International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Machine Learning Technique to Predict the Right Buying and Selling for EUR_USD

Mohamed EL Mahjouby*[1], Younes Manzali[2], Mohamed Taj Bennani[3], Mohamed Lamrini[4], Mohamed EL FAR[5]

Submitted: 04/12/2023 **Revised**: 12/01/2024 **Accepted**: 27/01/2024

Abstract: Predicting foreign exchange movements is an extensively studied and widely notable domain in finance. They have many studies using machine learning for the exchange market. This research explores and uses machine-learning techniques such as gradient boosting, random forest, bagging, extreme gradient boosting classifier, adaptive boosting, gaussian naïve, decision tree, and logistic regression and combines the adaptive boosting classifier with a base estimator decision tree. The goal of this combination is to forecast the optimal moments for purchasing and selling the euro against the dollar currency pair. This method entails suggesting the inclusion of 21 technical indicators into the training dataset to enhance the precision of the methodologies and our approach. The objective of this enhancement is to predict upcoming instances of buying and selling the currency pair euro against the dollar. The set of four metrics involves accuracy and measurements within the area under the receiver-operating characteristic curve, utilized for comparing multiple machine-learning models and assessing the effectiveness of various classification models. Analysis of the experiment demonstrates that our method achieves higher accuracy when compared to the decision tree classifier and other models, which obtained an accuracy of 0.763.

Keywords: *Technical indicators, Machine learning, Classification, Adaptive boosting classifier, Decision tree classifier*

1. Introduction

The forex market, which is alternatively referred to as the foreign exchange market or FX market, operates as a decentralized global platform where various currencies are traded among each other. As of my last knowledge update in September 2021, the forex market stands as the biggest and most liquid financial market globally, boasting an average daily trading volume surpassing $6 trillion. The forex market runs continuously, 24 hours a day, from Monday to Friday, due to the international nature of currency trading and the fact that it spans different time zones around the world.

Forecasting stock market movements presents a significant challenge for both researchers and traders. The potential for substantial profits presented by the stock market has consistently drawn a significant array of investors. Nonetheless, this earnings potential is closely linked to an elevated possibility of experiencing losses, primarily driven by the market's unpredictable fluctuations. Thus, emphasizing the significance of making well-informed trading choices through precise anticipation of market trends [1].

Regarding the methods employed for stock market analysis, some are based on artificial intelligence and machine learning techniques [2], and others are statistical methods [3]. Within the realm of literature, deep learning methods have commenced making appearances in the realm of financial research. Several deep learning implementations, including LSTM [4], temporal convolutional networks (TCN) [5], and convolutional neural networks (CNN), coexist with several machine learning techniques like the support vector machine [6].

For this purpose, we have used different machine learning methods and our strategy for forecasting the correct purchase and sale of the EUR/USD currency pair. Our goal is to classify the buying and selling the next day. This classification relies on a group of computed technical indicators, feature basics like Open, High, Low, Close, and additional attributes that dictate the decisions of buying and selling.

The organization of the document will take the following form: The second section explains and provides a literature review and related works. In the third section, we illustrate the datasets used, and the methodology, describe the structures of the proposed model architecture, and describe performance metrics in this area. The fourth part introduces and talks about the findings, while the last section provides concluding comments and outlines future research

[1]Student, Faculty of Sciences, USMBA, Fez-30000, Morocco,
ORCID ID : 0009-0006-7627-9863
[2]Student, Faculty of Sciences, USMBA, Fez–30000, Morocco,
younes.manzali@usmba.ac.ma
ORCID ID : 0000-0002-5912-3094
[3]Professor, Faculty of Sciences, USMBA, Fez–30000, Morocco,
Taj.bennani@usmba.ac.ma
ORCID ID : 0000-0003-1443-6502
[4] Professor, Faculty of Sciences, USMBA, Fez–30000, Morocco,
mohamed.lamrini@usmba.ac.ma
ORCID ID : 0009-0008-8400-1508
[5]Professor, Faculty of Sciences, USMBA, Fez–30000, Morocco,
mohamed.elfar@usmba.ac.ma
ORCID ID : 0009-0008-8400-1508
** Corresponding Author Email: mohamed.elmahjouby@usmba.ac.ma*

directions.

