Web Science - Network Analysis Project

Analysing the Twitter Search Network and Facebook Like/Comment Network for Two Recent Movies: ***The Hobbit*** and ***The Interview***

Authors:

Georgiana Ciocirdel, 1428503

Mihai Varga, 1428505

Maxim Domentii, 1428506

Abstract

We are hereby analysing four different networks, from two social media portals, Twitter and Facebook. The networks are related to the movies '*The Hobbit*'[1] and '*The Interview*'[2]. The movie '*The Hobbit*' has first appeared on the 1st of December 2014 in the UK and in the month that followed in the rest of the world. The movie '*The Interview*' is quite controversial[3] and has faced many issues before being released solely in the US, first on the 11th of December 2014 and then during Christmas time. The Twitter networks we are analysing are directed, with edges linking users who have authored a post to users that have retweeted it. The Facebook networks are being analysed in three consecutive steps, marking important release dates around the world of the two movies; these are unimodal, undirected networks, with edges linking users that have liked or commented on the same post made by the official Facebook pages of the two movies. The analysis reveals a few important things about the four networks: the two Twitter networks differ a lot between them - although big, '*The Interview*' 's Twitter graph is a "young" one, meaning that authorities and hubs haven't yet been formed properly, the posts are sparse and with very few users retweeting more than one post. In comparison, '*The Hobbit*' 's Twitter graph has well defined authorities and most of its users have retweeted more than one popular tweet. The Facebook networks also differ: this time '*The Hobbit*' has more users liking and commenting on their posts, especially around the US release, while '*The Interview*' network is smaller and doesn't change much in time.

Collecting the data

Issues

Using existing tools

The internet is huge, so one could say data "is running rampant" in form of bits and bytes through the wires worldwide. Which is absolutely true, but one must also take into consideration the limitations of the existing APIs and the available data retrieving tools.

A few examples of tools that have failed would be: the social network analysis feature from Wolfram Mathematica, which claimed it can import data from networks such as Facebook, Twitter or Linkedin, but was utterly disappointing because of access failures; NameGenWeb is currently offline, but instead, they suggest to use Social Network Importer for Node XL. This was quite an easy tool to use, but the latest version, 1.9.4 had access issues for Facebook pages. Instead, we tried to use the older 1.9.2 version, which did work wonders for Facebook data, but was useless for Twitter, as it would only retrieve 26 statuses[4]. We have also tried to

[1] http://www.thehobbit.com/
[2] http://www.theinterview-movie.com/
[3] usmagazine.com/entertainment/news/sony-hack-interview-movie-controversy
[4] *Twitter status* = Twitter posts gathered under the JSON "status" field.

use NetVizz to retrieve Facebook page data, but it doesn't allow any limitations on the number of users to return, so Gephi wasn't able to properly parse such huge graphs.

Developing custom tools

When all failed, we decided to try to write our own code for retrieving data, so we have used raw API calls. First, we tried to build some simple queries with the Facebook Graph API[5] and Facebook Query Language[6] (FQL). We used an online tool called the Graph API Explorer[7] to test our queries, before diving into creating the app itself. However, we soon discovered that Facebook only returns results from users that have given the specific permissions to the app that tries to scrape information. So this was a dead end.

The Twitter API[8] is more permissive if one doesn't mind the limitations of it - which are extreme. We will immediately dig into more details about this. In order to test some simple Twitter queries, we used an online tool, similar to the Graph API Explorer, called the API Console[9]. Finally, we did manage to gather some information, so it was time we wrote some code to do the tedious labour automatically.

We discovered on Github the source code of the Bear Python-Twitter[10] API, which, as the authors describe it, "*provides a pure Python interface for the Twitter API*". In other words, we would only have to register an app on the Twitter Application Management[11] and then use our consumer and access token key and secret to be able to use the library's methods.

The light at the end of the tunnel (?)

Twitter

Now that it was possible to retrieve data from Twitter, we only had to save it in an appropriate format so that Gephi would be able to parse it. Gephi is able to import a number of file formats, one of them being the GDF format[12].

We wrote a Python script, which makes use of the Bear Python-Twitter API (mentioned above). We will now provide chunks from the script and highlight important lines:

The script starts with the following lines of code, which create an instance of the Api class with authentication included.

---

[5] https://developers.facebook.com/docs/graph-api
[6] https://developers.facebook.com/docs/reference/fql/
[7] https://developers.facebook.com/tools/
[8] https://dev.twitter.com/rest/public
[9] https://dev.twitter.com/rest/tools/console
[10] https://github.com/bear/python-twitter
[11] https://apps.twitter.com/
[12] http://gephi.github.io/users/supported-graph-formats/gdf-format/

```
api = twitter.Api(consumer_key='***',
      consumer_secret='***',
      access_token_key='***',
      access_token_secret='***')
```

As previously stated, the consumer and access token key and secret can be obtained by registering the app on the Twitter Application Management. We have replaced ours with '`***`' in the code above for obvious reasons.

One of Twitter's limitations is that one cannot gather search data from more than 7 days in the past. As we have made the queries on the 30th and 31st of December, we have queried data from 22.12.2014 to 29.12.2014.

When querying for search data, one can specify in the query what kind of posts the query should retrieve. The available types are `recent`, `popular` and `mixed`. However, when using the `recent` tag, the query cannot return more than 100 tweets and when using the `popular` tag, the query returns a varying, but small number of tweets.

We decided to query for the popular tweets, as it would get us data about more influential users. Therefore, our final query looks like this:

```
result = api.GetSearch('theinterview', until=until[i], since=since[i],
count=100, result_type='popular')
```

This gives us a nice JSON object with a lot of information about the most popular tweets. We decided to get the retweeters of each of the popular tweets. However, although the most popular of the tweets had around 2000+ retweeters, Twitter does not allow the user to make more than 60 GetRetweeters queries and no more than 180 GetUser queries (which yield the user info) during a time interval of 15 minutes[13]. A GetRetweeters query looks like this:

```
retweeters_id = api.GetRetweeters(tweet_json['id'])
```

and the GetUser query would be made like this:

```
for retweeter_id in retweeters_id:
     user = api.GetUser(retweeter_id)
```

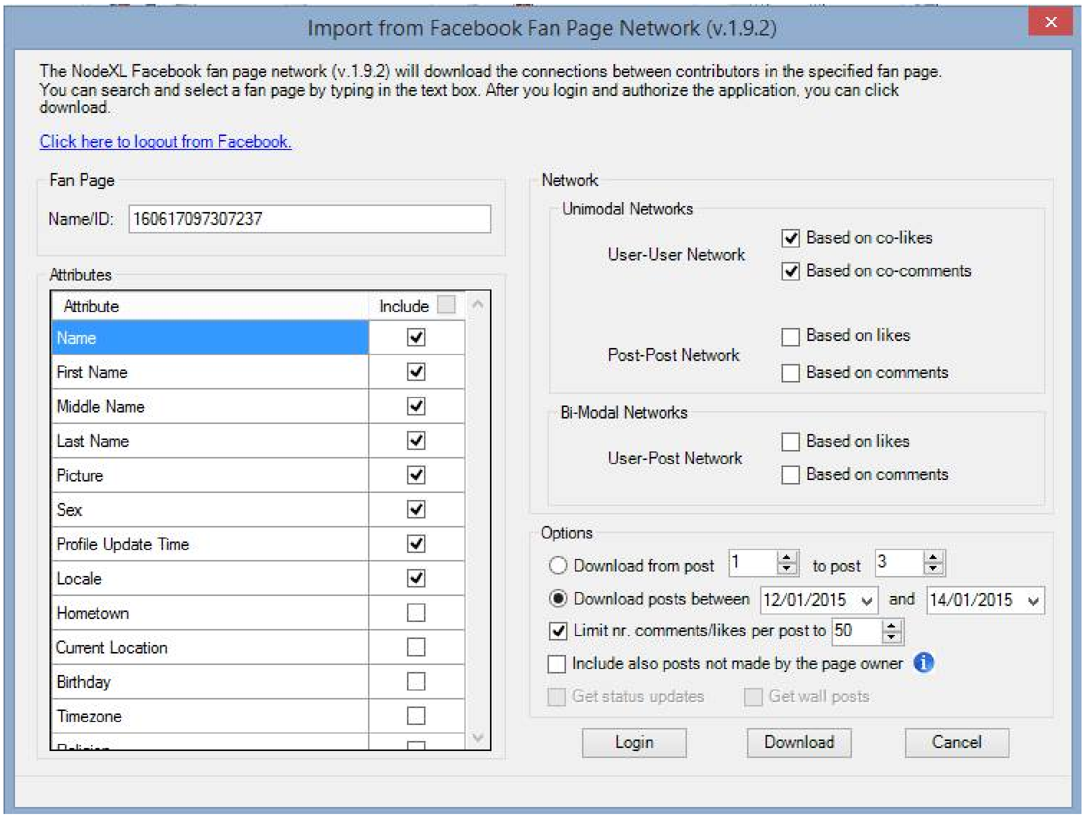
With all these limitations, we managed to retrieve part of the data that we needed about the two movies, '*The Hobbit*' and '*The Interview*'. We printed it in .gdf files with the appropriate format and then we were able to open them with Gephi, in order to analyse the networks.

In the .gdf files, we have created a node for each user that has retweeted a popular tweet and a node for each user that authored a popular tweet. We then added an edge between each user who has retweeted a tweet and each user that had authored the specific tweet.

[13] https://dev.twitter.com/rest/public/search (Rate Limits)

Facebook

Version 1.9.2 from Node XL was the tool we have used to retrieve data from the Facebook pages of the two movies, '*The Hobbit*' [14] and '*The Interview*'[15]. More precisely, we have used the Facebook Fan Page feature of the Social Network Importer for Node XL plug in:



The Hobbit

For '*The Hobbit*' we have taken into consideration various release dates around the world[16]. Therefore, we have collected data from between:

- 01.12.2014 - 02.12.2014, which marked the release dates in the UK and France for the London and Paris premieres;
- 04.12.2014 - 09.12.2014, which marked the release dates in the USA for the New York and Los Angeles premieres;
- 10.12.2014, which marked the release date in several European countries, including Austria, Germany, the Netherlands and the rest of France.

[14] https://www.facebook.com/TheHobbitMovie
[15] https://www.facebook.com/TheInterview
[16] http://www.imdb.com/title/tt2310332/releaseinfo

The Interview

For '*The Interview*' we have taken into consideration various release dates around the world[17]. Therefore, we have collected data from between:

- 11.12.2014, which marked the Los Angeles premiere in the USA;
- 24.12.2014, which marked the internet release in the USA;
- 25.12.2014 - 26.12.2014, which marked the release date in the rest of the USA.

As we all know, '*The Interview*' is quite a controversial movie, so there was no release so far in any other country than the US, where it is actually limited.

In the .graphml files, we have created a node for each user that has liked a post or commented on a post on the specific Facebook page. We then added an edge between each user who has liked/commented on posts on the page.

Facebook and Twitter network analysis

A brief introduction

In this part we are going to analyse the four networks formed with the data we have retrieved as presented in the previous part.

The two Facebook networks were taken from the official Facebook pages of the two movies, '*The Hobbit*' and '*The Interview*'. Each node represents a user and the links between two users are formed when both have liked or commented on the same post made by the page. Therefore, this is an undirected network.

The two Twitter networks were formed from the search results using the queries 'the hobbit' and 'the interview'. We have considered the most popular posts, i.e. the "trending topics". We have added a node to the network for each user that has either posted or retweeted a post and then added a directed edge from the user that has authored the tweet (the source) to the user that has retweeted it (the target). Therefore, this is a directed network.

In this section we will analyse the four networks by interpreting the visual layouts (the graph representation) and by combining certain metrics so as to obtain a mathematical insight into the data.

[17] http://www.imdb.com/title/tt2788710/releaseinfo

The Interview

The Twitter network

Raw measurements

The Twitter network for the movie '*The Interview*' is directed and consists of **7 872 nodes** and **8152 directed edges**. This suggests our network is not dense at all. The **average degree** is **1.036**, and the **graph density** is **0**. The longest graph distance between any two nodes in the network, i.e. the **diameter** of the graph, is **2**, with an **average path length** of **1.022**. The graph consists of **13 connected components**. The **average clustering coefficient (ACC)** for the graph is **0**, if we consider the directed network and, if we turn the graph into an undirected one, we obtain an ACC of **0.001**.

Metrics condensed

Metric	*Result*
Network Type	Directed
Number of Nodes	7872
Number of Edges	8152
Average degree	1.036
Network density	0
Diameter	2
Average Path Length	1.022
Connected Components	13
Average Clustering Coefficient (ACC) for the directed nework	0
ACC for the undirected network	0.001

Important nodes/users in the network

We have used the Fruchterman-Reingold Algorithm in order to produce a nice visual network of all the data we have gathered from Twitter.

Figure 1.1 shows the Fruchterman-Reingold layout of the directed Twitter network. Here we can already see why the graph is so sparse. First of all, we can easily notice how certain areas are denser than others - these are popular tweets that have been retweeted by many other users, so the central node points to many other nodes. There are many people who have retweeted *one* popular tweet, but there aren't many people who have retweeted *many* popular tweets.

Figure 1.2 shows how the network changes when we filter the nodes and only keep those with a degree (in-degree + out-degree) greater than a certain number.

