, which provides a useful illustration of the closeness centrality values of the nodes. The size of each node is proportional to its closeness centrality value, with larger nodes indicating higher closeness centrality and smaller nodes indicating lower closeness centrality.

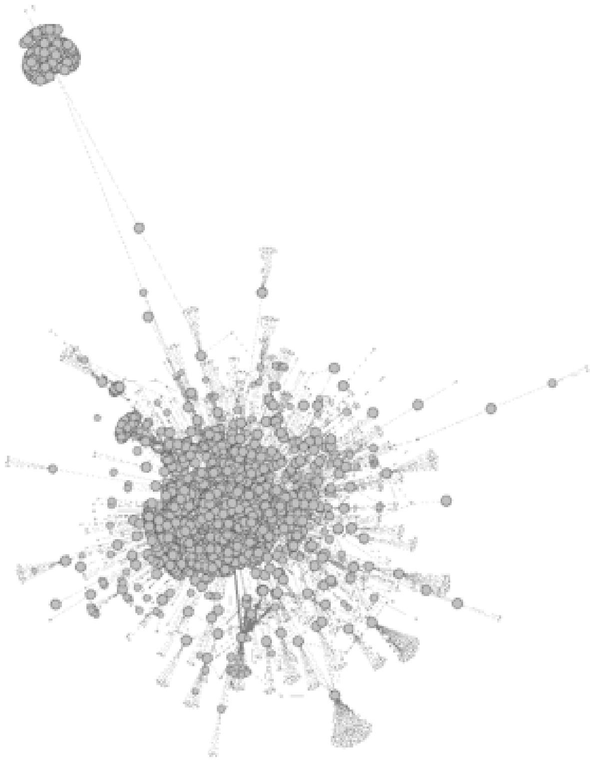
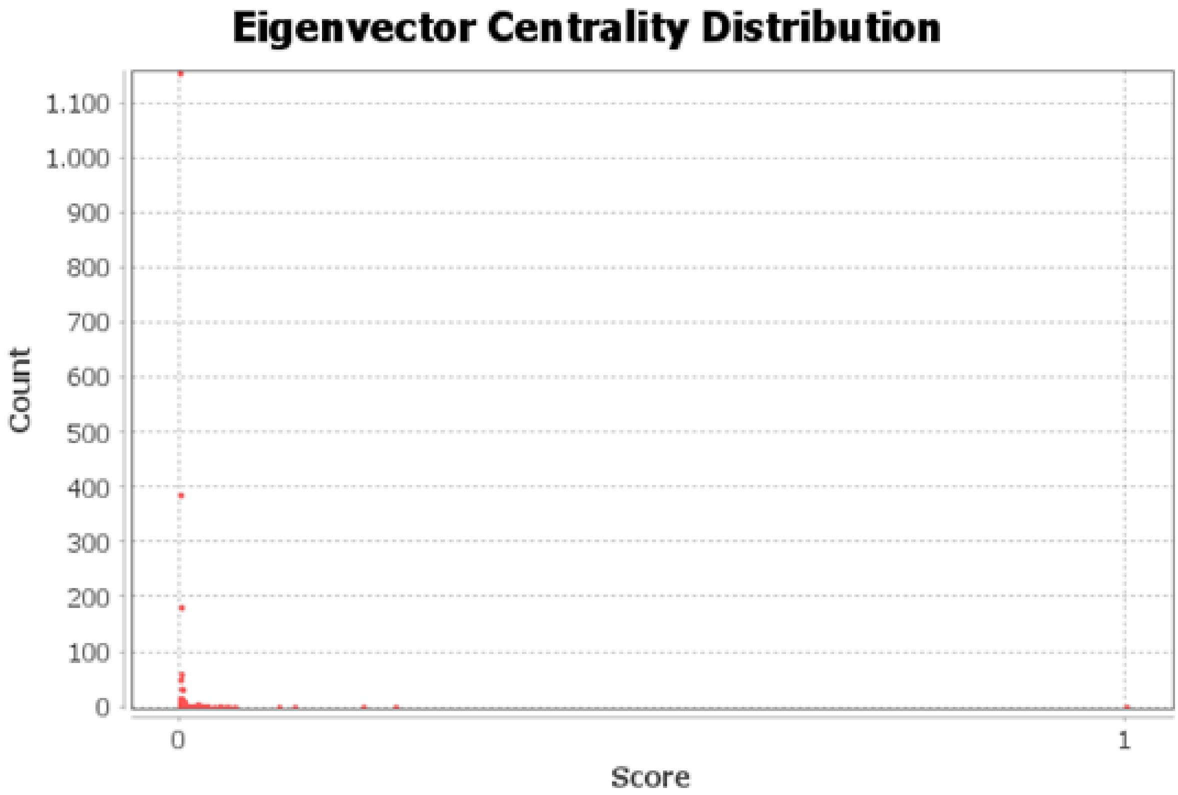


Figure 15: Closeness Centrality

7.4 Eigenvector Centrality

Eigenvector centrality is a measure of the importance of a node in a network, based on the idea that a node is important if it is connected to other important nodes. Unlike other centrality measures, such as degree centrality or betweenness centrality, eigenvector centrality considers the centrality of the nodes that a node is connected to. In mathematical terms, eigenvector centrality is the eigenvector of the adjacency matrix of a network that corresponds to the largest eigenvalue. The eigenvector centrality of a node is proportional to the sum of the centralities of its neighbors, with weights proportional to the strengths of the connections.

