Daily Financial Investment Risk Aid

Fernando Rodriguez-Guzman Jr.
School of Engineering and Applied Science
Columbia University
New York, United States
fr2510@columbia.edu

***Abstract*— This investment risk aid seeks to explore the feasibility of combining trending world topics, historical stock information, and future stock value predictions to provide a user with the right information for investment decisions leading to investment portfolio growth. Technological automation tools and virtualization techniques are utilized to leverage available datasets through daily scheduled data acquisitions. The datasets are then processed and visualized to aid an investor in their daily investment decision making process. This project aims to understand the value of utilizing daily real-time datasets of financial information, but understands the risk of attempting to predict market trends. For this reason, the financial aid does not attempt to make final investment decisions on behalf of the investor, but rather acts as a tool to provide the best information available for the investor's final investment choices.**

Keywords*— *Finance, Stocks, Mutual Funds, Google Cloud, Yahoo Finance, Virtual Environments, Risk, Investments, Visualization, DAG, Airflow, Apache

I. Introduction

The financial investment world has, in recent years, become more reliant and sensitive not only to world events, but also to the information and opinions of influential entities on social media and other communication platforms. This interconnectivity between the financial world and the potential meddling of influential entities on the market via their opinions is growing of particular importance when it comes to investment decisions made by individual and small investors. For this reason, an aggregated and unified visualizer for investment items of interest is required. This aid will assist in the decision-making process for the best daily investment actions for particular investment vehicles in the form of stocks and mutual funds.

The goal of the daily financial investment risk aid is to combine financial hard data with data from social media into one unified financial aid. This aid will provide a user with historical trending information on stocks and predictions for future closing costs in combination with trending viral words and phrases on social media. This will result in a tool which will best provide a user with daily relevant information that can be used as part of a portfolio investment decision toolset for daily use. While the financial aid does not make investment decisions on behalf of the investor, it does provide a world snapshot which can best prepare a user for the daily investment workflow resulting in portfolio growth and decision-making optimization.

II. Related Work

A range of software based financial tools are available to investors which combine the performance of their stocks and allow for the tracking of individual investment vehicles, but a toolset was not identified prior to this research which combines stock information, predictions, and world news or sentiments via twitter analysis into one web interface. One tool which enables the tracking of stock and individual's investment performance is that of Sisense's Finance System [26]. While another exemplar is that of XB Software's Rate Management System [27].

The work explored by this project will analyze the feasibility of adding further market insights by also providing investors with trending world information, stock historical data, and linear regression enabled stock closing predictions. This however does not limit this project from being a part of an investor's overall workflow as a force multiplier alongside other available software, like those depicted earlier, for investment portfolio growth.

III. Data

A total of 10 unique datasets are being collected daily for the processing and visualization of relevant financial and global event data. The first 9 datasets include the stock market historical performance of 3 subsections of investment vehicles. The first section of investment vehicles includes 3 mutual funds with historically stable returns. The next set of investment vehicles includes 3 historically profitable yet stable stocks predominantly from the energy sector. The last set of investment vehicles includes 3 fast growing stocks for rapid portfolio growth. This is followed by the daily data acquisition of trending Twitter data split into general top trending hashtags in addition to the filtering on the 9 stock's acronyms to detect if the stock is currently trending.

A. Mutual Funds

Created in 1992, Vanguard Total Stock Market Index Fund (VSMPX) [19] is designed to provide investors with exposure to the entire U.S. equity market, including small-, mid-, and large-cap growth and value stocks. The fund's key attributes are its low costs, broad diversification, and the potential for tax efficiency. Investors looking for a low-cost way to gain broad exposure to the U.S. stock market who are willing to accept the volatility that comes with stock market investing should consider this fund as either a core equity holding or only domestic stock fund.

The industry's first index fund for individual investors, the 500 Index Fund (VFIAX) [18] is a low-cost way to gain

diversified exposure to the U.S. equity market. The fund offers exposure to 500 of the largest U.S. companies, which span many different industries and account for about three-fourths of the U.S. stock market's value. The key risk for the fund is the volatility that comes with its full exposure to the stock market. Because the 500 Index Fund is broadly diversified within the large-capitalization market, it may be considered a core equity holding in a portfolio.

The Growth Fund of America's (AGTHX) [8] investment objective is to provide investors with growth of capital. This fund takes a flexible approach to growth investing, seeking opportunities in traditional growth stocks as well as cyclical companies and turnarounds with significant potential for growth of capital. Geographic flexibility also allows portfolio managers to pursue opportunities outside of the U.S. This differentiated approach has the potential to enable the fund to navigate a variety of market environments. AGTHX invests at least 65% of its assets in common stocks. It may also invest in convertibles, preferred stocks, U.S. government securities, bonds and cash equivalents.

B. Stocks

Occidental Petroleum Corporation (OXY) [4][14][21] is an American company engaged in hydrocarbon exploration in the U.S. and the Middle East as well as petrochemical manufacturing in the U.S., Canada, and Chile. It is organized in Delaware and headquartered in Houston. The company ranked 183rd on the 2021 Fortune 500 based on its 2020 revenues and 670th on the 2021 Forbes Global 2000.

