UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

FIB

IRRS Lab 5 Report

Network Analysis

Emmanuel Werr
MSc student
emmanuel.werr@estudiantat.upc.edu

Gergö Greiner
MSc student
greiner.gergo@estudiantat.upc.edu

Facultat d'Informàtica de Barcelona
Master of Data Science

January 3, 2023

1 Introduction

The purpose of this lab is to perform network analysis on some synthetic and real networks using some existing network analysis libraries like networkx or igraph as well as any other existing software or program that might aid analysis (in our case, pyvis). We will have to analyze several networks; in the first section it is proposed to reproduce well-known facts about network models that we have seen in class. This first part is understood as a warming-up exercise. In the second part, we will have to build our own network and study it with the type of tools we have seen in theory class.

2 Analyzing Network Models

Erddös-Rényi model (ER model). The ER model takes two parameters: n, the number of vertices in the resulting network, and p, the probability of having an edge between any two pairs of nodes. A graph following this model is generated by connecting pairs of vertices with probability p, independently for each pair of vertices.

- Plot the average shortest-path length as a function of the network size of the ER model.

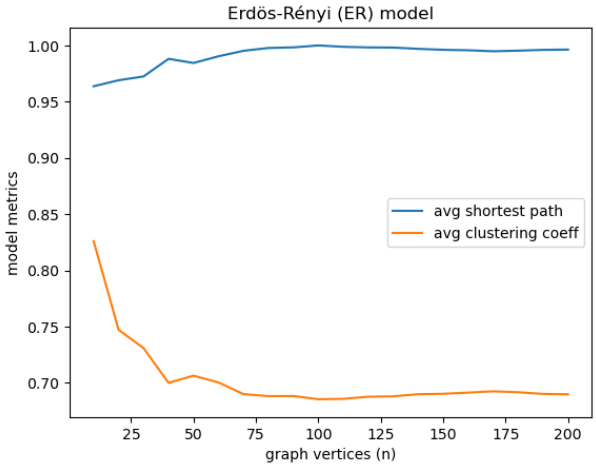

Figure 1: Average shortest-path length as a function of the network size of ER model

Watts-Strogatz model (WS model). The WS model takes two parameters as well: n, the number of vertices in the resulting network, and p, the probability of rewiring the edges in the initial network. A graph following this model is generated by initially laying all nodes out in a circle, and connecting each node to its four closest nodes. After that, we randomly reconnect each edge with probability p.

- Plot the clustering coefficient and the average shortest-path as a function of the parameter p of the WS model.

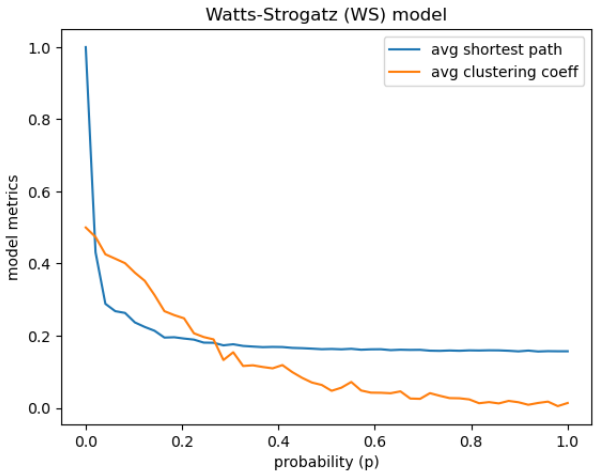

Figure 2: Clustering coefficient and average shortest-path as a function of the parameter p of WS model

Barabasi-Albert model (BA model). The BA model takes two parameters: n, the number of vertices in the resulting network, and m, the number of edges a "new" vertex brings to attach itself to existing nodes. A graph in this model is generated by adding new nodes according to the preferential attachment principled until the resulting graph has the desired size.

- Plot a histogram of the degree distribution of a BA network.

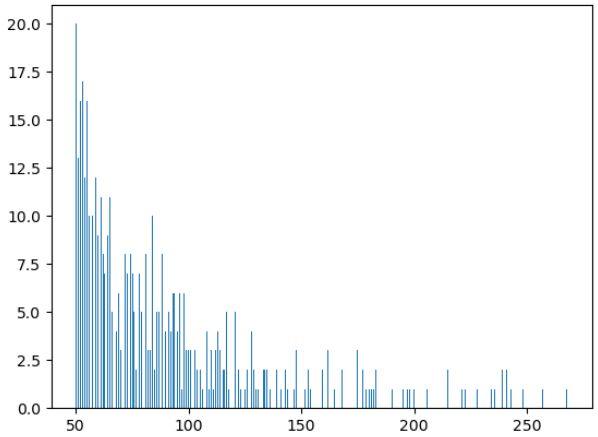

Figure 3: Histogram of the degree distribution of BA network

3 Building a Custom Network

For our custom network, we have decided to build a network using news text data. Initially, the goal was to scrape news articles from the web that mentioned the top publicly traded companies in the US Stock Market In order to build a network of people's/organization's relationships to one another within the business world (Several variation of this network were generated and one will be included in the end of this report as a little treat). However, in order to align better with the scope of this lab, our goal shifted to scraping news data from the web regarding the top 25 most influential countries and political organizations. Instead of Identifying entities within these news articles, the goal became to summarize the news articles in some way, and link them to one another based on a similarity measure, in order to generate a connected network.

