Stock trend prediction based on social media articles

Rishabh Paraswani
rishiparaswani@gmail.com

Abstract—This paper presents a machine-learning approach to predicting stock trends using social media articles. The proposed method consists of three steps: data scraping, data analysis, and prediction models. First, relevant social media articles are collected from different sources. Then, the acquired data is preprocessed by filtering out stop words, punctuation, and non-alphanumeric characters. Finally, sentiment analysis is conducted using the Classification algorithms to determine the sentiment of the article and the corresponding stock trend. The experimental results demonstrate that the proposed method yields a satisfactory prediction accuracy for stock trends.

Index Terms—Classification, Prediction, Data Scrapping, Stock Trend.

I. INTRODUCTION

The purpose of this paper is to discuss the use of machine learning algorithms for classifying reddit posts in order to predict stock prices. This paper focuses on the development of a model that can accurately classify posts into positive and negative categories in order to predict the future trend of stocks. The goal of this project is to develop a model which can effectively analyze posts and accurately predict stock prices.

In recent years, the use of social media, particularly Reddit, has become increasingly popular for stock market analysis and prediction. Reddit has become an important source of data for stock market analysis due to its high volume of data and its ability to capture sentiment. There have been several studies on the use of Reddit for stock market analysis and prediction. These studies have shown that R data can be used to accurately predict stock prices.

However, in order to use Reddit data for stock market prediction, it is necessary to classify the reddit post into positive, negative, and neutral categories. This is where machine learning algorithms come in. Machine learning algorithms are useful for classifying posts into the correct categories. This can be done by feeding the posts into a machine learning algorithm which is trained to categorize them.

The goal of this project is to develop a model that can accurately classify posts into the correct categories so that we can predict whether a stock will go high or low only. The model will use machine learning algorithms to classify the tweets into the correct categories. The model will then use the classified reddit posts or comments to predict stock prices.

The project will involve the development of a model that can accurately classify reddit posts into the correct categories. The model will be trained using a dataset of tweets and stock prices. The model will then be tested to evaluate its accuracy. The results of the tests will be used to determine the effectiveness of the model for stock market prediction. The paper will discuss the development of the model, the evaluation of its accuracy, and the results of the tests. The report will also discuss the implications of the model for stock market prediction.

II. DATASET

In this project, we used two main datasets:

1) The Reddit data set utilized in this paper includes publicly available posts on Apple, Inc. collected from Reddit developers between June 1, 2024 and November 14, 2024. The search term for reddit posts contains the following words: ”Apple”, ”Tim Cook”, ”@Apple”, ”#Apple”, ”$AAPL” and ”iPhone”. The idea for this is that we want to capture a broader picture of sentiment in order to acquire a more overall attitude about the company rather than just posts specifically linked to Apple stock. The data includes the Date, Title, Text, comment,score for every Post during that period.

2) The Apple stock data is obtained from Yahoo Finance, and contains the date, opening, high, low, closing price and volume.

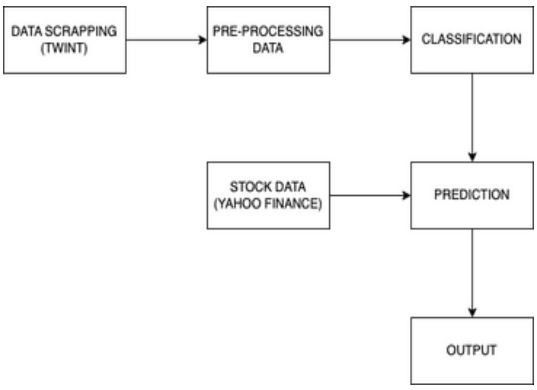
Aggregated values and sample data from Reddit

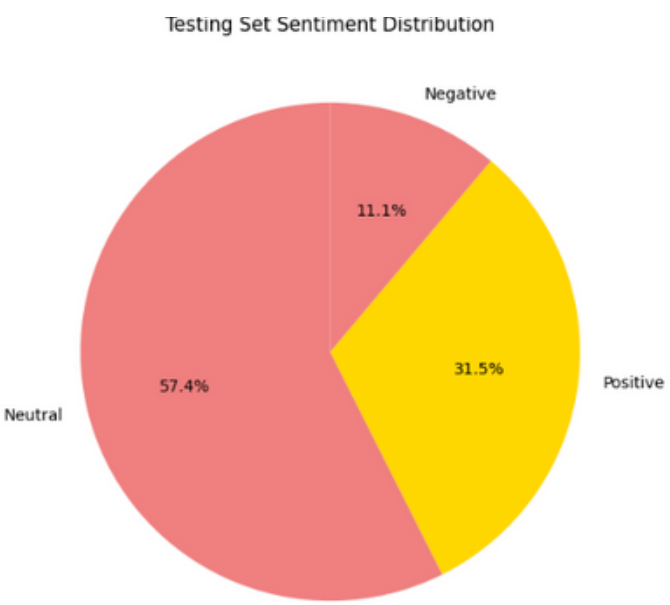
Date	Title	text
2024-06-01 15:38:02	AccessWallST	Apple Stock Sinks As Market Gains

