Available online at www.HEFJournal.org

ISSN: 2785-2997

Journal of
Human, Earth, and Future

Vol. 3, No. 4, December, 2022

Tesla Stock Close Price Prediction Using KNNR, DTR, SVR, and RFR

Edward [1], Suharjito [2*]

[1] *Computer Science Department, BINUS Graduate Program – Master of Computer Science, Bina Nusantara University, Jakarta, 11480, Indonesia.*

[2] *Industrial Engineering Department, BINUS Graduate Program – Master of Industrial Engineering, Bina Nusantara University, Jakarta, 11480, Indonesia.*

Received 10 September 2022; Revised 18 November 2022; Accepted 28 November 2022; Published 01 December 2022

Abstract

The growth of every nation's economy depends heavily on the stock market. Because of their strong predictive powers, "K-Nearest Neighbor Regression (KNNR), Random Forest Regression (RFR), Decision Tree Regression (DTR), and Support Vector Regression (SVR)" is commonly used. From the previous study, it shows that the chosen state-of-the-art of previous studies is by Shah et al. (2021) using Bi-LSTM. But the state-of-the-art model shows a high Mean Squared Error (MSE) value that is 5% for predict Tesla Stock. Therefore, we propose the SVR, KNNR, DTR, and RFR model hope error value can be reduced. However, those models have difficulties finding suitable parameters for prediction. Due to that, we proposed the best parameters that were commonly used in studies which is SVR with RBF kernel, C = [1; 10; 100], gamma = [0, 1; 0, 2], epsilon = 0,1, KNNR with K = [1, 3, 5, 7, 9], and DTR with a criterion = ['squared_error';'friedman_mse'], RFR with 0 < n_estimators < 100 is picked to enhance the predictive capabilities of each proposed model. To prove this, a comparison is made with each proposed model, and choose the best proposed model then compared to the state-of-the-art of the previous study, which is Bi-LSTM. The dataset used was the Tesla Stock historical data from yahoo finance. The result showed RFR with n_estimators = 87 is superior to its comparison with results of 0.005452 RMSE, 0.000029 MSE, and 0.999296 R^2 and compared to Bi-LSTM have about 0.11773 RMSE, 0.01386 MSE, and 0.791263 R^2. Based on this study, it can be concluded that RFR with n_estimators = 87 has superior predictive capabilities performance compared with other models and the state-of-the-art of previous studies.

Keywords: Tesla Stock; Forecasting; Close Price; SVR; KNNR; DTR; RFR.

1. Introduction

The stock market are place of different kinds of company shares that are sold publicly so that buyers can choose to buy the stock price of a company [1, 2]. The stock itself is a piece of paper that signifies the owner as the owner of the company issuing the stock [3]. Depending on the percentage of shares they own, shareholders are either part owners of the company [4]. According to a statistical report from the Indonesia Stock Exchange, stocks have recently become very popular since the Covid-19 pandemic where there was a very significant increase in stock transactions in 2021, which was 3,577,864 in the first nine months, much more than the previous year which only reached 2,752.471 shares [5].

Investment is the purchase of a future asset with the expectation that the investor will receive a return, including both profit and loss, at some point in the future [6]. Returns from ownership of stock include dividends and capital gains [7]. Shareholders receive profits in the form of dividends and capital gains, which are distributed in proportion to the number

* Corresponding author: suharjito@binus.edu

http://dx.doi.org/10. 28991/HEF-2022-03-04-01

of shares each shareholder owns. As a result, shareholders benefit financially [8]. Investors can earn investment profits by selling them at a higher price and buying shares at a low price [9]. Therefore, investors need to estimate the stock price so that they can buy and sell their shares at the right time so that they can make a profit [10]. That's why we chose stocks to be predicted considering that stocks are an investment that aims to get a profit from buying an asset and selling it back at a higher price.

Basically, the main concern of stock investors is predicting data trends to make the best judgments and plans for the future [11]. How investors predict stock prices can be through fundamental analysis or technical analysis [12]. This analysis is used to make decisions for investors before buying or selling shares [13]. Fundamental analysis bases its predictions on news information, company conditions, and macroeconomic factors such as dividend yield, return on equity, income, P/E, and EBITDA, while technical analysis relies on mathematical indicators that are constructed from the history of the prior stock price time series to make its forecasts, [14]. Technical analysis has proven to be effective in predicting stock prices but is difficult to do properly as evidenced by the percentage of research using technical analysis at most compared to fundamental analysis or a combination of both [15]. For this reason, it's crucial to forecast stock prices to make sure that the investments made will be profitable [16].

Technical analysis is divided into two types, namely candlestick analysis and pattern analysis. Candlestick is a unit representation of stock price movements that contain open, close, high, and low at a certain time [17]. Meanwhile, pattern analysis is analysis of the pattern of stock price chart movements which can take the form of double bottom, saucer, cup and handle, and others [18]. So that technical analysis, both candlestick and pattern analysis, is only a derivative of daily historical price data. Therefore, it can be said that stock price predictions from the next time to the present can be predicted effectively only by using historical stock price data [9]. Historical trend data is crucial since it helps investors to choose whether to purchase, sell, or keep assets based on the company's stock market performance [19]. Therefore we use candlestick analysis in this study.

Although technical analysis is effective in predicting stock prices, it is difficult to do considering that stock price fluctuations change rapidly every time by virtue of the characteristics of stock data are time series data that move continuously over time [11] which are high volatility, nonlinearity, and non-stationary [20] make researchers keep looking to find better and more appropriate models for analysis. The hypothesis of market efficiency states that stock market predictions are impossible [21], however, with current advances in computing technology, it is very possible to make stock market predictions [22]. Therefore, prediction of stock prices has become a very popular topic between finance and computer science [23] where in recent times, researchers have concentrated on issues such the usage of machine and deep learning, the function of artificial intelligence, and related themes. financial industry [24].

Candlestick is a unit representation of stock price movements that contain open, close, high, and low at a certain time [17]. This study was conducted to predict the close price by virtue of the close price is important in the finance sector and affects the calculation of return and valuation values [25]. Investors can get returns from investing by selling stocks at high prices and buying them when stocks are low [9]. Therefore, we chose the close price as the variable to be predicted because it affects the profits earned and has an impact on stock investors.

By market capitalization, Tesla rose to become the most valuable automaker in North America and a significant force in the sector, to the world's leading electric vehicle manufacturer [26]. Additionally, in 2018 one of its design models not only held the top market share position in the automobile sector in the United States but also outsold every other midsize premium sedan combined, accounting for 52% of the entire market segment [27]. Among the most creative businesses in recent years is Tesla, with other companies including Amazon, Apple, Microsoft, Samsung, and Facebook [28]. Therefore, we chose Tesla's stock to predict its stock price because it dominates the automotive market segment as well as many innovative ideas to develop in the future so that future profits can be seen clearly.

Several machine learning methods [29] or deep learning [30] to predict stock prices using only historical stock price data with a fairly high level of accuracy and accuracy such as stock predictions Microsoft with ARIMA, Support Vector Regression (SVR), and Neural Network shows that SVR is superior to all other models considering that it has the smallest RMSE value [31]. Another example, predicting the stock of PT. Waskita Karya (Persero) Tbk with SVM for stock data classification, SVR as a method for feature selection, and KNNR for stock price prediction which shows KNNR has a sufficient percentage of accuracy which is around 93.33% [32]. Other research on stock price predictions from Telkom Indonesia, XL, SmartFren, and Indosat with MLR, KNNR, DTR, SVR, and which shows DTR is superior to all other models because it has the lowest MAPE value [33]. Furthermore, research on Apple stock price predictions with SVR, DTR, PR, RFR, SLR, and which shows RFR is superior to all other models for the reason that it produces the highest accuracy [34]. Therefore, we chose SVR, KNNR, DTR, and RFR machine learning models to predict tesla stock price is commonly used because of its high predicting capabilities.

For Tesla stock predictions, models that have been used include research by Bhattacharjee & Bhattacharja (2019) [35] conducting research using the Ridge Regression, Single Layer Perceptron (SLP), Multi-Layer Perceptron (MLP), Random Forest, KNN, SVM, Lasso Regression, Simple Linear Regression (SLR) model, and LSTM. Another study by

Agrawal (2021) [36] uses Linear Regression and Deep Learning. As well as research by Shah et al. (2021) [1] using LSTM and BI-LSTM. However, until now, a focused model was not been developed to predicted the movement of Tesla shares using RFR, KNNR, DTR, and SVR. The regression prediction model predicts a value in the form of an integer or a decimal number, then the regression model is evaluated by calculating error value in the prediction [37]. There are many ways to calculate the error value, which we use as an evaluation of the regression model are MSE, R^2, and (RMSE). The minimum error value for predict stock achieved by study of Rubi et.al. (2022) with Artificial Neural Networks (ANN) model is the best. But research from Ma (2020) [38] compare LSTM for predicting stock prices using ANN and ARIMA models, the results indicate that LSTM model has the best predictive power. Also, from research of Shah et al. (2021) [1], the result show that Bi-LSTM produces best result than LSTM. Therefore Bi-LSTM model has been chosen as the state-of-the-art of previous studies.

From previous study above, it shows that the chosen state-of-the-art of previous studies is by Shah et al. (2021) [1] using Bi-LSTM. But the state-of-the-art model show a high Mean Squared Error (MSE) value that is 5% for predict Tesla Stock. While SVR, KNNR, DTR, and RFR models has many advantages like predicting with minimum MSE and (RMSE), it is also having a weakness such as it is very dependent on its parameters for prediction. In solving Machine Learning dependency of parameters for its prediction, finding suitable parameters that can be used in prediction which was shown in many of the previous research such as manually looking for the best parameter like Muflikhah & Haryanto (2018) [39] algorithms study to solve the dependency of parameters. However, there are plenty of parameters to be tuned in machine learning. Therefore, implementing the best parameters to be tuned which had better performance than the other parameters in solving a constrained parameters problem, could make a new performance that surpassed the previous state-of-the-art of the previous studies. To validate this, this study compares each models so that this study method is capable of outperforming earlier studies to predict stock hope error value can be reduced.

The following succinct summary represents the study's main contributions:

- Using the KNNR, RFR, SVR, and DTR each of these models will be carried out various kinds of tests such as the SVR will be tested with Kernel RBF with C = [1, 10, 100], gamma = 0.1 or 0.2, and epsilon = 0.1. The K-Nearest Neighbor Regression (KNNR) model will be tested with five K values, namely from K = [1, 3, 5, 7, 9], the Decision Tree Regression (DTR) model will be tested with two criterion types, namely "squared_error" and "friedman_mse", and the Random Forest Regression (RFR) model will be tested with 100 "n_estimator" values, namely n_estimator = 1 to n_estimator = 99.
- Choosing the best model from the KNNR, SVR, RFR, and DTR models based on RMSE, R^2 and MSE, which has the smallest RMSE, R^2, and MSE from other models to help investors and the government to support their decision-making progress with best model as a predictive method.
- Proving that the chosen best model gets the results of MSE and RMSE which are smaller and R^2 which is greater than the earlier research model in the previous Tesla stock prediction where the BI-LSTM model [1] has been choose as state-of-the-art model comparison from earlier research.
- With smaller MSE and RMSE results, Since MSE and RMSE both show the cube root of the difference in average between the model's predicted value and the actual value, where the value increases as the difference in value increases, the best model can more correctly predict Tesla's stock closing price in the future [40].
- If the R^2 is high, it may be concluded that the regression model's performance is accurate and good value is close to 1, the better and more accurate the resulting model [41].

