Comparative Analysis of Leading Machine Learning Models for Stock Price Predictive Analytics

Suprad Suresh Parashar
Arizona State University
sparash8@asu.edu

Pushpak Jaju
Arizona State University
pjaju1@asu.edu

Sanjai T S
Arizona State University
strichin@asu.edu

Sanjith P K
Arizona State University
skalveer@asu.edu

Venkat Nikhil Mangipudi
Arizona State University
vmangip1@asu.edu

Abstract

In an era characterized by the rapid evolution and inherent complexities of financial markets, the ability to make accurate predictions about stock price movements holds paramount importance for investors and traders. In this project proposal, we delineate a comprehensive research endeavor aimed at conducting an exhaustive comparative analysis of the foremost machine learning models in the context of stock price predictive analytics.Our primary goal is to discern and highlight the most efficacious machine learning algorithms and methodologies for enhancing the precision of stock price forecasts. To achieve this, we will embark on a multifaceted exploration, spanning data collection, preprocessing, model selection, training, validation, ensemble method investigation, feature importance analysis, and a meticulous comparative assessment.

1 Introduction

The endeavor to predict stock market movements is a perennial task that has attracted the intrigue and efforts of researchers and practitioners alike due to its high complexity and substantial financial implications. The dynamic, non-linear, and volatile nature of financial markets makes them a challenging domain yet a lucrative frontier for the application of machine learning (ML) algorithms. Predictive analytics in stock market scenarios aims to foresee price movements by analyzing historical data, thereby aiding in informed decision-making which is crucial for both individual and institutional investors.

The objective of this comparative analysis is to discern the efficacy and robustness of various machine learning models as expounded in five seminal papers. Each of these papers introduces unique methodologies employing a range of ML models such as Artifical Neural Networks (ANN), Long Short-Term Memory (LSTM), Random Forest (RF), Autoregressive Integrated Moving Average (ARIMA), and Ensemble Support Vector Machine (ESVM) to predict stock prices in different markets and contexts. The first paper employs Artificial Neural Networks to forecast the Indian stock market using tick data, while the second utilizes LSTM, a deep learning approach, for predicting prices in the Indian share market. In the third paper, Random Forest is applied for selecting macroeconomic variables crucial in stock market forecasting. The fourth paper offers an insight into the ARIMA model's application in predicting banking stock market data, and the fifth introduces an Ensemble Support Vector Machine strategy for enhanced efficiency in stock market prediction.

Motivated by the potential to unlock new insights into predictive accuracy and to furnish a holistic understanding of the landscape, this analysis seeks to juxtapose these models on various fronts

Preprint. Under review.

including accuracy, computational efficiency, and ease of implementation. The selected papers represent a spectrum of approaches and encapsulate the ongoing evolution of predictive analytics in the realm of financial markets.

The expected outcome of this analytical endeavor is to provide a consolidated perspective on the strengths and limitations of the discussed ML models, thus guiding future research directions and practical applications in stock market predictive analytics. Through a meticulous comparison, we aim to unveil nuanced insights that could foster the development of more robust and accurate predictive models, thereby contributing to the overarching goal of enhancing the financial decision-making process.

2 Related Work

Stock price prediction, a cardinal segment of financial analytics, remains an enduring challenge due to the multifarious factors affecting stock markets. The significance of stock price prediction isn't merely academic but also holds paramount importance in practical applications like algorithmic trading, portfolio optimization, and risk management, among others.

Dharmaraja Selvamuthu et al. explored the application of ANN on tick data to forecast the Indian stock market. Their research underscores the increasing reliance on ANNs, especially given the high-frequency nature of tick data. Recognizing the potential of ANN to capture complex non-linear relationships, the authors employed it to decipher patterns from the flux of stock prices, asserting its superiority in certain scenarios over traditional time series models. Challenges, however, remain in determining the optimal architecture, as well as avoiding overfitting, especially given the noisy nature of tick data.

Building upon this neural network theme, Ghosh et al. venture into a more specialized domain with their paper "Stock price prediction using LSTM on Indian Share Market." They examine the capabilities of LSTM networks, a sophisticated form of deep learning, in stock price prediction. Their work illuminates LSTM's prowess in capturing long-term dependencies in time-series data, a critical aspect in the volatile Indian stock market. This study, therefore, complements the earlier findings and further expands our understanding of deep learning in stock market analysis.

