UJM 13 (1) 2024
UNNES Journal of Mathematics
http://journal.unnes.ac.id/sju/index.php/ujm

Time Series Modeling of Stock Price Using CNN-BiLSTM with Attention Mechanism

Riska Nur Fitrianingsih*, Iqbal Kharisudin

Department of Mathematics, Faculty of Mathematics and Natural Science,
Universitas Negeri Semarang, Semarang, 50229, Indonesia

Article Info

Article History:
Received …….
Accepted …..
Published …..

Keywords:
time series forecasting; CNN; BiLSTM; attention mechanism; stock prices

Abstract

One of the sectors that has garnered attention is the telecommunications industry, which is rapidly growing alongside the increasing number of internet users and the public's demand for more advanced telecommunications services. PT Indosat Ooredoo Hutchison, as one of the leading telecommunications companies in Indonesia, has become an attractive investment choice for investors. However, the stock market is known for its fluctuating and irregular nature. Therefore, it is important for investors to understand stock price movements before making investments in order to reduce the risk of significant losses. One method that can be used to address that risk is by forecasting stock prices. Time series forecasting is a prediction about future values based on historical data. One of the techniques that is becoming increasingly popular in forecasting is deep learning. In this study, a combination of Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory (BiLSTM) with an attention mechanism is used. CNN excels at extracting data features, while BiLSTM is better at handling data with long time ranges. The addition of the attention mechanism allows the model to assign different weights to data features, enabling it to focus on the most relevant information. The combination of these three elements (CNN-BiLSTM with an attention mechanism) has the potential to yield higher prediction accuracy. Based on the results of the test evaluation for forecasting accuracy, the CNN-BiLSTM model with Attention Mechanism is proven to be the most superior model compared to other models. This can be seen from the lowest loss value and smaller MAE, MAPE, and RMSE evaluation matrices while the highest R^2 value.

How to cite:

Fitrianingsih, Riska Nur & Kharisudin, Iqbal. (2024). Time Series Modeling of Stock Price Using CNN-BiLSTM with Attention Mechanism. *UNNES Journal of Mathematics*. 13 (1): 11-20

*Correspondence Address
E-mail: riskanurfitrianingsih7@students.unnes.ac.id

p-ISSN 2252-6943
e-ISSN 2460-5859

1. Introduction

Time series is a series of data that is ordered by time. In time series analysis, data is sorted by consistent intervals such as daily, weekly, or monthly. In this case, statistics plays a very important role, especially in time series forecasting (Prasad & Venkatesham, 2021). Time series forecasting is a prediction or forecast of future values based on historical data. Statistical methods are used in time series forecasting which allow it to identify patterns and trends in historical data, as well as model the relationship between different variables over time. Deep learning techniques are widely used for time series forecasting, one of which is in the stock market. With the existence of deep learning can help investors in predicting stock price indexes through data information analysis (Hu *et al*., 2021).

Siami-Namini *et al*. (2019) conducted a study to compare the performance of LSTM and BiLSTM models in predicting stock prices. The test results found that the BiLSTM model was better in time series forecasting. Other studies, Lu *et al.* (2021) conducted a study to predict the stock index of Shanghai Composite Index using the CNN-BiLSTM-AM model. The main finding of the study is that when compared with 8 other time series forecasting models, the CNN-BiLSTM-AM hybrid model outperforms all of them with the lowest MAE and RMSE values and the R^2 value closest to one.

This model combines CNN-BiLSTM with an attention mechanism.The model combines a Convolutional Neural Network (CNN) that excels at extracting data features (Prasetyo *et al*., 2023). However, CNN has limitations in handling data over a long period of time (Nie *et al*., 2021). This problem can be solved by integrating the BiLSTM model, so that the combination of the two models can produce better prediction accuracy. Bidirectional Long Short-Term Memory (BiLSTM) consists of forward LSTM and backward LSTM which can learn patterns in data better, this makes BiLSTM superior in handling data with a long time span. However, both CNN and BiLSTM models still have difficulty reflecting the inconsistency of the level of importance of time series features in the time dimension. To overcome this, the addition of an attention mechanism to the model can provide different levels of weight to data features so that the model can focus on the most effective information (Sun *et al.,* 2022).

This research aims to analyze the application of various models in forecasting the stock prices of PT Indosat Ooredoo Hutchison, including CNN, LSTM, BiLSTM, CNN-BiLSTM, as well as variations of these models with Attention Mechanism, such as CNN with Attention Mechanism, LSTM with Attention Mechanism, BiLSTM with Attention Mechanism, and CNN-BiLSTM with Attention Mechanism. This research also aims to compare the accuracy results of stock price forecasting using each of these models to determine the most effective one.

The combination of CNN, BiLSTM, and attention mechanisms has the potential to produce more accurate predictions. In addition, this study can provide insights to analysts in developing forecasting models with deep learning so that it can support the development of science and technology. This study can be applied directly to other company stocks to help investors, portfolio managers, and policy makers in making strategic decisions. Thus, the development of the CNN-BiLSTM model with attention mechanisms not only has the potential to increase prediction accuracy, but also has the potential to pave the way for further innovation in financial analysis and risk management, which can ultimately improve the efficiency and stability of the financial market as a whole.