Considering Tesla's market capitalization, it is possible to anticipate stock market values, this study employed the SVR, KNNR, DTR, and RFR models. In this study, it was also examined if machine learning might be used to the Tesla stock market. However, up until recently, there wasn't a lot of study done on Tesla Stock's market capitalization forecasting using machine learning techniques. This study's goal is to offer the best machine learning model for predicting Tesla's stock. This study includes a comparison between SVR, KNNR, DTR, RFR as well as the optimal technique for stock price forecasting with earlier studies. The comparative study's findings beat previous stock price prediction investigations, and its top finding enabled investors to select a new strategy based on the predicted value.

The rest of the study is formed as the following part: The second part shows the related work on time series prediction. The third part displays the theoretical foundations of the study. The fourth part discusses the proposed method framework. The fifth part shows the experimental results and discussion. Finally, the last part displays the conclusion of the study and possible future studies.

2. Literature Review

Stock prediction in terms of predicting prices is a topic that has been researched quite a lot in the last decade. Due to that, stock prediction of stock prices has become a very popular topic between finance and computer science [23] where

in recent times, researchers have concentrated on issues such the usage of machine and deep learning, the function of artificial intelligence, and related themes. financial industry [24]. Various kinds of methods and techniques have been found and implemented so that quite a lot of developments have occurred in this research topic. Deep Learning methods and Machine Learning are needed to be able to solve existing problems, especially in the current technological era, there is always a way for investors to be able to predict stock prices for profit. Here is the research that has existed to solve stock price prediction problem.

Bhattacharjee & Bhattacharja (2019) examines stock price predictions by comparing statistical approaches, Neural Network models and machine learning models. This study uses the ANN, Exponential Smoothing, WMA, SMA as a statistically representative approach. SLR, RF, SVM, Lasso Regression, KNN, Ridge Regression as machine learning representatives. LSTM, MLP, and SLP as Deep Learning representatives. From this study, it was found that the LSTM produces the MAPE and lowest MSE of all other models in 2 different datasets [35].

Agrawal (2021) conducted a study to predict stock prices using deep learning based non-linear regression and machine learning. This study uses Linear Regression (LR) as a representative of machine learning and ANN as a representative of deep learning based non-linear regression. The ANN model used 5 layers with 1 input layer, 3 hidden layers, and 1 output layer. From this study, it was found that ANN produced the lowest RMSE of the other two models in 3 different datasets [36].

Shah et al. (2021) did a study to use Deep Learning to forecast stock prices. This study uses Bi- LSTM as a representative of Deep Learning. From this study, it was found that the Bi-Directional LSTM resulted in better Accuracy compared to the ordinary LSTM in 2 different datasets. So that it was concluded that the BI-LSTM model was better than the ordinary LSTM model [1].

Rubi et.al. (2022) conducted a study to predict stock prices using deep learning. This study uses (ARIMA) Auto Regressive Integrated Moving Average and ANN as a representative of deep learning. From this study, it was found that ANN produced the lowest RMSE of the other models in Dhaka Stock Exchange (DSE) dataset [16].

From Table 1, it can be seen that Research by Rubi et al. (2022) [16] with the Artificial Neural Networks (ANN) model is the best. But research from [38] compare LSTM for predicting stock prices using ANN and ARIMA models, the results indicate that LSTM model has the best predictive power. Also, from research of Shah et al. (2021) [1], the result show that Bi-LSTM produces best result than LSTM. From the previous research, it was shown many methods were used to predict tesla stock data. The best model to predict tesla stock is Bi-LSTM with 0.05772 Mean Squared Error (MSE). Therefore, the Bi-LSTM model from Shah et al. (2021) [1] is implemented as a comparison.

Table 1. Previous Research Table

Author	Model Proposed	Evaluation Result	Dataset
Bhattacharjee & Bhattacharja (2019) [35]	"Single Layer Perceptron (SLP), Multi-Layer Perceptron (MLP),Long Short Term Memory (LSTM), Support Vector Machine, Random Forest (RF), K-Nearest Neighbors (KNN), Ridge Regression, Lasso Regression, and Simple Linear Regression".	LSTM produces the MAPE and lowest MSE, namely 3.7708 and 0.6993% for the Tesla Dataset and 3.5248 and 0.7013 % for the Apple dataset.	Tesla Stock and Apple Stock Dataset
Agrawal (2021) [36]	"Linear Regression (LR) and Artificial Neural Networks (ANN)".	ANN produces the lowest Root Mean Squared Error (RMSE), which is 3.25 for Google dataset and 0.25 for Apple dataset and 1.24 for Tesla dataset.	Tesla Stock, Google Stock, and Apple Stock Dataset
Shah et al. (2021) [1]	"Bi-Directional Long Short-Term Memory (Bi-LSTM)".	BI-LSTM produces 98% better accuracy than LSTM with 78% accuracy.	Tesla Stock and Citigroup Stock Dataset
Rubi et al. (2022) [16]	"Artificial Neural Networks (ANN), Auto Regressive Integrated Moving Average (ARIMA)" .	ANN produces Root Mean Squared Error (RMSE) 0,0303001 better than ARIMA model with 0,1184182	Dhaka Stock Exchange (DSE) Dataset

3. Materials and Methods

To achieve the research goal, a number of underlying processes must be finished, including data gathering, data processing, model training and fitting dataset (SVR, KNNR, DTR, RFR and BI-LSTM model implementation), Model Evaluation, and Testing. An outline of the proposed study's operational process is showed in Figure 1.

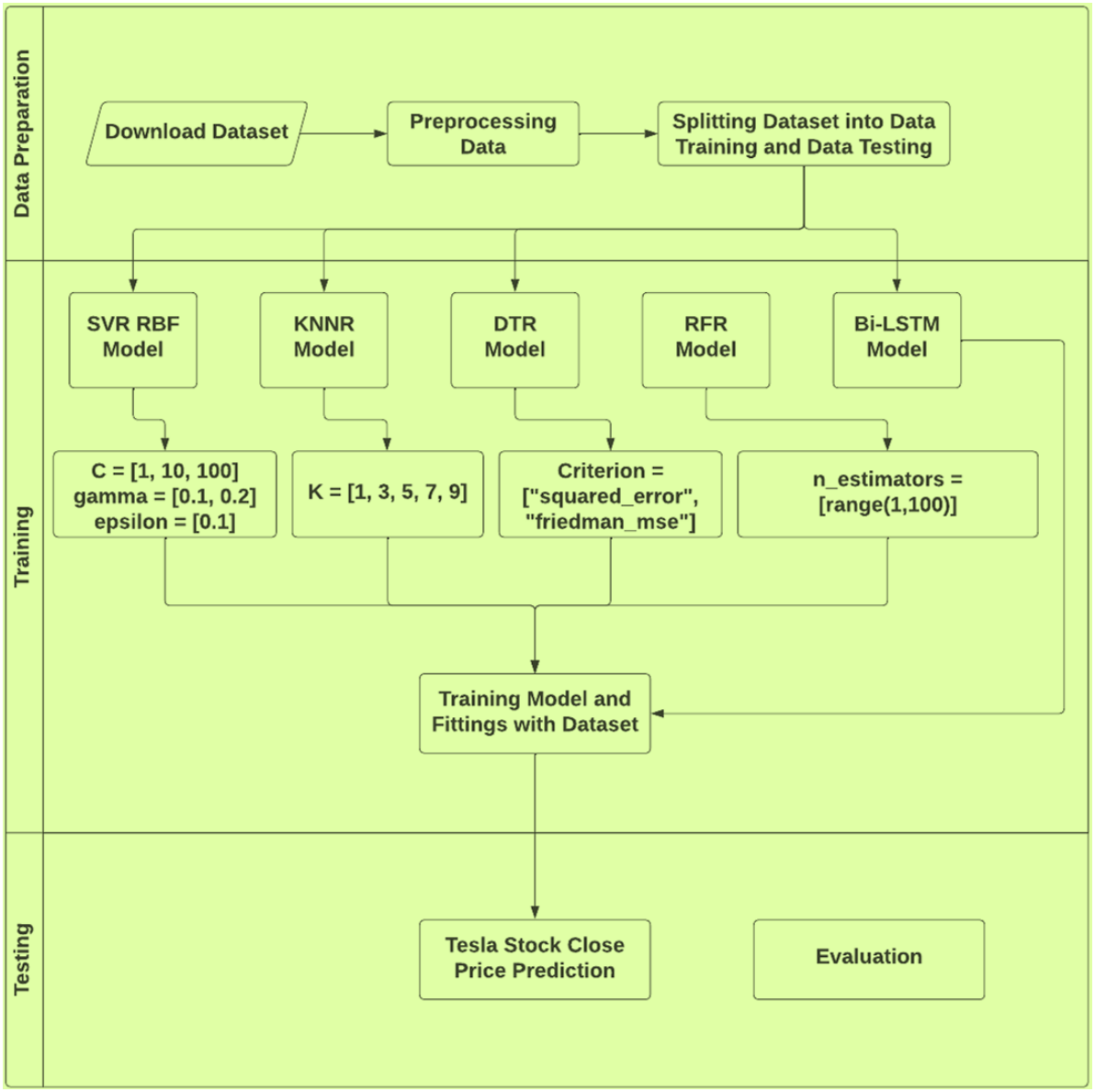

Figure 1. Research steps of stock prediction model

The data used in this study was collected from https://finance.yahoo.com/ for open-source Tesla shares which include several attributes such as low, high, open, and close. After getting the dataset, the next step is to ensure the column that has a String value, in this case Time, has a label to prepare before the testing and training process is carried out. Data preparation is done to separate inappropriate data and align all data by making all data the same size in order to decrease the possibility that errors will arise during the training phase. The following step is to divide the data set into two groups, testing data and training data.

In this study, training process using the machine learning model proposed were RBF Kernel SVR, DTR, RFR, and KNNR, which will later be searched for the best MSE, RMSE and R^2 Errors and carried out trials of the BI-LSTM model by Shah et al. (2021) [1]. The implementation of the machine learning and BI-LSTM regression models will be applied to the Tesla stock data set. To try out each Machine Learning model using the existing library on the Sklearn platform and BI-LSTM using the existing library on the Keras platform which runs via the Jupyter Notebook format and running on Google Colab.

