Electric Vehicles (EVs)

Group No: 23

Roll Numbers:

24280030

24280054

Repo Link:

https://github.com/aht007/data-engineering

Contributions:

We both took each other's input in almost all the tasks. Ahmad worked on reddit script mainly while Ahtasham worked on Yfinance script. All other things were discussed and collaborated by both of us. We prepared the report mutually and discussed all the various questions being asked. We identified several open source datasets and chose this one as it was most closely related to our use case and was also from US govt's official website.

Why Did We Choose This Topic?

We chose to analyze the **electric vehicles (EVs) sector** because it's one of the fastest-growing industries, with companies like **Tesla, Rivian, and BYD** leading the shift toward sustainable transportation. Given the increasing global push for **clean energy, government incentives, and advancements in battery technology**, EV stocks are a key focus for investors and analysts.

By looking at **historical stock data**, we wanted to understand how different EV companies have performed over the past two years and how their stock prices react to external factors like market trends, policy changes, and industry disruptions. This kind of analysis can provide insights into **investment potential, volatility, and overall market sentiment** surrounding EVs.

What Data Did We Expect to See?

For this project, we expected to gather historical **stock closing prices** for major EV companies. We process the closing prices of the stocks which can provide us with the following valuable insights:

- **Daily closing prices** of selected EV stocks
- We can infer **Market trends** showing how stock prices have fluctuated over time
- We can infer **patterns** in stock prices, giving us knowledge about sharp movements in prices.
- Perform **a Comparative** study about different stocks

This data can help in identifying **investment opportunities, market trends, and industry growth patterns** in the EV space.

Data Collection Process

Reddit Data Collection

For the Reddit data collection, we performed the following steps:

- **Subreddit lookup**

  We identified and selected the relevant subreddits that discuss EV trends, marktets and ETFs. The step involved analyzing the subreddits activties, engagements and interests.

- **Finding relevant keywords**

  It was a challenge to find relevant keywords that could be used to target the posts which discuss the EV markets, trends and companies. These keywords were used to extract the relevant posts.

- **Handling authentication and errors**

  Reddit APIs require OAuth authentication to provide its access. For that, we implemented proper authentication and error handling. We also took care of rate limits by extracting a limited amount of posts.

These steps were followed to create a quality dataset of proper posts while minimizing the costs.

yfinance Data Collection

For the yfinance component, We followed these steps:

- **Ticker Lookup:**

  We started by identifying the EV companies and ETFs to track. However, we discovered that some well-known EV brands had tickers and ETFs that didn't match their common names. This required additional web research to accurately map the correct tickers.

- **Fetching Data:**

  Using the yfinance library, we extracted 2 years of historical stock data, specifically focusing on the 'Close' price. This allowed us to generate a time-series dataset for each selected ticker.

- **Handling Errors:**

  During data retrieval, we encountered errors such as:

  - `$BrandName: possibly delisted; no timezone found`
    This error seems to be part of the library's built-in error handling, which handles delisted or non-existent stocks. We documented these occurrences and made note of potential data gaps for analysis.

Datasets Collection

While exploring additional datasets to supplement the EV analysis, We faced several challenges:

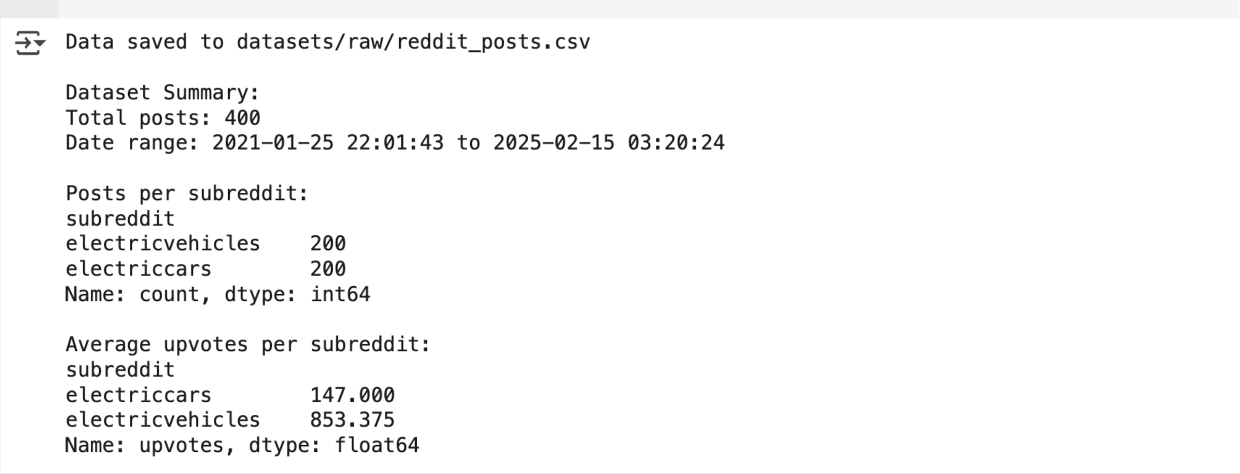
- **Limited Public Datasets:**
  There is a lack of comprehensive and freely available public datasets specifically focused on electric vehicles.