Below we can see the scatter plot that Gephi creates for the closeness centrality distribution of the network.



We observe here that there is only one node with a value of eigenvector centrality to one, which is the biggest value in the list. All the other nodes have a lower value from 0.23 and the majority of them has zero eigenvector centrality. The mean of eigenvector centrality is about 1.67. This can be seen in the following screenshot.

```
In [89]: df['eigenvector centrality'].describe()

Out[89]: count    2.312000e+03
         mean     1.676984e+11
         std      3.191449e+11
         min      0.000000e+00
         25%      0.000000e+00
         50%      0.000000e+00
         75%      0.000000e+00
         max      9.783618e+11
         Name: eigenvector centrality, dtype: float64
```

Below we can see the top eigenvector centrality nodes of the list and the bottom eigenvector centrality nodes in the list.

Label	Eigenvector Centrality
@bbcnews	1.0
@ikabirbedi	0.227817
@timcook32	0.1939
@keveeyes	0.12113
@narendramodi	0.104684
@letina241	0.057947
@profnfenton	0.051558
@battakashmiri	0.048663
@jacobbe79601492	0.043089
@mjavinod	0.043089
@bbcworld	0.041721
@bbc	0.036289
@un	0.029877
@bjp4india	0.029365
@skynews	0.028497
@cnn	0.027302
@omnisis	0.024888
@soswhitstable	0.024516

Label	Eigenvector Centrality
@hilina21	0.0
@rohanmurudkar4	0.0
@shrinivasaraob	0.0
@unnidev	0.0
@abdulpuncturew5	0.0
@caherciveen	0.0
@sanemind9	0.0
@dipakshi_choksi	0.0
@ibharatwasi	0.0
@uk_republic	0.0
@officialpremjit	0.0
@eagle_sonja	0.0
@confuseforever	0.0
@saltydesouffle	0.0
@bilalkhalid1984	0.0
@kumarsdilip	0.0
@nitinkjain	0.0
@urs_ethically	0.0

In BBCNews network there is a clear distinction between a single node with the highest eigenvector centrality value of 1 and all other nodes in the network which have significantly lower eigenvector centrality values ranging from 0 to 0.23. This is also clear from the scatter plot. This observation suggests a highly centralized structure in the network, where the node with the eigenvector centrality value of 1 serves as the hub and the majority of the other nodes have limited impact on the overall network structure and connectivity.

This pattern could indicate that the network has a hierarchical structure, with a few highly influential nodes and many less influential nodes. The presence of a single node with an eigenvector centrality value of 1 highlights its exceptional importance in the network, acting as a central point for information dissemination and influence. The lower eigenvector centrality values of the other nodes suggest a more peripheral role in the network, with limited direct impact on the overall network structure and connectivity.

Also, we can see below a visual representation of the network, which provides a useful illustration of the eigenvector centrality values of the nodes. The size of each node is proportional to its eigenvector centrality value, with larger nodes indicating higher eigenvector centrality and smaller nodes indicating lower eigenvector centrality.

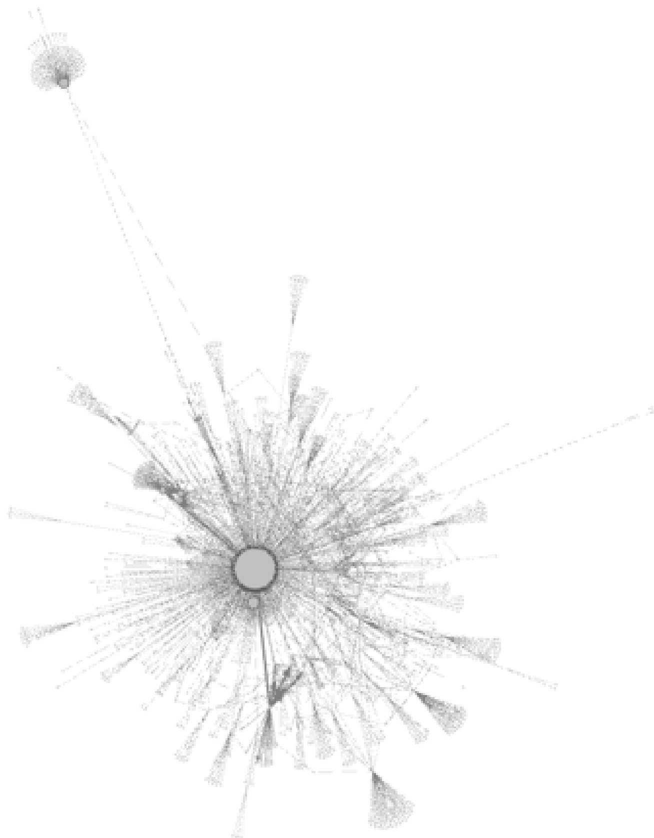


Figure 16: Eigenvector Centrality

8 Clustering Effects

Clustering effects in networks refer to the phenomenon of groups of nodes in a network being more densely connected to each other compared to the rest of the network. In other words, clusters, or communities of nodes in a network have a higher number of connections within the group compared to connections between groups.

