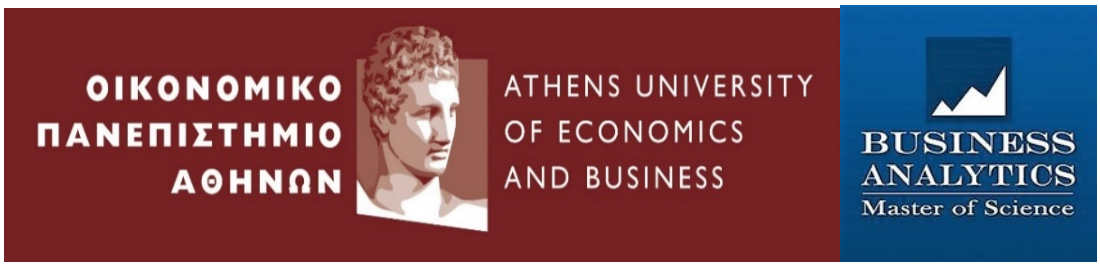


Department: Department of Management Science and Technology

M.Sc.: Business Analytics (Full-Time)

Course: Social Network Analysis

Professor: Katia Papakonstantinopoulou

Deliverable: First Project

Student Name: Evangelos Lakkas-Pyknis

Registration Number: f2822306

E-mail: eva.lakkaspyknis@aueb.gr

Περιεχόμενα

Network properties

Number of vertices and edges

The constructed graph has 796 vertices and 2823 undirected edges. This in simpler words means that in the network, 796 distinct characters of Game of Thrones are present and there are 2823 two-way relationships between them.

List of vertices and list of edges

The lists of vertices and edges contain the names of the characters in total and of the characters that are being related to each other. For space-saving reasons I will not be able to display these two lists to their full extent because they are too big, however, the full results can be extracted by executing the submitted code in the moodle. Below a sample of these two lists is displayed.

Vertice (Character) Name
Aegon-Frey-(son-of-Stevron)
Aegon-I-Targaryen
Aegon-Targaryen-(son-of-Rhaegar)
Aegon-V-Targaryen
Aemon-Targaryen-(Maester-Aemon)
Aenys-Frey
Aeron-Greyjoy
Aerys-II-Targaryen
Aegon-Frey-(son-of-Stevron)
Aegon-I-Targaryen
[…]
Rugen
Rolph-Spicer
Tristifer-IV-Mudd
Walda-Frey-(daughter-of-Merrett)
Theobald
Ulf-son-of-Umar
William-Foxglove
Werlag
Willow-Witch-eye
Wulfe

From	*To*	*Weight*
Addam-Marbrand	*Tyw.Lannister*	*6*
Addam-Marbrand	*Varys*	*4*
Aegon-Frey-(son-of-Stevron)	*Stevron-Frey*	*3*
Aegon-Frey-(son-of-Stevron)	*Walder-Frey*	*6*
Aegon-I-Targaryen	*Stannis-Baratheon*	*4*
Aegon-I-Targaryen	*Torrhen-Stark*	*3*
Aegon-I-Targaryen	*Tyr.Lannister*	*3*
Aegon-I-Targaryen	*Visenya-Targaryen*	*3*
Aegon-Targaryen-(son-of-Rhaegar)	*Rolly-Duckfield*	*11*
Aegon-Targaryen-(son-of-Rhaegar)	*Tyr.Lannister*	*23*
Aegon-Targaryen-(son-of-Rhaegar)	*Tyw.Lannister*	*3*
Aegon-Targaryen-(son-of-Rhaegar)	*Viserys-Targaryen*	*3*
Aegon-Targaryen-(son-of-Rhaegar)	*Ysilla*	*4*
Aemon-Targaryen-(Dragonknight)	*J.Lannister*	*3*
Aemon-Targaryen-(Maester-Aemon)	*Alleras*	*4*
Aemon-Targaryen-(Maester-Aemon)	*Al. Thorne*	*7*
Aemon-Targaryen-(Maester-Aemon)	*B. Marsh*	*4*