2. A Review of the Literature

Prior research in this subject used the currency pair to categorize either buying or selling activity, employing various machine and deep learning methods.

Viewing stock market prediction as a time series forecasting challenge, one of the conventional methods for addressing this type of issue is the Autoregressive Integrated Moving Average (ARIMA) algorithm [7]. While ARIMA is effective for linear time series, its performance is lacking when dealing with non-linear and non-stationary data present in the stock market.

Following that, the author in reference [8] put forth an approach for automated stock trading that merges technical analysis with a nearest neighbor classification algorithm. Technical analysis tools such as stop loss, stop gain, and RSI filter were integrated. This suggested technique enhances profitability while also mitigating the risks associated with market exposure.

In reference [9], the authors investigated the application of RNN and CNN algorithms. It is possible to assess the precision of these models by comparing their accuracy against actual stock market values in real-world scenarios.

The authors of [10], employed the deep convolutional LSTM algorithm for predicting fluctuations in the stock market. They utilized the Rider-based Monarch Butterfly Optimization method on a model, resulting in RMSE and MSE values of 2.6923 and 7.2487, respectively.

In reference [11], the authors presented a new stock market prediction approach that used a hybrid method of LSTM and GA. The technique outperformed the benchmark model, according to their findings.

The authors of [12], adjusted four ensemble models designed for evaluating macroeconomic factors to enhance the accuracy of predicting stock market movements one month in advance. These models include the boosting, bagging, neural network ensemble regressor, and random forest regressor. An additional objective was to employ a hybrid approach incorporating LSTM to illustrate that macroeconomic variables yield the most effective predictions for the stock market.

3. Materials and Procedures

In this section, the data source is specified, along with various machine-learning algorithms utilized for classification, technical indicators, and performance evaluation.

3.1. Dataset

This study classified the future buying and selling of the currency pair EUR_USD, The datasets consisting of daily prices from 20^{th} August 2018 to 18^{th}August 2023 had been collected with no missing data. As a result, the cumulative observation period amounts to a total of 5 years. This dataset included Open, High, Low, and Close.

3.2. Feature Engineering

The data we possess includes the subsequent attributes: Date, High, Low, Open, and Close. We calculate 21 technical indicators. The technical indicators are outlined in table 1.

Table 1. Different technical indicators

Technical indicators	Description
EMA10	Exponential moving average for 10 days
EMA30	Exponential moving average for 30 days
EMA200	Exponential moving average for 200 days
ROC10	Change rate over 10 days
ROC30	Change rate over 30 days
MOM30	Price momentum for 30 days
RSI10	Relative strength index for 10 days
RSI30	Relative strength index for 30 days
RSI200	Relative strength index for 200 days
%K10	Stochastic indicator for 10 days
%D10	Stochastic indicator for 10 days
%K30	Stochastic indicator for 30 days
%D30	Stochastic indicator for 30 days
%K200	Stochastic indicator for 200 days
%D200	Stochastic indicator for 200 days
MA21	Moving average for 21 days
MA63	Moving average for 63 days
MA252	Moving average for 252 days
upper_band	Bollinger upper line
lower_band	Bollinger lower line

Next, we included a new column labeled "signal," which encompassed the attributes (VA, VV, SA, and SV), along with an additional column named "Target" for the subsequent day. The description is shown in Table 2.

Table 2. Signal attribute record.

Signal attribute record	Description
VA	Means strong buying
SA	Means simple selling
VV	Means strong buying
SV	Means simple selling

We removed the VV and VA entries. Afterward, we converted the data in the two columns (signal, Target) into a digital format.

0: Means simple buying

1: Means simple selling

4. Machine Learning

Machine Learning is the use of specialized algorithms for datasets for trend prediction, categorization, or demarcation, and the methods have historically been applied to large databases with many dimensions [13].

