SOCIAL NETWORK ANALYSIS

E-Covid19 Graph In Morocco Study Case

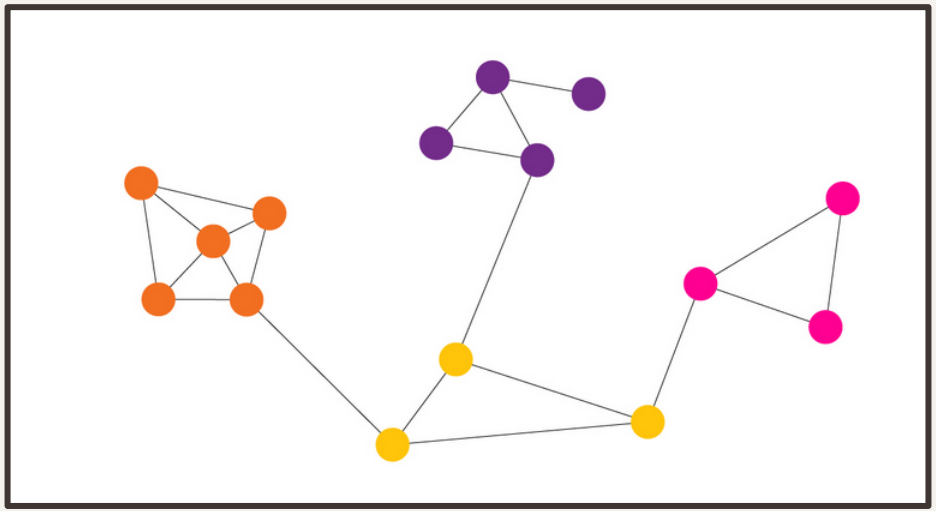
Nisrine CHOUIKH

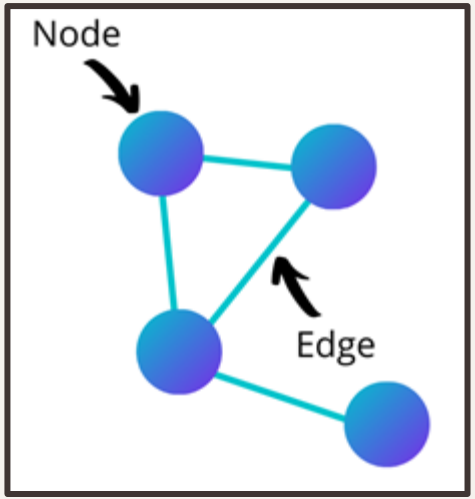
Table of contents

01

Introduction of social network analysis

Social network analysis (SNA), **also known as network science, is a field of data analytics that uses networks and graph theory to understand social structures.**





It characterizes networked structures in terms of nodes (individual actors, people, or things within the network) and the ties, edges, or links (relationships or interactions) that connect them.

Nodes and edges are a key concept in networks:

- **In social networks, nodes can represent people.**
- **In social networks, edges can represent connections.**

02

Provenance of the data - tools

Tools used to collect data

TWINT - Twitter Intelligence Tool

Twitter

Tools used to display the network

Why Choosing Twitter?

Personal expression

- Brings together hundreds of millions of users using messages of 140 characters

Availability of the material

- Possibility for researchers to analyze its contents

This frequently lead to questions on influence measuring

Scrapping From Twint

For the key words that we used are :

- **Hashtags** : #covid19, #maroc
- **Sentence** : covid19, maroc

```
In [18]: #configure seach requirements
         c=twint.Config()
         c.Search = ['#covid19','#maroc']
         c.Limit = 2000
         profile_full = True
         c.Since="2020-03-02"
         c.until="2021-03-02"
         c.Store_csv = True
         c.Retweets = True
         c.Native_retweets=True
         c.Output = r"C:\Users\Pro\Downloads\Stage Ete - S8\Scrapping\TWEETS"
         #run
         twint.run.Search(c)
```

We ran multiple times the code to capture the retweets

Scrapping From Twint

Important Note :

We are collecting only retweets for our upcoming analysis.

```
1423344273060139011 2021-08-05 19:04:53 +0100 <viralvideovlogs> RT @Ministere_Sante: وزارة الصحة تحث المواطنات والمواطنين الذين يعانون من مشا
كل الحساسية والنساء المرضعات والنساء الحوامل ابتداء من الشهر ا…
1423344267427254274 2021-08-05 19:04:52 +0100 <Vendetta_M_dima> RT @Ministere_Sante: وزارة الصحة تحث المواطنات والمواطنين الذين يعانون من مشا
كل الحساسية والنساء المرضعات والنساء الحوامل ابتداء من الشهر ا…
1423335890387185667 2021-08-05 18:31:35 +0100 <Rozlbrahim> RT @Le360fr: MA🧉 | A #Tanger, les forces de l’ordre se sont déplo
yées à chaque entrée et sortie de la ville pour assurer le respect de ces…
1423324390159179779 2021-08-05 17:45:53 +0100 <mouhayarnajia> RT @Khalid25666691: @Knizakenzaa #كلنا_لمحاربة_كورونا  #grippe #vaccin
s #نبقاو_على_بال #maroc #وقايتنا# نخرجو_بأمان #COVID19  https://t.co/f…
```

	G	H	I	J	K	L	M	N	O	P	Q
1	user_id	username	name	place	tweet	language	mentions	urls	photos	replies_co	retweets_count
2	9.52E+17	viralvideovlogs	#StayHome		RT @Ministere_Sante: ا بهر	ar	[{'screen_name': 'ministere_sante', 'name': 'ministère-santé-maroc ministry of health morocco', 'id': '2683723274'}]	[]	[]	0	2
3	1.37E+18	vendetta_m_dima	VMAPatriotM		RT @Ministere_Sante: ا بهر	ar	[{'screen_name': 'ministere_sante', 'name': 'ministère-santé-maroc ministry of health morocco', 'id': '2683723274'}]	[]	[]	0	2

03

EDA – Data Processing

Workflow

01

Type of Network

Understand and specify the type of network based on the issue that we want to analyze

02

First Sketch of the network

A clear picture of what we want to achieve

03

Anatomy of the retweet

Key elements of a retweet that we are going to use.