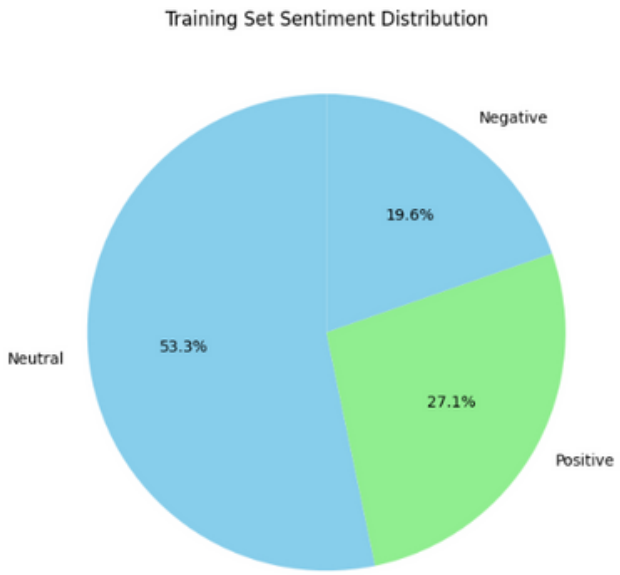
Aggregated values and sample data from Apple

date	opening	high	low	closing	volume
2024-06-01	20.40	21.00	19.92	20.67	174879000

III. PROPOSED MODEL







Data Scraping Procedure for Reddit Apple Stock Data

To collect data from Reddit related to Apple stock discussions, we used the Reddit API and the Python library PRAW (Python Reddit API Wrapper). The steps are outlined as follows:

1. Setting up Reddit API Credentials:
   - We registered a new application on the Reddit Developer Portal to obtain the Client ID, Client Secret, and User Agent needed for authentication.
2. Establishing Connection:
   - Using the PRAW library, we authenticated to Reddit's API with the credentials.
   - We configured the PRAW client to access subreddits such as r/stocks, r/investing, and r/Apple.
3. Defining Search Parameters:
   - We focused on keywords like "Apple," "$AAPL," and "Tim Cook" to retrieve posts discussing Apple stock.
   - Posts were collected between June 1, 2024, and Nov, 2024, as specified in the project's scope.
4. Data Collection Process:
   - For each relevant post, we extracted attributes such as:
     - Title and Text: The main content of the post.
     - Author: Username of the post creator.
     - Timestamp: Time the post was created.
     - Subreddit: The subreddit where the post was published.
     - Upvotes and Comments: Metrics to gauge post engagement.
   - Posts were then saved to a CSV file for further analysis.

Challenges Faced

1. API Rate Limits:
   - Reddit's API imposes strict rate limits on the number of requests per minute, causing delays during large-scale data collection.
   - Solution: We implemented efficient batching and used sleep intervals to avoid hitting the limit.
2. Irrelevant Data:
   - Many posts retrieved using broader keywords like "Apple" included non-relevant content (e.g., discussions about the fruit or unrelated products).
   - Solution: We refined our search terms (e.g., "$AAPL," "Tim Cook") and filtered out posts using text-processing techniques.
3. Handling Missing or Incomplete Data:
   - Some posts lacked attributes like comments or upvotes, making it challenging to analyze engagement accurately.
   - Solution: We handled missing values by imputing or excluding them during data preprocessing.
4. Time Frame Limitations:
   - Reddit's API does not allow direct retrieval of data beyond a certain time range.
   - Solution: We used a combination of historical pushshift.io data dumps and Reddit API queries to overcome this limitation.
5. Text Noise:
   - Post content often contained emojis, URLs, and unnecessary characters that interfered with sentiment analysis.
   - Solution: We applied text cleaning techniques to preprocess the data.