4.1. Logistic Regression

Logistic Regression is a supervised machine learning technique for nonlinear data with categorical class variables [14]. However, it is more suitable for binary classification.

Logistic Regression is a statistical analysis method that describes the link between two or more explanatory factors (independent variables) sliding categories or intervals and the response variable (dependent variable) [15].

4.2. AdaBoost Classifier

AdaBoost is a technique in ensemble learning that sequentially trains and utilizes trees. It employs boosting by connecting a series of weak classifiers, with each weak classifier aiming to enhance the classification of samples previously misclassified by its predecessor. Through this process, boosting effectively combines weak classifiers in a series to form a robust classifier [16].

4.3. Bagging Classifier algorithm

The Bagging classifier technique involves partitioning sample data from datasets into training and testing data subsets. The Bagging Classifier produced certain hypotheses or probability estimates and collectively decided to arrive at a single accurate value [17].

4.4. XGBClassifier algorithm

The XGBoost Classifier, short for Extreme Gradient Boosting Classifier, is a powerful machine-learning algorithm known for its efficiency and high performance. Chen and Guestrin devised the gradient tree boosting method in [18].

4.5. Random Forest Classifier

The Random Forest is a model employed for tasks in both classification and regression. It employs a procedure known as bootstrap sampling, which involves randomly selecting samples from the initial training dataset. Once these samples are chosen, a decision tree is constructed for each new set of samples [19]. Additionally, the random forest is renowned in the field of pattern recognition, which explains its frequent application in feature selection. Random Forest's adaptable and simple machine-learning technology produces excellent results even when the superparameters are not defined, and it is the most widely utilized because of its obvious advantages [20].

4.6. Decision Tree

The Decision Tree is a commonly employed supervised learning method that is applicable for solving both regression and classification tasks. Its primary objective is to predict a target variable by creating straightforward decision rules derived from the dataset and its associated attributes [21].

DT has a tree-like structure, with nodes representing features, branches representing experiments, and leaf nodes representing class labels. A decision criterion for such a characteristic can be found in the root and every inner node. This classifier meets a sample set divided into two or more subsets. This approach is repeated until the specified standard is met satisfactorily[22].

4.7. Gradient Boosting Classifier Algorithm

The Gradient Boosting Classifier Algorithm is employed to enhance the training of classification and regression models, typically characterized as non-linear, and commonly recognized as decision or regression trees [23].

4.8. GaussianNB Algorithm

The gaussian naive bayes algorithm, based on Bayes' theorem in statistics, is a method that performs probabilistic classification by taking into account the associations among the attributes within a dataset [24].

The Gaussian Naive Bayes (GaussianNB) algorithm is a probabilistic machine learning algorithm used for classification tasks. It is a variant of the Naive Bayes algorithm specifically designed for datasets with continuous features that can be assumed to follow a Gaussian (normal) distribution. The algorithm is based on Bayes' theorem and makes use of probability distributions to classify instances into different classes.

4.9. Our Approach

This study's contribution is the creation of a prediction model for predicting the next purchasing and selling EUR_USD price using machine learning techniques.

Our goal is to propose a classification model. We have utilized various machine learning algorithms and adaboost classifier a base estimator decision tree classifier to predict the next buying and selling. The main idea behind our suggested model is to combine the adaboost classifier and the decision tree classifier. Our approach consists of seven phases: as shown in Figure1 below:

Download dataset: The dataset includes daily price records spanning from August 20, 2018, to August 18, 2023, and is complete without any missing data.

Feature Engineering: In our dataset, we have included the features Open, High, Low, and Close. A total of 21 technical indicators have been computed, and the details of this procedure can be found in the feature engineering section.

Preparing data for classification: Following the computation of these indicators, we introduced a fresh column called "target," where the signal incorporates entries of 0 or 1. A value of 0 signifies a suggestion to buy, while 1 indicates a suggestion to sell.

Preprocessing dataset: During this gathering we utilized the MinMaxScaler Python library to standardize our dataset.