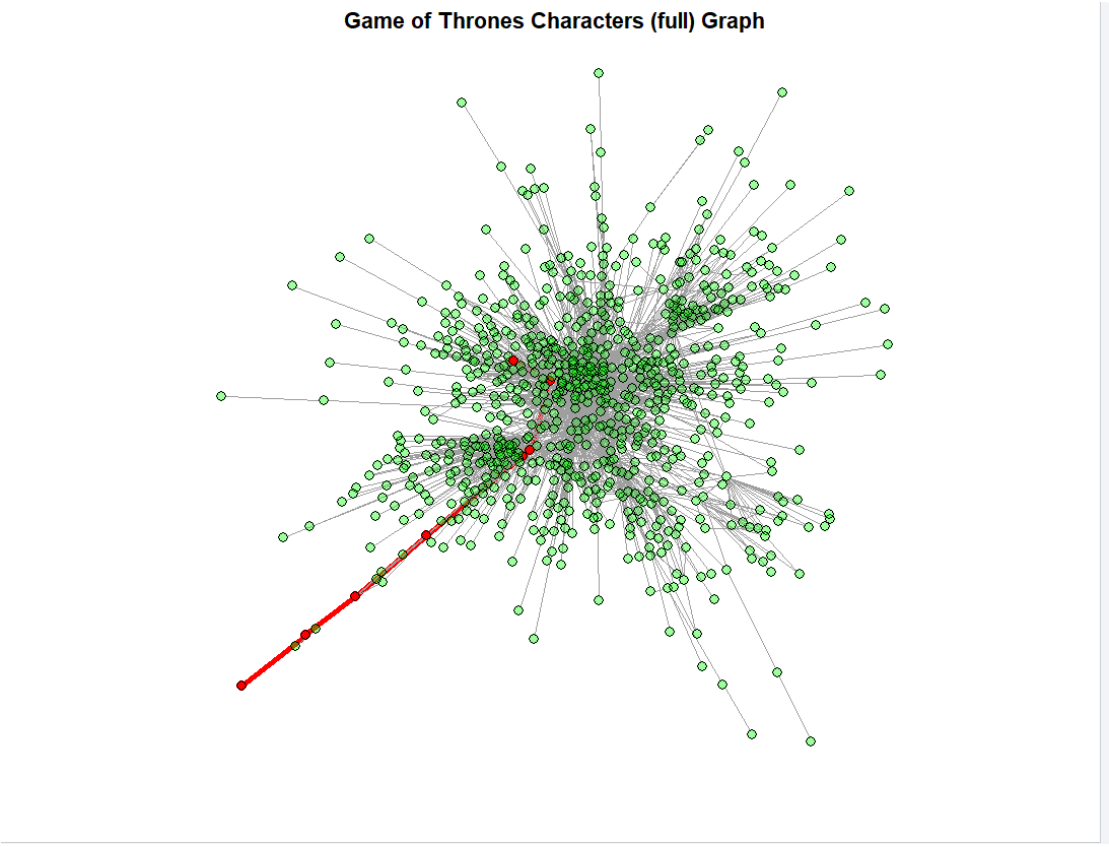
Aemon-Targaryen-(Maester-Aemon)	*Chett*	*9*
Aemon-Targaryen-(Maester-Aemon)	*Clydas*	*33*

Table 1: List of Vertices (left) and List of Edges (right)

As stated before, vertices in this particular graph are undirected and they also have a weight attached to them, so relationships are not equally important to each other.

Diameter of the graph

The diameter of the graph is equal to 53. This means that the sum of the weights of the edges that shape the distance between the two most distant nodes in the network is 53. The path that corresponds to the diameter (the longest shortest path) is shown below with a red line and the color of the vertices that belong in the diameter is also red.



Number of triangles in the graph and of edges having a weight more than 12

The total number of triangles in this graph is 5655 and the edges that weigh more than 12 are 610.

Top 10 characters of the network as far as their degree and weighted degree are concerned

The difference between the “degree” and “weighted degree” measures is that while the first count is the number of characters that interact with a specific vertex, the weighted degree takes into account both the number of edges and their weight so it can be expressed

as the sum of weights of the edges that are connected with the vertex that interest us. The characters with the highest degree are the following:

Character Name	Degree
Tyrion-Lannister	122
Jon-Snow	114
Jaime-Lannister	101
Cersei-Lannister	97
Stannis-Baratheon	89
Arya-Stark	84
Catelyn-Stark	75
Sansa-Stark	75
Eddard-Stark	74
Robb-Stark	74

The characters with the highest weighted degree are the following:

Character Name	Weighted Degree
Tyrion-Lannister	2873
Jon-Snow	2757
Jaime-Lannister	2232
Cersei-Lannister	1762
Stannis-Baratheon	1649
Arya-Stark	1608
Catelyn-Stark	1569
Sansa-Stark	1547
Eddard-Stark	1508
Robb-Stark	1488

Top 10 characters of the network as far as their normalized degree is concerned (extra task)

Another measure that may be useful is the normalized degree which divides the computed degree with the number of all the edges minus 1 and thus can give us information on how important a vertex is in a scale from 0 to 1 in relation to all the existing edges and vertices. In our case, the results when compared to the simple degree do not differentiate significantly.

