SHM PUBLISHER
SURYA HIJAU MANFAAT

Journal of Soft Computing Exploration

Homepage: https://shmpublisher.com/index.php/joscex

JOSCEX

p-ISSN: 2746-7686
e-ISSN: 2746-0991

.

A model for determining stock purchase decisions based on gated recurrent units and decision tree C4.5

Muhammad Arsyad Rayhan Aziis[1], Wiranto[2], Arif Rohmadi[3]
[1,2,3] Department of Informatics, Universitas Sebelas Maret, Indonesia

Article Info

Article history:

Received May 21, 2026
Revised June 11, 2026
Accepted June 16, 2026

Keywords:

Gated Recurrent Unit
Decision Tree C4.5
Stock Market Prediction
Technical Indicators
Trading Decision Support System

ABSTRACT

The stock market is highly volatile but offers high profit potential. This makes it difficult for novice investors to make investment decisions. Current studies mostly focus on stock price prediction or trading signal classification, while interpretable hybrid frameworks to support stock purchase decisions are still limited. Many machine learning models offer limited interpretability. Numerical predictions are often generated by forecasting models, which are difficult to translate into investment decisions. Therefore, this study aims to design a hybrid decision support system with an interpretable model by combining a Gated Recurrent Unit (GRU) and a C4.5 Decision Tree classifier in stock purchase decision making. The proposed framework consists of two phases. The first step is to predict the closing price of stocks using the historical daily data with the GRU model. The predicted price is then combined with technical indicators such as Simple Moving Average (SMA), Relative Strength Index (RSI) and Moving Average Convergence Divergence (MACD) to produce trading signals using C4.5 Decision Tree. The dataset used in this study is BBNI.JK stock data from 2015 to 2025 with a walk-forward validation scheme and evaluation using RMSE, MAE, MAPE, Accuracy, Precision, Recall and F1-Score. The experimental results show that the MAPE of the GRU model is 6.16% and the proposed DT-GRU strategy produces the highest trading return of 83.07% with a Sharpe ratio of 2.75. These results indicate that the combination of interpretable forecasting and classification can provide effective and practical trading decision-making support.

Corresponding Author:

Muhammad Arsyad Rayhan Aziis,
Department of Informatics,
Faculty of Information Technology and Data Science, Universitas Sebelas Maret
Ir. Sutami Street No. 36A Kentingan, Surakarta, Central Jawa 57216
Email: arsyad.rayhan22@student.uns.ac.id
https://doi.org/10.52465/joscex.v7i2.133

1. INTRODUCTION

The stock market provides high potential returns but is also characterized by high volatility of prices which makes it difficult for novice investors to make investment decisions [1], [2], [3]. While there are many forecasting techniques that have been developed to predict future stock prices, it is still difficult for many investors to translate these predictions into practical buy or sell decisions. This results in the demand for decision support systems that provide not only price prediction but also actionable and understandable trading advice.

There have been many methods proposed for stock price prediction including statistical models such as ARIMA and machine learning and deep learning techniques. However, traditional models generally have difficulty in capturing complex non-linear patterns and long-term dependencies in the financial time-series data [4], [5], [6]. Deep learning approaches such as Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) have shown better predictive power, but their computational complexity and black-box nature reduce the interpretability for real investment decision making [4], [7]. Moreover, single-model approaches are usually less robust in highly volatile market conditions [5].

To address these limitations this research proposes a hybrid approach of Gated Recurrent Unit (GRU) and Decision Tree C4.5 (DT). Although macroeconomic variables such as interest rate, inflation rate and exchange rate are known to have an impact on stock market behaviour, these were not considered as input variables in this research. The proposed framework depends only on historical stock price data (Open, High, Low, Close and Volume) and technical indicators based on these variables. Stock price prediction is carried out using GRU which is effective in modelling sequential and non-linear data, which overcomes the inability of basic individual models to handle complex data structures [8]. DT is applied to convert the prediction output into interpretable stock purchase signals [4], [9], [10]. Deep learning study for stock prediction for the Indonesian stock market with the combination of GRU forecasting and interpretable decision support for stock purchase decisions is still limited (Zhu, 2025). The novelty of this study lies in the use of a two-stage hybrid framework combining GRU forecasting and C4.5 Decision Tree classification to convert stock price predictions into interpretable stock purchase recommendations. In contrast to many existing studies that mainly focus on forecasting performance, this framework highlights both predictive ability and transparent decision support. Therefore, this study aims to develop a hybrid DT and GRU model to produce predictive and interpretable stock purchase recommendations for novice investors.