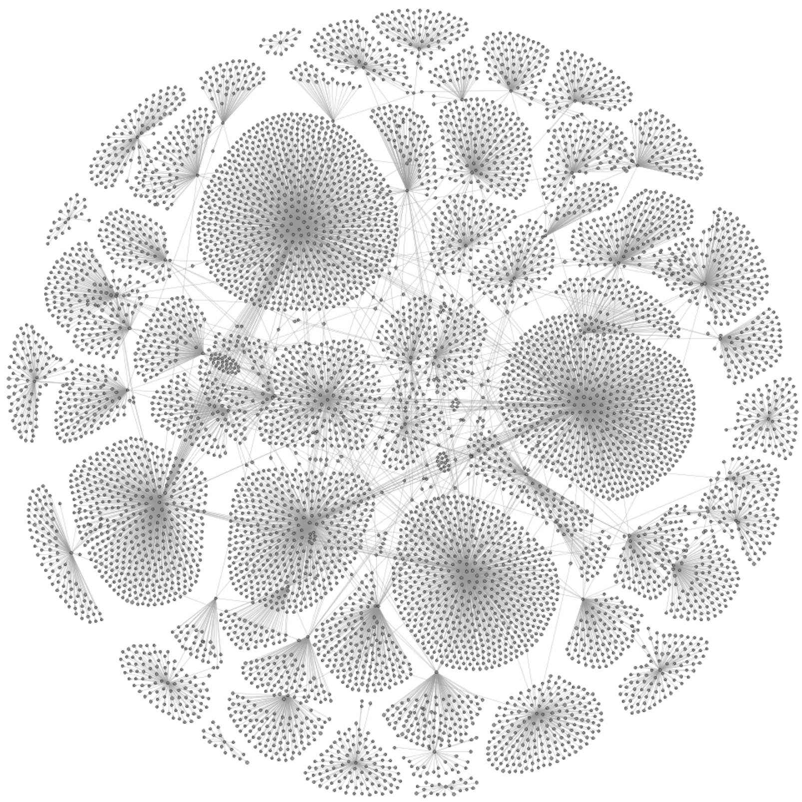
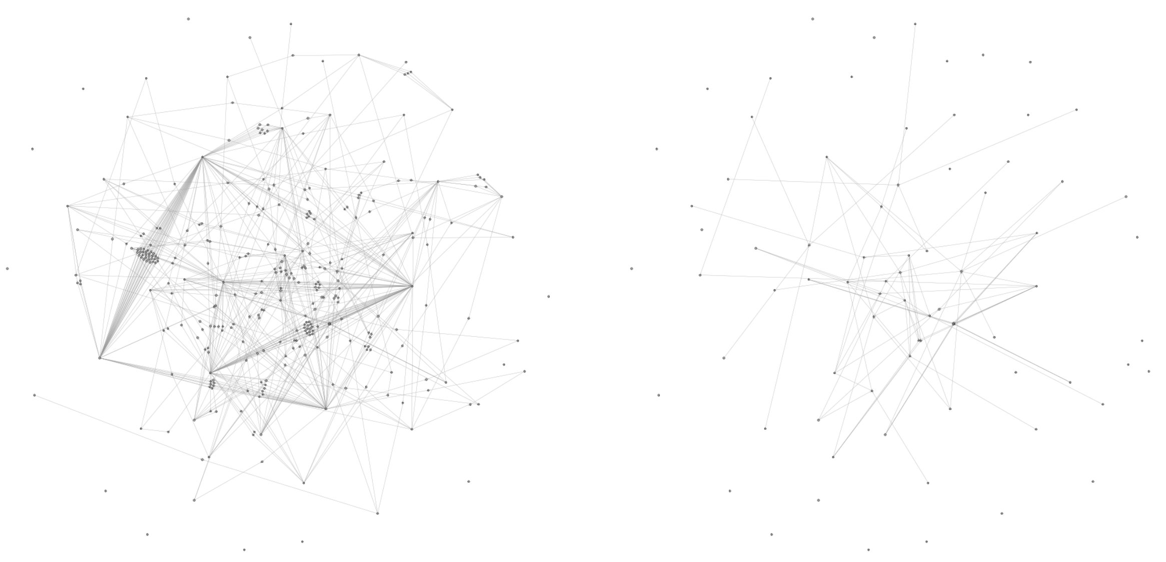


Figure 1.1



Degree ≥ 2 *Degree* ≥ 4

Figure 1.2

In Figure 1.3 we have enlarged the nodes with the highest *betweenness centrality* in the **undirected** network. This measures how often a node appears on shortest paths between other nodes in the network, in other words this highlights the Twitter users that have the greatest number of retweets or the Twitter users that get their tweets retweeted in cascade.

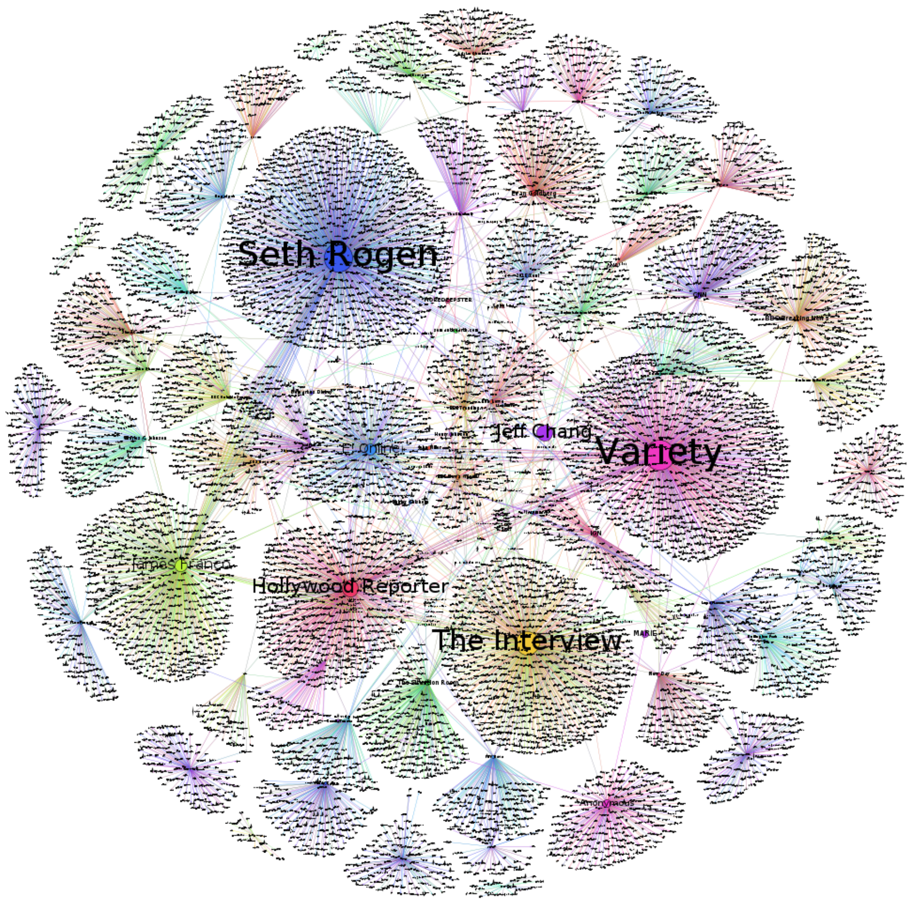


Figure 1.3

Unsurprisingly, the nodes with the greatest betweenness centrality correspond to users like Seth Rogen and James Franco, which play the main characters in the movie, The Interview, the official Twitter user of the movie, Hollywood Reporter, an American media publication[18] or Variety[19], an American magazine.

In Figure 1.4 we can see the nodes in the graph colored according to the number of followers the user has and the nodes with the most followers have been enlarged. It doesn't really come as a surprise that the user with the most followers, CNN (15 268 965 followers), doesn't actually have a high betweenness centrality in this network - this only means that their tweets

[18] http://en.wikipedia.org/wiki/The_Hollywood_Reporter
[19] http://en.wikipedia.org/wiki/Variety_%28magazine%29

were not as viral as those of the users above. A closer look tells us that *none* of the users with a high betweenness centrality makes it to the top when considering the number of followers.

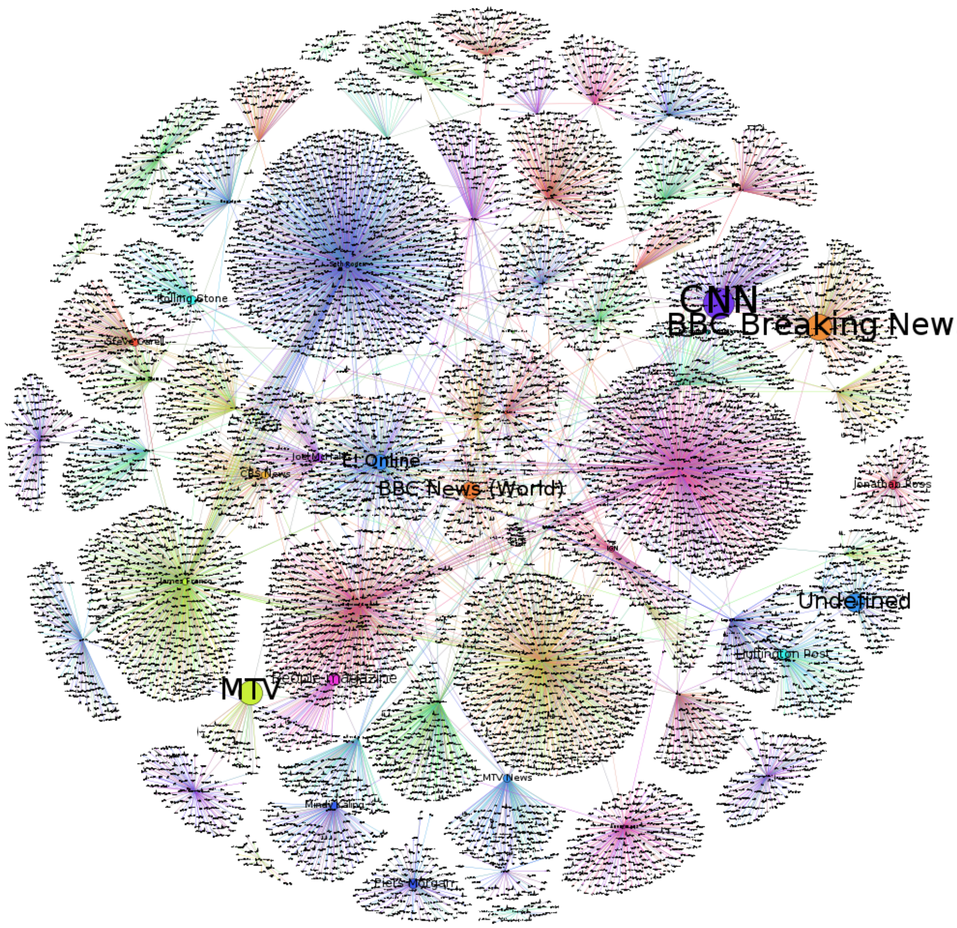


Figure 1.4

Hubs and authorities

In Figure 1.5 we have enlarged the nodes with the highest *authority* and *hub* values. This algorithm is used by Twitter to suggests users to follow[20]. A hub in the Twitter world represents a user that retweets other important (authoritative) users many times, so it is a “trustworthy second-hand” source of information; an authority is a user that gets retweeted many times, especially by important hubs, so authorities are “trustworthy first-hand” sources[21].

[20] http://en.wikipedia.org/wiki/HITS_algorithm
[21] http://survivingmyphd.blogspot.co.at/2010/10/authorities-and-hubs-in-twitter.html

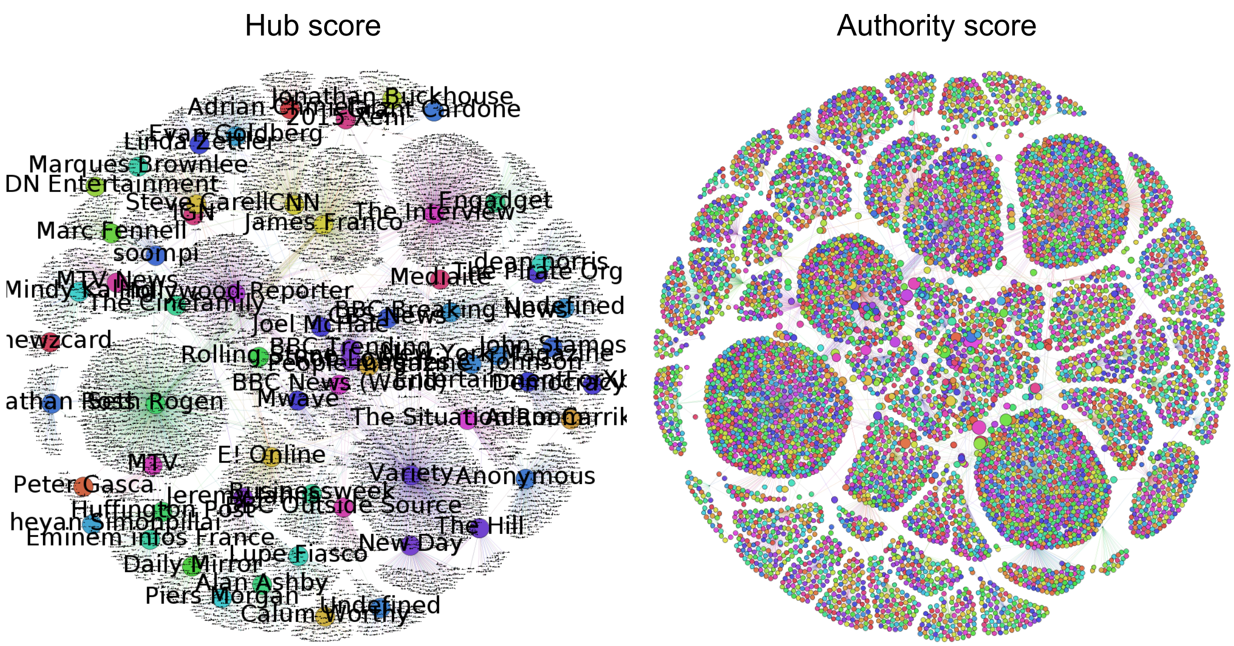


Figure 1.5

The fact that there are many hubs comes as no surprise (everyone likes to retweet important users), but in this network it seems that everybody is an authority - does everyone really retweet everyone about '*The Interview*' ?

