LINM2472 - Project 1 (Network)

Student : Thonnard Julien - Van Droogenbroeck Elise - Vermeulen Corentin

1 Co-occurrence network of characters

First we parsed the movie script to find the characters in each scene. For that we used the "INT." and "EXT." that cut out the scene. The character's name were always in uppercase so it was a good way to select them. We go some errors that we removed by hand as well as some duplicates with different spelling that we also managed. We finally have 42 characters. From the vector containing each scene we computed a co-occurrence matrix. We decided to not compute the co-occurrence matrix on the set of scene (every character only one time in each scene). We though that scene were the same character's name occurred multiple times was a good indicated of the strength of the interaction during the film. For example a long dialogue between Harry and Ron will enforce their connectivity while a small 'Hello harry' and 'Hi Ron' will only enforce the connectivity by one.

2 Analysing the communities of the graph

2.1 Network Visualisation

Our final graph is composed of 42 nodes and 179 edges. We clearly see that the main characters of the movie have the most edges. It seems that our network is a good representation of the interactions between the characters in the movie. We may suppose that one of the communities will be composed of all the main characters like Harry, Hermione, Ron, etc. The different teachers may also form their own community among themselves. Finally, two characters (Ernie and Hannah) only interact with each other so they will form their own community too. This will have an impact on the following results.

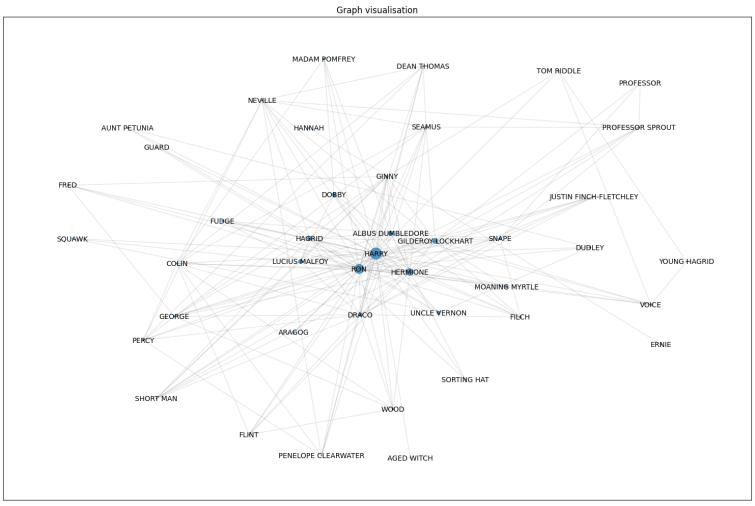

Figure 1: Movie network visualisation

2.2 Degree assortativity of the network

A negative degree assortativity coefficient implies relationships between nodes of different degree (graphically, nodes with many links are connected to nodes with few links) while a positive coefficient means that nodes tend to connect to similar degree nodes.
In the case of a movie, it is relevant to see a negative coefficient since the main characters are in general connected to all the other characters, independently of their degree.

2.3 Louvain algorithm

The communities are consistent with the story. The first and biggest cluster is composed of the main characters of the story (Harry, Ron, Hermione, Aragog, etc.). The second biggest cluster contains the secondary characters, such as Draco, Colin, etc. The other communities are composed of the less important characters of the story, such as the green community with Harry's in-laws.
As mentioned above, Hannah and Ernie form a separate community because they only interact with each other.

