Hedge Fund Holdings Tracker

Trihao Van

Task:

Find hedge funds which have NVDA holdings

User Profile

- Role: Hedge fund client manager

- Context: Client facing role with a primarily commercial focus. Not necessarily technical

- Objective: Quickly identify which hedge funds hold a specific asset (e.g. NVDA), potentially recommend assets for the client to buy

- Requirements:

  - Fast, intuitive access to information

  - Clear, well structured presentation

  - Easily digestible insights with minimal complexity

Data Sourcing

- Need a source of hedge funds and their holdings

- This can be difficult as they are private entities which do not publish their live data

- Ideally need an API so data can be programmatically updated

- Queried AI apps, asked "Whats the best public/freely available data source I could use?"

- Also asked friends who work in the finance industry!

- Found:

  - Paid services (E.g. WhaleWisdom)

  - Static pages (E.g. HedgeFollow)

  - All sourced their data from SEC EDGAR database using 13F filings

SEC EDGAR Database
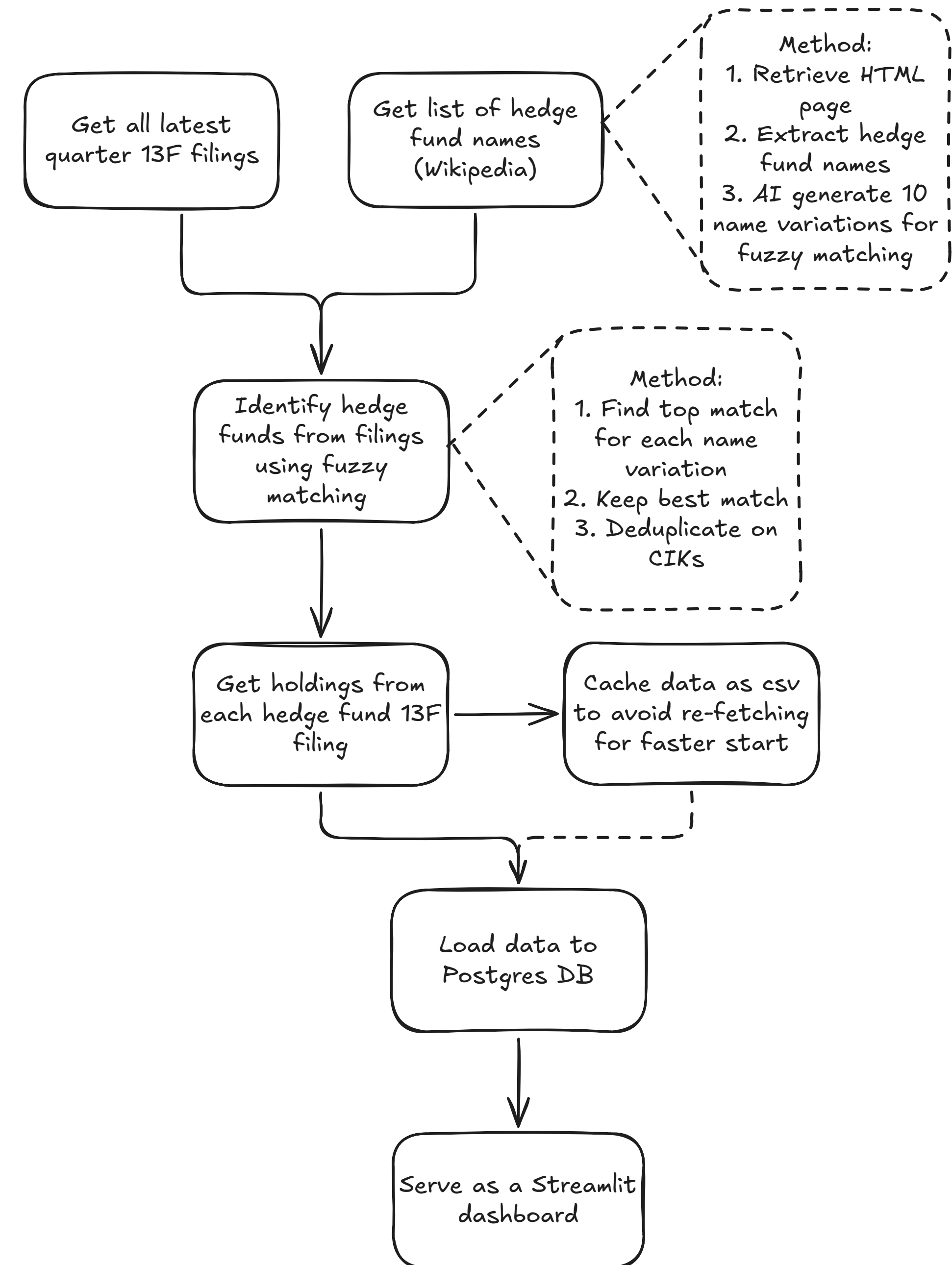
- 13F forms are filed by hedge funds with over $100M in US holdings

