
Comparative Study of Time Series Forecasting on Iron Sales Using CNN, MLP, and LSTM

Nabila Putri Listyanto¹, Wiyli Yustanti²

^{1,2} *State University of Surabaya, Surabaya, Indonesia*

nabila.18059@unesa.ac.id, wiyliyustanti@unesa.ac.id

ABSTRACT

Sales forecasting is essential for businesses to predict future demand and inform strategic and operational planning, especially in the building materials retail industry. Accurate sales prediction supports inventory management, cost control, and supply chain efficiency. This study compares the performance of 3 deep learning models, Convolutional Neural Network (CNN), Multilayer Perceptron (MLP), and Long Short-Term Memory (LSTM), in forecasting daily iron sales at PT Surya Aneka Bangunan from 2016 to 2020. The models were trained on 80% of the historical data and tested on 20%. Model performance was evaluated using Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and Coefficient of Determination R^2 . The results show that the CNN model achieved the best performance with an MAE of 0.293, RMSE of 0.357, MAPE of 0.081, and R^2 of 0.9989, indicating high accuracy and stability. The MLP model produced higher errors, while the LSTM model had the lowest MAPE but greater error variability. These findings suggest that the CNN model is the most reliable for capturing temporal patterns in iron sales data. The study contributes to the development of adaptive sales forecasting systems and opens opportunities for applying similar methods in other retail sectors to support data driven decision making.

Keyword: Forecasting, Time-Series, Deep-Learning, CNN, MLP, LSTM

Article Info:

Article history:

Received July 15, 2025

Revised August 02, 2025

Accepted September 01, 2025

Corresponding Author

Nabila Putri Listyanto

State University of Surabaya, Surabaya, Indonesia

nabila.18059@mhs.unesa.ac.id

1. INTRODUCTION

The construction industry in Indonesia has grown a lot in recent years. This growth is linked to the improving economy, which supports the development of infrastructure, housing, and commercial buildings. As construction grows, the demand for building materials also rises, giving building supply stores a chance to grow and meet customer needs. Sales forecasting is a way to predict future demand by looking at past sales data. In business, it helps companies make decisions about production [1]. One common method used for forecasting is the Time Series method [2]. In this study, the authors used Time Series forecasting on sales data from 2016 to 2020. The five years of data show that iron is the top-selling building material at PT Surya Aneka Bangunan and brings in the most revenue.

Time series methods have advanced with the use of deep learning, which uses more complex layers to improve accuracy and efficiency. Deep learning can provide better forecasting results than traditional methods because it can capture both linear and non-linear patterns in the data [3]. Deep learning methods for time series forecasting include various types of neural

network architectures, such as Convolutional Neural Network (CNN), Multilayer Perceptron (MLP), and Long Short-Term Memory (LSTM). Each of these approaches offers distinct strengths and limitations, which can be aligned with the specific characteristics of the dataset and the goals of the forecasting task.

The architecture of Convolutional Neural Network (CNN) includes convolution, pooling, and fully connected layers, enabling the model to construct more complex representations of the data in a hierarchical manner, converting raw time series data into high-level features that are more appropriate for forecasting tasks [4]. The key benefit of CNNs in forecasting time series is their capability to autonomously identify significant features without human intervention [5]. Nonetheless, CNNs possess constraints as well. The primary limitation of conventional CNNs is their challenge in capturing long-term dependencies, particularly in varying data conditions [6]. In research on the energy industry, the 1D-CNN model demonstrated its advantages. This study contrasts 1D-CNN with ANN-MLP and RNN-LSTM for predicting electricity demand. The model reached an average MAPE of 4.62% for short-term predictions and 1.45% for medium-term predictions, surpassing MLP and LSTM in this particular task [7].

Multilayer Perceptron (MLP) represents a core type of Artificial Neural Network (ANN) and commonly serves as the initial model in time series predictions [8]. The primary benefit of MLP is its structural straightforwardness and minimal complexity. Additionally, MLPs offer adaptable and uniform outcomes, rendering them a dependable and commonly chosen option in supervised learning tasks [9]. Despite its advantages, MLP has notable limitations for time series forecasting, as it finds it challenging to manage global or long-term dependencies in the data, resulting in the loss of crucial contextual information [10]. A study in the energy management sector demonstrated its effectiveness. This research focuses on forecasting the energy use of water-cooled chillers by assessing MLP and LSTM models. In particular, MLP reached an R^2 of 0.971, a MAE of 0.743 kW, and an RMSE of 1.157 [11].