3.1. Data Collection

The data used in this study uses the Tesla stock data set collected from *https://finance.yahoo.com/* containing the history of Tesla stock market from January 2010 - February 2022. In this data set there are 2,931 data including attributes such as open, close, high, and low which example can be seen in Table 2. This data collection has the advantages of time series model data, which makes predictions that are more accurate than those made using text models or mixed models [42]. Also, from the research by Weng et al. (2022) [43] using Tesla dataset from yahoo finance showed that there is an evident trend upward started after 2022, with recurrent seasonal variations of Tesla closing price tended to

reach the highest in January, with the lowest in July. Therefore, the dataset comprises seasonal or trending features, it is important to utilize the appropriate prediction technique in line with the characteristics of the data.

Table 2 is a description of the data downloaded from Yahoo Finance (https://finance.yahoo.com/) where the Tesla stock dataset has 7 attributes with the following explanation:

- *Volume*

  Represent the number of shares traded on that day usually in number of shares.

- *Close*

  Represent the closing price of shares opened in one trading day.

- *Low*

  Represent the lowest stock price that has ever occurred on that day.

- *High*

  Represent the highest stock price that has ever occurred on that day

- *Open*

  Represents the opening price of the stock when it opened on that day

- *Date*

  Represent the date the stock price transaction took place, because this dataset is in the form of historical stock price data, all transactions are recorded daily.

- *Adjusted Close Price*

  Represent the closing share price that has been adjusted for corporate actions by the company on that day.

Table 2. Example of the dataset

Date	Open	High	Low	Close	Adj Close	Volume
2010-06-29	3.800000	5.000000	3.508000	4.778000	4.778000	93831500
2010-06-30	5.158000	6.084000	4.660000	4.766000	4.776000	85935500
2010-07-01	5.000000	5.184000	4.054000	4.392000	4.392000	41094000
2010-07-02	4.600000	4.620000	3.742000	3.840000	3.840000	25699000
2010-07-06	4.000000	4.000000	3.166000	3.222000	3.222000	34334500
…	…	…	…	…	…	…
2022-02-10	908.369995	943.809998	896.700012	904.549988	904.549988	22042300
2022-02-11	909.630005	915.960022	850.700012	860.000000	860.000000	26492700
2022-02-14	861.570007	898.880005	853.150024	875.760010	875.760010	22515100
2022-02-15	900.000000	923.000000	893.380005	922.429993	922.429993	19095400
2022-02-16	914.049988	917.659973	901.219971	902.000000	902.000000	9167269

3.2. Dataset Splitting

The distribution of datasets is carried out 20% for testing data and 80% for training data based on several studies on stock predictions such as the prediction of Bank Central Asia (BCA) stock prices by Manurung et al. (2018) [44]. 20% testing, stock price predictions of Bank of Mandiri and Bank of Central Asia (BCA) during the COVID-19 pandemic by Budiharto (2021) [45] data separation by 80% training and 20% testing, FREN stock price prediction by Chrysmien & Jayadi (2022) [46] data separation is carried out by 80% training and 20% testing. The distribution of data for training is more in number than data for testing so that machine learning models are more trained so that predictions from the model are more optimal [47]. The training data is used to improve the performance of the RFR, SVR, KNNR, and DTR models and is tested against testing data in determining the best model. Testing is done with data testing because the hallmark of machine learning is the learning and training process, as a result, data must be studied, also known as training data, and tested, also known as testing data [47]. So, the details of dataset splitting can be seen in Table 3.

Table 3. Dataset Splitting Details

Dataset Part	Percentage	Total Data
Training Data	80%	2344
Testing Data	20%	587
Total	100%	2931

3.3. Pre-Processing Data & Data Normalization

After getting and splitting the dataset, the next step is pre-processing stage such as checking for missing values from the dataset, the "Date" column with string data type will be divided into three parts and labeled “Day”, “Month”, and “Year” with integer data type (Table 4). The splitting process is intended so that the split "Date" column can be read during the training [48]. After that, the data is normalized to create time series data with a normal or nearly normal distribution, with the same range of values for each input model [49]. We use the Min-Max normalization method because Maxi-Min normalization is the mostly common data normalization method and is widely used compared to other methods such as natural logarithm, Z-score normalization, and decimal scaling [49].

Table 4. Example of data after pre-processing normalization and min-max scaling

Day	Month	Year	High	Open	Low	Close	Volume
0.933333	0.454545	0.0	0.001350	0.000465	0.000422	0.001319	0.306605
0.966667	0.454545	0.0	0.002224	0.001568	0.001371	0.001309	0.280640
0.000000	0.545455	0.0	0.001498	0.001439	0.000871	0.001004	0.133184
0.033333	0.545455	0.0	0.001043	0.001114	0.000614	0.000554	0.082560
0.166667	0.545455	0.0	0.000543	0.000627	0.000140	0.000051	0.110956
…	…	…	…	…	…	…	…
0.300000	0.090909	1.0	0.758355	0.735181	0.736162	0.734779	0.070535
0.333333	0.090909	1.0	0.735898	0.736205	0.698271	0.698463	0.085170
0.433333	0.090909	1.0	0.722125	0.697169	0.700289	0.711310	0.072090
0.466667	0.090909	1.0	0.741575	0.728383	0.733428	0.749354	0.060844
0.500000	0.090909	1.0	0.737269	0.739795	0.739886	0.732700	0.028197

3.4. Stock Prediction

Stock itself is a piece of paper that signifies the owner as the owner of the company issuing the stock [3]. Some of the shares are publicly sold for public ownership which are usually referred to as public stock. The public can buy stock as a form of investment in the company. As a result, investors always do fundamental and technical research to be able to make stock forecasts and sell and buy stocks at the appropriate times to maximize profits [10]. Stock prices are difficult to do because of their non-linear, non-stationary, dynamic, and irregular nature [20]. According to Thakkar & Chaudhari (2021) [12], stock prices can be predicted through 2 main types of analysis, namely fundamental and technical analysis. Fundamental analysis bases its predictions on news information, company conditions, and macroeconomic factors such as dividend yield, return on equity, P/E, and EBITDA, while technical analysis based on its predictions mathematical indicators built on the history of the previous stock price time series [14].

3.5. Candlestick Data

Candlestick is a unit representation of stock price movements that contain open, close, high, and low at a certain time [17]. Open is the price of the first transaction that occurs in that time frame. Close is the price of the last transaction that occurred in that time range. High is the highest transaction price that has occurred in that time frame. Low is the lowest transaction price that has occurred in that time frame [50].

3.6. Support Vector Regression (SVR)

SVR is a regression algorithm that has the same principle as the SVM which is used as a class separation algorithm and tries to find a line that classifies two classes by expanding the dimensions of the data into multidimensional space to find a dividing line/ hyperplane [51]. SVR uses the SVM principle [52] and creates a function that is able to map data into a hyperplane on the training sample from the regression problem. The SVR defines a line within a predefined boundary (called a decision boundary) where this limit acts as an acceptable error to fit the data.

According to Wu et al. (2019) [53], Let S = $[(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)]$ represent a set of n samples, where each x_n is an input vector and y_n is the output target value, so the decision function of the SVR model can be equation as:

$$f(x) = w \cdot \phi(x) + b \tag{1}$$

where x is the vector of the input data for model's, W is the vector of function parameters, and b is the vector of function parameters. It is not linear to map the function (x). The SVR algorithm's next goal is to find the optimum values of w, b, and a few parameters of $\phi(x)$. By resolving the following optimization issues, it is possible to arrive at the SVR algorithm's optimal solution:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2} w^2 + C \sum_{i=1}^{n} (\xi_i + \xi_i^*) \tag{2}$$

$$here\ refers\ to \begin{cases} y_i + (w \cdot \phi(x) + b) \leq \varepsilon + \xi_i \\ (w \cdot \phi(x) + b) - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0, \quad i = 1, \ldots, n \end{cases} \tag{3}$$

where ξ_i and ξ_i^* quantify training mistakes, and C is a coefficient constant of establishes the magnitude of losses when training errors occur.

SVR can also accept more than one attribute input using Lagrange Multipliers [53], and the multiple variable optimization problem in SVR can be rewritten as:

$$\min_{a,a^*} \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} (a_i - a_i^*)(a_j - a_j^*) K + \varepsilon \sum_{i=1}^{n} (a_i + a_i^*) - \sum_{i=1}^{n} y_i (a_i - a_i^*) \tag{4}$$

$$where\ refers\ to \begin{cases} \sum_{i=1}^{n} (a_i - a_i^*) = 0 \\ 0 \leq a_i, a_i^* \leq C, i = 1, \ldots, n \end{cases} \tag{5}$$

where a_i and a_i^* are Lagrange Multipliers; K is a kernel function, which can be calculated as $K = K(i,j) = \phi(x_i)^T \phi(x_j)$ and commonly used kernel functions in SVR include Linear in Equation 6, Radial Basis Function (RBF) in Equation 7, and Sigmoid in Equation 8, shown as follows:

- Linear Kernel

$$K(i,j) = x_i \cdot x_j \tag{6}$$

- RBF Kernel

$$K(i,j) = \exp\left(-\gamma \left| x_i - x_j \right|^2\right) \tag{7}$$

- Sigmoid Kernel

$$K(i,j) = tanh\left(\gamma\left(x_i \cdot x_j\right) + r\right) \tag{8}$$

where 'γ' and 'r' are kernel parameters. represents the RBF parameter structure and Sigmoid and r represents the residue. Therefore, the Support Vector Regression (SVR) decision function in equation 9 can be expressed as:

$$f(x_i) = \sum_{i=1}^{n} (a_i - a_i^*) K(x_i, x_k) + b \tag{9}$$

3.7. K-Nearest Neighbor Regression (KNNR)

KNNR is a regression algorithm that uses KNN concept which utilizes the similarity of features between existing data points to predicted of new data points bases on how similar the point is to existing data points [54]. KNNR works like other regression models in that it uses the training set object and the corresponding value to estimate the value of the new object by calculating the mean k-Nearest Neighbor of the training set [55].

The K-closest neighbors are used to decide the value that should be allocated to the new data point after calculating the distance between it and the old data point (K denoting the number of neighbors) [56]. To determine the separation between data points, a number of methods are known. The three most popular approaches are Hamming distance, Euclidean, and Manhattan; in this case, Euclidean distance will be employed in the Equation 9 expressed as [55]:

$$d(x_i, x_{new}) = \sqrt{\sum_{j=1}^{d} W_j \left(x_{new.j} - x_{i.j}\right)^2} \tag{10}$$

where q represents the new data point, and p represents the existing data point.