This study has the objectives to: (1) implement the GRU model to the daily stock price prediction using historical data of the Indonesia Stock Exchange from 2015 to 2025, (2) combine the prediction results of the GRU model with DT algorithm to generate stock buying signals, and (3) evaluate the performance of the proposed hybrid model by using the classification accuracy metrics such as precision, recall, and error measurements to ensure the reliability of the predictive system [11].

2. METHOD

In this paper, we are proposing a hybrid model with two stages that combines the GRU model to predict stock prices and the Decision Tree C4.5 (DT) model classify trading signal. this hybrid model aims to predict stock prices and generate a buy or sell signals for the closing price of an LQ45 index member stocks on the Indonesia Stock Exchange (IDX) by using historical stock price data from Yahoo Finance with the yfinance python library [12]. The Decision Tree C4.5 Algorithm is selected as it can give an interpretable rule-based decision structure that can be understood by investors [13]. Unlike many machine learning models that have a low degree of transparency in the decision-making process, C4.5 produces a single and interpretable tree structure, such that the classification results can be presented transparently as if-then decision rules. In addition, C4.5 uses the Gain Ratio criterion, which eliminates bias in the attribute selection process and produces more balanced decision trees [14], [15]. These characteristics enable the resulting rules to be directly translated to buy, sell, or hold recommendation. The overall workflow of the research is shown in Figure 1.

.

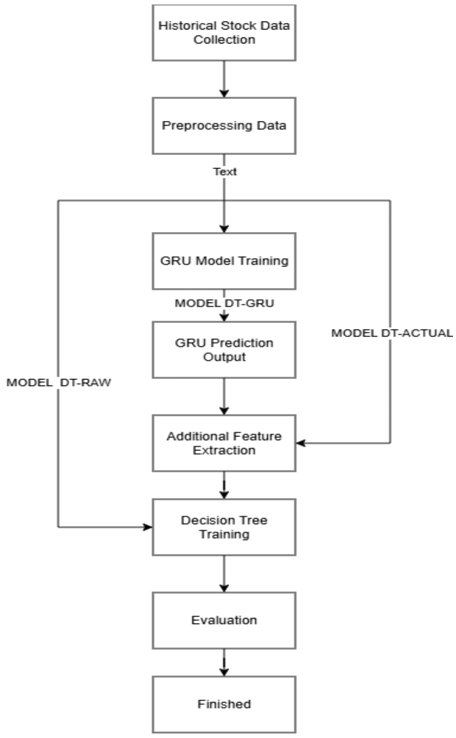

Figure 1. Workflow Diagram of the GRU and DT Stock Purchase Decision Model Development.

Data Collection and Pre-processing

This study uses daily historical stock price data (Open, High, Low, Close, and Volume) of Bank Negara Indonesia (BBNI.JK), one of the constituents of the LQ45 index, covering the period from 2015 to 2025. No macroeconomic, fundamental, or sentiment variables were included; therefore, all model inputs were derived solely from historical price and volume data. This is then followed by three pre-processing stages:

- Data Cleaning: Missing values due to exchange holidays or trading suspensions are filled by forward-fill imputation to maintain the temporal continuity without look-ahead bias [9].
- Min-Max Normalization: All numeric features are scaled to the range [0, 1] to prevent features with large magnitudes (e.g., Volume) from dominating gradient updates during GRU model training [16], [17].
- Time-Window Construction (Windowing): The time series data are converted from the 2-D format to the 3-D format by using the sliding window of n days to predict the closing price at step t+1, so that the model can capture the temporal dependencies [16].

Data is split chronologically into 70% training, 15% validation and 15% test sets, without shuffling to maintain temporal order and prevent data leakage [14], [18].

GRU-based Price Prediction Model

The GRU is a variation of the RNN that is used to address the vanishing-gradient problem in long sequential data [19], [20]. GRU merges the forget gate and the input gate of LSTM into an update gate and thus has a simpler architecture and similar performance but is computationally more efficient [12], [21].

GRU computation involves two main gates:

Update gate [19]:

$$z_t = \sigma(W_z \cdot x_t + U_z \cdot h_{t-1} + b_z) \quad (1)$$

Reset gate [12], [22]:

$$r_t = \sigma(W_r \cdot x_t + U_r \cdot h_{t-1} + b_r) \quad (2)$$

Candidate hidden state [20]:

$$\widetilde{h_t} = tanh(W_h \cdot x_t + U_h \cdot (r_t \odot h_{t-1}) + b_h) \quad (3)$$

Final hidden state [16]:

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \widetilde{h_t} \quad (4)$$

Description: σ denotes the sigmoid activation function, while $\odot$ represents element-wise multiplication (Hadamard product). The variable z_t functions as the update gate, regulating the balance between retaining previous memory and incorporating new information. Values of z_t approaching 1 indicate that the model emphasizes newly acquired information, whereas values approaching 0 signify stronger preservation of historical memory. [14], [23].