Long Short-Term Memory (LSTM) is a form of Recurrent Neural Network (RNN) that has been specifically created to address the shortcomings of conventional RNNs. Its primary function in time series prediction is attributed to its distinct structure, featuring memory cells and gating mechanisms which are input, forget, and output [12]. The primary benefit of LSTM lies in its exceptional capacity to acquire and retain long-term dependencies, a task that conventional RNNs or other models frequently struggle to achieve. Nevertheless, LSTMs also have their own limitations. LSTM models are often quite resource-intensive and demand substantial memory, particularly when handling extensive datasets, leading to increased runtimes [13]. Research within the financial industry validates these benefits. The research evaluated the effectiveness of LSTM, MLP, and CNN in predicting stock market trends. Utilizing data from the Malaysian stock market in both univariate and multivariate formats and assessed through MAE, MSE, and RMSE metrics, the LSTM model consistently demonstrated superior performance. LSTM succeeded in attaining the least prediction error compared to the other two models. [14].

The main objective of this study is to predict the demand for iron products at PT Surya Aneka Bangunan's building store, using historical sales data from 2016-2020. To achieve this goal, we have selected 3 different models CNN, MLP, and LSTM as methods for forecasting. The decision to compare these three methods is an important step in this research because it will help business determine the most suitable and effective forecasting method.

2. METHODS

This study adopts the Knowledge Discovery in Database (KDD) methodology, which systematically identifies patterns within datasets to facilitate easier understanding and interpretation. The KDD process consists of five key stages: data selection, pre-processing,

transformation, data mining, and evaluation. An illustration of the research flow can be seen in Figure 1.

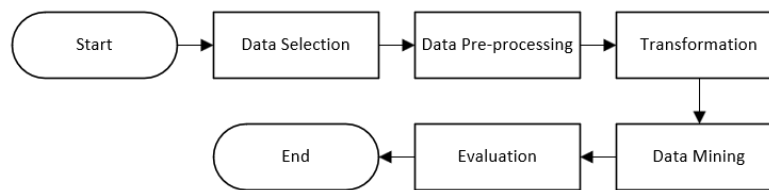


Figure 1. Research Flow Diagram

2.1 Data Selection

The dataset used in this study is daily building material sales data from PT Surya Aneka Bangunan, a retail store located in Surabaya, Indonesia. The dataset spans five years, comprising 60 months of historical records from January 2016 to December 2020. It consists of 45.151 samples and includes 14 features (columns), such as Date, No. Reff, No. invoices, Costumer Name, Group, item code, item name, unit, Qty, price Include, price DPP, Brutto, PPN, and Netto. However, this study only uses one feature as input, which is quantity (Qty) representing the total number of iron units sold. Since the focus of this research is on forecasting iron, the most frequently sold product, the dataset is filtered specifically to include only sales of iron. The forecasting model developed in this study is a univariate time series model, meaning it utilizes only one feature, namely Qty, to predict future values. In addition, the units used in this study are adjusted only using kilogram (kg) units to avoid inconsistencies in the number of sales that can occur if there are different units such as tons, bars, and so on. The data types are shown in the following Table 1.

Table 1. Dataset Structure Description

No.	Column	Data Type	Description
1.	Date	Date	The date the sales transaction occurred.
2.	No. Reff	String	Transaction reference number that links to related transactions.
3.	No. Faktur	String	Invoice number as a unique identifier of the sales transaction.
4.	Customer Name	String	Name of the customer making the transaction.
5.	Group	String	Item category based on material type.
6.	Item Code	String	Unique code to identify the item sold.
7.	Item Name	String	Name of the item recorded in the transaction.
8.	Unit	String	Type of unit of measurement used in the transaction.
9.	Qty	Int	Number of items sold in a specific unit.
10.	Price <i>Include</i>	Int	Unit price of the item including tax.
11.	Price DPP	Int	Taxable base price per unit of the item.
12.	Bruto	Int	Total gross price of the transaction before tax.
13.	PPN	Int	Value Added Tax amount from the transaction.
14.	Netto	Int	Total net price to be paid by the customer.

2.2 Data Pre-Processing

This stage is crucial before applying dataset to deep learning models [15]. This stage involves 3 main processes, which are handling missing values, normalization and data splitting. Here is Figure 2 illustrates the time series plot of annual total iron sales during the entire observation period, which runs from 2016 to 2020. The plot shows dynamic fluctuations in sales

quantities, with several prominent peaks and drops across five year span. These variations reflect the seasonal and irregular patterns that must be captured by forecasting models to produce accurate predictions.

