

CAREERFOUNDRY

Data Visualization with Python: Building a Historical Network Analysis

Objective

Imagine you're a freelance analyst contracted by the Institute for Public Policy to give an analytical overview and visualization of the interrelations between countries throughout the twentieth century. Given the volatility of world politics, the Institute is looking for historical ties between different countries to inform its research.

Many new job positions seek freelancers, so you'll need to choose which tools you'll use to structure your schedule and complete the assigned task. As a freelancer, some of the most common tasks you'll receive will require you to create visualizations to share your results. Academic institutions, startups, and other companies hire freelance data analysts to interpret and explain data that can help fit into the context of larger projects.

Context

For this project, imagine that the Institute for Public Policy has employed you to analyze and create a visualization as part of a larger initiative it's working on.

In this scenario, the institute believes that past events heavily influence the current geopolitical climate. To delve deeper into this hypothesis, a clearer understanding of the historical relationships between countries in the twentieth century is needed.

Suitable results can be achieved by looking at major global events throughout the twentieth century. To do so, you'll need to give a visualization and an analysis of the chain of events that led to the current state of affairs. To be effective, your visualization must highlight these past relationships. This way, other researchers can understand and tie your findings into the final report.

Data and Tool Requirements

For this project, you'll scrape data from a web page (using Wikipedia), which will be the basis for the visual analysis in the report. As you'll be looking for textual content, the data analysis will also involve text mining. Once you have the data ready for your analysis, you'll apply a basic **natural language processing** (NLP) algorithm to the collected data to extract a relationships map from it. As a final step, you'll create a **network analysis visualization** for all the content you scraped and preprocessed.

Data Sourcing Criteria

- Scrape a list of countries from a web page;
- Scrape a twentieth-century history and politics page from Wikipedia.

Analysis Criteria

- Use NLP to analyze the scraped content, compile a list of the countries involved in major twentieth-century events, and document how they're connected.
- Determine the degree, closeness, and betweenness centrality of the countries in your graph.

Visualization Criteria

- Create a dynamic network chart showing the interactions between the countries.
- The chart clearly distinguishes between different relationships and communities in a network.

Success Factors

- Present a polished, dynamic chart that provides an overview of the interrelations between world powers in the twentieth century. Your visualization should be clear so that your client can understand the relationships between different countries during historical milestones, as the final report will be available for viewing by the public.

Project Deliverables

Throughout this Achievement, you'll work from Exercise to Exercise to complete your project. For each Exercise task, you'll submit a deliverable that directly contributes to the final product—in this case, a network visualization.

Here is a breakdown of your course project deliverables by Exercise.

Exercise 1.1 Intro to Freelance and Python Tools

- Implement freelance skills by creating a timeline and setting up a workspace;
- Analyze tools for writing and executing Python code;
- Evaluate the advantages and disadvantages of different tools by summarizing their efficacy in various data analytics projects.

Exercise 1.2 Setting up the Python Workspace

- Apply knowledge of GitHub commands by cloning a repository and performing push and pull requests;
- Create an SSH key to access a repository from a local machine;
- Execute code to push new requests in a GitHub repository.

Exercise 1.3 Virtual Environments in Python

- Demonstrate an understanding of anaconda by creating and activating a virtual environment;
- Execute `apply()` and `lambda()` functions to create a flag column in Anaconda.

Exercise 1.4 Accessing Web Data with Data Scraping

- Explain the legal and ethical intricacies of data scraping by checking the terms of use;
- Organize an environment to perform a data scrape by implementing Python libraries into a virtual environment;
- Execute a data scrape on a website to collect web data.

Exercise 1.5 Text Mining

- Apply text mining techniques to a data set;
- Evaluate various text mining techniques to perform a particular analysis;
- Create bar plots to showcase text mining data.

Exercise 1.6 Intro to NLP and Network Analysis

- Evaluate data format by determining if a data set needs wrangling;
- Apply a natural language processing (NLP) algorithm to a text;
- Apply output from an NLP algorithm to create a dataframe for network analysis.

Exercise 1.7 Creating Network Visualizations

- Create a network graph using Python code;
- Analyze the quality of a network graph by explaining how it can be improved;
- Apply improvements to a network graph to create a final iteration.