Jurnal Elektro
p-ISSN: 1979-9780
e-ISSN: 2746-4288

Jurnal Elektro Vol. 19 No. 1 April 2026
journal homepage: https://ejournal.atmajaya.ac.id/index.php/JTE

Enhancing Stock Price Prediction Using Temporal Convolutional Network with Moving Average Features

Rizki Surya Permana*, **Aisyah Novfitri, Putri Rahmawati**

[1]Bachelor of Telecommunications Engineering, School of Electrical Engineering, Telkom University, Jakarta, Indonesia

Article Info

Article history:

Received
13-04-2026

Accepted
30-05-2026

Keywords:
TCN, CNN, Prediction, Stock Price

Abstract

Stock price prediction is a complex problem in the financial domain due to the non-linear, dynamic nature of the data and its dependence on various external factors. This study proposes a deep learning–based approach using a Temporal Convolutional Network (TCN) to predict the stock price of Bank Central Asia Tbk. The model is evaluated under two scenarios: using a single feature (close price) and using three features MA5, and MA10. The dataset consists of historical BBCA stock data from 2010 to 2018, split into 80% training data and 20% testing data. The model is trained with 100 epochs, a window size of 60, batch size of 32, the Adam optimizer, and a learning rate of 0.0005. Experimental results show that the TCN model using a single feature achieves an RMSE of 227.1920, MAE of 192.8089, and MAPE of 4.3411%. Meanwhile, the TCN model with additional features (MA5 and MA10) demonstrates improved performance, achieving an RMSE of 176.4599, MAE of 145.7689, and MAPE of 3.2750%, indicating an accuracy improvement of more than 20%.These findings indicate that TCN is effective in capturing temporal patterns in financial time series data, and that incorporating simple technical indicators such as Moving Averages can significantly enhance model performance. This study contributes to the development of efficient and practical stock price prediction methods, particularly in the Indonesian stock market context.

Info Artikel

Histori Artikel:

Diterima:
13-04-2026

Disetujui:
30-05-2026

Kata Kunci:
TCN, CNN, Prediksi, Harga Saham

Abstrak

Prediksi harga saham merupakan permasalahan kompleks dalam bidang keuangan karena sifat data yang non-linear, dinamis, dan dipengaruhi oleh berbagai faktor eksternal. Penelitian ini mengusulkan pendekatan berbasis deep learning menggunakan Temporal Convolutional Network (TCN) untuk memprediksi harga saham BBCA. Model diuji dalam dua skenario, yaitu menggunakan satu fitur (close price) dan menggunakan tiga fitur (close price, Moving Average 5 hari (MA5), dan Moving Average 10 hari (MA10)). Data yang digunakan merupakan data historis saham BBCA periode 2010–2018 dengan pembagian 80% data training dan 20% data testing. Model dilatih menggunakan parameter epoch sebanyak 100, window size 60, batch size 32, optimizer Adam, serta learning rate sebesar 0.0005. Hasil eksperimen menunjukkan bahwa model TCN dengan satu fitur menghasilkan RMSE sebesar 227.1920, MAE sebesar 192.8089, dan MAPE sebesar 4.3411%. Sementara itu, model TCN dengan penambahan fitur MA5 dan MA10 menghasilkan performa yang lebih baik dengan RMSE sebesar 176.4599, MAE sebesar 145.7689, dan MAPE sebesar 3.2750%, yang menunjukkan peningkatan akurasi lebih dari 20%. Hasil ini menunjukkan bahwa TCN efektif dalam menangkap pola temporal pada data time series finansial, dan penambahan indikator teknikal sederhana seperti Moving Average mampu meningkatkan performa model secara signifikan. Penelitian ini memberikan kontribusi dalam pengembangan metode prediksi harga saham yang efisien dan aplikatif, khususnya pada pasar saham Indonesia.

*Corresponding author. Rizki Surya Permana
Email address: rizkisuryapermana@telkomuniversity.ac.id

1. INTRODUCTION

The rapid development of the capital market has made stock price prediction one of the critical challenges in the fields of finance and artificial intelligence. Stock prices exhibit dynamic and non-linear characteristics and are influenced by various complex factors, including global economic conditions, monetary policies, corporate performance, and investor sentiment. This complexity makes stock price movements inherently difficult to predict accurately, thereby increasing the risk in investment decision-making. Therefore, there is a need for approaches that are capable of effectively capturing stock price patterns and adapting to market changes.

One of the stocks frequently used as a research object in Indonesia is Bank Central Asia Tbk, which is categorized as a blue-chip stock with high liquidity. Despite its relatively stable fundamentals, BBCA stock still exhibits significant price fluctuations and complex movement patterns. This indicates that even stocks with strong fundamentals remain challenging to predict, thus requiring more advanced analytical approaches.