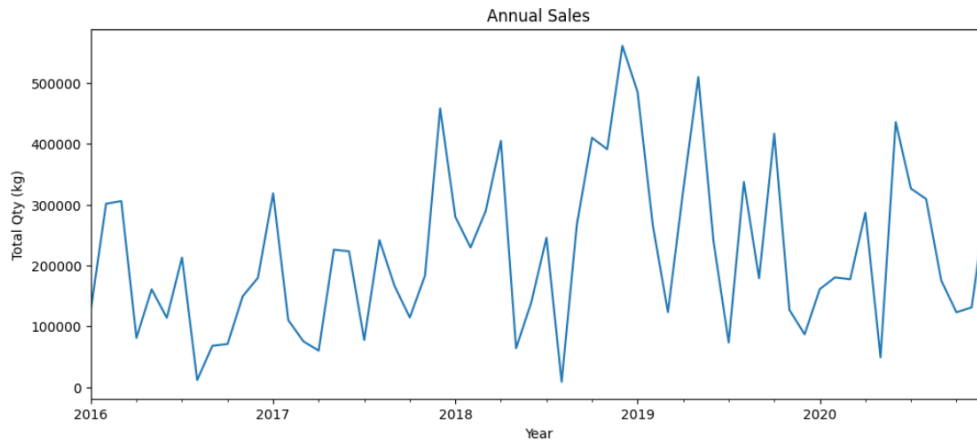


Figure 2. Time Series Plot

Normalization is performed to adjust the data, allowing the model to learn patterns more effectively. This procedure is particularly crucial in models that rely on deep learning. The Min-Max Scaling method is used to normalize the data to the range (0, 1). The goal is to eliminate bias caused by scale variations among variables, allowing the model to analyze and compare the data efficiently [16]. In this study, the min-max normalization formula is used in equation (1), where x_{norm} refers to the scaled data in the range (0, 1), x is the input sample, x_{min} is the minimum value, and x_{max} is the maximum value of the feature.

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

After normalization, data splitting refers to the act of dividing data into smaller groups, typically for training and evaluating models. In time series analysis, data splitting is very important to test model performance objectively and avoid overfitting. Dataset is divided into 2 parts, namely 80% training data and 20% test data [17]. Data training covers the period from January 2016 to December 2019, while data testing covers January 2020 to December 2020 as shown in Table 2.

Table 2. Splitting Data	
Data Training	Data Testing
80%	20%
January 2016 - December 2019	January 2020 - December 2020

2.3 Transformation

There are 2 steps of transformation for this study, such as feature engineering and reshape input the x_{train} and x_{test} shapes. In this study, the feature engineering process is performed by adding temporal information in the form of lag features and moving averages. Lag features are used to represent sales values in the previous period, while moving averages are calculated to capture sales trends in the short term. These two features provide historical context that helps the model recognize recurring patterns in time series data [18]. Lastly, the input data

for both training and testing sets are reshaped into a three-dimensional structure. This transformation is essential to meet the input requirements of LSTM and CNN, which expect sequential data with explicit temporal dimensions [19].

2.4 Data Mining

Convolutional Neural Network (CNN) is a sophisticated deep learning architecture frequently used for handling structured data like images and time series. As shown in Figure 3, CNNs are made up of convolutional layers that autonomously recognize and extract significant features from the input. These layers utilize filters to analyze the input sequence, allowing the model to identify temporal patterns and spatial relationships. The resulting feature maps are subsequently fed into fully connected layers to carry out either classification or regression tasks. CNNs have demonstrated their effectiveness in time series forecasting tasks, such as estimating wind speed [10], solar irradiance, and trends in the stock market [20]. In one-dimensional convolutions, the kernel acts as a moving window that identifies important patterns, whereas max pooling is utilized to decrease input dimensions by choosing the highest value within a specified window. Generally, Rectified Linear Unit (ReLU) serves as the activation function for feature extraction, while a sigmoid function is utilized in the output layer for regression tasks.