2. Method

2.1. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) is a type of deep neural network designed to process two-dimensional data with high network depth.(Son, 2016). Filters (kernels) are used to perform convolution on image data (two dimensions), so that CNN can recognize visual patterns such as edges, corners, and object shapes. Layers in CNN have a 3D neuron arrangement (width, height, depth). Width and height are the size of the layer while depth is the number of layers. CNN consists of several layers, namely the convolution layer, pooling layer, and fully connected (FC) layer (Hou *et al*., 2023). The layers of CNN are as follows.

1. The convolution layer is the main process underlying the CNN architecture network. This layer performs a convolution operation on the output of the previous layer. Convolution is a mathematical term for a function on the output of another function repeatedly. The convolution process is carried out to extract features from the input data. The convolution layer detects the layer using the filter method. Each different filter instance will detect and produce a different 3 x 3 matrix (Arifin *et al*., 2021).
2. The pooling layer is the process of reducing the size of data (Ghosh *et al*., 2019). This layer generally uses a filter that operates on each slice of its input in turn. The output of this process is a matrix with a smaller dimension compared to its input matrix, thus reducing the number of parameters and calculations in the network, to control overfitting. There are two types of pooling that are commonly used, namely mean pooling and max-pooling. Mean pooling takes the value from the average, while max-pooling takes the value from the maximum value.
3. Fully connected layeris a layer where all activation neurons from the previous layer are connected to the neurons of the next layer. The purpose of this layer is to perform data dimension transformation so that the data can be classified linearly. Each neuron from the convolution layer needs to be transformed into one-dimensional data before being fed into the fully connected layer. A 1 x 1 convolution layer can perform the same function as a fully connected layer but still maintains the spatial character of the data. This results in the use of fully connected layers in CNNs not being widely used (Asrafil *et al*., 2020).

2.2. ***Long Short Term Memory*** **(LSTM)**

Long Short Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) that can handle long sequential data better while considering time series and non-linear properties of the data (Zhang *et al*., 2023). LSTM is designed to solve the vanishing and exploding gradient problems of RNN. The LSTM architecture consists of memory blocks called cells. There are two states transferred to each cell layer, namely the cell state and the hidden state (Le *et al*., 2019). Cell state is the main component of LSTM that stores long-term information over time. Hidden state is the output of LSTM at a certain time and serves as input for the next. Each cell consists of an input gate, a forget gate and an output gate (Wu *et al*., 2024).

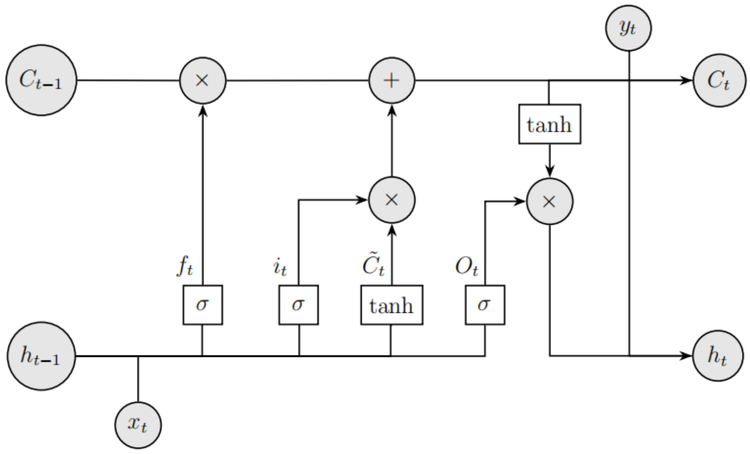

Figure 1 LSTM cell architecture

Figure 1. is the architecture of Cell LSTM with C_{t-1} is the cell state from the previous process, h_{t-1} is the hidden state from the previous LSTM, x_t is the current input, σ is the activation function, f is the output of the current forget gate, i_t is the output of the current input gate, $\widetilde{C_t}$ is is the current candidate cell state, o_t is the output of the current output gate, C_t is the current cell state, h_t is the hidden state of the current LSTM (Lu *et al*., 2021).