04

Extraction

Extract the information that would be the key element for the nodes and edges

Type of network

Types of network ties that can displayed from the twitter data:

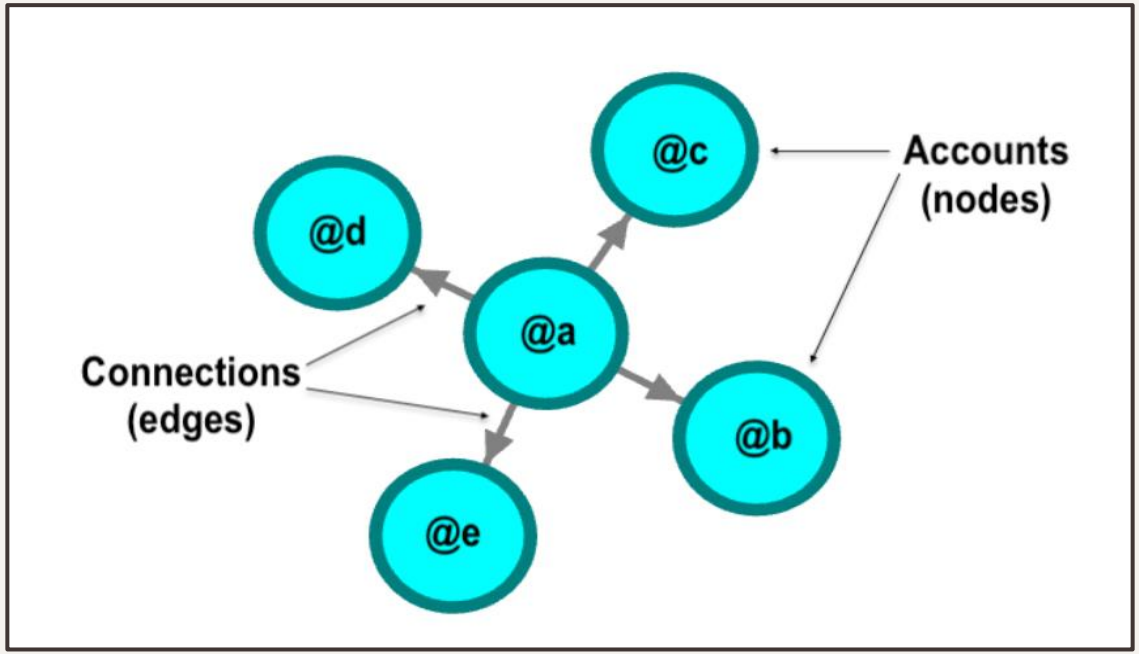
- Retweets
- Quotes
- Replies

For our internship, we chose a **Retweets network** based on **retweets** ties.

Retweets network:

- The retweets network is based on the interaction between users who tweeted at first and the people who retweeted, mainly focused on who passes information to whom.

First Sketch of the network



Anatomy of a tweet

@alice:

RT *@bob* VIA ***@carol*** : just felt an #earthquake in #Fairfax!

Means:

@alice has retweeted from *@bob*

@bob mentioned ***@carol***

Hashtags: #earthquake and #Fairfax are related

Extraction of the Data

```
G = nx.DiGraph()
for r in twt.iterrows():
    for user in r[1]['splitted_users']:
        G.add_edge(r[1]['username'], user)
```

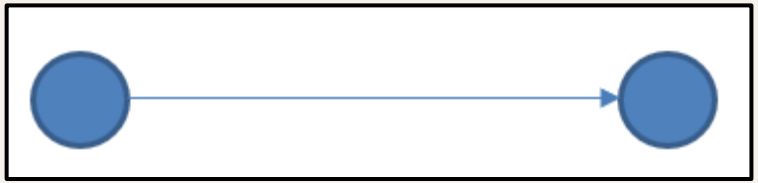
Nodes : the author (Username) of the original tweet

Edges: the username of the people that retweeted that specific tweet from the author.

04

Graph Analysis

Choice of Graph : Directed



In our case, we are working with twitter, which means it is a directed graph

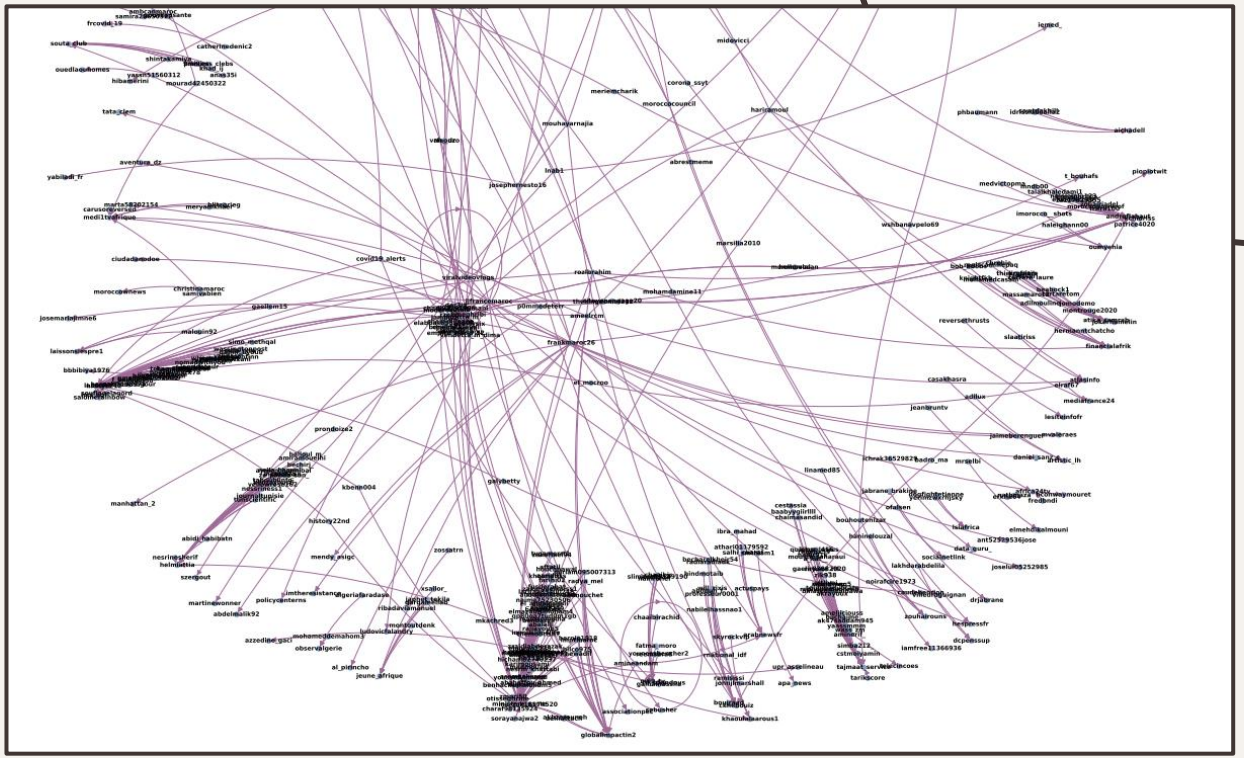
For instance, I may follow you on twitter, but it doesn't mean that you follow me. On the other hand, Facebook graph can be Undirected since if we are "friends", it means instantly that we have a reciprocated relationship.