Communities in the network

A community on Twitter could be defined as a set of users that have more links within the set than outside of it, in other words, users that share quite the similar views or prefer similar tweets and therefore, retweet them.

Figure 1.6 shows the network colored based on different communities and the users with the highest numbers of followers have been enlarged. These results have been yielded when determining communities with a 20.0 modularity (we wanted to highlight only the most important communities). Figure 1.7 shows the network with the same features, but now the communities can be seen even better.

We can see from these layouts that the communities are rather fuzzy and intertwined, but a very important thing that can be deducted is the fact that one or more users with a huge number of followers (influential users) are part of each of the biggest communities highlighted (the pink, the blue or the green community). Moreover, we can now clearly see how the ideas are spread from one small group to the other: there are users that retweet one tweet in their small group; one of the retweeters also gets retweeted by one other user, which then gets retweeted by many other users in his/her small group. This is, in fact, the very way the Twitter network functions.

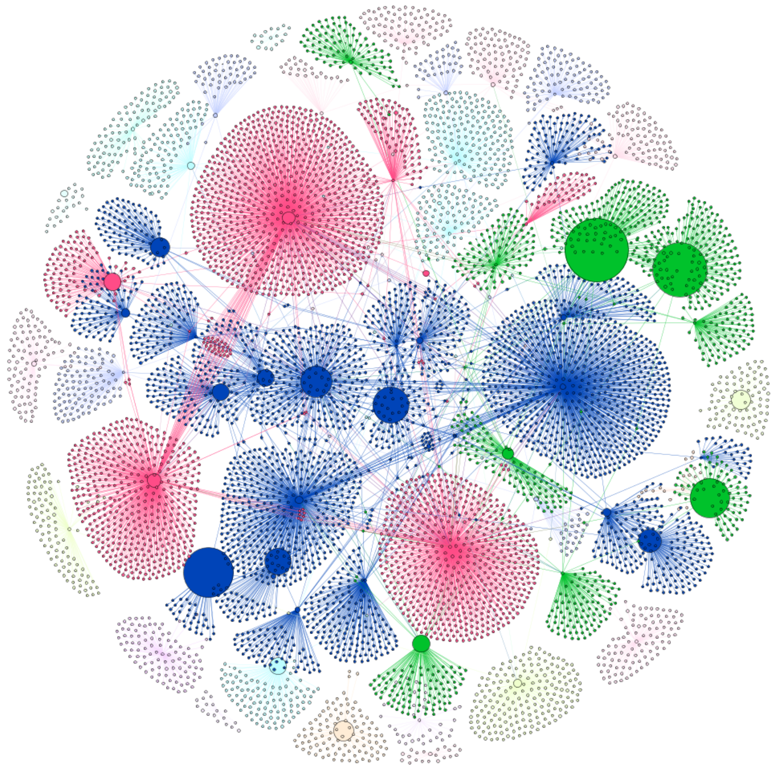


Figure 1.6

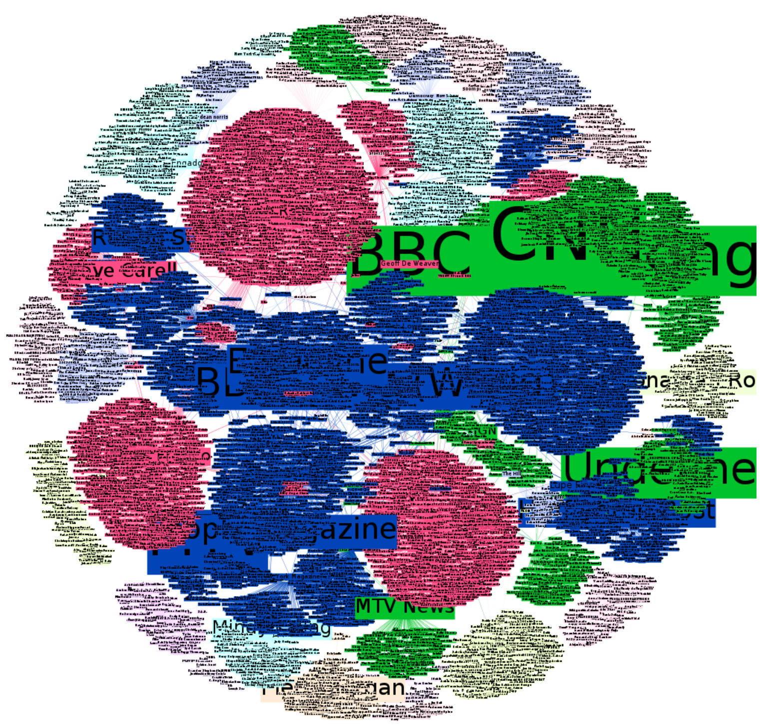


Figure 1.7

Connected components

There are 13 connected components in the graph, which are highlighted in Figure 1.8.

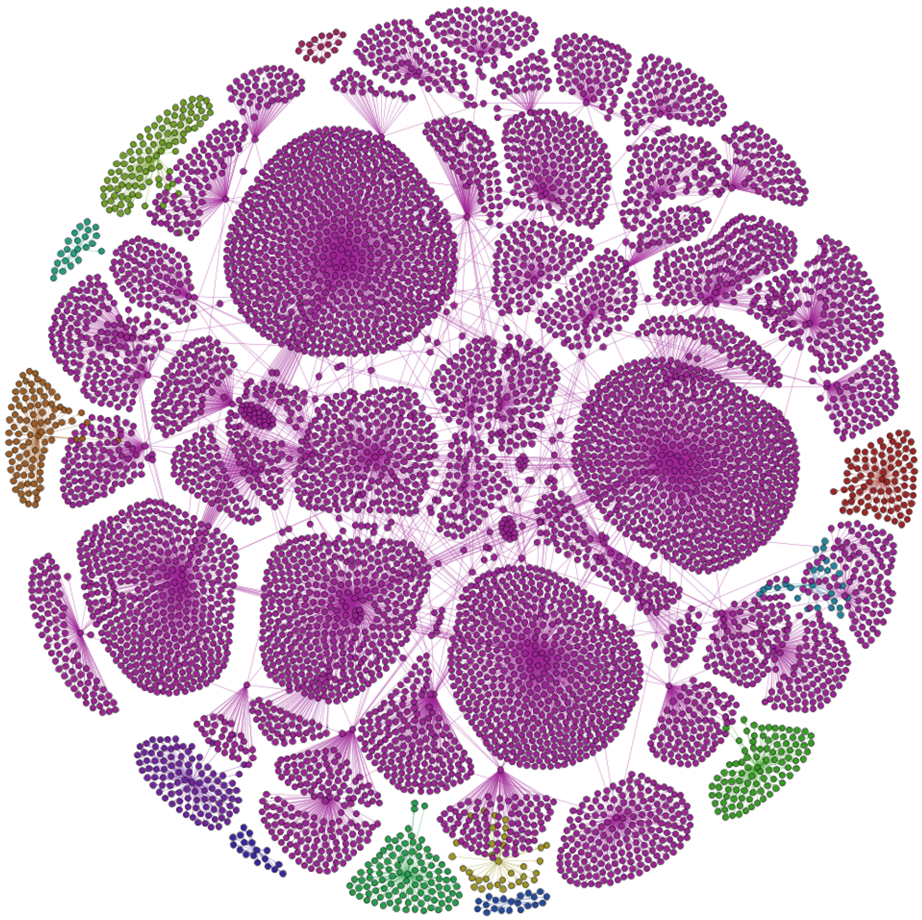


Figure 1.8

The Facebook network

Raw measurements

We have analysed the Facebook network for the movie '*The Interview*' for three consecutive periods of time: between the 11th and the 12th of December, on the 24th of December and between the 25th and 26th of December, which marked important release dates in the US.

Unlike Twitter, the Facebook network for the movie '*The Interview*' is undirected and a lot smaller when in comes to the number of edges. In it also very different with regards to the edge meaning. The metrics for the two instances of the network are better observed in the condensed way (see below).

Metrics condensed

Metric	*Network*	*Result*
Network Type	11-12.12.2014	Undirected
	24.12.2014	Undirected
	25-26.12.2014	Undirected
Number of Nodes	11-12.12.2014	301
	24.12.2014	263
	25-26.12.2014	388
Number of Edges	11-12.12.2014	7742
	24.12.2014	6689
	25-26.12.2014	10609
Average degree	11-12.12.2014	51.442
	24.12.2014	50.87
	25-26.12.2014	54.686
Network density	11-12.12.2014	0.171
	24.12.2014	0.194
	25-26.12.2014	0.141
Diameter	11-12.12.2014	4
	24.12.2014	2
	25-26.12.2014	4
Average Path Length	11-12.12.2014	2.387
	24.12.2014	1.494
	25-26.12.2014	2.017
Connected Components	11-12.12.2014	1
	24.12.2014	3
	25-26.12.2014	2
Average Clustering Coefficient	11-12.12.2014	0.948
	24.12.2014	0.962
	25-26.12.2014	0.936

Metrics in time

The diagrams in Figure 2.1 show how the metrics have changed in time:

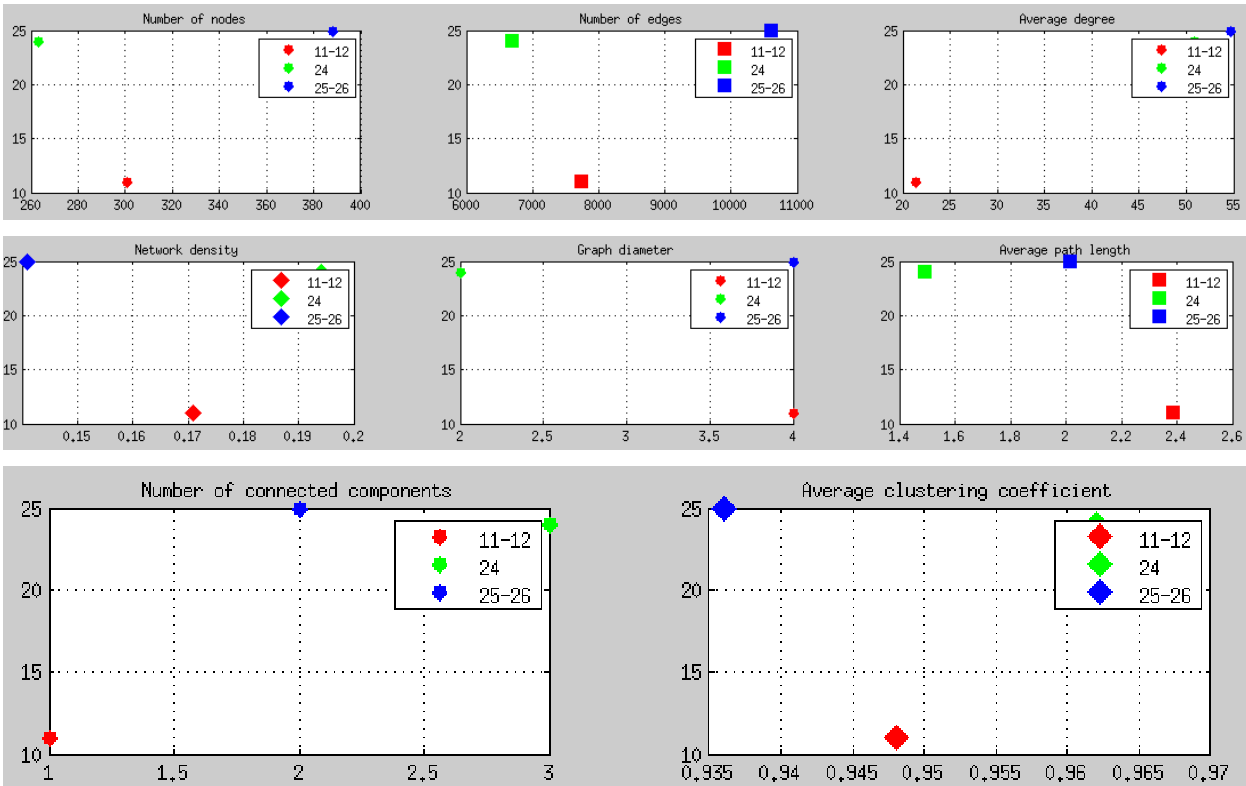


Figure 2.1

As one can notice, the network is smaller but somewhat denser on the 24th of December, which marks the Christmas Eve.

Important nodes/users in the network

We have used the Fruchterman-Reingold Algorithm so as to produce a visual representation of the network in time.

Figure 2.2 shows the Fruchterman-Reingold layout of the undirected Facebook network.

In Figure 2.3 we have enlarged the nodes with the highest *betweenness centrality* in the network. For the Facebook network this shows the users that have liked or commented on more than one post in the same day, so users who are "brokers" between two or more clusters in the network.

One can easily notice that there is no major difference between the three instances of the network. In all cases, there are users who have liked more than one post and users who have liked only one, so the bigger nodes represent "brokers" between the communities that have formed around each post.