Then arrange the distances between each point in the data set in ascending order. Then, select the k points closest to x and insert it into the set of nearest neighbors, k and find the estimated value of the response function. Therefore, the decision function of the KNNR in Equation 11 can be expressed as:

$$\hat{y}_{new} = \frac{\sum_{i=1}^{k} K(x_{new}, x_{(i)}) y_i}{\sum_{i=1}^{k} K(x_{new}, x_{(i)})} \quad (11)$$

3.8. Decision Tree Regression (DTR)

Recursive partitioning is achieved via the Decision Tree (DT) from the root node (also known as First Parent), and DTR is a regression approach that makes use of this similar concept. Each node may be divided into right and left child nodes using DTR. Then, these nodes might be further divided, becoming themselves the parent nodes of the offspring nodes [57]. DTR works like other regression models where it uses the training set object and the appropriate value to estimate the value of the new object by calculating the resulting Information Gain starting from the root that produces the largest (IG) [57]. To separate the nodes, you must first define the objective function you want to optimize using the DT algorithm so that equation can be used to specify the objective function information acquisition at any node separation in Equation 12 as follows [57]:

$$IG(D_p, f) = I(D_p) - \left(\frac{N_{left}}{N_p} I(D_{left}) + \frac{N_{right}}{N_p} I(D_{right}) \right) \quad (12)$$

where f is the feature for splitting, D_p, D_{left}, and D_{right} are the data sets of child nodes and the parent, I is the size for the splitting criteria, N_{right} and N_{left} are the number of samples in the child nodes, N_p is the total number of samples on the parent node.

3.9. Random Forest Regression (RFR)

RFR is a regression algorithm that used the concept of Random Forest (RF) where its a regression technique that is simple for distributed processing, highly effective, and capable of handling high-dimensional data without feature filtering [58]. RFR works by building several Decision Trees during the training time and issuing the class average as a prediction of all Trees [59]. Where Node Importance is calculated for each DT using Gini Importance, with the assumption that there are only two child nodes (binary tree) [60]:

$$ni_j = W_j C_j - W_{left(j)} C_{left(j)} - W_{right(j)} C_{right(j)} \quad (13)$$

where $right(j)$ is the child node of the right side of the j-th node, $left(j)$ is the child node of the left side of the jth node, C_j is the value of the j-th node, W_j is the Weight value of the sample that reaches the j-th node, ni_j is the value of the importance of the j-th node. So the importance level value for each feature in the Decision Tree is then calculated as [60]:

$$fi_i = \frac{\sum_{j:node\ of\ -j\ split\ to\ feature\ i\ of} ni_j}{\sum_{k\ \epsilon\ all\ node} ni_k} \quad (14)$$

In order to simplify it to a number between 0 and 1, it may then be normalized by dividing by the total of all significant values of each characteristic [60]:

$$normfi_i = \frac{fi_i}{\sum_{j\ \epsilon\ all\ feature\ from} fi_j} \quad (15)$$

Thus, the average of all trees is used to generate the Random Forest algorithm. Calculating and dividing the total number of trees by the number of significant feature values in each tree results in the decision function of the Random Forest in Equation 16 can be written as [60]:

$$RFfi_i = \frac{\sum_{j\ \epsilon\ all\ tree} normfi_{ij}}{T} \quad (16)$$

where $RFfi_i$ is the value of the i-th featuring calculate from all trees in the RF model, $normfi_{ij}$ is the normalized value of the i-th feature from the j-th tree, and T is the total number of trees.

3.10. Bi directional-Long Short Term Memory (Bi-LSTM)

Paliwal & Schuster developed the Bi-LSTM to train a network with both historical and upcoming input data sequences [61]. Two interconnected layers are used to process the input data [62]. Bi-directional LSTM uses a limited sequence depending on the context of items in the future and past to predict or tag the sequence of each element. This results from the simultaneous operation of two LSTMs, one moving from right to left and the other from left to right. Composite output refers to a target signal's prediction. This approach has shown to be quite helpful. Equations 17 and 18 are used to establish a bidirectional LSTM's forward function with inputs of H as the number of hidden units and L units. The bidirectional LSTM structure is shown in Figure 3. The Bi-LSTM network stores two values in its hidden layer. The forward calculation involves A, and the reverse calculation uses A transposed. The values of A and A transposed determine the final output value, y.

$$a_h^t = \sum_{l=1}^{L} x_l^t w_{lh} + \sum_{h',t>0}^{H} b_{h'}^{t-1} w_{h'h} \quad (17)$$

$$a_h^t = \theta_h(a_h^t) \quad (18)$$

3.11. Training and Fitting Model

Training process will be carried out where daily Tesla stock price data with Close as the output result and the Day, Month, Year, Open, High, Low, and Volume columns as input to predict output. Input selection based on research by Shastri et al. (2019) [63] to predict BSE stocks using ANN with input sentiment score, volume, low of the day, high of the day, opening price and Date. This study does not use sentiment analysis to estimate the sentiment score because the current sentiment analysis and lexicons cannot predict the stock market very accurately so it is limited in its ability to provide light on the relationship between emotion and stock prices [64].

For the training process in this study using the python open source library Sklearn. Sklearn already has a library that will be used to regress Tesla stock data. Each model will be tuning parameters based on research conducted by Ertuğrul et al. (2019) [65] regarding "Performance tuning for machine learning-based software development effort prediction models" based on MAR values. Where the best tuning parameters for the SVR model are "C, Epsilon and Kernel", the KNNR model with the value of "K", the DTR model with the "criterion" type, and the RFR model with the "n_estimator" value.

For this type of kernel, the SVR model will use the RBF kernel because the RBF kernel produces the best results compared to linear, sigmoid, and polynomial kernels [66]. In addition to the C and Epsilon parameters, the gamma parameter will also be tuned as a conditional parameter [67]. The Epsilon, gamma, and values of C are based on research conducted by Jassim et al. (2022) [68] regarding "The Analytical Performance of Support Vector Regression and Artificial Neural Networks" in predicting the generation of municipal waste where the values of C: [1, 10, and 100], the gamma value as small as possible is [0.1 or 0.2] to prevent overfitting, and the epsilon value is [0.1] because it adjusts the predicted value distance from the epsilon tube to the actual value.

The K value in the KNNR model is based on research conducted by Cui et al. (2020) [69] regarding efficient modification of the KNN model for antenna optimization and design where the value that is often used in many cases is $K \leq 10$ to prevent overfitting. A K value that is too small has an effect on overfitting and a K value that is too large for underfitting, so that an odd K value must be chosen, whether it is small or large [70]. Therefore, the K values used are [1, 3, 5, 7, 9].

For the criterion type on the DTR model, it is based on research conducted by Anmala & Turuganti (2021) [71] regarding water quality prediction using Decision Tree and Extreme Learning Machine (ELM) which shows that tuning the criterion with 'squared_error' and 'friedman_mse' produces very good performance for the model.

The n_estimator value in the Random Forest Regression (RFR) model is based on a study conducted by Contreras et al. (2021) [72] regarding "Influence of Random Forest Hyperparameterization on Short-Term Runoff Forecasting in an Andean Mountain Catchment" where the number of "n_estimators" who have the greatest influence on the performance of the Random Forest model between values 0 < "n_estimators" < 100 because the greatest improvisation occurred at the limit of that value. The model parameter and parameter value that was used in this study can be seen in table 5 below.

Table 5. Model Boundary Parameter and Parameter Value

Model	Parameter Used	Parameter Value
"Support Vector Regression (SVR)"	[C, Gamma, Epsilon]	C = [1, 10, 100]; Gamma = [0.1, 0.2]; Epsilon = [0.1]
"K-Nearest Neighbor Regression (KNNR)"	[K Value]	[1, 3, 5, 7, 9]
"Decision Tree Regression (DTR)"	[Criterion Type]	['squared_error','friedman_mse']
"Random Forest Regression (RFR)"	[n_estimators]	[range(1,100)]
"Bi directional-Long Short-Term Memory (Bi-LSTM) [1]"	[Input Layer, Subsequent Layer (2 Layer), and Output Layer]	Input Layer: Sigmoid; Subsequent Layer: Sigmoid (First Layer); Tanh (Second Layer); Output Layer: Tanh (for conversion); Sigmoid (combined output)

Where Training is run on a computer system with specifications as in Table 6.

Table 6. Google Colab Specifications for Model Training Process

No.	Hardware	Specification
1	Processor	Intel(R) Xeon(R) CPU @ 2.20GHz x 2
2	Memory	12GB
3	GPU Card	Tesla K80 with 12GB GDDR5 VRAM

The training process can be done in two ways, namely using the using Graphic Processing Unit (GPU) computing and computing capabilities of the Central Processing Unit (CPU). To shorten the training time in this research, we will utilize the parallel capabilities of the 2496 CUDA Core found on the Google Colab GPU to shorten the maximum training time.

3.12. Performance Evaluation

Evaluation in this study will be carried out every time the training process on the computer is completed and test the results of the training model with data testing. What is done at the evaluation stage is to measure the error value of each architecture used due to the performance of the regression model is representative in the form of an error value [41]. The error value alone is not enough to judge whether the performance of a model is good for the reason there is no standard for a small error value, it can be said that the model's performance is good and therefore the Coefficient of Determination (R^2) is needed as a standard statistical measure to evaluate the regression model for the reason that there is a range the value of [0,1] where if Coefficient of Determination (R^2) is close to 1, the better and more accurate the resulting model [41]. To measure the error value, we used evaluation parameters in the form of RMSE and MSE. The choice of RMSE and MSE because both metrics have a monotonous relationship with R^2 [41]. The explanation of the three evaluation parameters is as follows:

- Mean Squared Error (MSE)

  MSE is a method for evaluating the prediction results in a way that is squared error, if the error is large, the greater the MSE generated. The MSE formula is defined as follows [73]:

  $$MSE = \sum \frac{(Y'-Y)^2}{n} \quad (19)$$

  where Y' is the predicted value of each model, Y is the actual data value, and n is the number of data.

- Root Mean Squared Error (RMSE)

  The sum of squared errors, also known as the discrepancy between the actual (actual) value and the anticipated value, is known as the RMSE [73]. If the error is large, the greater the RMSE generated. The RMSE formula is defined as follows [74]:

  $$RMSE = \frac{1}{n}\sum \sqrt{(Y'-Y)^2} \quad (20)$$

  where Y' is the predicted value of each model, Y is the actual data value, and n is the number of data.

- Coefficient of Determination (R^2)

  To evaluate how well the model fits the dependent and independent data it receives, the R^2 is utilized [75]. If R^2 is close to 1, the better and more accurate the resulting model is so that the R^2 formula is defined as follows [74]:

  $$R^2 = 1 - \frac{\sum (Y'-Y)^2}{\sum (Y_{average}-Y)^2} \quad (21)$$

  where Y' is the predicted value of each model, Y is the actual data value, and Y_{mean} is the average of Y' response values.

4. Results and Discussion

4.1. Main Findings of the Present Study

Each Machine Learning model used has a different model. The measurement of the model evaluation is by measuring the RMSE, MSE, and R^2, on a different data set from the training data set. If the RMSE and MSE values are low and the R^2 values is close to 1, then the model can be interpreted as good [41]. MSE, RMSE, and R^2 measurements include 112 Machine Learning models used in this study consisting of six SVR models in Table 7, five KNNR models in Table 8, two DTR models in Table 9, and ninety-nine RFR models that can be seen in Table 10.