Machine learning algorithms, adaboost classifier a base estimator decision tree classifier: Following the execution of the aforementioned steps, which include downloading the dataset, computing technical indicators, adding the target and signal columns, and normalizing the data, we proceeded to employ machine learning models. Specifically, we utilized the Adaboost classifier, with a decision tree classifier serving as the base estimator in our model.

Classification: Based on the execution of our model, we observed the classification outcomes regarding the prospective buying and selling activities for EUR_USD.

Results: Anticipation of forthcoming purchases and sales for the currency pair EUR_USD.

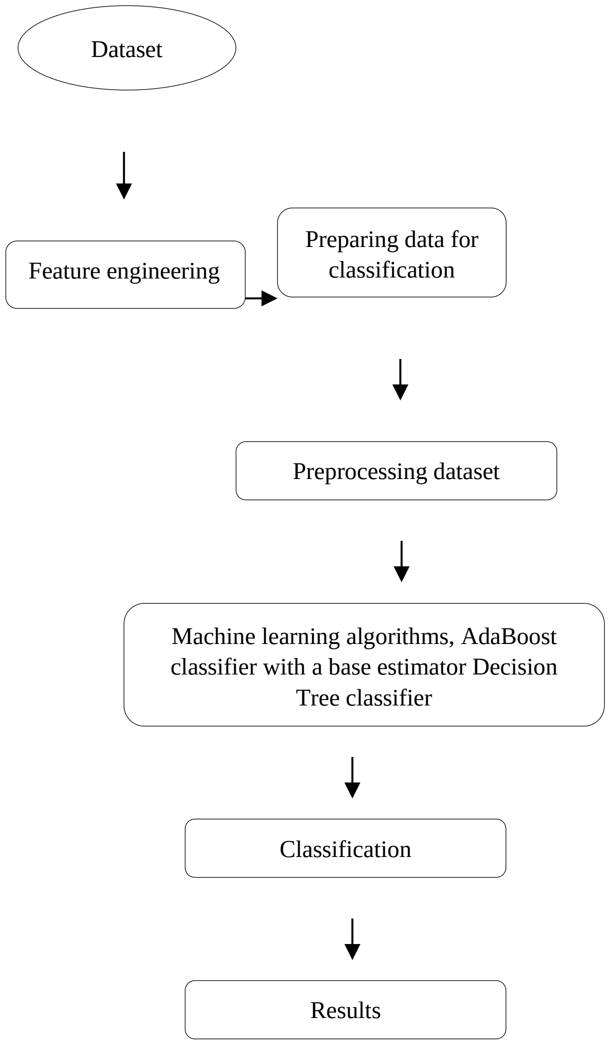

Fig. 1. Architecture of our method

5. Performance Evaluation

Performance metrics are employed to assess and compare our models. The evaluation of these models relies on measures such as accuracy, precision, recall, F1 score, and ROC-AUC. The **ROC-AUC** metric enables us to determine whether our models can impeccably differentiate between all the points in the negative and positive classes. Moreover, the conventional procedure is employed, involving the division of the dataset into a training set comprising 80% of the data and a testing set comprising 20% of the data. Furthermore, for the calculation of these metrics, we employ the symbols n (representing negative), p (representing positive), tn (indicating true negative), tp (indicating true positive), fn (representing false negative), and fp (representing false positive). Therefore, accuracy, precision, recall, and f1-score are defined as follows [25]. The equations for the four measures mentioned in (1), (2), (3), and (4) are as follows:

$$Accuracy = \frac{tn+ +tp}{tp+tn+fp+fn} \quad (1)$$

$$Precision = \frac{tp}{fp+tp} \quad (2)$$

$$Recall = \frac{tp}{tp+fn} \quad (3)$$

$$F1\_Score = \frac{precision \times recall}{precision + recall} \times 2 \quad (4)$$

6. Findings and Discussion

In this study, we utilized the EUR_USD dataset, calculated 21 distinct technical indicators, employed another attribute signal, and considered our target class. Then, the dataset was divided into testing sets with 20% and 80% for training sets. After completing these procedures, we applied several of the machine-learning techniques outlined above during the modelization phase. Among these algorithms, the AdaBoost Classifier shows more performance than the others in foreign exchange market prediction with the EUR_USD dataset, then the bagging classifier came in the second position. The outcomes of these models are displayed in Table 3.