8.1 Clustering Coefficient

The clustering coefficient of a node in a network is defined as the fraction of its neighbors that are also connected to each other. In other words, it measures the extent to which the neighbors of a node form a densely connected subgroup. The clustering coefficient can be calculated for a single node, for all nodes in a network, or for different groups of nodes in a network. It ranges from 0 to 1, where 0 indicates no clustering, and 1 indicates that all neighbors of a node are connected to each other.

Label	Clustering Coefficient
@janezlenarcic	0.0
@abdulpuncturew5	0.0
@ashwinisahaya	0.0
@schmittnyc	0.0
@newsmax	0.0
@unwomenuk	0.0
@unicef_uk	0.0

Label	Clustering Coefficient
@bbcquestiontime	1.0
@fixediti	1.0
@bloktopia	1.0
@davepowell_	1.0
@markthehibby	1.0
@scotlibdems	1.0
@construzives	1.0

The average Clustering Coefficient is 0.145. This metric shows the average value of the atomic coefficients, which highlights the probability that two randomly selected nodes are neighbors of a particular node.

More specifically, the clustering coefficient shows how well connected a node's neighborhood is and is a metric of the degree to which nodes in a graph tend to form clusters with each other. Again if the neighborhood is fully connected, then the clustering coefficient is equal to 1, while, if the value of the clustering coefficient is close to 0, then this means that there are hardly any connections within a neighborhood. In BBC News network we notice that there are some connections between the nodes of a neighborhood.

Also, we can see below a visual representation of the network, which provides a useful illustration of the clustering coefficient values of the nodes. The size of each node is proportional to its clustering coefficient value, with larger nodes indicating higher clustering coefficient and smaller nodes indicating lower clustering coefficient.

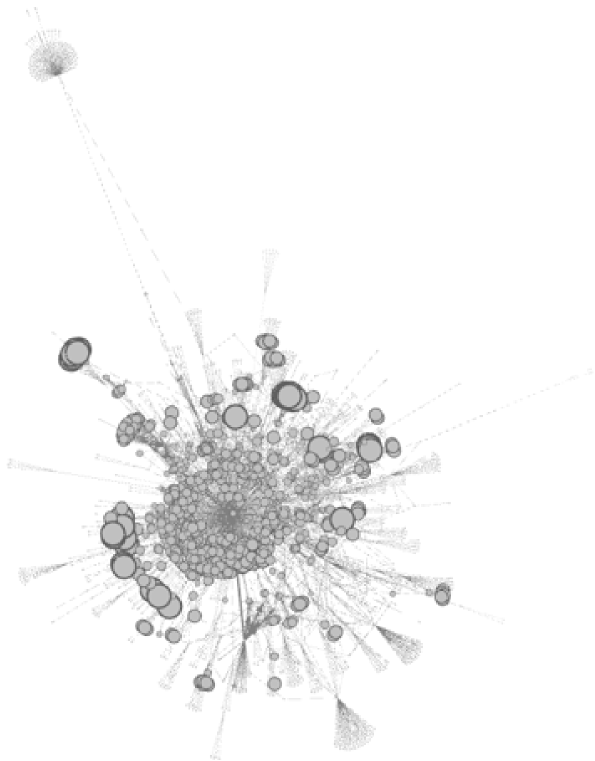


Figure 17: Clustering Coefficient

8.2 Triangles

Triangles in a network refer to groups of three nodes that are all connected to each other. In other words, if three nodes form a triangle in a network, it means that each node has a direct edge or connection to the other two nodes.

The BBC network is a directed network where each edge between nodes has a specific direction. When analyzing the network, it is possible to switch the representation from a directed graph to an undirected graph, which results in the loss of direction for the edges and the creation of bidirectional connections between nodes. This change in representation can result in the formation of triangles that were not present in the original directed network, but this is not guaranteed.

This happens because the direction of the edges in a directed graph imposes constraints on the relationships between nodes, while in an undirected graph, there are no such constraints, and all nodes are connected to all other nodes they share an edge with.

When switching to an undirected graph we observe that there are 1790 triangles. More specifically, we can see below the top nodes that create triangles with other nodes. The node with the most triangles is @bbcnews, which is the main node of the network, as we used BBCNews hashtag to collect the data.

Label	Number of triangles
@bbcnews	716
@ikabirbedi	186
@letina241	153
@keveeyes	131
@narendramodi	117
@alindamjan	106
@nickdtrt	80

Below we can also see the visual representation of the graph based on the number of triangles.