Table 7. The forecast performance measures for SVR RBF model Testing Data

Proposed SVR RBF Model	MSE	RMSE	R^2
SVR with C =1 ; gamma=0.1; epsilon=0.1	0.006094	0.078067	0.855664
***SVR with C = 1 ; gamma=0.2; epsilon=0.1**	**0.005040**	**0.070993**	**0.880638**
SVR with C = 10; gamma=0.1; epsilon=0.1	0.006104	0.078125	0.855449
SVR with C =10 ; gamma=0.2; epsilon=0.1	0.005040	0.070993	0.880638
SVR with C = 100 ; gamma=0.1; epsilon=0.1	0.006105	0.078126	0.855450
SVR with C = 100; gamma=0.2; epsilon=0.1	0.005041	0.070994	0.880639

* Represents SVR best model result

Table 8. The forecast performance measures for KNNR model Testing Data

Proposed KNNR Model	MSE	RMSE	R^2
KNNR with K = *1	**0.000070**	**0.008380**	**0.998337**
KNNR with K = *3*	0.000090	0.009489	0.997867
KNNR with K = *5*	0.000123	0.011102	0.997081
KNNR with K = *7*	0.000169	0.013011	0.995991
KNNR with K = *9*	0.000181	0.013450	0.995715

* Represents KNNR best model result

Table 9. The forecast performance measures for DTR model Testing Data

0	MSE	RMSE	R^2
DTR with criterion = *"squared_error"	**0.000047**	**0.006903**	**0.998871**
DTR with criterion = *"friedman_mse"*	0.000058	0.007637	0.998618

* Represents DTR best model result

Table 10. The forecast performance measures for RFR model Testing Data

Proposed SVR Model	MSE	RMSE	R^2
RFR with *n_estimators = 1*	0.000034	0.005892	0.999178
RFR with *n_estimators = 2*	0.000032	0.005697	0.999231
RFR with *n_estimators = 3*	0.000037	0.006108	0.999116
…	…	…	…
…	…	…	…
…	…	…	…
RFR with *n_estimators = 87	**0.000029**	**0.005452**	**0.999296**
RFR with *n_estimators = 97*	0.000030	0.005501	0.999283
RFR with *n_estimators = 98*	0.000030	0.005507	0.999282
RFR with *n_estimators = 99*	0.000030	0.005513	0.999280

* Represents RFR best model result

As presented in Table 7, the SVR RBF model performance result is shown in terms of RMSE, MSE, and R^2. SVR RBF with C = 1, epsilon = 0.1 and gamma = 0.2, outperform other SVR model. Even though all of them is SVR RBF model tuned the SVR parameter which are C, epsilon, and gamma, the performance of SVR reveal that it has better capability to predict stock price and prevent overfitting.

As presented in Table 8, the KNNR model performance result is shown in terms of MSE, RMSE, and R^2. KNNR with K = 1 outperform other KNNR model. Even though all of them is KNNR model tuned the KNNR parameter which are K value, the performance of KNNR reveal that it has better capability to predict stock price. Besides the capability, it is also proving the research conducted by Cui et al. (2020) [69] and Fu et al. (2021) [70], K values that has been used which is [1, 3, 5, 7, 9] can prevent overfitting based on result from Table 8.

As presented in Table 9, the DTR model performance result is shown in terms of RMSE, MSE, and R^2. DTR with criterion = "squared_error" outperform other DTR model. Even though all of them is DTR model tuned the DTR parameter which are criterion type, the performance of DTR reveal that it has better capability to predict stock price. The results proved the research conducted by Anmala & Turuganti (2021) [71], Criterion Type that has been used which is 'squared_error' and 'friedman_mse' has a good performance based on result from Table 9.

As presented in Table 10, the RFR model performance result is shown in terms of RMSE, MSE, R^2, and Mean Squared Error (MSE). RFR with n_estimators = 87 outperform other RFR model. Even though all of them is RFR model tuned the RFR parameter which are n_estimators value, the performance of RFR reveal that it has better capability to predict stock price. Based on the result from Table 10, it is also proved that the research conducted by Contreras et al. (2021) [72] n_estimators values that has been used which is [range (1, 100)] have the greatest influence on the performance of the Random Forest model.

4.2. Comparison with Other Studies

The results of our study show that the values for predicting stock prices are fairly consistent across all analyses when compared to those of other studies. Some academics are making an effort to predict the stock price with their chosen technique. Table 11, Figures 2, and 3 contrasts our proposed work with state of the art of the earlier research about the

high Mean Squared Error (MSE) value of the model used. Diverse research focuses on certain goals, with some aiming to use the ANN model, traditional time series ARIMA model, LSTM model. And Bi-LSTM model on their data. Some researchers are seeking to forecast stock price using various stock data. To address the limitation of Tesla stock price prediction, this study employs an SVR, KNNR, DTR, and RFR model with a minimum error. One of the main outcomes of our work is accurate stock price predictions. The size of the market changed throughout the epidemic, rising, falling, or remaining stable. Another thing to think about is that any model's most challenging component is its most accurate forecast, and our RFR model exceeded state of the art of the previous work with a lower MSE score (0.000029), RMSE score (0.005452), and higher R^2 score (0.999296).

Table 11. The forecast error measures for selected RFR model

Reference	Dataset	Best Model	MSE	RMSE	R^2
Shah et al. (2021) [1]	Tesla Stock	*Bi-Directional Long Short-Term Memory* (BI-LSTM)	0.01386	0.11773	0.791263
Present Study	Tesla Stock	RFR with n_estimators = 87	**0.000029**	**0.005452**	**0.999296**

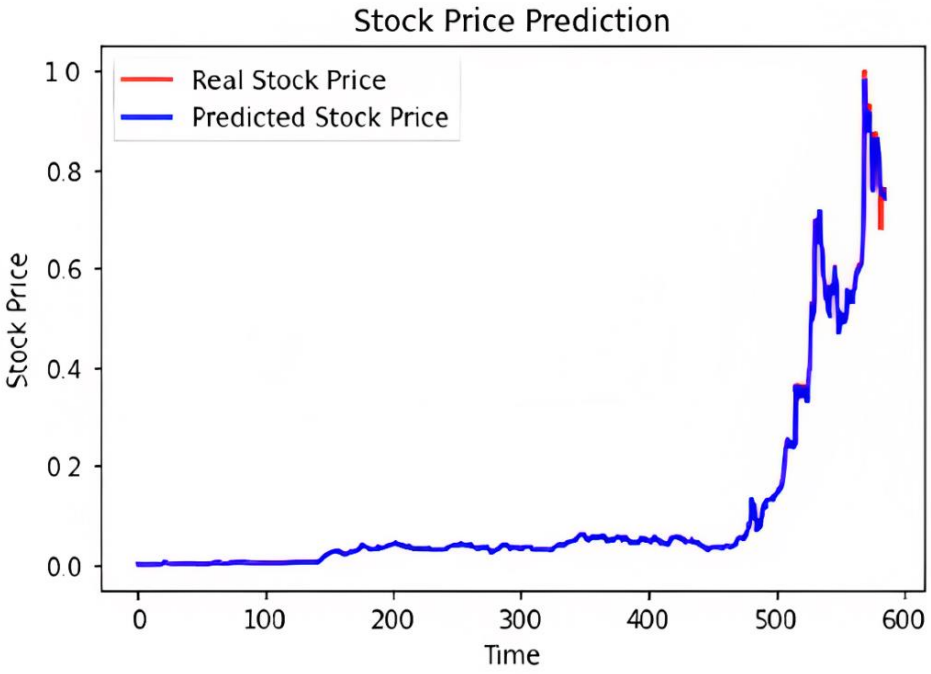

Figure 2. The forecasted pattern of RFR model

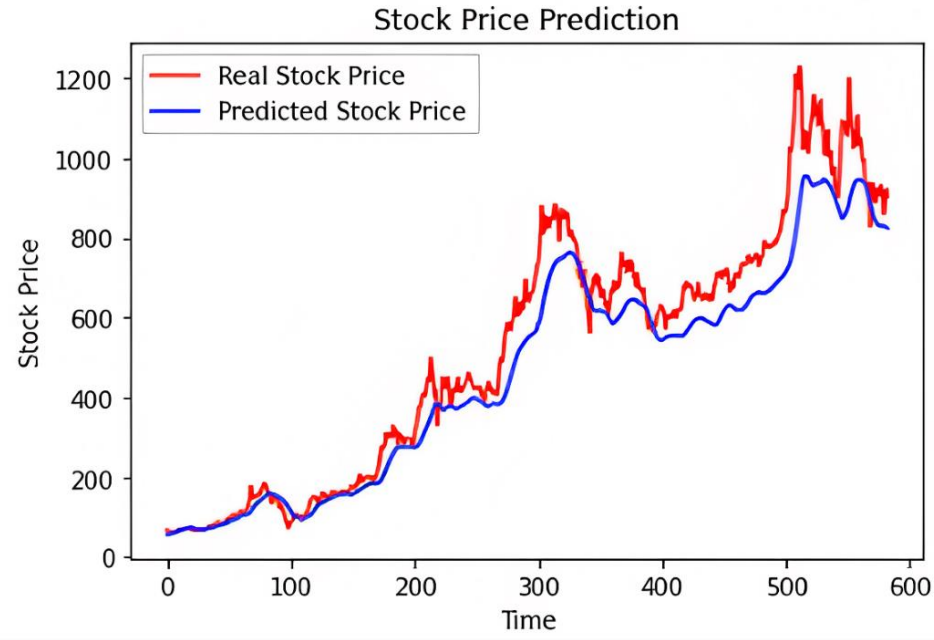

Figure 3. The forecasted pattern of Bi-LSTM model

4.3. Implication and Explanation of Findings

Looking at Figures 4 to 6, RFR with n_estimators = 87 have the highest performance from all the other model. As a result, unlike other model without bagging learning and ensemble approach, RFR with n_estimators = 87 has the highest predictive capability to the other model with the result of MSE, RMSE, and R^2 respectively has 0.000029, 0.005513, and 0.99928.

