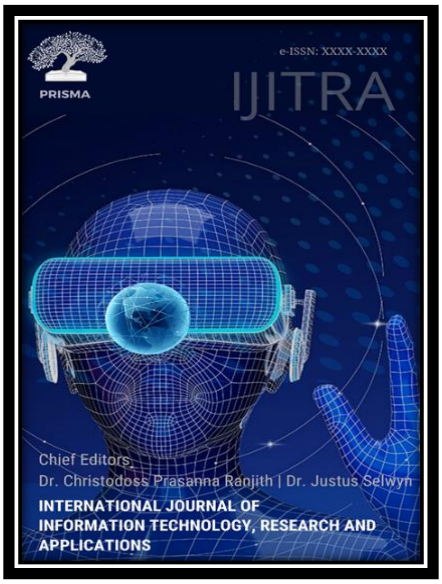

International Journal of Information Technology, Research and Applications (IJITRA)

Jency Jose, Varshini P, (2024). Integrating Technical Indicators and Ensemble Learning for Predicting the Opening Stock Price, 3(2), 01-15.

ISSN: 2583 5343

DOI: 10.59461/ijitra.v3i2.96

The online version of this article can be found at:
https://www.ijitra.com/index.php/ijitra/issue/archive

Published by:
PRISMA Publications

International Journal of Information Technology, Research and Applications (IJITRA) is a journal that publishes articles which contribute new theoretical results in all the areas of Computer Science, Communication Network and Information Technology. Research paper and articles on Big Data, Machine Learning, IOT, Blockchain, Network Security, Optical Integrated Circuits, and Artificial Intelligence are in prime position.

PRISMA
PUBLICATIONS
https://www.prismapublications.com/

Integrating Technical Indicators and Ensemble Learning for Predicting the Opening Stock Price

Jency Jose[1], Varshini P[2]
[1, 2]Department of Computer Science, Mount Carmel College, Autonomous, Bengaluru, India

Article Info

Article history:

Received March 31, 2024
Accepted May 30, 2024
Published June 15, 2024

Keywords:

Technical Indicators
Open Price
Ensemble Technique
ARIMA
Random Forest Regressor
Support Vector Regressor
Gradiant Boosting Regressor

ABSTRACT

Accurately predicting stock prices poses a significant challenge due to the dynamic and complex nature of financial markets. This paper introduces a novel method that combines technical indicators with ensemble learning techniques to effectively forecast opening stock prices. Technical indicators offer valuable insights into market trends and patterns, while ensemble learning methods merge multiple models to enhance predictive precision. The study utilizes various technical indicators such as moving averages, Relative Strength Index (RSI), and Bollinger Bands to capture diverse aspects of market behaviour. Ensemble learning techniques like Random Forest, Gradient Boosting, Support Vector Regressor, and ARIMA model are then employed to consolidate the forecasts from these indicators. The proposed framework is assessed using historical stock market data, and extensive experiments showcase its superior performance compared to individual indicators and traditional forecasting approaches. The findings reveal that integrating technical indicators with ensemble learning leads to a significant improvement in accuracy, with a success rate of 91.45% in predicting opening stock prices, thus providing valuable insights for investors and financial analysts.

Corresponding Author:

Varshini P
Department of Computer Science,
Mount Carmel College, Autonomous,
Bengaluru, India
E-mail: varshinibabu17@gmail.com

1. INTRODUCTION

Financial markets have evolved from traditional exchanges to complex electronic trading systems that connect globally, reflecting the progress of human civilization [1]. A substantial portion of the financial sector involves investors trading securities among themselves, without directly investing in the issuing companies [2]. Financial instruments can be foreign exchange rates, securities, crypto currency, index and funds [3]. Stocks are a financial tool that offers a greater potential for both returns and risks in contrast to the other instruments in the capital market [4]. The share value is influenced by various factors, including the general economic situation, advancements in industries, the operations of companies, and investments. As a result, the stock's value undergoes continuous fluctuations at a fast rate. Information pertaining to the price of the stocks exhibits a low signal-to-noise ratio, thereby presenting a significant obstacle in the realm of forecasting stock prices [5].

The stock market serves firms with the opportunity to raise revenue via the issuance of shares, while also establishing a platform for the trading of these shares [6]. Consequently, the equity market plays a critical role mechanism for facilitating the growth and development of businesses, ultimately contributing to overall economic prosperity. Accurately predicting stock market trends is crucial and highly sought after, as it can result in substantial financial gains through informed decision-making [7]. The opening stock price, in particular, sets the tone for trading throughout the day and is extremely important to market participants. The commencement stock price acts as a pivotal standard, moulding investor outlook and impacting trading selections during the day. The capacity to predict these initial prices accurately is highly significant in enhancing investment profits and lessening risks.

Investors, traders, and financial experts are constantly on the hunt for reliable methods to anticipate stock price variations, as such forecasts are critical for making informed investment decisions. Formerly, the stock market forecasting trends was reliant on statistical analysis and methodologies involving time-series Modeling [8]. Hence, there is a growing interest in exploring new approaches that employ sophisticated data science tools and machine learning.

There are a couple of major categories of prediction methodologies: Technical analysis and Fundamental analysis [9]. Fundamental analysis entails a thorough examination of financial statements, an evaluation of business fundamentals, and an assessment of macroeconomic factors. Conversely, technical analysis offers an alternative viewpoint by interpreting stock prices as a result of supply and demand interactions in financial markets. Technical analysts use charts to visualize past price and volume data, patterns, and technical indicators to predict future price changes. By analysing past and present stock price trends, analysts are able indications to forecast the future [10].

Several models have been utilized previously to forecast the closing prices of forthcoming days, exhibiting a progressive enhancement in accuracy. Nevertheless, in accordance with the Efficient Market Hypothesis, achieving a 100% accurate prediction of the market is unattainable. However, the reality has not deterred traders from persistently endeavouring to attain the utmost potential return on their investments [11].

In general, technical analysis is relies on technical indicators [12]. Stock Technical Indicators encompass statistical computations derived from the price, volume, or significance of a particular share, security, or contract [13]. A technical indicator (TI) is calculated by analysing the time series of prices for stocks and trading volume. This involves considering a specific time period and combining the values of open, low, high, and close prices with the volume size [14]. Various technical indicators, namely, the moving average, moving average convergence and divergence, relative strength index, and commodity channel index, have been utilized to forecast the stock price [15].

