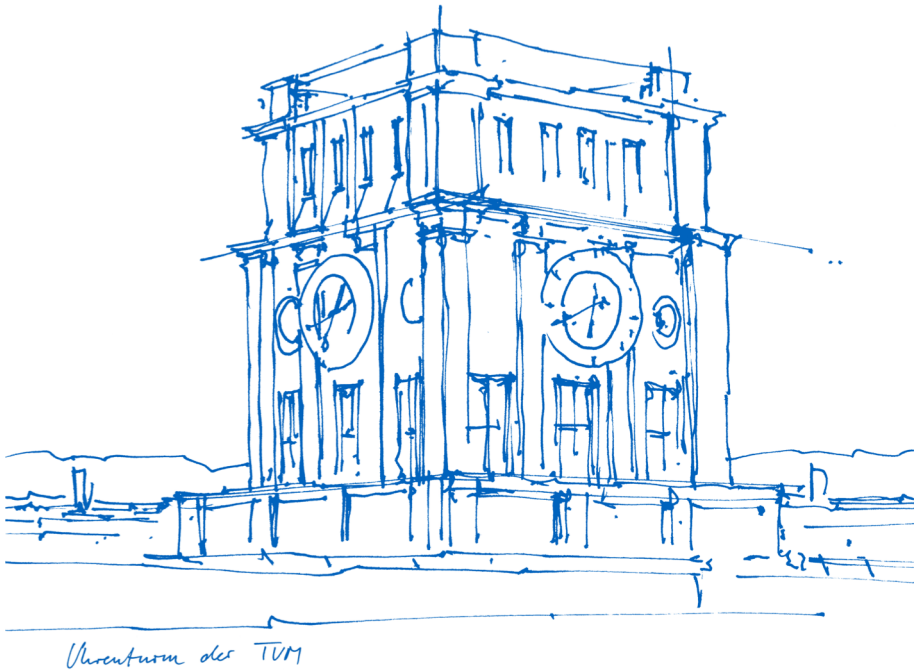
Exercises for
Social Gaming and
Social Computing
(IN2241 + IN0040) –
Exercise Sheet 2 -
Network Analysis



Exercise Sheet 2: Network Analysis

goal: analyse the Centralities of Game of Thrones characters

The Data: Game of Thrones

bookX.csv: all pairs of character co-occurances with their weights

- one for all 5 books

Task 2.1 - Creating a network

First, you will need to create a network from our dataset. Our dataset consists of five csv files - one for each book - representing the co-occurence network of the characters in the books. Two characters are considered to co-occur if they are mentioned within 15 words of one another in a book. (This kind of relationship may seem arbitrary, but co-occurence is a useful measure in Natural Language Processing.)

a) Import the first book and print some basic information about it. The column 'weight' corresponds to the number of co-occurences per pair.

b) Create and fill graph. In order to do things with our data, we have to have it in graph form.

1. Use the **networkx** library to create an empty graph for book1. Then populate it with the edges. Don't forget the edge weights!
   Hint: It is generally advised to not iterate naively over dataframes as it is much slower than other methods. But, our dataset is quite small, so you can use iterrows().
2. We are going to need a graph for each book so that we can look at the evolution of our social network. Create and fill the graphs as before, and add them to the list books.

Task 2.2 - Centrality

Who is the most important character? Or more generally in social networks: who is the most influental member? To answer these questions, we are going to complete an important step network analysis: measuring centrality. In the lecture, you have already learned about some centrality measures, such as degree centrality, betweenness centrality, and so on. In this task, you will use the existing networkx functions to compute centralities, then use the results to compare different approaches and reason about your network.

2.2.1 Betweeness centrality

The more shortest baths go through this node, the higher its betweenness centrality. (Note that our dataset only contains undirected edges.)

a) The top 10 in the first and last books

Let's calculate the betweeness centrality of the first and last books. Then, for both books, sort for only the characters with the top 10 centralities.

b) The distribution of betweeness centralities

Now, let's look at the distribution of betweeness centralities in the first book.

How does the distribution look like? Does the result remind you of a certain law on real life networks? Can you think of another network which looks similar to this distribution?

2.2.2 The evolution of character importance

According to betweeness centrality, the most important character in the first book is Eddard Stark but he is not even in the top 10 of the fifth book. The importance of characters changes over the course of five books because, you know, stuff happens... ;)

Let's look at the evolution of betweeness centrality of a couple of characters like Eddard Stark, Jon Snow, and Tyrion, which showed up in the top 10 of betweeness centrality in the first book.

a) The evolution of degree centrality

We can see that the importance of Eddard Stark dies off as the book series progresses. With Jon Snow, there is a drop in the fourth book but a sudden rise in the fifth book.

In the following, we will look at the evolution of the top characters using different centrality measures, and see if we can find anything interesting.

Plot the evolution of degree centrality of this network over the five books. Take the evolution of the **top four characters** of every book and plot it.

b) Eigenvector-centrality

Eigenvector centrality takes into account the importance of a nodes' neighbours. You may have many friends, but if none of them are famous, chances are you are not either. If you want a more detailed explanation of eigenvector centrality (and a handy site for other centralities), refer to this website [5].

Task

c) PageRank Centrality

PageRank was the initial way Google ranked web pages. It evaluates the inlinks and outlinks of webpages in the world wide web, which is, essentially, a directed network. Let's look at the importance of characters in the Game of Thrones network according to PageRank.

What does PageRank measure in our co-occurence network?

d) Calculating correlations

Let's look at the correlation between PageRank, eigenvector centrality, betweenness centrality and degree centrality for the fifth book using Pearson correlation.[3]

e) Reasoning

What can you say about the correlation of these measures? Does any pair stand out? Why might this be?

2.2.3. Conclusion

So we've been looking at different ways to find the important characters in the Game of Thrones co-occurrence network. According to degree centrality, Eddard Stark is the most important character initially in the books. But who is/are the most important character(s) in the fifth book according to these four measures?

b) Observation

Lastly choose a character (not Stannis Baratheon) and explain why their centralities might change between the different measures.

Submitting your solution

- work by expanding the .ipynb iPython notebook for the exercise that you downloaded from Moodle
- save your expanded .ipynb iPython notebook in your working directory
- submit your .ipynb iPython notebook via Moodle (nothing else)
- remember: working in groups is not permitted. Each student must submit their own .ipynb notebook!
- we check for plagiarism. Each detected case will be graded with 5.0 for the whole exercise
- deadline: check Moodle