# AI-Based Form 4 Filing Interpretations for Improved Stock Market Information Availability to Public

Pratyush Saxena, Winston Wang, Sumedh Anuganti, Prudhvi Kodali

**Abstract**

*This project provides raw information from SEC Form 4 filings, visual interpretations of this data, and uses an artificial intelligence model to interpret the filings for traders of all levels to better understand the motivation of certain insider transactions. Form 4 filings can offer unique insights into the potential changes in the stock of a company, yet they are often not utilized to their full potential because they are complex to understand and many novice traders aren't even aware of their existence. Our goal is to create a Flask-based website that uses web scraping techniques to gather data from SEC's EDGAR API and Yahoo Finance. Then, this data will be analyzed by a model to generate the most likely explanation as to why a certain transaction was made, allowing the public to make more informed decisions in their future trades.*

## Introduction

Form 4 filings are an often overlooked tool to understand the potential success of a company. Many novice traders are not even aware of their availability, and all traders may struggle with understanding why certain transactions were made and what they mean for the future of the company.

This project introduces a web-based tool that aims to assist novice traders by creating an interface to easily provide information from Form 4 filings and visualize the impact on the stocks. The Flask-based website also has an artificial intelligence model created in Python that will interpret the web-scraped data and provide an explanation to the user as to why a certain transaction was likely made and what that means for the company.

First, we will provide a background for all the technologies used in our project, as well as the technical concepts we address. Then, we will explain how we created our Flask-based website and how this differed from our original plan. Finally, we will discuss where we hope to take this project in the future and what more can be done to expand on our current progress.

## Background

Form 4 filings, required by the U.S. Securities and Exchange Commission (SEC), disclose changes in ownership by company insiders—such as executives, directors, and major shareholders (owning more than 10%). These filings are submitted through the SEC's EDGAR system and serve as a public record of insider equity movements. Since insider transactions can reflect internal confidence or caution, Form 4 data is often used by investors to anticipate market trends (Team, 2024b). However, the raw filings are presented in complex, unstructured formats, making it difficult for beginners to interpret their significance. This creates a barrier for new investors who may miss valuable insights from insider behavior simply because the information is not easily accessible or understandable.

Several tools exist to analyze or track insider trades, including financial platforms like Yahoo Finance, Bloomberg, and OpenInsider. While these platforms display insider activity, they often lack context or simplified explanations. More advanced tools such as EDGAR Online and BamSEC cater to institutional users and assume prior financial literacy. Most platforms offer limited customization or filtering options for specific filing attributes and are not designed with educational clarity in mind. The datasets used by these tools are typically pulled from the SEC's EDGAR database or its public API, which return filings in XML/HTML formats. These contain structured elements like insider names, titles, transaction dates, and share counts—fields that can be extracted and organized into more digestible formats using data processing tools like Python.

# Figure 1. Example Form 4 Filing

SEC Form 4

**FORM 4**

**UNITED STATES SECURITIES AND EXCHANGE COMMISSION**
Washington, D.C. 20549

**STATEMENT OF CHANGES IN BENEFICIAL OWNERSHIP**

Filed pursuant to Section 16(a) of the Securities Exchange Act of 1934
or Section 30(h) of the Investment Company Act of 1940

| OMB APPROVAL | |
|---|---|
| OMB Number: | 3235-0287 |
| Estimated average burden hours per response: | 0.5 |

☐ Check this box if no longer subject to Section 16. Form 4 or Form 5 obligations may continue. *See* Instruction 1(b).

| 1. Name and Address of Reporting Person* | 2. Issuer Name **and** Ticker or Trading Symbol | 5. Relationship of Reporting Person(s) to Issuer (Check all applicable) |
|---|---|---|
| Maestri Luca | Apple Inc. [ AAPL ] | Director / 10% Owner / X Officer (give title below) / Other (specify below): Senior Vice President, CFO |
| (Last) (First) (Middle): ONE APPLE PARK WAY | 3. Date of Earliest Transaction (Month/Day/Year): 09/26/2021 | |
| (Street): CUPERTINO CA 95014 | 4. If Amendment, Date of Original Filed (Month/Day/Year) | 6. Individual or Joint/Group Filing (Check Applicable Line): X Form filed by One Reporting Person / Form filed by More than One Reporting Person |
| (City) (State) (Zip) | | |