Conclusion

The Reddit API and PRAW library proved to be effective tools for collecting structured data. Despite encountering challenges like rate limits, irrelevant data, and noise in textual content, we implemented solutions that ensured the integrity and relevance of the collected data.

MODEL PREDICTIONS

We utilized multiple algorithms to build this model, including Naive Bayes, Logistic Regression, AdaBoost, Decision Trees, and Random Forest. Each algorithm brings unique strengths and limitations, which allowed us to leverage their diverse capabilities. By combining these methods, we developed a robust model that performs effectively on our dataset. Using various algorithms also enabled us to analyze different aspects of the data, uncover patterns, and enhance the model's accuracy through comparative evaluation and insights.

1)Naive Bayes for Reddit Post Classification

We utilized the Naive Bayes algorithm to classify Reddit posts. This algorithm is simple, efficient, and particularly well-suited for text classification tasks. It assumes that the occurrence of each word in a post is independent of the occurrence of other words, which simplifies calculations.

For our Reddit post classification, we adopted a bag-of-words approach. This method counts the frequency of words in the text and uses these counts to calculate the probability of a given class (positive or negative) based on the words present.

Using this probability, the model classified each post into one of two classes: positive or negative. The Naive Bayes model demonstrated solid performance, achieving an accuracy of 76% in classifying Reddit posts accurately.

```
Naive Bayes Accuracy: 0.7592592592592593
              precision    recall  f1-score   support

    Negative       1.00      0.50      0.67         6
     Neutral       0.70      1.00      0.83        31
    Positive       1.00      0.41      0.58        17

    accuracy                           0.76        54
   macro avg       0.90      0.64      0.69        54
weighted avg       0.83      0.76      0.73        54
```

2) Logistic Regression for Sentiment Analysis of Reddit Data

Logistic Regression was employed to classify sentiments (Negative, Neutral, Positive) in the Reddit dataset. The model achieved an overall accuracy of 79.63% on the test set.

- Class-wise Performance:
  - Negative Sentiments: Precision of 1.00 and recall of 0.67 indicate perfect identification of true negatives but limited coverage.
  - Neutral Sentiments: High precision (0.75) and excellent recall (0.97) showcase the model's strong ability to detect neutral sentiments.
  - Positive Sentiments: Precision of 0.90 and recall of 0.53 suggest it identifies most positives correctly but misses some.
- Weighted Metrics:
  - Precision (0.82): Indicates the proportion of correctly identified sentiments out of all predicted ones.
  - Recall (0.80): Reflects the proportion of true sentiments captured by the model.
  - F1-Score (0.78): Balances precision and recall, providing a comprehensive performance measure.

The results demonstrate that Logistic Regression effectively captures Neutral sentiments while showing room for improvement in handling Positive and Negative classes.

3)Random Forest for Reddit Post Sentiment Analysis

The Random Forest algorithm was applied to classify Reddit posts into three sentiment categories: Negative, Neutral, and Positive. This ensemble method combines multiple decision trees to improve accuracy and reduce overfitting, making it well-suited for complex datasets.

The model achieved an accuracy of 79.63% on the test set. It excelled in identifying Negative posts, with a precision of 1.00 and recall of 0.83. For Neutral posts, the recall was perfect at 1.00, capturing all neutral instances, though with a lower precision of 0.74, indicating some misclassification. However, for Positive posts, while precision remained high at 1.00, the recall dropped to 0.41, showing difficulty in identifying all positive sentiments.

The macro average precision, recall, and F1-score were 0.91, 0.75, and 0.78, respectively, while the weighted F1-score was 0.77, reflecting strong overall performance. These results highlight the strengths of Random Forest in handling diverse sentiment categories, though improvements are needed to enhance the recall for positive sentiments. The algorithm’s robust structure made it effective in analyzing sentiment patterns across Reddit posts.

4) Decision Tree for Reddit Post Sentiment Analysis

The Decision Tree algorithm was utilized to classify Reddit posts into three sentiment categories: Negative, Neutral, and Positive. This method splits the data into subsets based on feature values, creating a tree-like structure to make predictions.

The model achieved an accuracy of 77.78% on the test set. It performed consistently across sentiment classes:

For Negative posts, both precision and recall were 0.83, indicating reliable identification of negative posts.

Neutral posts were classified with a precision of 0.83 and recall of 0.81, showcasing the model's strong ability to detect neutral sentiments.

For Positive posts, precision was 0.67, and recall was 0.71, suggesting some challenges in accurately capturing all positive posts.