The first part of this process was the scrape. For this we build a script that makes use of several functions to iterate over a list of search topics (countries and organizations), request the HTML for each respective search from google news, identify the links of the top 10 news articles for each search and scrape their paragraph text content from their source website. All of this information was structured in a simple pandas DataFrame and stored in local storage as CSV files (both raw individual and processed complete).

The next part was to summarize the article text in some way. This was done in order to ensure better relationships with similar articles. If similarity is measured between entire articles, they will all be very similar due to the reporting language and style. In order to do this we removed punctuation and stopwords from the text and selected only the top 10 most frequently mentioned words. We also identified Named entities from each article using Spacy and kept only those corresponding to 'PERSON', 'ORG', or 'GPE'. Finally, we cleaned the text from the title of the news article and combined the words from all three previously mentioned processes into a new field we called the "text_summary". This text summary was used as the singular representation for each news article.

The final part was to build the network. For this, we used the previously mentioned "text_summary" field in order to generate TF-IDF vectors for each individual article. We then used these vectors to calculate the cosine similarity between each. At the moment of generating the graph using networkx, we created a node for each article and then an edge between two nodes whose cosine similarity was above a certain threshold (we chose 0.04 because it was the lowest value for which we were able to generate a graph with no isolated nodes).

In order to review the code used for each of the steps mentioned above, go to the **/notebooks** directory within the main project directory. There are three .ipynb notebooks there. **'random_networks.ipynb'** contains the code for the first part of the lab on random models. **'financial_network.ipynb'** contains the code for the financial news network that was initially generated (still a work in progress but interesting nonetheless). And **'world_network.ipynb'** contains the code for the world news network that is the protagonist of this lab.

4 Analyzing the Custom Network

After the World News Network was generated, we began analyzing its properties. They will be included as screenshots and figures within this section of the report.

```
-----*----- Network Summary -----*-----

--- Nodes and Edges ---
Total nodes: 235
Total edges: 5030
There are no isolated nodes in your network!

--- Diameter and Transitivity ---
The Diameter of your network is: 4
The Transitivity of your network is: 0.469

--- Node Degrees ---
The max degree value for a node is: 112
The average degree value for all nodes is: 42.809
```

Figure 4: World News Network Summary Statistics

The screenshot above is from the notebook. It contains a summary of the main network properties such as total nodes and edges, diameter, transitivity, degrees of nodes, etc.

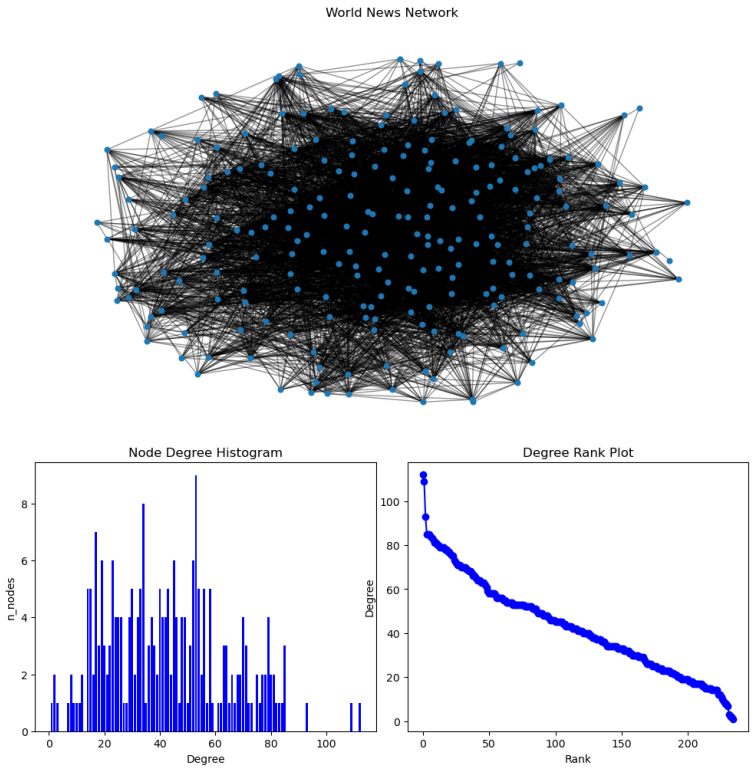

Figure 5: World News Network Summary Statistics

Above was a basic plot of the network using the spring layout as well as two plots of the node degree distributions of the network. Parameters for generating the graph were selected in order to make these plots closely resemble those of a random graph. Below is a histogram of the pagerank distribution for each node.