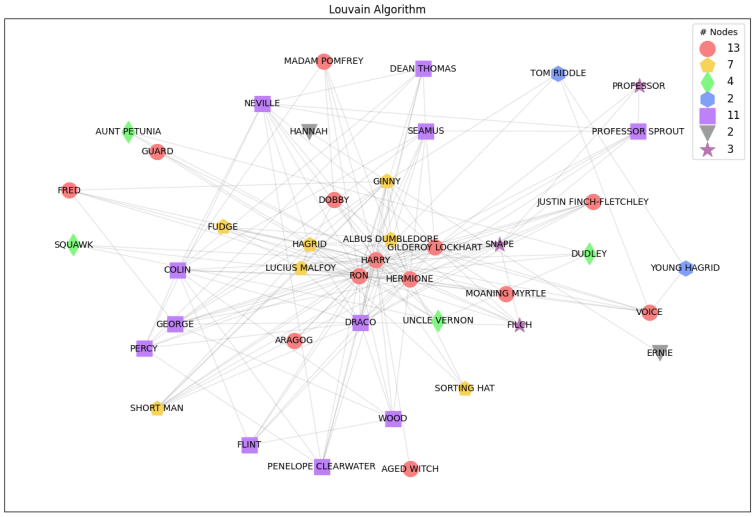

Figure 2: Communities visualisation with Louvain algorithm

2.4 Spectral clustering on the Laplacian

The `nx.spectracl_layout()` function doesn't give exploitable results since most of the points are gathered on the same spot and only one other point is on a different axis. This single point represents Hannah and Ernie since they are not linked to the rest of the graph. Better results could be obtained if Ernie and Hannah were removed from the characters. Therefore we decided to visualise the spectral clusters using a K-means on the spectral coordinates.
As expected, most of the points are purple and are gathered in the top right corner. Hannah and Ernie are located in the same cluster which is logical since they are only linked to each other. The other clusters are only composed of one individual with a low degree.

2.5 Comparaison between Louvain Algorithm and the spectral decomposition on the Laplacian

The spectral decomposition doesn't return good results because there are only a few communities. It is coherent since when we give a first glance at the network visualisation no community really stands out.
When we take the story into account the louvain algorithm is more consistent, as said before it really fits the character's hierarchy.

3 Maximising the influence in the graph

3.1 Definition of the ICM

The `ICM()` function returns the average number of reached nodes of graph G when starting from a seed set A0 and where the probability of reaching a non-reached node is p.

Figure 3: Spectral decomposition

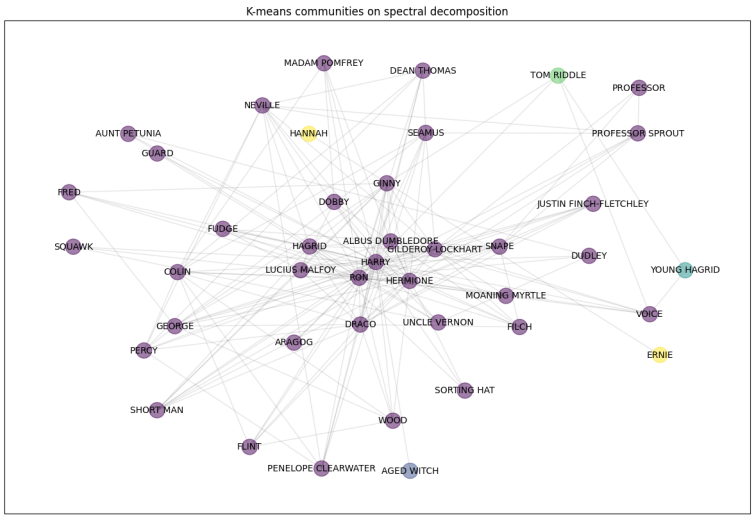

Figure 4: K-means communities on spectral decomposition

3.2 Greedy algorithm

The `greedy_algorithm()` function returns the set of nodes that will maximize the mean reached people with the `ICM()` function starting from that set of nodes as seed.
Afer computing the greedy algorithm on our graph we find that the best seed set is composed of Harry and Hannah. Harry is the lead character, he has the highest degree thus he is the most likely to reach everybody. Hannah on its side is completely disconnected from the main graph. She's only connected to Ernie. Starting with Harry first allows us to reach almost the full graph and adding Hannah (or Ernie) allows us to reach two more people.

3.3 ICM on MI set compared to ICM on largest degree set

Choosing the two nodes with the highest degree don't show us the best result compared to the choice made by the Greedy algorithm. Indeed, we say before that we have two characters disconnected from the others characters. Moreover, this two characters doesn't have a high degree so they won't be chosen as one of the two character with the highest degree. So, with the Greedy algorithm who choose one of this node, the model can influence more people.

