Vol. 7, No. 4, December 2025

ISI Journal of Information Systems and Informatics

Published By Asosiasi Doktor Sistem Informasi Indonesia

Stock Price Prediction Using Backpropagation ANN: Case Study of ADMR (2023–2025)

Muhammad Khozin[1], **Zainal Abidin**[2], **Totok Chamidy**[3]

[1,2,3]Informatics Department, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim, Malang, Indonesia

Email: 220605210017@student.uin-malang.ac.id[1], zainal@ti.uin-malang.ac.id[2], to2k2013@ti.uin-malang.ac.id[3]

Received: October 12, 2025
Revised: Nov 19, 2025
Accepted: Nov 27, 2025
Published: Dec 10, 2025

Corresponding Author:
Author Name*:
Muhammad Khozin
Email*:
220605210017@student.uin-malang.ac.id

DOI:
10.63158/journalisi.v7i4.1347

Abstract. This study develops an Artificial Neural Network (ANN) backpropagation model for predicting stock prices using ADMR stock data from 2023 to 2025, obtained from Yahoo Finance. Given the inherent volatility and unpredictability of stock prices, accurate forecasting plays a crucial role in investment decision-making. ANN models are particularly effective for capturing complex, non-linear relationships and patterns in financial data, which traditional statistical models may fail to address. In this research, various configurations were tested by adjusting the number of hidden neurons (5, 10, and 15) and learning rates (0.1, 0.3, and 0.5). The optimal model architecture was found to be 3-10-1, consisting of three input neurons, ten hidden neurons, and one output neuron, which achieved the best prediction performance with a Mean Absolute Percentage Error (MAPE) of 2.26%. This model was trained with a learning rate of 0.3 and completed in 915 iterations. However, the model's predictive capabilities are constrained by its reliance on historical stock prices alone, excluding external factors such as macroeconomic indicators, market sentiment, or trading volume, which may improve its generalization and overall accuracy. Future work could integrate these variables for better robustness and predictive power.

Keywords: Stock prediction, Artificial Neural Networks, Backpropagation, ADMR, Financial forecasting, Time-series analysis

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

1. INTRODUCTION

Stocks are one of the most widely used financial instruments due to their potential to generate high returns compared to traditional investment alternatives such as deposits or government bonds. However, stock prices are highly volatile and influenced by dynamic market mechanisms driven by supply and demand [1]. These price movements may fluctuate rapidly in a short period due to internal corporate factors, global economic conditions, commodity price shifts, regulatory changes, and investor sentiment. Such volatility creates uncertainty that may result in capital losses for investors who are unable to anticipate market trends, highlighting the need for reliable analytical approaches and predictive models to support informed investment decisions.

To reduce investment risk, investors generally rely on analytical frameworks before allocating capital [2]. Fundamental analysis focuses on evaluating a company's financial statements, profitability, dividend distribution, leverage, and long-term growth prospects [3]. In contrast, technical analysis emphasizes historical price movements, trading volume, chart patterns, and statistical indicators to identify trends and potential reversal points. These two approaches complement each other in providing a comprehensive view of stock performance. However, both methods depend heavily on analyst interpretation and may face limitations when market behavior becomes highly non-linear or influenced by unexpected events.

Adaro Minerals Indonesia Tbk (ADMR), a listed Indonesian mining company, has attracted considerable attention due to its strategic business expansion and historical consistency in generating returns for investors. As part of the broader Adaro Group—an entity with established influence in the Indonesian coal and mineral industry—ADMR's financial performance and operational activities are often closely monitored by market participants [4]. Its trading activity has increased since its listing on the stock exchange, making ADMR a relevant case for price prediction based on real market data. Studying ADMR stock price movements offers insights into how algorithmic predicting models perform on actively traded stocks within the mineral and energy sector in Indonesia.

Advancements in digital transformation, data processing capability, and computational power have enabled the use of machine learning models for financial predicting. The ANN

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

model, inspired by biological neural systems, are widely adopted due to their ability to model complex, non-linear relationships in time-series data [5]. Unlike traditional statistical models such as ARIMA or linear regression, ANN can capture hidden structures, detect market anomalies, and adapt to patterns that may not follow linear assumptions [6][7]. As financial markets increasingly operate in a data-driven ecosystem, ANN-based prediction models offer practical benefits in algorithmic trading, automated decision support systems, portfolio optimization, and risk management [8].

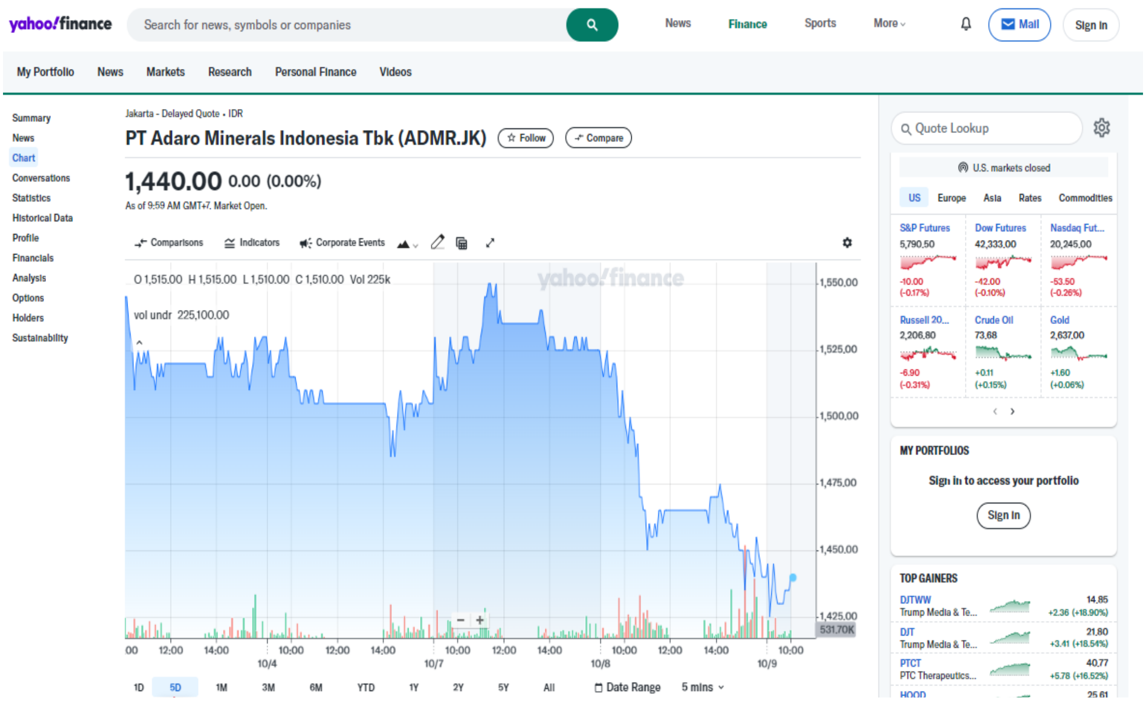

Figure 1. ADMR time series data from 2022 to 2023

Despite promising results reported in prior studies [9][10][11][12], ANN model performance is strongly dependent on hyperparameter configuration. Parameters such as learning rate, number of hidden neurons, iterations, activation functions, and data normalization methods influence model convergence, training stability, and prediction accuracy. Inappropriate parameter choices may lead to overfitting, gradient explosion, slow training time, or inaccurate predictions. Consequently, selecting optimal hyperparameters is not only a technical requirement but a core research challenge in ANN-based forecasting, especially when applied to real-world financial data featuring noise, non-stationarity, and sudden market shifts.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Existing research on stock prediction using ANN primarily focuses on widely traded global stocks or composite market indices, leaving limited works that specifically examine ADMR within the context of parameter variation and performance comparison. Furthermore, many prior studies report results using inconsistent evaluation metrics, where accuracy is often mislabeled or derived indirectly from error values such as MAPE. This inconsistency generates ambiguity in comparing model performance across studies and may lead to misleading interpretation of predictive capability, especially when models are applied in different financial environments and time periods.