**Table I - Non-Derivative Securities Acquired, Disposed of, or Beneficially Owned**

| 1. Title of Security (Instr. 3) | 2. Transaction Date (Month/Day/Year) | 2A. Deemed Execution Date, if any (Month/Day/Year) | 3. Transaction Code (Instr. 8) | | 4. Securities Acquired (A) or Disposed Of (D) (Instr. 3, 4 and 5) | | | 5. Amount of Securities Beneficially Owned Following Reported Transaction(s) (Instr. 3 and 4) | 6. Ownership Form: Direct (D) or Indirect (I) (Instr. 4) | 7. Nature of Indirect Beneficial Ownership (Instr. 4) |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Code | V | Amount | (A) or (D) | Price | | | |

**Table II - Derivative Securities Acquired, Disposed of, or Beneficially Owned**
**(e.g., puts, calls, warrants, options, convertible securities)**

| 1. Title of Derivative Security (Instr. 3) | 2. Conversion or Exercise Price of Derivative Security | 3. Transaction Date (Month/Day/Year) | 3A. Deemed Execution Date, if any (Month/Day/Year) | 4. Transaction Code (Instr. 8) | | 5. Number of Derivative Securities Acquired (A) or Disposed of (D) (Instr. 3, 4 and 5) | | 6. Date Exercisable and Expiration Date (Month/Day/Year) | | 7. Title and Amount of Securities Underlying Derivative Security (Instr. 3 and 4) | | 8. Price of Derivative Security (Instr. 5) | 9. Number of derivative Securities Beneficially Owned Following Reported Transaction(s) (Instr. 4) | 10. Ownership Form: Direct (D) or Indirect (I) (Instr. 4) | 11. Nature of Indirect Beneficial Ownership (Instr. 4) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Code | V | (A) | (D) | Date Exercisable | Expiration Date | Title | Amount or Number of Shares | | | | |
| Restricted Stock Unit | (1) | 09/26/2021 | | A | | 68,065 | | (2) | (2) | Common Stock | 68,065 | $0 | 68,065 | D | |
| Restricted Stock Unit | (1) | 09/26/2021 | | A | | 68,065 | | (3) | (3) | Common Stock | 68,065 | $0 | 68,065 | D | |

**Explanation of Responses:**

1. Each restricted stock unit represents the right to receive, at settlement, one share of common stock.

## Applications

The tool we developed has a range of practical applications in both educational and financial contexts. For beginner investors, it serves as a learning platform that simplifies complex insider trading data into clear, understandable insights, helping users recognize how insider actions can influence stock prices. In financial analysis, the tool provides real-time updates from SEC Form 4 filings, allowing traders and researchers to monitor significant transactions by CEOs and major shareholders. Investors can also use the platform to teach students how to interpret market trends using real-world data. Overall, the system bridges the gap between raw SEC filings and actionable market insights, making insider trading information more accessible and useful for a wider audience.

## Methodology

1. Basic Website Creation
    a. HTML

        HTML is the foundation of the World Wide Web and is the backbone of any website. It provides the fundamental structure and content. HTML elements include headings, paragraphs, tables, forms, and links. HTML is often supported by styling languages such as Cascading Style Sheets (CSS) and programming languages such as Javascript.

    b. CSS

        Cascading Style Sheets (CSS) is a styling language used to control the visual appearance of a website. It includes styling elements, adjusting layout, managing fonts, colors, and responsiveness to enhance the user experience. With CSS, developers can ensure a website is visually appealing and accessible across different devices.

    c. JavaScript

        JavaScript is a dynamic scripting language that adds interactivity and functionality to websites. It enables real-time updates, animations, event handling, and complex user interactions, making web applications more engaging and responsive. JavaScript is essential for features like form validation, data visualization, and API interactions.