In practice, stock price analysis is often conducted using technical indicators, one of which is the Moving Average (MA). Moving Average is a smoothing technique used to identify trends by reducing noise in price movements. In this study, MA5 and MA10 are employed, representing short-term and medium-term trends, respectively. The combination of these indicators is commonly used in crossover strategies to generate buy and sell signals. However, Moving Average has inherent limitations as it is a lagging indicator, resulting in delayed responses to actual price changes. This limitation can reduce its effectiveness, particularly in fast-moving and volatile markets.

To address these limitations, various machine learning methods have been developed and applied for stock price prediction [1]. Techniques such as Decision Tree, Random Forest, Support Vector Machine (SVM), and Recurrent Neural Network (RNN) have been widely explored. Decision Tree offers high interpretability but tends to suffer from overfitting in highly variable data [2]. Random Forest mitigates overfitting through ensemble learning but is not specifically designed to capture temporal dependencies in sequential data [3]. Meanwhile, SVM demonstrates strong generalization capabilities but is highly sensitive to parameter selection and less effective in modeling long-term temporal dependencies [4][5].

On the other hand, RNN is designed to process sequential data and capture temporal relationships. However, it faces challenges related to training stability and computational efficiency, especially when dealing with long sequences [6]. These limitations highlight that existing methods are still not fully capable of handling the complexity of stock data, which involves non-linear patterns and long-term dependencies.

With the advancement of deep learning, Temporal Convolutional Network (TCN) has emerged as a promising approach for time series modeling. TCN is a Convolutional Neural Network (CNN)-based architecture specifically designed for sequential data by utilizing causal convolution and dilated convolution. Causal convolution ensures that the model only uses past and present information, preserving temporal order, while dilated convolution enables the model to capture long-term dependencies without significantly increasing computational complexity. Additionally, TCN offers advantages in training stability and parallelization, making it more efficient than recurrent-based models [7][8].

Although TCN has demonstrated promising performance in various time series prediction studies, most existing research focuses on comparisons with other deep learning models or applications in global markets. Studies examining the application of TCN in the Indonesian stock market, particularly for stocks with characteristics such as BBCA, remain limited. Furthermore, many previous studies rely on complex or numerous features, while the exploration of simple technical indicators such as Moving Average in combination with deep learning models is still relatively scarce. In fact, simple indicators like MA5 and MA10 are advantageous in representing market trends intuitively and are widely used in practical trading.

Based on these observations, there exists a research gap in developing a deep learning model that can efficiently capture long-term temporal dependencies while integrating simple and practical technical indicators. Therefore, this study proposes the use of a Temporal Convolutional Network (TCN) combined with Moving Average indicators (MA5 and MA10) for stock price prediction. This approach aims to bridge the gap between conventional methods that are interpretable but limited, and deep learning methods that are powerful yet often complex.

The main contribution of this study lies in the application of TCN within the context of the Indonesian stock market using simple technical indicators as primary features. This approach is expected not only to improve prediction accuracy but also to produce a model that is more aligned with practical technical analysis used by investors. Consequently, this research is anticipated to provide a significant contribution to the development of more effective, efficient, and applicable stock price prediction methods.

2. BASIC THEORY

TCN is a convolutional architecture specifically developed for time-series data processing. Compared to conventional CNN, TCN incorporates causal convolutions, ensuring that the output at a given time step depends only on present and past inputs, thereby preserving the temporal sequence of the data [9]. In addition, TCN employs dilated convolutions to effectively enlarge the receptive field without increasing the number of parameters. This enables the model to capture long-range dependencies within sequential data. The fundamental structure of the dilated convolution model is illustrated in Figure 1, along with the receptive field coverage of TCN under different dilation factors (d = 4, d = 2, and d = 1) [9].

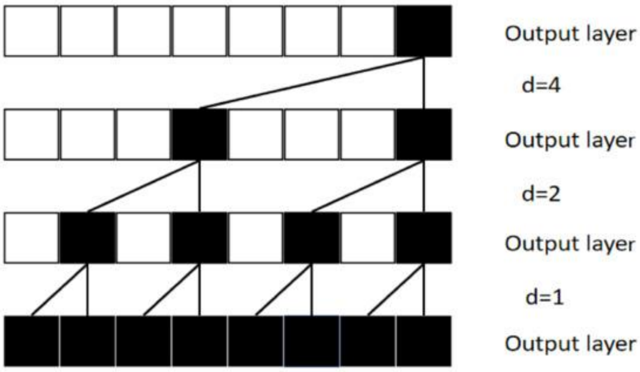

Figure 1. Basic Structure Of The Dilated Convolutions Model

In general, TCN operates by utilizing three main components: causal convolution, dilated convolution, and residual connections. These components enable TCN to preserve the temporal order of the data, expand the range of historical information (receptive field), and improve network stability during the training process. Causal convolution is a fundamental principle in TCN that ensures the output at a given time step is influenced only by inputs from the current and previous time steps, it shown in equation (1). Thus, the model does not use future information, making it well-suited for time-series data such as stock prices.