ExxonMobil (XOM) [2] Corporation is an American multinational oil and gas corporation headquartered in Irving, Texas. It is the largest direct descendant of John D. Rockefeller's Standard Oil, and was formed on November 30, 1999, by the merger of Exxon and Mobil, both of which are used as retail brands, alongside Esso, for fueling stations and downstream products today. The company is vertically-integrated across the entire oil and gas industry, and within it is also a chemicals division which produces plastic, synthetic rubber, and other chemical products. ExxonMobil is incorporated in New Jersey.

Vertex Pharmaceuticals (VRTX) [7] is an American biopharmaceutical company based in Boston, Massachusetts. It was one of the first biotech firms to use an explicit strategy of rational drug design rather than combinatorial chemistry. It maintains headquarters in South Boston, Massachusetts, and three research facilities, in San Diego, California, and Milton Park, near Oxford, England.

C. Fast Growing Stocks

Enphase Energy, Inc. (ENPH) [1][22] is an American energy technology company headquartered in Fremont, California, that develops and manufactures solar micro-inverters, battery energy storage, and EV charging stations primarily for residential customers. Enphase was established in 2006 and is the first company to successfully commercialize the solar micro-inverter, which converts the direct current power generated by a solar panel into grid-compatible alternating current for use or export. The company had shipped more than 48 million microinverters to 2.5 million solar systems in more than 140 countries.

Shift4 (FOUR) [6] is an American payment processing company publicly listed on the New York Stock Exchange and based in Allentown, Pennsylvania. The company, founded in 1999 by the then 16-year-old Jared Isaacman, processes payments for over 200,000 businesses in the retail, hospitality, leisure and restaurant industries. Shift4 specializes in commerce solutions such as mobile payment software and hardware. When the company went public in 2020, Isaacman was still the CEO.

Onsemi (ON) [5] is an American semiconductor supplier company, based in Phoenix, Arizona and ranked #483 on the 2022 Fortune 500 based on its 2021 sales. Products include power and signal management, logic, discrete, and custom devices for automotive, communications, computing, consumer, industrial, LED lighting, medical, military/aerospace and power applications. Onsemi runs a network of manufacturing facilities, sales offices and design centers in North America, Europe, and the Asia Pacific regions. Based on its 2016 revenues of $3.907 billion, Onsemi ranked among the worldwide top 20 semiconductor sales leaders.

D. Trending Twitter Hashtags Phrases

Daily trending Twitter tweets are being gathered representing the day's trending events, social mindset, and any other possible noteworthy news item. The Twitter data is further subdivided into two types of datasets. The first is a list of the top hashtags regardless of region, language, or type. This is to ensure that an unbiased world view of current events is represented and visualized for the user. The second set is a curated dataset of top trending financial keywords which are of specific interest to investors. The combination of both of these datasets will allow users to determine the best investment decision based on their personal needs and portfolio growth goals.

IV. Methods

In order to provide the best possible aid for portfolio growth, a systematic approach was taken by selecting the visualization of 3 types of investment vehicles and trending world information via Twitter tweet information. All datasets are scheduled to be acquired on a daily basis and at the same time in order to ensure the maximum relevance prior to investment decisions being made by a user. This methodology ensures that an end user has all the required investment vehicle and trending world topic information in one intuitive visualized webpage. Those visualizations are in the form of a word cloud, stock line graphs, and relevant closing vs predicted closing price data.

Other approaches that could have been attempted include the visualization of more generic stock information, the exclusion of Twitter data, or the separation of datasets into different or separate webpages for visualization. All of those approaches were considered, but were ultimately not pursued due to the power of having specific investment vehicle datasets and Twitter data all in one unified webpage. Investors have a range of factors which come into play when making their investment decisions. For this reason, it is best if the financial risk aid display all of its relevant visualized dataset information into one webpage. This would allow for the integration of the risk aid into a larger investment workflow for maximum portfolio growth.

Furthermore, alternative or multi-platform design approaches could have been attempted, from a system design approach, in order to diversify the computing power of different cloud systems or platforms. This however, creates fragmentation in the backend system design which requires careful design, dependency, and timing considerations. This could ultimately lead to a faulty or inaccurate dataset acquisition system. This led to the final decision to utilized Google based cloud services for data acquisition and design.

This Google based approach ensures a unified and concise approach to the acquisition, timing, processing, and storage of all the datasets being acquired for this project. Additionally, any debugging, system integration, and processing will follow similar methodologies set by the Google based platform leading to a stable platform for data processing and visualization.

V. System Overview

The system design for this project leveraged several of Google's and Apache's resources in order to maximize the efficiency of the financial aid. The overall system design was built upon Google Cloud's Compute Engine resources for ease of interconnectivity and to leverage the available computational power. The processing pipeline was implemented via Apache's Airflow which enabled the task coordination via the use of Directed Acyclic Graphs (DAG). The DAGs were then scheduled on a daily interval via the use of Apache Airflow's Scheduler platform.

Ten major dataset acquisition requirements were implemented via the use of Yahoo Finance's *yfinance* API for 9 unique investment vehicle datasets in conjunction with Tweepy for the daily acquisition of Tweet datasets. The acquired datasets where then processed, by the DAGs, utilized to train the appropriate linear regressions models, and formatted & stored the resulting data in CSV format to a publicly accessible GitHub project repository. This repository serves to store the raw processed datasets as well as a dataset interface for the front-end HTML/CSS visualizer. This visualizer leverages the JavaScript based open-source Data-Driven Documents (D3) library for visualization. An overview of the complete system architecture is available in Fig 1.