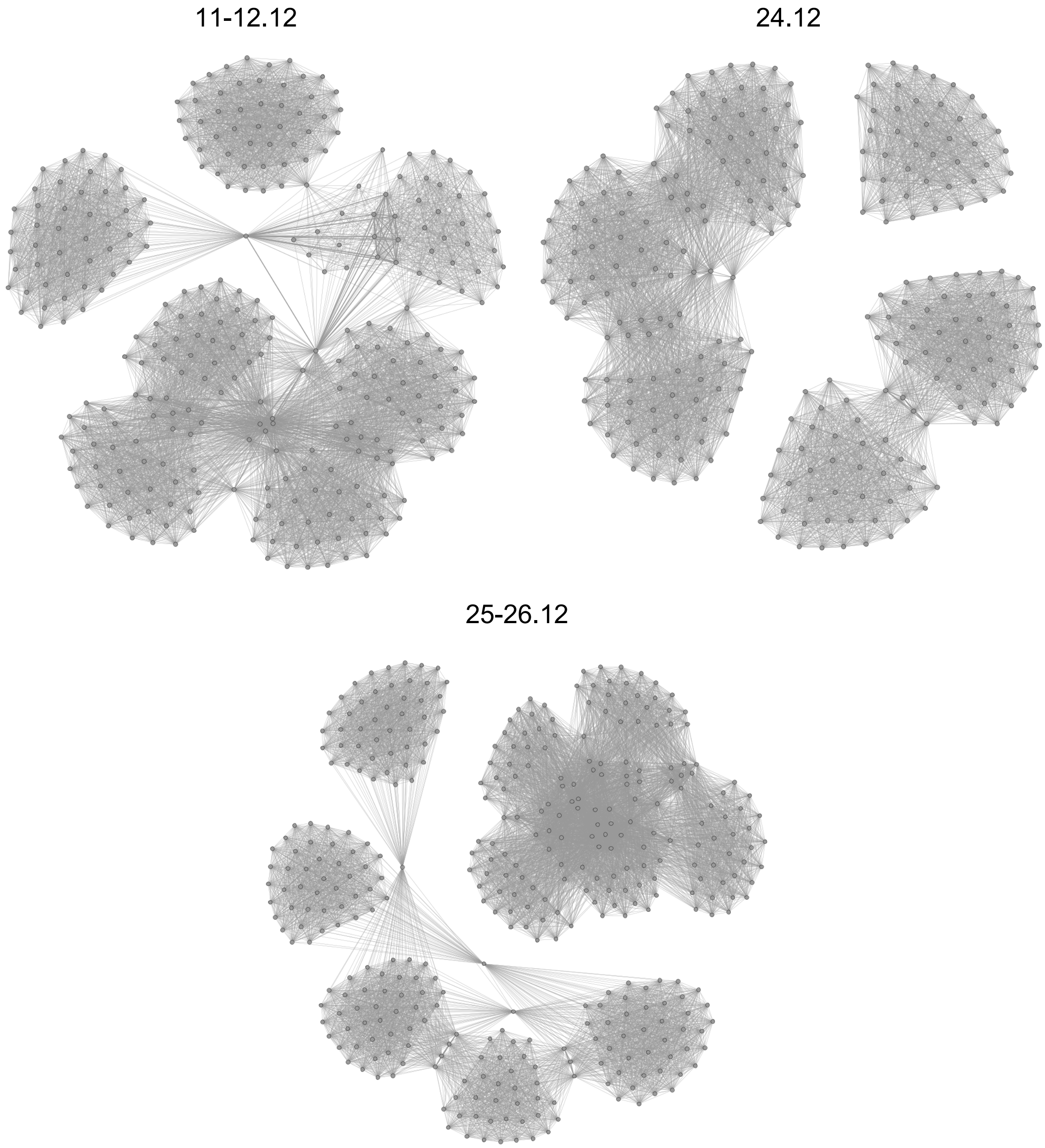


Figure 2.2

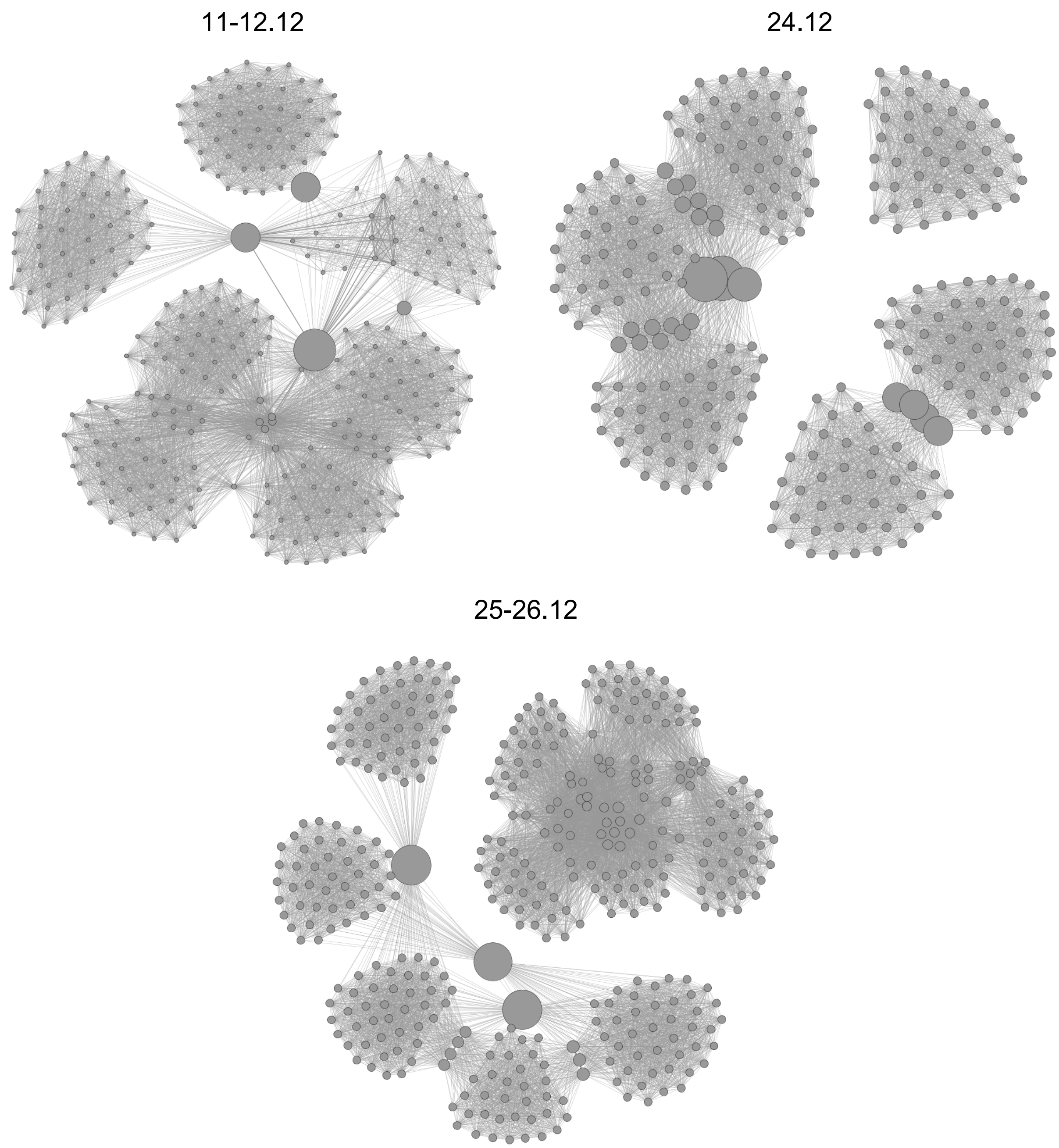


Figure 2.3

In Figure 2.4 we have enlarged the nodes with the highest *eigenvector centrality* in the network. For the Facebook network this is a measure of a user's importance in the network, based on its connections, so this metric would highlight the "central" users that have liked or commented on a post that has received a lot of attention from many other people in the network. Yet again, there is no obvious difference between the three states of the network: all three networks present one or more popular posts, which have been liked or commented on by users who have also liked or commented on other posts (popular or not).

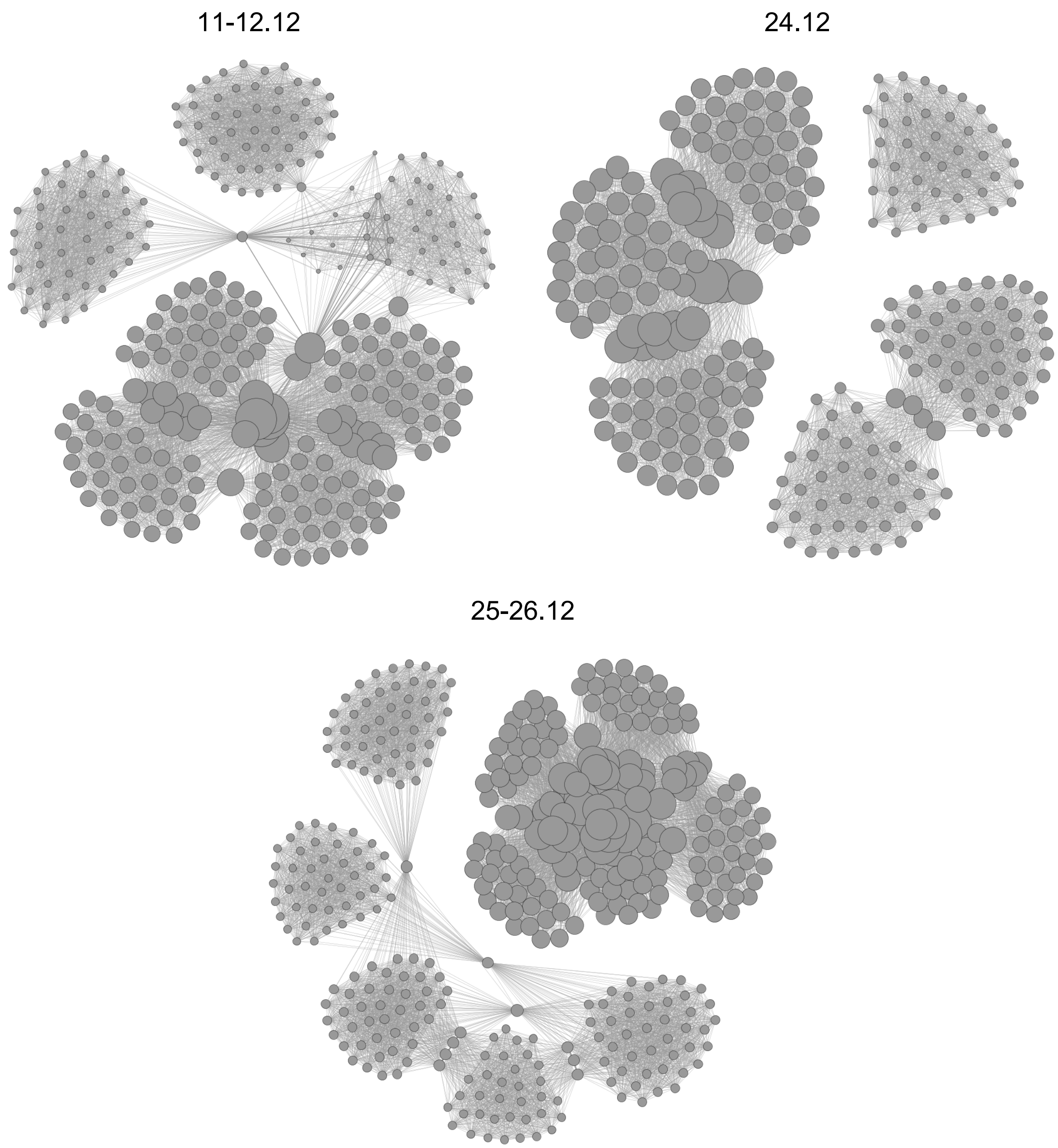


Figure 2.4

Communities in the network

A community on the Facebook page of the movie could be defined as a set of users that have liked or shared one or more similar posts.

Figure 2.5 shows the network colored based on different communities and the users with the highest betweenness centrality have been enlarged. These results have been yielded when determining communities with a 1.0 modularity.