Character Name	Normalized Degree
Tyrion-Lannister	0.1534591
Jon-Snow	0.1433962
Jaime-Lannister	0.127044
Cersei-Lannister	0.1220126
Stannis-Baratheon	0.1119497
Arya-Stark	0.1056604
Catelyn-Stark	0.09433962
Sansa-Stark	0.09433962
Eddard-Stark	0.09308176
Robb-Stark	0.09308176

The top 10 characters of the network based on their local clustering coefficient and value of the global clustering coefficient of the graph

The clustering coefficients explore the tendency that the network has to potentially develop groups of vertices that are connected and have similar characteristics aka clusters. The global coefficient expresses this tendency for the whole graph and the local coefficients calculate the fraction of the connections that the neighbors of the vertex have to each other divided by all the possible connections among the vertex neighbors'. The vertices with the highest local clustering coefficient are the following.

Character Name	Local Clustering Coefficient
Aegon-Frey-(son-of-Stevron)	1
Albett	1
Alerie-Hightower	1
Allar-Deem	1
Alys-Karstark	1
Alysane-Mormont	1
Amabel	1
Arron	1
Baelor-Blacktyde	1
Baelor-I-Targaryen :	1

The value of the global cluster coefficient in the graph is approximately 0.21.

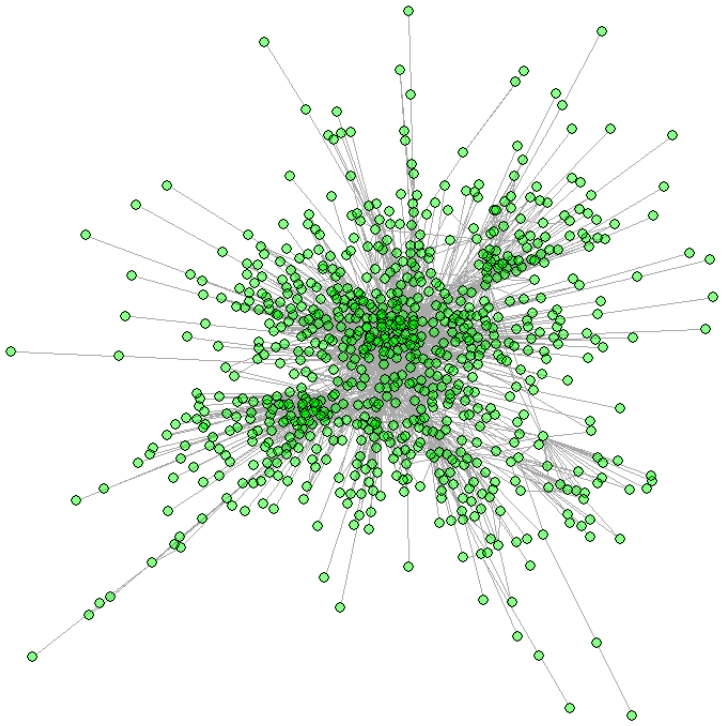
Comparing the networks' global clustering coefficient with random networks (extra task)

The global cluster coefficient of the network given above seems to be significant but in order to further explore the results of this measure and if they could come from a random occurrence I implemented a Z-score test. To do this I constructed 10000 random networks (according to the Erdös-Rényi model) that had the same number of vertices and edges as my real network. The mean global clustering that came up from these 10000 random graphs was approximately 0.009 which is significantly lower than the 0.21 value that is founded in the original graph. The Z-score was equal to 175.7733 and testifies for a higher clustering tendency in the Game of Thrones graph than expected by chance in random networks and that clusters and communities are much more likely to be formed in this graph than in random graphs(Z-score >2).

