ISSN:0975-9646
Vanisha Rastogi et al / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 14 (3) , 2023, 37-41

Machine Learning Approaches to Stock Market Prediction: An Analysis of Support Vector Regression, Artificial Neural Networks, and Random Forest Regression

Vanisha Rastogi[1], Dr Shikha Singh[2], Vineet Singh[3], Dr Bramah Hazela[4]

[1,2,3,4] *Department of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University, Uttar Pradesh, Lucknow Campus, India.*

[1] vanisharastogi13@gmail.com

[2] ssingh@lko.amity.edu

[3] vsingh@lko.amity.edu

[4] bhazela@lko.amity.edu

***Abstract*— There has been immense research in prediction of stock prices and its movement. It's an interesting area of research for traders and economists alike. In this paper, we build four machine learning models which include Random Forest, XGBoost, Support Vector Regressor, and Artificial Neural Network. Technical indicators such as moving averages, RSI, MACD are calculated and used as input features for these models after feature selection. The models are then used to predict closing price for the next day based on the current set of features. The performance of these models has been evaluated based on their RMSE, MAE and MSE values. The findings reveal that XGBoost works the best among other algorithms, with the lowest RMSE value of 1.27. Random Forest algorithm has the second lowest score with only a small difference in its RMSE and MAE values compared to XGBoost. ANN and SVR had a slight difference in their performance. This study compares the performance of these machine learning algorithms which would help in carrying out further research in this area.**

Keywords—Stock Exchange, Data Science, Machine Learning, Stock Market Prediction, Stock Technical Indicators, SVR, ANN, Random Forest Regression.

I.INTRODUCTION

For both investors and financial professionals, predicting stock prices has always been a challenging and fascinating activity. The difficulties of this work have only increased with the financial world's constant change and rapid pace. As a result, several analysis methods—including technical and time series analysis—have become well-liked ways to forecast future stock prices.

Technical analysts typically use tools like moving averages, Bollinger Bands, and the Relative Strength Index (RSI) to look for trends and patterns in historical market data. This analysis is then utilized to predict the future trend of stock prices. Technical analysis has proven to be a useful technique for forecasting stock prices. Although the accuracy of those forecasts can be affected by the quality and quantity of previous data employed.

Machine learning has drawn attention as a promising method to forecast stock prices and market movements as the financial industry has developed. Large volumes of data can be used by machine learning to find links and patterns that humans would miss. It can predict future stock prices and swings thanks to this ability.

In order to predict the closing prices of the following day, this study focuses on employing technical indicators like the moving average, Bollinger Bands, and RSI as input characteristics for machine learning algorithms. On the basis of their RMSE, R2-Square, and MAE values, the study analyses the effectiveness of various machine learning models, including Random Forest regression, Support Vector regression, XGBoost, and Artificial Neural Network. Data for the analysis, which takes Google stocks into account, is gathered from the Yahoo Finance collection. The findings of the study offer insightful information on how well various machine learning models forecast stock prices which can be helpful to both investors and financial experts.

This paper starts with the introduction and is further divided into following sections. The Section 2 summarizes the earlier research on this subject. The methodologies and algorithms utilized in this paper are presented in detail in Section 3, along with an overview of the machine learning models and technical indicators used. Section 4 describes the results of the analysis on the dataset, including the performance metrics of the different models considered. Finally, in the last section, the study discusses the results

obtained, compares them with other methods, and proposes future work.

II. RELATED WORK

In their study, Dinesh et al. [1] investigate the usage of machine learning techniques to lessen the lag time of trading signals produced by the moving average crossover strategy. Moving averages are used to spot trends and produce trading signals, but as they are lagging indicators, they don't indicate a trend until there has been a major change in price. The trading signal's latency can be decreased by using machine learning techniques. The difficulty of effectively anticipating stock market movements due to internal causes and market performance, as well as the potential decline in prediction accuracy when projections are made too far in advance, are counterarguments to consider.