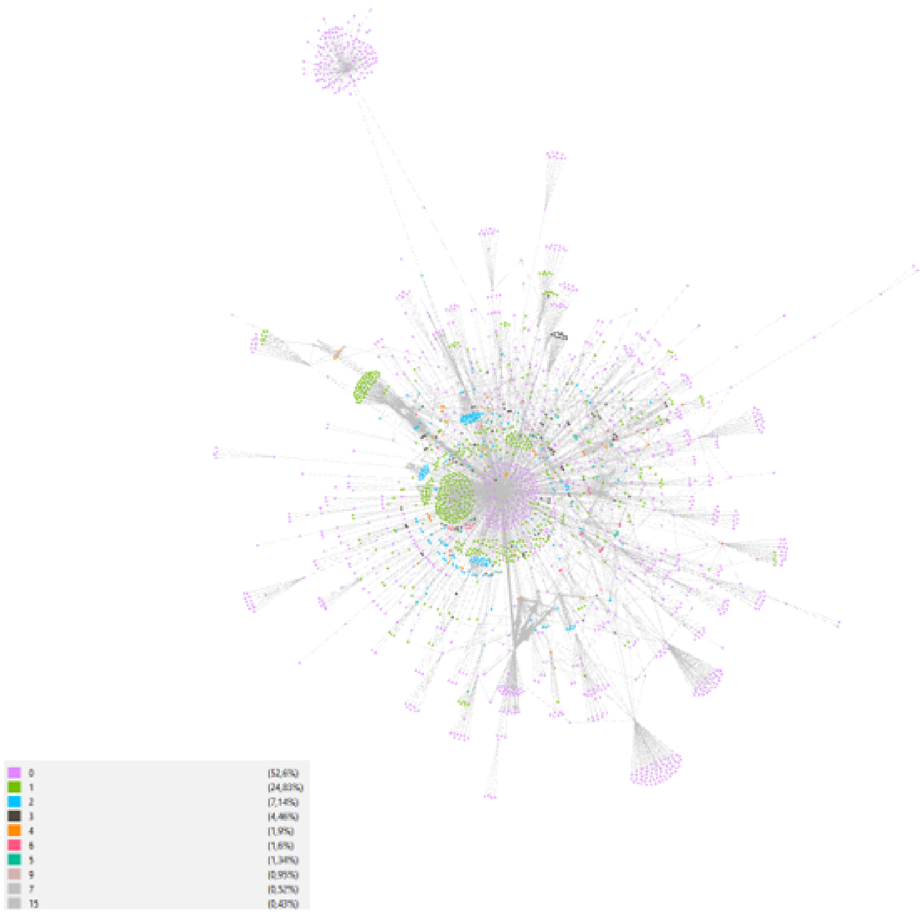


Figure 18: Number of Triangles

8.3 Triadic Closure

Triadic closure is a concept in network theory that refers to the process by which nodes in a network form new connections to complete existing triangles. In other words, triadic closure is the process by which three nodes that are already partially connected become fully connected to form a triangle.

In the directed graph of BBC News, as there are no triangles triadic closure does not exist. However, as we did before we convert the graph to an undirected graph and make the following analysis.

Below we can see the calculation of the triadic closure of the whole network using python. The triadic closure of the network is about 0.006.

```
import pandas as pd

# Calculate the total number of triangles in the network
total_triangles = df['triangles'].sum()

# Calculate the maximum number of triangles in the network
max_triangles = 0
for i in range(df.shape[0]):
  max_triangles += df.iloc[i]['Degree'] * (df.iloc[i]['Degree'] - 1) / 2

# Calculate the triadic closure
triadic_closure = total_triangles / max_triangles

# Print the triadic closure
print("Triadic Closure:", triadic_closure)
```

```
Triadic Closure: 0.006081739943893967
```

In this case, a triadic closure value of 0.006 indicates that the node or network has a relatively low triadic closure. A value close to 1 would indicate a high triadic closure, meaning that there are many triads in the network that are fully connected.

I also calculated the triadic closure for each node separately as we see below.

```
import pandas as pd

# Calculate the triadic closure for each node
triadic_closure = []
for i in range(df.shape[0]):
  # Calculate the number of triangles the node participates in
  triangles = df.iloc[i]['triangles']

  # Calculate the maximum number of triangles the node could participate in
  max_triangles = df.iloc[i]['Degree'] * (df.iloc[i]['Degree'] - 1) / 2

  # Calculate the triadic closure for the node
  triadic_closure.append(triangles / max_triangles)

# Add the triadic closure to the DataFrame as a new column
df['triadic_closure'] = triadic_closure

df['triadic_closure']
```

And the results are the following.

```
0       0.163636
1       0.032861
2       0.089474
3       0.043561
4       0.000982
          ...
2307    1.000000
2308    0.333333
2309    0.000000
2310    0.000000
2311    0.333333
Name: triadic_closure, Length: 2312, dtype: float64
```

There are 120 nodes in the network, which have a value of 1.0 as a triadic closeness. However, the most nodes have a much smaller value. The minimum value of triadic closeness in the network, which is 0 is appeared in 643 nodes.

9 Bridges and Local Bridges

Bridges and local bridges are important concepts in network analysis that refer to the connections between nodes in a network. Bridges are edges in a network that, if removed, would increase the number of connected components. In other words, bridges represent the minimum number of edges that need to be removed to isolate one node or a group of nodes from the rest of the network.

Local bridges, on the other hand, are a subset of bridges that connect nodes in the same component. In other words, local bridges represent the minimum number of edges that need to be removed to isolate a node or a group of nodes within the same component. Below we can see a script in python, which calculates the bridges and the local bridges of the network.

```python
import networkx as nx
import pandas as pd

# Read the CSV file into a pandas DataFrame
df = pd.read_csv("BBCNews_edges.csv")

# Create a graph using NetworkX
G = nx.Graph()

# Add edges to the graph from the DataFrame
for index, row in df.iterrows():
    G.add_edge(row['Source'], row['Target'])

# Find all bridges in the graph
bridges = list(nx.bridges(G))

# Find all local bridges in the graph
local_bridges = list(nx.local_bridges(G))

# Print the bridges and local bridges
print("Length of Bridges:", len(bridges))
print("Length of Local Bridges:", len(local_bridges))
```

```
Length of Bridges: 982
Length of Local Bridges: 1723
```

Having 982 bridges and 1723 local bridges in your network means that the network has 982 edges that are crucial for maintaining the connectivity of the whole network and 1723 edges that are important for maintaining the connectivity within subnetworks.