2. Python Libraries
    a. Flask

Flask is a lightweight framework that primarily provides essential features like routing and templates to create websites with Python. It is designed to be simple, extensible, and compatible with external libraries, allowing users to add advanced features easily. It is known as a part of the “micro-framework” family, meaning that it provides the necessary features to build websites without requiring too many dependencies. Its modularity and flexibility makes it a common choice for developers of all levels of expertise.

b. MatPlotLib

MatPlotLib is a 2D Python plotting library that allows users to plot various kinds of visualizations such as line plots, bar plots, scatter plots, histograms, and others. It is used to plot data trends, patterns, and distributions and thus is crucial in data analysis and scientific research. Users can easily produce meaningful graphs through simple functions like plt.plot() to gain deeper insights into the data.

c. NumPy

NumPy is a fundamental library employed for numerical computation in Python. It facilitates operations on large, multi-dimensional matrices and arrays, and has a variety of mathematical functions to perform array operations efficiently. NumPy plays a vital role in numerical analysis, linear algebra, and scientific computing. Operations like np.array() and np.mean() allow users to carry out complex calculations with ease and speed.

d. Pandas

Pandas is a data manipulation and analysis library that establishes two principal data structures: Series (1D) and DataFrames (2D, similar to Excel spreadsheets).

It is most often applied to data reading, cleaning, analysis, and data transformation. With functions like pd.read_csv() for dataset importation and df.describe() for data statistics summary, Pandas has emerged as a fundamental tool in data science, machine learning, and business analytics workflows.

e. yFinance

yFinance is a Python library that provides users with current and historical data about stocks based on data from Yahoo Finance. Examples of the information available include stock prices, financial statements, and more.

3. Data Collection

a. Web Scraping

Web scraping is the process of using an automated simulation of human browsing to extract data from websites. This is done by parsing the HTML content of web pages and retrieving certain parts of the webpage based on the type of element, ID of an element, or other specification. This can be done using simple Python libraries like BeautifulSoup, or more advanced tools like Selenium.