The market was significantly impacted by the 2019 pandemic. Analysis was done on how the COVID-19 epidemic affected the stock performance of Chinese companies. While some industries saw a large decline, the healthcare and telecommunications industries did well. The study found that by adjusting their investment strategy to focus on companies in the healthcare sector or those with high levels of digitalization during the epidemic, investors were able to reduce risks and even make money [2].

Adebiyi et al.'s research shows that ARIMA models, which are frequently used to forecast stock prices, have significant potential for short-term forecasting. It makes them a useful tool for investors looking to make profitable investment decisions. The article offers a thorough procedure for developing ARIMA models for stock price prediction and shows that they are equally accurate as current stock price prediction techniques in terms of short-term stock price predictions [3].

In order to determine how well news sentiment analysis can be used to forecast stock trends, [4] used three different classification models (Naive Bayes, Random Forest, and SVM). The Bag of Word text mining technique is used by the sentiment detection algorithm, which is based on a lexicon of positive and negative words. The findings show that RF and SVM have an accuracy rate of over 80% in all testing categories. According to the article, the efficient-market hypothesis, which maintains that stock market values are basically unpredictable, questions the accuracy of the prediction model. It may be difficult for the sentiment detection algorithm to accurately capture the sentiment of increasingly complex news pieces.

In [5], the authors examine the thirteen-year association between stock prices and five financial indexes that cover several industries. According to their analysis, there is a significant relationship between the stock prices in the pharmaceutical sector and all five indexes. In contrast, compared to other indexes, the Electric Information industry's revenue or net profit exhibited a higher correlation. Stock prices showed a greater link with Revenue per share or Earnings per share in the wine-making sector. Trading Volume and stock prices showed a stronger link in the real estate sector. In sectors like the Power Equipment business, there was little link between the indexes and stock prices. These results can help investors make wise judgements about their long-term investments by providing important guidance.

Using ten technical indicators from ten years of historical data, Chhajer et al. investigated the effectiveness of several Tree-Based models and Neural Networks for forecasting stock prices from the Tehran Stock Exchange [6]. The LSTM algorithm had the lowest error rate and the highest capacity to match data, according to the study, which evaluated nine machine learning models and two deep learning techniques. The Adaboost Regressor showed the highest accuracy among the Tree-Based models, along with outstanding fit and runtime. The goal of the study was to reduce stock market risks associated with trend forecast.

III. Methodology

IV. The project code is written in Jupyter Notebook. The dataset is then loaded and manipulated through mathematical operations using libraries like Pandas and Numpy. Utilising Sklearn, four distinct machine learning algorithms are implemented. Matplotlib is employed to plot graphs. The stock of interest was Google (ticker symbol: GOOGL), and historical data going back 13 years were obtained from the Yahoo Finance website. In other subsections, the steps are detailed in greater detail.

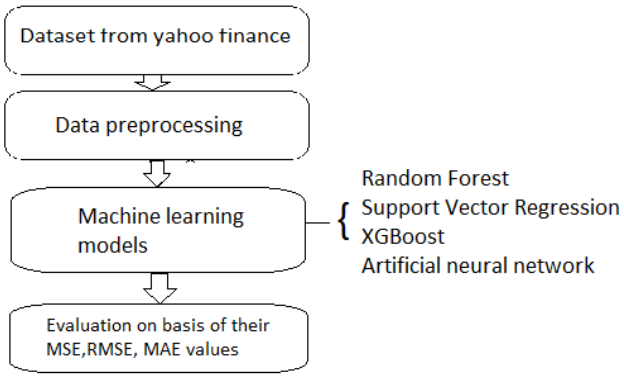

Fig. 1: Flow diagram of Model.

Fig. 1 shows the methodology adopted in this project.

Dataset

For this analysis, daily Google stock price data from March 2010 to March 2023 were used. The yahoo finance website is used to download the data. The dataset consists of different parameters, such as the date, the price at the start of the day, the price at the end of the day, the lowest price and the highest price in a day, the adjusted closing price, and the total amount of training activity done on that particular date. Figure 2 displays the dataset's first five components. There are 3273 rows and 7 columns in the dataset.