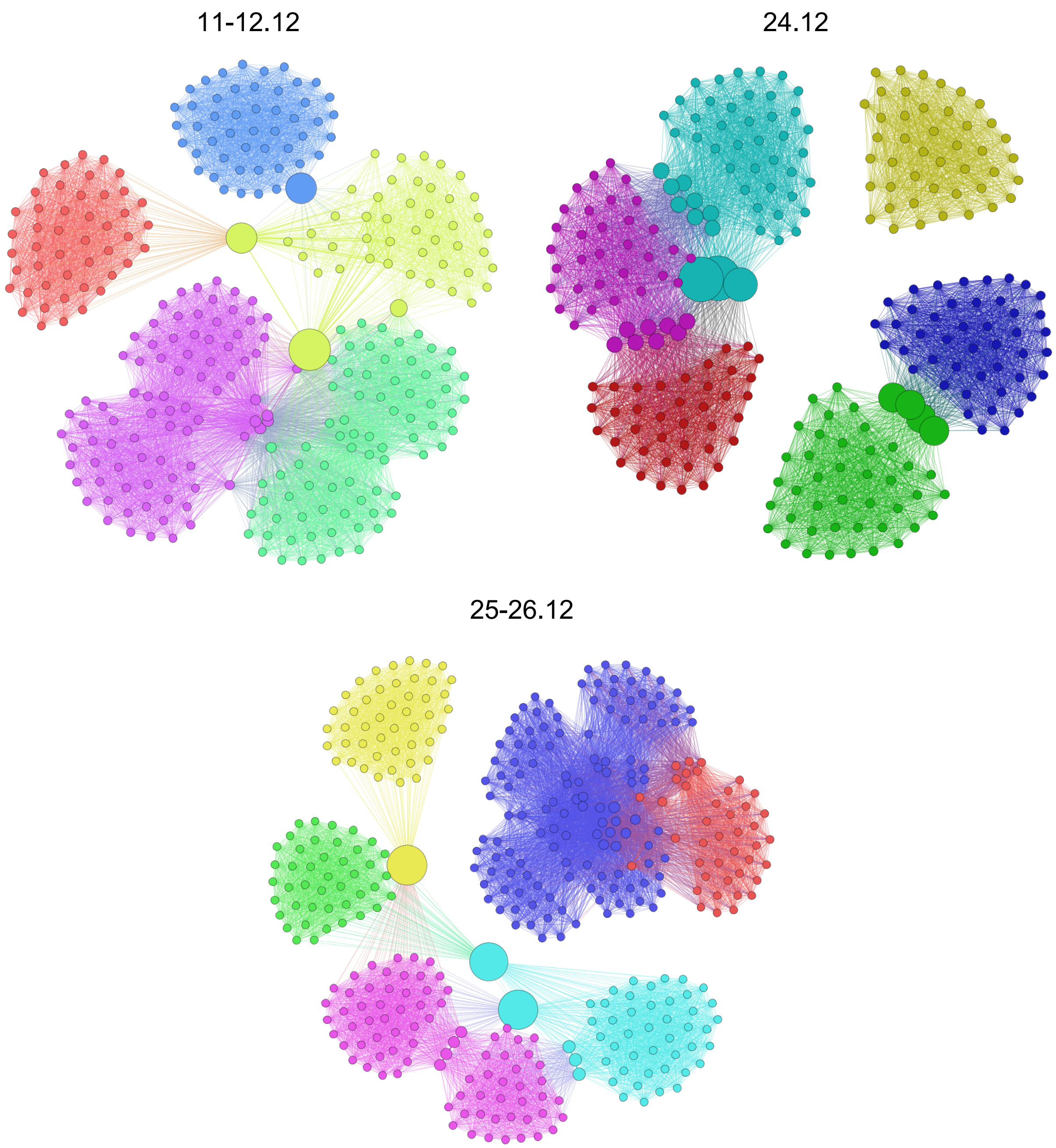


Figure 2.5

We can see from these layouts that the communities are very well defined and very well separated from one another. It would make sense to think that one user who has liked the Facebook page of the movie and has then liked or commented on a post, has maybe received the post on their timeline or has browsed through the page's previous posts. It would be safe to consider that someone who has liked *one* post might have liked *at least another one*, but if we look at the diagrams, it is clear that there aren't many users who have done this. However, it also makes sense to consider that a user who has liked a post will not like another one, due to the manner of how the timeline shows the posts (it doesn't often show two updates of the same page in a row) or simply because one page cannot publish that many important notes per day (or during a two-day timeline).

Connected components

The connected components are displayed in Figure 2.5.

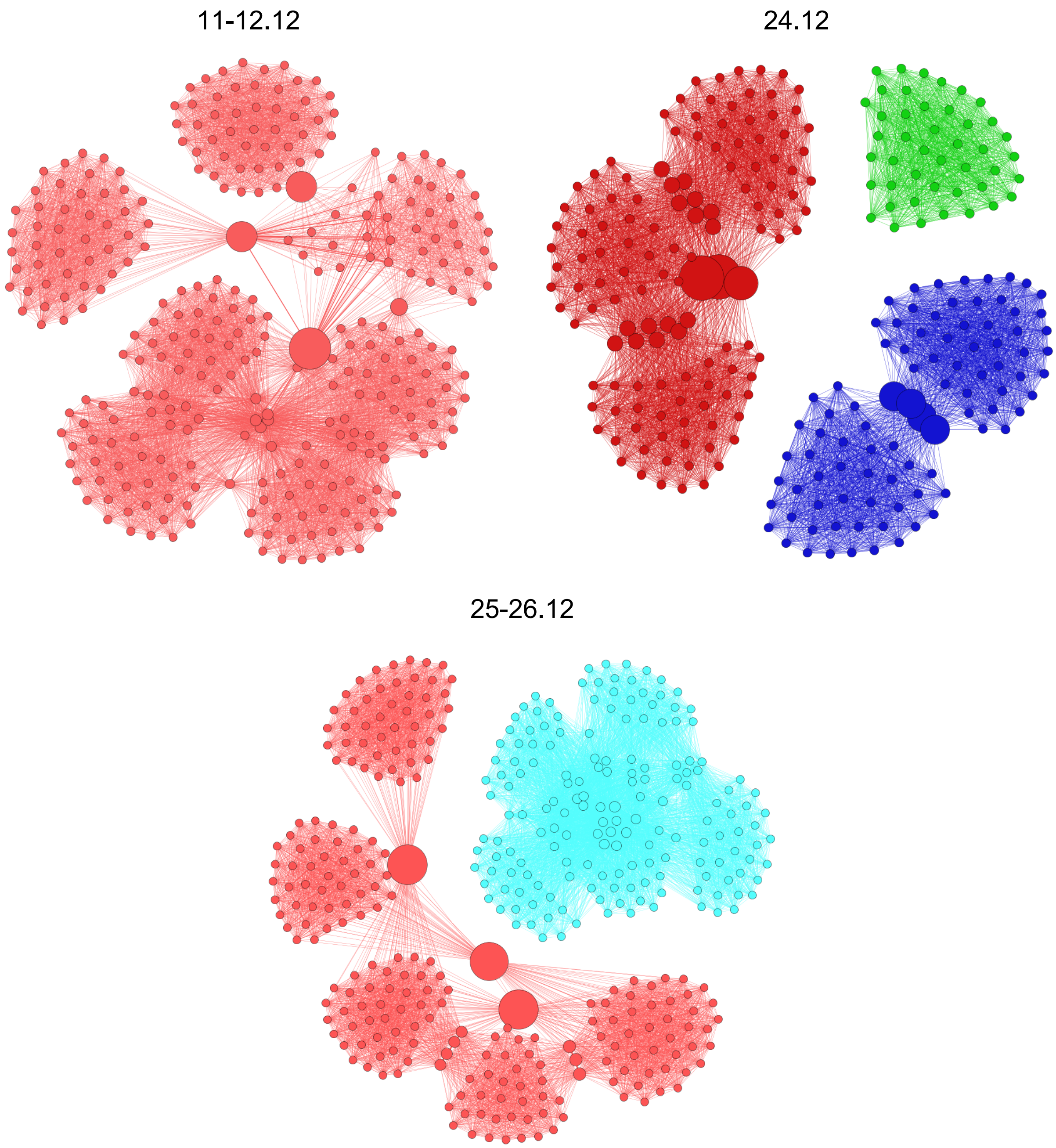


Figure 2.5

The network obtained between the 11th and the 12th presents only one connected component. This might suggest that the page's posts were all very interesting and/or important, so there were people who have liked or commented on more than one.

The Hobbit

The Twitter network

Raw measurements

The Twitter network for the movie '*The Hobbit'* is directed and consists of **2 646 nodes** and **2 908 directed edges**. Just like the previous Twitter network, this one is also sparse: the **average degree** is **1.099**, and the **graph density** is **0**. The **diameter** of the graph is **2**, with an **average path length** of **1.031**. The graph consists of **5 connected components**. The **average clustering coefficient (ACC)** for the graph is **0.001**, if we consider the directed network and, if we turn the graph into an undirected one, we obtain an ACC of **0.028.**

This network is smaller but "more connected" than the Twitter network of the movie '*The Interview*', but this comes as no wonder, because the '*The Hobbit*' has been released at the beginning of December, whereas '*The Interview*' has only just appeared.

Metrics condensed

Metric	*Result*
Network Type	Directed
Number of Nodes	2646
Number of Edges	2908
Average degree	1.099
Network density	0
Diameter	2
Average Path Length	1.022
Connected Components	5
Average Clustering Coefficient (ACC) for the directed nework	0.001
ACC for the undirected network	0.028

Important nodes/users in the network

We have again used the Fruchterman-Reingold Algorithm in order to produce a nice visual network of all the data we have gathered from Twitter.

Figure 3.1 shows the Fruchterman-Reingold layout of the directed Twitter network. This Twitter network has a similar visual structure as the previous one. We can easily see on the margin of the circle users whose posts get retweeted only once. However, in the center of the circle, there is quite a dense network of users that got their tweets shared multiple times in a row.

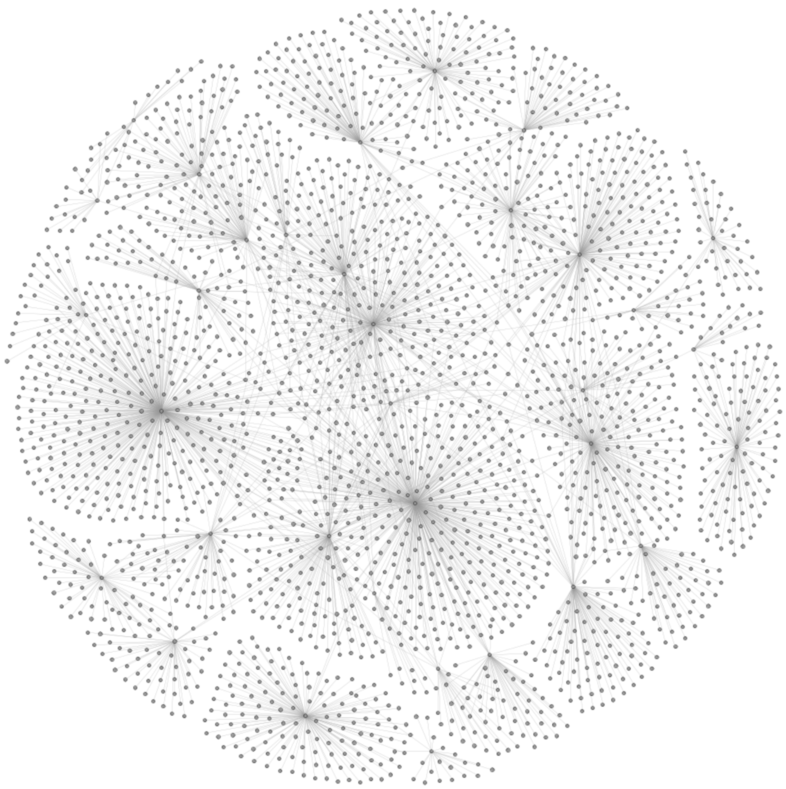
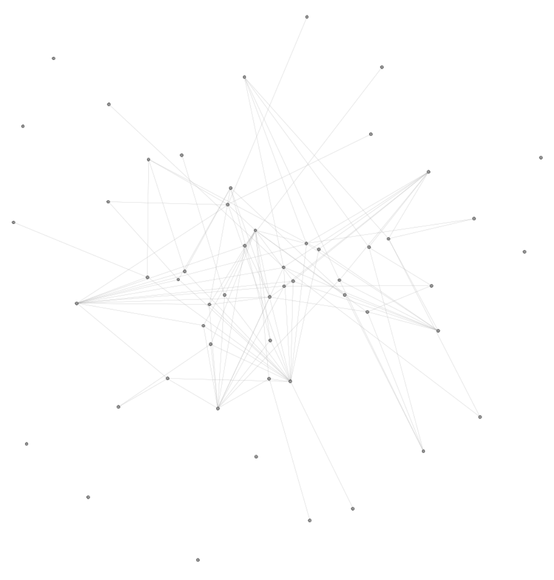


Figure 3.1



Degree ≥ 2 *Degree* ≥ 4

Figure 3.2

Figure 3.2 shows how the network changes when we filter the nodes and only keep those with a degree (in-degree + out-degree) greater than a certain number. When comparing with the Twitter network of '*The Interview*', we can easily see that '*The Hobbit*' has a denser network around the center, with nodes with higher degrees.

In Figure 3.3 we have enlarged the nodes with the highest *betweenness centrality* in the **undirected** network.

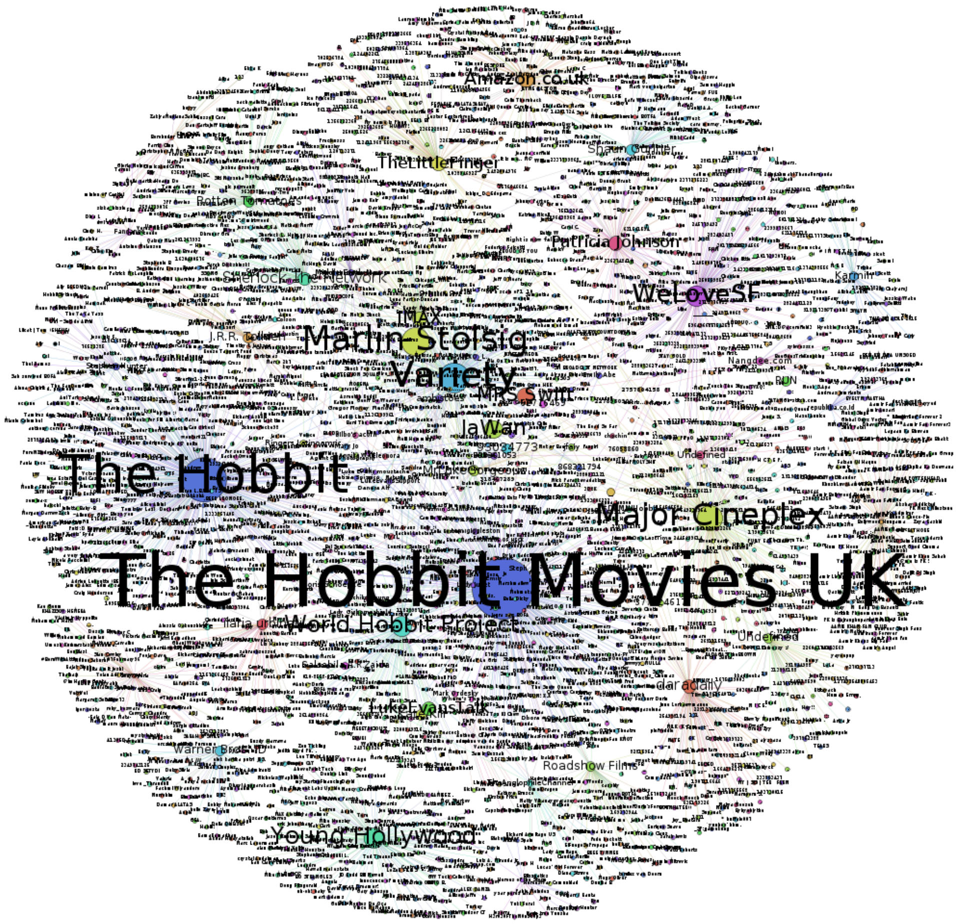


Figure 3.3

The nodes with the greatest betweenness centrality correspond to users like The Hobbit Movies UK, The Hobbit, Major Cineplex[22], an operator of movie theaters in Thailand, again the Variety magazin and a user named Marlin Stolsig, which we have no idea who it is. Unlike the Twitter network of '*The Interview*', the most popular users in this network are spread across the world and are not only localized in the US.