According to the survey, scholars have determined that Ensemble learning and deep learning are the two predominant approaches in the realm of machine learning [16]. One extensively studied issue in Machine Learning coursework is the class imbalance datasets [17]. The term "ensemble" pertains to techniques that combine and incorporate multiple base-learners to create a classifier that surpasses them individually [18]. Ensemble learning merges the valuable knowledge acquired from learned features, using several machine learning approaches that have yielded subpar results, to achieve knowledge discovery and enhance predictive performance through adaptive voting schemes [19].

In this paper, a detailed exposition of the proposed methodology is presented, delineating the rationale behind the approach and elucidating the intricacies of integrating mathematical technical indicators with ensembled learning techniques. Technical indicators are frequently employed in financial analysis to identify trends, momentum, and volatility in stock prices. Ensemble learning, On the other hand, is a profound machine learning strategy that combines many base learners to increase prediction performance. By integrating technical indicators with ensemble learning, this study aims to develop a robust predictive model capable of generating accurate forecasts of opening stock prices.

2. LITERATURE REVIEW

This paper [20] introduced the Self-Adapting Variant PSO-Elman Neural Network Model, that combines the Elman neural network with a self-adapting variant Particle Swarm Optimization (PSO) algorithm to forecast stock market opening prices. The model addresses limitations of traditional neural networks by utilizing the dynamic system representation of the Elman network and the optimization capabilities of the PSO algorithm, resulting in improved accuracy and fault resilience. Future enhancements could include parameter optimization, exploration of alternative techniques, and integration of real-time data feeds for better adaptability to changing market conditions.

Another paper [21] The study explored data analysis and stock market prediction using correlation analysis, Facebook Prophet, ARIMA models, and other forecasting techniques such as Weighted Moving Average, Exponential Moving Average, LSTM, and machine learning models. Suggestions for improvement included refining forecasting models, integrating additional predictive algorithms, and incorporating advanced data analysis techniques. Expanding the dataset with more recent stock market data and conducting comparative studies with alternative forecasting methods could enhance prediction accuracy. The research gap identified was the need for a more comprehensive analysis and comparison of various forecasting models, including advanced predictive algorithms and external factors' influence on stock market trends. Overcoming current model limitations and exploring new approaches would bridge this research gap.

The paper [22] analysed the use of machine learning, specifically the SVR model, for predicting stock market trends. It emphasized the importance of ensemble learning by combining LR and KNN models with SVR to improve accuracy. The research aimed to refine stock market prediction techniques and reduce investment risks. It identified a gap in considering influential factors in stock price forecasting and proposed including additional factors. It also suggested optimizing parameter settings for the SVR model to enhance accuracy. These improvements aimed to advance stock market forecasting and improve investment decision-making.

The paper [23] explored a deep neural network framework for predicting stock prices using Stock Technical Indicators (STIs) and LSTM models. Its goal was to provide investors with a reliable method for sustainable investment by combining technical indicators and deep learning strategies. Potential future improvements include advanced optimization techniques, a wider variety of STIs, more stocks, real-time data integration, and improved model interpretability. The paper did not thoroughly examine external factors' influence on stock price prediction and could benefit from comparing different deep learning frameworks or ensemble methods to enhance its impact.

3. METHOD

The research methodology involves a series of essential stages focused on constructing and assessing the suggested framework for forecasting opening stock prices. This study employs a methodology that combines technical indicators and ensemble learning methods to predict opening stock prices. The methodology used in this paper includes analyses, synthesis and mathematical methods with special attention on the method of moving averages. Data analysis can be enhanced through the utilization of correlation plots, moving averages, RSI, volatility, and RMSE[24]. The entire data flow is indicated in the figure 1.

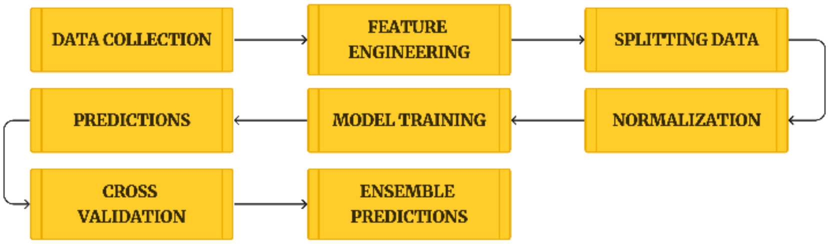

Figure 1. Overview of Prediction Modeling

3.1 Data Collection: -

Data collection is a critical step in the development of Forecasting models for stock price prediction. We obtained the dataset for this paper from Yahoo Finance, a reputable source of financial data. In this paper, yfinance library is utilized to download historical stock price and volume data for the stock symbol "RELIANCE.NS" (Reliance Industries Limited) from January 1, 2018, to March 1, 2024 which is 1523 trading days. The resulting dataset is consisting of Date, Open, High, Low, Close, Adj Close, and Volume for the selected stock symbol over the specified time period.

This analysis will primarily concentrate on the 'Open' price for every trading day, which is regarded as the target variable for the proposed model and the first five rows of the dataset is shown in the Table – I.

```
Data columns (total 6 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Open       1523 non-null   float64
 1   High       1523 non-null   float64
 2   Low        1523 non-null   float64
 3   Close      1523 non-null   float64
 4   Adj Close  1523 non-null   float64
 5   Volume     1523 non-null   int64
dtypes: float64(5), int64(1)
```

TABLE – I

The info() method is called to display information about the dataset, including the data types of each column and the number of non-null values. This provides a summary of the dataset's structure and characteristics. The dataset was sourced from Yahoo Finance does not necessitate extensive data cleaning since it originates from a reliable source.

3.2 Feature Engineering: -

A share technical indicator is a set of data points that are obtained by using a function to the price feature at time *t* and study period n [25]. There are enormous trading indicators that are widely recognized and extensively used by traders worldwide [12]. In this paper (shown in Figure 2), 8 technical indicators have been implemented as part of feature engineering.

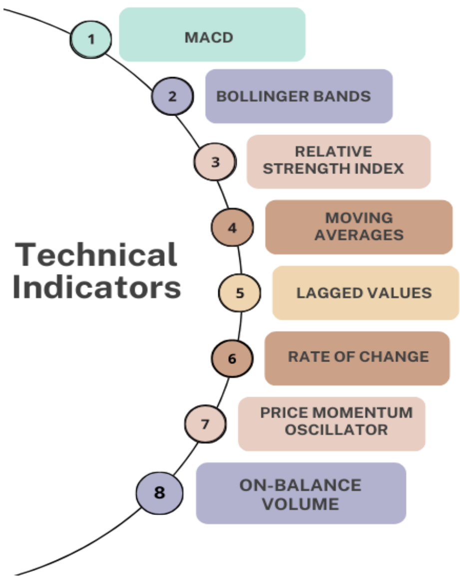

Figure 2. Overview of Technical Indicators

The Moving Average Convergence/Divergence (MACD):

The Moving Average Convergence/Divergence (MACD) falls under the category of trend indicators that illustrate the correlation between prices and moving averages [26]. The advantage of application of this indicator is to equally useful for both the short-term and long-term oriented investors [27]. The mathematical formula for the MACD:

Calculate the 12-period Exponential Moving Average (EMA) for the opening prices:

$$EMA_{12} = \frac{2}{12+1} \times Open_{current} + \left(1 - \frac{2}{12+1}\right) \times EMA_{12_{previous}}$$

Where:

- $Open_{current}$ is the opening price of the current period.
- $EMA_{12_{previous}}$ is the 12-period EMA value from the previous period.
- The value $\frac{2}{12+1}$ is the smoothing factor for the 12-period EMA.