$$y(t) = \sum_{i=0}^{k-1} w(i)\, x(t-i) \tag{1}$$

Where $y(t)$ represents the output at time t, $x(t-i)$ denotes the historical input, and $w(i)$ corresponds to the convolutional kernel weights [10]. This mechanism is crucial for maintaining prediction validity, particularly in financial applications.

Furthermore, to enhance the model's ability to capture long-term dependencies, TCN employs dilated convolution shown in (2). In this approach, the convolution operation is performed with specific intervals between input elements, referred to as the dilation factor. By introducing dilation, the model can expand its receptive field without significantly increasing the number of layers.

$$y(t) = \sum_{i=0}^{k-1} w(i)\, x(t-d\cdot i) \tag{2}$$

Where $d$represents the dilation factor [9]. The larger the value of d, the wider the range of information that can be captured by the model. This allows TCN to understand long-term patterns in the data, such as stock price trends over extended periods.

To maintain training stability and prevent performance degradation in deep networks, TCN also adopts residual connections. In this mechanism, the output of a layer is not only derived from the convolutional transformation but also includes the original input of that layer. In simple terms, a residual connection can be expressed in (3).

$$y = F(x) + x \tag{3}$$

where F(x) represents the transformation output of the convolutional layer [11]. This approach has been proven to accelerate convergence and maintain gradient flow during the training process.

Overall, the combination of causal convolution, dilated convolution, and residual connections makes TCN an effective and efficient model for capturing temporal patterns [12]. Moreover, since it does not rely on recurrent mechanisms, TCN enables parallel computation, making it faster than recurrent-based models [13].

3. LITERATURE REVIEW

Stock price prediction has become an increasingly important topic in the fields of finance and artificial intelligence, particularly with the growing adoption of machine learning and deep learning techniques for time series analysis. In recent years, research has indicated a shift from traditional approaches toward deep learning-based methods, which are more capable of capturing non-linear patterns and temporal dependencies in financial data [14], [15].

Traditional machine learning methods such as Decision Tree, Random Forest, and SVM are still widely used as baseline approaches in stock prediction studies. Random Forest, as an ensemble method, has been shown to improve prediction stability compared to a single Decision Tree. For instance, Abraham et al. demonstrated that the combination of Genetic Algorithm and Random Forest can effectively predict stock trends with satisfactory accuracy [16]. Additionally, Saberironaghi et al. highlighted that Random Forest is also effective for feature selection and market trend prediction [17]. However, tree-based methods are inherently limited in capturing long-term temporal dependencies, as they are not specifically designed for sequential data.

Moreover, Support Vector Regression (SVR) and other machine learning approaches are still commonly applied in stock price prediction, particularly when combined with technical indicators. Mostafavi et al. reported that incorporating a large number of technical indicators can improve the performance of models such as SVR and Random Forest, but it also increases model complexity and the risk of overfitting [18]. Similarly, Ayyildiz demonstrated that machine learning methods remain relevant for predicting stock price movements, although their performance is highly dependent on data characteristics [19].

With the increasing demand for modeling time series data, deep learning models have gained significant attention in stock price prediction. Recurrent Neural Networks (RNNs) and their variants have become prominent due to their ability to model temporal relationships. Kundu et al. showed that deep learning models such as RNN and CNN outperform traditional methods in capturing non-linear patterns in stock data [20]. Furthermore, Gholami et al. concluded that deep learning approaches generally provide superior performance compared to conventional models in stock prediction tasks [21].

However, recurrent-based models suffer from limitations in computational efficiency and training stability, especially when handling long sequences. To address these issues, convolution-based approaches have been introduced for sequential data modeling, one of which is the Temporal Convolutional Network (TCN). TCN utilizes causal convolution and dilated convolution mechanisms to preserve temporal order while efficiently expanding the receptive field.

In recent years, TCN has been increasingly applied in financial time series prediction. Chen et al. proposed an improved TCN model for stock index prediction and demonstrated its effectiveness in extracting temporal patterns [22]. Yao et al. introduced a MEMD-TCN model that integrates signal decomposition with TCN to enhance prediction accuracy for non-linear data [23]. Additionally, Liu et al. developed an attention-based TCN model (ATCGAN), showing that TCN is capable of capturing temporal dependencies effectively, although it remains sensitive to noise [24].