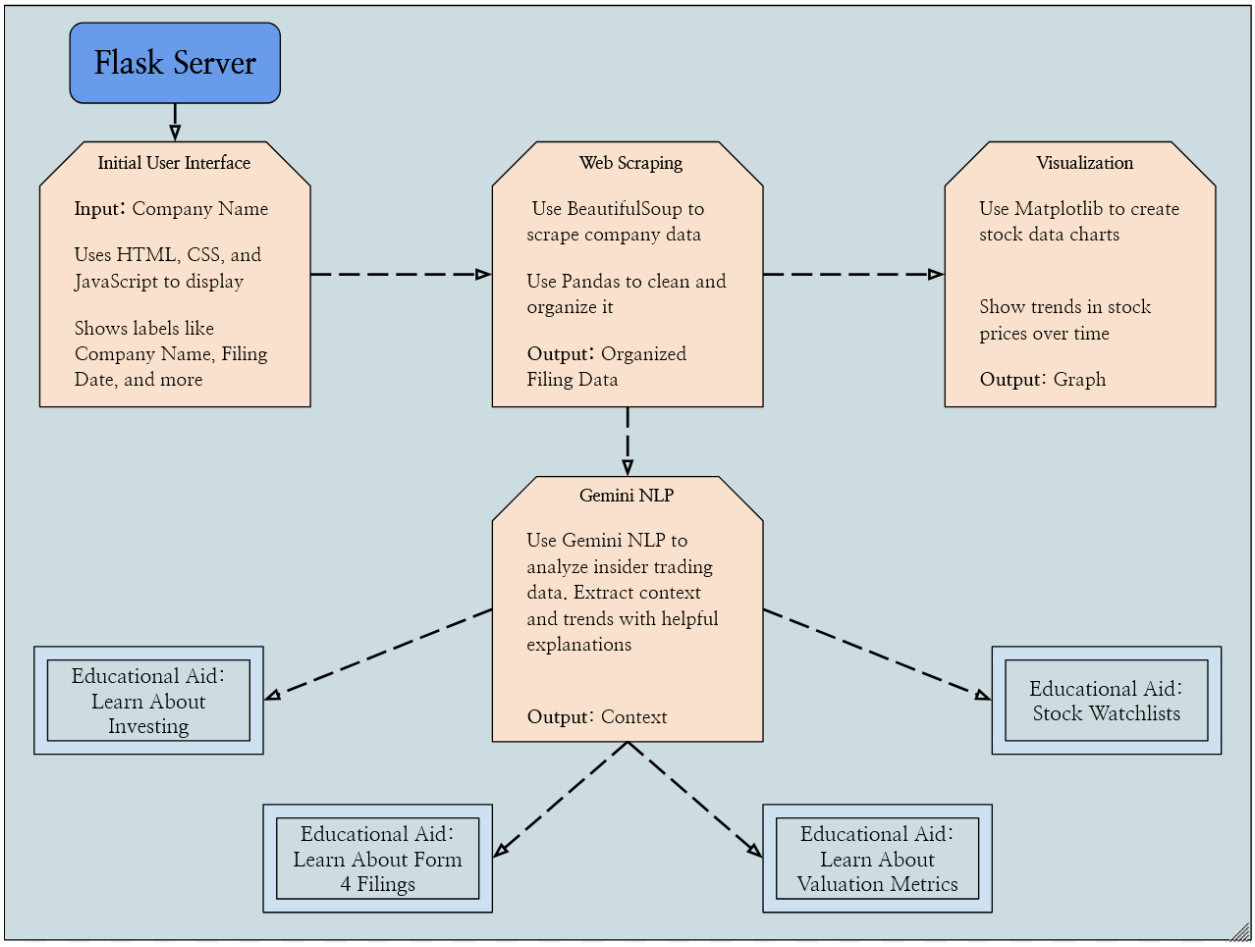
4. Data Analysis

a. Natural Language Processing

Natural language processing (NLP) is a subset of artificial intelligence based on the interaction between computers and language (Das, 2024). It involves developing algorithms that can interpret and generate human language. The application of NLP relevant to this project is sentiment analysis. This requires processes like tokenization, which involves separating chunks of text into single words that can be vectorized to be understood by the machine in a numerical

manner. Specifically, the stock analysis utilizes the Gemini API to interpret the stock movements.

**Figure 2. Systems Architecture**

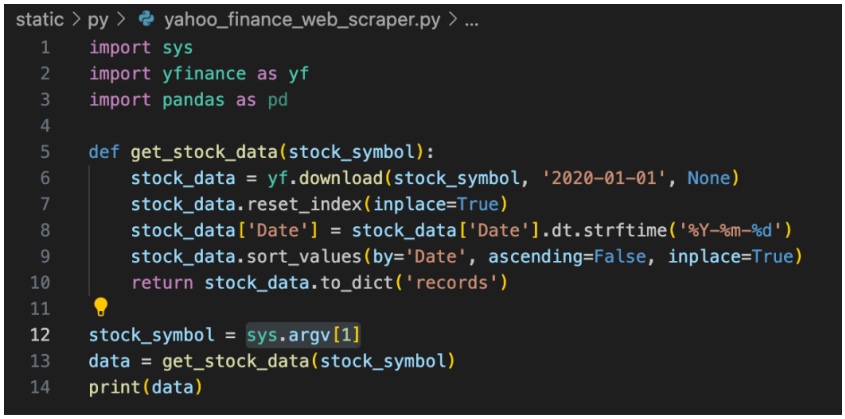


## Results

This project is deemed successful since the website provides an easy-to-understand interpretation of trends in the stock market based on Form 4 filings. It also accurately depicts historical information related to the stock.