Calculate the 26-period Exponential Moving Average (EMA) for the opening prices:

$$EMA_{26} = \frac{2}{26+1} \times Open_{current} + \left(1 - \frac{2}{26+1}\right) \times EMA_{26_{previous}}$$

Where:

- $Open_{current}$ is the opening price of the current period.
- $EMA_{26_{previous}}$ is the 26-period EMA value from the previous period.
- The value $\frac{2}{26+1}$ is the smoothing factor for the 26-period EMA.

Calculate the MACD Line by subtracting the 26-period EMA from the 12-period EMA:

$$MACDLine = EMA_{12} - EMA_{26}$$

The MACD Line is calculated by utilizing a specific indicator that is based on the exponential moving averages (EMAs) of the opening prices with periods of 12 and 26. Subsequently, the 26-period EMA is subtracted from the 12-period EMA in order to derive the MACD Line.

Bollinger Bands:

Bollinger Bands generate a bullish signal when the price breaks above the lower bands and reverses towards the upper direction, while a bearish signal is triggered when the price breaches the upper bands and reverses downwards [28]. The bands adapt flexibly to changes in market volatility, expanding in times of heightened volatility and contracting in periods of stability within the market. The mathematical formula is:

Calculate the 20-period Simple Moving Average (SMA) for the opening prices:

$SMA_{20} = \frac{1}{20}\sum_{i=1}^{20} Open_i$

Where:

- $Open_i$ is the opening price of the i^{th} period within the 20-period window.

Calculate the 20-period Standard Deviation (SD) for the opening prices:

$SD_{20} = \sqrt{\frac{1}{20}\sum_{i=1}^{20}(Open_i - SMA_{20})^2}$

Where:

- $Open_i$ is the opening price of the i^{th} period within the 20-period window.
- SMA_{20} is the 20-period Simple Moving Average calculated in step 1.

Calculate the Upper Bollinger Band (Upper Band):

$Upper\_Band = SMA_{20} + 2 \times SD_{20}$

Calculate the Lower Bollinger Band (Lower Band):

$Lower\_Band = SMA_{20} - 2 \times SD_{20}$

The Upper Band is the SMA plus two times the SD, while the Lower Band is the SMA minus two times the SD.

Relative Strength Index (RSI):

The RSI is a widely used indicator for predicting market price changes in research and simulations [29]. The RSI first computes the price changes, separating gains and losses, calculating the average gains and losses over the specified window period, and then using these values to compute the Relative Strength (RS) and finally the Relative Strength Index (RSI).

Calculate Price Changes (Delta):

$\Delta = Open_{current} - Open_{previous}$

Where:

- $Open_{current}$ is the opening price of the current period.
- $Open_{previous}$ is the opening price of the previous period.

Separate Gains and Losses:

- Calculate gains when $\Delta > 0$ and set losses to zero.
- Calculate losses when $\Delta < 0$ and set gains to zero.

Calculate Average Gain (gain) over the Window Period:

$\text{gain} = \frac{\sum_{i=1}^{window} \text{gain}_i}{window}$

Where:

- gain_i is the gain for the i^{th} period within the window.
- $window$ is the specified window period (default value is 14).

Calculate Average Loss (loss) over the Window Period:

$\text{loss} = \frac{\sum_{i=1}^{window} \text{loss}_i}{window}$

Where:

- loss_i is the loss for the i^{th} period within the window.
- $window$ is the specified window period (default value is 14).

Calculate the Relative Strength (RS):

$RS = \frac{\text{gain}}{\text{loss}}$

Calculate the Relative Strength Index (RSI):

$RSI = 100 - \left(\frac{100}{1+RS}\right)$

Moving Averages (MA):

Moving averages (MA) are widely used in finance to reduce price variations and identify patterns over a specific time period. They validate trends and signal potential buying nor selling opportunities by analysing intersections or deviations between different moving averages. Calculating the moving averages are straightforward averages of the opening prices over the specified window periods (50 for MA_50 and 200 for MA_200). These moving averages helps in smoothing out price feature, providing a clear indication of the underlying trend over the respective timeframes.

Calculate the 50-period Simple Moving Average (MA_50):

$MA_{50} = \frac{1}{50} \sum_{i=1}^{50} Open_i$

Where:

- $Open_i$ is the opening price of the i^{th} period within the 50-period window.

Calculate the 200-period Simple Moving Average (MA_200):

$MA_{200} = \frac{1}{200} \sum_{i=1}^{200} Open_i$

Where:

- $Open_i$ is the opening price of the i^{th} period within the 200-period window.

Lagged Variables:

Lag variables, also known as lagged values, are historical data points used in forecasting future prices of a time series variable. They are used as features in predictive Modeling to capture trends and improve forecasting accuracy. For each lag *i* the code shifts the opening prices backward in time by *i* periods, generating lagged variables that capture the historical values of the opening prices. These lagged variables can be useful for various purposes in data analysis, such as Modeling time series data or capturing the autocorrelation structure in the dataset.

The mathematical formula for generating lag features using the given code is:

$$\text{lag}_i = \text{Open}_{t-i}$$

Where:

lag_i represents the lag feature at time t for the specified lag i.

Open_{t-i} represents the opening price at time $t-i$, where i ranges from 1 to the specified number of lags.

Rate of Change:

The Rate of Change (ROC) is a tool used to gauge the forward motion of a financial instrument by calculating the percentage change in price over a set timeframe. It aids traders and analysts in evaluating both the velocity and trajectory of price fluctuations within the market. ROCS, a derivative of ROC, minimizes oscillations in figures, smoothing the data and reducing interference for a distinct representation of the pattern. The rate of change is determined by analysing the percentage difference in the 'Open' price between consecutive periods.

Rate of Change (ROC):

The rate of change is calculated as the percentage change in the 'Open' price from one period to the next.