- **API Cost Barriers:**
  Many available datasets with API access are behind a paywall, which restricts free access and also limits access to parts of data.

- **Other Constraints:**
  We noticed that there are some datasets that constrain the usage of data. For example, some datasets allowed using the data for educational purposes or personal usage but not for professional/business purposes.

Observations

Reddit



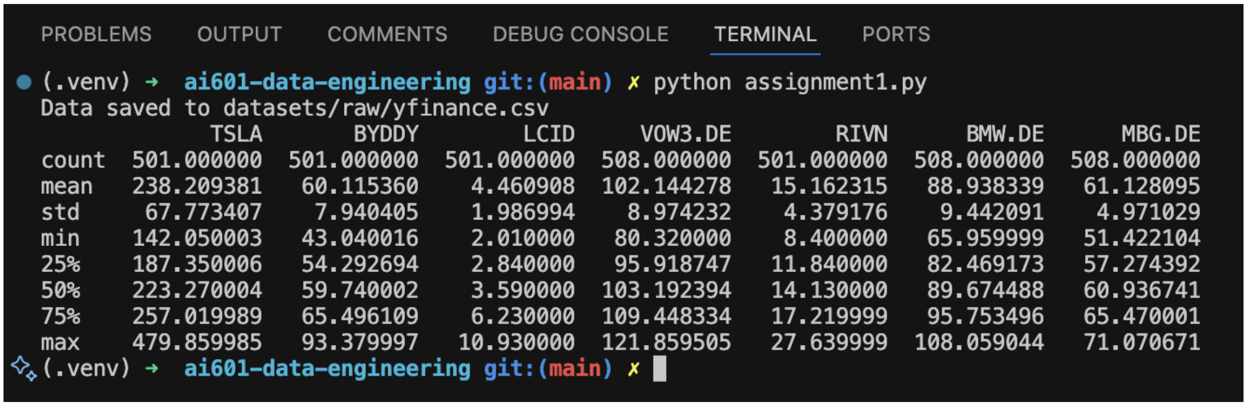
```
Data saved to datasets/raw/reddit_posts.csv

Dataset Summary:
Total posts: 400
Date range: 2021-01-25 22:01:43 to 2025-02-15 03:20:24

Posts per subreddit:
subreddit
electricvehicles    200
electriccars        200
Name: count, dtype: int64

Average upvotes per subreddit:
subreddit
electriccars        147.000
electricvehicles    853.375
Name: upvotes, dtype: float64
```

Yfinance



```
● (.venv) → ai601-data-engineering git:(main) ✗ python assignment1.py
  Data saved to datasets/raw/yfinance.csv
               TSLA       BYDDY        LCID     VOW3.DE        RIVN      BMW.DE      MBG.DE
  count  501.000000  501.000000  501.000000  508.000000  501.000000  508.000000  508.000000
  mean   238.209381   60.115360    4.460908  102.144278   15.162315   88.938339   61.128095
  std     67.773407    7.940405    1.986994    8.974232    4.379176    9.442091    4.971029
  min    142.050003   43.040016    2.010000   80.320000    8.400000   65.959999   51.422104
  25%    187.350006   54.292694    2.840000   95.918747   11.840000   82.469173   57.274392
  50%    223.270004   59.740002    3.590000  103.192394   14.130000   89.674488   60.936741
  75%    257.019989   65.496109    6.230000  109.448334   17.219999   95.753496   65.470001
  max    479.859985   93.379997   10.930000  121.859505   27.639999  108.059044   71.070671
  (.venv) → ai601-data-engineering git:(main) ✗ 
```

Public Datasets

https://catalog.data.gov/dataset/electric-vehicle-title-and-registration-activity

We have also added a python script to perform some analysis on the dataset and also added the output to repo in the file named: `public_dataset_analysis.txt`

The CSV file has also been added to the repo as well.