Table 3. Comparison table

Model	Accuracy	Precision	Recall	F1_Score
Adaboost classifier	0.761	0.878	0.73	0.797
Bagging classifier	0.692	0.815	0.655	0.727
Logistic Regression	0.625	0.755	0.75	0.753
GaussianNB	0.686	0.836	0.622	0.713
XGBClassifier	0.685	0.82	0.676	0.741
Random Forest classifier	0.706	0.833	0.676	0.746
Gradient boosting	0.716	0.853	0.669	0.75
Decision Tree	0.726	0.824	0.757	0.789
Our Approach	0.763	0.853	0.784	0.817

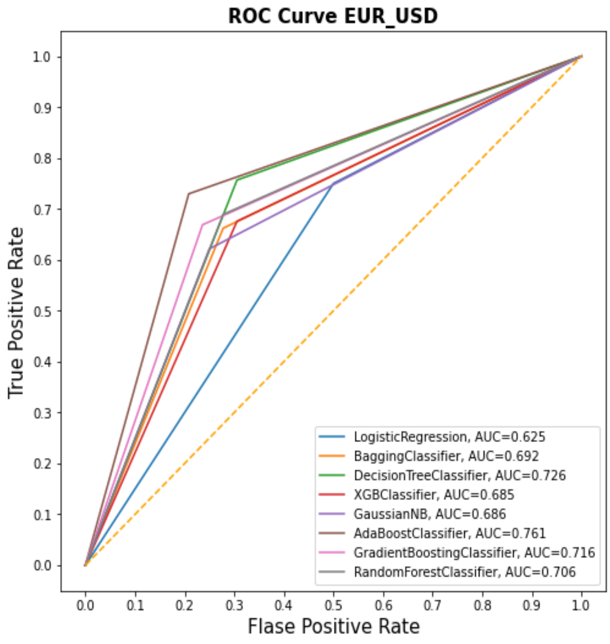

Fig. 2. ROC Curve AUC-ROC

As we can see, all AUC-ROC model values are close to 0.9, which means that these models perfectly distinguish between positive and negative cases. For example, the AdaBoost Classifier has 0.761, the Gradient Boosting Classifier has 0.716, and so on. Figure 2 shows the AUC-ROC evaluation metric that is suitable for binary classification. This evaluation metric ranges between 0 and 1.

To sum up, our proposal model shows the following figure 3.

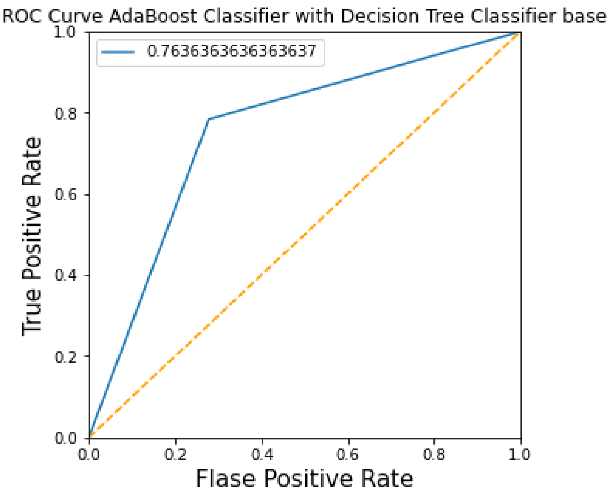

Fig. 3. ROC the our approach

Figure 3 shows the AUC-ROC of our approach. For a representation of each model's performance and to understand how they accurately identify positive and negative examples. We have included all confusion matrices from Table 4 through Table 7. As an illustration, our method yielded 52 correct positive identifications, 32 incorrect positive identifications, missed 20 positive identifications, and correctly identified 116 negative cases. This pattern continues for other instances as well.