Technical Indicator Feature Extraction

To enrich the input feature set for DT classification, three principal technical indicators are extracted from historical price data [14], [23]:

- Simple Moving Average (SMA): Smooths price fluctuations by averaging closing prices over a given period *n* [14], [24]:

$$SMA_t(n) = \frac{1}{n}\sum_{i=1}^{n} C_i \tag{5}$$

Description: $\mathrm{SMA}(t, n)$ represents the SMA calculated at time t over a rolling window of size n, while C_i denotes the closing price at period i.

- Relative Strength Index (RSI): A momentum oscillator on a 0–100 scale; values above 70 indicate overbought conditions and below 30 indicate oversold conditions [7], [23]:

$$RSI = 100 - \left(\frac{100}{1 + RS}\right) \tag{6}$$

$$RS = \frac{AverageGain}{AverageLoss} \tag{7}$$

Description: RS denotes the Relative Strength, defined as the ratio between the average price gain and the average price loss over a specified observation period.

- Moving Average Convergence Divergence (MACD): A trend-following momentum indicator derived from the difference between the 12-period EMA and the 26-period EMA, with a 9-period EMA signal line [7], [23], [24]:

$$MACDLine_t = EMA_t(12) - EMA_t(26) \tag{8}$$

$$SignalLine_t = EMA_t(MACDLine_t, 9) \tag{9}$$

Description: In the MACD formula, $\mathrm{MACD\ Line}_t$ represents the primary MACD line at time t, calculated as the difference between two Exponential Moving Averages (EMA), namely $\mathrm{EMA}_t(12)$ and $\mathrm{EMA}_t(26)$, corresponding to the 12-period and 26-period EMA, respectively. Meanwhile, $\mathrm{Signal\ Line}_t$ denotes the signal line, which is obtained by applying a 9-period EMA to the MACD values.

Trading Signal Classification with DT

In the second stage, the meta-learner is DT that classifies trading signals (Buy/Sell) from a combined feature vector of GRU price predictions and technical indicators [14], [25]. Gain Ratio is used for attribute selection to normalize information gain in order to avoid bias towards high cardinality attributes [25]:

$$GainRatio = \frac{H(Class) - H(Class|Attribute)}{H(Attribute)} \tag{10}$$

Description: Information Gain is defined as H(Class) - H(Class|Attribute). It quantifies the decrease in the uncertainty of the target class when the dataset is split according to a specific attribute. Meanwhile H(Attribute) is the attribute entropy (also called Split Information) which is used as a normalization factor in calculating the Gain Ratio.

The target variable is a binary label defined as a label 1 (uptrend) if tomorrow's closing price is greater than today's, and label 0 (downtrend) otherwise. We also use pruning to avoid overfitting on volatile financial data [23], [25].

.

Hybrid Model Integration

The Sequential Hybrid Two-Stage architecture combines both models as follows: Let X be the initial historical feature set, and f1 be the GRU mapping function. The first stage prediction is expressed as [26], [27]:

$$\widehat{y_1} = f_1(X) \tag{11}$$

Description: X is the input feature set from market data and technical indicators, and $\hat{y}_1$ is the GRU prediction output to serve as the initial representation of stock price movement.

Then the DT (f2) takes as input the concatenation of the original features and the output of the GRU to produce the final trading signal as follows:

$$\widehat{Y_{final}} = f_2(X, \widehat{y_1}) \tag{12}$$

Description: f_2 represents the DT classifier and $\hat{Y}_{\text{final}}$ }is the final classification output in the form of a Buy/Sell trading signal. This sequential architecture enables the second-stage model to learn higher-level patterns from the representations generated by the first-stage GRU model, leading to a more rich and discriminative feature space. This integration has been shown to enhance the predictive and classification performance over single model approaches. [15], [27].

Model Evaluation

Price Prediction (GRU) Three regression metrics are used. The Root Mean Square Error (RMSE) is sensitive to price spikes [9], [12] as it penalizes more the extreme deviations:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y_i})^2} \tag{13}$$

Description RMSE is the difference between actual stock price (y_i) and predicted value $(\hat{y}_i)$, where each error term is squared $(y_i - \hat{y}_i)^2$. The squared errors are summed, averaged over the total number of observations (n), and square-rooted to bring the value back to the original price scale. RMSE is very sensitive to large prediction errors due to the squaring, which makes it useful for identifying large deviations or extreme forecasting errors.