What AI product will you make using this data?

We would build an **EV Stock & Sentiment Analysis Tool** powered by **AI** that helps investors track trends and make better decisions. By combining historical stock data from Yahoo Finance with real-time sentiment analysis from Reddit, the tool would provide insights into how public perception aligns with stock performance. Using machine learning, it could predict potential price movements, detect shifts in sentiment, and send alerts when something significant happens..

Which terms of service constraints or privacy issues might arise when collecting data from Reddit and Yahoo? Consider limitations on storing or redistributing user-generated content.

When collecting data from Reddit and Yahoo Finance, we have to keep in mind that they impose several restrictions on usage and redistribution of the data. Reddit's API has restrictions on storing and redistributing user-generated content, meaning that one can analyze and summarize data but can't publicly share raw posts or comments without proper citation or attribution. Yahoo Finance, on the other hand, provides stock market data, but it's mainly for personal or research use—redistributing or using it commercially could violate their TOS. Another issue we ran into was that some EV-related stocks were either delisted or had unexpected ticker symbols, which required additional research. To stay compliant, We made sure to only use the data for analysis without redistributing it and followed the platform's rate limits and compliance with TOS to avoid API access issues.

How does collecting from multiple sources help or hinder data quality? What conflicts or discrepancies might you face?

Collecting data from multiple sources helps improve data quality by providing a more complete picture, filling in gaps, and allowing cross-checking. For example, in my EV stock analysis, Yahoo Finance gave structured historical stock prices, while Reddit could offer real-time sentiment and industry discussions. Combining these sources helps capture trends that might not be obvious from just one dataset.

But pulling data from different sources also brings challenges. One of the biggest issues is **inconsistency in data formats**—Yahoo Finance provides clean numerical data, while Reddit is mostly unstructured text. **Timing differences** are another problem since stock prices update daily or in real-time, while sentiment on Reddit shifts much faster. There's also the risk of **biased or unreliable data**, especially from social media. To deal with this, we have to clean and standardize the data, to make sure everything lines up correctly.

Can you think of ways to store and combine all of this data?

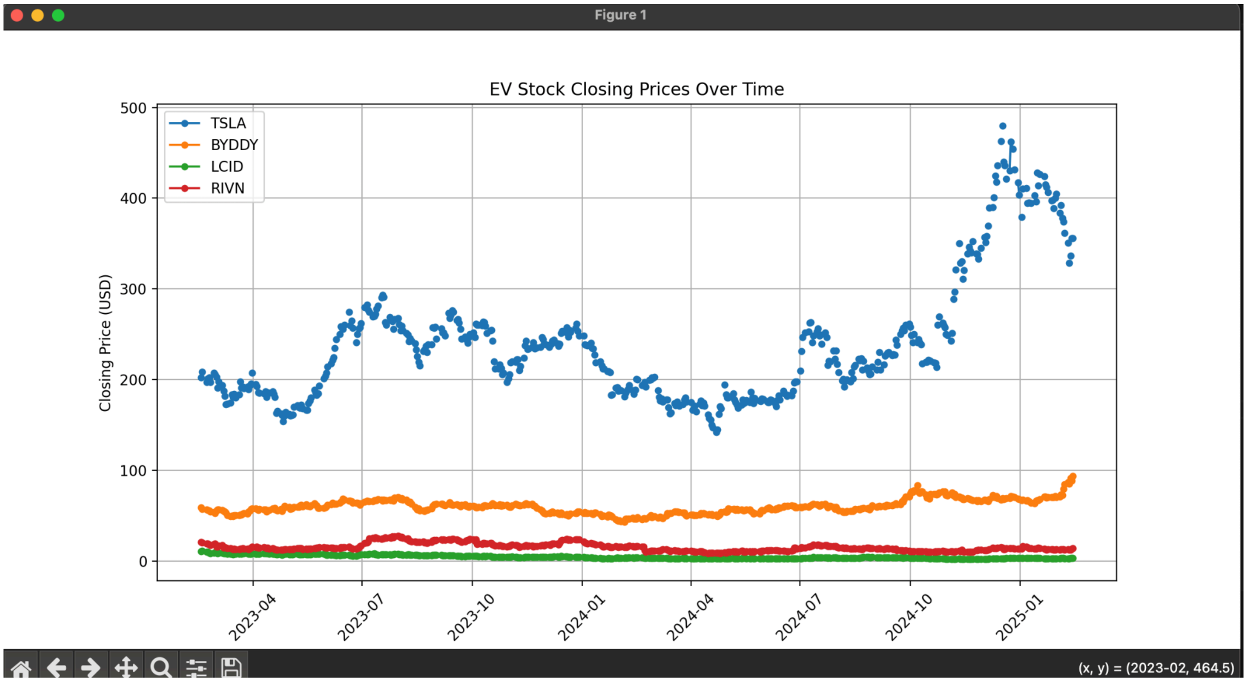
To store and combine all the collected data, we would use a structured approach with a combination of databases like Postgres and data processing tools and languages like Python and Pandas. Since Yahoo Finance provides structured numerical data, and Reddit has unstructured text, a relational database like PostgreSQL or a NoSQL database like MongoDB could be used depending on the data type.

