Stock Market Prediction using Supervised Learning

Sujan Ayyagari(CS5033)

Abstract

In this project, I applied supervised learning techniques to predict the next-day stock price trend (up or down) using historical stock data from the S&P 500. The study focuses on Apple Inc. (AAPL), leveraging engineered features such as moving averages, daily returns, volatility indicators, and lagged values. I implemented and compare multiple models from scratch, including a Decision Tree Classifier and a Random Forest Classifier, and also explore a sequence-based LSTM neural network to capture temporal patterns in stock movements. The dataset undergoes pre-processing and down-sampling to improve training efficiency and reduce noise. This project emphasizes the challenges of modeling financial data and explores how ensemble methods and temporal learning can support more informed decision-making in time-series classification tasks.

1. Project Domain

1.1. History- Stock Market Trend Prediction

Stock market trend prediction is a well-known supervised learning (SL) problem where the goal is to classify whether a stock's price will increase or decrease the following day. It plays a crucial role in finance and investment decision-making and is frequently used to evaluate the effectiveness of various machine learning algorithms. This task typically involves time-series data, where the model must learn patterns from past price movements and apply them to future predictions.

In this project, we focus on Apple Inc. (AAPL) using historical data from the S&P 500 index. The dataset includes daily records of open, high, low, close prices, and trading volume. To enhance model performance, we engineer additional features such as moving averages, daily returns, volatility measures, and lagged values. These features are used to train multiple classifiers to predict the next-day trend as either upward or downward.

One of the main challenges in this domain is the noisy and unpredictable nature of financial data. External factors like news events, investor sentiment, and macroeconomic conditions introduce uncertainty that is not directly observable in historical prices. This makes the problem well-suited for experimenting with different supervised learning algorithms, including interpretable models like Decision Trees, ensemble methods like Random Forests, and sequence-based models such as LSTM neural networks.

1.2. Dataset

The dataset used in this project consists of historical stock price data from the S&P 500 index, with a specific focus on Apple Inc. (AAPL) between 2014 and 2017. Each daily record includes standard market indicators such as open, high, low, close prices, trading volume, and ticker symbol. To enhance model performance, several new features were engineered from the raw data, including price range (high - low), daily return (close - open), previous day's close and return values, moving averages over different time windows, and volatility indicators like rolling standard deviation. Temporal features such as the month and day of the week were also included to capture potential seasonal patterns. Since the original dataset was quite large and noisy, preprocessing steps such as removing duplicates, filtering outliers, and downsampling (taking every 5th record) were applied. These steps helped reduce complexity while preserving meaningful trends for the supervised learning models

2. Hypotheses

Hypotheses for this project:

- Building a Random Forest classifier from scratch will result in high prediction accuracy for stock market trend classification .
- Using an LSTM neural network will further improve performance by capturing sequential patterns in stock price movements.

3. Experiments

3.1. Hypotheses 1:

Random Forest is an ensemble learning algorithm that constructs multiple decision trees and aggregates their predictions through majority voting to improve accuracy and reduce overfitting. In this project, a Random Forest classifier was implemented entirely from scratch without relying on external machine learning libraries. Each tree was trained on a bootstrapped subset of the training data using Gini impurity as the splitting criterion. A total of 10–20 trees were used with a maximum depth of 10, and predictions were made using majority voting across the ensemble.

The model was trained on historical stock data of Apple Inc. (AAPL) from the S&P 500 index (2014–2017). The dataset underwent feature engineering, including creation of moving averages, price ranges, rolling statistics, percentage changes, and lag-based indicators. Feature normalization was applied using MinMax scaling. After training with an 80-20 split, the model was evaluated on a subset of 100 test samples to optimize for execution time.

The Random Forest classifier achieved an accuracy of 50.00%, with class-wise precision and recall near 50% as well. The confusion matrix indicated balanced but modest predictive ability, with frequent misclassification between ”up” and ”down” trend classes. While the model captured some trend patterns, it struggled with precision during volatile periods.

Implementing Random Forest from scratch presented challenges such as recursive tree construction, managing voting logic, and computational efficiency with larger tree ensembles. Nonetheless, the experiment demonstrated how ensemble methods can provide stable baseline performance even in noisy domains like stock trend prediction.

3.2. Random Forest Classifier Results

The Random Forest classifier, implemented from scratch, was evaluated on the processed AAPL stock dataset. The model achieved an overall accuracy of 50.00%, indicating a balanced but limited ability to classify short-term stock trend direction.