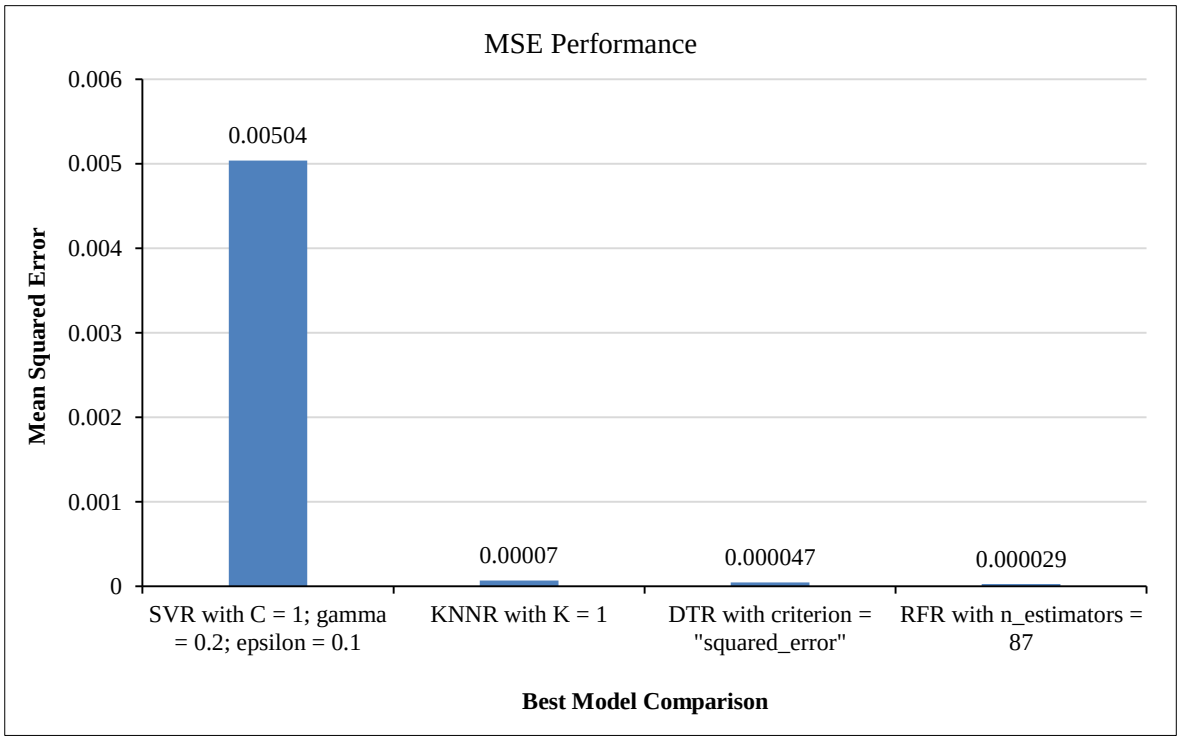

Figure 4. Performance comparison of the approaches in Mean Squared Error

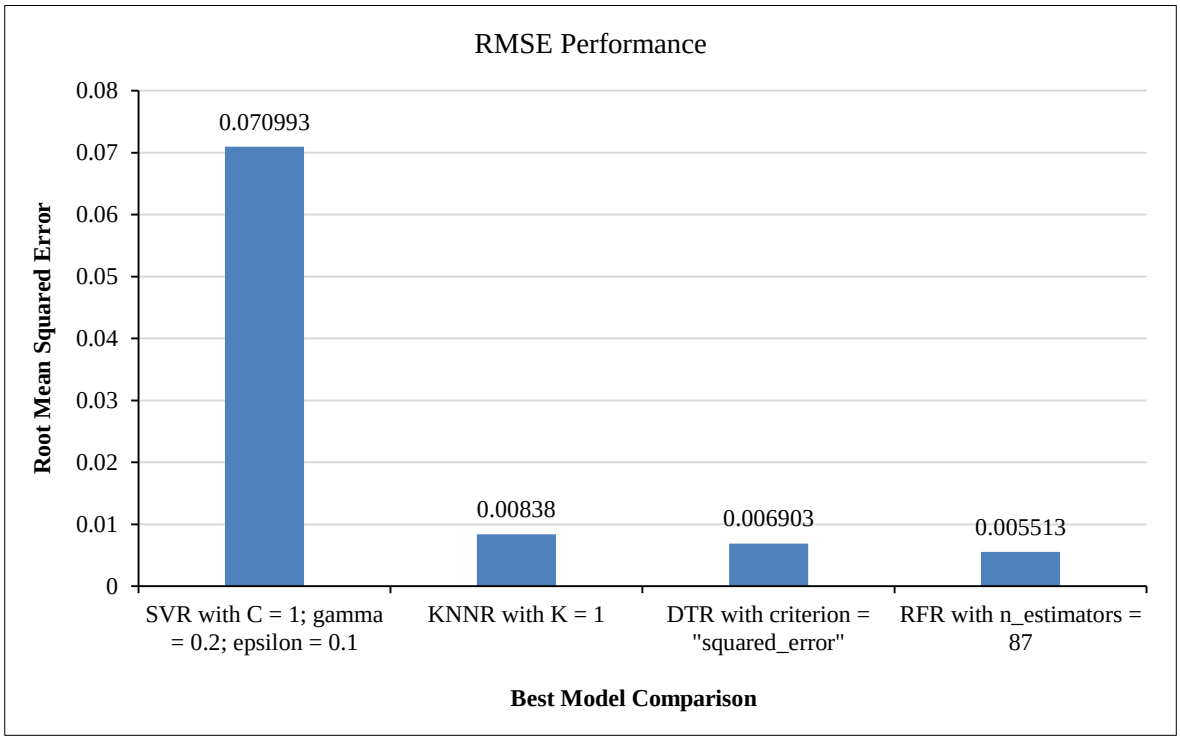

Figure 5. Performance comparison of the approaches in Root Mean Squared Error

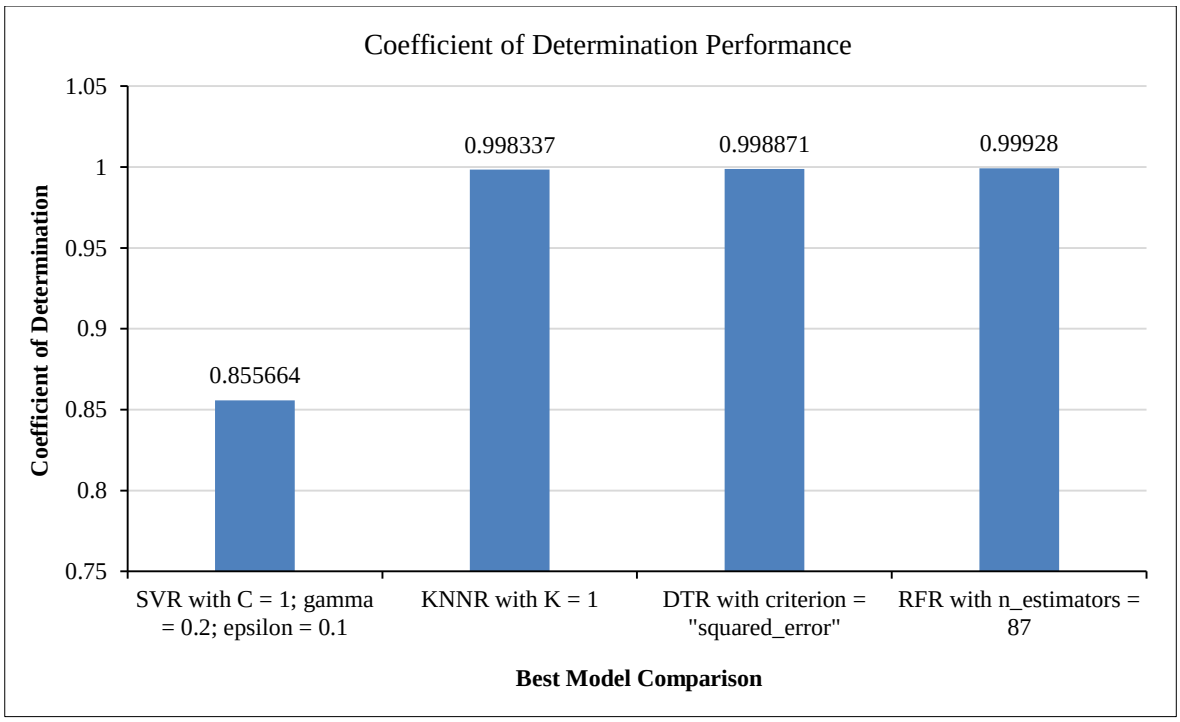

Figure 6. Performance comparison of the approaches in Coefficient of Determination

Regarding DTR with criterion = "squared_error", it has the second highest performance compared to another model. This proves that, compared to RFR, DTR with criterion = "squared_error" has lower performance in predicting Tesla stock. The performance of DTR with criterion = "squared_error" resulted in MSE, RMSE, and R^2 of 0.000047, 0.006903, and 0.998871, respectively.

Concerning KNNR with K = 1, it has the third highest performance compared to another model. It also has the second-lowest performance compared to another model. This showed that KNNR with K = 1 has a lower performance in predicting Tesla stock compared to RFR with n_estimators = 87 and DTR with criterion = "squared_error". The performance of KNNR with K = 1 resulted in MSE, RMSE, and R^2 respectively has 0.000070, 0.008380, and 0.998337.

Finally, for SVR with C = 1, gamma = 0.2, and epsilon = 0.1, it has the lowest performance compared to the other model. This proved that SVR with C = 1, gamma = 0.2, and epsilon = 0.1 has the lowest performance in predicting Tesla stock compared to another model. As a result, SVR with C = 1, gamma = 0.2, and epsilon = 0.1 has the lowest performance compared to other models with the results of 0.005040 MSE, 0.070993 RMSE, and 0.855664 Coefficient of Determination (R^2).

4.4. Strengths and Limitations

Based on the results above, RFR with n_estimators = 87 has the highest performance of all the other models. Unlike other models, the RFR employs a bagging learning technique that yields a sizable number of de-correlated trees [76] and utilizes the ensemble technique, where each RF tree is created from randomly selected subsets of the characteristics of the sample data, allowing some features to be used in decision-making [77].

On the other hand, SVR with C = 1, gamma = 0.2, and epsilon = 0.1 has the lowest performance compared to the other model. Due to those results, manually looking for the best parameter for SVR may not be the best result. So other studies used optimization algorithms such as Grid Search [78], Particle Swarm Optimization (PSO) [79], Bat Optimization Algorithm [80], and Genetic Algorithm (GA) [81] to achieve a better result and a more adaptable model to solve the dependency of SVR parameters.

5. Conclusion

In this study, Kernel RBF and SVR with C = [1, 10, 100], gamma = 0.1 or 0.2, and epsilon = 0.1, K-Nearest Neighbor Regression (KNNR) with five K values, namely from K = [1, 3, 5, 7, 9], Decision Tree Regression (DTR) with two criterion types, namely "squared_error" and "friedman_mse", and Random Forest Regression (RFR) with 100 "n_estimator" values, namely n_estimator = 1 to n_estimator = 99 model is proposed to predict close price of Tesla Stock. The best approaches of each model obtained by SVR with epsilon = 0.1, gamma = 0.2, and C = 1, KNNR with K = 1, DTR with criterion = "squared_error", and RFR with n_estimators = 87, which is the most optimal parameter obtained in this study. After the outcome of each best approach model is obtained, then compare the models with each other. From the result RFR, with n_estimators = 87 have the highest performance of all the other models.

Utilizing Tesla Stock to compare the efficiency of the suggested approach to other methods in https://finance.yahoo.com/. The outcome result of RFR with n_estimators = 87 and its comparison measured by R^2, RMSE, and MSE evaluation methods. The evaluation was performed to verify the effectiveness of RFR with n_estimators = 87 and compared to the state-of-the-art of an earlier study that used a Bi-LSTM model. The study from the experiment resulted in an RFR with n_estimators = 87 has better performance in predicting Tesla stock with a result of 0.005452 RMSE, 0.000029 MSE, and 0.999296 R^2.