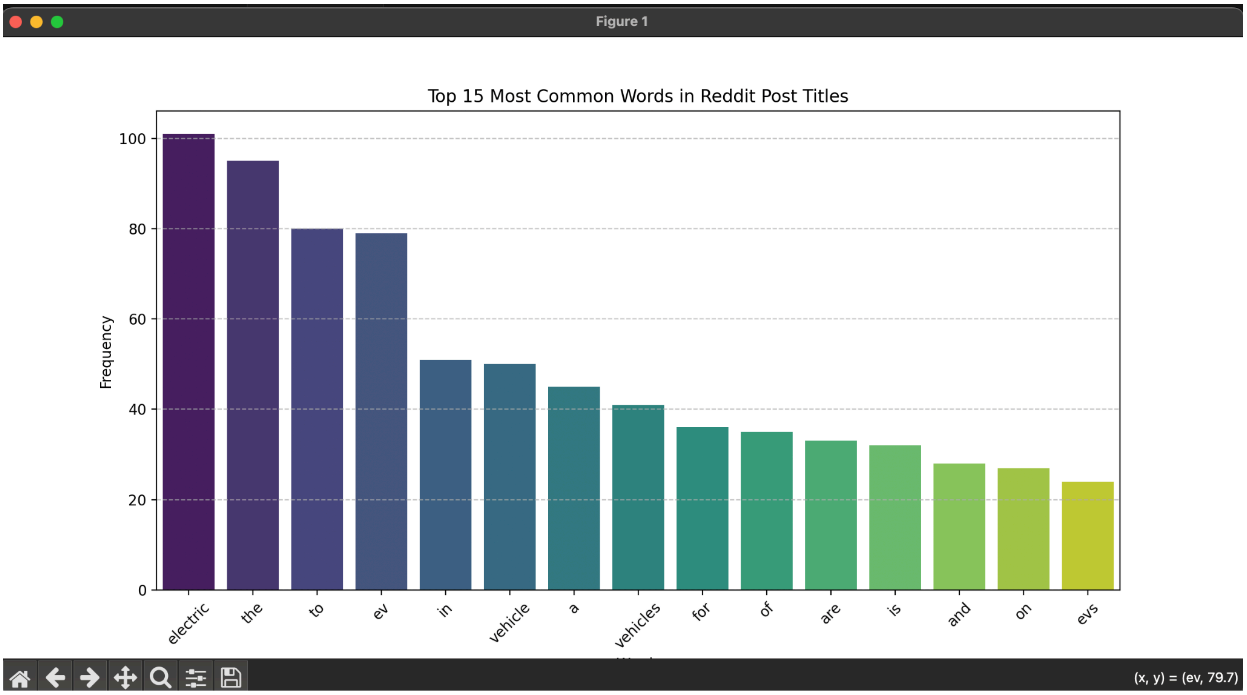
For structured financial data (stock prices, trends), PostgreSQL(or any other SQL database) would be ideal since it allows efficient querying and time-series analysis. We can decide our own model as to whether we want to store all data in one table or store stocks of different companies in different tables.

For unstructured text data from Reddit, MongoDB or Elasticsearch would work better since they handle JSON-like documents and allow full-text searches on posts and comments.

To combine the data, we would use any scripting language such as Python. we could use Pandas for preprocessing, cleaning, and merging datasets based on some attribute example timestamp. Ultimately, one would use Python scripts to aggregate and analyze stock trends alongside sentiment from Reddit, ensuring everything aligns properly before drawing insights.

Charts:





*** For the public dataset detailed analysis has been added to the repo.**