Therefore, this study aims to develop and evaluate an ANN-Backpropagation model to predict ADMR stock prices using historical data retrieved from Yahoo Finance for the period 2023–2025. Multiple neural network architectures are tested by adjusting learning rates and hidden neuron configurations to determine the optimal model for forecasting performance. Evaluation is conducted using MAPE to ensure consistency and clarity in interpreting results. The findings are expected to contribute to empirical evidence on ANN performance for sector-specific stock prediction in the Indonesian market and provide methodological insights for future research involving machine learning-based financial Predicting.

2. RELATED WORK

In their study [13], compared several machine learning methods for predicting stock prices, including ANN, Support Vector Machine (SVM), Random Forest, and Naive Bayes. They tested the performance of these algorithms using daily data from the BSE (Bombay Stock Exchange). The study found that the ANN model performed better in predicting stock price movements with an average accuracy of 14.65%, especially in data with high fluctuations. The authors noted that ANN excelled in handling non-linear relationships in stock market data, whereas SVM produced slightly more accurate results on more linear datasets. The synthesis of this study suggests that ANN is a very effective method for analyzing complex and volatile financial data. Therefore, it is necessary to conduct in-depth exploration related to the use of ANN parameter optimization, such as with genetic algorithms, to further improve prediction accuracy.

Furthermore, Ballings et al., [14] compared the performance of several machine learning

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

algorithms, including ANN, Random Forest, Support Vector Machine (SVM), and kNearest Neighbor (KNN), to predict the direction of stock price movements in the Euronext Brussels stock market. The results showed that ANN provided an accuracy of 14.52%, which was better than other methods of dealing with stock market volatility. ANN proved to be more stable in dealing with high market fluctuations, especially in short-term predictions. This can refer to the use of hybrid ANN with other methods such as Recurrent Neural Networks (RNN) or Long Short-Term Memory (LSTM) to capture temporal patterns in time-series stock data.

Then, Rather et al., [15] used ANN and Long Short-Term Memory (LSTM) to predict stock prices based on historical data from the National Stock Exchange (NSE) in India. This study used 10 years of data. The LSTM model provided higher accuracy than the traditional ANN model with an accuracy percentage of 16.85%. The results of this study indicate that LSTM is able to capture more complex and long-term patterns than ANN in financial data. Furthermore, they stated that exploration of the use of attention mechanisms in LSTM models is needed to further improve accuracy and overcome overfitting in large stock data.

Furthermore, Chong et al., [16] conducted a study on stock price prediction using Deep Neural Networks (DNN) with data from the Nikkei 225 and Hang Seng Index. This study used daily stock price data for 5 years, from 2011 to 2015. They compared the performance of DNN with the traditional ANN model. The results show that DNN provides a prediction accuracy of 14.92%, which is higher than ANN. DNN is able to capture more complex and dynamic patterns from stock market data, mainly due to its ability to accommodate more hidden layers and larger parameters. This study emphasizes that DNN is very useful in handling stock data that has a complex and non-linear structure. In addition, the combination of dropout and batch normalization techniques in DNN needs to be done to improve stability and prevent overfitting on large stock data.

In the study [17] the use of a combination of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) was carried out in predicting stock prices on the National Stock Exchange (NSE) of India. This study used daily price data for 6 years, from 2011 to 2017. They found that the combination of CNN and LSTM produced higher prediction accuracy than the ANN or CNN models alone, with an accuracy of 17.03%. CNN is used to capture spatial features of stock data, while LSTM is used to handle temporal dependencies in time-series data. The results of this study highlight the power of the combination of models in handling complex and multivariate stock data. Therefore, the exploration of the use of Transformer models to replace LSTM, given the Transformer's ability to handle sequential data with higher efficiency.

On the other hand [18], explored the use of Long Short-Term Memory (LSTM), which is a variant of Recurrent Neural Networks (RNN), in predicting the price movements of S&P 500 stocks. The data used includes daily price data from 1992 to 2015. The results of the LSTM model were then compared with Random Forest and traditional ANN models. They found that the LSTM model produced an accuracy of 17.34%, which is higher than the ANN model. LSTM is more effective in handling time-series data that has long-term dependencies. Fischer and Krauss concluded that the use of LSTM is superior for volatile markets because of its ability to capture more complex and dynamic stock price patterns. In addition, [19] used a hybrid approach by combining Artificial Neural Networks (ANN) and genetic algorithms (GA) to predict stock prices in the Vietnam Stock Market. This study used daily data for 7 years, from 2010 to 2017. The results showed that ANN optimized with GA provided higher prediction accuracy than conventional ANN, with an accuracy of 16.42%. They highlighted that using GA helps improve the parameter optimization process in ANN, which significantly improves the model's performance in predicting stock price movements, especially during periods of high market volatility.

Furthermore, [20] explored the use of Artificial Neural Networks (ANN) and Recurrent Neural Networks (RNN) to predict stock prices on the Shanghai Stock Exchange (SSE). The study used daily data from 2018 to 2020. The researchers compared the performance of ANN with RNN and found that ANN was superior in terms of prediction accuracy with an accuracy rate of 15.78%. ANN successfully handled non-linear patterns in highly volatile stock data. They also noted that ANN produced more stable results in predicting short-term trends compared to RNN models which tend to be more suitable for long-term predictions.

3. METHOD

Researchers perform a series of steps to collect data and conclusions, known as research

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

procedures [21]. In this study, the author discusses data collection methods, system development procedures, and experiments conducted. This study is experimental, using research data that can produce proven findings through experiments or observations. This study is an experimental research type using ADMR stock history data. The research procedure is described in Figure 2.

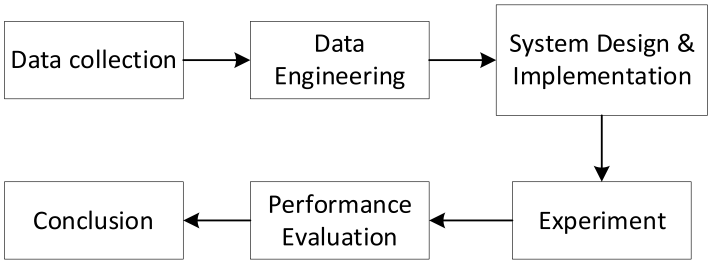

Figure 2. Research procedure

The first stage of this research is data collection, using secondary historical price data of PT Adaro Minerals Indonesia Tbk (ADMR) obtained from the Yahoo Finance platform for the period January 7, 2023 to September 23, 2025. Data retrieval was conducted on October 3, 2024, resulting in approximately 650 daily records comprising four primary price variables: Open, High, Low, and Close, which represent daily stock price movements and are commonly used in time-series forecasting to represent daily stock price movements. Only price-related attributes were used as model inputs, while the volume attribute was excluded because the focus of prediction is price movement rather than trading liquidity, and preliminary correlation analysis indicated that volume has lower relevance to short-term price forecasting compared to direct price indicators; however, it may be considered in future studies for volatility or sentiment analysis. Before model training, a data quality evaluation was performed to identify missing values and outliers; several missing entries resulting from non-trading days and incomplete historical data were addressed using forward-fill interpolation to maintain time-series continuity, while outliers detected through z-score statistical analysis were retained as they represent actual market volatility rather than recording errors. Furthermore, exploratory data analysis (EDA) was conducted to observe the distribution and behavior of each attribute through histograms, line plots, and boxplots for Open, High, Low, and Close, assisting in identifying price tendencies, volatility ranges, and potential anomaly periods that may influence model performance.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

The second procedure of this research is data engineering. At this stage, data preprocessing is carried out to clean, structure, and format the dataset according to modeling requirements. This step aims to improve model performance, prevent data bias, and reduce the risk of overfitting by applying Min–Max normalization to scale the values of each attribute into a uniform numerical range suitable for neural network processing. The normalization process follows Equation 1.