Drawing the network

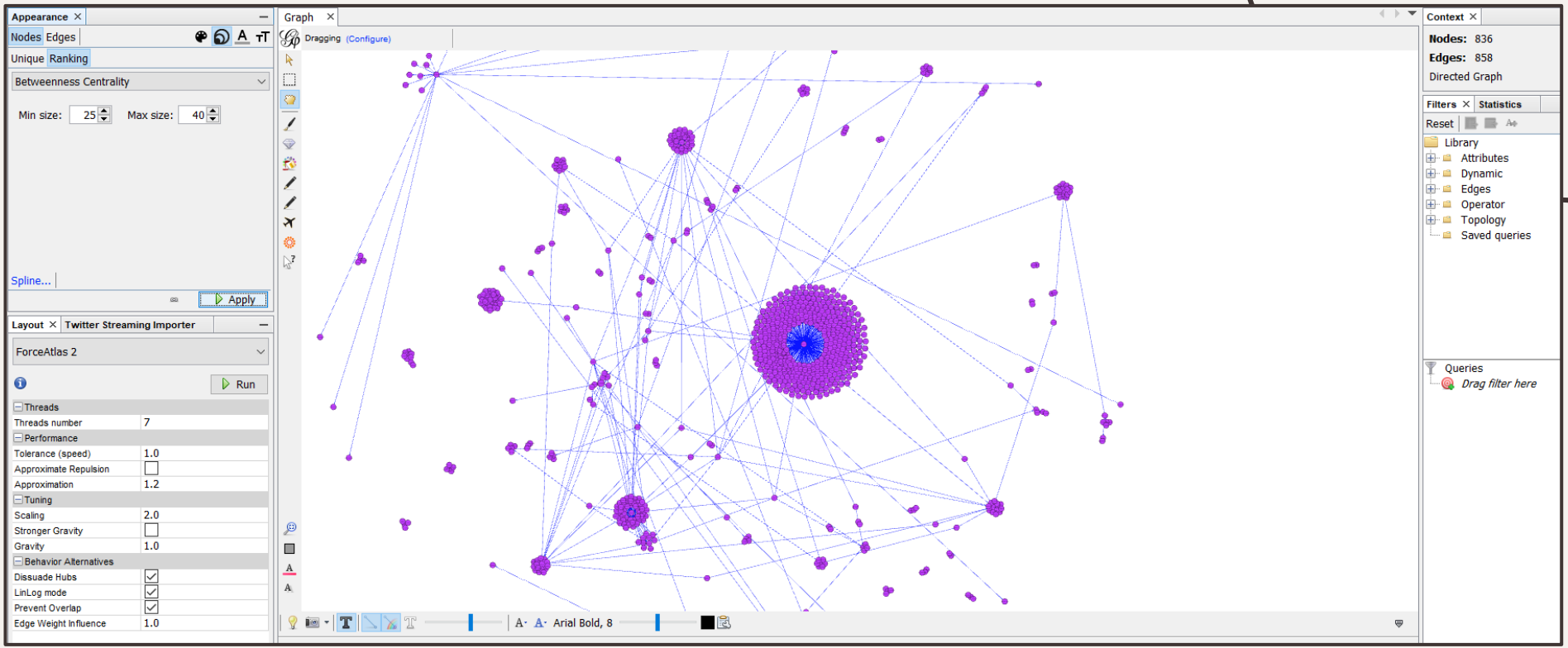
This is the resulting graph using NetworkX.

Unfortunately, we can’t detect the nodes nor have a good visualization

For better visualization, we are going to use Gephi.



Drawing the network



Basic Information about the network

```
G.number_of_nodes()
```

```
836
```

```
G.number_of_edges()
```

```
858
```

```
print(nx.info(G))
```

```
DiGraph with 836 nodes and 858 edges
```

Advanced Analysis of the graph

Centrality measures address the question:
"Who is the most important or central person in this network?"

Centrality measures:

- Degree centrality
- In-Degree / Out-Degree
- Closeness centrality
- Betweenness centrality
- Eigenvector

Advanced Analysis – Degree

Degree of a node defines the number of connections a node has.

```
print("Degree of nodes:")
for D in nx.degree(G):
    print("\tDegree:", D)
```

```
        Degree: ( smr_093 , 1)
        Degree: ('mediavenir', 395)
        Degree: ('latifaa85400248', 1)
        Degree: ('radya_mel', 2)
        Degree: ('amineandam', 3)
        Degree: ('baratto m', 27)
```

We notice that the node "**mediavenir**" has the highest degree with 395 connections : to make sure it is the biggest influencer, we discover the degree centrality

Advanced Analysis : In-Degree / Out-Degree

1. In-Degree

The nodes with higher indegree are more prestigious (choices received).

```
#in Degree : The nodes with higher indegree is more prestigious (choices received)
print("IN-Degree of nodes:")
for D in G.in_degree():
    print("\tIn Degree:", D)
```

```
        In Degree: ('zmori50', 0)
        In Degree: ('smr_093', 0)
        In Degree: ('mediavenir', 395)
        In Degree: ('latifaa85400248', 0)
```

Advanced Analysis : In-Degree / Out-Degree

2. Out-Degree

- The nodes with higher outdegree are more central (choices made).

```
#out Degree : The nodes with higher outdegree is more central (choices made)
print("OUT-Degree of nodes:")
for D in G.out_degree():
    print("\tout Degree:", D)
```

```
        out Degree: ('sidneyazoulay', 0)
        out Degree: ('marzok18174520', 1)
        out Degree: ('salomerainbow', 1)
        out Degree: ('christinamaroc', 1)
        out Degree: ('moroccownews', 0)
        out Degree: ('frankmaroc26', 17)
        out Degree: ('piopiotwit', 0)
```

Advanced Analysis – Degree Centrality

Nodes with highest degree have highest degree centrality.

```
#Nodes with highest degree have highest degree centrality.
centrality = nx.degree_centrality(G)
[(x, centrality[x]) for x in sorted(centrality, key=centrality.get, reverse=True)]
```

```
[('mediavenir', 0.47305389221556887),
 ('ministere_sante', 0.08263473053892216),
 ('le360fr', 0.05149700598802395),
 ('tajmaat_service', 0.03473053892215569),
 ('baratto_m', 0.032335329341317366),
```

Before, we got the following resultat :
"**mediavenir**" has the highest links with 395 connections

⇨ Which explains also that it has the highest degree centrality

Advanced Analysis – Closeness Centrality

Nodes with the shortest paths have highest closeness centrality.