1. Forget gate: determines what information will be kept or deleted from the cell. The forget gate receives the previous LSTM hidden state and the new input is then processed using the sigmoid function. The output of the forget gate is calculated using the equation (2.1). This equation produces values from 0 to 1 (Le *et al*., 2019). The closer the result obtained is to 1, the more information will be stored, conversely, if the result obtained is closer to 0, the information will be deleted.
$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2.1}$$

2. Input gate: determine and store information from new input into cells to update information. Information from new input is combined and processed using the sigmoid function and the tanh function (Le *et al*., 2019). The sigmoid function decides whether new information should be updated or ignored. The calculation of the sigmoid function uses the equation (2.2). If the results obtained are closer to 1, it means that the information is important information and must be updated and vice versa. The tanh function gives a weight that will determine the level of importance. The calculation of the tanh function uses the equation (2.3).
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2.2}$$
$$\widetilde{C_t} = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{2.3}$$
The two results of the function are multiplied and then added together with the multiplication of the forget gate result with the previous process cell state to update the cell state (Lu *et al*., 2021). The calculation of this process is shown in the equation (2.4).
$$C_t = f_t * C_{t-1} + i_t * \widetilde{C_t} \tag{2.4}$$

3. Output gate: determines the output based on the updated cell from the input gate process. The information is processed with the sigmoid function to decide which information is output. The calculation of the process is shown in the equation (2.5). The result of the sigmoid function is multiplied by the tanh function from the input gate process to obtain the information that will be stored in the new hidden state (Le *et al*., 2019). The hidden state and the new cell state will then be forwarded to the next cell. Calculation shown in the equation (2.6)
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{2.5}$$
$$h_t = o_t * \tanh(C_t) \tag{2.6}$$

2.3. Bidirectional Long Short Term Memory (BILSTM)

Bidirectional Long Short Term Memory (BiLSTM) is a development of the LSTM model that moves in two directions, namely forward and backward. There are two LSTM networks in the BiLSTM model. The first LSTM network functions to process the sequence of input data in the forward direction (forward layer) and the second LSTM network functions to process the sequence of data in the reverse direction (backward layer). The output of BiLSTM is a combination of the forward and backward LSTM networks. With the presence of two layers in opposite directions, the BiLSTM model can learn past information and future information for each input sequence (Puteri, 2023). Each hidden layer output unit in the lower and upper layers is combined to form a longer feature value than the usual LSTM. This results in the information that will be processed in the next process will classify in more detail. The advantage of BiLSTM is that it can detect and extract more time dependencies than LSTM networks and solve them more precisely (Lindemann *et a*l., 2021).

2.4. Attention Mechanism

The attention mechanism (Attention Mechanism) originates from the human visual attention process (Zhang *et al.*, 2023). Human vision can quickly find the main area and add focus of attention to the main area to obtain the necessary detailed information. Similarly, the attention mechanism selectively gives attention to some more important information, ignores unimportant information, and allocates the importance of the information (Lu *et al.*, 2021). The attention mechanism acts based on weight allocation, determining the most effective information by distributing higher weights.

In the research (Brauwers & Frasincar, 2023) It is explained that the attention mechanism process is divided into 4 sub-model components, namely feature model, query model, attention model, and output model. Feature model is a basic model used to extract input data into a feature vector n_f. The collection of feature vectors n_f is called a matrix $F = [f_1, \dots, f_{n_f}] \in \mathbb{R}^{d_f \times n_f}$ with d_f is the size of the feature vector and is the number of feature vectors. The F matrix is extracted into 2 different matrices, namely the Key matrix $K = [k_1, \dots, k_{n_f}] \in \mathbb{R}^{d_k \times n_f}$ and Value matrix $V = [v_1, \dots, v_{n_f}] \in \mathbb{R}^{d_v \times n_f}$where d_k and d_v are the sizes of the key vector and the value vector. The key matrix and value matrix are written as follows:

$$K = W_K \times F \tag{2.7}$$

$$V = W_V \times F \tag{2.8}$$

In the equation (2.7) the value of the key matrix (K) is obtained from the multiplication of the weight of the key matrix (K) with the F matrix. The value of the value matrix (V) in the equation (2.8) is obtained by multiplying the weight of the value matrix (V) by the F matrix.. The weights of $W_K = \mathbb{R}^{d_k \times d_f}$ and $W_V = \mathbb{R}^{d_v \times d_f}$ can be learned automatically by the model during training or can be predetermined, for example it can be defined as an identity matrix to retain the original feature vectors. The feature vectors are further processed into inputs of the query model. The output of the query model is a Query vector (q) which is used to determine which vectors should be considered or given greater weight.

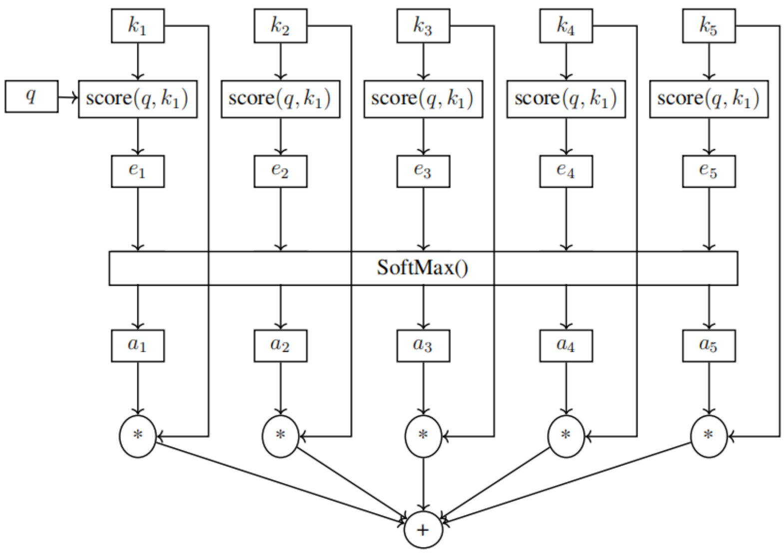

Figure 2 Attention mechanism architecture

Figure 2. is a flow of how the attention mechanism process takes place which is adopted from research Li *et al.* (2023). The values of Key (K), Value (V), and Query (q) are used as input for the attention model (attention mechanism).

