IRisk Lab Data Discovery and Consolidation

Week 14 Report
Nov. 28
Rohit Valmeekam

- ❖ **What was Accomplished**
  - ➢ A Python-based web scraper was developed to extract real estate information from LoopNet. The scraper utilized the BeautifulSoup library for parsing HTML and extracting relevant data.
    - ■ The scraper extracted property details from LoopNet listings, including property address, market information, and other relevant attributes. Different parsing techniques were applied based on the structure of the LoopNet page.
      - ● One parsing technique for properties and the other for listings, etc.
  - ➢ To enhance the search capability, the scraper integrated Google search functionality. This allowed the system to find LoopNet property listings based on user-provided addresses. This also enhances the search algorithm for the frontend since the API call will be using Google's search algorithm
    - ■ This means that the user doesn't have to type in the correct spelling of the address or type in the full address
  - ➢ The extracted data was formatted into a structured dictionary, facilitating easy retrieval and display. The data formatting process included handling different sections of LoopNet pages, such as listing details and property information.
  - ➢ All of this functionality is hosted on an API in Flask, so that anyone can make calls to the API from a frontend for example and be able to run it with many different addresses
    - ■ Two endpoints:
      - ● /get-scraped-data which takes in a LoopNet URL and outputs the scraped information from it
      - ● /search-and-scrape which takes in an address, searches for the LoopNet URL and outputs the scraped information from it
    - ■ The JSON functionality was enhanced to customize the filename based on the property name. Special characters were removed to ensure file system compatibility.
  - ➢ A React-based frontend was developed to provide a user interface for interacting with the web scraping functionality. The application allowed users to input addresses, trigger the scraping process, and visualize the extracted data.
    - ■ The frontend design was optimized for a better user experience, ensuring responsiveness across various devices.

- A feature was implemented to export the scraped data in JSON format. Users could download a JSON file containing the property details.

This week I successfully implemented a web scraping solution for real estate information retrieval, coupled with a user-friendly frontend. The system efficiently processes LoopNet data, providing users with a seamless experience for obtaining and exporting property details.

- ❖ Next Steps
  - ➢ Fix bugs in google searching
  - ➢ Add additional endpoints to API
  - ➢ Do mass address testing