[22] http://www.majorcineplex.com/

In Figure 3.4 we can see the nodes in the graph colored according to the number of followers the user has and the nodes with the most followers have been enlarged. This time we can determine intersections between the users with the highest betweenness centrality and the users with most followers. Users that belong to both groups would be The Hobbit, Young Hollywood, Major Cineplex, Variety, IMAX. One user that does not belong to both groups is Rotten Tomatoes.

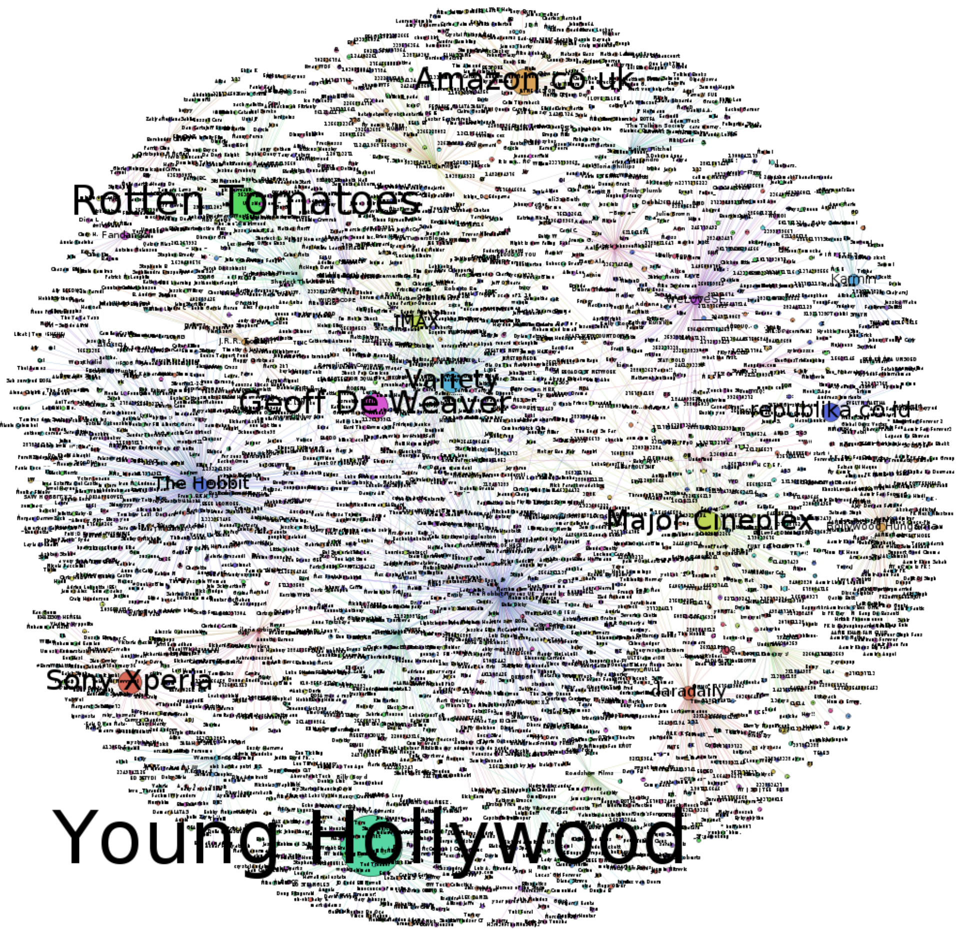


Figure 3.4

Hubs and authorities

In Figure 3.5 we have enlarged the nodes with the highest authority and hub values. Unlike the previous Twitter network, where everyone seemed to be an authority, here we can see that only the users The Hobbit and The Hobbit Movies UK have the highest authority in the network, so the information they share is the most important one, with respect to the users' preferences. There are many hubs in the network, including the previous two authorities and other important users that have high betweenness centrality.

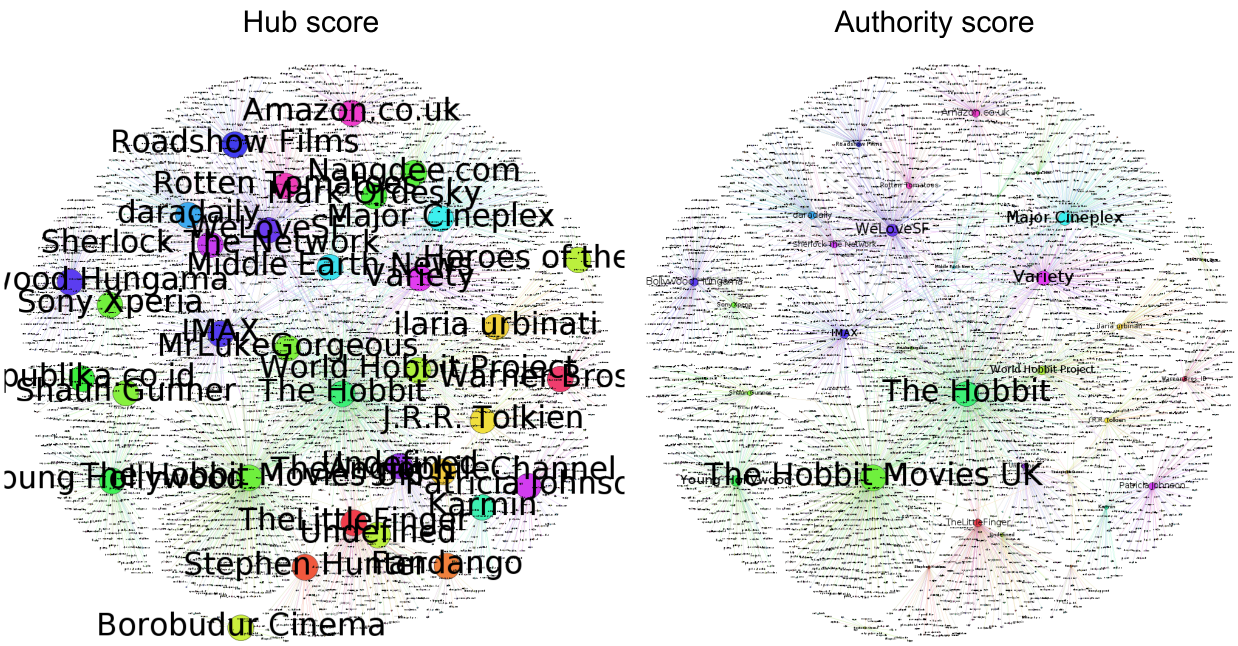


Figure 3.5

Communities in the network

Figure 3.6 shows the network colored based on different communities and the users with the highest numbers of followers have been enlarged. These results have been yielded when determining communities with a 20.0 modularity (we wanted to highlight only the most important communities). Figure 3.7 shows the network with the same features, but now the communities can be seen even better.

We can see from these layouts that the communities are again intertwined, but this time there are less communities than in the previous network. Just like before, one or more users with a huge number of followers (influential users) are part of all of the biggest communities highlighted.

Connected components

This time there are only 5 connected components in the graph, which are highlighted in Figure 3.8. It is very clear how this components overlap the previous communities.

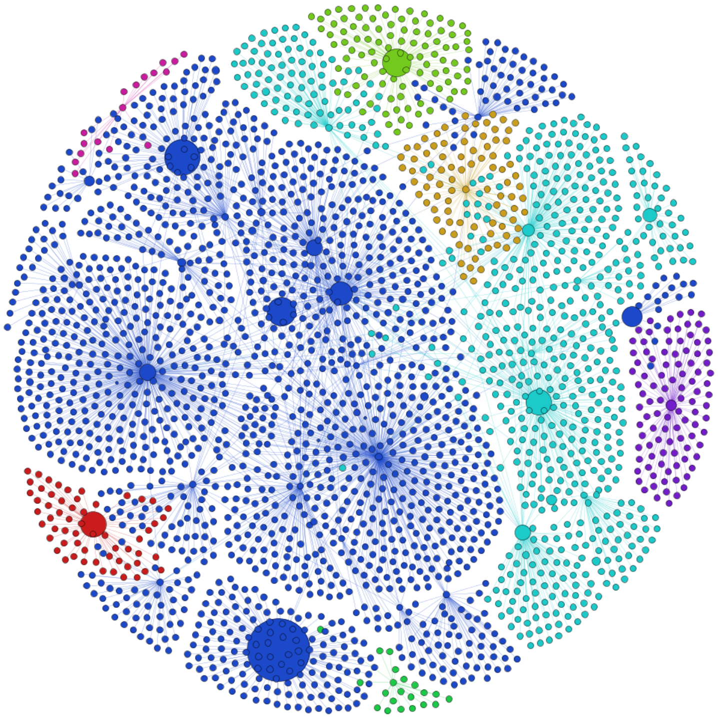


Figure 3.6

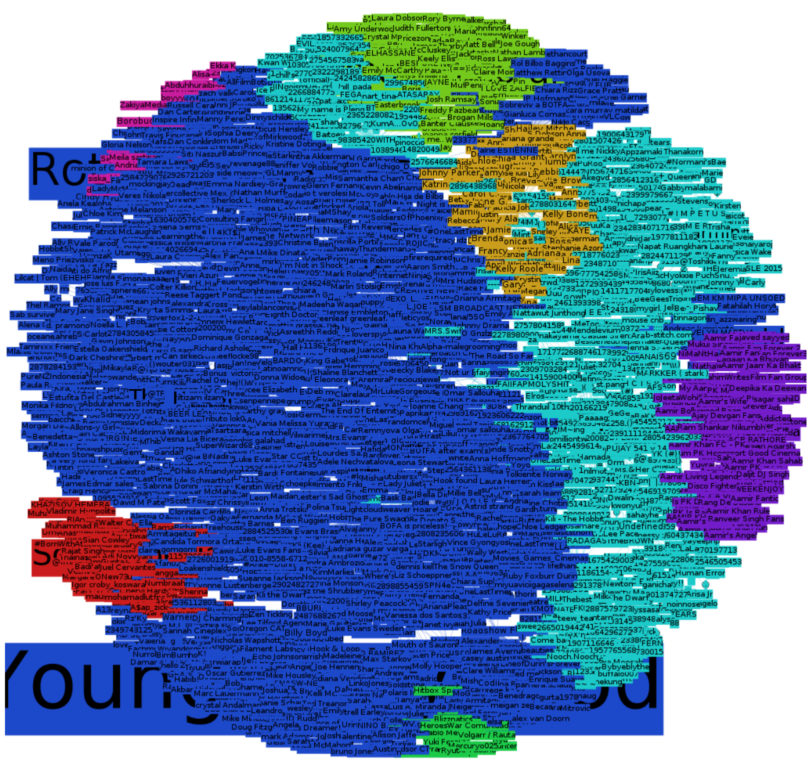


Figure 3.7

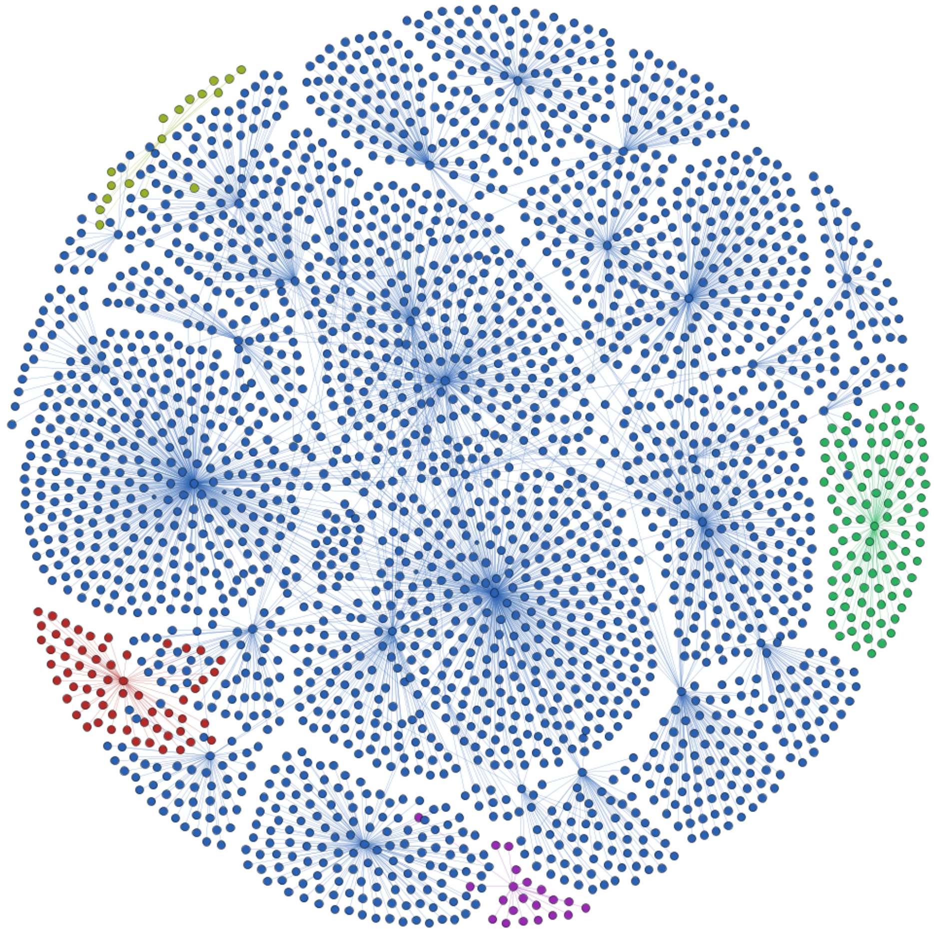


Figure 3.8

The Facebook network

Raw measurements

We have analysed the Facebook network for the movie '*The Hobbit*' for three consecutive periods of time: between the 1st and 2nd of December, between the 4th and 9th of December and between the 10th and 12th of December, which marked important release dates around the world.