MAPE measures the relative percentage error to facilitate cross-scale comparison [12], [17]:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \widehat{y_i}}{y_i}\right| \times 100\% \tag{14}$$

Description: Mean Absolute Percentage Error (MAPE) computes the absolute difference between the actual value and the predicted value $| y_i - \hat{y}_i |$, and then divides it by the actual value $| \frac{y_i - \hat{y}_i}{y_i} |$. The results are averaged and multiplied by 100 % to give a percentage based error value. MAPE is the relative error of a model's prediction versus the actual stock price. It is useful to compare the prediction performance of different stocks with different price scales.

Mean Absolute Error (MAE) provides an average absolute error that is more robust to extreme values since errors are not squared [20], [21]:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y_i}| \tag{15}$$

Description: MAE calculates the average absolute difference between the actual stock price $(y_i)$and the predicted value $(\hat{y}_i)$using the expression $| y_i - \hat{y}_i |$. Unlike RMSE, MAE does not square the error values, meaning each prediction error is treated linearly and proportionally. As a result, MAE is generally more stable and more robust to outliers or extreme price fluctuations in stock market data.

Trading Signal Classification (DT): Four metrics derived from the Confusion Matrix are used [14], [27].

Precision measures the confidence of generated Buy signals, ensuring that when the system issues a "Buy", the price genuinely rises, thereby minimising false positives and potential losses [15]. Recall measures the system's ability to capture all available upward price opportunities, minimising missed opportunities [26]. Both metrics are important given the potential class imbalance between Buy and Sell signals. F1-Score balances the two [27]:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{16}$$

$$Precision = \frac{TP}{TP + FP} \tag{17}$$

$$Recall = \frac{TP}{TP + FN} \tag{18}$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{19}$$

Description: TP (True Positive) When the model predicts that the stock price will go up and the stock price actually goes up. TN (True Negative): This is the case when the model correctly predicts a price decline and the price actually declines. FP (False Positive) is when the price is predicted to go up but it actually goes down and gives a false Buy signal. On the other hand, FN (False Negative) refers to the model predicting a price drop, but the price actually increases, leading to a missed profit opportunity.

Accuracy is the ratio of the number of correct predictions to the total number of samples and provides a general overview of performance; high accuracy indicates that the model can be able to distinguish profitable and unprofitable market conditions [14], [28], [29].

Evaluation is conducted comparatively across three model variants: DT-Actual (upper bound using real prices), DT-Raw (baseline without GRU input), and DT-Hybrid (the proposed model). This comparison aims to quantify the incremental value of GRU prediction integration on DT performance, consistent with findings that hybrid models improve evaluation metrics over single models [15].

The validation is done with walk-forward validation with expanding window mechanism to guarantee realism and to avoid data leakage [23]. The final performance is assessed not only by the mean value of the metric for each fold, but also by the standard deviation across folds to assess model stability and consistency under varying market conditions throughout the observation period [23].

also evaluate portfolio performance using the cumulative return, annualized Sharpe ratio (assuming 252 trading days), and Maximum Drawdown on the backtesting simulation with 0.1 % transaction cost [24], [27], [30]. The benchmark is a buy and hold strategy.

3. RESULTS AND DISCUSSIONS

Data Collection

Historical Data BBNI.JK has the attributes Date, Open, High, Low, Close, Adj Close and Volume. The price during the observation period ranged from IDR 2,000 up to more than IDR 6,000, with a steep drop in early 2020 and a price rebound in the 2023–2024 period. Table 1 shows an example of the research data.

Table 1. Research Data Sample of BBNI.JK

Date	High	Low	Close	Open	Volume
2015-01-02	3,050.0	3,062.5	3,025.0	3,025.0	23,401,400
2015-01-05	3,012.5	3,037.5	3,000.0	3,037.5	27,649,600
...	...	...	...	...	...
2025-12-29	4,260.0	4,320.0	4,260.0	4,270.0	24,490,400
2025-12-30	4,370.0	4,380.0	4,250.0	4,260.0	44,229,200

Data Preprocessing

The preprocessing steps were data cleaning, Min-Max Scaling normalization, sequence formation (windowing), feature engineering and data splitting. After cleaning, the dataset contained 2711 rows without any missing values. The final dataset was 2,661 rows after the application of the technical indicators and the deletion of the NaN values. The data was divided sequentially in a 70:15:15 ratio as shown in Table 2.

Table 2. Dataset Split Results

Split	Data Count	Proportion	Time Range
Train	1,862	70.0%	Jan 2, 2015 – Sep 20, 2023

.

