Stocks and Google Search Trends

Dayton Roberts and Nicholas Moore

Data We Used

- Stock Ticker Data: The daily performance of 9000+ stocks anywhere from the 1960s to the modern day
- Search History Data: A scale from 0 to 100 of the monthly searches of 4500+ stocks

More Data We Used

- Company Info Data: Stock Symbol, Company Name, Industry, etc. for 6800+ US stocks

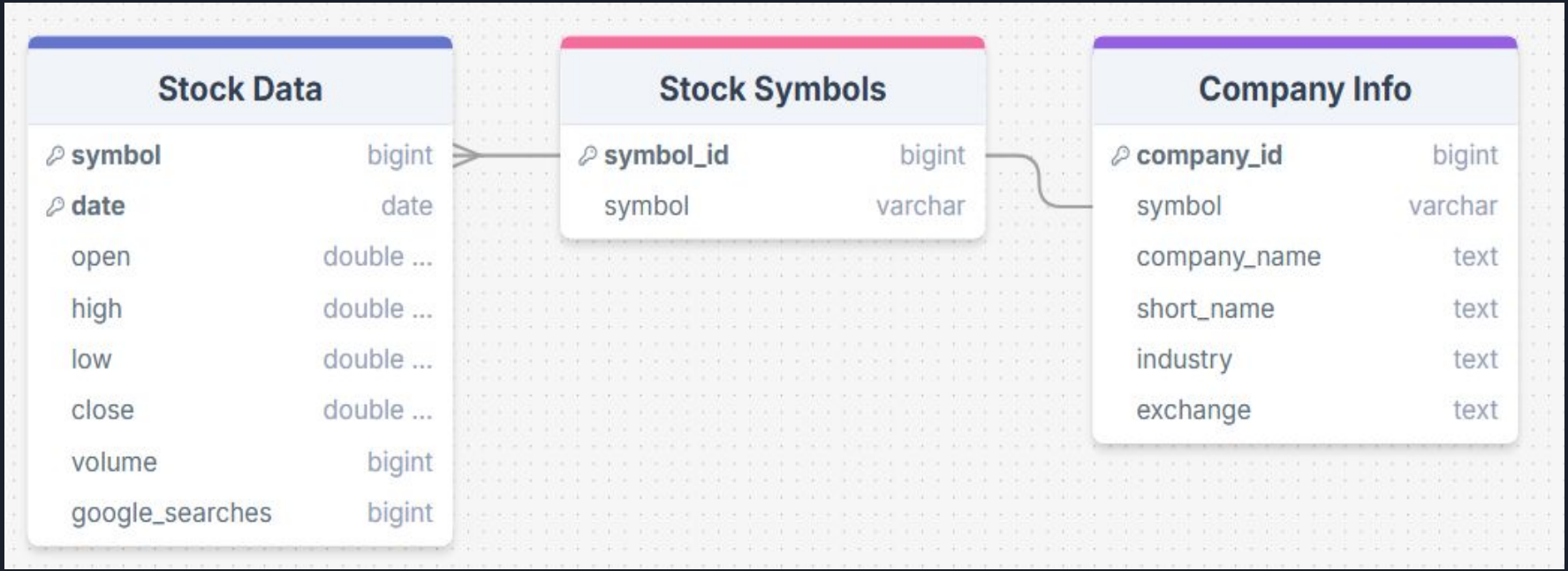
Data Wrangling

- The Stock Ticker dataset had lots of information, at over 34 million rows, but simple columns
- The Search History Data not as large but had 4500 columns, one for each stock. This exceeds the column limit of 1600 on our version of Beekeeper

Data Wrangling

- We used python dataframes to parse the search history table to get the google search rating for that month to populate our database
- The dates were in different formats across the datasets which caused comparison problems in python

Database


Stock Data
symbol bigint
date date
open double ...
high double ...
low double ...
close double ...
volume bigint
google_searches bigint
Stock Symbols
symbol_id bigint
symbol varchar
Company Info
company_id bigint
symbol varchar
company_name text
short_name text
industry text
exchange text

Data Preservation

- Stock symbol, date, company name and industry cannot be null
- Search ratings must be between 0 and 100
- Stock price must not be NULL
- All values for the stock must not be NULL

Homework Questions

A. What stock prices decreased when the previous month had a search count rating of less than 25?
B. What 5 stocks had the lowest average closing price when their google search count was less than 50? Submit your answer as a CSV file.
C. What industry in which year had the largest increase in total value of all stocks based on this dataset?

Interesting Things Found by Solutions

How google searches affect/signify the increase and/or decrease in the value of a stock across different industries

Learning Objectives of the Questions

- Use GROUP BY to compare values across categorical variables.
- Use HAVING and can implement it accordingly to select a subset of groups.
- Output the results of a query to a CSV file.
- Compute many different aggregate calculations over an entire column.
- Write a SELECT query that utilizes basic WHERE filtering and ORDER BY.
- When a self-JOIN is necessary and can compute one in those instances.

Questions?