Stock Movement Analysis based on the Social Media Sentiments

Data Scraping:

1. Authentication to Reddit's API:
   The data scraping process started with creating an account on Reddit and an application named Rscraper on this website. After making the application, I got the authentication credentials, which helped me establish a connection with Reddit's API. This ensured authorized access to fetch data from Reddit. The provided Python script leverages the **PRAW (Python Reddit API Wrapper)** library to scrape data from the Reddit API.
2. Defining the Target Subreddit:
   The subreddit () method is used to specify the subreddit you want to scrape, in this case, "stocks."
3. Scrapping Recent Posts:
   To fetch the most recent posts from the subreddit, the subreddit.new() method used to fetch the most recent post on Reddit and also set limit=1000.
4. Extracting Post Attributes:
   To capture relevant data fields from each post, such as the title, content, score, number of comments, and creation time. Attributes like "post.title" and "post.selftext" are extracted for analysis. Data is stored in a dictionary, which is appended to a list (posts).
5. Converting Data to Pandas DataFrame:
   To structure the scraped data in a tabular format for easier manipulation and analysis.

Challenges Encountered and their Resolutions:

- **Authentication Issues:**
  Challenge:
  Invalid or expired credentials can prevent successful API authentication, resulting in errors.
  Resolution:
  - Double-check credentials against Reddit's developer portal.
  - Ensure the client_id, client_secret, and user_agent are correctly specified.
  - Avoid sharing credentials publicly.
- **API Rate Limits:**
  Challenge:
  Reddit enforces a rate limit on API requests (60 requests per minute by default). Exceeding this limit results in temporary blocks.
  Resolution:
  Reduce the number of API calls, e.g., fetch fewer posts (limit=1000 instead of 2000).
- **Data Volume and Storage**
  Challenge:
  Large volumes of data can exceed memory or cause processing delays.
  Resolution: Store data incrementally (e.g., write to a CSV file in chunks)

Feature Extraction and Their Relevance to Stock Movement Predictions

In feature extraction, multiple features are extracted from Reddit scraped data like: body, title, comments, and score for further processing body and title features are combined in the new column content for sentiment analysis after the sentiment analysis we got a sentiment score for each content and saved it in new column sentiment.

After the sentiment analysis for stock movement prediction, historical stock price data was scraped using yfinance which contains multiple features like: Label, change_price, close_date, Volume, etc.

The label feature contains 0 and 1(0 shows the price decrease and 1 shows the price increase).

The stock movement prediction considers sentiment, volume, and label features.

If sentiment is positive and Volume is high then its price will also increase.

Model evaluation metrics

In model evaluation metrics two factors are discussed which are:

1. **Accuracy**:
    - **89%** accuracy indicates the model correctly predicts 89% of the labels. This is a strong performance overall.
2. **Classification Report (Precision, Recall, and F1-Score)**:
    - **Class 0 (Negative Class)**:
        - **Precision (90%)**: Of all predictions made as Class 0, 90% are correct.
        - **Recall (86%)**: The model correctly identifies 86% of all true Class 0 instances.
        - **F1-Score (88%)**: A balance between precision and recall for Class 0.
    - **Class 1 (Positive Class)**:
        - **Precision (88%)**: Of all predictions made as Class 1, 88% are correct.
        - **Recall (92%)**: The model correctly identifies 92% of all true Class 1 instances.
        - **F1-Score (90%)**: A balance between precision and recall for Class 1.
3. **Macro and Weighted Averages**:
    - **Macro Avg (89%)**: Indicates the model performs equally well across both classes.
    - **Weighted Avg (89%)**: Accounts for the class distribution and shows balanced performance.

Performance Insights

- **Balanced Metrics**: The precision, recall, and F1-scores for both classes are close, indicating consistent performance.
- **Class Strengths**:
    - Class 0 has slightly better precision, meaning fewer false positives.
    - Class 1 has higher recall, meaning fewer false negatives.
- **High Accuracy**: Reflects that the model generalizes well to the given dataset.