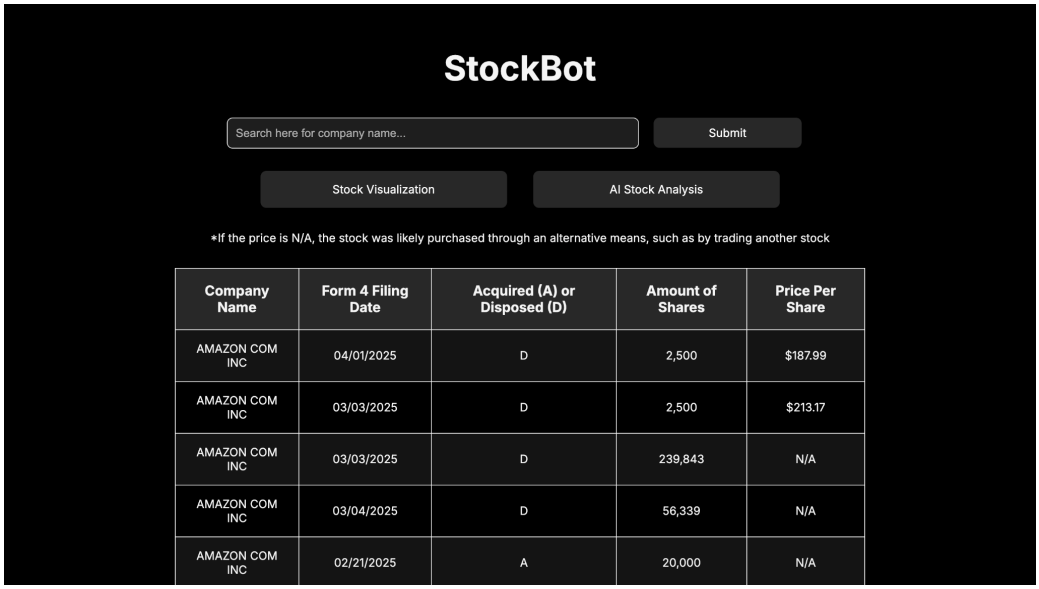
1. Developed web scraper for Yahoo Finance stock information using the yFinance library. Organized it by date with the idea of graphing it in the future.