Further developments indicate that convolution-based models continue to evolve into more complex architectures. Wang et al. proposed a localized convolution model for stock index prediction by considering both spatial and temporal relationships [25]. Meanwhile, Xie et al. introduced a hybrid model combining convolutional and transformer architectures to improve the modeling of long-term dependencies [26].

In the context of the Indonesian stock market, research on stock price prediction remains relatively limited. Indra et al. showed that although patterns exist in Indonesian stock data, predictive signals are often weak due to high market noise [27]. Additionally, Bellaly et al. demonstrated that an experimental setup using an 80% training and 20% testing split, combined with a sliding window approach, is effective for stock prediction studies [28].

Based on this literature review, it can be concluded that despite the extensive use of various methods, there remains a research gap in applying efficient deep learning models such as TCN to the Indonesian stock market using simple yet representative features. Most previous studies rely on complex features or hybrid models that are difficult to interpret. Therefore, this study focuses on the use of TCN with simple features, namely close price, MA5, and MA10, to develop a model that is not only accurate but also interpretable and aligned with practical technical analysis used by investors.

4. METHODOLOGY

This study proposes a deep learning–based approach using the TCN architecture to predict the stock price of BBCA. The dataset consists of historical BBCA stock data from 2010 to 2018. The research methodology is systematically designed, including stages of data collection, preprocessing, model design, training, and performance evaluation. The research flowchart is illustrated in Fig 2.

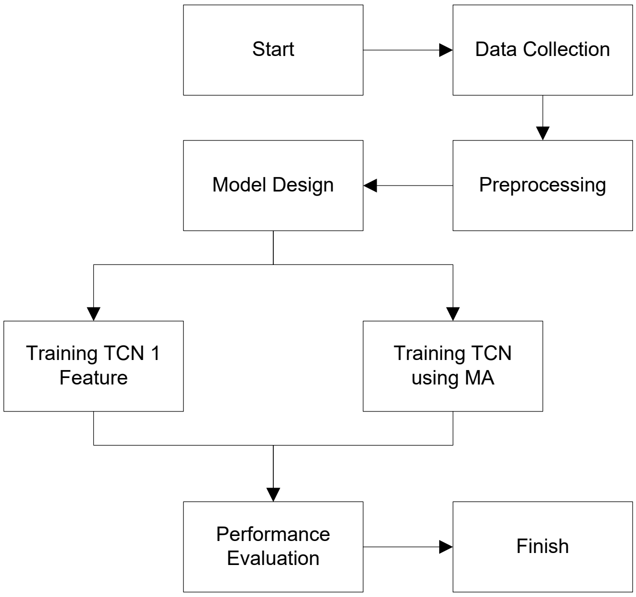

Figure 2. Research Flow Diagram

This study is designed with two experimental scenarios to evaluate the impact of feature selection on model performance. The first scenario uses a single feature, namely the closing price (close price), while the second scenario utilizes three features: close price, MA5, and MA10.

4.1. Data Collection

The data used in this study consists of historical stock price data of BBCA obtained from reliable capital market data sources. The data period spans from 2010 to 2018, with a daily frequency. The main focus of this study is on the closing price as the target variable to be predicted.

4.2. Preprocessing

The preprocessing stage is conducted to ensure data quality and readiness before being used in the model training process. This stage includes data cleaning, time-based sorting, feature construction, and data normalization. In the first scenario, the input data consists of a single feature, namely the closing price. In the second scenario, two additional technical features, MA5 and MA10, are incorporated, both calculated from the closing price data. Moving Average is used to represent price trends, where MA5 reflects short-term trends and MA10 represents medium-term trends.

After feature construction, all data are normalized using the Min-Max Scaling method to avoid differences in feature scales that could affect the training process. Subsequently, the data are transformed into sequences using a sliding window approach with a window size of 60.

The processed data are then divided into two parts: 80% for training and 20% for testing. The split is performed sequentially based on time (time-based split) to preserve the integrity of the time series data and prevent information leakage from future data.

The training data are used to build and train the model, while the testing data are used to evaluate the model's performance on unseen data. To capture temporal patterns in the time series data, a sliding window approach with a window size of 60 is applied. Thus, the model takes 60 previous time steps as input to predict the closing price at the next time step. In the first scenario, the input sequence dimension is (60, 1), whereas in the second scenario, the input dimension is (60, 3). This approach enables the model to learn temporal relationships either from the closing price alone or from the combination of price and technical indicators.