Unfortunately, I cannot visualize the bridges and the local bridges as I cannot pass the new dataset, that contains the bridges back to Gephi.

10 Gender

Gender is an important demographic characteristic that can impact various aspects of individuals' experiences and behaviors, and it is therefore important to include gender in demographic analyses where possible. However, in the absence of reliable information about the gender of individuals in a network, any attempt to estimate the gender balance would be based on assumptions and would lack validity

11 Homophily

Homophily refers to the tendency for people to form ties with others who share similar characteristics, and for these ties to be stronger and more frequent than those with individuals who are dissimilar. This can result in the formation of homogeneous groups within a larger network, with relatively little cross-group communication or interaction.

Homophily has important implications for the spread of information, norms, and behaviors in networks. For example, because people tend to be more likely to share information and ideas with others who are similar to themselves, homophily can result in the reinforcement of existing beliefs and opinions and can lead to the creation of "echo chambers," where individuals are exposed primarily to ideas and perspectives that are similar to their own.

In BBC News network we tried to find homophily firstly between the nodes based on the number of friends of every node and then based on the betweenness centrality. As we can observe below, there is no homophily between the nodes based on the number of friends of every node.

```
import pandas as pd
import networkx as nx

nodes_df = pd.read_csv('BBCNews.csv')
edges_df = pd.read_csv('BBCNews_edges.csv')

G = nx.Graph()

for i, row in nodes_df.iterrows():
    G.add_node(row['Id'], label=row['Label'], friends_count=row['friends_count'])

for i, row in edges_df.iterrows():
    G.add_edge(row['Source'], row['Target'], weight=row['Weight'])

def jaccard_similarity(G, u, v):
    set_u = set([G.nodes[u]['friends_count']])
    set_v = set([G.nodes[v]['friends_count']])
    jaccard = len(set_u & set_v) / len(set_u | set_v)
    return jaccard

count = 0
similarity_count = 0
for u, v in G.edges:
    count += 1
    similarity = jaccard_similarity(G, u, v)
    if similarity > 0:
        print(similarity)
    else:
        similarity_count += 1
if similarity_count == count:
    print('All pairs of nodes have zero (0) similarity')
```

```
All pairs of nodes have zero (0) similarity
```

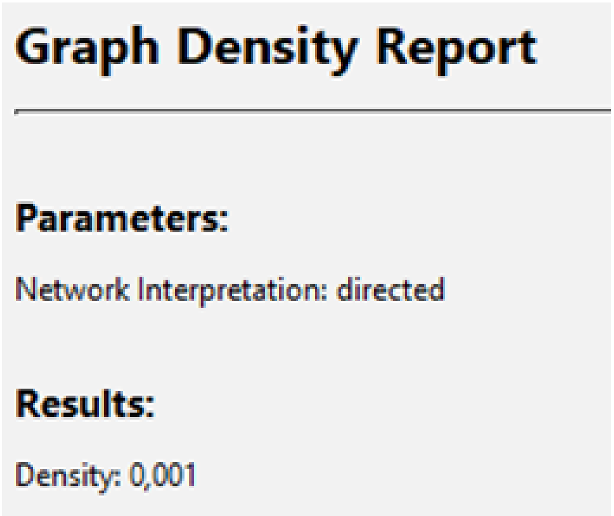
However, we can observe below, that there is homophily based on the betweenness centrality between the nodes. More specifically, there are 3183 homophily relationships between the nodes out of 4479 that were checked. All these relationships have a Jaccard similarity score of one. Generally, the higher the Jaccard similarity score, the higher the level of homophily between the nodes.

```python
import pandas as pd
import networkx as nx

nodes_df = pd.read_csv('BBCNews.csv')
edges_df = pd.read_csv('BBCNews_edges.csv')

G = nx.Graph()

for i, row in nodes_df.iterrows():
    G.add_node(row['Id'], label=row['Label'], betweenesscentrality=row['betweenesscentrality'])

for i, row in edges_df.iterrows():
    G.add_edge(row['Source'], row['Target'], weight=row['Weight'])

def jaccard_similarity(G, u, v):
    set_u = set([G.nodes[u]['betweenesscentrality']])
    set_v = set([G.nodes[v]['betweenesscentrality']])
    jaccard = len(set_u & set_v) / len(set_u | set_v)
    return jaccard

count = 0
not_similarity_count = 0
similarity_count = 0
for u, v in G.edges:
    count += 1
    similarity = jaccard_similarity(G, u, v)
    if similarity > 0:
        similarity_count += 1
    else:
        not_similarity_count = 0
if similarity_count == count:
    print('All pairs of nodes have zero (0) similarity')
else:
    print('There are', similarity_count, 'homophily relationships base on the betweenness centrality, out of
```

```
There are 3183 homophily relationships base on the betweenness centrality, out of 4479 that were cheked!
```

12 Graph Density

Graph density is a measure of how densely connected a graph is, that is, how many edges there are in relation to the number of possible edges. In other words, it measures how tightly connected the nodes in a graph are. For a complete graph, where every node is connected to every other node, the graph density is 1. For a sparse graph, where there are few edges, the graph density is close to 0. In general, a dense graph is one with a high graph density, while a sparse graph is one with a low graph density. BBC News network seems to be a sparce graph, as it has 0.001 density, which is very close to zero.