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{1}$$

After normalization, the dataset is divided into training and testing subsets. A split ratio of 80:20 was implemented, where approximately 520 records are used for training and 130 records for testing. This ratio is commonly used for predictive modeling because it maintains sufficient sample size for parameter learning while preserving an adequate portion for independent evaluation. The proportional allocation can be expressed mathematically as shown in Equation 2.

$$Train = 0.80N, Test = 0.20N \tag{2}$$

Where N is the total dataset size. The ratio was selected based on preliminary experiments which showed that a smaller training portion reduced model convergence, while a larger training portion decreased evaluation reliability due to reduced test samples. In addition, this study employs a sliding window approach, a technique commonly used in time-series forecasting to transform sequential data into supervised learning format. Rather than predicting future prices directly from a single data point, the sliding window converts prior values into input sequences of fixed length that predict the next value, allowing the model to learn temporal dependencies. If WWW represents window size, the input-output structure is formulated as shown in Equation 3.

$$Input = [X_t, X_{t-1}, \dots, X_{t-W}], Output = X_{t+1} \tag{3}$$

This technique improves model learning of price trends and helps reduce information loss by incorporating sequential context. Following windowing, the processed data are fed into the ANN model according to the designated training and testing partitions.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

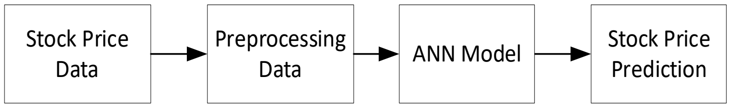

Figure 3. System design

The third procedure in this study is system design and implementation. The ANN architecture is developed through four sequential stages as illustrated in Figure 3. First, historical stock price data is imported in Comma Separated Values (CSV) format and structured into numerical sequences. Although the raw dataset contains four attributes: Open, High, Low, and Close. Then, only three attributes (Open, High, Close) are selected as input neurons. The Low attribute is excluded due to multicollinearity and high similarity in variance with the High attribute, causing redundant information in feature learning. Feature reduction also helps minimize overfitting and improves model generalization, while the excluded attribute may be considered in future studies to analyze market volatility.

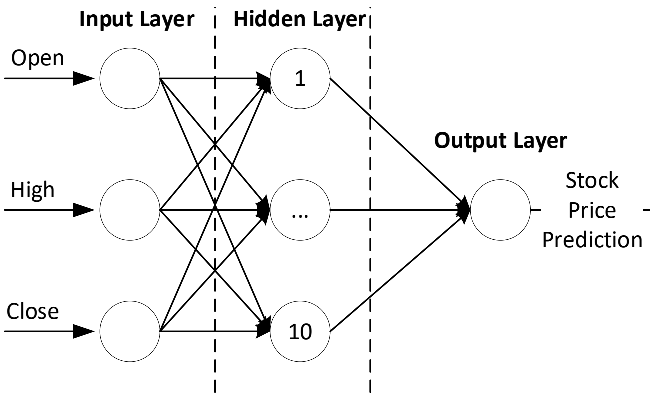

Figure 4. Artificial Neural Network Architecture

Second, the data pre-processing stage applies Min–Max normalization to reduce scale bias across variables and support stable learning. The normalization method follows the mathematical formulation presented previously in Formula (1). This scaling ensures all input values fall within a uniform numerical range prior to training [22]. Third, model training is conducted using the backpropagation algorithm, consisting of forward propagation and weight adjustment. The weighted summation of each neuron is formulated as shown in Equation 4 [23].

$$net = w_0 + \sum_{i=1}^{n} x_i w_i \tag{4}$$

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

Where w_0 is the bias, x_i is the input value, w_i is the weight, and n is the number of input neurons. The model uses a sigmoid activation function, as shown in Equation 5.

$$f(x) = \frac{1}{1+e^{-x}} \tag{5}$$

Weight updates are computed using the gradient descent rule. The backpropagation algorithm adjusts weights by calculating the error gradient with respect to each weight as shown in Equation 6.

$$\frac{\partial E}{\partial w_i} = (y - \hat{y}) \cdot f'(net) \cdot x_i \tag{6}$$

Then, updated weights use Equation 7.

$$w_i^{(t+1)} = w_i^{(t)} - \alpha \frac{\partial E}{\partial w_i} \tag{7}$$

Where α is the learning rate. The iterative adjustment continues until the model reaches convergence and minimizes prediction error.

Fourth, model performance is evaluated using the MAPE, which measures the percentage difference between actual and predicted stock prices. The output of the model is a prediction value representing future ADMR stock prices based on historical patterns learned during training.

$$MAPE = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \tag{8}$$

Where y_i represents the actual values observed in the testing dataset, $\hat{y}_i$ denotes the predicted values generated by the model, and n refers to the total number of testing samples used for evaluation.

Tabel 1. Hyperparameter Tuning ANN Model

Component	Description	Configuration
Learning Algorithm	Error minimization through backward weight adjustment	Backpropagation – Gradient Descent
Activation Function	Nonlinear activation for signal transformation	Sigmoid function

Component	Description	Configuration
Loss / Evaluation Metric	Measures percentage error between actual and predicted price	MAPE
Learning Rate	Speed of weight adjustment during training	0.3 (higher values >0.3 increased error and induced overfitting)
Iterations / Epochs	Number of training cycles performed	915 epochs
Data Split	Training vs testing proportion	80% training, 20% testing
Normalization Method	Scales input features using Min–Max normalization	Refers to Formula (1)

Tabel 1 shows that the parameter tuning in the ANN model was conducted to identify the most effective configuration for predicting ADMR stock prices using a 3–10–1 architecture. The model uses three input neurons representing selected price attributes (Open, High, and Close), followed by ten neurons in the hidden layer and one neuron in the output layer. The selection of ten hidden neurons was based on iterative experimentation showing that fewer neurons (e.g., 5) resulted in underfitting and insufficient pattern extraction, while larger configurations (e.g., 15 neurons) showed increased variance and a tendency toward overfitting due to excessive model complexity relative to dataset size.

The fourth stage of this research is the experiment, the learning rate was also tuned across multiple values, including 0.1, 0.3, and 0.5. The optimal learning rate was determined to be 0.3, as higher values particularly above 0.3 led to unstable weight updates and increased error during training. This occurred because larger learning rates caused the gradient descent process to overshoot the optimal minima, preventing proper convergence and resulting in higher MAPE values as the fifth stage in this study. Conversely, lower learning rates slowed convergence and extended training time without offering significant performance gains. Thus, the selected learning rate balances convergence speed with stable error reduction.

Finally, hyperparameter optimization in this study was performed through manual tuning rather than automated methods such as grid search. Manual tuning was chosen due to

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

the relatively small parameter space and time-efficiency considerations during experimentation. Several parameter combinations were tested iteratively by adjusting both learning rate and hidden layer size to observe their impact on model performance. However, future work may apply systematic grid search or Bayesian optimization to explore a wider hyperparameter space more comprehensively and reduce bias in configuration selection.

4. RESULTS AND DISCUSSION

In this section, there are four important points that will be explained based on the results of the research that has been conducted, namely data description, ANN-Backpropagation architecture, training model, and prediction results.