Plotting the Graph and the Subgraph

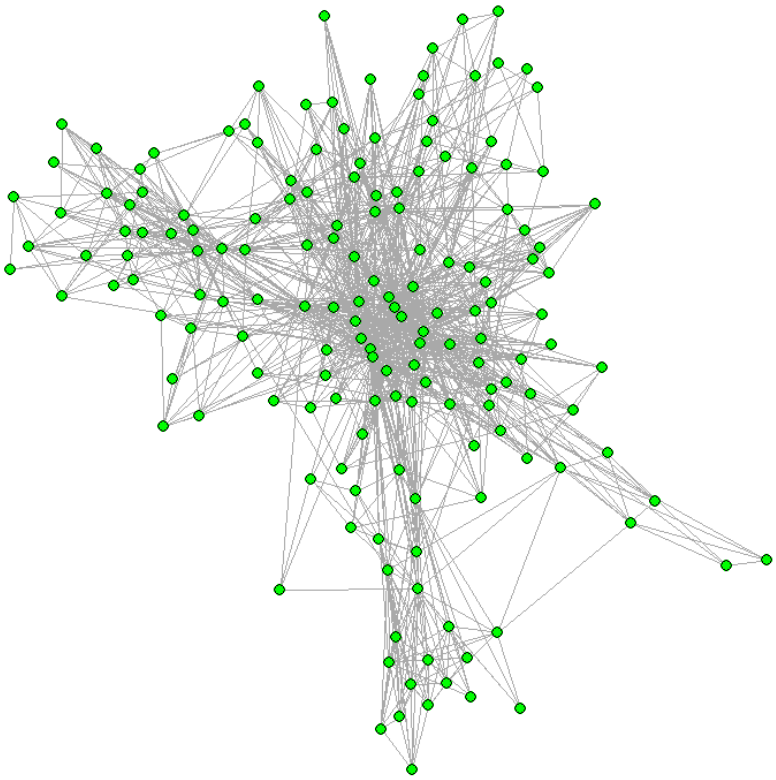
The full graph can be visualized as shown below. This is the full graph that consists of 796 vertices and 2823 undirected edges as previously referred (the edges do not point to a specific direction they are simply lines that connect the vertices in a mutual way).

Game of Thrones Characters (full) Graph



If we create a subgraph that focuses on the vertices with a higher degree (at least 9) the visualization takes the following form

Game of Thrones Characters with high degree Subgraph



Graph's and subgraph's edge density

Regarding the edge density of the entire graph and the subgraph it was calculated equal to 0.008921968 and 0.09494239 respectively. Generally, edge density is a measure of how many edges are in the graph compared to the maximum possible number of edges. And is derived from the following formula:

$$Edge\ Density = \frac{2 \times Number\ of\ Edges}{Number\ of\ Vertices\ \times (Number\ of\ Vertices - 1)}$$

More precisely for our case, the value of the entire graph's density (0.0089) testifies for a very sparse network where only 0.89% of all the possible connections among characters exist in the graph and that means that most characters do not interact with each other. The edge density in the subgraph is significantly higher (0.0949) meaning that approximately 9.5% of all the possible connections among characters exist in the subgraph. This result is reasonable and expected because by removing the vertices that had less than 9 connections I ended up with a less sparse network because the remaining vertices have higher degrees. From this higher density, we can conclude that these characters are more central to the original network and have many more interactions with each other.

Centrality

The ranking of the characters based on their closeness centrality value is given below

Character Name	Closeness
Jaime-Lannister	0.09587554
Robert-Baratheon	0.09244186
Stannis-Baratheon	0.09118018
Theon-Greyjoy	0.09111748
Jory-Cassel	0.09075342
Tywin-Lannister	0.09044369
Tyrion-Lannister	0.08984066
Cersei-Lannister	0.08981021
Brienne-of-Tarth	0.08939615
Jon-Snow	**0.08895603**
Joffrey-Baratheon	0.08785501
Rodrik-Cassel	0.08773866
Eddard-Stark	0.08682831
Doran-Martell	0.08654474
Robb-Stark	0.08653532

The ranking of the characters based on their betweenness centrality value is the following