Graph Density Report

Parameters:

Network Interpretation: directed

Results:

Density: 0,001

13 Community Structure (modularity)

Community structure, also known as modularity, refers to the grouping of nodes in a network into coherent and densely interconnected sub-groups or communities. The idea behind community structure is that nodes within a community have many more connections to other nodes in the same community than to nodes in other communities.

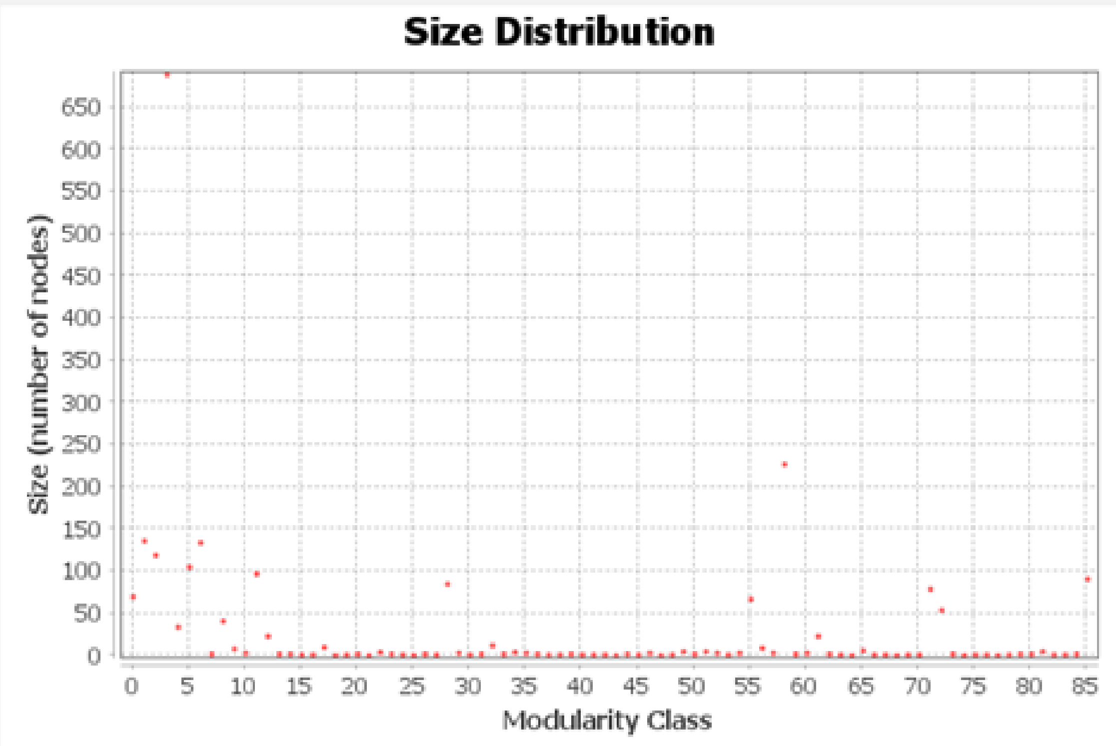
Modularity is a metric that quantifies the quality of a community structure. It measures the degree to which the nodes in a network are divided into communities, and the strength of these communities. Higher modularity values indicate a stronger community structure, where nodes within communities are more densely interconnected than nodes across communities. Below we can see the results of the calculation of the modularity of the BBC News network.

Parameters:

Randomize: On
Use edge weights: On
Resolution: 1.0

Results:

Modularity: 0,715
Modularity with resolution: 0,715
Number of Communities: 86



The modularity value of 0.715 indicates that the network has a strong community structure. The higher the modularity value, the stronger the community structure. A modularity value of 0.715 is considered a high value, which means that the nodes in your network are divided into relatively well-defined communities.

The number of communities in the network (86) means that the network has been divided into 86 separate communities. A large number of communities can indicate that the network has a very fine-grained community structure, while a small number of communities may indicate a coarser community structure. We could say that 86 communities are a lot for a network with 2312 nodes in total.