Closeness centrality is a way of detecting nodes that can spread information very efficiently through a graph.

```
#Nodes with the most shortest paths have highest closeness centrality
centrality = nx.closeness_centrality(G)
[(x, centrality[x]) for x in sorted(centrality, key=centrality.get, reverse=True)]
```

```
[('mediavenir', 0.47305389221556887),
 ('ministere_sante', 0.08444049401197605),
 ('le360fr', 0.05149700598802395),
 ('tajmaat_service', 0.03473053892215569),
```

Advanced Analysis – Betweenness Centrality

Nodes that appear most often in shortest paths have highest betweenness centrality.

Betweenness centrality quantifies the number of times a node acts as a bridge along the shortest path between two others nodes.

```
#Nodes that appear most often in shortest paths have highest betweenness centrality. Many paths must flow throw nodes with high betweenness
centrality = nx.betweenness_centrality(G)
[(x, centrality[x]) for x in sorted(centrality, key=centrality.get, reverse=True)]
```

```
[('baratto_m', 0.0001191860882551444),
 ('khalid25666691', 4.7387239908671866e-05),
 ('hmeghribi', 3.7335401140165716e-05),
 ('viralvideovlogs', 0.0),
 ('ministere_sante', 0.0),
```

Advanced Analysis – Eigenvector Centrality

In graph theory, eigenvector centrality (also called eigencentrality or prestige score) is a measure of the influence of a node in a network.

A high eigenvector score means that a node is connected to many nodes who themselves have high scores.

```
#Most important
for w in sorted (most_important_bond,key=most_important_bond.get,reverse=True):
  print(w,most_important_bond[w])
```

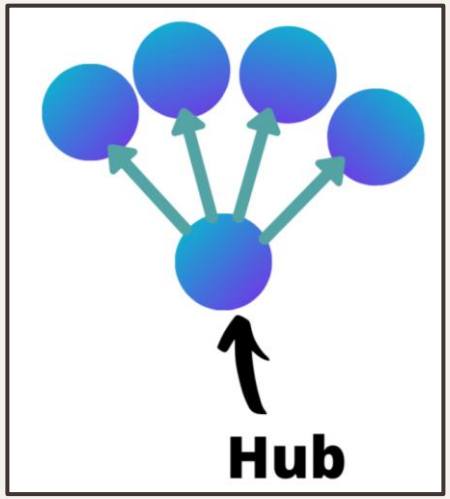
```
baratto_m 0.5075427294797531
```

Hubs and authorities

1. Hubs

A hub is a node that has many edges pointing out of it.

Hubs serve as bridges between the small degree nodes.

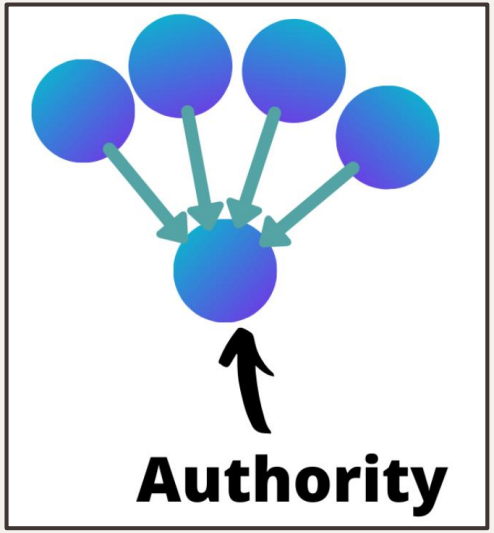



```
hubs, authorities = nx.hits(G, max_iter = 50, normalized = True)

for w in sorted (hubs,key=hubs.get,reverse=True):
  print(w,hubs[w])

viralvideovlogs 0.0025901439186865735
ameelrcm 0.0025848829507856535
```

Hubs and authorities

1. Authorities

An authority, on the other hand, is a node that has many edges pointing to it.




```
for w in sorted (authorities,key=authorities.get,reverse=True):
  print(w,authorities[w])
```

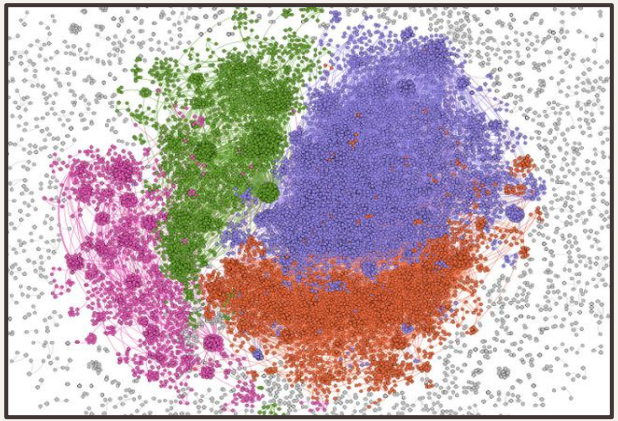
```
mediavenir 0.9633952416359641
ministere_sante 0.009190272762494962
le360fr 0.005757597354105726
```

Community detection – Using Gephi

Communities or sub-units are the sub-networks in a network that are highly interconnected nodes.

The community indicates the existence of internal structures that have special characteristics or play the same role in a network.

