Predicting Next-Day Stock Closing Prices using Machine Learning and News Sentiment

Sara Mezuri[1] **and Alireza Golkarieh**[2]

[1]*saramezuri@oakland.edu*
[2]*golkarieh@oakland.edu*

This manuscript was compiled on November 8, 2024

Abstract

Problem Statement: The project aims to predict the next day's stock closeing price using a combination of historical stock data and sentiment analysis on financial news articles.
Motivation Accurately predicting stock prices is crucial for investors and traders, as it enhances decision-making and profitability. By integrating sentiment analysis, the project aims to provide a more comprehensive model, offering deeper insights into how news events influence market trends.

1. Data availability

The project utilizes two primary data sources: historical stock data and financial news articles.

Stock Data: We are collecting historical data for key attributes, such as open, close, high, and low prices, for stocks in the $S\&P500$ and Nasdaq-100 indices. The dataset spans from January 1, 2000, to October 1, 2024, providing over two decades of comprehensive market information.

News Data: Gathering news articles covering the same stocks over this extensive time frame is more complex due to limitations in archival access and the constraints of current API solutions. While paid news APIs offer historical data, they often limit access to shorter time frames, making it challenging to retrieve news data from 2000 onwards. To bridge this gap, we are exploring the use of Bloomberg's Terminal and its blpapi Python API for accessing older archives, which may enable us to gather richer, long-term data on news sentiment for these stocks.

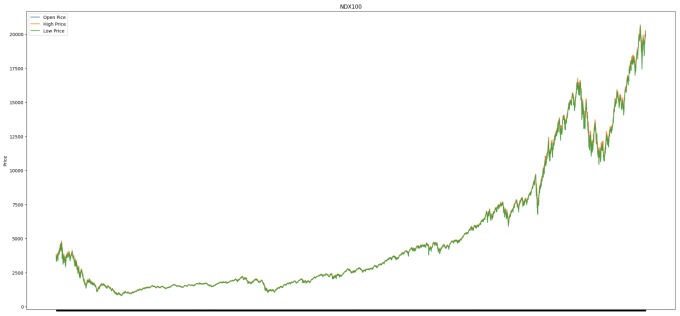

Figure 1. The Nasdaq-100 Stock Chart

2. Data Preparation

2.1. Data Structures and Formats

Stock Data: Stock data is structured as numerical, tabular data, ideal for time series analysis.

News Data: News articles are stored in text format and will require transformation into sentiment scores for compatibility with our models.

2.2. Data Cleaning and Transformation

Stock Data: Fortunately, the stock data shows no missing values, simplifying the preprocessing steps. Since this is a regression task, issues like imbalanced data do not apply. However, to ensure the suitability of our dataset for time series modeling, we may perform transformations such as log scaling or differencing to address potential trends and seasonality.

News Data: Cleaning and structuring news articles for analysis is more complex. Text data must be processed, tokenized, and converted into sentiment-based features. Using NLP techniques, we will apply sentiment analysis to generate numerical scores from the textual data.

2.3. Challenges in Data Collection and Preparation

Despite a solid data plan, obtaining high-quality news data for sentiment analysis presents significant challenges:

- **API Limitations**: Free APIs generally restrict data retrieval for historical archives, limiting access to recent years or imposing high costs for extensive historical data. Overcoming this will require creative solutions, such as the Bloomberg Terminal, which may still impose constraints on data availability.
- **Web Scraping Obstacles**: For a more comprehensive dataset, we considered web scraping; however, many financial news sites lack full archives from the early 2000s. This limits our ability to analyze the influence of older events on stock prices, potentially impacting the predictive power of our sentiment models.
- **Back-up Plan**: Given these challenges, we are developing two parallel models to accommodate possible data limitations. The first will rely solely on stock data, predicting price movements without sentiment analysis. The second, if news data access improves, will incorporate sentiment analysis but may be limited to recent years. This approach will still provide valuable insights into how recent news sentiment influences stock performance, though it may not capture long-term trends as thoroughly.

3. Design Specifications

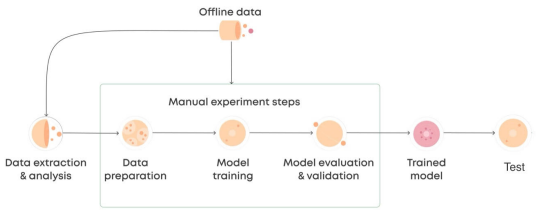

Figure 2. ML Pipeline Architecture

4. Approach

4.1. Machine Learning Approach

The project employs a supervised learning approach, using regression analysis to predict the continuous variable of next-day stock closeing prices. Additionally, sentiment analysis will be applied to news data to capture qualitative insights, transforming textual information into numerical sentiment scores that can be incorporated into the regression model.