- Has an API

- Initially tried to use the API and manually parse the 13F data using AI and VSCode/Copilot

- Found an open source python package specifically aimed at processing EDGAR data (edgartools)

- Caveat: The 13F filings are only submitted on a quarterly basis

- Caveat: The holdings section in the 13F filings contain multiple security types including shares and options. The total value will be a combination of these

- Caveat: 13F filings are due 45 days after quarter end, so data can be up to ~4.5 months lagged

Design Considerations

- The user may want to look at other holdings than just NVDA

  - The solution must support searching for different assets

- The data must be as up to date as possible

  - Data must be easily refreshable to retrieve new information

- A use case for this might be to recommend purchases to hedge fund clients

  - A suggestion might be top holdings of other hedge funds who also hold NVDA
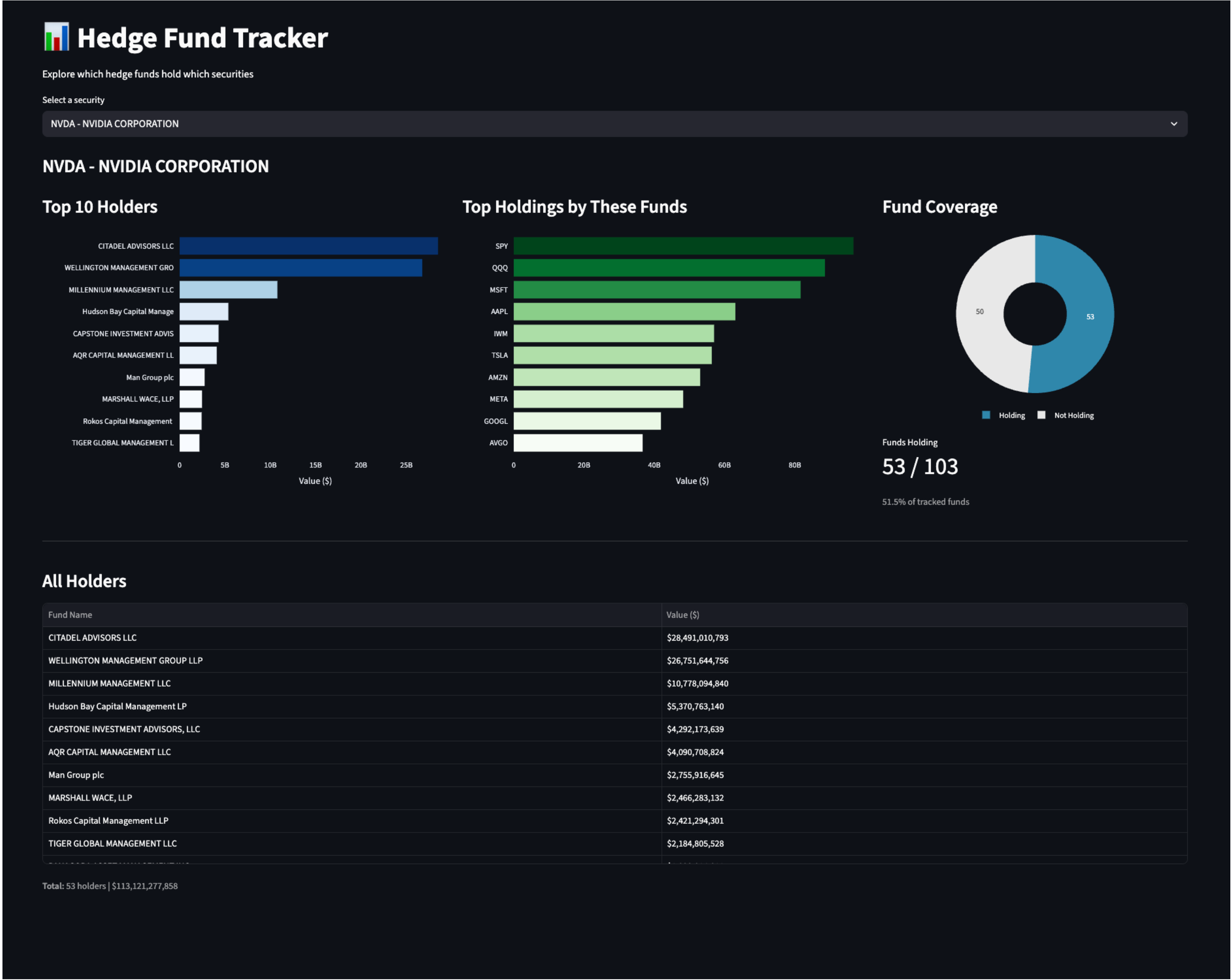
Architecture