The macro and weighted averages for precision, recall, and F1-score were all 0.78, reflecting balanced performance across the three sentiment categories.

While Decision Tree models are interpretable and effective for smaller datasets, their susceptibility to overfitting might have impacted performance on positive sentiment classification. Overall, the algorithm provided competitive results and valuable insights into sentiment patterns in Reddit posts.

Comparing Model Accuracy's implemented	
Model	Accuracy
NaiveBayes	76%
RandomForest	79.63%
DecisionTree	77.78%
AdaBoost	76%
LogisticRegression	79.63%

We can notice that the class imbalance was also main reason for relatively lesser accuracy's.

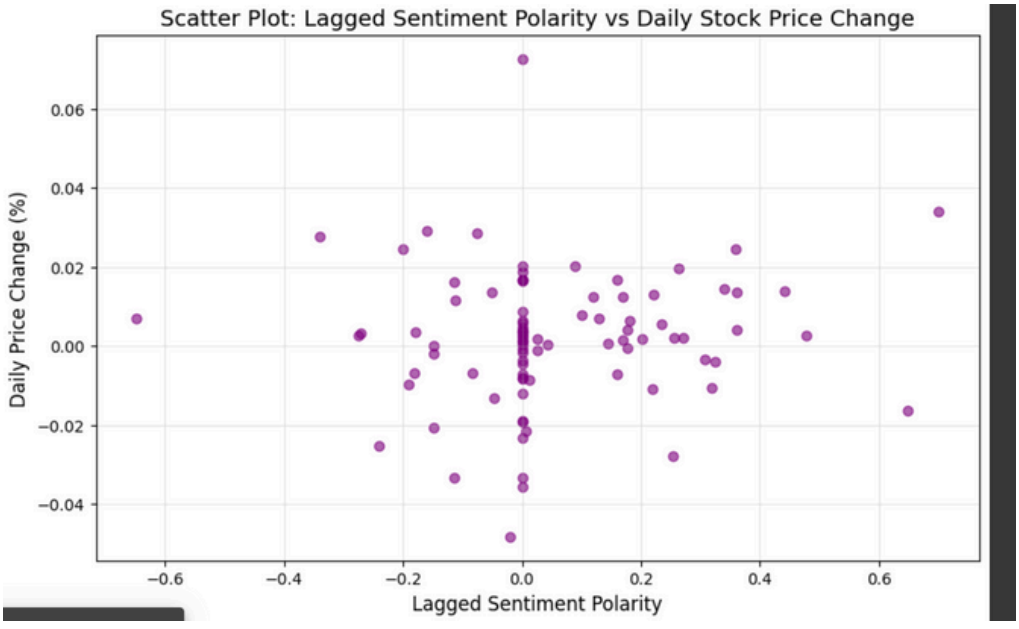


Fig. 3. Co-relation between Lagged sentiments and Daily Price Change

Conclusion on Correlation Between Sentiment and Stock Price Change

To explore the relationship between sentiment and stock price movement, we calculated the average sentiment for each day and used it to predict stock price changes. Our hypothesis suggested that as average sentiment increases, stock prices would follow a similar upward trend. However, the correlation coefficient between average sentiment and stock price change was approximately 0.3, indicating a weak positive correlation. This suggests that while sentiment does not have a strong influence on stock price changes, it still has some level of impact that should not be entirely disregarded. Therefore, sentiment analysis can provide valuable insights, but other factors likely play a more significant role in driving stock price fluctuations.

Conclusion and Acknowledgment

Finally, our analysis has some limitations, as it does not account for several important factors. One key limitation is the lack of representation of the broader public sentiment, as the dataset only includes posts from Reddit users who speak English. This restricted sample may not fully reflect global sentiment, and future research could explore how sentiment across different languages and regions influences stock prices. Additionally, while our model identified a weak correlation between sentiment and stock price changes, it did not account for many external factors that could impact investment decisions. We speculate that investors' moods, influenced by external factors, might have a more significant role, which could lead to a stronger correlation if studied further. Although our model cannot predict the exact percentage change in stock prices, it successfully indicates whether a stock's price is likely to rise or fall based on sentiment. This serves as a foundation for further research into the connection between public sentiment and stock market trends.

Acknowledgment

We would like to acknowledge the Reddit community for providing valuable insights through their posts, and thank the developers of the PRAW library for making the data collection process possible. This project would not have been feasible without their contributions.