Validation	399	15.0%	Sep 21, 2023 – Nov 6, 2024
Test	400	15.0%	Nov 7, 2024 – Dec 30, 2025
Total	2,661	100.0%	Jan 2, 2015 – Dec 30, 2025

The GRU was trained using a window size of 40 time steps. The MinMaxScaler normalization was fitted only on the training data to prevent data leakage, and then applied to the validation and test sets. The number of sequences in the training, validation, and test sets followed the chronological division after preprocessing.

Model Development

This model was developed in stages. First, the GRU model made stock price predictions. The predictions from the GRU model were used as one of the input features for the DT model, along with technical indicators obtained through the feature engineering process.

GRU Model

Table 3. Best Hyperparameter Configuration of the GRU Model

Parameter	Value
Window Size	40
Units / N_layers	32 / 2
Dropout / Learning Rate	0.1 / 0.0005
Batch Size / Patience	32 / 15
Maximum Epoch	200

Table 4. GRU Model Architecture

Layer	Function
GRU(32)	Capture time dependencies in price data
Dropout(0.1)	Reduces risk of overfit
Dense(16, ReLU)	Nonlinear Representation
Dense(1)	Outputs closing price prediction

DT Model

Three variants of the DT model were built using criterion = "entropy" and random_state = 42. Feature engineering employed technical indicators including SMA (5, 10, and 20 days), RSI (14 periods), and MACD. The classification target was binary: 1 if the next day's price increases, and 0 if it decreases or remains the same. The feature combinations and configurations for each model are presented in Table 5.

Table 5. DT Model Variant Configurations

Model	Input Features	max_depth	min_split	min_leaf
DT-Actual	Actual Close + SMA + RSI + MACD	5	10	2
DT-GRU	GRU Prediction + SMA + RSI + MACD	8	10	2
DT-RAW	Close only (no indicators)	3	5	2

Trading signals (buy/sell/hold) were generated based on the difference between the predicted price and the actual price using a threshold value. The trading simulation used an initial capital of IDR 10,000,000 with a transaction fee of 0.1% per trade, a next-bar execution approach, and walk-forward validation with an expanding window.

Model Evaluation

GRU Model Evaluation

The inverse transformation was applied to evaluate the model on the original IDR scale using RMSE, MAE and MAPE metrics. The results of the evaluation are presented in Table 6.

Table 6. GRU Model Evaluation Results

Split	RMSE	MAE	MAPE
Train	IDR 199	IDR 149	4.64%
Validation	IDR 313	IDR 282	5.72%
Test	IDR 336	IDR 292	6.16%

The GRU model resulted in an RMSE of IDR 336 and an MAPE of 6.16% on the test data, showing that the model can follow the pattern of price movement with a relatively low error rate. Although the directional prediction was not yet optimal given the volatile stock data, the output of GRU was still employed as an additional feature in the DT-GRU model.

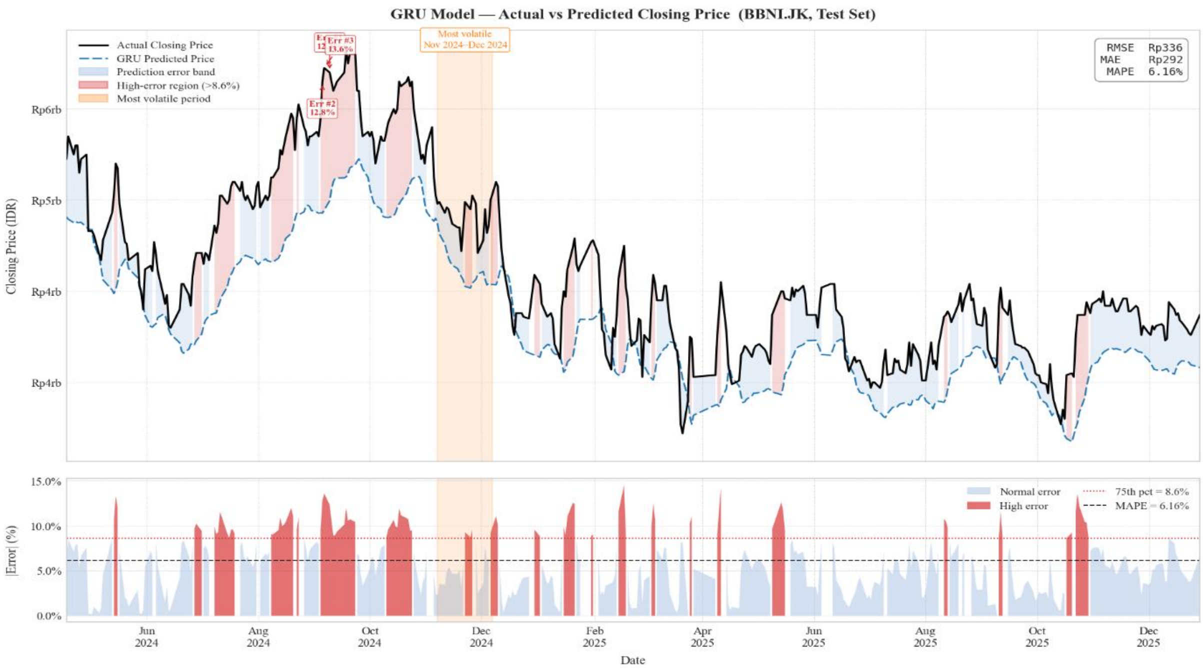

Figure 2. Actual and predicted closing prices of the GRU model on the BBNI.JK test dataset.