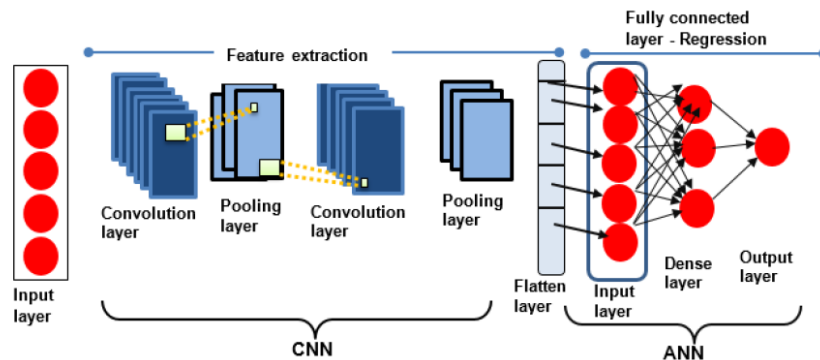


Figure 3. Architecture of CNN

Multilayer Perceptron (MLP), depicted in Figure 4, is a type of deep learning model based on a feedforward neural network structure. The model is built with an input layer, followed by one or more hidden layers, and concludes with an output layer [21]. Each of these layers is made up of multiple interconnected processing units, often referred to as neurons. To capture complex patterns and relationships within the data, MLP utilizes non-linear activation functions. During the training phase, it applies the backpropagation algorithm, which fine-tunes the network's weights and biases by propagating the error backward from the output to earlier layers. Due to its flexibility and effectiveness, MLP is widely implemented in a variety of applications, including data classification, function approximation, and pattern detection.

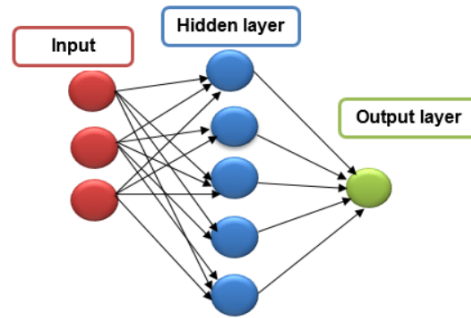


Figure 4. Architecture of MLP

Long Short-Term Memory (LSTM) is a specific type of Recurrent Neural Network (RNN) that shows exceptional performance in handling sequential data [22]. By utilizing memory cells in combination with gating mechanisms, the model effectively mitigates the vanishing gradient problem, ensuring that relevant information can be preserved across many time steps and allowing for more reliable learning in deep sequential data. LSTM networks are capable of retaining information over extended sequences, which makes them especially useful for tasks such as time series prediction. These models feature an internal memory system that chooses which information to keep or ignore, enabling them to create long-range dependencies. Within the model, input, forget, and output gates coordinate the flow of data by selecting which inputs to keep, which past information to erase, and which outputs to generate. The tanh function ensures that the information passing through remains within a stable, non-linear range. Below is the formula presented in equation 2.

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2)$$

In equation (2), the components of the system include the forget gate (f_t), the memory state (c_t), the input gate (i_t), the output gate (o_t), the hidden state (h) and the memory state (c), as shown in Figure 5.

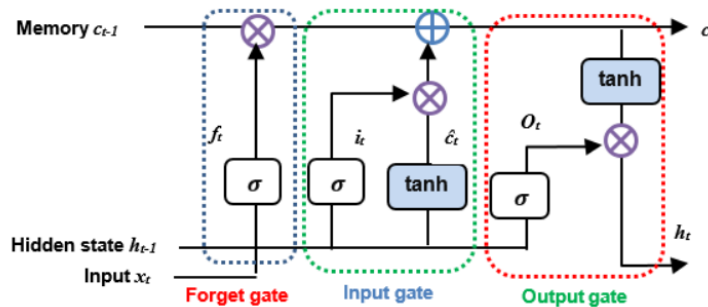


Figure 5. Architecture of LSTM

The next stage after transformation is building models of CNN, MLP, and LSTM. Once each model is constructed, the process continues with the training phase using several supporting parameters including Epoch, Batch Size, Optimizer, Loss, Learning Rate, and Validation Split as shown in Table 3. Validation Split of 0.2 is used to display the validation accuracy and loss during each epoch iteration in the training phase.

Table 3. Parameter of Models

Parameter	CNN	MLP	LSTM
-----------	-----	-----	------

Epoch	100	100	200
Batch Size	32	32	64
Optimizer	Adam	RMSprop	Adam
Loss	MSE	MSE	MSE
Learning Rate	0.001	0.001	0.0001
Validation Split	0.2	0.2	0.2

Table 3 outlines the specific parameter settings employed during the model training process for CNN, MLP, and LSTM. Each architecture was constructed and trained using distinct configurations tailored to improve forecasting outcomes. CNN was built with a single convolutional layer consisting of 64 filters and a kernel size of 2, followed by a flatten operation, a dense layer with 50 neurons using ReLU activation, and an output layer. This model was trained using the Adam optimization method, applying a learning rate of 0.001 and utilizing Mean Squared Error (MSE) to measure loss. Training spanned 100 epochs with a batch size of 32 and used 20 percent of the data for validation.