Table 4: Classification matrix of our approach

Prediction	True	
	Postive	Negative
Positive	52	20
Negative	32	116

Table 5: Classification matrix of adaboost classifier

Prediction	True	
	Postive	Negative
Positive	57	15
Negative	40	108

Table 6: Classification matrix of gradient boosting classifier

Prediction	True	
	Postive	Negative
Positive	55	17
Negative	50	98

Table 7: Classification matrix of bagging classifier

Prediction	True	
	Postive	Negative
Positive	55	17
Negative	47	101

7. Conclusion

We have developed several machine-learning algorithms, and our approach for the classification of future buying and selling the currency pair EUR_USD with different technical indicators. Other attribute signals used, and our target class. After, testing all these techniques with our performance evaluation. Our methodology delivers favorable classification outcomes, attaining an accuracy score of 0.763, precision of 0.853, recall of 0.784, and an F1 score of 0.817. The AdaBoost Classifier approach yields noteworthy results, with an accuracy of 0.761, precision of 0.878, recall of 0.730, and an F1 score of 0.797. This work shows that our approach is a good algorithm for predicting EUR_USD. In our upcoming research, we plan to employ the same technical indicators with other techniques like deep learning to achieve better prediction results.

References

[1] B. Labiad, A. Berrado, and L. Benabbou, "Machine learning techniques for short term stock movements classification for Moroccan stock exchange," *SITA 2016 - 11th Int. Conf. Intell. Syst. Theor. Appl.*, pp. 1–6, 2016, doi: 10.1109/SITA.2016.7772259.

[2] R. K. Nayak, D. Mishra, and A. K. Rath, "A Naïve SVM-KNN based stock market trend reversal analysis for Indian benchmark indices," *Appl. Soft Comput. J.*, vol. 35, no. C, pp. 670–680, 2015, doi: 10.1016/j.asoc.2015.06.040.

[3] N. S. Arunraj and D. Ahrens, "A hybrid seasonal autoregressive integrated moving average and quantile regression for daily food sales forecasting," *Int. J. Prod. Econ.*, vol. 170, no. PA, pp. 321–335, 2015, doi: 10.1016/j.ijpe.2015.09.039.

[4] K. Chen, Y. Zhou, and F. Dai, "A LSTM-based method for stock returns prediction: A case study of China stock market," *Proc. - 2015 IEEE Int. Conf. Big Data, IEEE Big Data 2015*, pp. 2823–2824, 2015, doi: 10.1109/BigData.2015.7364089.

[5] S. Mehtab and J. Sen, "Stock Price Prediction Using Convolutional Neural Networks on a Multivariate Timeseries," 2020, doi: 10.36227/techrxiv.15088734.v1.

[6] W. Lu, J. Li, Y. Li, A. Sun, and J. Wang, "A CNN-LSTM-based model to forecast stock prices," *Complexity*, vol. 2020, 2020, doi: 10.1155/2020/6622927.

[7] A. A. Adebiyi, A. O. Adewumi, and C. K. Ayo, "Stock price prediction using the ARIMA model," *Proc. - UKSim-AMSS 16th Int. Conf. Comput. Model. Simulation, UKSim 2014*, pp. 106–112, 2014, doi: 10.1109/UKSim.2014.67.

[8] L. A. Teixeira and A. L. I. De Oliveira, "A method for automatic stock trading combining technical analysis and nearest neighbor classification," *Expert Syst. Appl.*, vol. 37, no. 10, pp. 6885–6890, 2010, doi: 10.1016/j.eswa.2010.03.033.

[9] G. Rekha, B. D Sravanthi, S. Ramasubbareddy, and K. Govinda, "Prediction of stock market using neural network strategies," *J. Comput. Theor. Nanosci.*, vol. 16, no. 5–6, pp. 2333–2336, 2019, doi: 10.1166/jctn.2019.7895.

[10] A. Kelotra and P. Pandey, "Stock Market Prediction

Using Optimized Deep-ConvLSTM Model," *Big Data*, vol. 8, no. 1, pp. 5–24, 2020, doi: 10.1089/big.2018.0143.