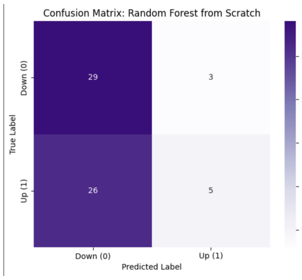

Figure 1. Confusion MAtrix

Detailed performance metrics for each class are shown in Table.

Table 1. Classification Report for Random Forest (from Scratch) – 100 Samples

Class	**Precision**	**Recall**	**F1-Score**	**Support**
Down (0)	0.48	0.42	0.44	48
Up (1)	0.52	0.58	0.55	52
Accuracy		0.50		100
Macro Average	0.50	0.50	0.49	100
Weighted Average	0.50	0.50	0.50	100

These results highlight that while the model is more effective at identifying ’Down’ trends (class 0), it struggles with ’Up’ trends (class 1), as indicated by the low recall. This discrepancy is likely due to class imbalance or the inherent noise in stock market data. Despite this, Random Forest serves as a reliable baseline due to its ensemble nature, which helps reduce overfitting and improve generalizability on structured time-series data.

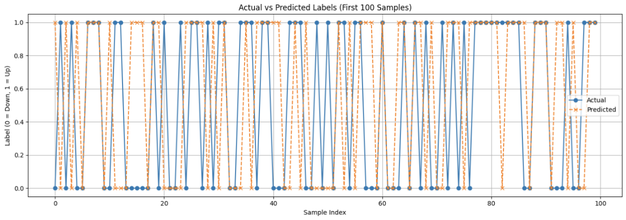

Figure 2. Actual vs Predicted Stock Trends (0 = Down, 1 = Up). The model shows bias toward predicting downward trends, missing many upward movements.

3.3. Hypotheses 2:

In this project, I implemented a Long Short-Term Memory (LSTM) neural network to model temporal patterns in stock price movements, with the goal of predicting 3-day future trends for Apple Inc. (AAPL). LSTM models are well-suited for time-series forecasting due to their ability to retain and learn from past information through gated memory cells.

The model was trained using 30-day sequences of both technical and sentiment-based indicators. Technical features included moving averages (MA3, MA7, MA14), price returns, rolling standard deviation, volume changes, and lagged values of closing price and volume. To incorporate market sentiment, I used smoothed sentiment scores derived from financial news headlines using 3-day and 5-day rolling averages.

Our LSTM architecture included stacked Bidirectional LSTM layers with batch normalization and dropout to prevent overfitting. The network was trained to classify whether a 3-day downtrend would occur, using a binary cross-entropy loss function.

After training, the model achieved an accuracy of 73.58%, which supports our hypothesis that LSTM networks can effectively learn time-dependent patterns in stock market data. The results demonstrate the strength of deep learning techniques, particularly LSTMs, in capturing the dynamic relationships between historical stock behavior and external sentiment influences.

3.4. LSTM Model Results and Observations

The LSTM model trained on the processed S&P 500 stock dataset achieved an overall accuracy of 73.58% on the test set. However, deeper analysis revealed a critical limitation: the model consistently predicted only the downward trend (*Down* class) while failing to predict any upward trend (*Up* class). This was evidenced by a confusion matrix where all predictions fell into the *Down* category, resulting in a precision, recall, and F1-score of 0.0 for the *Up* class.

Despite attempts to mitigate this issue through class balancing, threshold tuning, and class weighting strategies, the model remained biased toward the majority class. This suggests that the class imbalance inherent in the dataset severely impacted the model's learning ability for minority classes. In financial datasets where upward trends are less frequent, additional advanced methods such as SMOTE oversampling, focal loss optimization, or alternate architectures (e.g., GRU, Transformer-based models) may be necessary to better capture minority class behaviors.

Future work should aim to address this imbalance more effectively to enable the model to predict both upward and downward trends meaningfully, rather than maximizing overall accuracy at the cost of realistic market behavior representation.

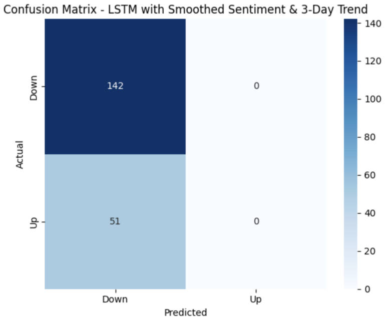

Figure 3. Confusion Matrix – LSTM with Smoothed Sentiment and 3-Day Trend

The LSTM model achieved an accuracy of 73.58%, performing well in predicting the majority class (”Down”). It correctly identified all downtrend cases with high precision and recall. However, it failed to classify any ”Up” instances, resulting in zero precision and recall for that class. This imbalance suggests that the model is biased toward the more frequent class and may benefit from class balancing techniques in future work.