This study showed that using RFR is more capable of obtaining optimal results compared to other models. By looking at MSE, RMSE, and R-Squared. Because of those results, the chosen best model, which is RFR with n_estimators = 87, gets results for MSE and RMSE that are smaller and an R^2 that is greater than the state-of-the-art of an earlier study that used the Bi-LSTM model. With smaller MSE and RMSE results, the best model can more correctly predict Tesla's stock closing price in the future. Also, with R^2 is high, the better and more accurate the resulting model.

However, this study only used a limited amount of data which is 2931 data and manually looked for the best parameter for each model. Due to those limitations, SVR with C = 1, gamma = 0.2, and epsilon = 0.1, it has the lowest performance compared to the other model. In a future study, a higher amount of dataset higher than this study (2931 data) and used optimization algorithm such as PSO, BA, GA, or other optimization algorithm can be experimented with to see if it could perform as well as this study showed. Those optimization algorithms used to look at the SVR best parameter. These are retained for further work.

6. Declarations

6.1. Author Contributions

Conceptualization, E. and S.; methodology, E.; software, E.; validation, E. and S.; formal analysis, E. and S.; investigation, E. and S.; data curation, S.; writing—original draft preparation, E. and S.; writing—review and editing, E. and S.; visualization, E.; supervision, S. All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

Publicly available datasets were analyzed in this study. This data can be found here: https://finance.yahoo.com/.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Acknowledgements

This study was conducted and supported under Bina Nusantara University, Indonesia.

6.5. Institutional Review Board Statement

Not applicable.

6.6. Informed Consent Statement

Not applicable.

6.7. Declaration of Competing Interest

The authors declare that there is no conflict of interests regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

7. References

[1] Shah, J., Jain, R., Jolly, V., & Godbole, A. (2021). Stock Market Prediction using Bi-Directional LSTM. 2021 International Conference on Communication Information and Computing Technology (ICCICT). doi:10.1109/iccict50803.2021.9510147.

[2] Pahwa, K., & Agarwal, N. (2019). Stock Market Analysis using Supervised Machine Learning. 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon). doi:10.1109/comitcon.2019.8862225.

[3] Alluch, J. B., Brown, L. V., & Sykes, T. (2018). The Complete Penny Stock Course Learn How to Generate Profits Consistently by Trading Penny Stocks. Millionaire Publishing LLC, Miami, United States.

[4] Akben-Selcuk, E. (2019). Corporate social responsibility and financial performance: The moderating role of ownership concentration in Turkey. Sustainability (Switzerland), 11(13), 1–10. doi:10.3390/su11133643.

[5] Statistik. (2021). Indonesia Stock Exchange. Available online: https://www.idx.co.id/id/data-pasar/laporan-statistik/statistik/ (accessed on March 2022). (In Indonesian).

[6] Barberis, N., Jin, L. J., & Wang, B. (2021). Prospect Theory and Stock Market Anomalies. Journal of Finance, 76(5), 2639–2687. doi:10.1111/jofi.13061.

[7] Suhadak, S., Kurniaty, K., Handayani, S. R., & Rahayu, S. M. (2019). Stock return and financial performance as moderation variable in influence of good corporate governance towards corporate value. Asian Journal of Accounting Research, 4(1), 18–34. doi:10.1108/ajar-07-2018-0021.

[8] Rustam, Z., & Kintandani, P. (2019). Application of Support Vector Regression in Indonesian Stock Price Prediction with Feature Selection Using Particle Swarm Optimisation. Modelling and Simulation in Engineering, 2019, 1–6. doi:10.1155/2019/8962717.

[9] Lien Minh, D., Sadeghi-Niaraki, A., Huy, H. D., Min, K., & Moon, H. (2018). Deep learning approach for short-term stock trends prediction based on two-stream gated recurrent unit network. IEEE Access, 6, 55392–55404. doi:10.1109/ACCESS.2018.2868970.

[10] Sutjipto, E., & Setiawan, W. (2021). Accuracy of dividend discount model and p/e ratio valuation in Indonesia stock exchange. SSRN Electronic Journal. doi:10.2139/ssrn.3865915.

[11] Idrees, S. M., Alam, M. A., & Agarwal, P. (2019). A Prediction Approach for Stock Market Volatility Based on Time Series Data. IEEE Access, 7, 17287–17298. doi:10.1109/ACCESS.2019.2895252.

[12] Thakkar, A., & Chaudhari, K. (2021). Fusion in stock market prediction: A decade survey on the necessity, recent developments, and potential future directions. Information Fusion, 65, 95–107. doi:10.1016/j.inffus.2020.08.019.

[13] Thanathamathee, P., & Sawangarreerak, S. (2022). Discovering Future Earnings Patterns through FP-Growth and ECLAT Algorithms with Optimized Discretization. Emerging Science Journal, 6(6), 1328–1345. doi:10.28991/ESJ-2022-06-06-07.

[14] Picasso, A., Merello, S., Ma, Y., Oneto, L., & Cambria, E. (2019). Technical analysis and sentiment embeddings for market trend prediction. Expert Systems with Applications, 135, 60–70. doi:10.1016/j.eswa.2019.06.014.

[15] Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2020). A systematic review of fundamental and technical analysis of stock market predictions. Artificial Intelligence Review, 53(4), 3007–3057. doi:10.1007/s10462-019-09754-z.

[16] Rubi, M. A., Chowdhury, S., Abdul Rahman, A. A., Meero, A., Zayed, N. M., & Islam, K. M. A. (2022). Fitting Multi-Layer Feed Forward Neural Network and Autoregressive Integrated Moving Average for Dhaka Stock Exchange Price Predicting. Emerging Science Journal, 6(5), 1046–1061. doi:10.28991/esj-2022-06-05-09.

[17] Liang, M., Wu, S., Wang, X., & Chen, Q. (2022). A stock time series forecasting approach incorporating candlestick patterns and sequence similarity. Expert Systems with Applications, 205, 117595. doi:10.1016/j.eswa.2022.117595.

[18] Jeon, S., Hong, B., & Chang, V. (2018). Pattern graph tracking-based stock price prediction using big data. Future Generation Computer Systems, 80, 171–187. doi:10.1016/j.future.2017.02.010.

[19] Fianto, B. A., Ridhwan, M. M., Shah, S. A. A., Faris, M., & Mohd Ruslan, R. A. H. (2022). Response of Financial Markets to COVID-19 Pandemic: A Review of Literature on Stock Markets. Emerging Science Journal, 7, 29–54. doi:10.28991/esj-2023-sper-03.

[20] Polanco-Martínez, J. M. (2019). Dynamic relationship analysis between NAFTA stock markets using nonlinear, nonparametric, non-stationary methods. Nonlinear Dynamics, 97(1), 369–389. doi:10.1007/s11071-019-04974-y.

[21] Rezaei, H., Faaljou, H., & Mansourfar, G. (2021). Stock price prediction using deep learning and frequency decomposition. Expert Systems with Applications, 169, 1–29. doi:10.1016/j.eswa.2020.114332.

[22] Wen, M., Li, P., Zhang, L., & Chen, Y. (2019). Stock market trend prediction using high-order information of time series. IEEE Access, 7, 28299–28308. doi:10.1109/ACCESS.2019.2901842.

[23] Jiang, W. (2021). Applications of deep learning in stock market prediction: Recent progress. Expert Systems with Applications, 184, 1–97. doi:10.1016/j.eswa.2021.115537.

[24] Wall, L. D. (2018). Some financial regulatory implications of artificial intelligence. Journal of Economics and Business, 100, 55–63. doi:10.1016/j.jeconbus.2018.05.003.

[25] Ma, A. (2022). Profitability of technical trading strategies under market manipulation. Financial Innovation, 8(1), 1–9. doi:10.1186/s40854-021-00304-7.

[26] Thomas, V. J., & Maine, E. (2019). Market entry strategies for electric vehicle start-ups in the automotive industry – Lessons from Tesla Motors. Journal of Cleaner Production, 235, 653–663. doi:10.1016/j.jclepro.2019.06.284.

[27] Huang, Y. (2019). A Potential Company or Not: the Analysis of Tesla. Proceedings of the 2019 4th International Conference on Financial Innovation and Economic Development (ICFIED 2019). doi:10.2991/icfied-19.2019.76.

[28] Claudia, O. (2019). Some Insights on the World's Most Innovative Companies and their Defining Characteristics. Studies in Business and Economics, 14(2), 88–104. doi:10.2478/sbe-2019-0027.

[29] Henrique, B. M., Sobreiro, V. A., & Kimura, H. (2019). Literature review: Machine learning techniques applied to financial market prediction. Expert Systems with Applications, 124, 226–251. doi:10.1016/j.eswa.2019.01.012.

[30] Hu, Z., Zhao, Y., & Khushi, M. (2021). A survey of forex and stock price prediction using deep learning. Applied System Innovation, 4(1), 1–30. doi:10.3390/ASI4010009.

[31] Sharma, D., Singla, S. K., & Sohal, A. K. (2021). Stock Market Prediction Using ARIMA, ANN and SVR. ICCCE 2020, Lecture Notes in Electrical Engineering, 698, Springer, Singapore. doi:10.1007/978-981-15-7961-5_100.

[32] Puspitasari, D. A., & Rustam, Z. (2018). Application of SVM-KNN using SVR as feature selection on stock analysis for Indonesia stock exchange. AIP Conference Proceedings. doi:10.1063/1.5064204.

[33] Hindrayani, K. M., Fahrudin, T. M., Prismahardi Aji, R., & Safitri, E. M. (2020). Indonesian Stock Price Prediction including Covid19 Era Using Decision Tree Regression. 2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI). doi:10.1109/isriti51436.2020.9315484.

[34] Ravikumar, S., & Saraf, P. (2020). Prediction of Stock Prices using Machine Learning (Regression, Classification) Algorithms. 2020 International Conference for Emerging Technology (INCET). doi:10.1109/incet49848.2020.9154061.

[35] Bhattacharjee, I., & Bhattacharja, P. (2019). Stock Price Prediction: A Comparative Study between Traditional Statistical Approach and Machine Learning Approach. 2019 4th International Conference on Electrical Information and Communication Technology (EICT). doi:10.1109/eict48899.2019.9068850.

[36] Agrawal, S. C. (2021). Deep learning based non-linear regression for Stock Prediction. IOP Conference Series: Materials Science and Engineering, 1116(1), 012189. doi:10.1088/1757-899x/1116/1/012189.

[37] Marchai, F. L., Martin, W., & Suhartono, D. (2021). Stock Prices Prediction Using Machine Learning. 2021 8th International Conference on Information Technology, Computer and Electrical Engineering (ICITACEE), Semarang, Indonesia. doi:10.1109/icitacee53184.2021.9617222.

[38] Ma, Q. (2020). Comparison of ARIMA, ANN and LSTM for Stock Price Prediction. E3S Web of Conferences, 218. doi:10.1051/e3sconf/202021801026.

[39] Muflikhah, L., & Haryanto, D. J. (2018). High Performance of Polynomial Kernel at SVM Algorithm for Sentiment Analysis. Journal of Information Technology and Computer Science, 3(2), 194–201. doi:10.25126/jitecs.20183260.