[11] H. Chung and K. S. Shin, "Genetic algorithm-optimized long short-term memory network for stock market prediction," *Sustain.*, vol. 10, no. 10, 2018, doi: 10.3390/su10103765.

[12] B. Weng *et al.*, "Macroeconomic indicators alone can predict the monthly closing price of major U.S. indices: Insights from artificial intelligence, time-series analysis and hybrid models," *Appl. Soft Comput.*, vol. 71, no. October 2019, pp. 685–697, 2018, doi: 10.1016/j.asoc.2018.07.024.

[13] R. Ali, M. M. Yusro, M. S. Hitam, and M. Ikhwanuddin, "Machine Learning With Multistage Classifiers For Identification Of Of Ectoparasite Infected Mud Crab Genus Scylla," *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 19, no. 2, pp. 406–413, 2021, doi: 10.12928/TELKOMNIKA.v19i2.16724.

[14] H. Takci, "Improvement of heart attack prediction by the feature selection methods," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 26, no. 1, pp. 1–10, 2018, doi: 10.3906/elk-1611-235.

[15] T. Mantoro, M. A. Permana, and M. A. Ayu, "Crime index based on text mining on social media using multi classifier neural-net algorithm," *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 20, no. 3, pp. 570–579, 2022, doi: 10.12928/TELKOMNIKA.v20i3.23321.

[16] S. Misra and H. Li, *Noninvasive fracture characterization based on the classification of sonic wave travel times*. Elsevier Inc., 2019. doi: 10.1016/B978-0-12-817736-5.00009-0.

[17] Y. P. Huang and M. F. Yen, "A new perspective of performance comparison among machine learning algorithms for financial distress prediction," *Appl. Soft Comput.*, vol. 83, p. 105663, Oct. 2019, doi: 10.1016/J.ASOC.2019.105663.

[18] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, vol. 13-17-Augu, pp. 785–794, 2016, doi: 10.1145/2939672.2939785.

[19] Z. Jin, J. Shang, Q. Zhu, C. Ling, W. Xie, and B. Qiang, "RFRSF: Employee Turnover Prediction Based on Random Forests and Survival Analysis," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12343 LNCS, pp. 503–515, 2020, doi: 10.1007/978-3-030-62008-0_35.

[20] H. A. Saleh, R. A. Sattar, E. M. H. Saeed, and D. S. Abdul-Zahra, "Hybrid features selection method using random forest and meerkat clan algorithm," *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 20, no. 5, pp. 1046–1054, 2022, doi: 10.12928/TELKOMNIKA.v20i5.23515.

[21] M. Nabipour, P. Nayyeri, H. Jabani, S. Shahab, and A. Mosavi, "Predicting Stock Market Trends Using Machine Learning and Deep Learning Algorithms Via Continuous and Binary Data; A Comparative Analysis," *IEEE Access*, vol. 8, pp. 150199–150212, 2020, doi: 10.1109/ACCESS.2020.3015966.

[22] G. Pattnaik and K. Parvathi, "Machine learning-based approaches for tomato pest classification," *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 20, no. 2, pp. 321–328, 2022, doi: 10.12928/TELKOMNIKA.v20i2.19740.

[23] J. K. Mandal, Emerging Technologies in Data Mining and Information Security Proceedings of IEMIS 2018, Volume 2 by Ajith Abraham, Paramartha Dutta, Jyotsna Kumar Mandal, Abhishek Bhattacharya, Soumi Dutta (z-lib.org).pdf, vol. 2. 2018. doi: 10.1007/978-981-13-1498-8.

[24] P. Venkata and V. Pandya, "Data mining model and Gaussian Naive Bayes based fault diagnostic analysis of modern power system networks," *Mater. Today Proc.*, vol. 62, no. P13, pp. 7156–7161, Jan. 2022, doi: 10.1016/J.MATPR.2022.03.035.

[25] M. Shohel Rana, C. Gudla, and A. H. Sung, "Evaluating machine learning models for android malware detection - A comparison study," *ACM Int. Conf. Proceeding Ser.*, no. March 2019, pp. 17–21, 2018, doi: 10.1145/3301326.3301390.