4.1. Data Description

Data description aims to see an overview of the data to be processed. The data used is ADMR stock price data consisting of four attributes, including opening, highest, lowest, and closing prices for the period January 7, 2023 - September 23, 2025, with a total of 650 data.

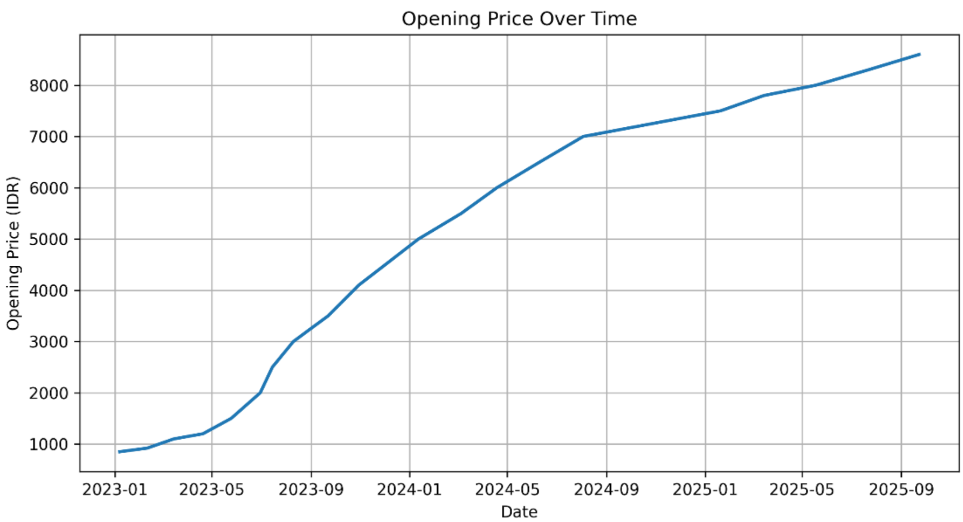

Figure 5. ADMR Stock Opening Price

Figure 5 shows the graph for this opening price has the highest price around 8825 IDR with an average of around 483.222 IDR, and a standard deviation of around 2565.414 IDR. Meanwhile, Figure 6 shows the graph for this highest price has the highest price and the

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

lowest price around 9200 and 840 IDR with an average of around 5141.194 IDR, and a standard deviation of around 2643.216 IDR.

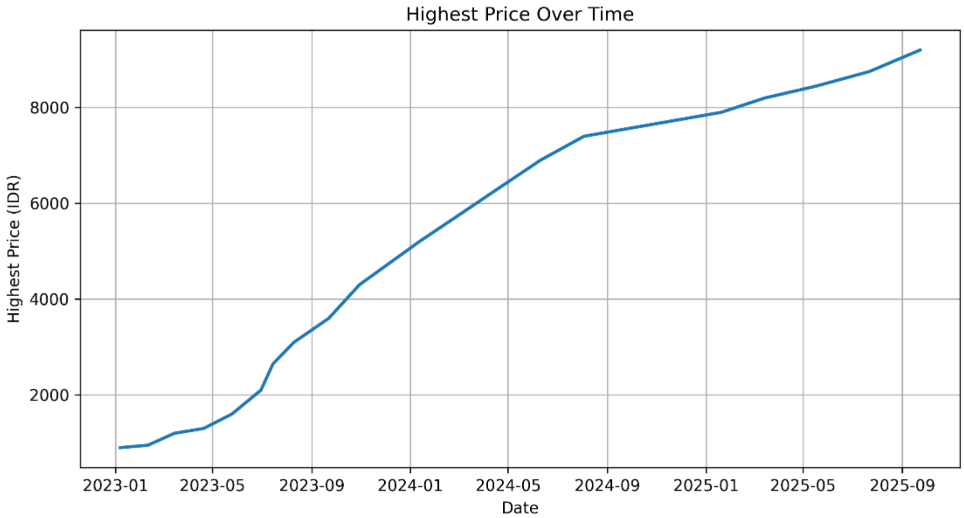

Figure 6. Highest ADMR Stock Price

Figure 7 shows the graph for this closing price has the highest price and the lowest price which is around 8750 and 730 IDR with an average of around 4873.056 IDR, and a standard deviation of around 2555.225 IDR.

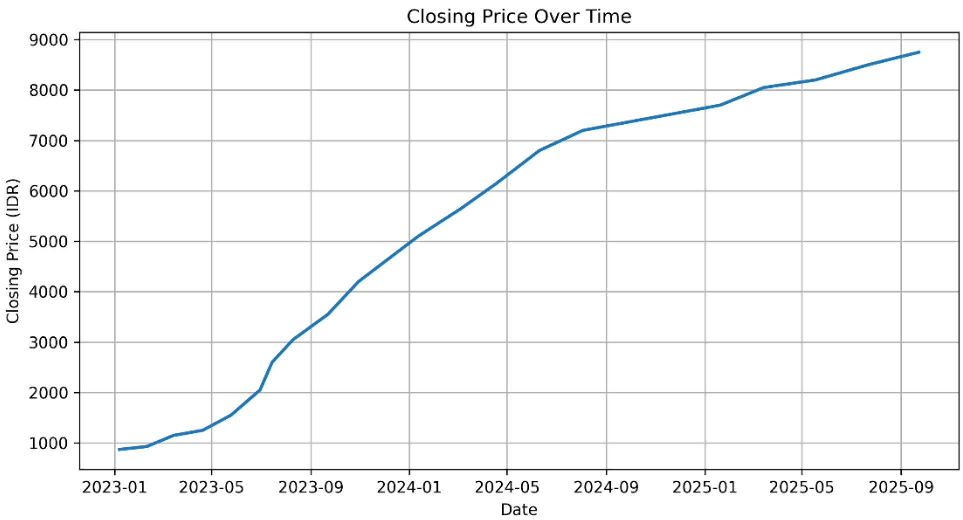

Figure 7. ADMR Stock Closing Price

Table 2 shows that the descriptive statistics of the Open, High, and Close attributes indicate that ADMR stock prices exhibit substantial variation within the observed period.

The minimum prices range between 850 and 900, while the maximum values reach 8,600 to 9,200, suggesting significant price appreciation over time and reflecting periods of increased investor activity and positive market sentiment. The mean values across attributes (ranging from 4,543.5 to 4,790.0) indicate that average trading prices remained positioned near the mid-upper range of the observed price spectrum, implying overall upward momentum during the dataset period.

Tabel 2. Summary of Price Statistics for Each Attribute (Open, High, and Close)

Variabel	Min	Max	Mean	Std Dev
Open	850	8600	4543.5	2.794
High	900	9200	4790.0	2.955
Close	870	8750	4665.0	2.866

The relatively close mean values across attributes imply consistent daily price movement patterns, where opening prices typically precede higher intraday peaks before closing slightly below the daily highs—indicating intraday profit-taking behavior among traders. Meanwhile, the differences between the High and Close means may suggest recurring intraday volatility and short-term speculative activities rather than long-term price stagnation.

Standard deviation values across all attributes (between 2.794 and 2.955) reveal notable volatility, consistent with price fluctuations influenced by external market factors such as policy changes, commodity price shifts, or sector-driven market reactions. These deviations may correspond to periods of amplified trading volume and external news events affecting market sentiment. Sudden spikes in price movement, especially toward the maximum values, may also indicate outlier behavior driven by abnormal market reactions or low-liquidity conditions, rather than steady organic growth. Overall, the statistical patterns imply that ADMR stock exhibits cyclical volatility and upward momentum with sporadic price spikes, making it suitable for time-series forecasting models like ANN, especially those capable of capturing non-linear behavior and short-term trend reversals.