Figure 2 compares the real closing prices and the GRU predicted closing prices for the test dataset. The inverse transformation was applied to evaluate the model on the original IDR scale using RMSE, MAE and MAPE metrics. The difference was even more pronounced during periods of increased market volatility, especially in late 2024, when sharp price swings resulted in higher prediction errors. These deviations notwithstanding, the predicted series remained relatively close to the actual prices over the test period, in line with the quantitative results presented in Table 6. The results obtained show that the GRU model is accurate enough to be used as an input feature for the generation of the next trading signals.

DT Model Evaluation

Classification evaluation used the Accuracy, Precision, Recall, and F1-Score metrics with a walk-forward validation scheme. Evaluation and trading simulation results are presented in Table 7 and Table 8.

Table 7. Classification Evaluation Results of DT Models on Test Data

Model	Accuracy	Precision	Recall	F1-Score
DT-Actual	0.6271	0.4737	0.4286	0.4500
DT-GRU	0.5254	0.3500	0.3182	0.3333

.

DT-RAW	0.5932	0.3846	0.2381	0.2941

Table 8. Trading Simulation Results of DT Strategies

Strategy	Return	Sharpe Ratio	Max Drawdown	Final Value
DT-GRU	83.07%	2.75	-5.13%	IDR 18,307,418
DT-Actual	54.80%	1.95	-4.98%	IDR 15,480,282
DT-RAW	57.52%	2.04	-6.59%	IDR 15,752,137

While DT-Actual got the best classification accuracy (62.71%) with F1 score (0.45), But DT-GRU got the best cumulative return (83.07%) with Sharpe ratio (2.75). These results indicate that classification accuracy is not a sufficient criterion for evaluating trading strategies. Profitable trading in financial markets depends not only on the number of correct predictions but also on the economic value and timing of trading signals [23].

DT-GRU improved financial performance can be attributed to detecting market movements with more economic significance. Classification metrics give equal weight to each prediction, but correctly predicting large upward movements or avoiding large downward movements has far greater contribution to the growth of a portfolio. By utilizing predicted future prices produced by GRU, DT-GRU is able to produce trading signals closer to actual profitable market turning points, resulting in improved entry and exit decisions[26].

These findings suggest that retail investors should not base stock recommendation system evaluations off of classification accuracy alone, as a lower accuracy model can still out-produce other models from a returns standpoint if it outputs more efficient trade signals.

They also lend support to prior works on GRU s ability to identify nonlinear temporal patterns [7] present in stock prices, as well as works on the interpretability of Decision Tree models for decision support purposes [14]. The DT-GRU framework we propose, as a whole benefits from both these attributes which are complimentary: We empirically show that forecasting ability and interpretable decision making can go hand-in-hand to provide greater practical value towards supporting stock investment decisions.

4. CONCLUSION

This paper proposes a hybrid decision support framework for stock purchase decision making using BBNI that combines Gated Recurrent Unit (GRU) model and C4.5 Decision Tree (DT) classifier. JK stock data from 2015 to 2025. The GRU model has demonstrated satisfactory forecasting performance with MAPE of 6.16%. The proposed DT-GRU strategy generated the highest cumulative return of 83.07%, compared to DT-Actual and DT-RAW, although the classification accuracy is lower. These results suggest that trading profitability is not only affected by the prediction accuracy, but also the economic significance and timing of the generated trading signals. The main contribution of this paper is the development of an interpretable hybrid framework integrating GRU-based forecasting with explainable C4.5 decision rules, linking predictive modeling and practical investment decision support. However, there are several limitations that warrant attention. First, this study only analyzed one stock (BBNI.JK), Second This study is only using technical indicators without considering macroeconomic, fundamental, or sentiment factors In further research, the study can be tested on several stocks under different conditions, and compared with other classification methods such as Random Forest, XGBoost, and LightGBM, as well as combining variables such as news sentiment and macroeconomic indicators.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Muhammad Arsyad Rayhan Aziis: Conceptualization, Methodology, Writing-original draft preparation. **Wiranto:** Supervision, Validation. **Arif Rohmadi**: Supervision, Validation.