Taking a different avenue, Isaac Kofi Nti, Adebayo Felix Adekoya, and B. Weyori in two separate studies, delved into the ensemble methods. Firstly, they unearthed the efficacy of Random Forest in feature selection, specifically focusing on macroeconomic variables. Their findings accentuate the importance of judicious feature selection in enhancing prediction accuracy, highlighting the utility of macroeconomic variables in capturing broader economic shifts that impact stock prices. In another investigation, the same authors explored the prowess of ensemble support vector machines (SVM) in stock market prediction. By leveraging the ensemble approach, they combined multiple SVMs to boost accuracy, thereby harnessing the power of collective intelligence. However, the challenge of determining the optimal number of SVMs and their individual parameters persists.

Lastly, M. Almasarweh and S. A. Wadi revisited the ARIMA model, specifically tailoring it for banking stock market data. Their research reaffirms ARIMA's position as a mainstay in time series forecasting. By delineating its application in the banking sector, they illustrate ARIMA's versatility, proving its mettle in a sector known for its susceptibility to external macroeconomic shocks. Nonetheless, the model's linear nature is a limitation, as it might struggle to capture abrupt and non-linear market shifts.

In summation, while numerous methods, ranging from conventional to contemporary, have been devised for stock price prediction, each comes with its set of advantages and challenges. The dynamic nature of stock markets, influenced by a plethora of factors, mandates continuous refinement of these models. Moreover, the choice of datasets, features, and algorithms remains pivotal, underpinning the efficacy of the predictive models.

3 Methodology

3.1 Data Acquisition

Our study utilizes the yfinance library to acquire historical market data from Yahoo Finance. This Python tool provides us with 20 years of stock price data for any tech companies listed on the NASDAQ. The data includes Open, High, Low, Close (OHLC) prices, volume, dividends, and stock splits, emphasizing OHLC data and trading volume for their significance in market trend analysis. We are considering two companies for our study, Amazon and Google

3.2 Implementation Libraries

Our predictive models are implemented using PyTorch, keras and Scikit-Learn. PyTorch's dynamic computation graphs facilitate the complex structuring of ANNs and LSTMs, while Scikit-Learn provides an efficient platform for traditional algorithms like SVM and Random Forest.

3.3 Training and Testing

We adopt a rolling window approach, mimicking real-world stock trading environments. This approach helps in continuously adapting to new data, reflecting market dynamics. To prevent overfitting, especially in complex models, we incorporate cross-validation strategies alongside traditional training and testing methods.

3.4 Predictive models

3.4.1 Multi-layer Perceptron

In the domain of stock market prediction, Multilayer Perceptron (MLP) neural networks emerge as a pivotal advancement. Characterized by multiple interconnected layers, MLPs excel in capturing intricate patterns and non-linear relationships within financial time series data. To ensure dataset integrity, missing values are replaced with column means, while MinMax scaling optimizes convergence during training. Adopting a sequence generation approach, 10-day close price sequences are utilized to discern temporal dependencies. The Keras-implemented MLP comprises an input layer with 10 nodes, two hidden layers (64 and 32 nodes), and an output layer for continuous stock price prediction. The model undergoes 50 epochs with Adam optimization, utilizing mean squared error as the loss function. Post-training, inverse transformation enhances interpretability. Performance metrics, including RMSE, MAE, MAPE, and R Squared, comprehensively evaluate predictive accuracy, enabling benchmarking against other models.

3.4.2 Long Short-Term Memory

LSTM is a recurrent neural network (RNN) architecture commonly used for time series prediction. In this context, LSTM was applied to forecast the closing prices of financial assets, specifically for Google (GOOG) and Amazon (AMZN) stocks. Prior to modeling, the dataset underwent Min-Max scaling, which normalized the closing prices to a range between 0 and 1, ensuring that the model could efficiently learn from the data. The LSTM model, built using Keras, featured a single layer with 50 units to capture temporal patterns in the scaled financial time series data. The dataset was divided into training, validation, and test sets with proportions of 70, 15, and 15, respectively, to enable effective model training and evaluation. The model underwent 50 epochs of training with a batch size of 32, utilizing the Adam optimizer to minimize mean squared error loss. This implementation illustrated the practical application of deep learning techniques, specifically neural networks, in analyzing and predicting financial market data, making it a valuable tool for forecasting stock prices and similar time-dependent datasets.