4.3. Model Design And Training

Table 1. Architecture Model TCN

Layer (type)	Output Shape	Param #
TCN	(none, 128)	444288
Dense	(none, 64)	8256
Dropout	(none, 64)	0
Dense	(none, 32)	2080
Dense	(none, 1)	33

The model used in this study is the Temporal Convolutional Network (TCN), which is a convolution-based architecture for sequential data modeling. TCN is selected due to its ability to capture long-term dependencies and its computational efficiency compared to recurrent-based models. The architecture of the model is illustrated in Table 1. The model is trained using the following parameters Epochs 100, Batch size 32, Window size 60, Learning rate 0.0005, and Optimizer is Adam.

This study is designed based on two main experimental scenarios. In the first scenario, the TCN model is trained using a single feature, namely the closing price. In the second scenario, the model incorporates multiple features, including the closing price, Moving Average 5 (MA5), and Moving Average 10 (MA10). The comparison between these two scenarios aims to evaluate the impact of adding technical indicator features on the overall performance of the model.

4.4. Performance Evaluation

The model performance is evaluated using three commonly used metrics in time series prediction, namely Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). RMSE assigns a larger penalty to large errors, making it more sensitive to outliers (4). MAE measures the average absolute error between the actual and predicted values (5). MAPE expresses the error in percentage form, making it easier to interpret the model's accuracy (6).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{4}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |\, y_i - \hat{y}_i \,| \tag{5}$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \tag{6}$$

5. RESULT AND DISCUSSION

Based on the experimental results, TCN model was evaluated under two scenarios: using a single feature (close price) and using three features (Close, MA05, and MA10). The evaluation was conducted using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE).

The evaluation results indicate that the TCN model with a single feature achieved an RMSE of 227.1920, MAE of 192.8089, and MAPE of 4.3411%. In contrast, the TCN model with additional features (MA5 and MA10) achieved improved performance, with an RMSE of 176.4599, MAE of 145.7689, and MAPE of 3.2750%. This comparison demonstrates that incorporating Moving Average features significantly enhances model performance, with error reductions of more than 20% across all evaluation metrics.

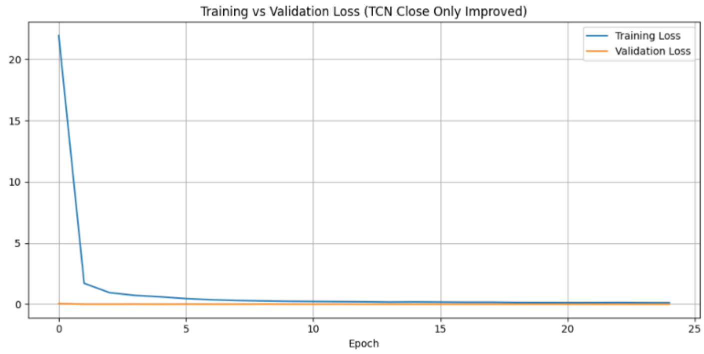

Figure 3. Comparation Training and Validation Loss using 1 Feature

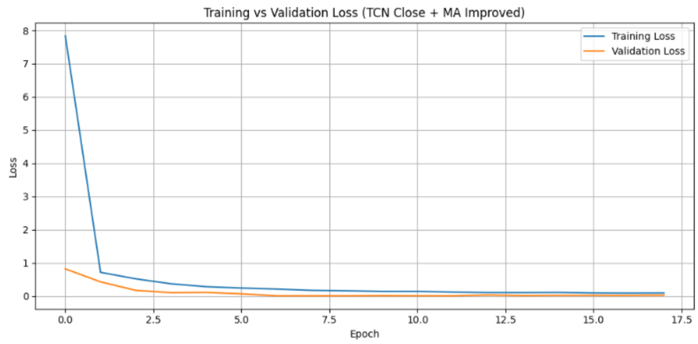

Figure 4. Comparation Training and Validation Loss using 3 Feature

From the training perspective, both models exhibit good convergence behavior. However, the TCN model with MA5 and MA10 features shows faster and more stable reductions in both training loss and validation loss compared to the single-feature model in Fig 3 and Fig 4. Moreover, the absence of a significant gap between training and validation loss in both models indicates that overfitting does not occur, suggesting good generalization capability. The use of adaptive learning rate mechanisms also contributes to maintaining training stability.