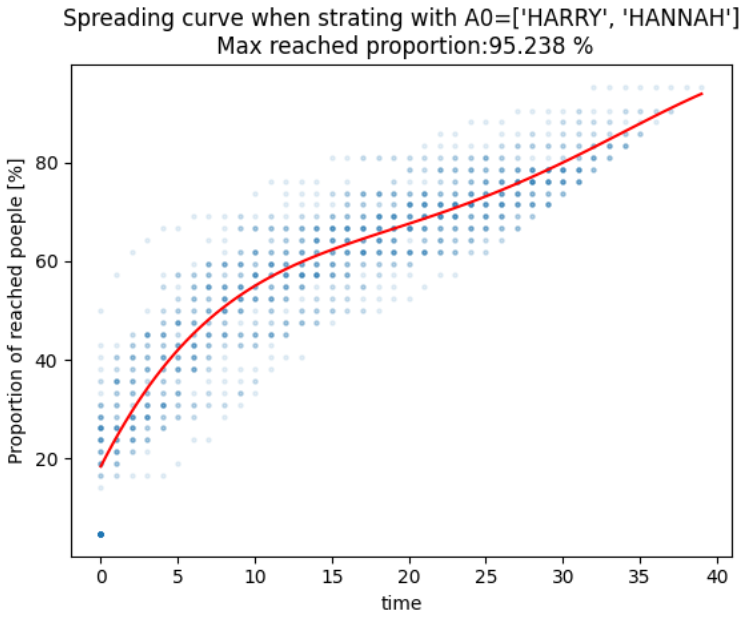

Figure 5: Average spreading curve on 100 runs of ICM starting with MI node

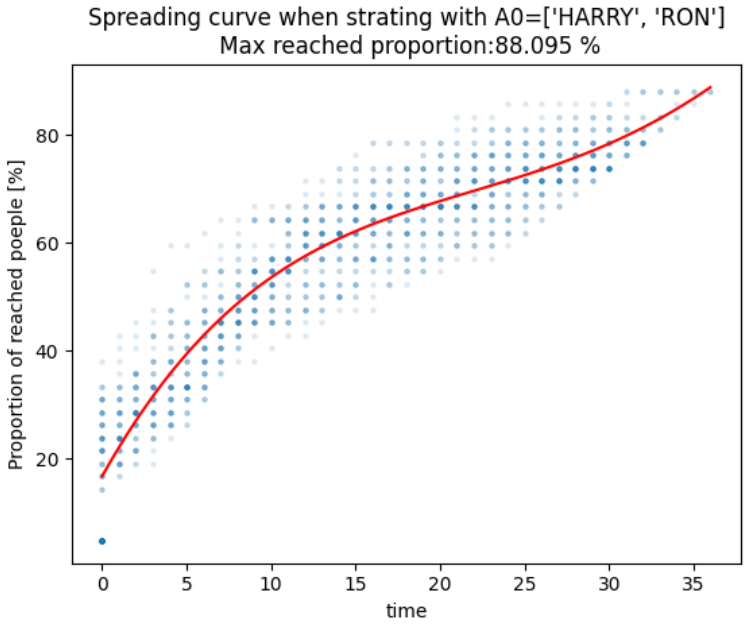

Figure 6: Average spreading curve on 100 runs of ICM starting with highest degree node

3.4 Barabasi-Albert network

We see that the degree assortivity of the Barabasi-Albert network (-0.250) is bigger than the degree assortivity of the network of our movie (-0.346). It is normal because the Barabasi-Albert network is more homogeneous. That means that the links are better distributed between the nodes in the Barabasi-Albert network.

With the previous idea, the percentage of influenced characters is bigger and the time needed too influenced those people is smaller on the Barabasi-Albert network.