The attention mechanism can be defined as a mapping from a query to a sequence of key pairs (Kavianpour ***et al***., 2023). The goal of the attention mechanism is to produce a weighted average value of the value vector (v) in the value matrix (V). The attention mechanism process involves three stages, namely.

1. Attention Scoring, calculation of similarity or correlation between query and each key. The result of this calculation shows how important the information in each key vector (k) is based on the query. Key matrix (K) and query value (q) will be passed to the attention scoring function to produce an attention score $e = [e_1, \dots, e_n] \in \mathbb{R}^{n_f}$. The calculation is as follows:
$$e_l = score(q, k_l) \tag{2.9}$$
The value e_l is the attention score and the score() function is a scoring function. The attention score e_l indicates how important the information in the key vector k_l based on query.
2. Attention Alignment, the scores obtained in attention scoring have a wide data range. However, since the goal of the attention mechanism is to produce a weighted average, the scores need to be redistributed through an attention alignment function. The function used to normalize the attention scores is the softmax function. The equation is as follows:
$$a_l = softmax(e_l, e) \tag{2.10}$$
The value $a_l \in \mathbb{R}^1$ is the attention weight of the l-th vector. The vector of attention weights $a = [a_1, \dots, a_n] \in \mathbb{R}^{n_f}$ used in the context vector process.
3. Context vector, The output of the attention mechanism is the weighted average result of the columns of the value matrix (V) obtained from the sum of the multiplication values between the attention weights (a_l) and the value (v) as follows:
$$c = \sum_{l=1}^{n_f} a_l \times v_l \tag{2.11}$$
The result of the context vector $c \in \mathbb{R}^{d_v}$ is the weighted sum of the previously calculated vectors. This means that throughout the first to last data iteration, the value of c will continue to be updated.

The context vector values (c) are then used in the model output to generate prediction output. There are several types of attention mechanisms used in deep learning, the following are the main types:

1. Bahdanau Attention / Additive Attention
Bahdanau Attention or additive attention is a type of attention mechanism that was first introduced by Bahdanau *et al.* (2014). In additive attention, the way to calculate attention scoring is by using the concat/additive method.
$$score(q, k_l) = w^T \tanh{(W_1 q + W_2 k_l)} \tag{2.12}$$
In the equation (2.12) query (q) and key vector (k) are multiplied by the weight matrix and then added together. Weights $w \in \mathbb{R}^{d_w}, W_1 \in \mathbb{R}^{d_w \times d_q}$, and $W_2 \in \mathbb{R}^{d_w \times d_k}$ are the weight matrices trained during model training, where d_w is the size of the weight matrix. The result of the calculation is multiplied by the neural network activation function to calculate attention scoring. The additive attention mechanism uses a feed-forward neural network to compute attention weights and uses the previous and current attention states during the computation which allows it to learn more complex relationships between data.

2. Luong Attention / Multiplicative Attention
Luong Attention or multiplicative attention is a type of attention mechanism introduced by Luong *et al.* (2015). The multiplicative attention mechanism uses a simple mathematical approach to calculate attention scores. There are three ways to calculate attention scoring, namely dot, general, and concatenation.
$$score(q, k_l) = \begin{cases} q^T k_l & dot - product \\ q^T W k_l & general \\ w^T \tanh\big(W(q + k_l)\big) & concat \end{cases} \tag{2.13}$$
In the equation (2.13) The Dot function calculates attention scoring by multiplying the query (q) by the key vector (k). The General function calculates attention scoring by multiplying the query (q) by the key vector (k) and the matrix weights $W \in \mathbb{R}^{d_k \times d_q}$. The Concat function is almost the same as attention scoring in Attention Bahdanau, the difference is that the query (q) and the key vector (k) do not have their own matrix weights, they only have general matrix weights $W \in \mathbb{R}^{d_w \times d_k + d_q}$. The multiplicative attention mechanism only uses the current state to calculate the attention score, which makes it more computationally efficient.

3. Results and discussions

3.1. *Data*

The data used in this study is secondary data, namely daily stock data from PT Indosat Ooredoo Hutchison for the period 1 January 2010 to 31 December 2023 with a total of 3377 data obtained from the official website.Google Finance. The data contains the opening price (open), highest price (high), lowest price (low), closing price (close), volume, and date on one day.