**Figure 3. Yahoo Finance Web Scraper**



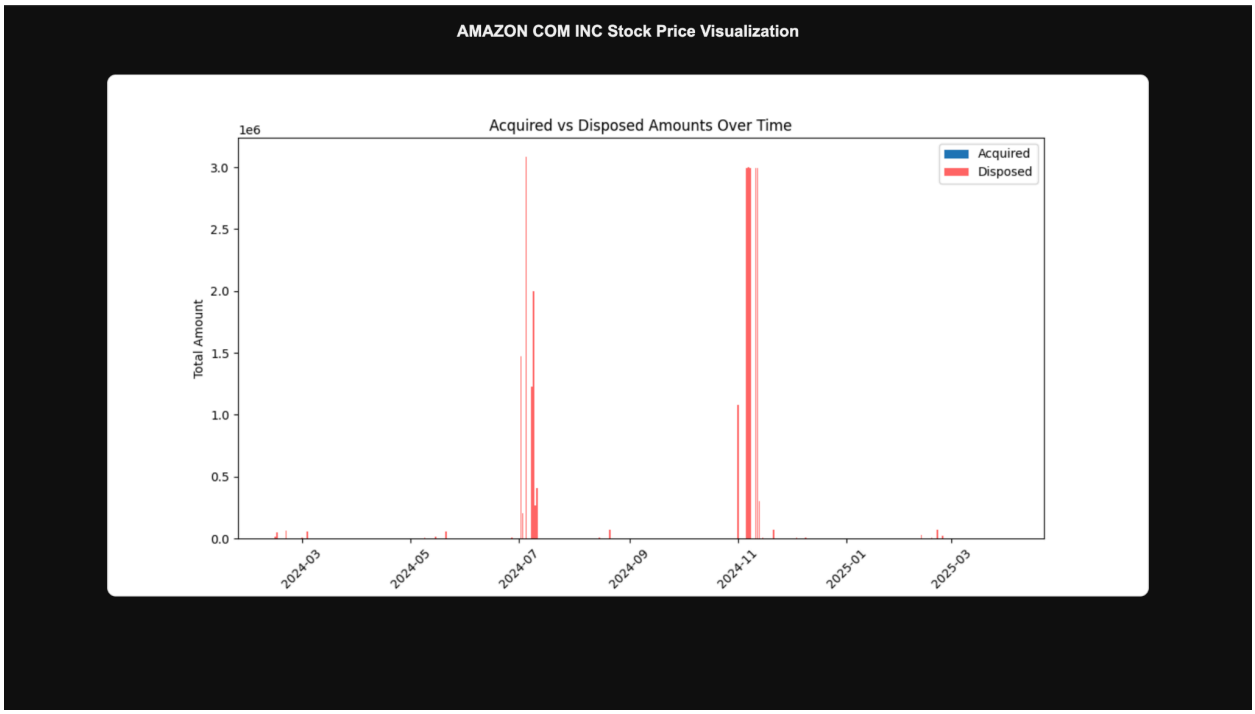

2. Developed web scraper for SEC's EDGAR API using the BeautifulSoup library. Two different functions were necessary to handle HTML files and XML files separately. Other than providing raw data from the Form 4 filings, we also provided visual interpretations of the effect on the stocks during that time period.
3. Used Flask to code the basic layout of a website with all the necessary pages, routes, and other functions (such as a search feature for specific companies).

## Figure 4. Home Page Layout



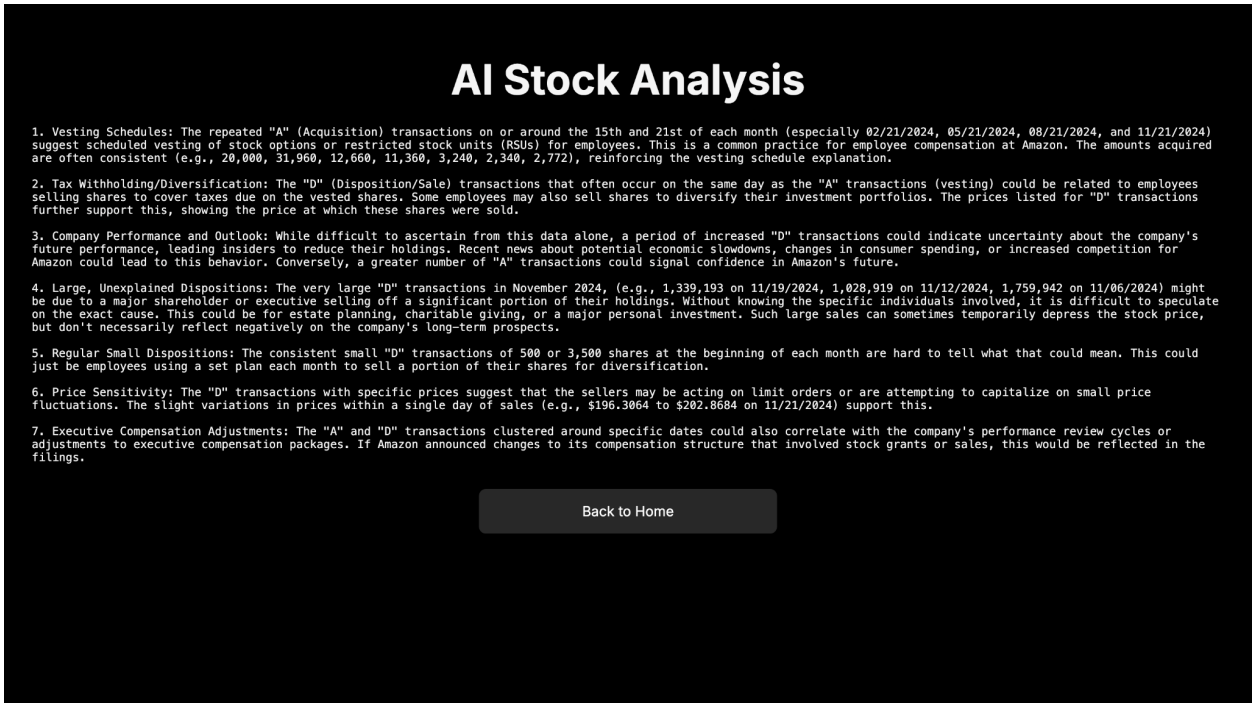
4. Integrated web scrapers and visuals for stock data with graphs into websites.