MLP model included two hidden dense layers with 100 and 50 units, both activated by the ReLU function. RMSprop was selected as the optimizer, with a learning rate set at 0.001 and MSE used as the loss criterion. This model was trained over 100 epochs using a batch size of 16, with 20% of the data reserved for validation purposes.

LSTM model, the structure comprised two sequential layers with 64 and 32 units, where the initial layer was designed to return the full sequence of inputs. A dense output layer followed this setup. The training process used the Adam optimizer with a smaller learning rate of 0.0001 and MSE as the loss function. Training was carried out for 200 epochs using a batch size of 64 and a validation split of 0.2.

2.5 Evaluation

In this study employs four widely used evaluation metrics, Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and the Coefficient of Determination (R^2). These metrics together offer a thorough evaluation of model precision. MAE and RMSE measure the typical size of errors, with RMSE being more responsive to significant deviations. MAPE represents error in percentage terms, enabling a relative interpretation across various scales, whereas R^2 assesses the model's effectiveness in explaining the variance of the actual data. Recent research has shown the significance of integrating these metrics, utilizing MAE, RMSE, and R^2 to assess deep learning model, providing dependable insights into predictive performance [14]. In a different study, MAE, MAPE, and RMSE were utilized to evaluate an air quality prediction model, emphasizing their significance in identifying both absolute and relative errors in successive forecasts [15]. In equations (3), (4), (5), and (6), the value of x_i represents the predicted output generated by each of the three models, $\bar{x}_i$ denotes the actual observed values from the dataset, and n is the total number of data points.

$$MAE = \frac{1}{n} \sum_{i=1}^n |\bar{x}_i - x_i| \quad (3)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x}_i)^2} \quad (4)$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{\bar{x}_i - x_i}{x_i} \right| \times 100 \quad (5)$$

$$R^2 = 1 - \frac{\sum_{i=1}^n |x_i - \bar{x}_i|^2}{\sum_{i=1}^n |x_i - \bar{x}|^2} \quad (6)$$

3. RESULTS AND DISCUSSION

In this study elaborates on the prediction results and performance evaluation of 3 deep learning models, CNN, MLP, and LSTM, in forecasting iron sales data. The predictive capability of each model is first illustrated through visualization plots that compare actual sales values with the corresponding predicted values, providing a preliminary insight into the accuracy and alignment of each model's output. Furthermore, a comprehensive performance assessment is conducted using four standard evaluation metrics, including MAE, RMSE, MAPE, and R^2 . These metrics are used to quantitatively measure the extent of prediction error and the ability of each model to generalize underlying patterns within the time series data. The detailed performance scores for each model are presented in Table 4.

Table 4. Performance Evaluation

Predictive deep -learning models	Performance Evaluation			
	MAE	RMSE	MAPE	R^2
CNN	0,293	0,357	0,081	0,9989
MLP	0,372	0,464	0,109	0,9975
LSTM	0,369	0,595	0,066	0,9962

Based on the performance evaluation results in Table 4, all three deep learning models, CNN, MLP, and LSTM exhibited strong predictive capabilities, as reflected by their high R^2 values and relatively low error rates. The CNN model achieved the best overall performance, with an MAE of 0.293 and RMSE of 0.357, along with a MAPE of 0.081 and an R^2 of 0.9989. These results indicate that CNN effectively captured temporal patterns in the sales data with minimal prediction error. The MLP model followed closely, recording an MAE of 0.372 and RMSE of 0.464. While its MAPE was slightly higher at 0.109, the model still demonstrated a high degree of accuracy with an R^2 of 0.9975. This suggests that MLP was also effective in modeling the underlying relationships in the data, although it incurred slightly greater deviations in absolute and relative terms compared to CNN. Interestingly, the LSTM model, despite achieving a slightly lower MAE of 0.369, produced the highest RMSE value at 0.595. However, it recorded the lowest MAPE among the three models, at 0.066, implying that it performed relatively better in terms of proportional accuracy. The R^2 value for LSTM was 0.9962, which still indicates a very strong fit to the actual data. This reflects LSTM's strength in handling sequence-based temporal dependencies, even though its overall error magnitude was slightly larger.

Considering the balance between absolute and percentage-based accuracy metrics, the CNN model is recommended as the most suitable for steel sales forecasting in this study. Its combination of low MAE, RMSE, and MAPE, along with a near-perfect R^2 , confirms its ability to deliver accurate and reliable predictions. This finding underscores CNN's potential in time

series forecasting applications, particularly in domains with relatively stable but nonlinear demand patterns.