$ROC = \frac{P_t - P_{t-1}}{P_{t-1}}$

Where:

- P_t represents the 'Open' price at time t.
- P_{t-1} represents the 'Open' price at the previous time period ($t-1$).

Smoothed Rate of Change (ROCS):

The smoothed rate of change is calculated by taking the rolling mean of the ROC values over a specified window size.

$ROCS_t = \frac{\sum_{i=t-n+1}^{t} ROC_i}{n}$

Where:

- $ROCS_t$ represents the smoothed rate of change at time t.
- n represents the window size (in this case, 10).
- ROC_i represents the rate of change at time i.
- The sum is taken over the previous n periods, including the current period.

Price Momentum Oscillator (PMO):

The PMO is a technical indicator that measures price momentum by comparing two moving averages. It helps traders and analysts identify changes in market sentiments and trend reversals. The primary purpose of utilizing the PMO is to ascertain the trajectory and intensity of a price trend in a security. Traders often seek out disparities between the PMO and the security's price, as these disparities can serve as indicators for possible reversals in the trend's direction. Both traders and analysts rely on the PMO to detect signals that indicate bullish or bearish market conditions.

Short-term Simple Moving Average (Short_SMA):

- This calculates the average price over a short window size (in this case, 10 periods).

$Short\_SMA_t = \frac{\sum_{i=t-n+1}^{t} P_i}{n}$

Where:

- $Short\_SMA_t$ represents the short-term SMA at time t.
- P_i represents the 'Open' price at time i.
- n represents the short window size (in this case, 10 periods).
- The sum is taken over the previous n periods, including the current period.

Long-term Simple Moving Average (Long_SMA):

- This calculates the average price over a longer window size (in this case, 30 periods).

$Long\_SMA_t = \frac{\sum_{i=t-m+1}^{t} P_i}{m}$

Where:

- $Long\_SMA_t$ represents the long-term SMA at time t.
- P_i represents the 'Open' price at time i.
- m represents the long window size (in this case, 30 periods).
- The sum is taken over the previous m periods, including the current period.

Price Momentum Oscillator (PMO):

- This calculates the difference between the short-term SMA and the long-term SMA.

$PMO_t = Short\_SMA_t - Long\_SMA_t$

Where:

- PMO_t represents the PMO at time t.
- $Short\_SMA_t$ represents the short-term SMA at time t.
- $Long\_SMA_t$ represents the long-term SMA at time t.

Smoothed PMO (Smoothed_PMO):

- This calculates the moving average of the PMO values over a specified window size (in this case, 10 periods).

$Smoothed\_PMO_t = \frac{\sum_{i=t-w+1}^{t} PMO_i}{w}$

Where:

- $Smoothed\_PMO_t$ represents the smoothed PMO at time t.
- PMO_i represents the PMO at time i.
- w represents the window size for smoothing (in this case, 10 periods).
- The sum is taken over the previous w periods, including the current period.

On-Balance Volume (OBV):

On-Balance Volume (OBV) is a trading indicator that connects volume with price fluctuations. It operates on the principle that volume tends to precede price changes. When prices rise, OBV increases volume, and when prices fall, OBV decreases volume. This indicator provides a continuous measure of volume, helping traders determine the direction of volume flow in a security.

Given OBV_t represents the OBV value at time t, the smoothed OBV value at time t, denoted as $SOBV_t$, is calculated as:

$$SOBV_t = \frac{1}{n} \sum_{i=t-(n-1)}^{t} OBV_i$$

Where:

- t represents the current time point.
- n is the window size, representing the number of periods over which the moving average is calculated.
- OBV_i represents the OBV value at time i.

Following the addition of technical indicators, the dataset has expanded to 27 columns. In order to handle potential null values, the dataset is cleaned again before splitting into training and testing data.

3.3 Splitting the Data and Normalization:
The dataset is separated into training and testing subsets, with 70% for training and the rest for testing. Missing values in the training dataset are replaced with column means. Remaining missing values are checked and a warning is given if found. The training dataset summary is reviewed to ensure readiness for analysis, important for accurate machine learning model training. Min-Max scaling is employed to standardize the numerical characteristics in both the training and testing datasets. This scaling guarantees that all features are presented on a comparable scale, which is of utmost importance for proficient model training. In particular, we rescale features associated with stock market indicators, including moving averages, MACD, Bollinger Bands, RSI, and lag features. Following the normalization process, we verify the structures of the training and testing datasets to ensure the integrity of data and its preparedness for subsequent analysis.

3.4 Predictive Model:
We're training a few machine learning models (shown in figure 3) to predict stock market prices. We start by attempting to train an ARIMA model, using historical 'Open' prices as input and tuning its parameters for optimal performance. Then, we train Random Forest, Gradient Boosting, and Support Vector Regression models using scaled training data, which includes various stock market indicators. Each model is configured with specific hyperparameters and trained to learn the relationship between input features and stock prices. These models will be evaluated to determine the most effective approach for price prediction.

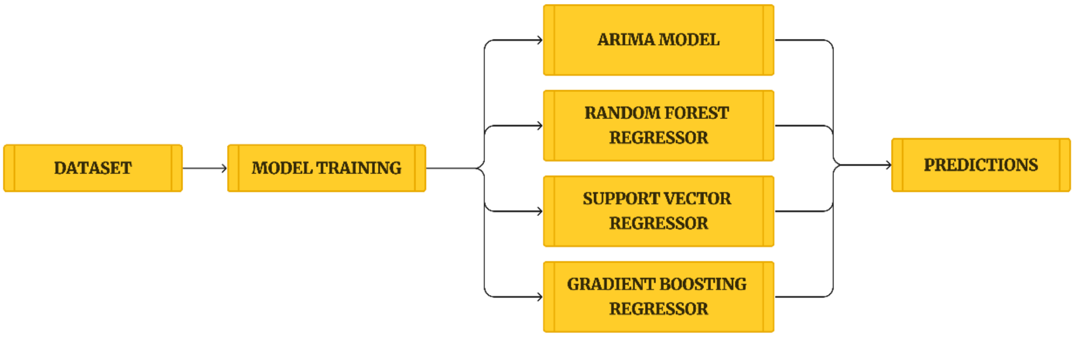

Figure 3 Overview of the proposed model

AutoRegressive Integrated Moving Average (ARIMA) Model:
The autoregressive integrated moving average model plays a major factor in the field of time series analysis [30]. The ARIMA model is characterized by three parameters (p, d, q) which correspond to the autoregressive order, the differencing order, and the moving average order, respectively. These parameters play a crucial role in determining the statistical significance of the model accuracy [31]. The ARIMA model is often written in a more simplified form:

Given a time series y_t, an ARIMA(p, d, q) model can be represented as:

$$(1 - \phi_1 L - \phi_2 L^2 - \ldots - \phi_p L^p)(1 - L)^d y_t = (1 + \theta_1 L + \theta_2 L^2 + \ldots + \theta_q L^q)\varepsilon_t$$

Where:

- L is the lag operator, such that $L^i y_t = y_{t-i}$.
- $\phi_1, \phi_2, \ldots, \phi_p$ are the autoregressive parameters representing the AR terms.
- $\theta_1, \theta_2, \ldots, \theta_q$ are the moving average parameters representing the MA terms.
- d is the degree of differencing applied to make the time series stationary.
- ε_t is white noise (error term) assumed to be normally distributed with mean zero and constant variance.