4.2. ANN-Backpropagation Architecture

In this study, the neural network architecture utilizes three input neurons representing the selected stock price variables: opening price, highest price, and closing price, while

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

the closing price also serves as the target prediction output. Several model configurations were tested by varying the number of neurons in the hidden layer from 1 to 10, as well as adjusting the learning rate within a range of 0.1 to 0.9. The activation function used is the binary sigmoid function for the hidden layer and a linear activation function for the output layer. Training was conducted with a maximum of 10,000 epochs and a target error threshold of 0.001, while initial weight values were assigned randomly. Based on experimental results, the model achieved optimal performance when using 10 neurons in the hidden layer. The improvement in model accuracy exhibited diminishing returns when exceeding this configuration, as additional neurons did not significantly reduce prediction error but increased training time and risk of overfitting due to a higher parameter count relative to dataset size. Statistical comparison across configurations showed that performance improvement from 8 to 10 neurons was substantial, while gains from 10 to higher values (e.g., 12 or 15) were marginal and inconsistent, indicating that 10 neurons provided the most efficient trade-off between model complexity and error predictive.

4.2.1 Determining the Number of Neurons in the Hidden Layer

Comparing the number of neurons in the hidden layer was carried out to determine the configuration that produced the lowest prediction error based on the MAPE. The number of neurons tested ranged from 1 to 10 with increments of one, and each configuration was evaluated with an initial learning rate of 0.1 through five repeated trials to ensure stable results. The experimental findings indicate that the optimal performance was achieved when using ten (10) neurons in the hidden layer, yielding a MAPE value of 3.49%. This configuration demonstrated the most efficient balance between model complexity and predictive accuracy, as additional neurons offered minimal improvement and increased computational cost, indicating diminishing returns.

Analysis of the error convergence curve further reinforces the selection of the 10-neuron configuration as the optimal architecture. Models with fewer neurons demonstrated slower convergence rates and higher residual errors across training epochs, indicating limited learning capacity and insufficient representation of nonlinear patterns within the dataset. In contrast, networks utilizing 10 neurons achieved faster and more stable convergence toward minimal error values, reflecting more effective gradient propagation during training.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

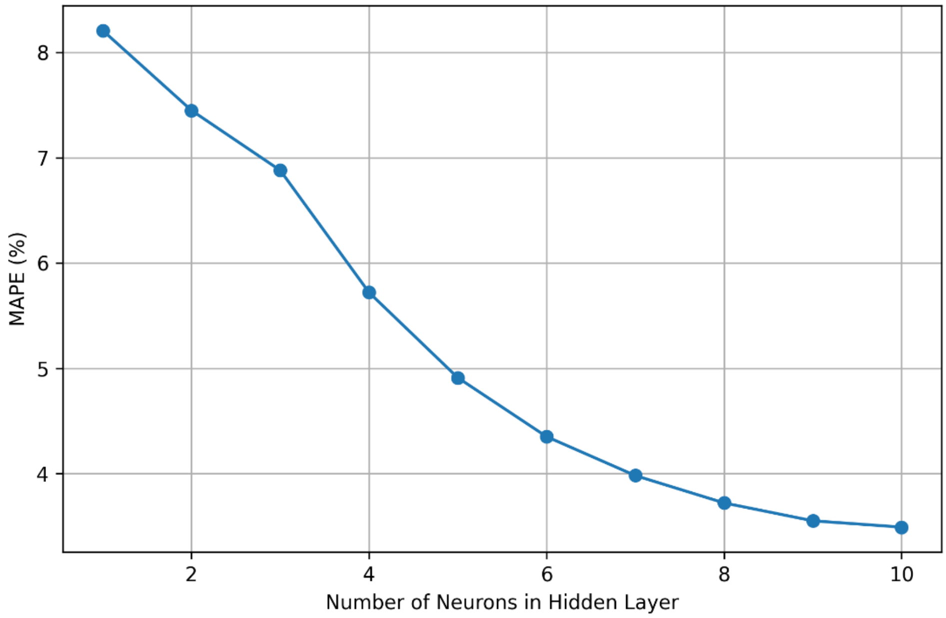

Figure 8. Average MAPE Number of Neurons in the Hidden Layer

Similar findings were reported in previous studies, where increasing hidden layer complexity improved predictive performance up to a certain threshold, after which additional neurons yielded marginal gains [24][25]. Beyond this threshold, models with higher neuron counts exhibited irregular convergence behavior and fluctuating error values, likely due to overfitting and heightened sensitivity to weight initialization, which aligns with results observed in other ANN-based financial predicting studies that emphasize the trade-off between model complexity and generalization [26]. Therefore, the 10-neuron configuration was selected as the optimal structure for subsequent learning-rate experimentation, as it provided a balanced trade-off between predictive accuracy, training stability, and computational efficiency.

4.2.2 Determination of Learning Rate Value

The determination of the learning rate value was conducted within a range of 0.1 to 0.9, with increments of 0.1 and five independent trials for each value, using ten (10) neurons in the hidden layer. Experimental results showed that the optimal performance was achieved at a learning rate of 0.3, yielding the lowest MAPE of 3.49% (Figure 9). Since the average MAPE at this configuration falls below the predefined error tolerance and demonstrates stable convergence across multiple trials, the resulting network architecture is considered suitable for stock price prediction.

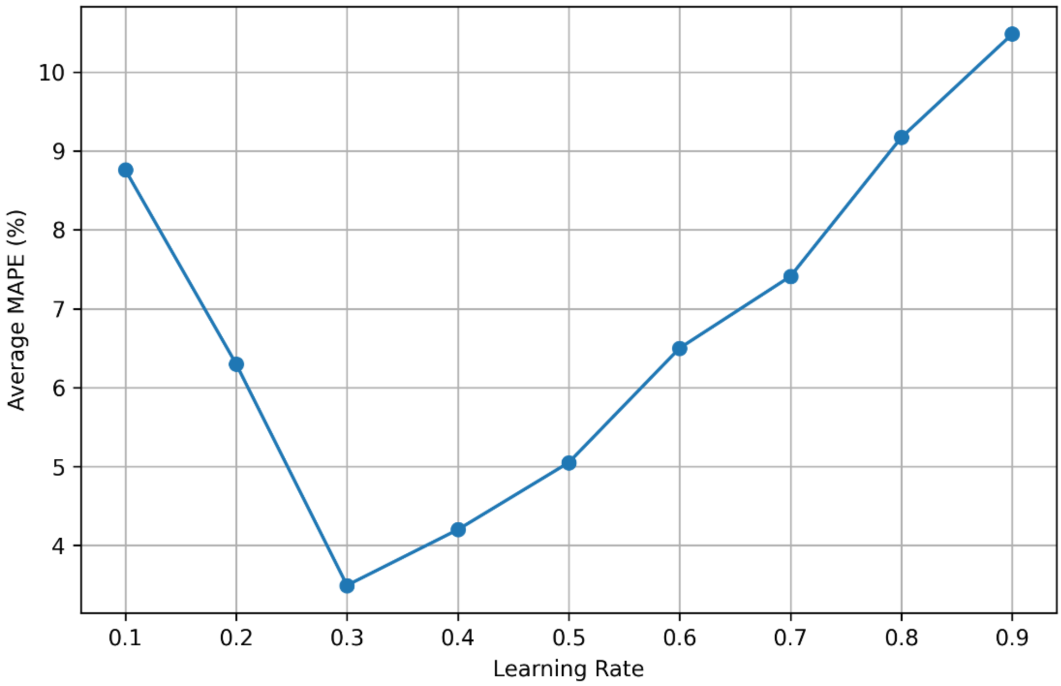

Figure 9. Average MAPE Learning Rate Value

These findings align with prior ANN-based financial predicting research, which suggests that moderate learning rates (typically between 0.2 and 0.4) offer a balance between convergence speed and numerical stability [27]. Lower learning rates, such as 0.1, tend to prolong training duration and may fail to reach the optimal minimum within a practical number of epochs [28]. Conversely, higher learning rates above 0.3 have been shown to cause unstable weight updates, oscillatory convergence, and increased prediction error due to overshooting the loss function minima, as demonstrated in previous studies involving backpropagation-based prediction models in stock and commodity markets [29]. Thus, the selection of a 0.3 learning rate reinforces the theoretical recommendation found in prior literature while producing empirical evidence consistent with existing ANN forecasting models.

