Sergiu Buhatel, Research Assistant – Computing/ML

Graph - Internetworking

Table of Contents

[1 Introduction 1](#_Toc206253386)

[2 Data 1](#_Toc206253387)

[2.1 Raw Data 1](#_Toc206253388)

[2.2 Preprocess Data 2](#_Toc206253389)

[2.2.1 prepare\_nodes.bat 2](#_Toc206253390)

[2.2.2 prepare\_edges.bat 3](#_Toc206253391)

[3 Frontend 3](#_Toc206253392)

[4 GitHub 3](#_Toc206253393)

[5 References 3](#_Toc206253394)

# Introduction

# Data

## Raw Data

The raw data is stored as CSV files for each year from 2017 to 2023.

For example, in **2017** the files are:

* 2017\_industries.csv
* 2017\_Network.csv
* Top50\_2017.csv
* Top100\_2017.csv
* Top200\_2017.csv

Similarly, in **2018** the files are:

* 2018\_industries.csv
* 2018\_Network.csv
* Top50\_2018.csv
* Top100\_2018.csv
* Top200\_2018.csv

In general, the data follows the structure:

<year>/<year>\_industries.csv

<year>/<year>\_Network.csv

<year>/Top50\_<year>.csv

<year>/Top100\_<year>.csv

<year>/Top200\_<year>.csv

The only exception is **2023**, where Top50\_2023.csv and Top100\_2023.csv are missing, and the available file is named with a lowercase prefix: top200\_2023.csv.

## Preprocess Data (create\_all\_data.bat)

**Open a command prompt and run: (this is the only command needed to create the data)**  
create\_all\_data.bat

This will generate all data based on the raw files located in .\frontend\public\data\<year>.

It assumes that each directory under .\frontend\public\data is named after a year and contains the corresponding raw data.

create\_all\_data.bat iterates through all year directories and calls three scripts in this order (order is important): **prepare\_nodes.bat**, **prepare\_edges.bat**, and **generate\_data.bat**.

### prepare\_nodes.bat

First, it calls **generate\_tickers.py** to create a ticker\_<industry>.csv file in each year directory. The ticker\_<industry>.csv file is generated from <year>\_industry.csv and contains all tickers from the industry data, for a particular industry, formatted as a CSV with no header and a single column representing the company ticker.

Next, it calls prepare\_nodes.py to generate files named industry\_ticker\_<industry>.csv, along with Top50\_tickers.csv, Top100\_tickers.csv, and Top200\_tickers.csv. The exception is the year 2023, where only Top200\_tickers.csv is generated in addition to the industry\_ticker\_<industry>.csv files.

**Example file:**

ID,Label,Industry

AMC,AMC,Fun

ATNM,ATNM,Fun

BWL,BWL,Fun

BYD,BYD,Fun

**Example usage of prepare\_nodes.py:**

python prepare\_nodes.py \

--industries ./frontend/public/data/2017/2017\_industries.csv \

--top ./frontend/public/data/2017/Top100\_2017.csv \

--output ./frontend/public/data/2017/Top100\_tickers.csv

### prepare\_edges.bat

First, it calls **create\_tickers.py** to create a **ticker\_<industry>.csv** file in each year directory. The **tickers.csv** file is generated from **<year>\_industry.csv** and contains all tickers from the industry data, formatted as a CSV with no header and a single column representing the company ticker.

Then, it calls prepare\_edges.py for each year, which generates **edgesBtwTop50\_<year>.csv**, **edgesBtwTop100\_<year>.csv**, and **edgesBtwTop200\_<year>.csv** based on **Top50\_<year>.csv**, **Top100\_<year>.csv**, **Top200\_<year>.csv**, and **tickers.csv**.

### generate\_data.bat

# Frontend

# GitHub

The code is available on GitHub at <https://github.com/sergiubuhatel/ra2/tree/main/graph>

# References