The prediction performance of the three deep learning models is illustrated through a series of visual comparisons between actual sales values and predicted outputs. Figures 6, Figure 7, and Figure 8 respectively depict the forecasting results of the CNN, MLP, and LSTM models. Each plot presents the actual steel sales data over time alongside the corresponding predicted values generated by each model. These visualizations provide a qualitative understanding of how well each model captures the temporal dynamics and fluctuations in the sales data.

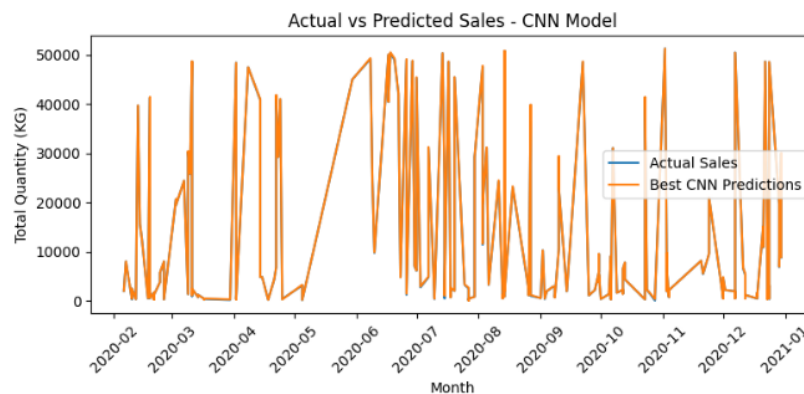


Figure 6. Plot Prediction CNN Model

Figure 6 presents the prediction results of the CNN model for total iron sales (in kilograms) based on the 2020 test data. The graph illustrates a comparison between the actual sales (depicted by the blue line) and the predictions generated by the best-performing CNN model (depicted by the orange line). The X-axis represents time in monthly format, while the Y-axis indicates total sales in kilograms. It can be observed that the predicted line closely follows the trend of the actual data throughout most of the period. This indicates that CNN model was able to effectively represent sales patterns on previously unseen data, providing predictions that closely approximate the actual values.

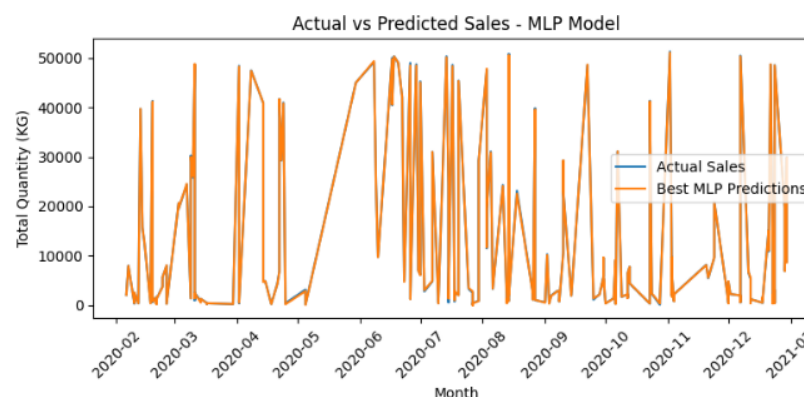


Figure 7. Plot Prediction MLP Model

Figure 7 displays the prediction results of the MLP model for total iron sales (in kilograms) based on the 2020 test data. The graph shows a comparison between the actual sales (blue line) and the predictions from the best-performing MLP model (orange line). The X-axis represents time in monthly format, while the Y-axis shows total sales in kilograms. It is evident that the MLP model's predicted line also follows the trend of the actual data, especially during periods with noticeable value changes. However, compared to CNN model, MLP predictions

exhibit sharper fluctuations at certain points. Nevertheless, the model is still capable of capturing the general sales pattern throughout the year.