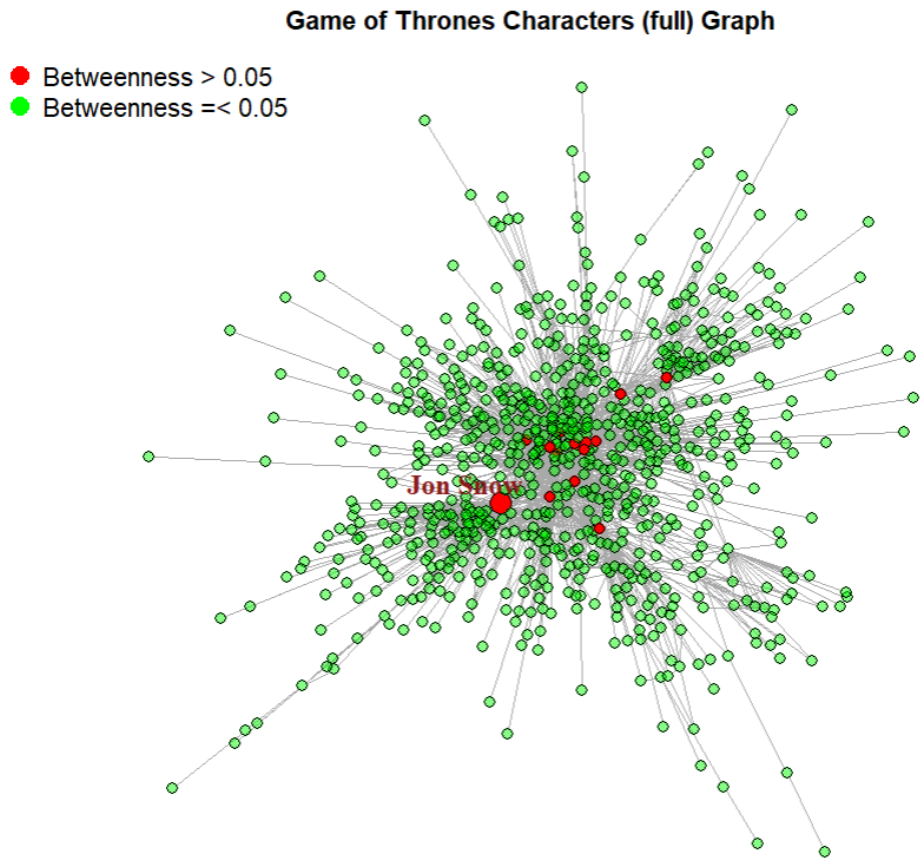
Character Name	Betweenness
Jon-Snow	**0.1321196**
Theon-Greyjoy	0.1232657
Jaime-Lannister	0.1167763
Daenerys-Targaryen	0.09419228

Stannis-Baratheon	0.0929144
Robert-Baratheon	0.09252286
Tyrion-Lannister	0.09162375
Cersei-Lannister	0.07734002
Tywin-Lannister	0.06358359
Robb-Stark	0.06295788
Arya-Stark	0.06132325
Barristan-Selmy	0.05630053
Eddard-Stark	0.0556227
Sansa-Stark	0.05042041
Brienne-of-Tarth	0.04947298

As we can see from the previous lists Jon Snow has a significant centrality value both in terms of closeness (10th position in the corresponding ranking) and betweenness (1st position in-betweenness ranking). The fact that he is found in the top 10 of the characters with the highest closeness indicates that he has a relatively small distance from all the other vertices in the network. However, the fact that significantly differentiates Jon Snow from all the other characters is that he has the highest betweenness value in the whole graph which means that Jon Snow is not only relatively close to many vertices but also "bridges" the interactions that take place in the network and if we supposedly took him out of the graph, we would likely lose a significant part of the network's information because Jon Snow functions as a bridge as I explained before.

Plotting the Graph based on betweenness and Jon Snow's position (extra task)

Also in the following visualization, we can see the vertices that have betweenness higher than 0.05 painted red and Jon Snow's position in the graph as well. From this graph, we can see that Jon Snow is not so central in terms of closeness as a group of red vertices that form a denser group but he is connected with a (slightly more sparse) part of the graph that is not so easily accessible from many other vertices and most red vertices. So, the importance of Jon Snow in terms of centrality and in the flow of information in the network that was concluded from betweenness and closeness measures before is also displayed in the following diagram.



Ranking and Visualization

PageRank is a measure that is widely used for measuring vertices' importance and is usually used for networks such as the World Wide Web. In our example, it can be interpreted as the probability that a random character reaches another Game of Thrones specific character based on a damping factor (c) for whom the PageRank value is calculated. In a narrative setting characters with high PageRank values are those who are referenced or connected to frequently by other significant characters and are indicated to be central in the story and have significant influence or importance in the evolution of events.

Character Name	PageRank Value
Jon-Snow	0.03570539
Tyrion-Lannister	0.03291094
Cersei-Lannister	0.02366461
Daenerys-Targaryen	0.0222804
Jaime-Lannister	0.01979001
Eddard-Stark	0.01896426
Arya-Stark	0.01857171
Stannis-Baratheon	0.01805099
Joffrey-Baratheon	0.01746037
Robb-Stark	0.01736071
Bran-Stark	0.01672032
Sansa-Stark	0.01632547
Robert-Baratheon	0.01569953

Catelyn-Stark	0.01523292
Theon-Greyjoy	0.01383743

In the following diagram, we can see the most important vertices as far as PageRank is concerned