Table 1 PT Indosat Ooredoo Hutchison stock data sample

Date	Open	High	Low	Close	Volume
1/4/2010 16:00:00	4700	4725	4650	4700	2599000
1/5/2010 16:00:00	4750	4800	4700	4750	6005000
1/6/2010 16:00:00	4700	4750	4675	4700	2824000
1/7/2010 16:00:00	4725	4725	4675	4725	7282500
1/8/2010 16:00:00	4900	4925	4725	4900	9277500
12/21/2023 16:00:00	9400	9450	9200	9400	7275800
12/22/2023 16:00:00	9400	9450	9250	9400	2460800
12/27/2023 16:00:00	9400	9425	9275	9375	3818100
12/28/2023 16:00:00	9375	9400	9275	9350	1019100
12/29/2023 16:00:00	9350	9400	9250	9375	1550500

3.2. Data Prepocessing

The process of preparing data by deleting or changing data.that are incorrect, incomplete, duplicated, or formatted incorrectly. Feature selection: The process of selecting the most relevant features from the original dataset to be used in modeling. The purpose of feature selection is to improve model performance and reduce overfitting. In a study conducted by Htun *et al.* (2023) concluded that feature selection improves the performance of the prediction model applied to stock price prediction.

Data is divided into two parts, namely training data and testing data with a ratio of 80:20. Training data will be used first to form each training model. There are different scales or ranges in the input data, so data normalization is carried out to maintain consistency and quality data. In the research Lu *et al.* (2021)The normalization method used to normalize input data is Z-score (standardization). Z-score normalization is a method for changing data so that it has an average of 0 and a standard deviation of 1 which is defined as follows:

$$y_i = \frac{x_i - \bar{x}}{s} \tag{3.1}$$

in where y_i is mark standardization, x_i is data input, $\bar{x}$ is average data input, and s is the standard deviation of the input data. With Z-score normalization, values greater than the mean will have a positive Z-score, while values less than the mean will have a negative Z-score.

3.3. Model Implementation

Building model architecture for each model (CNN, LSTM, BiLSTM, CNN-BiLSTM, CNN with attention mechanism, LSTM with attention mechanism, BiLSTM with attention mechanism, and CNN-BiLSTM with attention mechanism) by considering the best accuracy results. The process of optimizing hyperparameters in a deep learning model to improve model performance. Hyperparameters are parameters that are not learned from data but have been determined before the model training process begins. The hyperparameter tuning method used in this study is Grid search, which is by trying all combinations of a predetermined set of hyperparameters.

Table 2 Best parameters of the model

Model	Filters	Kernel size	Pooling size	Hidden units	Dropout rate	Batch size	Epoch
CNN	128	2	1	-	0.1	64	50
LSTM	-	-	-	64	0.1	32	100
BiLSTM	-	-	-	128	0.2	32	100
CNN-BiLSTM	128	2	2	128	0.1	32	100
CNN-Attention	128	2	2	-	0.1	32	100
LSTM-Attention	-	-	-	128	0.2	64	100

BiLSTM-Attention	-	-	-	128	0.2	32	100
CNN-BiLSTM-Attention	128	1	2	128	0.2	32	100

The result of hyperparameter tuning is the best parameters applied in each model. After the training model is formed, each model is trained using the train data. Then each training model is saved to be tested on the stock price test data. Testing the pre-trained model with testing data.

3.4. Result

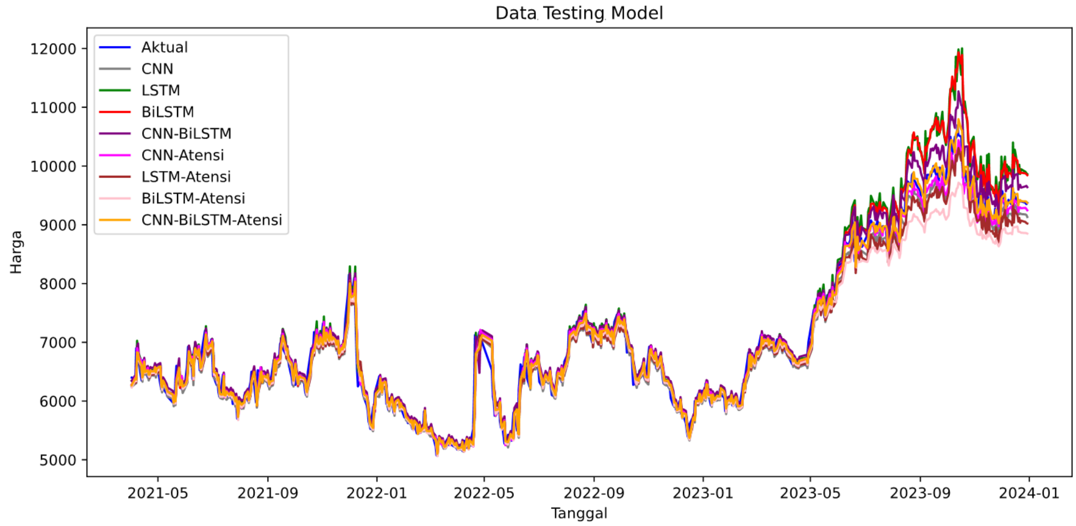

Figure 3 Model Prediction Results

Figure 3. displays a comparison between actual and predicted prices across all previously built models. Each model is represented by a different colored line, allowing for a visual comparison between each model's predictions and the actual data.