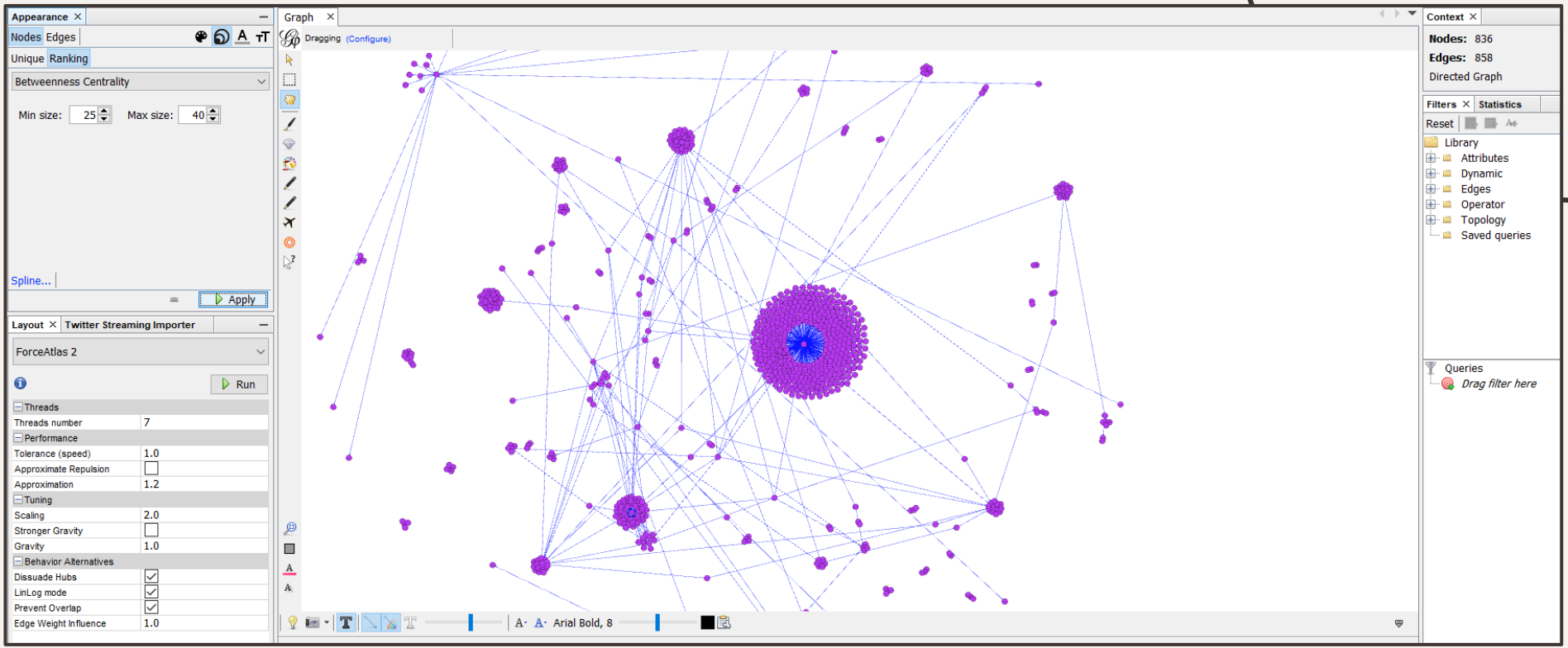
Highly connected groups of individuals or objects inside these networks are communities. It usually lies at the intersection point of the network and group.



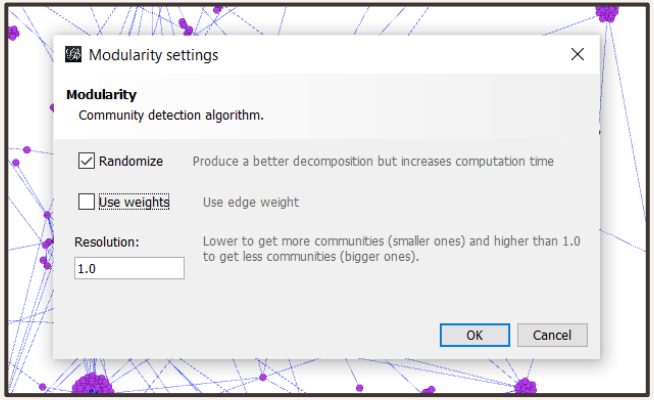
Modularity :

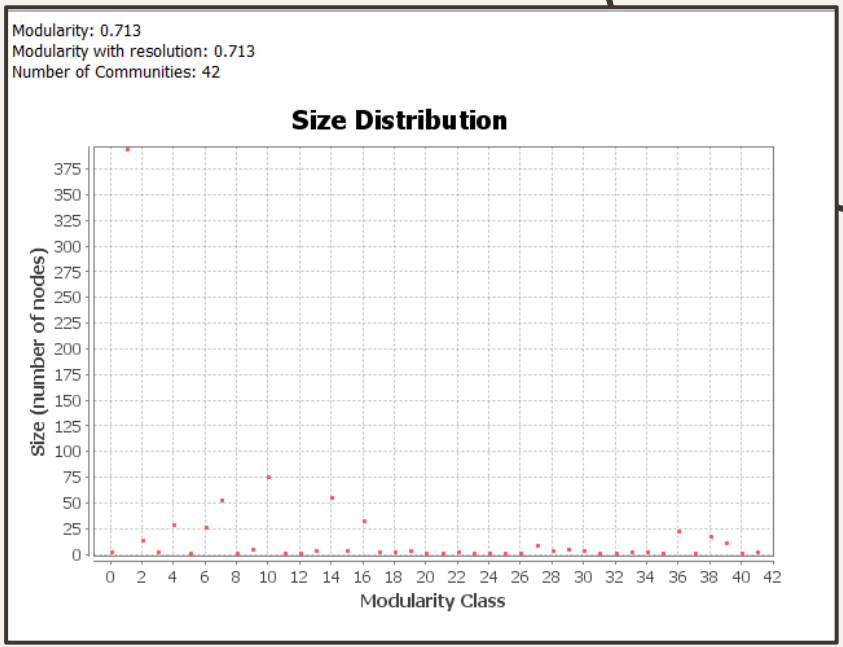
Modularity is **a measure of the structure of networks or graphs which measures the strength of division of a network into modules** (also called groups, clusters or communities).