3.4.3 Random Forest Regression

Random Forest Regression is an ensemble learning method that operates by constructing a multitude of decision trees at training time. The final prediction is made based on the mean or average predictions of these individual trees. This method combines the simplicity of decision trees with flexibility, resulting in a robust model that can capture complex nonlinear patterns in data. Random

Forest is particularly advantageous for stock price forecasting due to its ability to handle large datasets with numerous input variables. Its robustness to overfitting, especially in cases where the number of features is much larger than the number of observations, makes it well-suited for financial markets data. Furthermore, its capacity to model nonlinear relationships is essential in capturing the complex dynamics of stock prices. Scikit-Learn's RandomForestRegressor class is used to create and train the Random Forest model. Its simplicity and flexibility allow for easy implementation and experimentation. A Train-Test split of 80 to 20 was employed for this model. The training was done on two seperate datsets of Google stock prices and Amazon stock prices

3.4.4 ARIMA

ARIMA is a class of statistical models for analyzing and forecasting time series data. It is a tool used to better understand or predict future points in a series by essentially combining three aspects: autoregression (AR): captures the influence of past values on current values, differencing (I): to make the time series stationary, which means that its statistical properties such as mean and variance do not change over time, and moving average (MA): models the error term as a combination of past error terms. ARIMA is a forecasting technique that models time series data by using its own lags and the lagged forecast errors. It's implemented in Python using the statsmodels library. To ensure the data meets ARIMA's stationarity requirements, preprocessing like differencing or log transformation is often necessary. The model is typically trained on 80 percent of the data, with the remaining 20 used for testing to assess its predictive accuracy. Stationarity checks and order selection are critical steps in configuring an effective ARIMA model for tasks such as stock price forecasting.

3.4.5 Support Vector Regression

Support Vector Regression (SVR) is a type of machine learning algorithm that falls under the category of Support Vector Machines (SVM). SVR applies the principles of SVM to regression problems. It works by mapping input data into a high-dimensional feature space and then finding a hyperplane that best fits the data in this new space. The key idea in SVR is to find a function that deviates from the actual observed outputs by a value no greater than a specified tolerance level. This approach helps in creating a model that is not overly sensitive to small fluctuations in training data, making it highly effective for tasks like stock price prediction. SVR is particularly valuable in situations where the relationship between the input variables and the target variable is complex and non-linear. Its ability to manage high-dimensional spaces and its robustness against overfitting, even in scenarios with limited data points, make it a reliable choice for modeling the intricate patterns often seen in financial market data. SVR's capacity to provide continuous output also makes it advantageous for forecasting continuous variables like stock prices, capturing the subtle nuances in market movements.

3.5 Comparative Analysis

Our analysis will not only compare the models based on accuracy but also on computational efficiency and robustness in varying market conditions. Acknowledging the importance of feature engineering in time series forecasting, we apply methods to select and construct features that best represent underlying market trends and patterns.We aim to assess and compare the performance of each predictive model using several key metrics, tailored to the nuances of time series data and financial forecasting:

3.5.1 Accuracy Metrics:

Mean Absolute Error (MAE): Measures the average magnitude of errors in predictions, regardless of their direction. It's useful for understanding the average error magnitude in stock price predictions.

Root Mean Squared Error (RMSE): Offers a sense of the magnitude of error, with a focus on penalizing larger errors. This is particularly relevant in stock forecasting where large errors can be more damaging.

Mean Absolute Percentage Error (MAPE): Provides a relative view of the prediction accuracy, which is crucial for comparing performance across different stock price scales.

R-Squared: Measures the proportion of variance in the dependent variable (e.g., stock prices) that is predictable from the independent variables.

4 Results

This section presents the comparative analysis of various machine learning models – MLP, LSTM, RF, ARIMA and SVR – in predicting Amazon's stock prices and Google's stock prices. The evaluation is based on four metrics: RMSE, MAE, MAPE and R Squared.