- Assumptions:

  - The user would be managing/ monitoring a specific set of hedge funds - For the purposes of this exercise the hedge fund on the "List of Hedge Funds" Wikipedia page was used

  - The relevant hedge funds have significant US holdings and would therefore be required to submit 13F filings.

Get all latest quarter 13F filings

Get list of hedge fund names (Wikipedia)

Method:
1. Retrieve HTML page
2. Extract hedge fund names
3. AI generate 10 name variations for fuzzy matching

Identify hedge funds from filings using fuzzy matching

Method:
1. Find top match for each name variation
2. Keep best match
3. Deduplicate on CIKs

Get holdings from each hedge fund 13F filing

Cache data as csv to avoid re-fetching for faster start

Load data to Postgres DB

Serve as a Streamlit dashboard

Solution

A filterable, updatable dashboard showing holders of a given asset, top holdings of funds who hold the asset, and proportion of monitored funds who hold the asset

Considerations for production / further work

- Use a paid service to simplify data acquisition

- Automatically run the data download on a schedule to keep the information up to date (could be deployed as a lambda function)

- Allow the user to update the list of tracked hedge funds - replace the Wikipedia source with a configurable watchlist

- Use more data sources to capture data from funds with <$100m assets or non-US funds (some paid services include this)

- Build a graph database to better map the relationship between funds and assets, and also to track inherited holdings for fund-of-funds

- A collaborative filtering/recommendation ML model to recommend purchases automatically

- Add historical tracking to show position changes over time

AI Usage

- AI apps (Claude and GitHub Copilot) were used for:

  - Generating print statements for status updates

  - Researching available data sources

  - Generating a scraping function for the Wikipedia HTML page (Pasted the whole HTML into the prompt)

  - Initial attempt at manually parsing EDGAR data

  - Setting up the Postgres DB and writing some of the queries

  - Generating a skeleton for the Streamlit app (Created the image on the right and included it in the prompt)

  - Generating the makefile to streamline starting the app

  - Generating the README file

- Claude chat histories:

  - https://claude.ai/share/52284fae-d93d-43dc-bf8b-69daf5f53955

  - https://claude.ai/share/02bc5ca7-56a2-4027-88fe-63d8eb6ed966