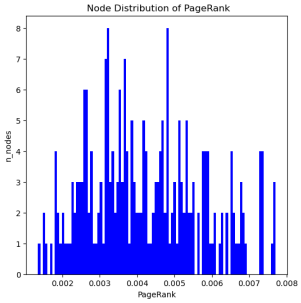

Figure 6: World News Network Summary Statistics

Below is the main attraction. A visualization of the World News Network using pyvis where node sizes are respective to their pagerank and we can observe communities identified by the Louvain community detection algorithm.

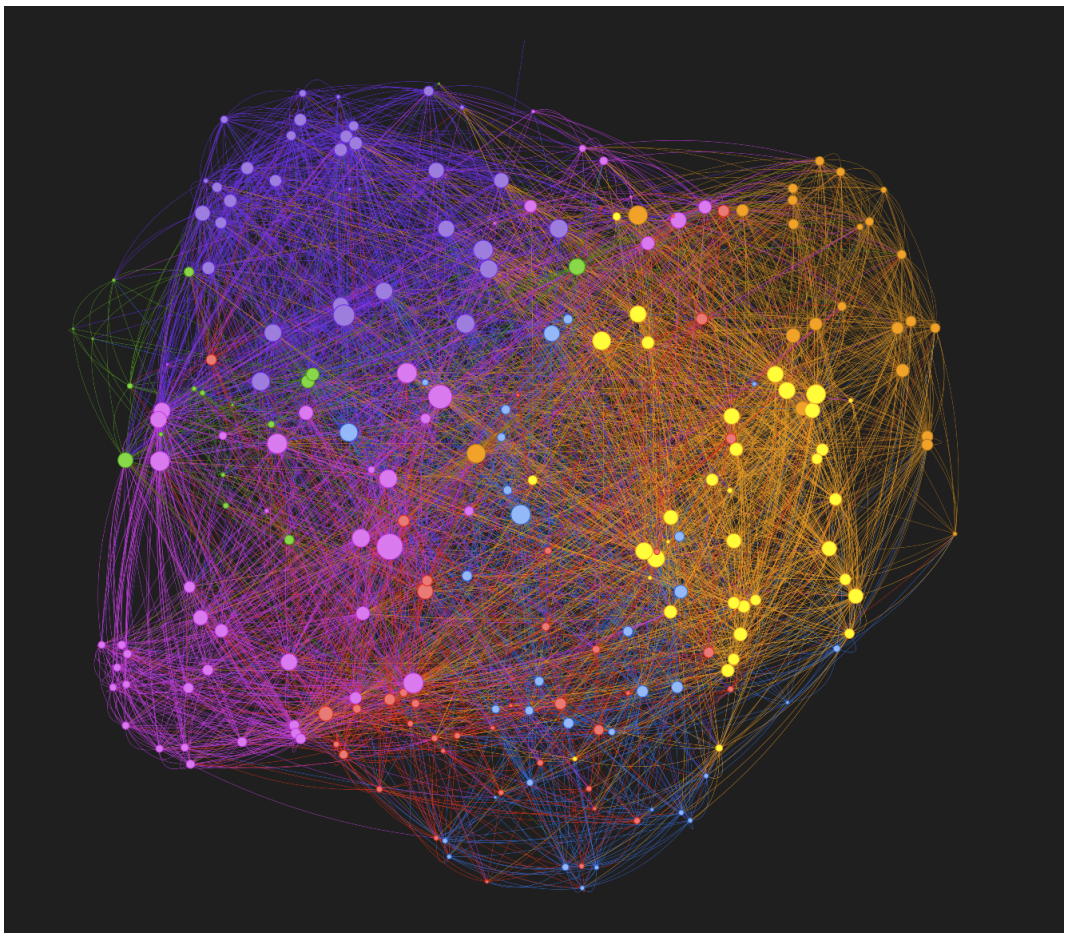

Figure 7: Louvain Communities on World News Network Visualization using Pyvis

5 Final Thoughts

This lab presented many difficulties, especially in cleaning the scraped text in Python in a way that generated the best results for the creation of the network. There is still much left to learn about network analysis but this lab proved to be a rewarding experience filled with many learning opportunities.