Metrics condensed

Metric	*Network*	*Result*
Network Type	01-02.12.2014	Undirected
	04-09.12.2014	Undirected
	10-12.12.2014	Undirected
Number of Nodes	01-02.12.2014	468
	04-09.12.2014	854
	10-12.12.2014	290
Number of Edges	01-02.12.2014	12299
	04-09.12.2014	23251
	10-12.12.2014	7110
Average degree	01-02.12.2014	52.56
	04-09.12.2014	54.452
	10-12.12.2014	49.034
Network density	01-02.12.2014	0.113
	04-09.12.2014	0.046
	10-12.12.2014	0.17
Diameter	01-02.12.2014	3
	04-09.12.2014	5
	10-12.12.2014	2
Average Path Length	01-02.12.2014	1.658
	04-09.12.2014	2.484
	10-12.12.2014	1.561
Connected Components	01-02.12.2014	5
	04-09.12.2014	2
	10-12.12.2014	3
Average Clustering Coefficient	01-02.12.2014	0.95
	04-09.12.2014	0.951
	10-12.12.2014	0.991

Metrics in time

The diagrams in Figure 4.1 show how the metrics have changed in time:

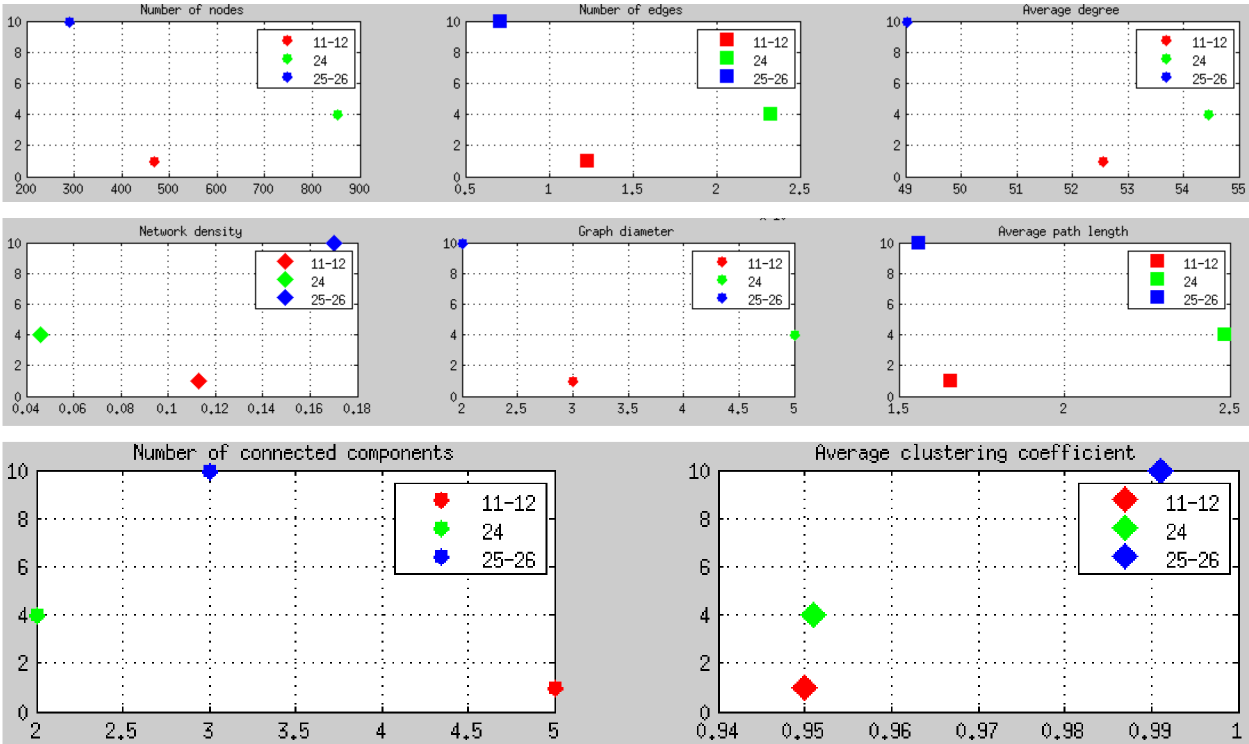


Figure 4.1

Important nodes/users in the network

We have used the Fruchterman-Reingold Algorithm so as to produce a visual representation of the network in time.

Figure 4.2 shows the Fruchterman-Reingold layout of the undirected Facebook network. As both the metrics and the visualizations suggest, the three instances of the network are different from one another: we can see that the first two networks are more “connected” than the last one, meaning that in the first two networks (and more in the second than in the first), there are many users who have liked or commented on more than one post, so the clusters that have formed around each post are linked to one another. Let us not forget that the movie has premiered in London and Paris on the 1st and 2nd of December, then during the 4th and 9th it has premiered in the US and between the 10th and 12th it has been released in other parts of Europe. Perhaps the three networks built around these dates suggest that ‘*The Hobbit*’ was more popular in the US than in Europe.

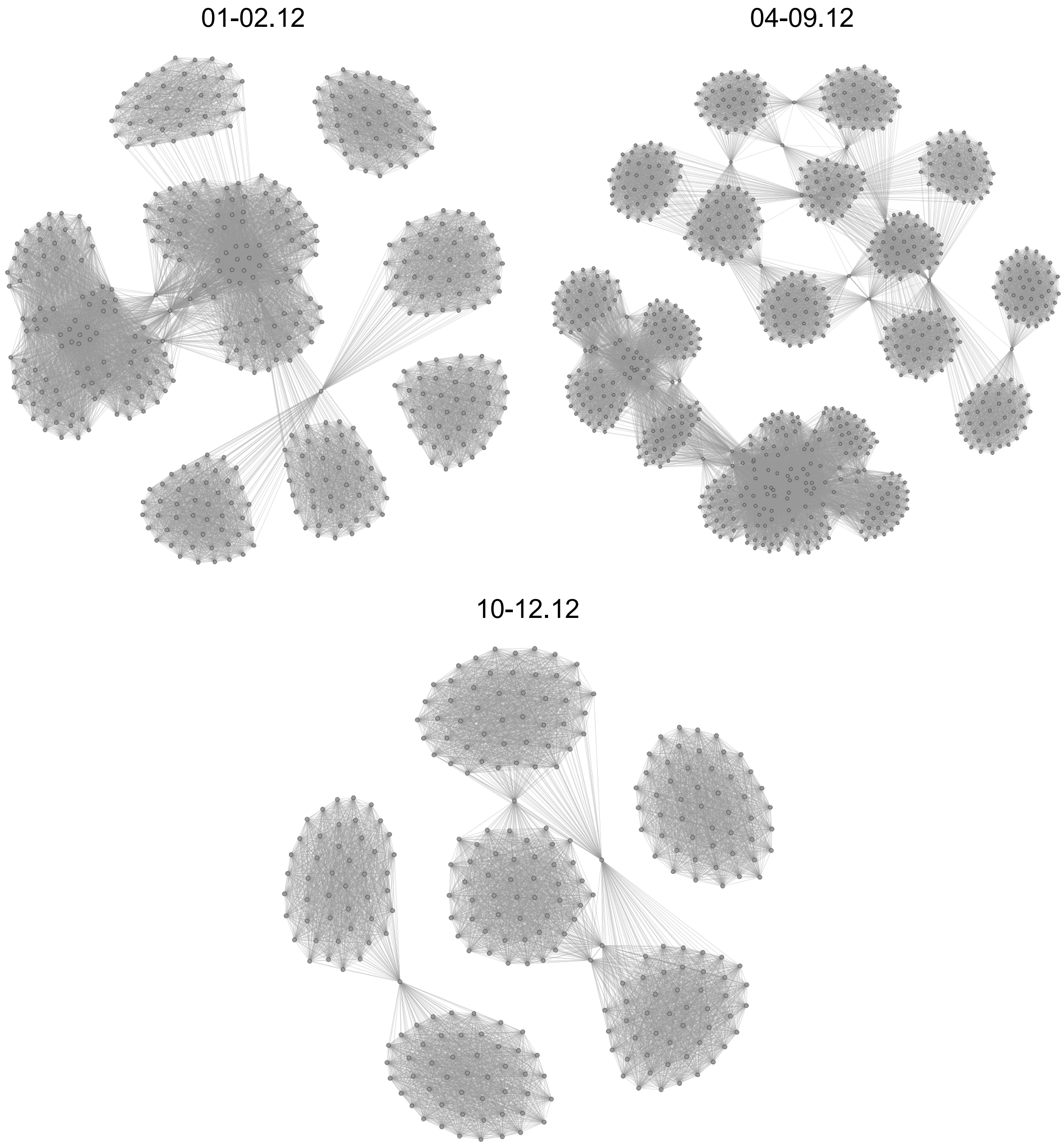


Figure 4.2

In Figure 4.3 we have enlarged the nodes with the highest *betweenness centrality* in the network. We can now see that there are some differences between the networks. The “brokers” in the second network are smaller in size, which means that less shortest paths pass through them than in the other networks. This happens because there are more central nodes in the second network than in the other two, so “the traffic” is split among them. Again, this suggests that the Facebook page of the movie has become more popular during the release dates in the US than in Europe.

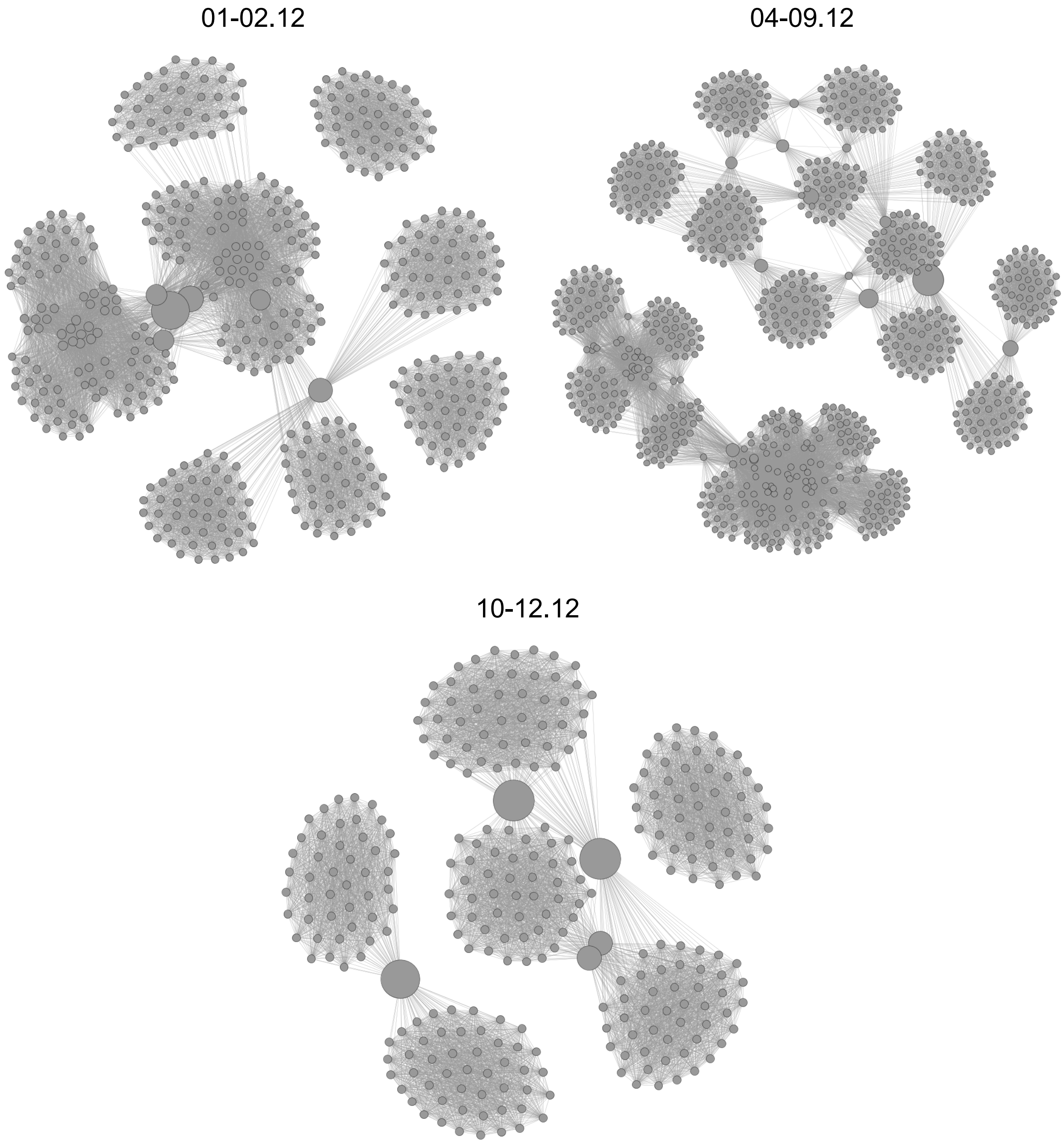


Figure 4.3

In Figure 4.4 we have enlarged the nodes with the highest *eigenvector centrality* in the network. This time, the first and the second network resemble one another, in that the nodes with high centrality are clustered together, whereas in the third network they seem to be sparse. This is a result of the fact that the graph corresponding to the last network has less nodes and vertices than the other two.