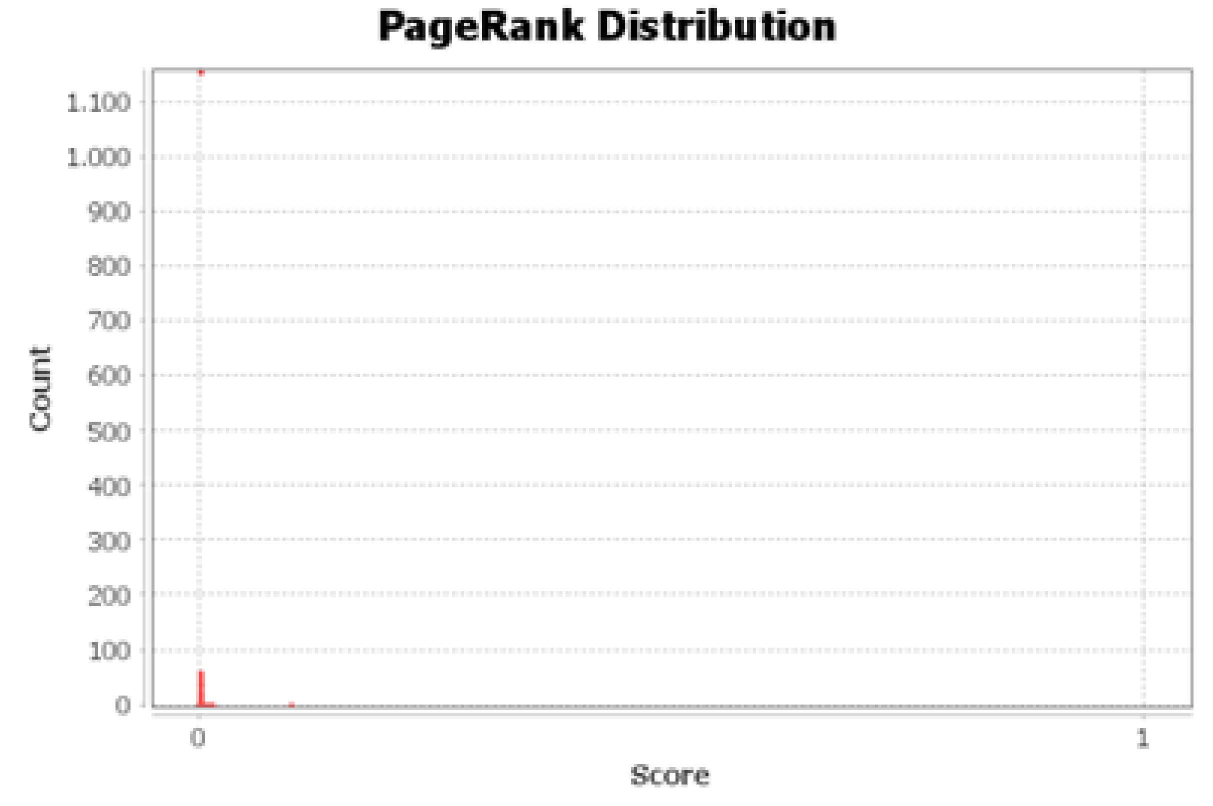
It has been observed that there is a dominant community in the network which contains the node @bbcnews as one of its members, as it is shown below. This community comprises a substantial number of nodes, approximately 700, and has been identified as the largest community in the network. The node @bbcnews is located at the center of this community and serves as a hub connecting many of the nodes within the community. This observation highlights the centrality and importance of the node @bbcnews within the network and the strong community structure that exists among the nodes in the network.

Figure 19: Modularity

14 PageRank

The idea behind PageRank is that a node is considered important if it has many incoming links from other important nodes. In a network, the PageRank of each node is calculated based on the number and quality of the links that point to that node. A node with many incoming links from high-PageRank nodes will have a high PageRank, while a node with few or no incoming links will have a low PageRank.

Below it is shown the PageRank Distribution that was calculated by Gephi.



As we can see from the screenshot above and from this table, the node with the higher PageRank is @bbcnews with a value of 0.095, which seems to be the most influential node with all the other nodes having a much smaller value.

Label	PageRank
@bbcnews	0.095873
@timcook32	0.012676
@ikabirbedi	0.010575
@narendramodi	0.00802
@keveeyes	0.004054
@profnfenton	0.002515
@letina241	0.002422

As we can see, the minimum PageRank of a node in the network is 0.0002, which is a value that more than the 50% of the nodes have. The maximum PageRank is about 0.0959. The mean of all the PageRanks is 0.000305.

```
df['pageranks'].describe()
```

```
count    2312.000000
mean        0.000305
std         0.002030
min         0.000202
25%         0.000202
50%         0.000202
75%         0.000238
max         0.095873
Name: pageranks, dtype: float64
```

Based on the above, we can do the following observations.

- High degree of centralization: The minimum PageRank of a node in the network is 0.0002, which is a value that more than 50% of the nodes have. This suggests that the network is highly centralized, with most of the PageRank being concentrated in a few nodes.

- Low average PageRank: The mean of all the PageRanks is 0.000305, which is a relatively low value. This suggests that the overall importance of nodes in the network is relatively low.

- Skewed distribution of PageRanks: The maximum PageRank is about 0.0959, which is much higher than the mean. This indicates that there is a skewed distribution of PageRanks in the network, with a few nodes having a much higher PageRank than the rest. Also, we can see below a visual representation of the network, which provides a useful illustration of the PageRank of the nodes. The size of each node is proportional to its PageRank, with larger nodes indicating higher PageRank and smaller nodes indicating lower PageRank.