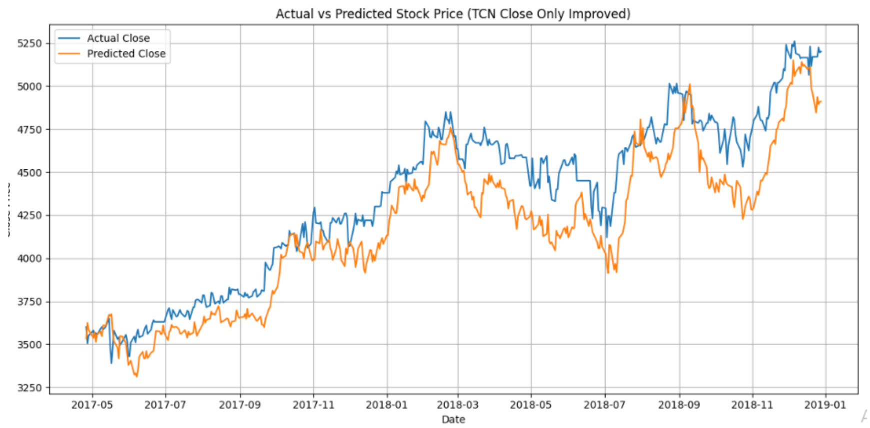

Figure 5. Comparison Actual and Predicted Price using 1 Feature

Further analysis of the prediction results reveals that the single-feature TCN model is capable of capturing the general trend of stock price movements, both in upward and downward trends shown in Fig 5. However, the model tends to produce smoother predictions compared to the actual data, making it less effective in capturing extreme fluctuations. This is evident in peak and valley regions, where predicted values tend to be lower than actual values at peaks and higher than actual values at troughs.

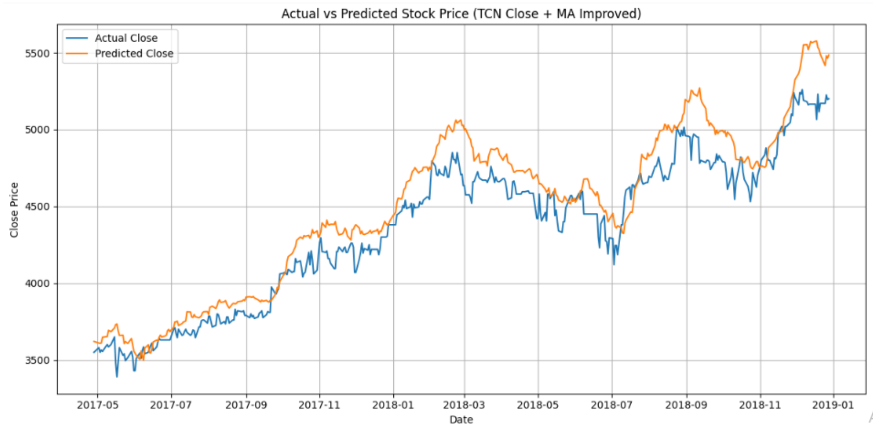

Figure 6. Comparison of the Actual and Predicted Price using 3 Features

In contrast, in Figure 6 the TCN model with additional MA5 and MA10 features demonstrates a better ability to follow stock price movements. The predictions are closer to the actual data and more responsive to trend changes. This indicates that the additional information provided by Moving Average indicators helps the model better understand the direction of price movements. Nevertheless, there is a tendency for the model to slightly overestimate values during certain periods.

The performance improvement observed in the multi-feature TCN model can be explained by several factors. First, Moving Average indicators provide additional information about short-term and medium-term trends, assisting the model in learning price movement patterns. Second, Moving Average acts as a smoothing filter that reduces noise in the data, facilitating feature extraction by the TCN model. Third, the use of multi-feature input enables the model to capture inter-variable relationships, resulting in a richer data representation compared to using a single feature.

From an architectural perspective, both models use the same configuration with a relatively similar number of parameters. Therefore, the performance improvement is not attributed to increased model complexity but rather to the quality of input features. This highlights the critical role of feature selection in stock price prediction tasks.

Overall, the results demonstrate that the TCN model can provide accurate predictions for stock price data, even when using a single feature. However, the inclusion of simple technical indicators such as MA5 and MA10 significantly improves prediction accuracy. This finding suggests that integrating deep learning models with feature engineering based on technical analysis is an effective approach for stock price prediction. Moreover, it indicates that using simple relevant features can yield optimal results without significantly increasing model complexity.

5.1. Comparative Analysis with Baseline Models

To further validate the effectiveness of the proposed Temporal Convolutional Network (TCN) model, comparative experiments were conducted using a baseline deep learning model, namely Long Short-Term Memory (LSTM). Both models were evaluated using the same dataset, feature configuration, training–testing split, and evaluation metrics to ensure a fair comparison. The comparison focused on two configurations: TCN using only the close price and TCN using multiple features consisting of close price, MA5, and MA10.