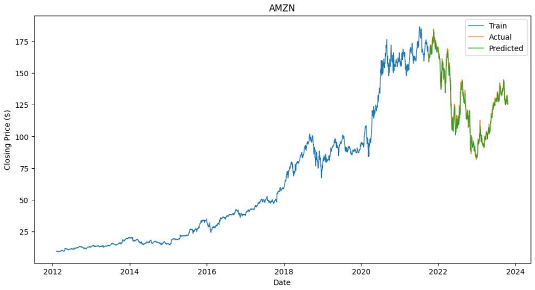

Figure 1: MLP Model prediction for Amazon

Fig 1 shows the Training, testing and predicted data for the Amazon stock prices by the Multi-layer Perceptron Model. The plot shows that the model is able to predict the stock prices really well.MLPs are a type of feedforward artificial neural network with multiple layers of perceptions. They are good at capturing non-linear relationships in data but may struggle with time-series data like stock prices due to their lack of temporal component handling.

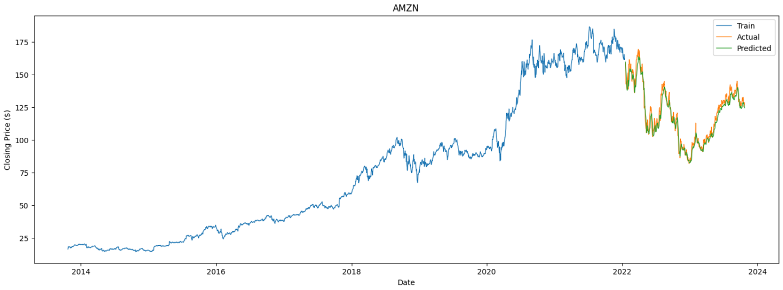

Figure 2: LSTM Model prediction for Amazon

Fig 2 shows the Training, testing and predicted data for the Amazon stock prices by the LSTM model. The plot shows that the model is able to predict the stock prices in a relatively good fashion. LSTMs are designed to handle sequential data and are capable of learning long-term dependencies. They are typically well-suited for time-series forecasting like stock prices due to their ability to remember past information. However, they can be computationally intensive and might require extensive hyperparameter tuning to achieve optimal results.

Fig 3 shows the Training, testing and predicted data for the Amazon stock prices by the RF model. The plot shows that the model is able to predict the stock prices in a good fashion but fails to predict the prices when there are sharp spike in prices. RF is an ensemble learning method that operates by constructing multiple decision trees during training. It's effective in handling various types of data, including non-linear relationships, and provides good generalization. RF's strong performance across all metrics in your study could be due to its ability to capture complex patterns without being too sensitive to noise in the data.

Fig 4 shows the Training, testing and predicted data for the Amazon stock prices by the SVR model. The plot shows that the model is able to predict the stock prices really well. SVR applies the principles of Support Vector Machines for regression problems. It's effective in finding a hyperplane

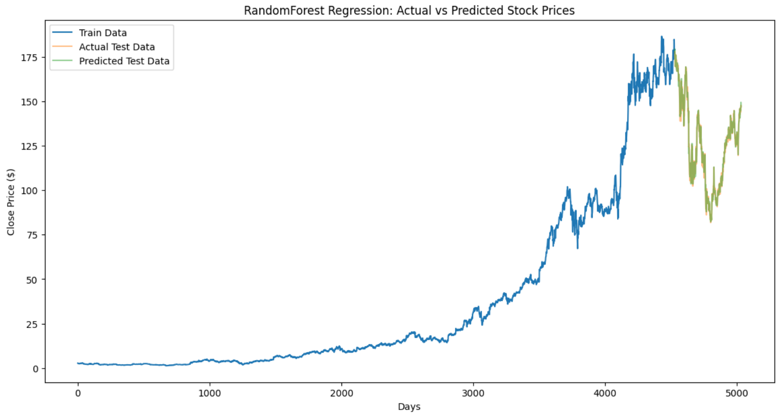

Figure 3: RF Model prediction for Amazon

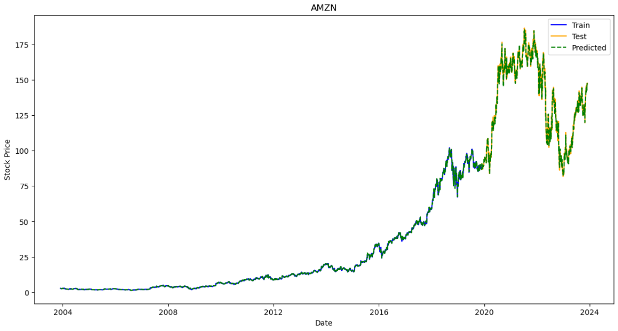

Figure 4: SVR Model prediction for Amazon

in a high-dimensional space to fit the data. The high performance of SVR suggests it was effective at capturing both the linear and non-linear patterns in the stock price data.