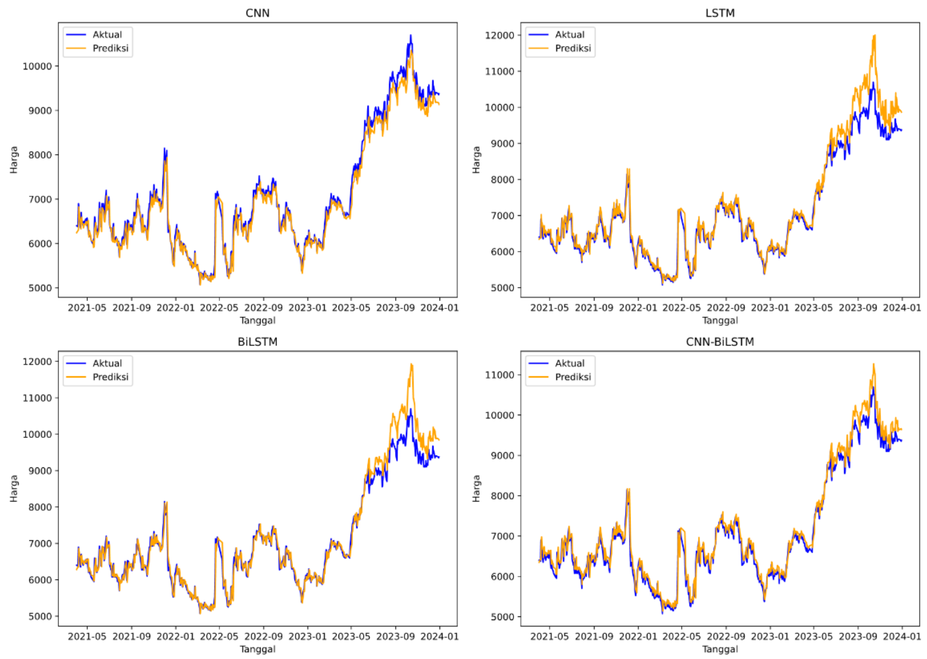

Figure 4 Predict model without attention mechanisms

The CNN model appears to have predictions that are closest to the actual values, especially during periods of high volatility. The LSTM and BILSTM models show slight overestimation on some price peaks, especially towards the end of the period. The CNN-BILSTM model seems to provide a good balance, with predictions that are quite accurate and consistent throughout the period. Despite some minor differences, the four models generally show comparable performance in predicting stock price movements.

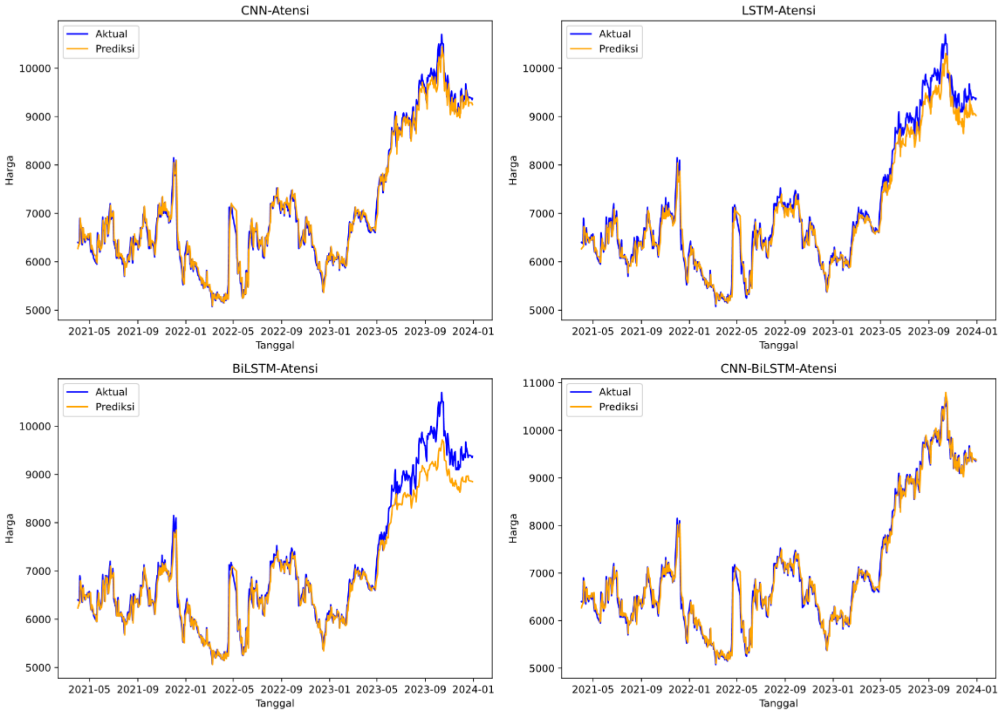

Figure 5 Predict model with attention mechanisms

All models tend to predict long-term trends well. All models seem to follow the general pattern of the actual data well. There are several differences in capturing short-term fluctuations, especially at price peaks. BiLSTM-Attention shows some larger differences between predictions and actual values, particularly at price peaks. The CNN-BiLSTM-Attention model appears to have predictions that are closest to the actual values, especially in the final period. The output values calculated by the output layer are compared with the actual values of the testing data, and the corresponding errors are calculated using Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE), and R-squared (R^2).