Gradient Boosting Regressor:
The Gradient Boosting Regressor (GBR) is a type of ensemble model consisting of a series of tree models arranged sequentially in an iterative manner, allowing each subsequent model to learn from the errors of its predecessor [32]. Known for its durability, thorough analysis of important features, and accurate predictions achieved through careful tuning of hyperparameters. The Gradient Boosting Regressor (GBR) model can be represented as follows:

$$\hat{y}_i = \sum_{k=1}^{K} \gamma_k h_k(x_i) + c$$

Where:

- $\hat{y}_i$ is the predicted value for observation i.
- K is the number of trees (boosting stages) in the ensemble.
- γ_k is the shrinkage parameter (learning rate) controlling the contribution of each tree.
- $h_k(x_i)$ is the prediction of the k-th tree for observation i.
- c is the constant term (intercept).

Support Vector Regressor:

SVR is a machine learning algorithm employed for regression purposes. It is a derivation of the SVM algorithm, typically utilized for classification objectives. The SVR model aims to reduce a loss function that penalizes inaccuracies in forecasting.

$$f(x) = \sum_{i=1}^{n} \alpha_i K(x, x_i) + b$$

Where:

- $f(x)$ is the predicted output for input x.
- α_i are the Lagrange multipliers obtained during training.
- $K(x, x_i)$ is the kernel function, representing the similarity between the input x and the training data x_i.
- b is the bias term.

Random Forest Regressor:

The Random Forest Regressor is a supervised learning algorithm designed for regression tasks. It combines predictions from multiple decision trees to make a final prediction. It is robust to overfitting, handles nonlinear relationships, and is effective for both small and large datasets. This makes it a versatile and powerful mechansim for data scientists and machine learning practitioners.

Here's a simplified explanation of the process:

$$\hat{y} = \frac{1}{T} \sum_{i=1}^{T} f_i(x)$$

Where:

- $\hat{y}$ is the predicted output.
- T is the number of trees in the forest.
- $f_i(x)$ is the prediction of the i-th tree for input x.

3.4 Initial RMSE Predictions:

The code evaluates machine learning models in a regression scenario. Random Forest has an RMSE of 88.78 and an R-squared of 0.8099, showing moderate performance. Gradient Boosting has a lower RMSE of 79.25 and a slightly higher R-squared of 0.8485, indicating a better fit. SVR performs similarly to Gradient Boosting, with an RMSE of 78.34 and an R-squared of 0.8520. However, ARIMA outperforms all other models with an RMSE of 54.16 and an R-squared of 0.9293, demonstrating its superior capability in capturing data variance. In summary, ARIMA is the most effective model among the assessed algorithms.

Through cross-validation, we assess performance and prevent overfitting, ensuring robustness. We then employ ensemble techniques like bagging, boosting, and stacking to combine model intelligence for more accurate forecasts. Visual representations of ensemble predictions offer concise insights into model performance. This phase is essential for enhancing the effectiveness and interpretability of our predictive models, aiding informed decision-making in real-world scenarios.

3.5 Cross Validation:

This code uses k-fold cross-validation to evaluate three machine learning models: Random Forest, Gradient Boosting, and SVR. The results show that Gradient Boosting has the best cross-validated RMSE of about 5.5, indicating its superior predictive accuracy. Random Forest performs decently with an RMSE of around 7.47. On the other hand, SVR has the highest cross-validated RMSE of approximately 35.95, suggesting poorer performance compared to the other models. In conclusion, Gradient Boosting is the top-performing model, followed by Random Forest, while SVR is less effective for this dataset.

3.6 Ensemble Technique:

The ensemble model combines forecasts from ARIMA, SVR, Random Forest, and Gradient Boosting models to produce collective predictions for initial stock prices. Performance metrics such as RMSE, MAE, and R-squared are computed to assess the accuracy of the combined predictions. The ensemble model exhibits robust performance with an RMSE value of 56.699, MAE value of 23.373, and R-squared value of 0.911, indicating a high level of precision in estimating real stock prices. By utilizing this blending approach, the model effectively captures inherent data patterns, resulting in a dependable outlook for initial stock prices.

4. RESULTS AND DISCUSSION

This paper uses a holistic methodology to analyse market dynamics by integrating various technical indicators, providing valuable insights on trends, momentum, and trade entry/exit points through visual representation.

Figure 4

The OHLC chart (as shown in the Figure 4) shows RELIANCE.NS stock's price movements from late 2018 to early 2023. Each vertical line represents a time period's price range, while small horizontal lines indicate opening and closing prices. The stock generally rose from below 1500 to near 3000, with some volatility in early 2020. Despite resistance at 2500, the stock broke through, indicating a strong bullish sentiment.

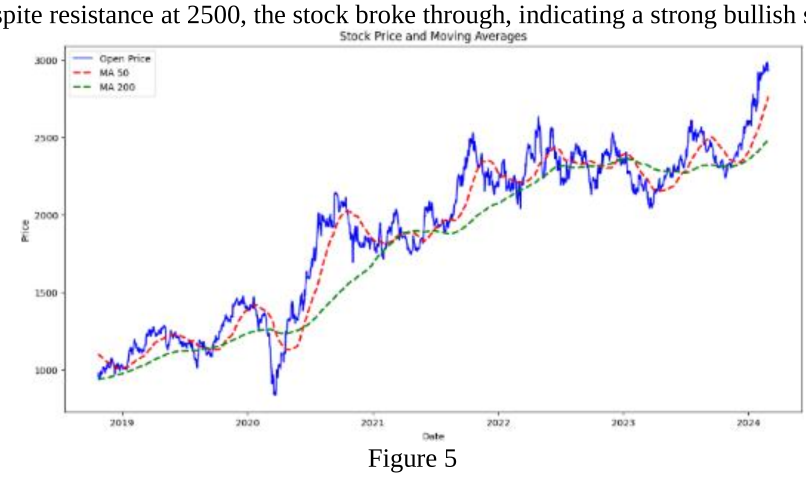

Figure 5

The line chart (as shown in Figure 5) shows stock price fluctuations over time, with the 50-day and 200-day moving averages. The blue line represents the daily open price, the red dashed line is the MA 50, and the green dashed line is the MA 200. Overall, there is an upward trend from early 2019 to early 2023. The moving averages help identify trends, with the MA 50 reacting faster to recent price changes than the MA 200. The chart also depicts "golden cross" and "death cross" moments when the MA 50 crosses above or below the MA 200. There are periods of volatility, including a significant decline and recovery in early 2020. During uptrends, the MA 200 acts as a support level, and the stock price ends above both moving averages, suggesting a positive short-term outlook.