Table 2. The Comparative Results LSTM and TCN

No	Model	Features	RMSE	MAE	MAPE (%)
1	LSTM + MA	Close + MA5 + MA10	607.0657	599.3536	13.3680
2	TCN + MA	Close + MA5 + MA10	176.4599	145.7689	3.2750
3	TCN	Close	227.1920	192.8089	4.3411

Based on the comparative analysis in Table 2, it can be concluded that the Temporal Convolutional Network (TCN) model outperformed the LSTM model in predicting BBCA stock prices. The TCN model with Moving Average features (MA5 and MA10) achieved the best performance, with an RMSE of 176.4599, MAE of 145.7689, and MAPE of 3.2750%. These results indicate a significant improvement compared to both the LSTM model using the same features and the TCN model using only the close price.

This finding suggests that the TCN architecture is more effective in capturing temporal dependencies in financial time series data than recurrent-based models such as LSTM. The use of dilated causal convolution enables TCN to learn both short-term and long-term patterns more stably and efficiently. In addition, the integration of simple technical indicators such as MA5 and MA10 enriches the representation of stock price trends and helps reduce noise in the data.

6. CONCLUSION

This study proposes the use of a Temporal Convolutional Network (TCN) for stock price prediction under two scenarios: using a single feature (close price) and using three features (close price, MA5, and MA10). Based on the experimental results, it can be concluded that the TCN model is capable of delivering accurate predictive performance in modeling financial time series data. In the first scenario, the TCN model using only the closing price achieves an RMSE of 227.1920, MAE of 192.8089, and MAPE of 4.3411%, indicating that the model falls within a highly accurate prediction category. However, the model still exhibits limitations in capturing extreme price fluctuations, as reflected by the smoothing effect in its predictions. In the second scenario, with the addition of Moving Average features (MA5 and MA10), the model performance improves significantly, achieving an RMSE of 176.4599, MAE of 145.7689, and MAPE of 3.2750%. The consistent reduction in error across all evaluation metrics demonstrates that incorporating simple technical indicators enhances the model's ability to capture stock price trend patterns. These findings indicate that although TCN is effective in capturing temporal dependencies, the quality and relevance of input features remain crucial factors in improving predictive performance. The integration of deep learning models with simple technical indicators proves to be an effective approach for stock price prediction.

Furthermore, comparative analysis with the LSTM baseline model demonstrates the superiority of the proposed TCN approach. Although both models utilized the same input features (Close, MA5, and MA10), the TCN + MA model significantly outperformed the LSTM + MA model across all evaluation metrics. The LSTM model produced an RMSE of 607.0657, MAE of 599.3536, and MAPE of 13.3680%, while the proposed TCN + MA model achieved substantially lower error values. This result indicates that TCN is more effective in capturing temporal dependencies and trend information in financial time series data. The use of dilated causal convolution enables TCN to learn both short-term and long-term patterns more efficiently while maintaining stable training behavior. In addition, the incorporation of MA5 and MA10 successfully reduced prediction noise and enriched trend representation, allowing the model to generate predictions that are closer to the actual stock prices. Overall, the results confirm that the proposed TCN-based framework combined with simple Moving Average indicators provides a robust and effective solution for stock price forecasting, particularly in the context of the Indonesian stock market.

Despite the promising results, several opportunities for future research remain. Future studies may explore the inclusion of additional features such as other technical indicators (e.g., RSI, MACD, Bollinger Bands) or external factors such as market sentiment and macroeconomic data, in order to further improve the model's ability to capture more complex market dynamics. In addition, future work may incorporate walk-forward validation, rolling-window evaluation, and comparisons with other advanced architectures such as GRU, Transformer, and hybrid attention-based models to further evaluate the robustness and generalization capability of the proposed approach.

REFERENCE

[1] X. Li, H. Yang, and J. Gao "Deep learning approaches for stock market prediction: A comparative study," *Neurocomputing*, 2022.

[2] R. Patel and M. Shah, "Stock market prediction using decision tree algorithm," *Journal of Finance and Data Science*, 2021.

[3] S. Kumar *et al.,* "Random forest-based stock price prediction using technical indicators," *IEEE Access*, 2022.

[4] J. Wang *et al.,* "Support vector machine for stock market prediction: A systematic review," *Applied Sciences*, 2020.

[5] Y. Liu *et al.,* "Temporal convolutional networks for stock prediction," *IEEE Access*, 2021.

[6] A. Sherstinsky, "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network," Physica D: Nonlinear Phenomena, vol. 404, p. 132306, Mar. 2020. doi: 10.1016/j.physd.2019.132306

[7] S. Bai, J. Z. Kolter, dan V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," arXiv preprint, arXiv:1803.01271, Mar. 2018. doi: 10.48550/arXiv.1803.01271.