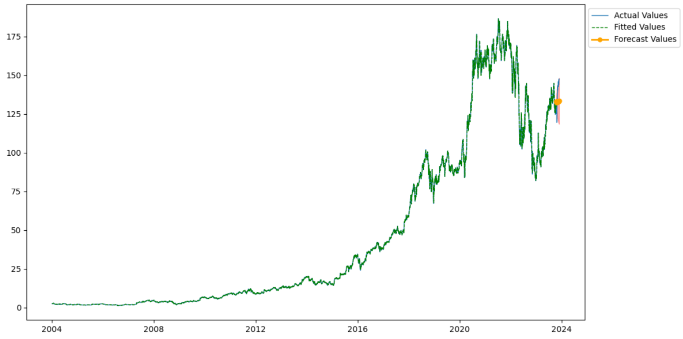

Figure 5: ARIMA Model prediction for Amazon

Fig 5 shows the Training, testing and predicted data for the Amazon stock prices by the ARIMA model. The plot shows that the model is not able to predict the stock prices as well as the other models. ARIMA models are commonly used for time-series forecasting, capable of capturing various temporal structures. They might not perform well when dealing with highly volatile data, as seen in stock markets.

Table 1 and Table 2 show the metrics for the predicted stock prices for Amazon and Google by the various models. The RF and SVR models consistently demonstrated high accuracy. For Amazon, RF and SVR had the lowest Root Mean Square Error (RMSE) at 1.705 and 1.668, respectively, and similar trends were observed for Google, with RF at 1.394 and SVR at 1.22. These models also

Table 1: Metrics for Amazon

Metrics	**MLP**	**LSTM**	**RF**	**ARIMA**	**SVR**
RMSE	3.608	4.731	1.705	9.312	1.668
MAE	2.731	3.693	1.316	8.357	1.303
MAPE	0.022	3.123	0.012	0.601	0.974
R Squared	0.981	0.947	0.994	0.524	0.996

Table 2: Metrics for Google

Metrics	**MLP**	**LSTM**	**RF**	**ARIMA**	**SVR**
RMSE	2.821	3.206	1.394	8.377	1.22
MAE	2.196	2.527	1.087	6.52	0.953
MAPE	0.019	2.230	0.009	0.503	0.877
R Squared	0.975	0.958	0.992	1.479	0.997

excelled in MAE, with RF and SVR showing the lowest values for both companies, indicating strong performance. In terms of MAPE, RF outperformed other models with extremely low values for both Amazon and Google, followed by SVR. The R-Squared values for these models were nearly perfect, reinforcing their effectiveness in fitting the stock price data. In contrast, the ARIMA model showed significantly higher RMSE and MAE values for both companies, indicating lower accuracy, with an unusually high R Squared value for Google's stock, indicating it may be less suitable for these particular forecasting tasks.

5 Milestone Review and Team Contributions

In the first phase of our research project on machine learning models for stock price prediction, we began with an extensive literature review and data collection for Amazon and Google stocks in the first two weeks. This foundational work was crucial for understanding the current landscape of machine learning in stock market prediction. In the third week, we focused on data preprocessing, which involved cleaning and standardizing the dataset to ensure its suitability for model training. This step was critical to remove biases and anomalies that could affect model performance.

From the fourth to the sixth week, our efforts were directed towards the implementation of the first set of machine learning models, using a consistent programming environment to ensure reliability and reproducibility of results. The following weeks, up to the eighth, were dedicated to implementing additional models and training them. This phase was vital for developing a diverse array of models to compare their effectiveness in predicting stock prices.