	Date	Open	High	Low	Close	Adj Close	Volume
0	2010-03-02	13.337010	13.590559	13.325303	13.475989	13.475989	17492521
1	2010-03-03	13.508367	13.651830	13.430908	13.582091	13.582091	12403919
2	2010-03-04	13.611481	13.851332	13.604009	13.812976	13.812976	12782934
3	2010-03-05	13.981344	14.138754	13.945230	14.052577	14.052577	15707455
4	2010-03-08	14.066774	14.076737	13.972876	14.009489	14.009489	9581379

Fig.2: The first five rows of the Google dataset

A. *Data Pre-processing*

The Technical Analysis module in Python is then used to calculate the technical indicators quickly. Table 1 includes the technical indicators, their usage, and the number of days for which each value is determined. A new column is introduced with the closing prices for the following day. This will serve as the target variable for the prediction we will make in the following phase. After that, the data is examined for contradictions. Technical indicators were the features we desired for this investigation, so a new dataframe containing these indicators is created. The data is split into training and testing sets, with seventy and thirty percent of data respectively. The data is then scaled using the Standard Scaler from the Sklearn module.

TABLE I

THE TECHNICAL INDICATORS USED IN THE STUDY [7]

Technical Indicators		**Definition**	**No. of Days**
SMA	Simple Moving Average	It is the average of the price of the stock over time period t. It assigns same weight to all values.	10,20,50,100,200
EMA	Exponential Moving Average	Same as SMA but assigns more weightage to recent data points	10,20,50,100,200
MACD	Moving Average Convergence/ Divergence	It is calculated by subtracting EMA for 26 days from EMA of 13 days	26,12
RSI	Relative Strength Index	It ranges from 0 to 100 and indicates if the stock has been overbought.	14,28
BB	Bollinger Bands	It consists of three bands where middle band denotes the SMA. The upper and lower bands are plotted *k* standard deviations away from middle band.	20
CCI	Commodity Channel Index	It measures the deviation of the stock price from its mean	14, 50, 100
ATR	Average True Range	It measures the volatility of the stock price movement for time *t*	14

B. *Feature Selection*

Scikit-learn's SelectKBest method chooses the k-best features from a dataset on the basis of their statistical performance. The F-statistic and p-value are calculated by f_regression for each feature. The F-statistic denotes the proportion of explained to unexplained variance, and the p-value denotes the statistical significance of the F-statistic. The ten most significant features with the highest F-statistic and lowest p-value, are the features that are most significant and insightful for our model. They have been chosen using SelectKBest and f_regression jointly. As a result, the dataset's dimensionality is decreased, and the accuracy, effectiveness, and interpretability of the model are all enhanced. The following features were selected after this process- SMA for 50 and 100 days, EMA for 100,200 days and lower, upper and middle bands, RSI and ATR for 14 days.

C. *Machine Learning Models*

Random Forest Regression- They are a combination of tree predictors in which each tree is reliant on the values of a random vector sampled independently and with the same distribution for all trees in the forest [8]. Each tree in the ensemble is built using a random subset of the features and a random subset of the training data in Random Forest. The algorithm's output is then the average of all the individual trees' forecasts. The randomization of both features and data subsets makes the model more resistant to overfitting and improves prediction accuracy. To develop a Random Forest model for this project, we use the 'RandomForestRegressor' class from the 'sklearn.ensemble' package.

Support Vector Regression: Based on the concept of building a hyperplane in a high-dimensional feature space that is mapped from the input space by a collection of basic functions, the support vector regression (SVR) method is used to analyse data [9]. SVR finds the margin between the support vectors and the hyperplane that is closest to the hyperplane, maximises this margin, and reduces prediction error. SVR is a good choice for stock price prediction using technical indicators since it is able to catch complicated patterns and trends in the data. To attain high prediction accuracy, SVR needs huge datasets and careful selection of hyperparameters and kernel functions.