Community detection

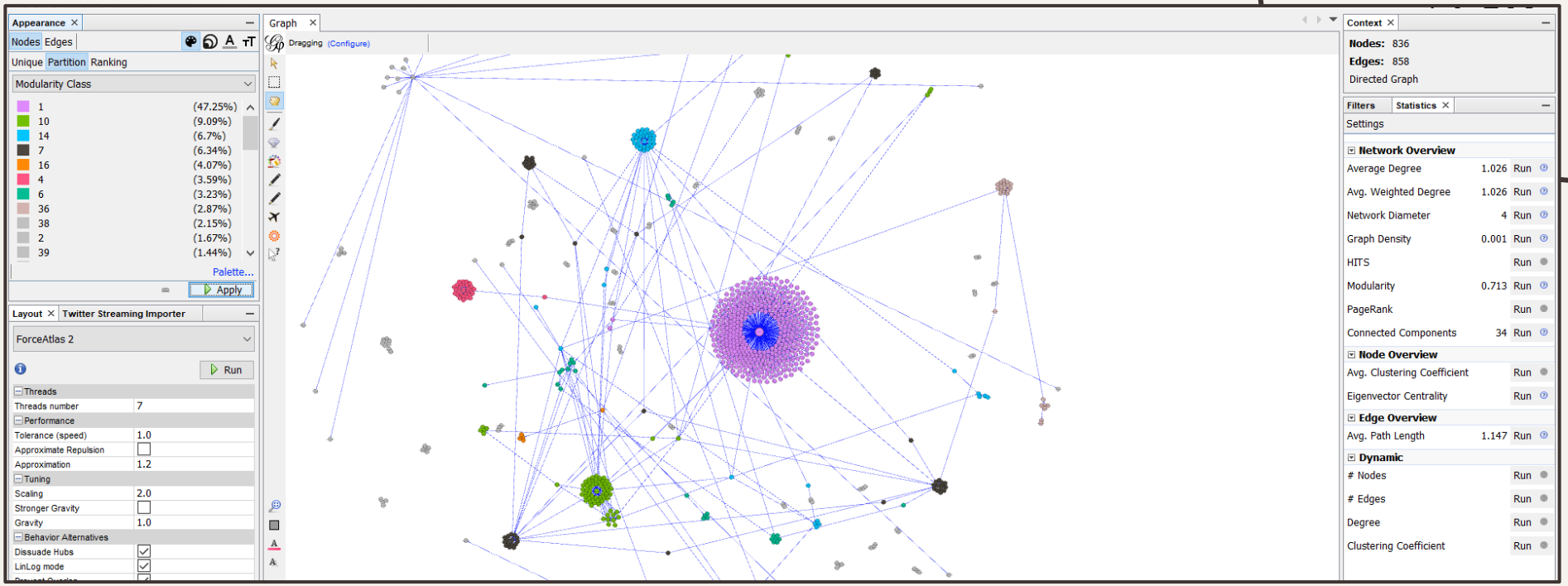


Community detection

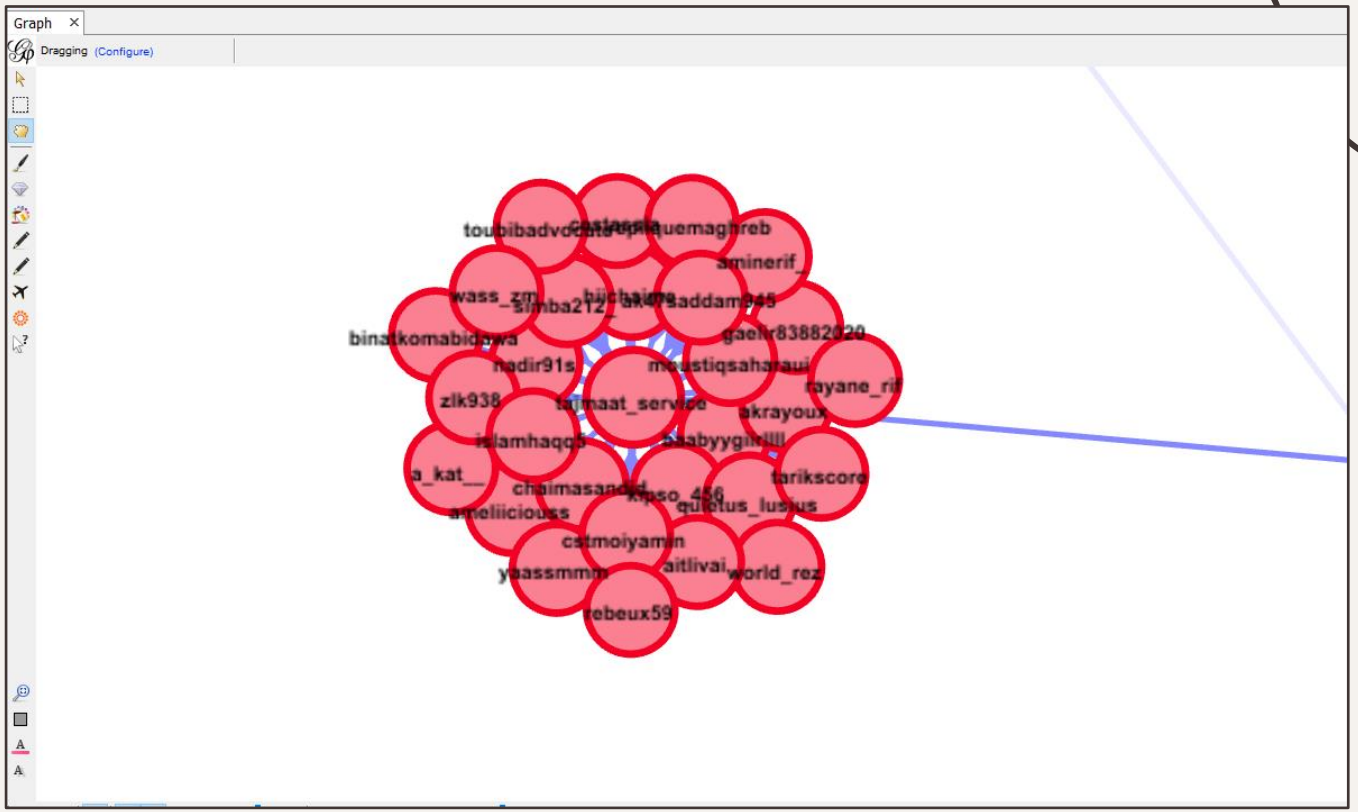




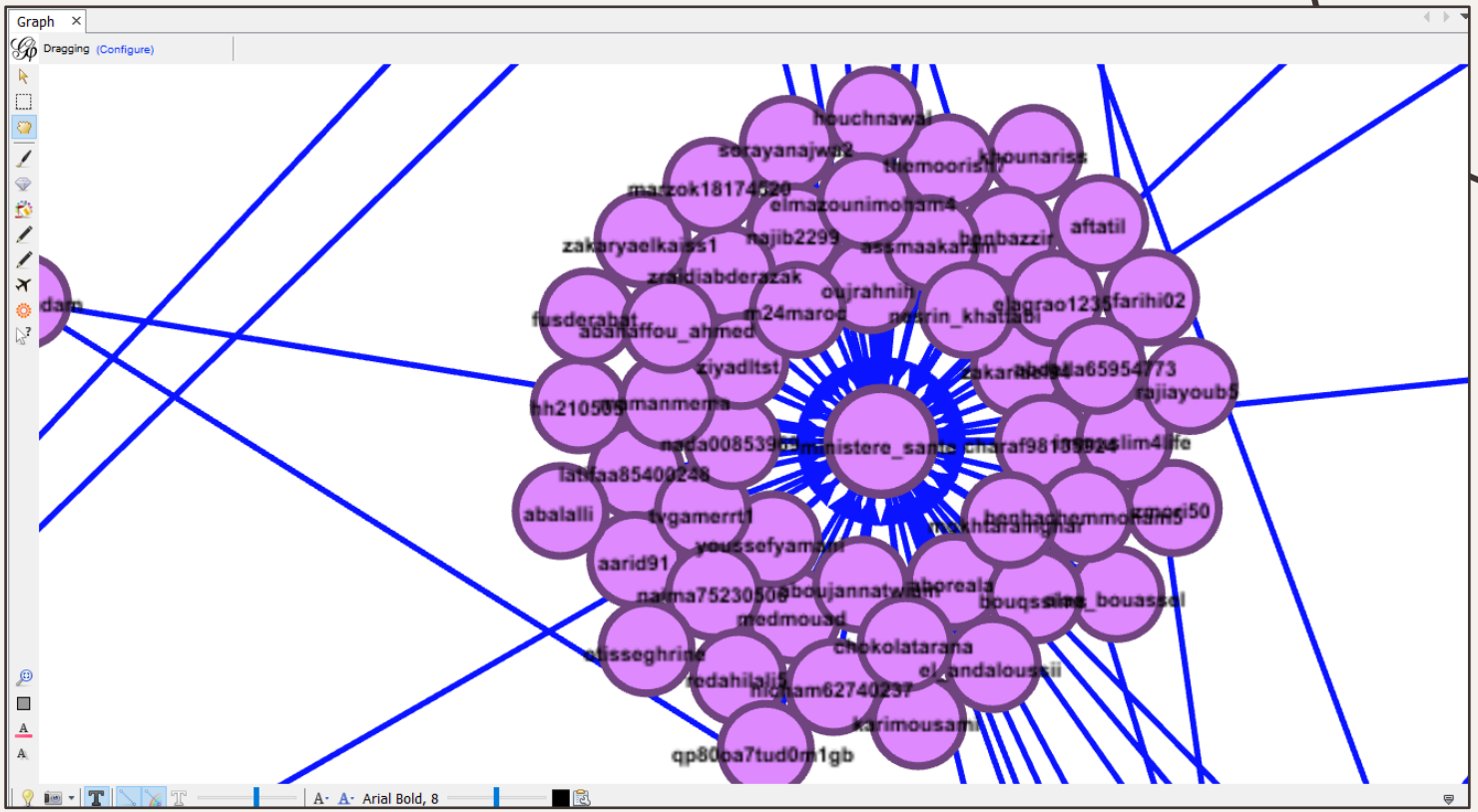