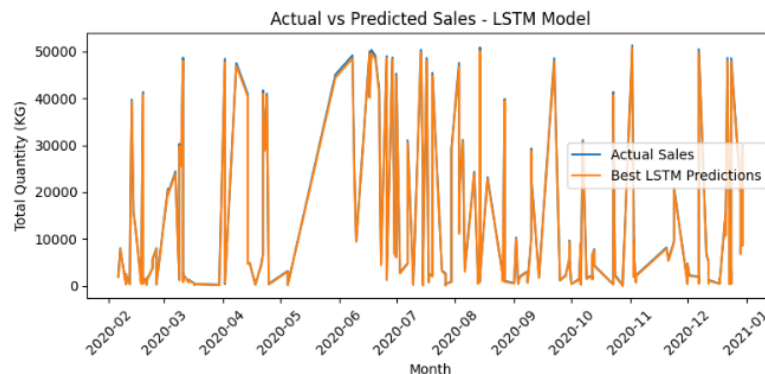


Figure 8. Plot Prediction LSTM Model

Figure 8 illustrates the prediction results of the LSTM model for total iron sales (in kilograms) based on the 2020 test data. The graph presents a comparison between the actual sales (blue line) and the predictions produced by the best-performing LSTM model (orange line). The X-axis represents time in monthly format, while the Y-axis indicates total sales in kilograms. The LSTM model is also shown to effectively track the trend of the actual data across various periods. Compared to the CNN and MLP models, the LSTM predictions appear more visually varied; however, they still reflect the overall sales trend reasonably well.

Based on the prediction visualization results for the 2020 test data, the CNN model showed the smoothest and most consistent prediction pattern, following the actual data line throughout the test period. The MLP model also followed the main sales trend well, although the resulting prediction line appeared more volatile at some points. Meanwhile, the LSTM model demonstrated the ability to recognize the direction of the data trend, but produced predictions that tended to be more varied than the CNN. Overall, the CNN model provided predictions that visually align most closely with the actual values compared to the other two models.

CONCLUSION

The performance evaluation of 3 deep learning models, CNN, MLP, and LSTM, on the iron sales data of PT Surya Aneka Bangunan shows that each model has different levels of predictive accuracy based on the evaluation metrics. CNN achieved the best overall results, with the lowest MAE of 0.293, the lowest RMSE of 0.357, a MAPE of 0.081, and the highest R^2 value of 0.9989. These results indicate that CNN is highly accurate, stable, and reliable in predicting sales quantities. Multilayer Perceptron (MLP) model produced a higher MAE of 0.372, RMSE of 0.464, and MAPE of 0.109, with an R^2 of 0.9975. Although the model still shows strong performance, its error rates are higher than those of CNN, which indicates a lower level of predictive accuracy. LSTM model recorded a MAE of 0.369 and achieved the lowest MAPE of 0.066, showing good accuracy in percentage terms. However, it also produced the highest RMSE of 0.595, indicating greater variability in prediction errors. In conclusion, while all three models are effective in capturing the temporal patterns of iron sales at PT Surya Aneka Bangunan, the CNN model emerged as the most reliable and consistent forecasting method. This finding supports the objective of the study in identifying the most accurate predictive model for retail iron sales. Moreover, the implementation of deep learning techniques as demonstrated in this study can serve as a valuable strategy for improving inventory planning and enhancing decision-making in the retail building material sector.

As a suggestion for future work, the forecasting can be further improved by incorporating external factors such as seasonality, promotions, and market trends to enhance

prediction accuracy. In addition, hybrid or ensemble modeling approaches may be explored to combine the strengths of different architectures. Integrating these models into a real-time decision support system could also help the company respond more proactively to changes in demand patterns.

REFERENCES

- [1] Douaioui, K., Oucheikh, R., Benmoussa, O., Mabrouki, C. (2024). Machine Learning and Deep Learning Models for Demand Forecasting in Supply Chain Management: A Critical Review. *Appl. Syst. Innov.*, 7, 93. <https://doi.org/10.3390/asi7050093>
- [2] Talagala, T. S., Hyndman, R. J., & Athanasopoulos, G. (2023). Meta-learning how to forecast time series. *Journal of Forecasting*, 42(6), 1476–1501. <https://doi.org/10.1002/for.2963>
- [3] Bousqaoui, H., Slimani, I., & Achchab, S. (2021b). Comparative analysis of short-term demand predicting models using ARIMA and deep learning. *International Journal of Power Electronics and Drive Systems/International Journal of Electrical and Computer Engineering*, 11(4), 3319. <https://doi.org/10.11591/ijece.v11i4.pp3319-3328>
- [4] Casolaro, A., Capone, V., Iannuzzo, G., & Camastra, F. (2023). Deep learning for time series Forecasting: advances and open problems. *Information*, 14(11), 598. <https://doi.org/10.3390/info14110598>
- [5] Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1). <https://doi.org/10.1186/s40537-021-00444-8>
- [6] Nguyen, T. A., & Tran, T. N. (2025). Improving Short-Term Electrical Load Forecasting with Dilated Convolutional Neural Networks: A Comparative Analysis. *Journal of Robotics and Control (JRC)*, 6(2), 560–569. <https://doi.org/10.18196/jrc.v6i2.24967>
- [7] Harish, B., Panda, D., Konda, K. R., & Soni, A. (2023). A Comparative Study of Forecasting Problems on Electrical Load Timeseries Data using Deep Learning Techniques. 2023 International Conference on Electrical, Computer and Energy Technologies (ICECET), 1–5. <https://doi.org/10.1109/icps57144.2023.10142125>
- [8] Dhenuvakonda, P., Anandan, R., & Kumar, N. (2020). Stock Price Prediction Using Artificial Neural Networks. *Journal of Critical Reviews*, 7(11), 846–850. <https://www.jcreview.com/?mno=118275>
- [9] Lazcano, A., Jaramillo-Morán, M. A., & Sandubete, J. E. (2024). Back to basics: the power of the multilayer perceptron in financial time series forecasting. *Mathematics*, 12(12), 1920. <https://doi.org/10.3390/math12121920>
- [10] Qiao, M., Liang, Y., Tavares, A., & Shi, X. (2023). Multilayer Perceptron network optimization for chaotic time series modeling. *Entropy*, 25(7), 973. <https://doi.org/10.3390/e25070973>
- [11] Nisa, E. C., & Kuan, Y. (2021). Comparative assessment to predict and forecast Water-Cooled Chiller power consumption using machine learning and deep learning algorithms. *Sustainability*, 13(2), 744. <https://doi.org/10.3390/su13020744>