Table 3 Accuracy Results

Model	**MAE**	**MAPE**	**RMSE**	**R^2**
CNN	0.0929	5.6590	0.1261	0.9826
LSTM	0.1364	6.9296	0.2201	0.9470
BiLSTM	0.1286	6.3723	0.2160	0.9490
CNN-BiLSTM	0.0976	6.3644	0.1320	0.9809
CNN-Attention	0.0769	4.9314	0.1100	0.9868
LSTM-Attention	0.1053	6.0851	0.1471	0.9764
BiLSTM-Attention	0.1271	6.3043	0.1980	0.9571
CNN-BiLSTM-Attention	0.0658	4.6217	0.0971	0.9897

Table 3. shows a comparison of the performance of various deep learning models in predicting daily shares of PT Indosat Ooredoo Hutchison. The CNN model shows quite good performance with a relatively low MAE of 0.0929 and RMSE of 0.1261, as well as a high R^2 of 0.9826. The LSTM model has the lowest performance among all models, with the highest RMSE of 0.2201 and the lowest R^2 of 0.9470. The BiLSTM model shows an improvement over the LSTM

model with the lowest MAPE of 6.3723% and an increase in R² of 0.9490. The CNN-BiLSTM model performs better than the LSTM and BiLSTM models with an R² of 0.9809 but is still below the CNN model. The CNN model with the attention mechanism shows an improvement in all evaluation matrices compared to the CNN model. The MAE value decreased by 0.0160 (an increase in accuracy of 17.22% compared to the CNN model without the attention mechanism). The LSTM model with attention mechanism shows significant improvement compared to the LSTM model without attention mechanism in all evaluation matrices. This can be seen in the MAE and RMSE values which experience a large increase in accuracy. The MAE value decreases by 0.0311 (accuracy increase of 22.80%) and the RMSE decreases by 0.0730 (accuracy increase of 33.17%) compared to the LSTM model. The BiLSTM model with attention mechanism shows a slight increase in all evaluation matrices compared to the BiLSTM model. This can be seen in the MAPE value which decreases by 0.0680 (accuracy increase of 1.07%) and the RMSE decreases by 0.0180 (accuracy increase of 8.33%). The CNN-BiLSTM model with attention mechanism shows a very significant increase significantly compared to the CNN-BiLSTM Model in all evaluation matrices. The MAE value decreased by 0.0318 (accuracy increase of 32.58%), the MAPE value decreased by 1.2058% (accuracy increase of 22.48%), the RMSE value decreased by 0.0349 (accuracy increase of 26.44%), and the R² value increased by 0.0108 (accuracy increase of 1.10%). The CNN-BiLSTM model with the attention mechanism is the model with the best performance among all models in predicting the daily stock price of PT Indosat Ooredoo Hutchison.

4. Conclusion

Based on the results of the test evaluation for forecasting accuracy, the CNN-BiLSTM model with Attention Mechanism is proven to be the most superior model compared to other models. This can be seen from the lowest loss value and smaller MAE, MAPE, and RMSE evaluation matrices while the highest R² value. Other models also show good performance in terms of accuracy but are still below the CNN-BiLSTM model with Attention Mechanism. The addition of attention mechanism is proven to be effective in improving model performance. Models with the addition of attention mechanism show a consistent decrease in error rate and increase in R², indicating that the attention mechanism helps the model to better capture important patterns in stock data.

References

Arifin, I., Haidi, R. F., & Dzalhaqi, M. (2021). Penerapan Computer Vision Menggunakan Metode Deep Learning pada Perspektif Generasi Ulul Albab. *Jurnal Teknologi Terpadu*, 7(2), 98–107. https://doi.org/10.54914/JTT.V7I2.436

Asrafil, A., Paliwang, A., Ridwan, M., Septian, D., Cahyanti, M., Ericks, D., Swedia, R., & Informatika, J. T. (2020). Klasifikasi Penyakit Tanaman Apel Dari Citra Daun dengan Convolutional Neural Network. *Sebatik*, 24(2), 207–212. https://jurnal.wicida.ac.id/index.php/sebatik/article/view/1060

Bahdanau, D., Cho, K. H., & Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align and Translate. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. https://arxiv.org/abs/1409.0473v7