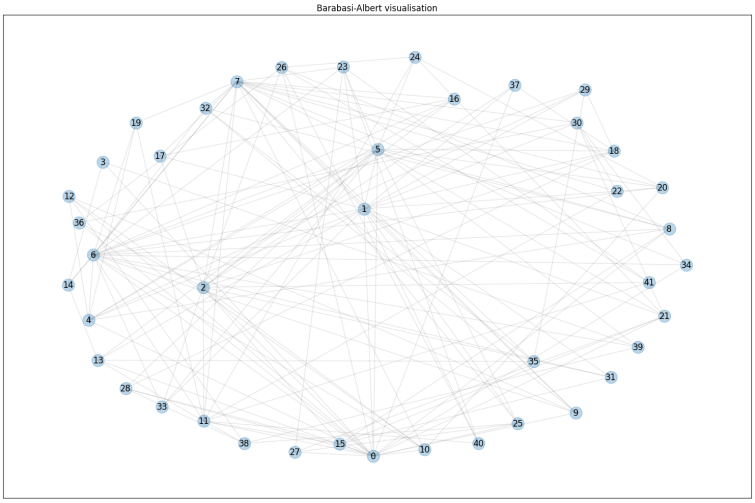

Figure 7: Visualisation of the Barabasi-Albert Network

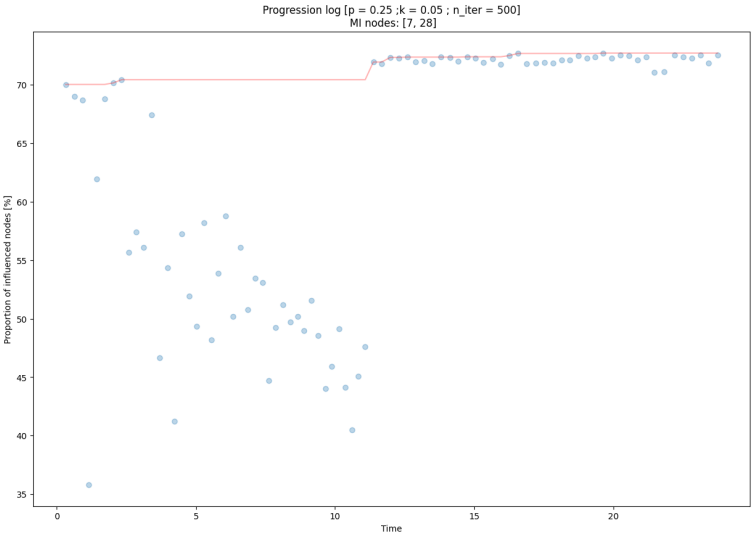

Figure 8: Greedy Algorithm progression over time on Barabasi-Albert network

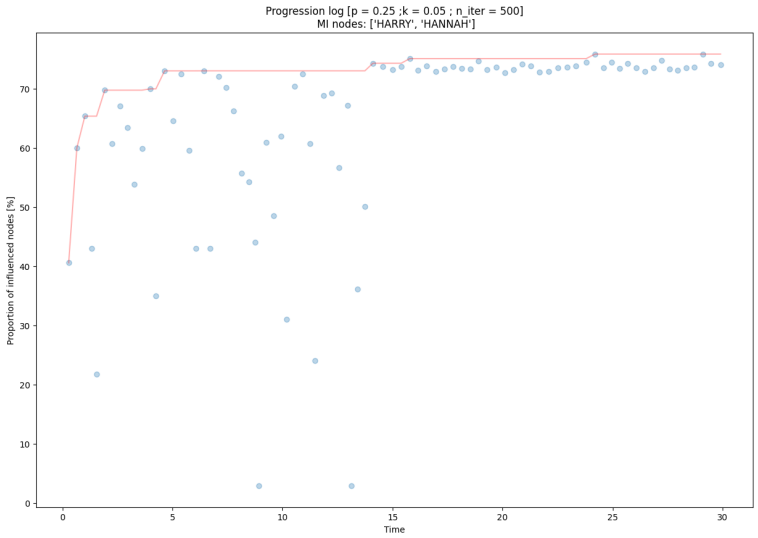

Figure 9: Greedy Algorithm progression over time on movie network