- [12] BiLgiLi, M., Arslan, N., ŞekertekiN, A., & Yaşar, A. (2021). Application of long short-term memory (LSTM) neural network based on deep learning for electricity energy consumption forecasting. *Turkish Journal Of Electrical Engineering & Computer Sciences*, 30(1), 140–157. <https://doi.org/10.3906/elk-2011-14>
- [13] Kong, Y., Wang, Z., Nie, Y., Zhou, T., Zohren, S., Liang, Y., Sun, P., & Wen, Q. (2025b). Unlocking the power of LSTM for long term time series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(11), 11968–11976. <https://doi.org/10.1609/aaai.v39i11.33303>
- [14] AbKhalil, M., & Bakar, A. A. (2023). A comparative study of deep learning algorithms in univariate and multivariate forecasting of the Malaysian stock market. *Sains Malaysiana*, 52(3), 993–1009. <https://doi.org/10.17576/jsm-2023-5203-22>
- [15] Sharifnia, A. M., Kpormegbey, D. E., Thapa, D. K., & Cleary, M. (2025). A primer of data cleaning in quantitative research: handling missing values and outliers. *Journal of Advanced Nursing*. <https://doi.org/10.1111/jan.16908>
- [16] Pranolo, A., Setyaputri, F. U., Paramarta, A. K. I., Triono, A. P. P., Fadhilla, A. F., Akbari, A. K. G., Utama, A. B. P., Wibawa, A. P., & Uriu, W. (2024). Enhanced multivariate time series analysis using LSTM: A comparative study of Min-Max and Z-Score normalization techniques. *ILKOM Jurnal Ilmiah*, 16(2), 210–220. <https://doi.org/10.33096/ilkom.v16i2.2333.210-220>
- [17] Davis, R. A., & Fernandes, L. (2025). Sample splitting and assessing goodness-of-fit of time series. *Biometrika*, 112(2). <https://doi.org/10.1093/biomet/asaf017>
- [18] Cerqueira, V., Moniz, N., & Soares, C. (2021). VEST: automatic feature engineering for forecasting. *Machine Learning*, 113(7), 4523–4545. <https://doi.org/10.1007/s10994-021-05959-y>
- [19] Wibawa, A. P., Utama, A. B. P., Elmunsyah, H., Pujianto, U., Dwiyanto, F. A., & Hernandez, L. (2022). Time-series analysis with smoothed Convolutional Neural Network. *Journal of Big Data*, 9(1). <https://doi.org/10.1186/s40537-022-00599-y>
- [20] Sarbu, I. (2021). Solar heating and cooling systems. In *Springer eBooks* (pp. 329–445). https://doi.org/10.1007/978-3-030-64781-0_5
- [21] Smeeth, L., & Haines, A. (2023). COP 28: Ambitious climate action is needed to protect health (Vol. 383): British Medical Journal Publishing Group.
- [22] Sivakumar, M., S, J. P., George, S. T., Subathra, M., Leebanon, R., & Kumar, N. M. (2023). Nine novel ensemble models for solar radiation forecasting in Indian cities based on VMD and DWT integration with the machine and deep learning algorithms. *Computers & Electrical Engineering*, 108, 108691. <https://doi.org/10.1016/j.compeleceng.2023.108691>