**Figure 5. Visualization of Stocks**




5. Created and integrated an AI data interpretation model to provide logical explanations as to why certain transactions were made and how they will affect the stock of the company. First, we used an API that gathered current news surrounding a company. Then, we used a different one to translate any news that was in a different language. Finally, we applied sentiment analysis to the news to determine potential alignment with trends that its stock is experiencing.

## Figure 6. AI-Based Trade Interpretation



## Limitations

One key limitation of the system is the delay between when an insider files a Form 4 and when it becomes publicly accessible on the SEC website. Although Form 4s are required to be submitted within two business days of a transaction, there is still a lag before the data appears online and can be scraped. This delay means that users are not seeing the insider trades in real time, which may reduce the immediacy of any trading or analytical decisions based on the data.

## Conclusion

Through our project, we were able to successfully develop a framework that provides public interpretation to SEC Form 4 filing information through data collection and data visualization. It combines multiple tools into a single, easy-to-use interface. Through a two-way approach scraper, we are successfully navigating the SEC's EDGAR database, scraping transaction details from both HTML and XML formats while gathering filing links, and our preprocessing functions are resolving data inconsistencies, especially in the price fields. The Matplotlib visualization components add detailed context to filing dates and is complemented by interactive features such as moving averages and cursors, where we transform the raw SEC data into information easy for beginners to understand.

Flask is the light-weight backbone of our application, integrating data collection, processing, and visualization. Our project will improve access to insider transaction information, and in the future, we can train a natural language processing model on past filings and future stock performance. It should also have a better user experience through filtering options and customizable watchlists. These enhancements would further equalize the playing field between retail and institutional investors, altering the way in which traders use insider transaction data in making investment choices.

We asked 7 of our peers to test our interface by allowing them to run our code and test its functionality for 10 minutes. Afterwards, they provided us with feedback. The general consensus was that it was very easy to navigate. In addition, they said that our AI model provided clear interpretations of the data. However, they said that they didn’t completely understand the advanced vocabulary. We think that after working on this project for so long, we forgot to

address the fact that this needs to be explained at a more basic level for those who have never heard of these terms. So, we edited our AI model's interpretation to be more understandable.

## Future Work

To enhance the system's analytical power and market relevance, future work will expand the platform to include additional SEC filings such as Form 5, offering a more comprehensive view of insider behavior over time. We also plan to integrate historical stock price data and machine learning models to detect patterns or predict potential impacts of insider trades. Improvements to the user interface—such as personalized dashboards, trade alerts, and deeper filtering options—will make the platform more interactive and insightful. Finally, scaling the system to handle more frequent updates and larger datasets will be critical for supporting real-time financial decision-making.

## References


Das, Satyesh, and Divyesh Das. “Natural Language Processing (NLP) Techniques: Usability in Human-Computer Interactions.” 2024 6th International Conference on Natural Language Processing (ICNLP), vol. 9, 22 Mar. 2024, pp. 783–787, https://doi.org/10.1109/icnlp60986.2024.10692776.

Google. Gemini API Documentation. https://ai.google.dev/gemini-api/docs.

Han, Meng, et al. Mining Public Business Knowledge: A Case Study in SEC’s EDGAR. 1 Oct. 2016, https://doi.org/10.1109/bdcloud-socialcom-sustaincom.2016.65.

Investopedia Team. “SEC Form 4: Statement of Changes in Beneficial Ownership Overview.” Investopedia, 6 June 2024, https://www.investopedia.com/terms/f/form4.asp.

Jeng, Leslie A., Andrew Metrick, and Richard Zeckhauser. “Estimating the Returns to Insider Trading: A Performance-Evaluation Perspective.” The Review of Economics and Statistics, vol. 85, no. 2, 2003, pp. 453–471, https://doi.org/10.1162/003465303765299936.

McKinney, Wes. Pandas: Powerful Python Data Analysis Toolkit. Version 2.2.2, https://pandas.pydata.org/.

Richardson, Leonard. Beautiful Soup Documentation. Crummy.com, https://www.crummy.com/software/BeautifulSoup/bs4/doc/.

U.S. Securities and Exchange Commission. EDGAR: Form 4 Filings. https://www.sec.gov/cgi-bin/browse-edgar?type=4.