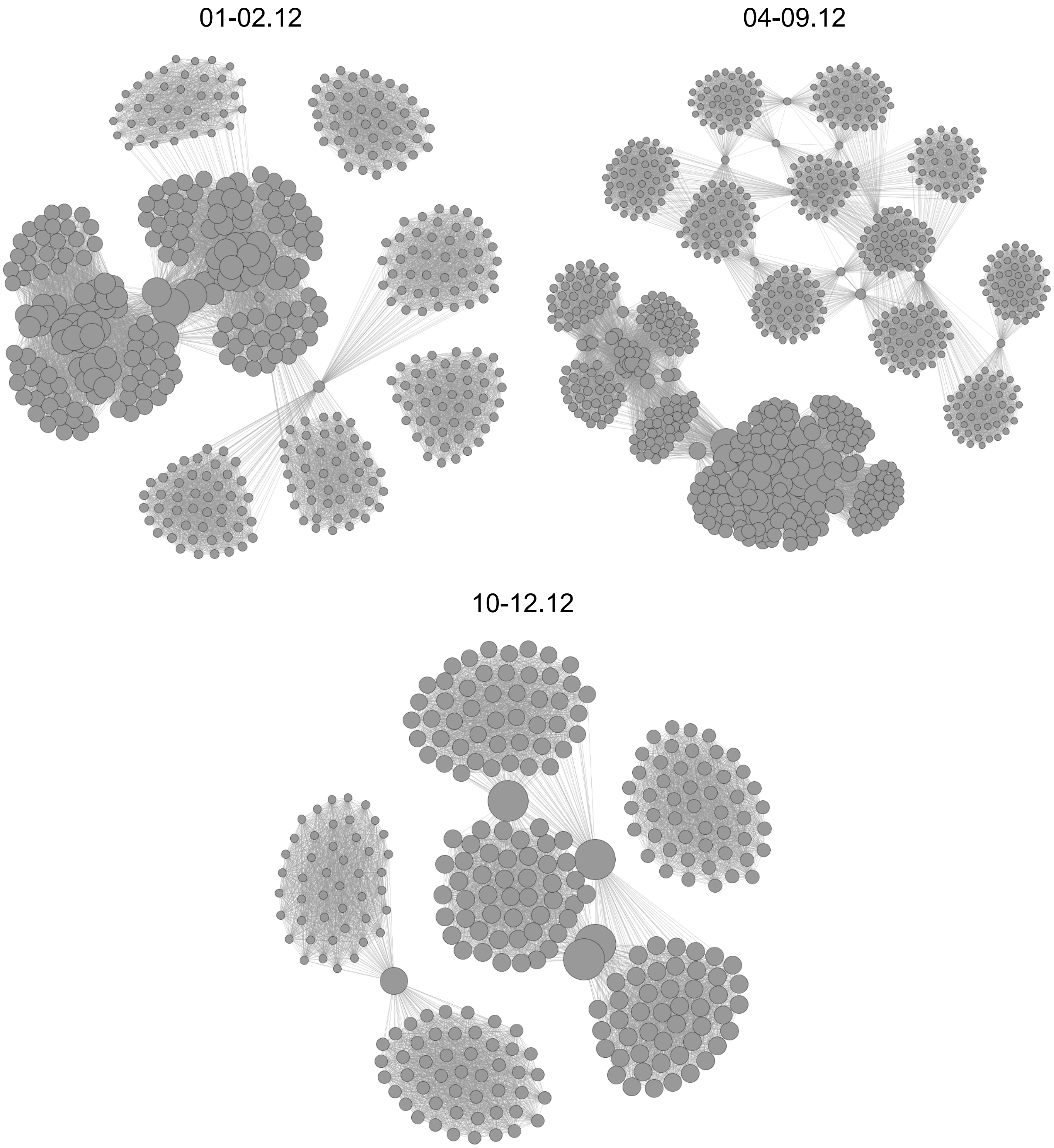


Figure 4.4

Communities in the network

Figure 4.5 shows the network colored based on different communities and the users with the highest betweenness centrality have been enlarged. These results have been yielded when determining communities with a 1.0 modularity.

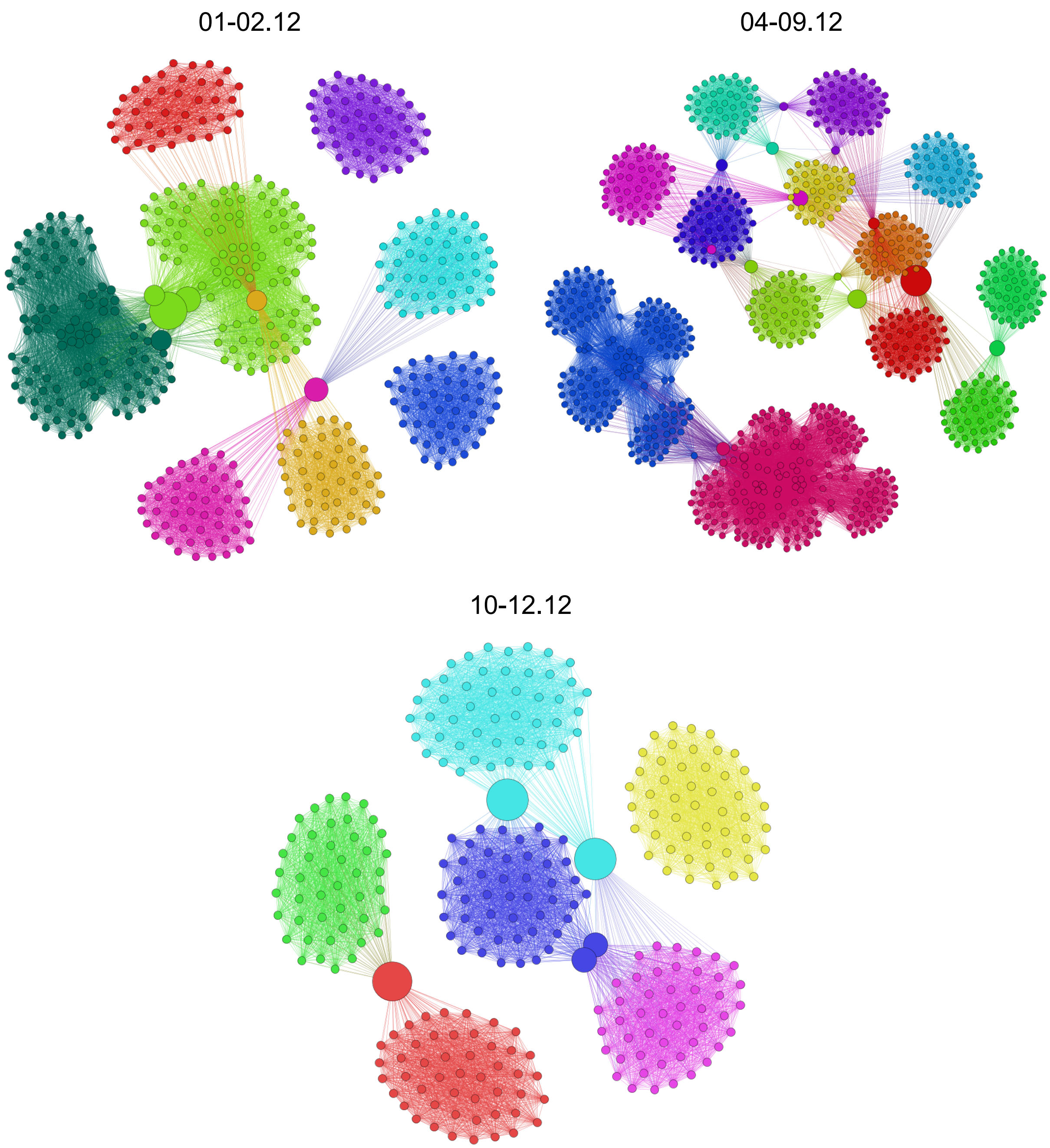


Figure 4.5

Just like the networks for the movie '*The Interview*', the communities here are very well defined and very well separated from one another. This time, though, we can spot some differences between the three networks: for the first and second graph, the communities have engulfed more than one post (which again proves how popular the page was during this timeframe); the third network, however is smaller and the communities are only defined around single posts, with a few brokers between them.

Connected components

The connected components are displayed in Figure 4.6. We can see that the first and the third network contain five, respectively three connected components, whereas the second one contains only two.

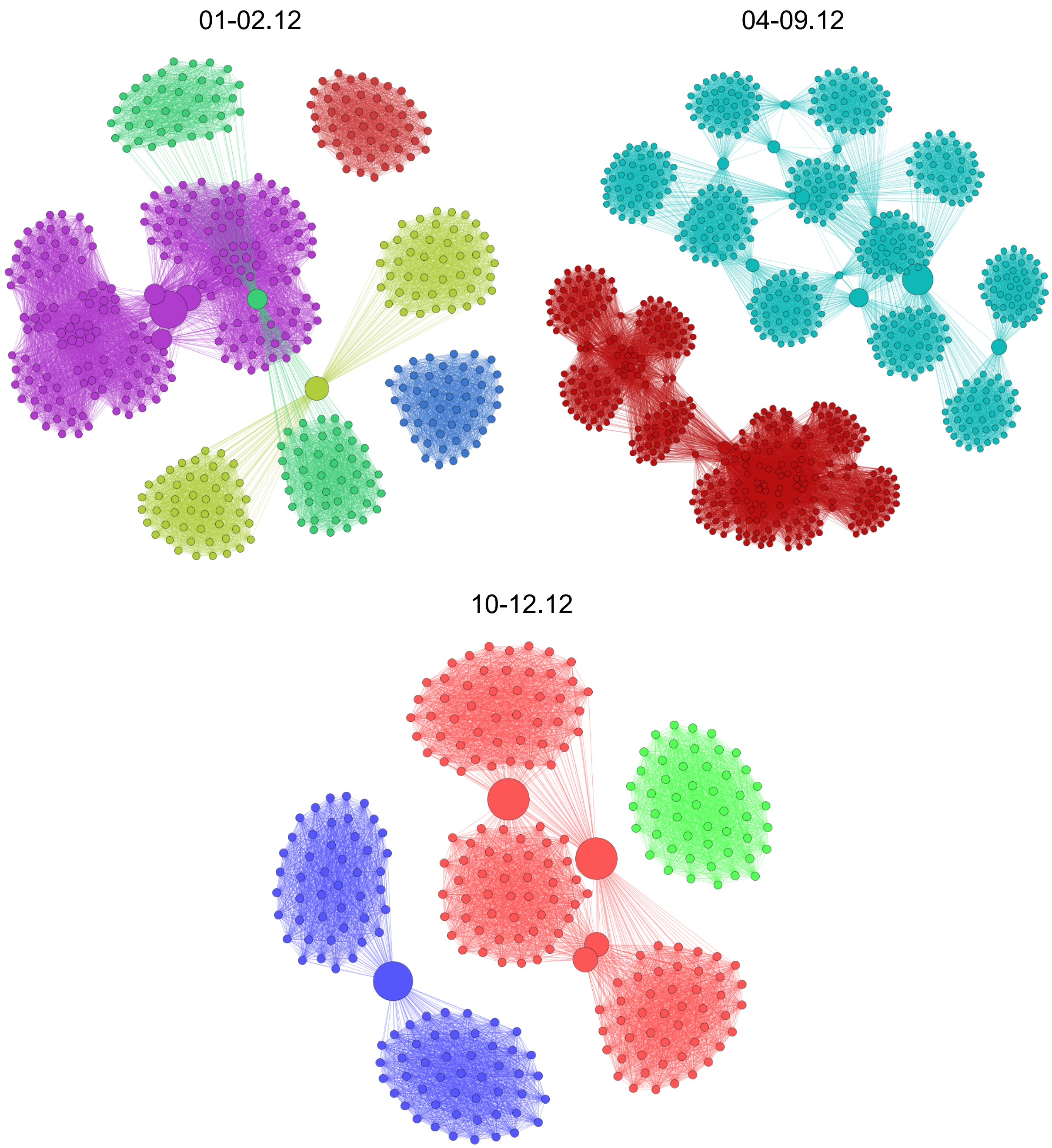


Figure 4.6

Conclusions

The analysis of the four networks was based on data collected from the Facebook and Twitter social web pages. However much we've strived, we could not import all the data we desired to, due to the limitations of the existing tools and APIs. Therefore, the graphs we have obtained are smaller than in reality. We tried as much as possible to improve the quality of the data, by querying the most popular tweets on Twitter and retrieving data from the release days on Facebook.

There are several differences between the networks we have analysed and we have highlighted these in the previous paragraph. The most important would be: while the Twitter networks are directed, the Facebook ones are undirected. In the Twitter network, we have liked users to other users whose posts they have retweeted, whereas in the Facebook network there is an edge between all the users that were active around a post. Moreover, we have analysed the Facebook network during certain periods of time, that have marked different release dates of the two movies. Another difference would be that '*The Interview*' Twitter network seems "younger" than '*The Hobbit*': the hubs and the authorities are not yet well defined and the graph is rather sparse - there aren't many users who have retweeted more than one post. Regarding the Facebook networks of the two movies, '*The Hobbit*' has more users liking and commenting on their posts, especially around the US release, while '*The Interview*' network is smaller and doesn't change much in time.