Brauwers, G., & Frasincar, F. (2023). A General Survey on Attention Mechanisms in Deep Learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(4), 3279–3298. https://doi.org/10.1109/TKDE.2021.3126456

Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A., & De, D. (2019). Fundamental concepts of convolutional neural network. *Intelligent Systems Reference Library*, 172, 519–567. https://doi.org/10.1007/978-3-030-32644-9_36

Hou, J., Wang, Y., Hou, B., Zhou, J., & Tian, Q. (2023). Spatial Simulation and Prediction of Air Temperature Based on CNN-LSTM. *Applied Artificial Intelligence*, 37(1). https://doi.org/10.1080/08839514.2023.2166235

Htun, H. H., Biehl, M., & Petkov, N. (2023). Survey of feature selection and extraction techniques for stock market prediction. *Financial Innovation*, 9(1), 1–25. https://doi.org/10.1186/S40854-022-00441-7/FIGURES/3

Hu, Z., Zhao, Y., & Khushi, M. (2021). A Survey of Forex and Stock Price Prediction Using Deep Learning. *Applied System Innovation 2021*, Vol. 4, Page 9, 4(1), 9. https://doi.org/10.3390/ASI4010009

Kavianpour, P., Kavianpour, M., Jahani, E., & Ramezani, A. (2023). A CNN-BiLSTM model with attention mechanism for earthquake prediction. *Journal of Supercomputing*, 79(17), 19194–19226. https://doi.org/10.1007/S11227-023-05369-Y/METRICS

Le, X. H., Ho, H. V., Lee, G., & Jung, S. (2019). Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting. *Water 2019*, 11(7), 1387. https://doi.org/10.3390/W11071387

Li, J., Xu, C., Feng, B., & Zhao, H. (2023). Credit Risk Prediction Model for Listed Companies Based on CNN-LSTM and Attention Mechanism. *Electronics 2023*, 12(7), 1643. https://doi.org/10.3390/ELECTRONICS12071643

Lindemann, B., Müller, T., Vietz, H., Jazdi, N., & Weyrich, M. (2021). A survey on long short-term memory networks for time series prediction. *Procedia CIRP*, 99, 650–655. https://doi.org/10.1016/J.PROCIR.2021.03.088

Lu, W., Li, J., Wang, J., & Qin, L. (2021). A CNN-BiLSTM-AM method for stock price prediction. *Neural Computing and Applications*, 33(10), 4741–4753. https://doi.org/10.1007/S00521-020-05532-Z/TABLES/3

Luong, M. T., Pham, H., & Manning, C. D. (2015). Effective Approaches to Attention-based Neural Machine Translation. Conference Proceedings - EMNLP 2015: *Conference on Empirical Methods in Natural Language Processing*, 1412–1421. https://doi.org/10.18653/v1/d15-1166

Nie, Q., Wan, D., & Wang, R. (2021). CNN-BiLSTM water level prediction method with attention mechanism. *Journal of Physics: Conference Series*, 2078(1). https://doi.org/10.1088/1742-6596/2078/1/012032

Prasad, J. P., & Venkatesham, T. (2021). Big data analytics in the retail sector. *International Journal of Computer Science and Mobile Computing, 10*(7), 34–38. https://doi.org/10.47760/ijcsmc.2021.v10i07.005

Prasetyo, B. S., Adytia, D., & Aditya, I. A. (2023). Time Series Forecasting of Electricity Load Using Hybrid CNN-BiLSTM with an Attention Approach: A Case Study in Bali, Indonesia. *2023 International Conference on Data Science and Its Applications*, ICoDSA 2023, 460–464. https://doi.org/10.1109/ICODSA58501.2023.10277176

Puteri, D. I. (2023). Implementasi Long Short Term Memory (LSTM) dan Bidirectional Long Short Term Memory (BiLSTM) Dalam Prediksi Harga Saham Syariah. *Euler : Jurnal Ilmiah Matematika, Sains Dan Teknologi,* 11(1), 35–43. https://doi.org/10.34312/EULER.V11I1.19791

Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2019). The Performance of LSTM and BiLSTM in Forecasting Time Series. *Proceedings - 2019 IEEE International Conference on Big Data, Big Data 2019*, 3285–3292. https://doi.org/10.1109/BIGDATA47090.2019.9005997

Sun, S., Sun, J., Wang, Z., Zhou, Z., & Cai, W. (2022). Prediction of Battery SOH by CNN-BiLSTM Network Fused with Attention Mechanism. *Energies 2022*, 15(12), 4428. https://doi.org/10.3390/EN15124428

Wu, M., Liu, W., & Zheng, S. (2024). Intelligent Food Safety: A Prediction Model Based on Attention Mechanism and Reinforcement Learning. *Applied Artificial Intelligence,* 38(1). https://doi.org/10.1080/08839514.2024.2379731

Zhang, J., Ye, L., & Lai, Y. (2023). Stock Price Prediction Using CNN-BiLSTM-Attention Model. *Mathematics 2023*, 11(9), 1985. https://doi.org/10.3390/MATH11091985