XGBoost: Extreme Gradient Boosting, often known as XGBoost, is an ensemble model that combines a number of weak decision trees to create a robust prediction model. XGBoost is well suited for large datasets due to its speed, scalability, and capacity to manage missing data and outliers, among other benefits. With an accuracy rate of 56.8% on a Chinese stock dataset, XGBoost outperformed other machine learning algorithms in predicting stock values, according to Wang et al [10]. XGBoost does, however, have significant drawbacks. Overfitting is one potential problem that can happen when a model is overly complicated and seeks to fit the noise in the data instead of the underlying trend. By adjusting the model's hyperparameters, such as the learning rate and the maximum depth of the trees, overfitting can be reduced. The interpretability of XGBoost is another drawback. Although the model can calculate feature importance scores, it might be challenging to pinpoint how the model came to a specific conclusion. Transparency and compliance with regulations may suffer as a result.

ANN: Artificial neural networks (ANNs) are a family of flexible, nonlinear models that can accurately estimate any function to any needed level [11] if given enough data and computing resources. They can recognise complex patterns and correlations within stock data. Hence, they are well suited for analysing large volumes of financial data for stock price prediction using technical indicators. However, they can be difficult to understand, require a lot of historical data, and are prone to overfitting. The versatility, speed, and ability of ANNs to predict a range of stock values have earned them a stellar reputation. The quality and quantity of the data, however, governs how effective ANNs are. ANNs are trained using a technique called backpropagation, which compares model predictions to actual outputs and uses the resulting error to update the network's weights.The model's projected outcomes are compared to the actual outputs by repeating this process. The ANN algorithm is built using the Keras sequential model. In this project, a sequential model is initialised using the Keras framework. The creation of two further hidden layers follows, the second having 32 units with ReLU activation function and the first having 64 units. The last step is to construct a single-unit output layer with a linear activation function.

D. Evaluation of the Models

RMSE, MAE and MSE are the metrics used to evaluate performance of regression models. They are explained below and the formulae for calculating them are given in Table 2. The actual value is represented as y_i and predicted value is represented as $\hat{y}_i$ in the given table.

TABLE II

EVALUATION METRICS AND THEIR FORMULAE.

Evaluation metric	Formulae
MSE	$1/n * \Sigma(y_i - \hat{y}_i)^2$
MAE	$1/n * \Sigma\|y_i - \hat{y}_i\|$
RMSE	sqrt(MSE)

E. Hyper-parameter Tuning

The choice of hyperparameters can have a considerable impact on the model's performance, hence hyperparameter tweaking is a crucial stage in the machine learning pipeline [10]. A model that has been fine-tuned can increase accuracy, lessen overfitting, and hasten training. To choose the ideal parameters for each method, we employ the grid search technique.

V.RESULTS AND DISCUSSION

There were two stages to the process. one in which technical indicators were computed. The 'ta' package in Python was used to calculate a total of twenty-two indicators. Then, the top ten indicators were picked to serve as the project's input features for additional research. In the second stage, we built machine learning models and fine-tuned them using the Grid search method, choosing the best parameters for each algorithm.

The primary goal of this project was to evaluate various prediction techniques to forecast future stock prices based on historical returns. The study aimed to identify the most effective algorithm for this purpose, and it has been determined that the XGBoost outperforms other machine learning algorithms in this research, yielding a Root Mean Squared Error (RMSE) value of 1.27. The Random Forest algorithm with a value of 1.84. In comparison, the SVR algorithm trailed the Artificial Neural Network algorithm in terms of RMSE value, coming in at 3.65. We have created a bar graph that displays the results and gives a detailed overview of the RMSE, and other metrics produced by several machine learning methods. We may evaluate and contrast the efficacy of various models using the graph in Fig.3 that shows the performance of various methods and their accompanying RMSE, MAE, and MAE values.