Community detection



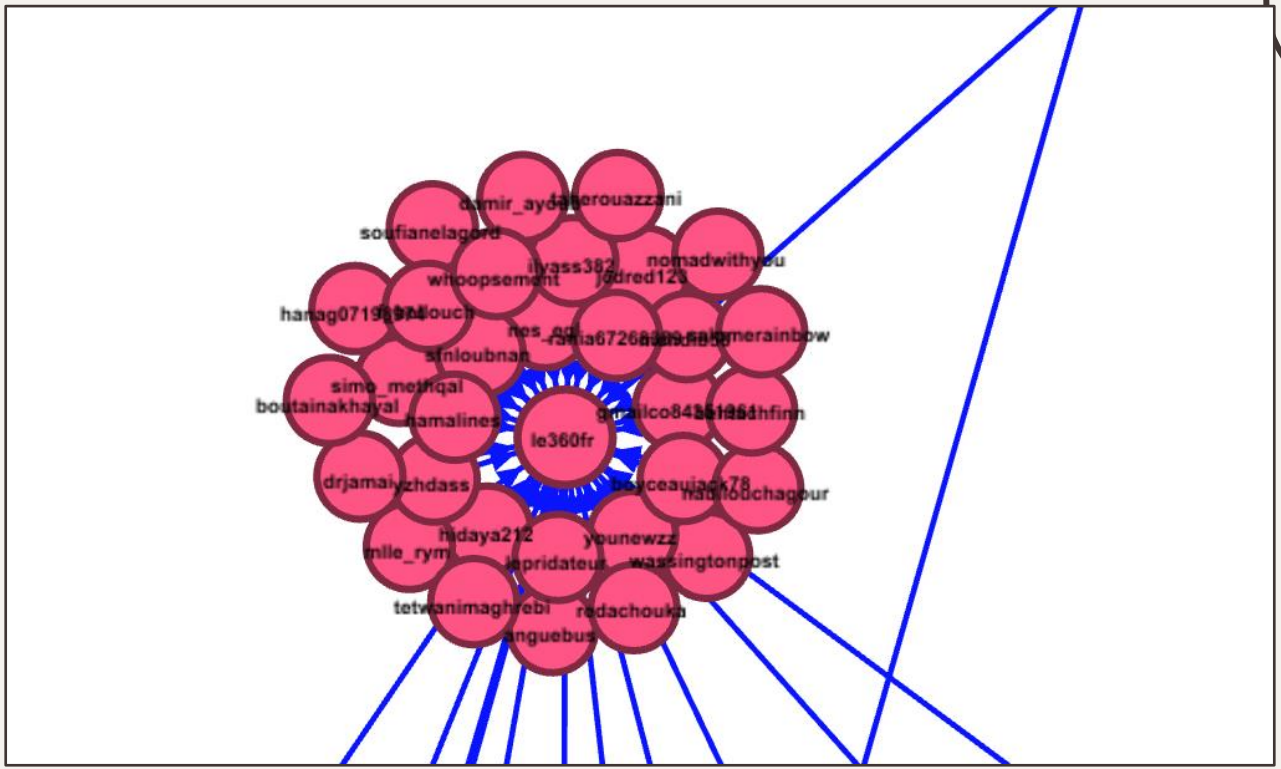
Community detection- Some Communities



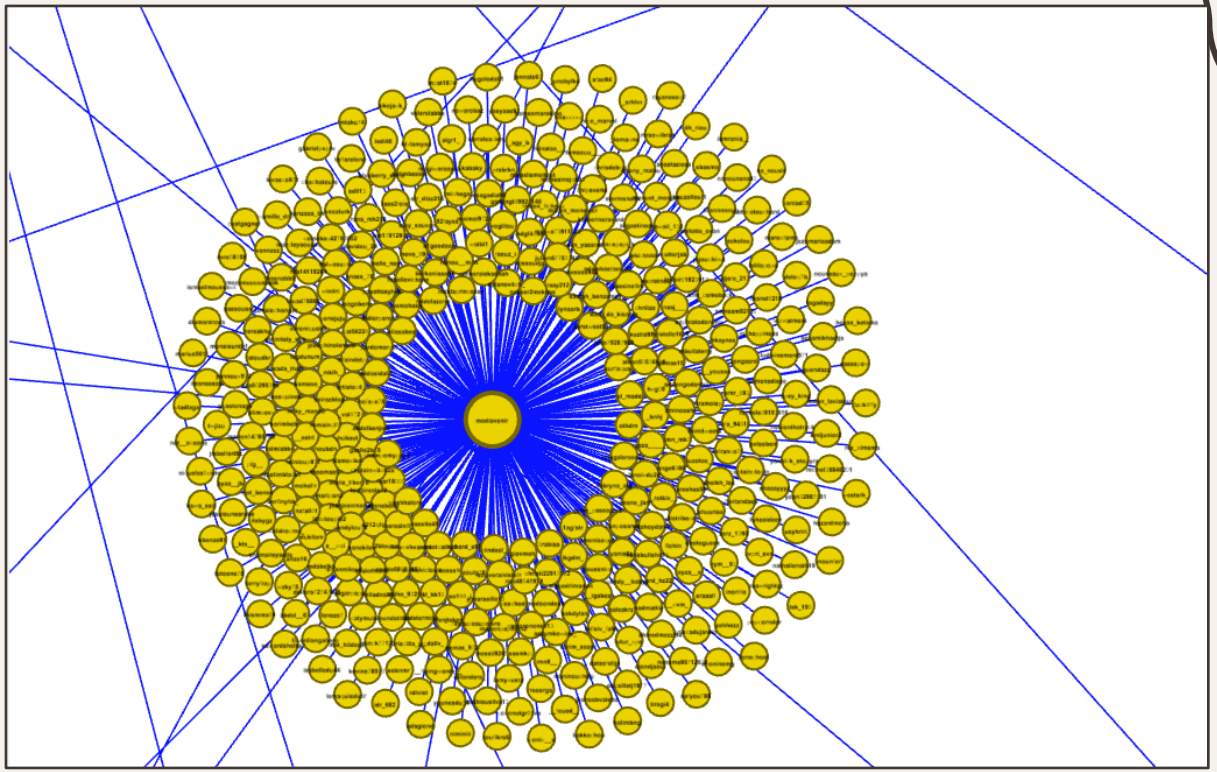
Community detection - Some Communities