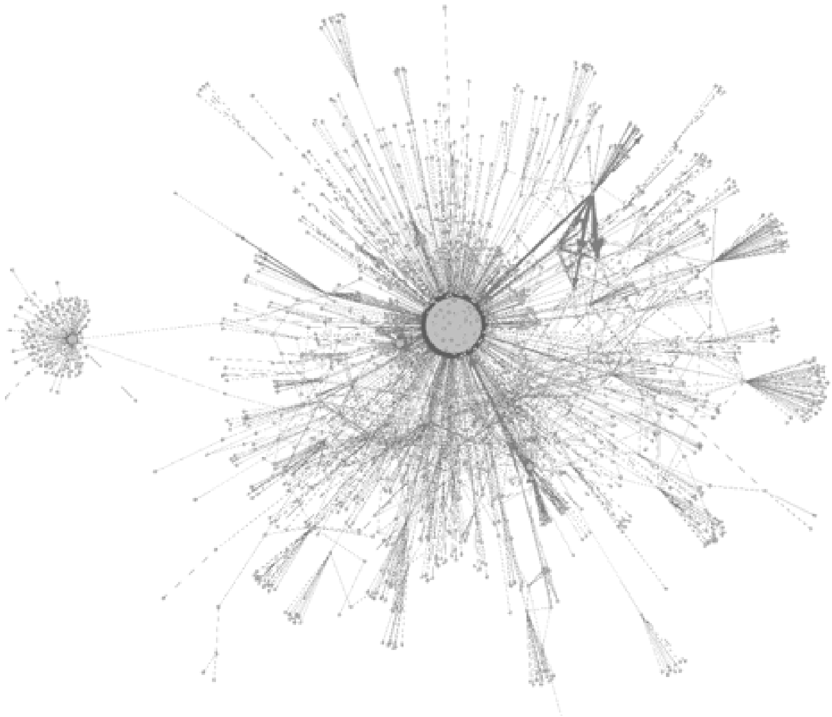


Figure 20: PageRank

15 Integrated Graphical Representation of the Network

Now that we have calculated all the appropriate measures we can represent the network with a better way, in order to gain a deeper understanding of the relationships between its nodes. Therefore, the following steps were followed:

- The Degree Range filter was set to a minimum value of 1 to eliminate all nodes that were not connected to any other nodes. This resulted in the removal of only 9 nodes, maintaining the overall structure of the network.
- The color of each node was assigned based on its Modularity Class attribute, enabling the identification of distinct communities within the network.
- The size of each node was based on its PageRank, with a range of 10 to 50, where larger nodes represented higher PageRank values.
- The size of the node labels, which display the names of each node, was based on the Betweenness Centrality ranking, with a range of 1 to 5, where larger labels represented higher Betweenness Centrality values.

As a result of these steps, the following representation of the network was produced:

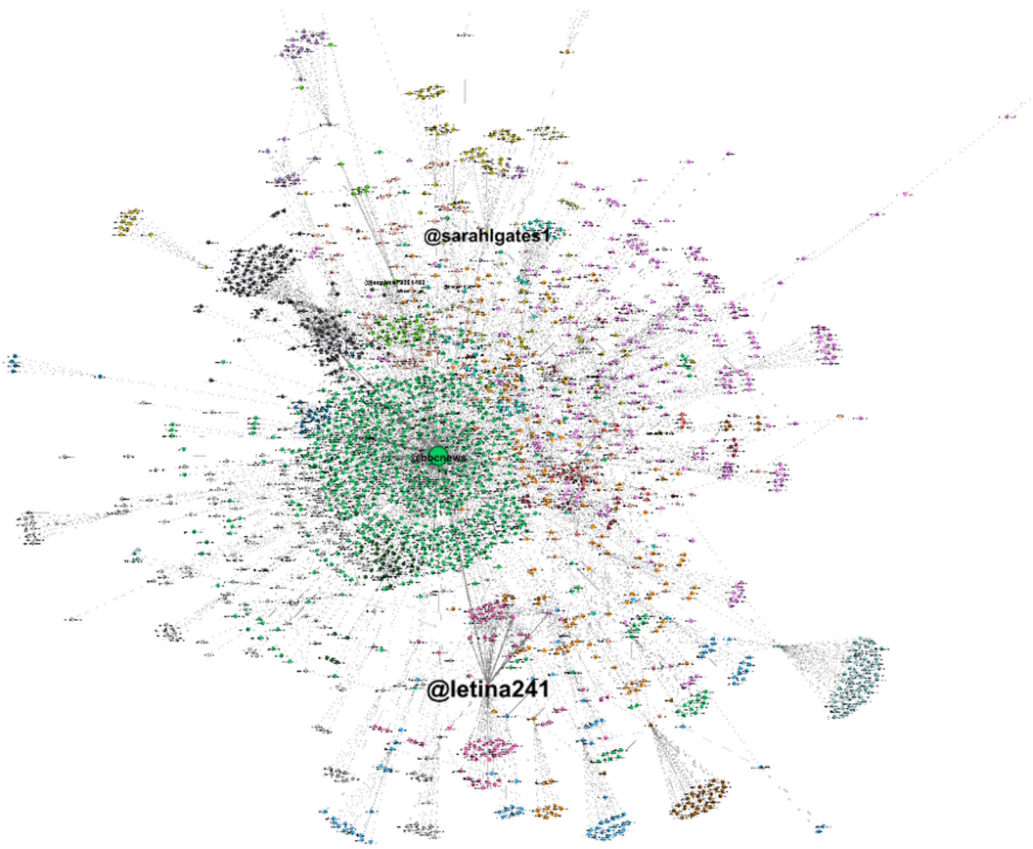


Figure 21: Multiple measures Network

The next two pictures are given, as the sizes and colors are more distinct.

Figure 22: Multiple measures Network - Yifan Hu's

Figure 23: Multiple measures Network - Fruchterman Reingold