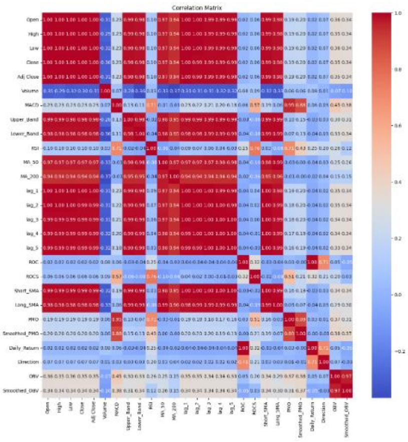

Figure 6

The illustration (Figure 6) shows a Correlation Matrix with correlation coefficients between variables. The diagonal elements are all 1s, indicating perfect correlation within each variable. The colour gradient on the right represents the magnitude and direction of correlation, with red for positive correlation and blue for negative correlation. Some variables, like Open, High, Low, and Close prices, have strong positive correlations with stock price fluctuations. Technical indicators like MA_50 and MA_200 also show significant correlation with stock prices. Negative correlations are rare, and variables like RSI have minimal to no correlation with other variables, shown by white or lightly-coloured cells.

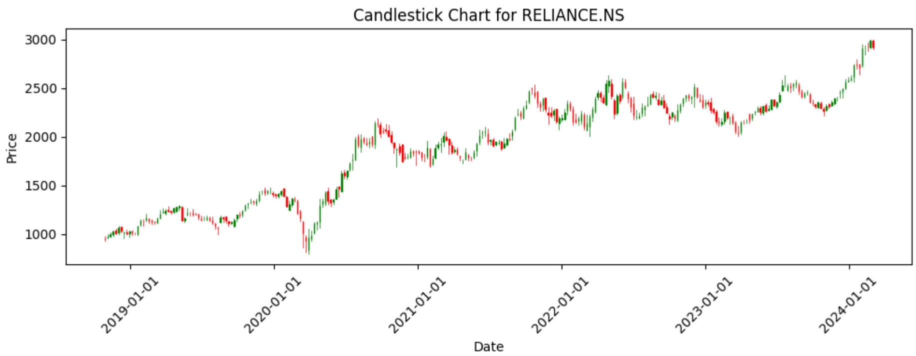

Figure 7

Candlestick charts visually represent price movements of financial instruments like securities, derivatives, or currencies. Each candlestick symbolizes one day of trading with a body and wicks. Analysing the chart (shown in as Figure 7) RELIANCE.NS stock chart from early 2019 to early 2024 shows price fluctuations between 1000 and 3000 currency units. Green candlesticks represent gains, red candlesticks indicate losses. Despite volatility, the stock price trend is upward, with an initial surge, decline, and recovery leading to consistent growth starting in 2021.

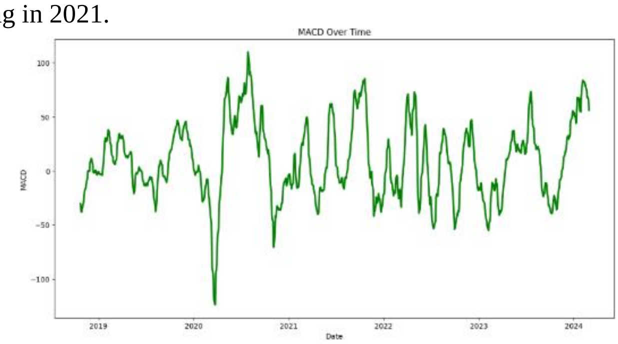

Figure 8

The MACD is a momentum indicator that shows the relationship between two moving averages of a security's price. Values range from -100 to +100 on the chart (as shown in Figure 8), with a MACD above zero indicating a bullish trend and below zero suggesting bearish momentum. The chart displays significant volatility in MACD values, with frequent fluctuations and crossings of the zero-line hinting at potential trend reversals. Traders using MACD for signals may find these crossovers important for entering or exiting positions.

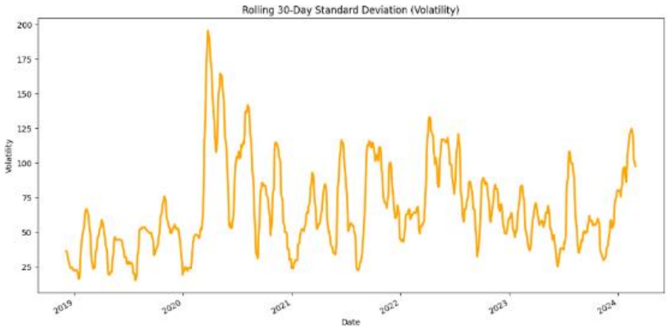

Figure - 9

The chart (shown in Figure 9) shows the Rolling 30-Day Standard Deviation of volatility from 2019 to 2024, covering a five-year period. It demonstrates the changes in volatility over time. The y-axis represents the standard deviation of returns, ranging from 0 to over 200. The graph highlights increase in volatility during times of market tension or uncertainty, as well as periods of decreased volatility indicating more stable market conditions. There are significant spikes in volatility that may be linked to specific events or market situations. Overall, volatility follows a cyclical pattern with no clear long-term trends.

5. CONCLUSION

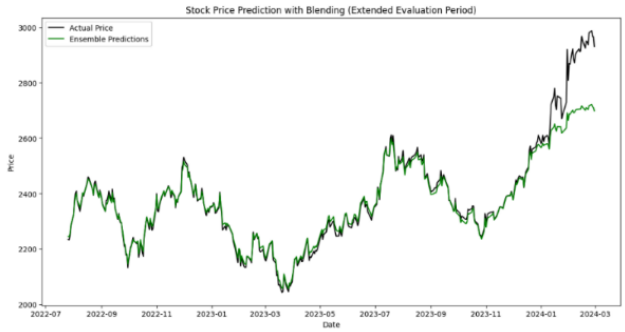

Figure 10

The chart (in the Figure 10) shows historical stock price data from July 2022 to April 2024, comparing actual prices to ensemble forecasts. The black line represents actual prices, fluctuating between 2200 and 3000 units. The green line shows ensemble predictions, closely tracking actual prices but underestimating them towards the end. The graph evaluates a stock price predictive model using diverse algorithms or data sources over an extended period.