[40] Khandakar, A., Chowdhury, M. E. H., Kazi, M. K., Benhmed, K., Touati, F., Al-Hitmi, M., & Gonzales, A. S. P. (2019). Machine learning based photovoltaics (PV) power prediction using different environmental parameters of Qatar. Energies, 12(14), 1–19. doi:10.3390/en12142782.

[41] Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. PeerJ Computer Science, 7, 1–24. doi:10.7717/PEERJ-CS.623.

[42] Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: principles and practice. OTexts Publisher, Australia.

[43] Weng, Q., Liu, R., & Tao, Z. (2022). Forecasting Tesla's Stock Price Using the ARIMA Model. Proceedings of Business and Economic Studies, 5(5), 38–45. doi:10.26689/pbes.v5i5.4331.

[44] Manurung, A. H., Budiharto, W., & Prabowo, H. (2018). Algorithm and modeling of stock prices forecasting based on long short-term memory (LSTM). ICIC Express Letters, 12(12), 1277–1283. doi:10.24507/icicel.12.12.1277.

[45] Budiharto, W. (2021). Data science approach to stock prices forecasting in Indonesia during Covid-19 using Long Short-Term Memory (LSTM). Journal of Big Data, 8(1). doi:10.1186/s40537-021-00430-0.

[46] Chrysmien, & Jayadi, R. (2022). Comparison Stock Price Prediction Between Arima, Multiple Linear Regression and LSTM Models By Adding Stock Sentiment Analysis and Usd/Idr Fluctuation. Journal of Theoretical and Applied Information Technology, 100(4), 1158–1169.

[47] Haryadi, D., Hakim, A. R., Atmaja, D. M. U., & Yutia, S. N. (2022). Implementation of Support Vector Regression for Polkadot Cryptocurrency Price Prediction. International Journal on Informatics Visualization, 6(1–2), 201–207. doi:10.30630/joiv.6.1-2.945.

[48] Hamami, F., & Fithriyah, I. (2020). Classification of Air Pollution Levels using Artificial Neural Network. 2020 International Conference on Information Technology Systems and Innovation (ICITSI). doi:10.1109/icitsi50517.2020.9264910.

[49] Alawsi, M. A., Zubaidi, S. L., Al-Bdairi, N. S. S., Al-Ansari, N., & Hashim, K. (2022). Drought Forecasting: A Review and Assessment of the Hybrid Techniques and Data Pre-Processing. Hydrology, 9(7), 1–23. doi:10.3390/hydrology9070115.

[50] Hung, C.-C., & Chen, Y.-J. (2021). DPP: Deep predictor for price movement from candlestick charts. PLOS ONE, 16(6), e0252404. doi:10.1371/journal.pone.0252404.

[51] Mathew, A., Sreekumar, S., Khandelwal, S., & Kumar, R. (2019). Prediction of land surface temperatures for surface urban heat island assessment over Chandigarh city using support vector regression model. Solar Energy, 186, 404–415. doi:10.1016/j.solener.2019.04.001.

[52] Zhong, H., Wang, J., Jia, H., Mu, Y., & Lv, S. (2019). Vector field-based support vector regression for building energy consumption prediction. Applied Energy, 242, 403–414. doi:10.1016/j.apenergy.2019.03.078.

[53] Wu, J., Liu, H., Wei, G., Song, T., Zhang, C., & Zhou, H. (2019). Flash flood forecasting using support vector regression model in a small mountainous catchment. Water (Switzerland), 11(7), 1–16. doi:10.3390/w11071327.

[54] Afzal, P., Farhadi, S., Konari, M. B., Meigoony, M. S., & Saein, L. D. (2022). Geochemical anomaly detection in the Irankuh District using Hybrid Machine learning technique and fractal modeling. Geopersia, 12(1), 191–199. doi:10.22059/GEOPE.2022.336072.648644.

[55] Luken, K. J., Norris, R. P., & Park, L. A. F. (2019). Preliminary results of using k-nearest-neighbor regression to estimate the redshift of radio-selected data sets. Publications of the Astronomical Society of the Pacific, 131(1004), 1–7. doi:10.1088/1538-3873/aaea17.

[56] Martínez, F., Frías, M. P., Pérez, M. D., & Rivera, A. J. (2019). A methodology for applying k-nearest neighbor to time series forecasting. Artificial Intelligence Review, 52(3), 2019–2037. doi:10.1007/s10462-017-9593-z.

[57] Parashar, P. (2020). Decision Tree Regression and it's Mathematical Implementation. Medium. Available online: https://medium.com/swlh/decision-tree-regression-and-its-mathematical-implementation-58c6e9c5e88e (accessed on March 2022).

[58] Zhang, W., Zhang, R., Wu, C., Goh, A. T. C., & Wang, L. (2022). Assessment of basal heave stability for braced excavations in anisotropic clay using extreme gradient boosting and random forest regression. Underground Space (China), 7(2), 233–241. doi:10.1016/j.undsp.2020.03.001.

[59] Bakshi, C. (2020). Random Forest Regression. Medium. Available online: https://levelup.gitconnected.com/random-forest-regression-209c0f354c84 (accessed on March 2022).

[60] Ronaghan, S. (2018). The Mathematics of Decision Trees, Random Forest and Feature Importance in Scikit-learn and Spark. Medium. Available online: https://towardsdatascience.com/the-mathematics-of-decision-trees-random-forest-and-feature-importance-in-scikit-learn-and-spark-f2861df67e3 (accessed on March 2022).

[61] Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. IEEE transactions on Signal Processing, 45(11), 2673-2681. doi:10.1109/78.650093.

[62] Althelaya, K. A., El-Alfy, E.-S. M., & Mohammed, S. (2018). Evaluation of bidirectional LSTM for short-and long-term stock market prediction. 2018 9th International Conference on Information and Communication Systems (ICICS). doi:10.1109/iacs.2018.8355458.

[63] Shastri, M., Roy, S., & Mittal, M. (2019). Stock Price Prediction using Artificial Neural Model: An Application of Big Data. EAI Endorsed Transactions on Scalable Information Systems, 6(20), 1–8. doi:10.4108/eai.19-12-2018.156085.

[64] Khan, W., Malik, U., Ghazanfar, M. A., Azam, M. A., Alyoubi, K. H., & Alfakeeh, A. S. (2020). Predicting stock market trends using machine learning algorithms via public sentiment and political situation analysis. Soft Computing, 24(15), 11019–11043. doi:10.1007/s00500-019-04347-y.

[65] Ertuğrul, E., Baytar, Z., Çatal, Ç., & Muratli, Ö, C. (2019). Performance tuning for machine learning-based software development effort prediction models. Turkish Journal of Electrical Engineering and Computer Sciences, 27(2), 1308–1324. doi:10.3906/elk-1809-129.

[66] Al-Fugara, A., Ahmadlou, M., Shatnawi, R., AlAyyash, S., Al-Adamat, R., Al-Shabeeb, A. A. R., & Soni, S. (2022). Novel hybrid models combining meta-heuristic algorithms with support vector regression (SVR) for groundwater potential mapping. Geocarto International, 37(9), 2627–2646. doi:10.1080/10106049.2020.1831622.

[67] Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. Neurocomputing, 415, 295–316. doi:10.1016/j.neucom.2020.07.061.

[68] Jassim, M. S., Coskuner, G., & Zontul, M. (2022). Comparative performance analysis of support vector regression and artificial neural network for prediction of municipal solid waste generation. Waste Management and Research, 40(2), 195–204. doi:10.1177/0734242X211008526.

[69] Cui, L., Zhang, Y., Zhang, R., & Liu, Q. H. (2020). A Modified Efficient KNN Method for Antenna Optimization and Design. IEEE Transactions on Antennas and Propagation, 68(10), 6858–6866. doi:10.1109/TAP.2020.3001743.

[70] Fu, G., Lü, Q., Yan, J., Farquharson, C. G., Qi, G., Zhang, K., Zhang, Y., Wang, H., & Luo, F. (2021). 3D mineral prospectivity modeling based on machine learning: A case study of the Zhuxi tungsten deposit in northeastern Jiangxi Province, South China. Ore Geology Reviews, 131, 1–19. doi:10.1016/j.oregeorev.2021.104010.

[71] Anmala, J., & Turuganti, V. (2021). Comparison of the performance of decision tree (DT) algorithms and extreme learning machine (ELM) model in the prediction of water quality of the Upper Green River watershed. Water Environment Research, 93(11), 2360–2373. doi:10.1002/wer.1642.

[72] Contreras, P., Orellana-Alvear, J., Muñoz, P., Bendix, J., & Célleri, R. (2021). Influence of random forest hyperparameterization on short-term runoff forecasting in an andean mountain catchment. Atmosphere, 12(2), 1–16. doi:10.3390/atmos12020238.

[73] Khan, Z. A., Ullah, A., Ullah, W., Rho, S., Lee, M., & Baik, S. W. (2020). Electrical energy prediction in residential buildings for short-term horizons using hybrid deep learning strategy. Applied Sciences (Switzerland), 10(23), 1–12. doi:10.3390/app10238634.

[74] Daghigh, V., Lacy, T. E., Daghigh, H., Gu, G., Baghaei, K. T., Horstemeyer, M. F., & Pittman, C. U. (2020). Heat deflection temperatures of bio-nano-composites using experiments and machine learning predictions. Materials Today Communications, 22(10078), 9. doi:10.1016/j.mtcomm.2019.100789.

[75] Ağbulut, Ü., Gürel, A. E., & Biçen, Y. (2021). Prediction of daily global solar radiation using different machine learning algorithms: Evaluation and comparison. Renewable and Sustainable Energy Reviews, 135, 1–14. doi:10.1016/j.rser.2020.110114.

[76] Gu, Y., Liu, D., Arvin, R., Khattak, A. J., & Han, L. D. (2023). Predicting intersection crash frequency using connected vehicle data: A framework for geographical random forest. Accident Analysis & Prevention, 179, 106880. doi:10.1016/j.aap.2022.106880.

[77] Neto, M. P., & Paulovich, F. V. (2021). Explainable matrix - Visualization for global and local interpretability of random forest classification ensembles. IEEE Transactions on Visualization and Computer Graphics, 27(2), 1427–1437. doi:10.1109/TVCG.2020.3030354.

[78] Dash, R. K., Nguyen, T. N., Cengiz, K., & Sharma, A. (2021). Fine-tuned support vector regression model for stock predictions. Neural Computing and Applications. doi:10.1007/s00521-021-05842-w.

[79] Guo, Y., Han, S., Shen, C., Li, Y., Yin, X., & Bai, Y. (2018). An adaptive SVR for high-frequency stock price forecasting. IEEE Access, 6, 11397–11404. doi:10.1109/ACCESS.2018.2806180.

[80] Zheng, J., Wang, Y., Li, S., & Chen, H. (2021). The stock index prediction based on SVR model with bat optimization algorithm. Algorithms, 14(10), 1–31. doi:10.3390/a14100299.

[81] Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2020). Efficient Stock-Market Prediction Using Ensemble Support Vector Machine. Open Computer Science, 10(1), 153–163. doi:10.1515/comp-2020-0199.