In the final week of this phase, we evaluated the performance of all the models implemented so far. This involved a thorough analysis of each model's accuracy and adaptability to market conditions using various metrics such as RMSE, MAE, MAPE, and R Squared. The insights gained from this evaluation were then compiled into a comprehensive report, detailing our findings and setting the groundwork for the next stages of the research, which include fine-tuning the models and a more exhaustive performance evaluation.

In this collaborative research effort, our team harnessed the collective expertise of diverse contributors, each playing a pivotal role in advancing the methodologies employed for our study. Sanjith brought forth a sophisticated understanding of the intricacies involved in SVR, contributing significantly to the refinement and implementation of this pivotal algorithm. Suprad specifically delved into the realm of MLP networks, tailoring the architectures to suit the nuances of our dataset. Nikhil spearheaded the application of Random Forests, enriching our predictive capabilities through the fusion of diverse decision trees. Sanjai brought forth a nuanced approach to capturing temporal dependencies within the data using LSTMs. Pushpak applied ARIMA models, contributing a crucial perspective that complements the ensemble of techniques applied in this study. Together, these collaborative efforts amalgamated diverse methodologies, thereby fortifying the robustness and comprehensiveness of our predictive model. This collective contribution reflects the interdisciplinary nature of our team, highlighting the synergy derived from the unique expertise each member brought to the table.

Our project adhered closely to the proposed milestones, demonstrating a high level of organizational efficiency and collaborative synergy within the team. Successfully achieving every milestone set forth in the project plan underscores the team's commitment to rigorous planning, effective communication, and adept execution, resulting in the timely accomplishment of our objectives.

6 Conclusion and Future Scope

The study provides a detailed comparative analysis of five prominent machine learning models - MLP, LSTM, RF, ARIMA, and SVR - in predicting stock prices for Amazon and Google. The evaluation, based on RMSE, MAE, MAPE, and R Squared metrics, reveals significant insights: MLP, RF and SVR models consistently show strong performance across all metrics for both companies, highlighting their ability to handle complex data patterns and resist noise in volatile markets. In contrast,LSTM model, while capable of capturing non-linear relationships and temporal patterns, do not perform as well as RF and SVR in terms of accuracy. The ARIMA model demonstrates variability in performance, suggesting its effectiveness may depend on specific data characteristics. This research underscores the importance of model selection in financial predictive analytics and highlights the need for further exploration into model optimization and feature engineering to enhance prediction accuracy.

Future research directions include exploring hybrid models that combine the strengths of individual models like RF and SVR, advanced feature engineering techniques, particularly for capturing market sentiment and macroeconomic factors, and incorporating real-time data streams to analyze model performance in a live market environment. Applying these models across different industries and sectors will help in understanding their generalizability. Further investigation into deep learning advancements, particularly fine-tuning LSTM networks and exploring newer architectures like Transformers, is also warranted. Integrating these models into algorithmic trading strategies and assessing their financial implications and risks would provide valuable insights for both academic research and practical applications in finance. Lastly, given the increasing use of AI in financial markets, future research should also consider the ethical and regulatory implications of deploying these models in real-world trading scenarios. This study lays a foundation for further exploration and innovation in the field of financial predictive analytics, aiming to bridge the gap between theoretical models and practical financial applications.

References

[1] D. Selvamuthu, V. Kumar, & A. Mishra, "Indian stock market prediction using artificial neural networks on tick data," *Financial Innovation*, vol. 5, no. 1, pp. 1–12, 2019.

[2] Ghosh, Achyut, et al, "Stock price prediction using LSTM on Indian Share Market," In *Proceedings of 32nd international conference*, vol. 63, 2019.

[3] K. O. Nti, A. Adekoya, & B. Weyori, "Random forest based feature selection of macroeconomic variables for stock market prediction," *American Journal of Applied Sciences*, vol. 16, no. 7, pp. 200–212, 2019.

[4] M. Almasarweh, & S. Alwadi, "ARIMA model in predicting banking stock market data," *Modern Applied Science*, vol. 12, no. 11, p. 309, 2018.

[5] I. K. Nti, A. F. Adekoya, & B. A. Weyori, "Efficient stock-market prediction using ensemble support vector machine," *Open Computer Science*, vol. 10, no. 1, pp. 153–163, 2020.