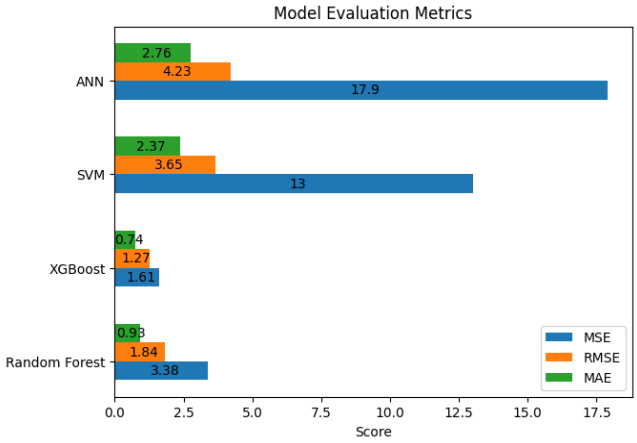

Fig.3: Bar plot of MAE,MSE, RMSE values for each ML model

Fig. 3 shows the bar plot of the error values for each machine learning model. With this information, we can conclude that the XGBoost algorithm and Random Forest Regressor worked the best in predicting stock prices in this study. However, the results may vary considering different datasets. In this paper, we have considered the historical prices of Google stocks. Hence the study is limited to a single dataset.

VI.CONCLUSION

The main goal of this paper was to develop a useful tool that would help stockbrokers and investors make smart stock market investment choices. We used Random Forest, Support Vector Regressor, XGBoost, and Artificial Neural Network machine learning techniques to preprocess data

from thirteen years of Google Stocks to accomplish this goal. After doing calculations, assumptions, and observations, we discovered that the Random Forest algorithm, which yielded a score of 1.84, was closely behind the XGBoost algorithm in terms of lowest RMSE value at 1.27. In comparison, the SVR algorithm trailed the Artificial Neural Network algorithm in terms of RMSE value, coming in at 3.65. The highest RMSE value was achieved by the Artificial Neural Network, at 4.23. To create a stronger model, the future scope may potentially include the integration of sentiment analysis of news, social media data, and their impact on stock prices. Additionally, we can combine the algorithms employed in this study to make meta-models, then compare their performance to previous research. Overall, the project's findings show how machine learning algorithms may utilize technical indicators to predict stock values and offer useful information to both investors and stockbrokers. Particularly the Random Forest and XGBoost algorithms show excellent results, which makes them ideal for further study and real-world financial industry applications.

REFERENCES

[1] Dinesh, Shoban & Rao, Nithin & Anusha, S & R, Samhitha. (2021). Prediction of Trends in Stock Market using Moving Averages and Machine Learning. 1-5. 10.1109/I2CT51068.2021.9418097.

[2] Wang Z, Zhang Z, Zhang Q, Gao J, Lin W (2021) COVID-19 and financial market response in China: Micro evidence and possible mechanisms. PLOS ONE 16(9): e0256879. https://doi.org/10.1371/journal.pone.0256879

[3] Adebiyi, Ayodele & Adewumi, Aderemi & Ayo, Charles. (2014). Stock price prediction using the ARIMA model. Proceedings - UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, UKSim 2014. 10.1109/UKSim.2014.67.

[4] Milosevic, N. (2018, November 22). Equity forecast: Predicting long term stock price movement using machine learning. arXiv.org. Retrieved Febuary 11, 2023, from https://arxiv.org/abs/1603.00751.

[5] E. Zhang, J. Li, H. Yu, H. Lin and G. Chen, Correlation analysis between stock prices and four financial indexes for some listed companies of mainland China, 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Shanghai, China, 2017, pp. 1-5, doi: 10.1109/CISP-BMEI.2017.8302166.

[6] Chhajer P., Shah M., Kshirsagar A., The applications of artificial neural networks, support vector machines, and long–short term memory for stock market prediction, Decision Analytics Journal, Volume 2, 2022, 100015, ISSN 2772-6622, doi:10.1016/j.dajour.2021.100015.

[7] Kara, Y., Boyacioglu, M. A., & Baykan, O. K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul stock exchange. Expert systems with Applications, 38(5), 5311-5319.

[8] Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

[9] Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.

[10] Wang, D., Wang, Y., & Yu, H. (2019). A hybrid model based on XGBoost and neural network for stock price prediction. In 2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm) (pp. 1-5). IEEE.

[11] Murphy, K. P. (2012). Machine learning: a probabilistic perspective. MIT press.