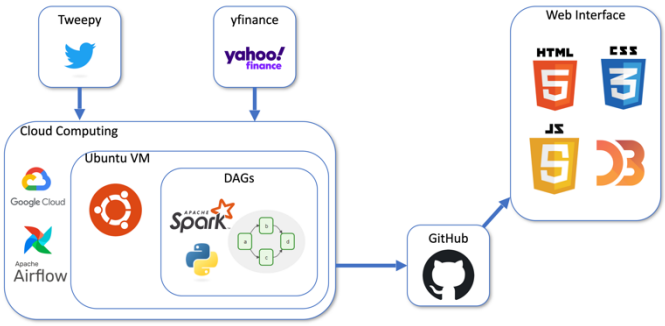

Fig. 1. System Architecture Overview

A. Google Cloud Compute Engine

The suite of cloud computing services provided by Google was utilized as the baseline for the virtualization and execution of the dataset acquisition for this project. Google Cloud's Compute Engine resources were utilized in order to leverage their Virtual Machine (VM) instance implementations. A previous project's virtual machine was leveraged and build upon for the construction of the dataset acquisition portion of this project.

The VM instance selected for this project was initialized with Ubuntu 18.04.6 LTS as the primary Operation System (OS). All the necessary packages required for this project were installed in this VM and updated as necessary. A complete set of the VM's parameters can be seen in Fig 2.

```
(airflow) fr2510@hw4:~$ cat /etc/os-release
NAME="Ubuntu"
VERSION="18.04.6 LTS (Bionic Beaver)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 18.04.6 LTS"
VERSION_ID="18.04"
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
VERSION_CODENAME=bionic
UBUNTU_CODENAME=bionic
(airflow) fr2510@hw4:~$
```

Fig. 2. Details of the Google Cloud Compute Engine VM utilized for this project.

B. Apache Airflow: Directed Acyclic Graphs

Apache Airflow, via Google Cloud, was utilized to implement the project's workflow management pipeline. The Airflow platform was organized via the use of Directed Acyclic Graphs which initialized tasks for dataset acquisition, dataset processing, and dataset storage.

The data acquisition portion of the project has been divided into three DAGs. The first DAG, depicted in Fig 3, was divided into 4 sections with a total of 12 tasks. These tasks allow for the stock data acquisition, parsing, regression model training, and CSV storage that will be ultimately be pulled by an HTML and JavaScript website for visualization.

The 4 task sections were designed to first perform a print starting operation to a terminal for logging. Next the 9 chosen investment vehicle historical datasets of interest are acquired and processed. This is followed by a wait operation to ensure all datasets are acquired. All this culminates in a print end of process to a terminal for logging purposes.

The complete task naming convention is as follows:

- Task 0: print_start
- Task 1: vsmpx_start
- Task 2: vfiax_start
- Task 3: agthx_start
- Task 4: oxy_start
- Task 5: xom _start
- Task 6: vrtx_start
- Task 7: enph_start
- Task 8: four_start
- Task 9: on_start
- Task 10: sleep_function
- Task 11: print_end

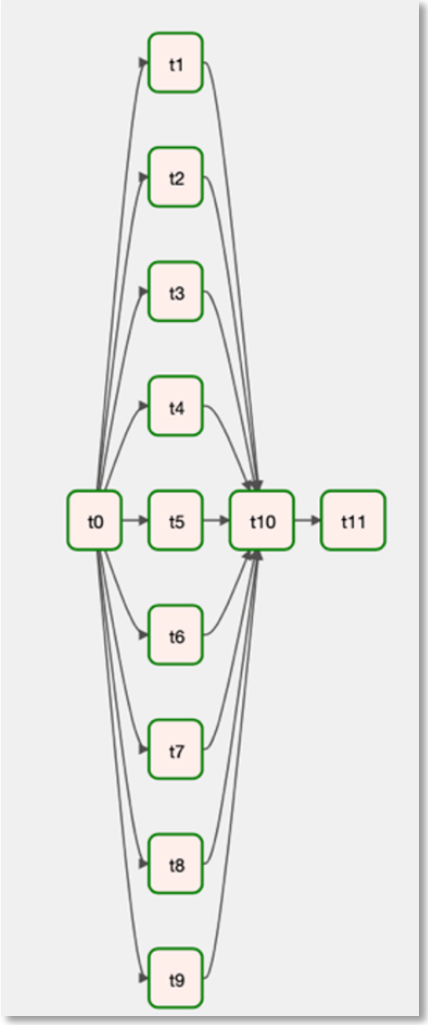

Fig. 3. First DAG which allows for the Daily Financial Investment Risk Aid's acquisition of stock data.

Next, the Twitter tweet client DAG was created which allows for the connection and acquisition of daily trending tweets. This DAG was subdivided into 2 sections and 2 tasks (Fig 4) that enable a Twitter connection.