4.2. Models and Algorithms

4.2.1. Time Series Baseline Models

As a starting point, we will apply traditional time series models to establish a baseline, considering models like ARIMA, SARIMA, and GARCH. The choice among these models will depend on the nature of the stock data, specifically in terms of seasonality, volatility clustering, and other time-dependent patterns. These models provide a strong foundation for understanding the intrinsic trends and seasonality in stock prices.

4.2.2. LSTM (Long Short-Term Memory Networks)

Given the time-dependent nature of stock price prediction, LSTMs, a type of recurrent neural network, are well-suited for our problem. We are using a 60-day moving window for training, capturing recent price patterns effectively. LSTMs are advantageous because they can handle long-term dependencies, making them ideal for forecasting future stock prices based on historical data recorded consistently at the end of each trading day.

4.2.3. Random Forest

For a more robust and ensemble-based approach, we are also using a Random Forest model with a baseline of 100 estimators. Random Forests are valuable for capturing complex relationships within the data, including non-linear patterns, which could contribute to more accurate predictions. While typically used for non-time-series data, Random Forests offer a comparative model that will help gauge the impact of sentiment analysis alongside stock data.

4.3. Justification for Chosen Approach

This approach is tailored to the unique demands of time-dependent regression, as stock prices follow a temporal pattern influenced by market events and news sentiment. Time series models are foundational, allowing us to capture trends and seasonal effects inherent in financial data. LSTM is particularly suited to our needs because it preserves temporal order and learns from sequential patterns, while the Random Forest provides an ensemble approach to understand feature importance, particularly when integrating sentiment data. By combining these models, we aim to achieve a more comprehensive prediction framework that captures both quantitative price trends and qualitative sentiment signals.

5. Technical Design

5.1. ML Libraries and Frameworks

Pandas for data manipulation, **NumPy** for numerical operations, **Scikit-learn** for Random Forest modeling, **Keras** for building and training the LSTM model, and **Matplotlib** for data visualization.

5.2. Development and Experimentation Tools

Jupyter Notebook and **Google Colab** are used for interactive coding and running experiments, especially for resource-intensive tasks. **GitHub** hosts all project files, providing version control and collaborative management.

5.3. Future Considerations

MLflow may be added for experiment tracking and model comparison as the project evolves.

6. Experimental Setup

6.1. Research Question

This project seeks to answer: Which model provides the most accurate prediction of next-day stock closeing prices by combining historical stock data with news sentiment analysis?

6.2. Comparative Approach

We will compare multiple models to determine which yields the most accurate predictions. These include baseline time series models (ARIMA, SARIMA, GARCH), LSTM for sequential data, and Random Forest for incorporating both stock and sentiment data. The comparison will highlight each model's predictive capability and suitability for the time-dependent regression task.

6.3. Evaluation Metrics

MSE (Mean Squared Error): Measures the average squared difference between predicted and actual values, helping assess model accuracy. Euclidean Distance Loss: This loss function will provide an additional measure of prediction error.

Augmented Dickey-Fuller (ADF) Test: Used to check stationarity in time series data, crucial for validating assumptions in certain time series models. Q-Q Plot: Evaluates the normality of residuals, particularly for time series model validation.

References

[1] Jason Brownlee. *Deep Learning for Time Series Forecasting*. Machine Learning Mastery, 2020. https://machinelearningmastery.com/deep-learning-for-time-series-forecasting/

[2] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, Inc., 2019. https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/

[3] Bing Liu. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 2012. https://www.cs.uic.edu/~liub/FBS/Sentiment-Analysis-and-Opinion-Mining.pdf

[4] Various Authors. "Financial News Predicts Stock Market Volatility Better Than Close Price." *Journal of Financial Economics*, 2019.

[5] Various Authors. "An Empirical Analysis of Stock Price Prediction Using ARIMA and LSTM." *Journal of Economic Dynamics and Control*, 2021.

[6] Various Authors. "Sentiment Analysis in Financial Texts." *Finance and Data Science*, 2022.

[7] Medium Author. "Sentiment Analysis with Python - A Beginner's Guide." *Medium Article*, 2023. https://medium.com/@eleanor.watson/a-beginners-guide-to-performing-sentiment-analysis-on-text-with-python-3ce80dcac22e

[8] Jason Brownlee. "Introduction to Time Series Forecasting with Python." *Machine Learning Mastery Blog*, 2023. https://machinelearningmastery.com/time-series-forecasting/

[9] Bloomberg. "Bloomberg Python API: blpapi." 2023. https://bloomberg.github.io/blpapi-python/

[10] Yahoo Finance. "Yahoo Finance API Documentation." 2023. https://www.yahoofinanceapi.com/

[11] News API. "News API Documentation." 2023. https://newsapi.org/docs/