DECLARATION OF COMPETING INTERESTS

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY

The data used in this study is publicly available from Yahoo Finance.

REFERENCES

[1] H. Mushtaq, M. Ishtiaq, S. Jamal, S. M. Raza Rizvi, and H. Raza, “Regime-Switching volatility and risk quantification in South Asian and developed stock Markets: A Comparative perspective using Markov-Switching GARCH with MLE and MCMC estimations,” *North American Journal of Economics and Finance*, vol. 82, Jan. 2026, doi: 10.1016/j.najef.2025.102576.

[2] W. Simatupang, N. Paramitha, N. Aini, and S. Dasman, “Manajemen Keuangan Analisis Risiko Investasi Saham melalui Diversifikasi Portofolio Secara Domestik Dan Internasional,” Nov. 2024, doi: 10.47467/elmal.v4i5.2895.

[3] S. Tiwari, H. Ramampiaro, and H. Langseth, “Machine Learning in Financial Market Surveillance: A Survey,” *IEEE Access*, vol. 9, pp. 159734–159754, 2021, doi: 10.1109/ACCESS.2021.3130843.

[4] W. Bao, Y. Cao, Y. Yang, H. Che, J. Huang, and S. Wen, “Data-driven stock forecasting models based on neural networks: A review,” *Information Fusion*, vol. 113, Jan. 2025, doi: 10.1016/j.inffus.2024.102616.

[5] S. Ren, X. Wang, X. Zhou, and Y. Zhou, “A novel hybrid model for stock price forecasting integrating Encoder Forest and Informer,” *Expert Syst. Appl.*, vol. 234, Dec. 2023, doi: 10.1016/j.eswa.2023.121080.

[6] A. M. ’Ainul Yaqin, R. Hamdi, M. I. Zamzani, C. D. P. Hertadi, and H. D. P. Nabiha, “A Hybrid Traditional and Machine Learning-Based Stacking-Based Ensemble Forecasting Approach for Coal Price Prediction,” *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 7, no. 2, p. 436, Jul. 2024, doi: 10.24014/ijaidm.v7i2.30547.

[7] X. Zhu, “The role of hybrid models in financial decision-making: Forecasting stock prices with advanced algorithms,” *Egyptian Informatics Journal*, vol. 29, Mar. 2025, doi: 10.1016/j.eij.2025.100610.

[8] H. Song and H. Choi, “Forecasting Stock Market Indices Using the Recurrent Neural Network Based Hybrid Models: CNN-LSTM, GRU-CNN, and Ensemble Models,” *Applied Sciences (Switzerland)*, vol. 13, no. 7, Apr. 2023, doi: 10.3390/app13074644.

[9] William, A. K. Wilham, and S. R. Manalu, “Stocks and models: A comparative benchmark of machine learning and deep learning approaches for stock market prediction,” in *Procedia Computer Science*, Elsevier B.V., 2025, pp. 1532–1545. doi: 10.1016/j.procs.2025.09.095.

[10] M. D. Lagzi, S. M. sajadi, and M. Taghizadeh-Yazdi, “A hybrid stochastic data envelopment analysis and decision tree for performance prediction in retail industry,” *Journal of Retailing and Consumer Services*, vol. 80, Sep. 2024, doi: 10.1016/j.jretconser.2024.103908.

[11] K. Tharageswari, N. Mohana Sundaram, and R. Santhosh, “A Hybrid Deep Learning Algorithm Based Prediction Model for Sustainable Healthcare System,” *Journal of Automation, Mobile Robotics and Intelligent Systems*, vol. 19, no. 2, pp. 89–98, Jun. 2025, doi: 10.14313/jamris-2025-019.

[12] A. Rahmadeyan and Mustakim, “Long Short-Term Memory and Gated Recurrent Unit for Stock Price Prediction,” in *Procedia Computer Science*, Elsevier B.V., 2024, pp. 204–212. doi: 10.1016/j.procs.2024.02.167.

[13] D. Muhammad, I. Ahmed, K. Naveed, and M. Bendechache, “An explainable deep learning approach for stock market trend prediction,” *Heliyon*, vol. 10, no. 21, Nov. 2024, doi: 10.1016/j.heliyon.2024.e40095.

[14] N. Ayyildiz and O. Iskenderoglu, “How effective is machine learning in stock market predictions?,” *Heliyon*, vol. 10, no. 2, Jan. 2024, doi: 10.1016/j.heliyon.2024.e24123.