The complete task naming convention is as follows:

- Task 0 : print_start
- Task 1: start_client

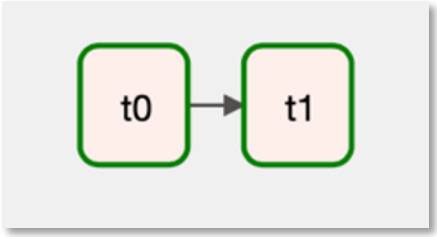

Fig. 4. Second DAG which allows for the Daily Financial Investment Risk Aid's Twitter connection for financial Tweet acquisition.

The third DAG, seen in Fig 5, connects to the Twitter Client DAG to gather trending Tweets and key words in finance, process the information, and save it to a master CSV for the visualization in the same HTML and JavaScript website.

The complete task naming convention is as follows:

- Task 0: print_start
- Task 1: start_stream
- Task 2: print_end

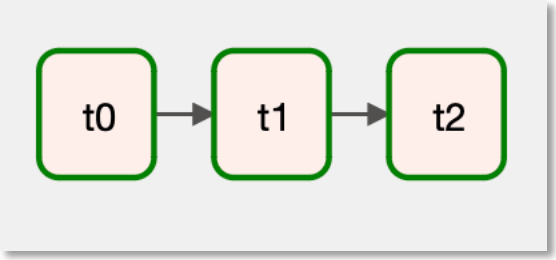

Fig. 5. Third DAG which allows for the Daily Financial Investment Risk Aid's Twitter Streaming of daily trending tweets.

C. *Apache Airflow: Scheduler*

Apache Airflow's Scheduler functionality was utilized in order to schedule all DAGs to occur daily at 7:00am EST (1200 UTC) to ensure that all of the data being acquired is of the same time and has the same level of relevance. Furthermore, 7:00am EST was chosen since this allows a user to conduct the proper follow-on research prior to the opening of the markets at 9:30am EST. Figs 6-7 depict the airflow scheduler interval initialization via the use of Cron job formatting [20].

```
with DAG(
    '1_Final_Project_Daily_Financial_Investment_Risk_Aid',
    default_args=default_args,
    description='DAG for Final Project: Daily Financial Investment Risk Aid ',
    #schedule_interval='*/05 * * * *',   # cron: min, hr, day, mth, day of week
    schedule_interval='0 12 * * *',   # At 12:00 UTC aka 07:00 EST everyday
    start_date=datetime(2022, 1, 1),
    catchup=False,
    tags=['Final_Project'],
) as dag:
```

Fig. 6. Stock data acquisition DAG airflow scheduler interval initialization.

```
with DAG(
    '2_Final_Project_TwitterClient_Daily_Financial_Investment_Risk_Aid',
    default_args=default_args,
    description='DAG for Final Project\'s Twitter Client ',
    #schedule_interval='* * * * *',   # cron: min, hr, day, mth, day of week
    #schedule_interval='*/05 * * * *',   # cron: min, hr, day, mth, day of week
    schedule_interval='0 12 * * *',   # At 12:00 UTC aka 07:00 EST everyday
    start_date=datetime(2022, 1, 1),
    catchup=False,
    tags=['Final_Project_Twitter_Client'],
) as dag:
```

Fig. 7. Twitter Client DAG airflow scheduler interval initialization.

D. *Dataset Acquisition API: Yahoo Finance*

The dataset acquisition process DAG was designed and implemented based on Yahoo Finance's *yfinance* python API library. This library was first initialized with the 9 target investment vehicles as a list used to instantiate a *yfinance* ticker for dataset gathering as seen in Fig 8.

```
# Stocks to track
ticker = ['VSMPX', 'VFIAX', 'AGTHX', 'OXY', 'XOM', 'VRTX', 'ENPH', 'FOUR', 'ON']

vsmpx = yf.Ticker(ticker[0])    # Vanguard Total Stock Market Index Fund;Institutional Plus
vfiax = yf.Ticker(ticker[1])    # Vanguard 500 Index Fund;Admiral
agthx = yf.Ticker(ticker[2])    # American Funds Growth Fund of America;A
oxy = yf.Ticker(ticker[3])      # Occidental Petroleum Corp.
xom = yf.Ticker(ticker[4])      # Exxon Mobil Corp.
vrtx = yf.Ticker(ticker[5])     # Vertex Pharmaceuticals Inc.
enph = yf.Ticker(ticker[6])     # Enphase Energy Inc.
four = yf.Ticker(ticker[7])     # Shift4 Payments Inc.
on = yf.Ticker(ticker[8])       # ON Semiconductor Corp.
```

Fig. 8. The yfinance investment vehicle ticker instantiation.