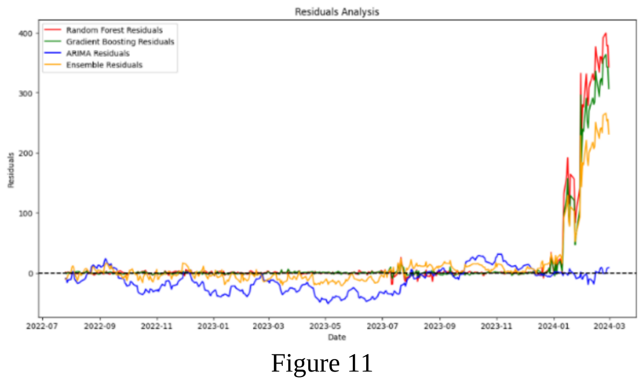

Figure 11

The graph (shown in Figure 11) displays residuals of predictive models over a specific time frame. The x-axis represents time from July 2022 to April 2024, while the y-axis represents disparities between observed and predicted values. The chart includes four lines: Random Forest (red), Gradient Boosting (blue), ARIMA (orange), and Ensemble (green). Overall, all models have residuals close to zero, indicating accurate predictions. However, towards the end of the period, the red and green lines show larger spikes, coinciding with a divergence in actual stock prices. The ARIMA model also exhibits some divergence, while the Gradient Boosting model consistently has the lowest residuals. This suggests that the Gradient Boosting model is the most accurate, particularly towards the end of the time frame. The spike in residuals implies challenges in predicting the sudden increase in stock prices, possibly due to a rapid market movement not captured by historical data.

```
Actual Opening Price and Predicted Values:
      Date      Actual   Predicted
2022-07-21 2301.961182 2297.850739
2022-07-22 2307.499268 2310.405718
2022-07-25 2277.455566 2287.949314
2022-07-26 2234.674561 2251.211556
2022-07-27 2233.613037 2248.961682
```

Figure 12

The combination of ensemble learning techniques and technical indicators has shown promising outcomes, achieving an impressive accuracy rate of 91.45%. This demonstrates the effectiveness of this approach in predicting stock prices. The ensemble model surpasses traditional methods and provides valuable insights for making investment decisions. However, there are instances where the model's predictions deviate from actual prices, especially during significant price surges. Forecasting extreme price movements is a complex task, and even sophisticated models may encounter difficulties. Despite occasional disparities, the model maintains a strong overall performance and delivers reliable forecasts for the majority of the evaluation period.

This can be attributed to various factors such as market volatility, the model's sensitivity to recent data, and changes in market conditions. Stock markets can exhibit high levels of volatility, particularly during economic crises, geopolitical events, or significant company announcements. This inherent volatility makes it challenging to accurately forecast stock prices due to unpredictable fluctuations. Some machine learning models, such as ARIMA, which are based on time series analysis, may be more sensitive to recent data points. Consequently, they may struggle to generalize well to future time periods, especially if the recent data includes unexpected events or anomalies. Moreover, market conditions can evolve over time, leading to shifts in the underlying relationships between technical indicators and stock prices. If the models are trained on data from a different market regime than the one being evaluated, they may fail to capture these changes, resulting in poor performance towards the end of the evaluation period.

REFERENCES

[1] K. K. Cetina *et al.*, "Global Microstructures: The Virtual Societies of Financial Markets l," 2002.

[2] P. Bond Alex Edmans Itay Goldstein, P. Bond, A. Edmans, and I. Goldstein, "THE REAL EFFECTS OF FINANCIAL MARKETS," 2011. [Online]. Available: http://www.nber.org/papers/w17719

[3] F. Kayım and A. Yılmaz, "Financial Instrument Forecast with Artificial Intelligence," vol. 7, no. 2, 2018, doi: 10.5195/emaj.2018.134.

[4] I. Nuzula Agustin, "THE INTEGRATION OF FUNDAMENTAL AND TECHNICAL ANALYSIS IN PREDICTING THE STOCK PRICE," vol. 18, no. 2, pp. 93–102, 2019.

[5] C. Zhao, P. Hu, X. Liu, X. Lan, and H. Zhang, "Stock Market Analysis Using Time Series Relational Models for Stock Price Prediction," *Mathematics*, vol. 11, no. 5, Mar. 2023, doi: 10.3390/math11051130.

[6] Fatima Ruhani, Md. Aminul Islam, Tunku Salha Tunku Ahmad, and Muhammad Ruhul Quddus, "Effects of Financial Market Variables on Stock Prices: A Review of the Literature," *Journal of Modern Accounting and Auditing*, vol. 14, no. 11, Nov. 2018, doi: 10.17265/1548-6583/2018.11.002.

[7] D. P. Gandhmal and K. Kumar, "Systematic analysis and review of stock market prediction techniques," *Computer Science Review*, vol. 34. Elsevier Ireland Ltd, Nov. 01, 2019. doi: 10.1016/j.cosrev.2019.08.001.

[8] R. Ruhal and E. Vandana Prashar, "A Comparative Study Of Statistical Methods And Machine Learning Approaches For Stock Price Prediction," vol. 10, 2023, doi: 10.13140/RG.2.2.19210.44483.

[9] J. Sharmila Vaiz and M. Ramaswami, "A Study on Technical Indicators in Stock Price Movement Prediction Using Decision Tree Algorithms," *American Journal of Engineering Research (AJER)*, vol. 5, pp. 207–212, 2016, [Online]. Available: www.ajer.org

[10] T. Anbalagan and S. U. Maheswari, “Classification and prediction of stock market index based on Fuzzy Metagraph,” in *Procedia Computer Science*, Elsevier B.V., 2015, pp. 214–221. doi: 10.1016/j.procs.2015.03.200.

[11] P. Soni, Y. Tewari, and D. Krishnan, “Machine Learning Approaches in Stock Price Prediction: A Systematic Review,” in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Jan. 2022. doi: 10.1088/1742-6596/2161/1/012065.

[12] S. Deng, T. Mitsubuchi, K. Shioda, T. Shimada, and A. Sakurai, “Combining technical analysis with sentiment analysis for stock price prediction,” in *Proceedings - IEEE 9th International Conference on Dependable, Autonomic and Secure Computing, DASC 2011*, 2011, pp. 800–807. doi: 10.1109/DASC.2011.138.