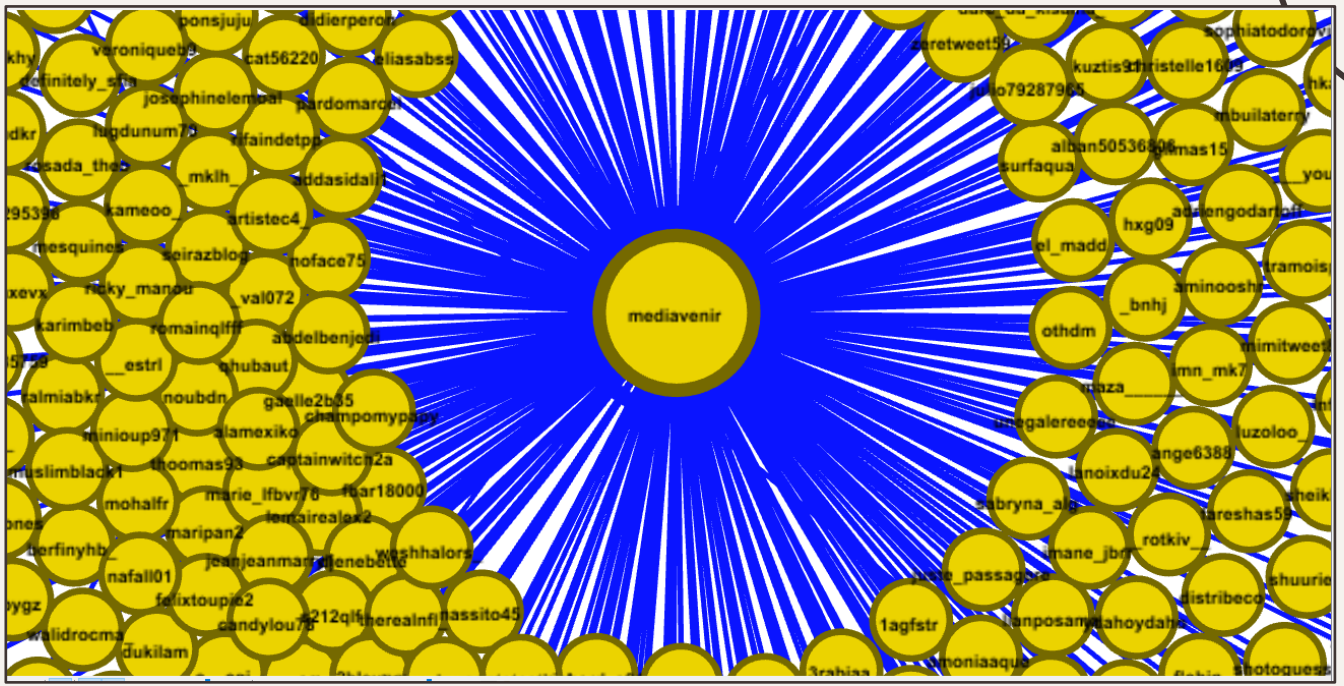
Community detection - Some Communities



Community detection - Some Communities



Community detection - Some Communities



Community detection