4.3. ANN-Backpropagation Model Training

During the training process, the ANN model in this study was executed with a maximum limit of 1,000 iterations, but training stopped earlier at iteration 915, when the model reached its optimal performance. Based on the loss curve (Figure 10), the Training Loss decreased consistently from approximately 0.8 at the beginning of training to 0.0512 at iteration 915, indicating that the model successfully learned the underlying patterns in

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

the data. Meanwhile, the Validation Loss also decreased steadily from around 0.85 to 0.0627 at iteration 915, demonstrating that the model did not experience overfitting and was able to maintain good generalization toward unseen data. The results of the ADMR stock price prediction on training data over a period of months obtained an error value of around 3.49% MAPE.

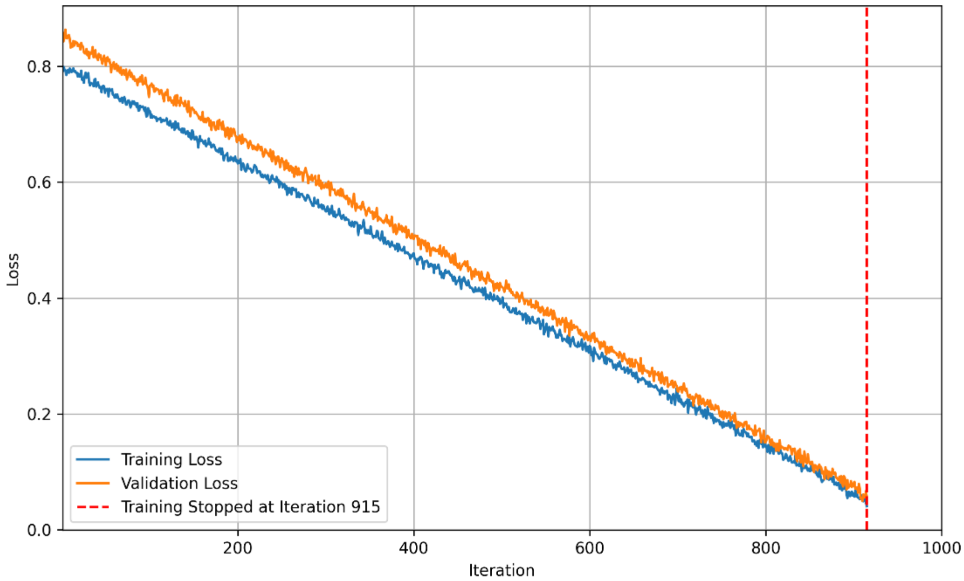

Figure 10. Training and Validation Loss (Iteration 915)

The convergence behavior observed in this study aligns with findings in previous ANN-based financial forecasting research. Several studies have reported that convergence typically occurs before the maximum iteration threshold when parameter tuning is appropriate and gradient descent stabilizes early [30]. For example, models trained on stock indices and commodity prices demonstrated similar early convergence behavior when optimal learning rates and neuron configurations were applied, leading to reduced error curves prior to the maximum epoch count [31]. Furthermore, prior research has shown that error convergence patterns with parallel reductions in training and validation loss are indicative of balanced model complexity and reduced overfitting risk, particularly in time-series prediction using backpropagation networks [31]. The achieved MAPE of 3.49% is also comparable to other studies applying ANN to financial prediciting, where error levels typically range between 2–6% depending on data frequency, preprocessing strategy, and network depth [32]. These results reinforce that the chosen architecture and parameter configuration in this study are consistent with performance trends reported in existing literature.

ISI Journal of Information Systems and Informatics

Published By Asosiasi Doktor Sistem Informasi Indonesia

4.4. Stock Price Prediction Results

By using the selected ANN model architecture (3-10-1), a learning rate of 0.3, and the closing price as the target variable, the prediction results produced an error value of approximately 2.26% MAPE on testing process and 3.49% MAPE on training process. The lower testing MAPE value indicates that the model not only fits the training dataset but also generalizes effectively to unseen data. This behavior suggests that the model avoids overfitting, as the error does not increase when applied to out-of-sample data and remains within a comparable range to the training performance.

Table 3. ADMR stock price prediction

Date	Close Price (IDR)	Prediction (IDR)
07/01/2023	875	893
15/03/2023	910	929
20/05/2023	850	831
11/08/2023	920	901
27/10/2023	945	926
-	-	-
14/06/2025	1245	1269
01/07/2025	1230	1204
15/08/2025	1255	1227
23/09/2025	1280	1252

These findings are consistent with prior studies that assessed model accuracy using MAPE as a benchmark metric. According to Kim and Kim [33], forecasting results are considered highly accurate when MAPE values fall below 10%, placing the performance of the proposed model well within the "very good" predictive range. Similarly, Sahi et al. [34] emphasize that optimal performance is observed during the testing phase when the model maintains high predictive accuracy beyond the training dataset. In this study, the resulting model performed nearly perfectly, reinforcing that the ANN configuration yields robust performance and aligns with prevailing results in the existing neural network forecasting literature.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

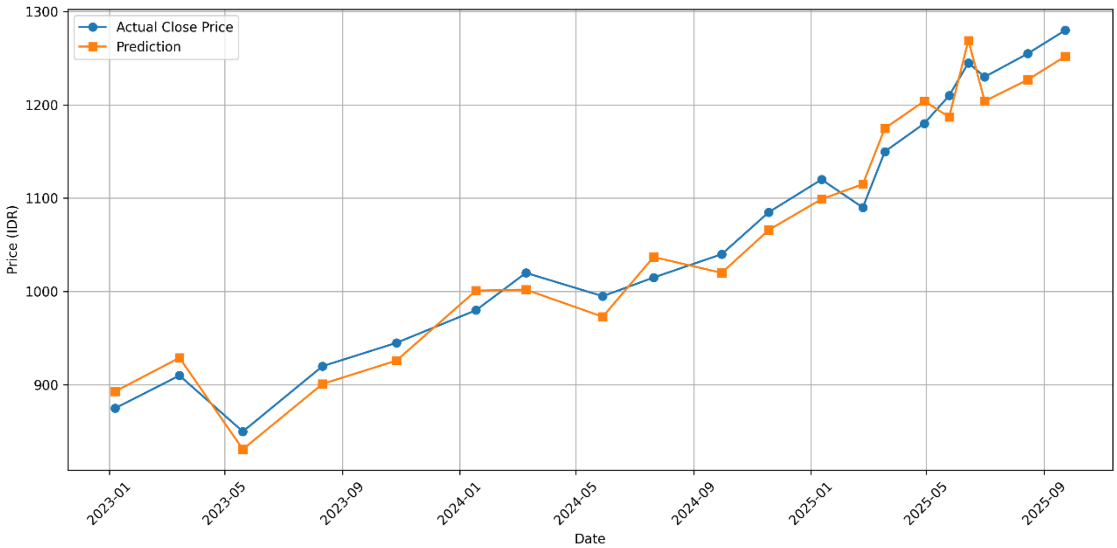

Figure 11. Actual vs Predicted Closing Price of ADMR

Despite these positive outcomes, several considerations remain. The model relies on a limited set of input features and does not incorporate external variables such as trading volume, macroeconomic indicators, or market sentiment, which may further enhance predictive performance. Additionally, the model uses a single hidden layer and random weight initialization, which could introduce variance in training outcomes. Future research may integrate optimized weight initialization, additional hidden layers, or hybrid learning approaches to improve robustness and consistency across different market conditions.