[13] M. Agrawal, A. U. Khan, and P. K. Shukla, “Stock price prediction using technical indicators: A predictive model using optimal deep learning,” *International Journal of Recent Technology and Engineering*, vol. 8, no. 2, pp. 2297–2305, Jul. 2019, doi: 10.35940/ijrteB3048.078219.

[14] Y. Shynkevich, T. M. McGinnity, S. A. Coleman, A. Belatreche, and Y. Li, “Forecasting price movements using technical indicators: Investigating the impact of varying input window length,” *Neurocomputing*, vol. 264, pp. 71–88, Nov. 2017, doi: 10.1016/j.neucom.2016.11.095.

[15] C. F. Tsai, Y. C. Lin, D. C. Yen, and Y. M. Chen, “Predicting stock returns by classifier ensembles,” in *Applied Soft Computing Journal*, Mar. 2011, pp. 2452–2459. doi: 10.1016/j.asoc.2010.10.001.

[16] I. D. Mienye and Y. Sun, “A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects,” *IEEE Access*, vol. 10. Institute of Electrical and Electronics Engineers Inc., pp. 99129–99149, 2022. doi: 10.1109/ACCESS.2022.3207287.

[17] I. D. Mienye and Y. Sun, “Performance analysis of cost-sensitive learning methods with application to imbalanced medical data,” *Inform Med Unlocked*, vol. 25, Jan. 2021, doi: 10.1016/j.imu.2021.100690.

[18] A. Mohammed and R. Kora, “An effective ensemble deep learning framework for text classification,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 10, pp. 8825–8837, Nov. 2022, doi: 10.1016/j.jksuci.2021.11.001.

[19] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, “A survey on ensemble learning,” *Frontiers of Computer Science*, vol. 14, no. 2. Higher Education Press, pp. 241–258, Apr. 01, 2020. doi: 10.1007/s11704-019-8208-z.

[20] Z. Zhang, Y. Shen, G. Zhang, Y. Song, and Y. Zhu, “Short-term Prediction for Opening Price of Stock Market Based on Self-adapting Variant PSO-Elman Neural Network.”

[21] A. Garlapati, D. R. Krishna, K. Garlapati, N. M. Srikara Yaswanth, U. Rahul, and G. Narayanan, “Stock Price Prediction Using Facebook Prophet and Arima Models,” in *2021 6th International Conference for Convergence in Technology, I2CT 2021*, Institute of Electrical and Electronics Engineers Inc., Apr. 2021. doi: 10.1109/I2CT51068.2021.9418057.

[22] “Stock Prediction Methods based on Ensemble Learning,” *Academic Journal of Business & Management*, vol. 3, no. 6, 2021, doi: 10.25236/ajbm.2021.030619.

[23] M. Agrawal, P. K. Shukla, R. Nair, A. Nayyar, and M. Masud, “Stock prediction based on technical indicators using deep learning model,” *Computers, Materials and Continua*, vol. 70, no. 1, pp. 287–304, 2021, doi: 10.32604/cmc.2022.014637.

[24] A. Garlapati, D. R. Krishna, K. Garlapati, N. M. Srikara Yaswanth, U. Rahul, and G. Narayanan, “Stock Price Prediction Using Facebook Prophet and Arima Models,” in *2021 6th International Conference for Convergence in Technology, I2CT 2021*, Institute of Electrical and Electronics Engineers Inc., Apr. 2021. doi: 10.1109/I2CT51068.2021.9418057.

[25] X. Di, “Stock Trend Prediction with Technical Indicators using SVM.” [Online]. Available: http://finance.yahoo.com

[26] Z. H. Lashkary, “Technical analysis of Forex by MACD Indicator.” [Online]. Available: https://www.researchgate.net/publication/274071851

[27] D. Eric, G. Andjelic, and S. Redzepagic, “Application of MACD and RVI indicators as functions of,” 2009.

[28] D. A. Daniswara, H. Widjanarko, and K. Hikmah, “THE ACCURACY TEST OF TECHNICAL ANALYSIS OF MOVING AVERAGE, BOLLINGER BANDS, AND RELATIVE STRENGTH INDEX ON STOCK PRICES OF COMPANIES LISTED IN INDEX LQ45,” *Indikator: Jurnal Ilmiah Manajemen dan Bisnis*, vol. 6, no. 2, p. 16, Apr. 2022, doi: 10.22441/indikator.v6i2.14806.

[29] L. A. G. de Almeida, "Technical indicators for rational investing in the technology companies: The evidence of FAANG stocks," *Jurnal Pengurusan*, vol. 59, 2020, doi: 10.17576/pengurusan-2020-59-08.

[30] R. Katoch and A. Sidhu, "An Application of ARIMA Model to Forecast the Dynamics of COVID-19 Epidemic in India," *Global Business Review*, 2021, doi: 10.1177/0972150920988653.

[31] T. Alghamdi, K. Elgazzar, M. Bayoumi, T. Sharaf, and S. Shah, "Forecasting traffic congestion using ARIMA modeling," in *2019 15th International Wireless Communications and Mobile Computing Conference, IWCMC 2019*, Institute of Electrical and Electronics Engineers Inc., Jun. 2019, pp. 1227–1232. doi: 10.1109/IWCMC.2019.8766698.

[32] D. A. Otchere, T. O. A. Ganat, J. O. Ojero, B. N. Tackie-Otoo, and M. Y. Taki, "Application of gradient boosting regression model for the evaluation of feature selection techniques in improving reservoir characterisation predictions," *J Pet Sci Eng*, vol. 208, Jan. 2022, doi: 10.1016/j.petrol.2021.109244.

BIOGRAPHIES OF AUTHORS

Jency Jose is an Assistant Professor at Mount Carmel College in Bengaluru. With a profound passion for computer science, Jency has dedicated herself to both teaching and research, aiming to impart knowledge and make impactful contributions to the field. By staying up-to-date with the latest developments and trends in the field, Jency ensures that her teaching is always relevant and cutting-edge. With 14 years of teaching experience, she brings a wealth of knowledge and expertise to her role. Over the course of her career, she has authored and co-authored numerous papers, each contributing to the advancement of computer science. Her primary research interests lie in wireless sensor networks.

Varshini Padmanabhan received her Bachelor's degree in Computer Technology from Sri Krishna Arts and Science College in 2022. She is pursing her Master's degree in Computer Science with Data Science specialization at Mount Carmel College, Autonomous in Bengaluru. Her primary research interests lie in Machine learning and big data analytics. She dedicated to leveraging data-driven approaches to solve real-world problems and make meaningful contributions to the data science community.