A function, called *get_stock*, was then generated to allow for the specific investment vehicle's historical data acquisition. This was followed by the storage of this each dataset as a CSV. Next, the data was processed in order to remove all unnecessary columns. This was subsequently followed by the creation of a data frame which implements an Exponential Moving Average (EMA) with a window of the 10 latest stock data values. Lastly the target dataset was split into training and testing data with a ration of 80% to 20%. This is all exemplified in Figs 9-12.

```
# Append stock informaiton
global hist
hist[stock_string] = stock_ticker.history(period='max')

# Save CSVs
hist[stock_string].to_csv(stock_csv_path)
```

Fig. 9. Stock dataset acquisition and storage to the stock's master CSV.

```
"""Process the data for training"""
# Remove other columns, but keep 'Close' price
stock_df = hist[stock_string][['Close']]
```

Fig. 10. Processing of required stock dataset columns.

```
# Add an Exponential Moving Average of the latest 10 stocks
stock_df.ta.ema(close='Close', length=10, append=True)   # Add EMA to dataframe by appending
stock_df = stock_df.iloc[10:] # Delete first ten rows since we lsot them due to averaging
```

Fig. 11. Implementation of an Exponential Moving Average to the stock dataset.

```
# Split the data into training and test data with 80% to 20% ratio
stock_X_train, stock_X_test, stock_y_train, stock_y_test = train_test_split(stock_df[['Close']],\
 stock_df[['EMA_10']], test_size=.2)
```

Fig. 12. Split of dataset into training and testing data with an 80% to 20% ratio.

Once the datasets are processed, a linear regression model is trained in order to be utilized for future stock pricing predications. Furthermore, a Mean Absolute Error (MAE) can be acquired for each day to determine how accurate the day's predictions are compared to the final real values. This implementation can be observed as Fig 13.

```
"""Training My Model"""
stock_model = LinearRegression() # Create Regression Model
stock_model.fit(stock_X_train, stock_y_train)  # Train the model
stock_y_pred = stock_model.predict(stock_X_test)   # Use model to make predictions and calculate MAE
stock_future_pred = stock_model.predict(stock_df[['Close']])   # Use model to make future predictions
```

Fig. 13. Implementation of the training of the linear regression model, MAE calculations, and future stock prediction calculations.

The final results for each investment vehicle are added to a final CSV which will be used to visualize the daily calculated information. This CSV contains the date of the acquisition the closing information, the predicated closing data, and lastly the calculation's MEA. This implementation is depicted in Fig 14.

```
"""Save MAE Entry to CSV"""
if os.path.exists(mae_csv_path):
    logging.info(f"File Exists: {os.path.exists(mae_csv_path)}")    # log that file exists

    #Append to CSV MAE Data
    data = {'Date': [str_date_time], 'Stock': [stock_string], 'Close': [stock_df['Close'][-1]],\
     'Predicted_Close': stock_future_pred[-1], 'Mean_Abs_Error': [mean_absolute_error(stock_y_test,\
      stock_y_pred)]}

    # Make data frame of above data
    df = pd.DataFrame(data)

    # append data frame to CSV file
    df.to_csv(mae_csv_path, mode='a', index=False, header=False)
```

Fig. 14. Implementation of the saving of the calculated parameters for each investment vehicle which is performed on a scheduled daily basis.

E. Dataset Acquisition API: Twitter Tweepy

Access the Twitter messages was implemented via the use of the Tweepy open-source library. This library allows for the creation of data stream client DAG which enables the connection and acquisition to tweets. The tweets are then tagged to acquire data of interest for the financial risk aid project.

Access to this data stream was implemented, via Tweepy, by providing a bearer token of the designer's developer Twitter account. This was then followed by the tagging of specific items of interest which included the '#' symbol for hashtag data acquisition and a list of investment vehicle stock nomenclature. This implementation can be observed in seen in Fig 15.

```
BEARER_TOKEN = 'XXXXXXXXXXXXXXXXXXXXXXX'
# the tags to track
tags = ['#','VSMPX', 'VFIAX', 'AGTHX', 'OXY', 'XOM', 'VRTX', 'ENPH', 'FOUR', 'ON']
```

Fig. 15. Tweepy client bearer token initialization and tag instantiation.

A client socket is then established in the final implementation steps for the Tweepy client. This socket is the instantiated to be run as a local host in the VM platform and set to function on TCP port 9001. This implementation can be observed in Fig 16.

```
class twitter_client:
    def __init__(self, TCP_IP, TCP_PORT):
      self.s = s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
      self.s.bind((TCP_IP, TCP_PORT))

    def run_client(self, tags):
      try:
        self.s.listen(1)
        while True:
          print("Waiting for TCP connection...")
          conn, addr = self.s.accept()
          print("Connected... Starting getting tweets.")
          sendData(conn,tags)
          conn.close()
      except KeyboardInterrupt:
        exit

def start_client():
    client = twitter_client("localhost", 9001)
    client.run_client(tags)
```

Fig. 16. Tweepy client socket implementation.

VI. Experimentation

The system designed for data acquisition was tested in order to prove out the proper implementing of the daily 7:00 am EST (1200 UTC) scheduled DAGs for Financial and Twitter dataset acquisitions. A daily historical dataset was successfully acquired for the investment vehicle datasets as seen in the VSMPX example in Fig 17. This was followed by the processing of each dataset, training of linear regression models for each investment vehicle, and MEA calculations. The final combined appended daily CSV can be observed in Fig 18.