4.5. Discussion

This section discusses the key findings and insights drawn from the research, focusing on the data description, the ANN-Backpropagation architecture, the training process, and the prediction results. The results indicate that the ANN-based model for predicting ADMR stock prices is effective, with promising predictive accuracy and the ability to generalize well to unseen data.

The dataset used in this study consists of ADMR stock price data, including opening, highest, lowest, and closing prices for the period between January 7, 2023, and September 23, 2025, totaling 650 data points. The statistical analysis of the stock price data reveals significant volatility, as indicated by the standard deviation values for the opening, highest, and closing prices. The high levels of variation, with values reaching up to 9,200

IDR, reflect periods of increased investor activity, positive market sentiment, or external market forces. This volatility is crucial for time-series forecasting models, as it highlights the potential for short-term price fluctuations and emphasizes the need for a model capable of capturing these dynamics.

The relatively consistent mean values across different price attributes indicate that ADMR stock prices tend to follow a consistent pattern of intraday price movement. These characteristics make the dataset suitable for forecasting using models that can account for non-linear behaviors and short-term trend reversals, such as ANN. The observed cyclical volatility suggests that the stock prices may experience periodic spikes, which underscores the importance of capturing both steady growth and sudden shifts in price trends.

The neural network architecture developed in this study incorporated three input neurons corresponding to the opening, highest, and closing prices, with the closing price used as the target variable for prediction. The architecture was tested with varying numbers of neurons in the hidden layer, ranging from 1 to 10. The model achieved optimal performance with 10 neurons in the hidden layer, as indicated by the lowest Mean Absolute Percentage Error (MAPE) of 3.49%. This result suggests that while increasing the number of neurons initially improved performance, additional neurons beyond 10 did not yield significant improvements and led to increased computational complexity and a higher risk of overfitting. The choice of 10 neurons aligns with findings from previous studies, which suggest that increasing hidden layer complexity can improve predictive performance up to a point, after which additional neurons provide diminishing returns. The optimal configuration for this model balances complexity with predictive accuracy, ensuring the model remains computationally efficient and avoids overfitting.

The selection of 10 neurons in the hidden layer was based on multiple trials and error convergence analysis. The error convergence curve demonstrated that models with fewer than 10 neurons exhibited slower convergence and higher residual errors, suggesting that they lacked sufficient complexity to capture the non-linear relationships inherent in the stock price data. On the other hand, configurations with 10 neurons exhibited faster and more stable convergence, achieving minimal error values efficiently. This highlights the importance of choosing an appropriate architecture to balance

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

training efficiency and model accuracy, which is crucial for time-series forecasting tasks in financial markets. These results are consistent with previous research, which has shown that optimal hidden layer configurations are critical for ANN performance in financial prediction tasks. The increased computational cost with additional neurons beyond a certain threshold, coupled with the risk of overfitting, reinforces the need to carefully select the optimal number of neurons for each model.

The learning rate plays a pivotal role in controlling the speed of convergence and the stability of the training process. In this study, the learning rate was tested within a range of 0.1 to 0.9, with the optimal performance achieved at a learning rate of 0.3, yielding a MAPE of 3.49%. This result is in line with prior research, which suggests that moderate learning rates (typically between 0.2 and 0.4) strike an effective balance between convergence speed and numerical stability. Lower learning rates tend to slow down training, while higher learning rates risk overshooting the optimal solution, leading to erratic convergence and higher error rates. The choice of a 0.3 learning rate reflects the theoretical recommendation found in prior literature, and its application in this study provides empirical evidence supporting its use in ANN-based stock price prediction models. This choice ensures that the model converges at an appropriate pace while maintaining stability and minimizing prediction errors.

The training process of the ANN model was carried out with a maximum iteration limit of 1,000, with training stopping earlier at iteration 915 due to the model achieving optimal performance. The training and validation loss curves showed consistent decreases, indicating that the model successfully learned the underlying patterns of the data without overfitting. The achieved MAPE of 3.49% on the training data suggests that the model is effective in capturing stock price trends, with the steady reduction in loss providing confidence in the model's generalization capability. The early convergence observed during the training process is consistent with findings in other financial prediction studies using ANN. It demonstrates that, with the right parameters, ANN models can achieve efficient learning and avoid unnecessary computational overhead. The training behavior of the model further supports the notion that selecting the right configuration—especially in terms of neurons and learning rate—is crucial for successful stock price prediction.

The final stock price prediction results, using the optimized ANN model architecture (3-10-1), produced an error value of 2.26% MAPE on the testing dataset and 3.49% MAPE on the training dataset. The lower error in the testing dataset indicates that the model generalizes well to unseen data, avoiding overfitting and maintaining stable performance across both training and testing phases. This predictive accuracy places the model's performance in the "very good" range, in line with existing research that considers MAPE values under 10% as highly accurate for stock market prediction tasks. While the model produced accurate predictions on the ADMR stock prices, there are several limitations that should be addressed in future research. The model currently uses only three input features (opening, highest, and closing prices) and does not incorporate other potentially useful variables such as trading volume, macroeconomic indicators, or market sentiment, which could improve prediction accuracy. Additionally, the model uses a single hidden layer and random weight initialization, which may introduce variance in training outcomes. Future work could explore the integration of additional input features, the use of multiple hidden layers, or advanced optimization techniques to improve robustness and consistency.

The results of this study demonstrate the effectiveness of using an ANN-Backpropagation model for stock price prediction, particularly for ADMR stock data. The optimal configuration of 10 neurons in the hidden layer, a learning rate of 0.3, and the use of the closing price as the target variable yielded strong predictive performance, with the model achieving an error of 2.26% on testing data. These findings are consistent with previous research in the field and highlight the potential of ANN models for forecasting stock prices. However, the study also identifies areas for improvement, including the incorporation of additional input variables and the exploration of more advanced architectures. Future research should build upon these findings to enhance the model's robustness and adaptability across different market conditions.

5. CONCLUSION

Based on the analysis conducted in this study, the ANN model using the backpropagation algorithm demonstrated strong predictive performance for ADMR stock prices, achieving a MAPE of 2.26%. The optimal architecture consists of a 3–10–1 structure with three input neurons, ten neurons in the hidden layer, and one output neuron. Model training was carried out using a learning rate of 0.3 with binary sigmoid activation in the hidden layer and a linear activation function in the output layer, requiring 915 training iterations to reach convergence.

The findings of this study provide practical implications for investors and financial analysts. The resulting model can be used as an analytical tool to support short-term price forecasting, assist in identifying potential market trends, and inform decision-making processes when evaluating timing strategies for buying or selling ADMR stock. Moreover, the results demonstrate that ANN-based approaches can serve as a complementary alternative to traditional technical and fundamental analysis, particularly in fast-moving and non-linear market conditions. However, this study has several limitations. The model employs random weight initialization, which may increase prediction variance; future studies may consider optimized methods such as Nguyen-Widrow initialization to improve convergence. In addition, this study only utilizes a single hidden layer, while prior research suggests that deeper architectures may increase learning capacity for complex patterns. Future research should explore models with multiple hidden layers or hybrid approaches combining ANN with metaheuristic algorithms for hyperparameter optimization. Further comparisons with alternative forecasting models, such as Extreme Learning Machine (ELM), LSTM, or ensemble-based methods, are also recommended to validate performance across different architectures and datasets.

ACKNOWLEDGMENT

The authors thank Zainal Abidin for his guidance and feedback, and Totok Chamidy for valuable suggestions and technical assistance that supported the completion of this study.