[8] S. Bai, J. Z. Kolter, and V. Koltun, “An empirical evaluation of convolutional and recurrent networks for sequence modeling,” arXiv preprint arXiv:1803.01271, relevance extended 2020–2023.

[9] Y. Wu, Z. Zhao, X. Li, F. Zhang, Z. Chen, dan G. Zhao, "Time series prediction using sparse autoencoder and high-order fuzzy cognitive maps," IEEE Transactions on Fuzzy Systems, vol. 28, no. 12, pp. 3110-3122, Dec. 2020, doi: 10.1109/TFUZZ.2020.2964344.

[10] K. He, X. Zhang, S. Ren, dan J. Sun, "Deep residual learning for image recognition," dalam Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, Jun. 2016, p. 770–778. doi: 10.1109/CVPR.2016.90

[11] C. X. Zhang, J. Li, X. F. Huang, J. S. Zhang, and H. C. Huang, "Forecasting stock volatility and value-at-risk based on temporal convolutional networks," Expert Systems with Applications, vol. 207, p. 117951, 2022. doi: 10.1016/j.eswa.2022.117951

[12] H. Zhang *et al.,* “Financial time series forecasting using TCN,” Expert Systems with Applications, 2023.

[13] 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, p. 102616, Jan. 2025. doi: 10.1016/j.inffus.2024.102616

[14] S. Giantsidi and C. Tarantola, "Deep learning for financial forecasting: A review of recent trends," International Review of Economics & Finance, vol. 104, 2025. doi: 10.1016/j.iref.2025.104719

[15] R. Abraham, M. El Samad, A. M. Bakhach, H. El-Chaarani, A. Sardouk, S. El Nemar, and D. Jaber, "Forecasting a stock trend using genetic algorithm and random forest," Journal of Risk and Financial Management, vol. 15, no. 5, p. 188, Apr. 2022. doi: 10.3390/jrfm15050188

[16] M. Saberironaghi dan A. Saberironaghi, "Stock market prediction using machine learning and deep learning techniques: A review," AppliedMath, vol. 5, no. 3, p. 76, 2025. doi: 10.3390/appliedmath5030076

[17] S. M. Mostafavi dan A. R. Hooman, "Key technical indicators for stock market prediction," Machine Learning with Applications, vol. 20, p. 100631, Jun. 2025. doi: 10.1016/j.mlwa.2025.100631

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

[19] T. Kundu et al., "Predicting daily stock price directions with deep learning models," Machine Learning with Applications, vol. 19, p. 100617, 2025. doi: 10.1016/j.mlwa.2024.100617

[20] M. T. Gholami dan A. Shahabi, "Comparison of deep learning methods with traditional financial time series models for forecasting the TEPIX index in the Tehran Stock Exchange," Systems and Soft Computing, vol. -, p. 200395, 2025. doi: 10.1016/j.sasc.2025.200395

[21] W. Chen, N. An, M. Jiang, dan L. Jia, "An improved deep temporal convolutional network for new energy stock index prediction," Information Sciences, vol. 682, p. 121244, Jul. 2024. doi: 10.1016/j.ins.2024.121244

[22] Y. Yao, Z. Y. Zhang, dan Y. Zhao, "Stock index forecasting based on multivariate empirical mode decomposition and temporal convolutional networks," Applied Soft Computing, vol. 142, p. 110356, 2023. doi: 10.1016/j.asoc.2023.110356

[23] Y. Liu, X. Huang, L. Xiong, R. Chang, W. Wang, and L. Chen, "Stock price prediction with attentive temporal convolution-based generative adversarial network," Array, vol. 25, p. 100374, Mar. 2025. doi: 10.1016/j.array.2025.100374

[24] C. Wang, J. Ren, H. Liang, J. Gong, and B. Wang, "Conducting stock market index prediction via the localized spatial–temporal convolutional network," Computers & Electrical Engineering, vol. 108, p. 108687, 2023. doi: 10.1016/j.compeleceng.2023.108687

[25] L. Xie, Z. Chen, dan S. Yu, "Deep Convolutional Transformer Network for Stock Movement Prediction," Electronics, vol. 13, no. 21, p. 4225, Okt. 2024. doi: 10.3390/electronics13214225

[26] D. I. Pirdaus, "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, p. 714, Des. 2025. doi: 10.3390/jrfm18120714

[27] A. Bellaly, S. Belattar, and E. K. Haimoudi, "Application of Artificial Intelligence in Stock Market Prediction," Eng. Proc., vol. 112, no. 1, p. 34, Oct. 2025, doi: 10.3390/engproc2025112034.