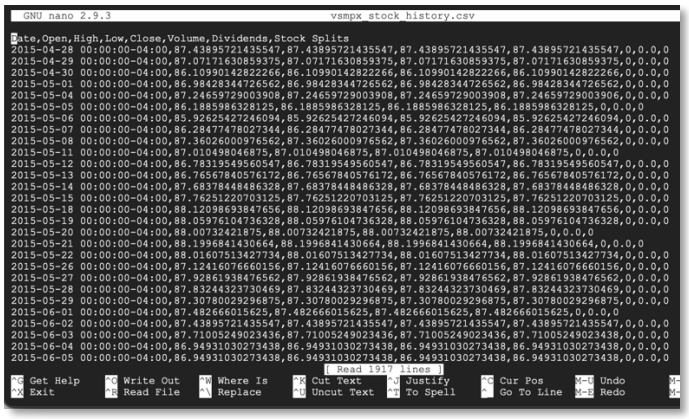

Fig. 17. VSMPX acquired historical dataset saved as CSV example.

Fig. 18. Combined daily investment vehicles and their closing data, predicted closing data, and MAE calculation.

This was followed by the successful acquisition of daily Twitter Tweets. The Tweets were parsed into two dataset CSVs for later visualization. The first, seen in Fig 19, is that of all the major daily tweets regardless of language, region, or financial interest. This was done to provide a world trending snapshot to investors. Next, a list of key words of interest was utilized to create a CSV for later visualization containing those words of interest and their number of occurrences during the acquisition window. This is demonstrated in Fig 20.

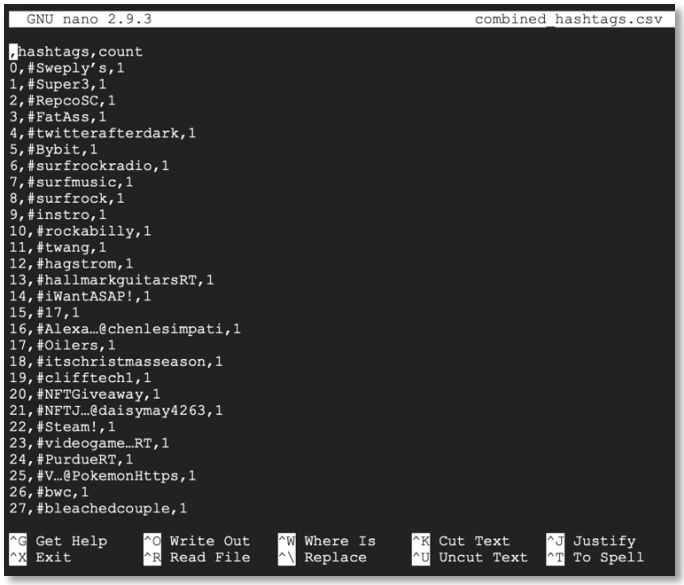

Fig. 19. Combined daily Twitter hashtag information.

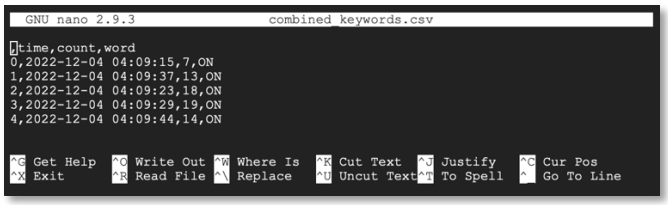

Fig. 20. Combined daily Twitter appearance of data of interest.

VII. Visualization

All of the acquired and processed datasets were stored, in CSV format, on a publicly available GitHub repository. This enabled the daily visualization of the processed information via a generated website. This website was created by utilizing HTML and CSS scripting alongside D3 [24][25] JavaScript scripts to generate one cohesive financial aid web-based tool for the quick visualization and access to financial investment vehicle and trending tweet hashtag data.

A. Visualizer: Tweet Word Cloud

The visualization webpage commences by providing the financial aid user with a D3 style word cloud of the DAG processed trending hashtags related to daily noteworthy news, events, and social mindsets. The proportion of the text's size quickly demonstrates the popularity and occurrence of each of the processed phrases. This therefore allows a user to quickly note any events that may influence their daily investment decisions as part of their daily investment workflow. Fig 21 depicts the D3 loading of the processed hashtag CSV information. The final website word cloud visualization can be observed as Fig 22.

```
//Start of Word Cloud
d3.csv("https://raw.githubusercontent.com/FernandoEE/\
  EECS6893_Daily_Financial_Aid/main/Financial_Data/tweets/combined/combined_hashtags.csv",
  function(data) {
```

Fig. 21. Code for the D3 enabled loading of the CSV Tweet data available via the project's GitHub repository.

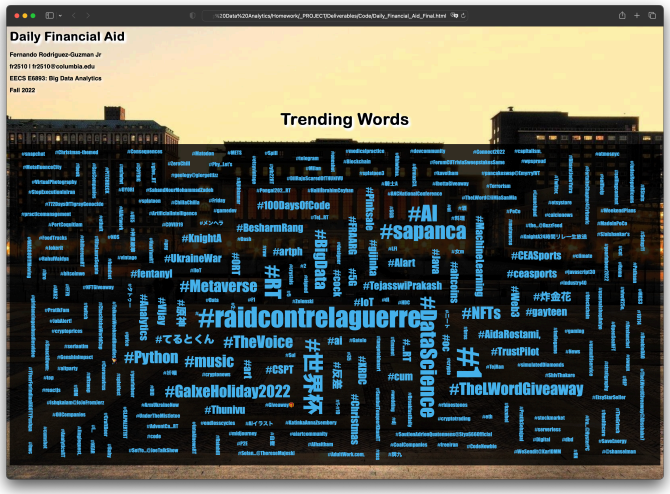

Fig. 22. Daily Financial Risk Aid web interface visualizing daily trending Twitter information via a D3 enabled Word Cloud visualizer.