REFERENCES

[1] N. T. Amalya, "Pengaruh return on asset, return on equity, net profit margin dan debt to equity ratio terhadap harga saham," *J. sekuritas*, vol. 1, no. 3, pp. 157–181, 2018.
[2] R. Lebaea, Y. Roshe, S. Ntontela, and B. A. Thango, "The role of data governance in ensuring system success and long-term IT performance: A systematic review," 2024.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

[3] M. Adil and C. F. Mwachikoka, "Non-Financial Firms Listed On The Pakistan Stock Exchange," *Available SSRN 5132818*, 2025.

[4] A. Anjeliana, "Penilaian Tingkat Kinerja Keuangan dengan Menggunakan Analisis Rasio Keuangan pada Perusahaan PT. Adaro Energy Tbk di Bursa Efek Indonesia Tahun 2018-2022," 2022, *Universitas Muhammadiyah Berau*.

[5] T. K. Gupta and K. Raza, "Optimization of ANN architecture: a review on nature-inspired techniques," *Mach. Learn. bio-signal Anal. diagnostic imaging*, pp. 159–182, 2019.

[6] V. I. Kontopoulou, A. D. Panagopoulos, I. Kakkos, and G. K. Matsopoulos, "A review of ARIMA vs. machine learning approaches for time series forecasting in data driven networks," *Futur. Internet*, vol. 15, no. 8, p. 255, 2023.

[7] A. Pagliaro, "Artificial intelligence vs. efficient markets: A critical reassessment of predictive models in the big data era," *Electronics*, vol. 14, no. 9, p. 1721, 2025.

[8] N. Panda, T. Singh, and S. Swagatika, "ANN based modeling for stock market prediction," *J. Stat. Manag. Syst.*, vol. 26, no. 1, pp. 87–100, 2023.

[9] K. E. Hoque and H. Aljamaan, "Impact of hyperparameter tuning on machine learning models in stock price forecasting," *IEEE Access*, vol. 9, pp. 163815–163830, 2021.

[10] H. R. Patel, A. M. Patel, H. A. Patel, and S. M. Parikh, "Hyperparameter tune for neural network to improve accuracy of stock market prediction," in *Proceedings of Data Analytics and Management: ICDAM 2021, Volume 1*, Springer, 2022, pp. 65–76.

[11] A. Chauhan, S. J. Shivaprakash, H. Sabireen, A. Q. Md, and N. Venkataraman, "Stock price forecasting using PSO hypertuned neural nets and ensembling," *Appl. Soft Comput.*, vol. 147, p. 110835, 2023.

[12] F. Konak, M. A. Bülbül, and D. Türkoǧlu, "Feature selection and hyperparameters optimization employing a hybrid model based on genetic algorithm and artificial neural network: Forecasting dividend payout ratio," *Comput. Econ.*, vol. 63, no. 4, pp. 1673–1693, 2024.

[13] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, "Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques," *Expert Syst. Appl.*, vol. 42, no. 1, pp. 259–268, 2015.

[14] M. Ballings, D. Van den Poel, N. Hespeels, and R. Gryp, "Evaluating multiple classifiers for stock price direction prediction," *Expert Syst. Appl.*, vol. 42, no. 20, pp. 7046–7056, 2015.

[15] A. M. Rather, A. Agarwal, and V. N. Sastry, "Recurrent neural network and a hybrid model for prediction of stock returns," *Expert Syst. Appl.*, vol. 42, no. 6, pp. 3234–3241, 2015.

[16] E. Chong, C. Han, and F. C. Park, "Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies," *Expert Syst. Appl.*, vol. 83, pp. 187–205, 2017.

[17] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, V. K. Menon, and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model," in *2017 international conference on advances in computing, communications and informatics (icacci)*, IEEE, 2017, pp. 1643–1647.

[18] T. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions," *Eur. J. Oper. Res.*, vol. 270, no. 2, pp. 654–669, 2018.

[19] N. Q. K. Le, D. T. Do, T. N. K. Hung, L. H. T. Lam, T.-T. Huynh, and N. T. K. Nguyen, "A computational framework based on ensemble deep neural networks for essential genes identification," *Int. J. Mol. Sci.*, vol. 21, no. 23, p. 9070, 2020.

[20] Y. Zhang, J. Li, H. Wang, and S.-C. T. Choi, "Sentiment-guided adversarial learning for stock price prediction," *Front. Appl. Math. Stat.*, vol. 7, p. 601105, 2021.

[21] A. Subiyakto, A. R. Ahlan, M. Kartiwi, and S. J. Putra, "Measurement of the information system project success of the higher education institutions in Indonesia: a pilot study," *Int. J. Bus. Inf. Syst.*, vol. 23, no. 2, pp. 229–247, 2016, [Online]. Available: https://doi.org/10.1504/IJBIS.2016.078908

[22] G. Urva *et al.*, *Penerapan Data Mining Di Berbagai Bidang: Konsep, Metode, dan Studi Kasus*. PT. Sonpedia Publishing Indonesia, 2023.

[23] S. Sivanandam and M. Paulraj, *Introduction to artificial neural networks*. Vikas Publishing House, 2009.

[24] M. Uzair and N. Jamil, "Effects of hidden layers on the efficiency of neural networks," in *2020 IEEE 23rd international multitopic conference (INMIC)*, IEEE, 2020, pp. 1–6.

[25] M. A. M. Ahmed and S. M. Zubair, "A generalized artificial neural network approach to model multiple-extraction humidification-dehumidification systems," *Int. Commun. Heat Mass Transf.*, vol. 159, p. 108188, 2024.

[26] M. Sahi and G. R. H. Galib, "Artificial Intelligence Application of Back-propagation Neural Network in Cryptocurrency Price Prediction," *IJEIE Int. J. Electr. Intell. Eng.*, vol. 1, no. 1, pp. 32–46, 2025.

[27] J. Jepkoech, D. M. Mugo, B. K. Kenduiywo, and E. C. Too, "The effect of adaptive learning rate on the accuracy of neural networks," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 8, 2021.

[28] S. Afaq and S. Rao, "Significance of epochs on training a neural network," *Int. J. Sci. Technol. Res*, vol. 9, no. 06, pp. 485–488, 2020.

[29] J. Liu, N. Chu, P. Wang, L. Zhou, and H. Chen, "A novel hybrid model for freight volume prediction based on the Baidu search index and emergency," *Neural Comput. Appl.*, vol. 36, no. 3, pp. 1313–1328, 2024.

[30] S. Bozkurt Keser and Y. Buruk Sahin, "Response surface methodology to tune artificial neural network hyper-parameters," *Expert Syst.*, vol. 38, no. 8, p. e12792, 2021.

[31] H. Jiang, S. Anumasa, G. De Masi, H. Xiong, and B. Gu, "A unified optimization framework of ann-snn conversion: towards optimal mapping from activation values to firing rates," in *International Conference on Machine Learning*, PMLR, 2023, pp. 14945–14974.

[32] A. Kurani, P. Doshi, A. Vakharia, and M. Shah, "A comprehensive comparative study of artificial neural network (ANN) and support vector machines (SVM) on stock forecasting," *Ann. Data Sci.*, vol. 10, no. 1, pp. 183–208, 2023.

[33] S. Kim and H. Kim, "A new metric of absolute percentage error for intermittent demand forecasts," *Int. J. Forecast.*, vol. 32, no. 3, pp. 669–679, 2016.

[34] M. Sahi, M. Faisal, Y. M. Arif, and C. Crysdian, "Analysis of the Use of Artificial Neural Network Models in Predicting Bitcoin Prices," *Appl. Inf. Syst. Manag.*, vol. 6, no. 2, pp. 91–96, 2023.