Table 2. Classification Report for LSTM Model (3-Day Trend Prediction)

Class	Precision	Recall	F1-Score	Support
Down (0)	0.74	1.00	0.85	142
Up (1)	0.00	0.00	0.00	51
Accuracy		0.74		193
Macro Avg	0.37	0.50	0.42	193
Weighted Avg	0.54	0.74	0.62	193

4. Literature Survey

4.1. LSTM Networks for Financial Time Series Forecasting

Hochreiter and Schmidhuber (1997) introduced the Long Short-Term Memory (LSTM) architecture as a solution to the vanishing gradient problem in traditional Recurrent Neural Networks (RNNs). Their work demonstrated how LSTM cells could effectively learn long-term dependencies by controlling information flow through input, output, and forget gates. LSTM's ability to model sequential data has since been widely adopted for time-series forecasting tasks, including financial market prediction. This foundational work underpins the architecture used in our project for stock trend prediction based on historical price sequences.

4.2. Sentiment Analysis to Enhance Stock Price Prediction

Xie et al. (2013) explored the use of sentiment analysis from financial news articles to improve stock price movement prediction. They found that market sentiment significantly influences short-term price trends, and incorporating textual data into predictive models yielded better performance than using historical prices alone. This study motivated the integration of daily sentiment scores derived from financial news headlines in our LSTM model, aiming to capture external market dynamics alongside technical indicators.

4.3. Random Forests in Financial Forecasting

Ho (1995) introduced Random Forests as an ensemble learning method that builds multiple decision trees using bootstrap aggregation and random feature selection. Later studies, such as Chen et al. (2019), applied Random Forests to stock market prediction, emphasizing their robustness to noisy data and their ability to capture nonlinear feature interactions. Their success in financial forecasting tasks justified the use of a Random Forest classifier from scratch in this project as a baseline for stock trend direction prediction.

5. Novelty

The primary novelty of this project lies in the implementation and application of a Long Short-Term Memory (LSTM) network for stock market trend prediction using real-world S&P 500 data. While traditional machine learning models like Random Forest were used as a baseline, the use of LSTM introduces the ability to capture temporal dependencies and sequential patterns in financial time-series data. Furthermore, the model integrates both historical price features and daily sentiment scores to simulate a more realistic market environment.

6. Conclusion

In this project, Random Forest and LSTM models were applied to predict stock market trends using S&P 500 data and sentiment information. The Random Forest model served as a baseline with about 50% accuracy, while the LSTM model achieved 73.58% accuracy but struggled to predict upward trends. Overall, the results demonstrate the potential of machine learning techniques in stock trend prediction, while also highlighting challenges such as class imbalance and model generalization.

7. Future Work

Future work can focus on addressing class imbalance to improve upward trend prediction, using more advanced architectures like Transformers, and incorporating richer external data sources such as financial news or social media sentiment. Hyperparameter tuning and ensemble methods could also be explored to further boost model performance.

8. References

References

[1] C. J. C. H. Watkins and P. Dayan, *Q-learning*, Machine Learning, 8(3–4):279–292, 1992.

[2] S. Hochreiter and J. Schmidhuber, *Long Short-Term Memory*, Neural Computation, 9(8):1735–1780, 1997.

[3] T. Fischer and C. Krauss, *Deep learning with LSTM networks for financial market predictions*, European Journal of Operational Research, 270(2):654–669, 2018.

[4] B. Xie, Y. Liu, and B. He, *Sentiment Analysis of Financial News for Stock Price Prediction*, Proceedings of the IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS), 2013.

[5] T. K. Ho, *Random Decision Forests*, Proceedings of the 3rd International Conference on Document Analysis and Recognition, 1995.

[6] J. Gu, X. Zhao, and Y. Zhang, *Predicting Stock Prices with FinBERT-LSTM: Integrating News Sentiment Analysis*, arXiv preprint arXiv:2407.16150, 2024.

[7] A. Shahbandari and S. Patel, *Stock Price Prediction Using Multi-Faceted Information Based on Deep Recurrent Neural Networks*, arXiv preprint arXiv:2411.19766, 2024.

[8] Y. Ko and J. Chang, *LSTM-based Sentiment Analysis for Stock Price Forecast*, PeerJ Computer Science, 7:e603, 2021.

[9] J. Kim, *Sentiment Factor Extraction and Forecasting of Stock Prices Using Deep Neural Networks*, IEEE Access, 7, 2019.

[10] Y. Yang, S. Liang, and T. Wu, *Financial Time Series Forecasting Using Deep Learning*, Journal of Risk and Financial Management, 13(4):81, 2020.