B. *Visualizer: Investment Vehicle Visualization*

The processed investment vehicle datasets were visualized after the displayed Word Cloud information set as part of the unified Daily Financial Risk Aid. Each of the 9 investment vehicles' DAG-Processed CSV stock history and linear regression models were loaded via D3 scripts (Fig 23-24) and subsequently visualized via D3 enabled historical line graphs (Fig 25). Furthermore, the linear regression model for each stock allowed for the stock prediction of closing prices vs predicted closing prices to be visualized on a single line interactive graph chart as seen in Fig 26.

```
// Start of AGTHX_Stock_History
d3.csv("https://raw.githubusercontent.com/FernandoEE/\
  EECS6893_Daily_Financial_Aid/main/Financial_Data/agthx_stock_history.csv",
  // When reading the csv, I must format variables:
  function(d){
    return { date : d3.timeParse("%Y-%m-%d")(d.Date.slice(0,10)), value : d.Close }
  },
  // Now I can use this dataset:
  function(data) {
```

Fig. 23. Code exemplifying the D3 enabled loading of the AGTHX sotck history CSV.

```
// Start of AGTHX_Stock_MAE
d3.csv("https://raw.githubusercontent.com/FernandoEE/\
  EECS6893_Daily_Financial_Aid/main/Financial_Data/stock_mae_history.csv",
  function(d){
      return { date : d3.timeParse("%Y-%m-%d")(d.Date.slice(0,10)),
      Stock: d.Stock, Close: d.Close, Predicted_Close: d.Predicted_Close}
  },
  // Now I can use this dataset:
  function(data) {
```

Fig. 24. Code exemplifying the D3 enabled loading of the AGTHX stock linear regression results CSV.

Fig. 25. Exemplar of the Daily Financial Risk Aid web interface visualizing AGTHX's stock performance history via a D3 enabled line graph visualizer.

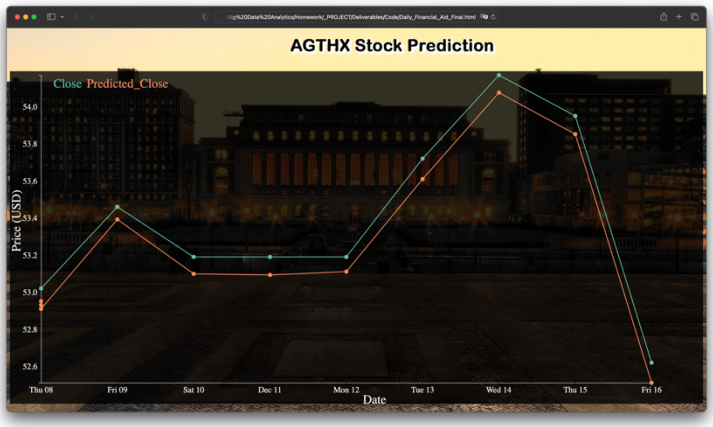

Fig. 26. Exemplar of the Daily Financial Risk Aid web interface visualizing AGTHX's computed linear regression prediction of closing cost vs actual closing cost via a D3 enabled interactive line graph visualizer.

VIII. Conclusion

The designed Daily Financial Risk Aid successfully enabled the exploration for the feasibility of combining trending world topics, historical stock information, and future stock value predictions to provide a user with the right information for investment decisions leading to investment portfolio growth. Through this project, a unified cloud computing platform was generated which provides investors with the appropriate investment vehicle and trending world topics for their daily investment workflow. Furthermore, this project successfully accomplished the fusion of a cloud computing back end, leveraged open-source dataset acquisition libraries, utilized open-source repository frameworks for storage, and developed a unified user facing front end financial visualizer.

Based on the concepts proven by this project, future work can therefore focus on enabling the visualization of different investment vehicles, sourcing a variety of social discord from other social media platforms, and utilize further models for investment predictions. This will enable greater fusion of modern technological frameworks and finance for further investment growth potentials.

IX. Acknowledgment

The author would like to acknowledge Dr. Ching-Yung Lin for all of his instruction and guidance throughout this project and the Big Data Analytics course. His passion and instruction during this course have given the author the necessary tools to explore and design future data analytics and visualization toolsets both in academia and in his professional work. The author would also like to acknowledge Rui Chu, Srividya Inampudi, Tejasri Kurapati, and Yunhang Lin for all of their thorough guidance and instruction as TAs for EECS 6983 Big Data Analytics. Their time and assistance throughout the entire course was greatly appreciated.