[15] M. R. P. Putra and E. Utami, “Comparative Analysis of Hybrid Model Performance Using Stacking and Blending Techniques for Student Drop-Out Prediction in MOOC,” *Jurnal RESTI*, vol. 8, no. 3, pp. 346–354, Jun. 2024, doi: 10.29207/resti.v8i3.5760.

[16] A. Lawi, H. Mesra, and S. Amir, “Implementation of Long Short-Term Memory and Gated Recurrent Units on grouped time-series data to predict stock prices accurately,” *J. Big Data*, vol. 9, no. 1, Dec. 2022, doi: 10.1186/s40537-022-00597-0.

[17] A. F. Kamara, E. Chen, and Z. Pan, “An ensemble of a boosted hybrid of deep learning models and technical analysis for forecasting stock prices,” *Inf. Sci. (N. Y).*, vol. 594, pp. 1–19, May 2022, doi: 10.1016/j.ins.2022.02.015.

[18] J. C. King and J. M. Amigó, “Integration of LSTM Networks in Random Forest Algorithms for Stock Market Trading Predictions,” *Forecasting*, vol. 7, no. 3, Sep. 2025, doi: 10.3390/forecast7030049.

[19] M. S. Islam and E. Hossain, “Foreign exchange currency rate prediction using a GRU-LSTM hybrid network,” *Soft Computing Letters*, vol. 3, p. 100009, Dec. 2021, doi: 10.1016/j.socl.2020.100009.

[20] Z. A. Hassaan, M. H. Yacoub, and L. A. Said, “Gated recurrent unit accelerators for financial time series prediction on field-programmable gate array,” *Eng. Appl. Artif. Intell.*, vol. 162, Dec. 2025, doi: 10.1016/j.engappai.2025.112534.

[21] Z. Li, Y. Wang, J. Cao, and C. Huang, “A hybrid prediction model for stock trend based on gated recurrent unit and wavelet transform,” *Applied Intelligence*, vol. 55, no. 12, Aug. 2025, doi: 10.1007/s10489-025-06702-z.

[22] N. Mabrouk, M. Chihab, and Y. Chihab, “Intraday Trading Strategy based on Gated Recurrent Unit and Convolutional Neural Network: Forecasting Daily Price Direction,” 2022. doi: https://dx.doi.org/10.14569/IJACSA.2022.0130369.

[23] S. Enkhbayar and R. Ślepaczuk, “Predictive modeling of foreign exchange trading signals using machine learning techniques,” *Expert Syst. Appl.*, vol. 285, Aug. 2025, doi: 10.1016/j.eswa.2025.127729.

[24] K. Yan *et al.*, “Flexible Target Prediction for Quantitative Trading in the American Stock Market: A Hybrid Framework Integrating Ensemble Models, Fusion Models and Transfer Learning,” *Entropy*, vol. 28, no. 1, p. 84, Jan. 2026, doi: 10.3390/e28010084.

[25] B. F. Tanyu, A. Abbaspour, Y. Alimohammadlou, and G. Tecuci, “Landslide susceptibility analyses using Random Forest, C4.5, and C5.0 with balanced and unbalanced datasets,” *Catena (Amst).*, vol. 203, Aug. 2021, doi: 10.1016/j.catena.2021.105355.

[26] S. Latif, N. Javaid, F. Aslam, A. Aldegheishem, N. Alrajeh, and S. H. Bouk, “Enhanced prediction of stock markets using a novel deep learning model PLSTM-TAL in urbanized smart cities,” *Heliyon*, vol. 10, no. 6, Mar. 2024, doi: 10.1016/j.heliyon.2024.e27747.

[27] N. Sukma and C. S. Namahoot, “An Algorithmic Trading Approach Merging Machine Learning With Multi-Indicator Strategies for Optimal Performance,” *IEEE Access*, vol. 12, pp. 188154–188173, 2024, doi: 10.1109/ACCESS.2024.3516053.

[28] A. Syukur and D. Istiawan, “Prediction of LQ45 Index in Indonesia Stock Exchange: A Comparative Study of Machine Learning Techniques,” *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 1, pp. 453–463, 2020, doi: 10.22266/IJIES2021.0228.42.

.

[29] Indra *et al.*, "Stock Return Prediction on the LQ45 Market Index in the Indonesia Stock Exchange Using a Machine Learning Algorithm Based on Technical Indicators," *Journal of Risk and Financial Management*, vol. 18, no. 12, Dec. 2025, doi: 10.3390/jrfm18120714.

[30] M. S. Alizza and T. Nirmala, "DETERMINAN YIELD OBLIGASI PEMERINTAH INDONESIA," *Indonesian Journal of Economy, Business, Entrepreneurship and Finance*, Aug. 2025, doi: 10.53067/ijebef.