References

[1] Google. (n.d.). *Enphase Energy Inc (ENPH) stock price & news*. Google Finance. Retrieved December 4, 2022, from https://www.google.com/finance/quote/ENPH:NASDAQ?window=MAX

[2] Google. (n.d.). *Exxon Mobil Corp (XOM) stock price & news*. Google Finance. Retrieved December 4, 2022, from https://www.google.com/finance/quote/XOM:NYSE?sa=X&ved=2ahUKEwj0zrSVvd77AhWvFVkFHdJ-DZMQ3ecFegQINxAY

[3] Google. (n.d.). *Google Cloud*. Google. Retrieved December 4, 2022, from https://cloud.google.com/

[4] Google. (n.d.). *Occidental Petroleum Corporation (OXY) Stock Price & News*. Google Finance. Retrieved December 4, 2022, from https://www.google.com/finance/quote/OXY:NYSE

[5] Google. (n.d.). *On Semiconductor Corp (ON) stock price & news*. Google Finance. Retrieved December 4, 2022, from https://www.google.com/finance/quote/ON:NASDAQ

[6] Google. (n.d.). *Shift4 Payments Inc (four) stock price & news*. Google Finance. Retrieved December 4, 2022, from https://www.google.com/finance/quote/FOUR:NYSE

[7] Google. (n.d.). *Vertex Pharmaceuticals Incorporated (VRTX) stock price & news*. Google Finance. Retrieved December 4, 2022, from https://www.google.com/finance/quote/VRTX:NASDAQ

[8] *The Growth Fund of America (AGTHX)*. A. (n.d.). Retrieved December 4, 2022, from https://www.capitalgroup.com/individual/investments/fund/agthx

[9] *Home*. Apache Airflow. (n.d.). Retrieved December 4, 2022, from https://airflow.apache.org/

[10] *How to merge multiple CSV files into a single pandas dataframe ?* GeeksforGeeks. (2021, May 9). Retrieved December 4, 2022, from https://www.geeksforgeeks.org/how-to-merge-multiple-csv-files-into-a-single-pandas-dataframe/

[11] Jesse Reza KhorasaneeJesse Reza Khorasanee 2. (1965, June 1). *How to push local files to github using python? (or post a commit via python)*. Stack Overflow. Retrieved December 4, 2022, from https://stackoverflow.com/questions/50071841/how-to-push-local-files-to-github-using-python-or-post-a-commit-via-python

[12] Joshua, S. (2022, April 25). *How to combine multiple CSV files using Python for your analysis*. Medium. Retrieved December 4, 2022, from https://medium.com/@stella96joshua/how-to-combine-multiple-csv-files-using-python-for-your-analysis-a88017c6ff9e

[13] Nnk. (2022, November 20). *Pandas add column names to DataFrame*. Spark by {Examples}. Retrieved December 4, 2022, from https://sparkbyexamples.com/pandas/pandas-add-column-names-to-dataframe/

[14] *OCCIDENTAL PETROLEUM CORP OXY*. NYSE. (n.d.). Retrieved December 4, 2022, from https://www.nyse.com/quote/XNYS:OXY

[15] *Top 25 mutual funds*. MarketWatch. (n.d.). Retrieved December 4, 2022, from https://www.marketwatch.com/tools/top-25-mutual-funds

[16] *Tweepy Documentation*. Tweepy Documentation - tweepy 4.12.1 documentation. (n.d.). Retrieved December 4, 2022, from https://docs.tweepy.org/en/stable/

[17] Twitter. (n.d.). Twitter. Retrieved December 4, 2022, from https://twitter.com/

[18] *Vanguard Mutual Fund profile (VFIAX)*. Vanguard. (n.d.). Retrieved December 4, 2022, from https://investor.vanguard.com/investment-products/mutual-funds/profile/vfiax

[19] *Vanguard Mutual Fund profile (VSMPX)*. Vanguard. (n.d.). Retrieved December 4, 2022, from https://investor.vanguard.com/investment-products/mutual-funds/profile/vsmpx

[20] Wikimedia Foundation. (2022, November 14). *Cron*. Wikipedia. Retrieved December 4, 2022, from https://en.wikipedia.org/wiki/Cron

[21] Wikimedia Foundation. (2022, October 19). *Occidental Petroleum*. Wikipedia. Retrieved December 4, 2022, from https://en.wikipedia.org/wiki/Occidental_Petroleum

[22] Yahoo! (2022, December 1). *3 reasons why growth investors shouldn't Overlook Enphase Energy (ENPH)*. Yahoo! Finance. Retrieved December 4, 2022, from https://finance.yahoo.com/news/3-reasons-why-growth-investors-174505497.html

[23] *Yfinance*. PyPI. (n.d.). Retrieved December 4, 2022, from https://pypi.org/project/yfinance/

[24] Holtz, Y. (n.d.). *The D3 graph gallery – simple charts made in d3.js*. The D3 Graph Gallery – Simple charts made with d3.js. Retrieved December 20, 2022, from https://d3-graph-gallery.com/index.html

[25] Bostock, M. (n.d.). *Data-driven documents*. D3.js. Retrieved December 20, 2022, from https://d3js.org/

[26] *Finance Analytics & Bi Software*. Sisense. (2022, October 24). Retrieved December 20, 2022, from https://www.sisense.com/solutions/finance/?utm_source=quora&utm_medium=referral

[27] *Financial charting software for Stock Market Data Visualization*. XB Software. (n.d.). Retrieved December 20, 2022, from https://xbsoftware.com/case-studies-webdev/rate-management-system/