



Contents lists available at [JESA](https://jesa.aks.or.id/index.php/jesa/index)

Journal of Engineering and Science Application

journal homepage:
<https://jesa.aks.or.id/index.php/jesa/index>



Stock Price Prediction Modeling Using Recurrent Neural Network and Long-Short Term Memory

Afrida Nur Salsabila^a, Sri Huning Anwariningsih^b, Dahlan Susilo^c

^{a,b,c}Sahid Surakarta University, Jl. Adisucipto No. 154 Jajar Surakarta, 57144, Indonesia

Article Info

Keywords:

BCA
LSTM
Prediction
RNN
Stock

ABSTRACT

Stock price fluctuations were very difficult to predict the direction of the changes. There were generally estimated to follow three analysis techniques: technical, fundamental, and sentiment. Technical analysis involves observing prices in the past, fundamental analysis is related to the analysis of ongoing business situations, while sentiment analysis includes stock prices that were affected by business aspects, current information, and business activities. Valid price data of the BCA Company used was the stock price from 2019 until 2024. The purpose of this study is to find alternative models of the RNN and LSTM models. The methods used in this study are the documentation method and the optimization method. Accuracy measurements used Mean Square Error (MSE) and Mean Absolute Error (MAE) metrics. The results of stock prediction using the RNN model got poor results with epoch 10 obtaining an accuracy of 61.3%, while using the LSTM model obtained quite good results with epoch 10 obtaining an accuracy of 87.7%. Stock predictions using the combined RNN-LSTM models were able to get good results with epoch 10 obtaining an accuracy of 93.3%.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Dahlan Susilo
Department of Informatics
University of Sahid Surakarta
Address campus Jl. Adisucipto No. 154 Jajar Surakarta, 57144, Indonesia
Email: dahlan.susilo@usahidsolo.ac.id

1. INTRODUCTION

Investing in stocks is currently one of the popular choices among investors because it is able to provide significant potential profits. However, the volatility and complexity of the stock market make stock price forecasting a challenge. Stock price movements are very difficult to predict the direction of the change. Stock price movements are generally estimated to follow three analysis techniques: technical, fundamental, and sentiment. Technical analysis involves observing prices in the past, fundamental analysis is related to the analysis of ongoing business situations, while sentiment analysis includes stock prices that are affected by

business aspects, current information, and business activities. Liquidity proxied with CR has no effect on the share price, while profitability proxied with ROA and market ratio proxied with EPS have a positive effect on the share price in the company [1].

Bank Central Asia Company (BCA) was one of the largest private banks in Indonesia, with a capital of IDR 42.93 trillion. BCA is included in the ranks of major banks in Indonesia listed on the Indonesia Stock Exchange (IDX). BCA's stock performance continues to show outstanding results in the banking sector in the country. According to information from CNBC Indonesia, Stock of BCA had the highest market capitalization value on the IDX. In general, stocks with large market capitalization values are often the top choice for investors for long-term investments because they reflect the company's good growth potential as well as have lower risk. When deciding to invest, BCA's share price is one of the most important aspects to consider. Partially the Current Ratio, Debt to Equity Ratio, Return On Asset and Price Earning Ratio have no effect on the stock price while the Return On Equity and Earning Per Share have a significant negative effect on the stock price[2].

A better analytical approach is indispensable to help investors make informed decisions. The growth of investors in the capital market continues to increase every year. In 2021, the number of capital market investors in Indonesia has the fastest growth, where the number of Single Investor Identification has increased significantly compared to the previous year. Meanwhile, the number of capital market investors in Indonesia in January 2024 is 12.32 million investors [3]. ROA has a negative and significant effect on the stock price of Textile and Garment companies, while CR and DAR have a positive and significant effect on the stock price [4].

Advances in artificial intelligence, especially deep learning, have become the most widely chosen method to improve the accuracy of predictions in the stock market. Deep learning uses artificial neural networks to analyze complex patterns in data by taking features from stock price history. Among the various approaches available, RNN and LSTM have proven to be effective in handling time series data, including stock price movements. RNNs excel at processing sequential data and recognizing temporal patterns, but often experience vanishing gradient issues when dealing with long-term data. To overcome the weaknesses of RNNs, LSTMs were developed as modifications that were able to recall information over a longer period of time and deal with the problem of gradient vanishing more effectively [5].

Research using the RNN model with the LSTM architecture can be optimized with the Adaptive Moment Estimation (ADAM) method capable of producing predictions that are close to the actual stock price. The results of this study are evidenced by a low loss value of 0.0109012. Meanwhile, the Mean Percentage Error (MPE) score was obtained at 1.74% resulting in good accuracy [6]. The LSTM model is effective in predicting sales trends across all price categories, so it can assist companies in stock management and more strategic marketing planning [7].

The research using the LSTM model with a BCA case study. From the models that have been made, the best performance is produced based on changing the number of batch sizes with the number of epochs as many as 5 while the batch size size is 1 and the number of units 50 is optimized with ADAM. This combination results in an average MAE score of 96,924, Mean Squared Error (MSE) of 16185,226 and Relative Mean Error (RME) of 127,221 [8].

Research using the RNN-LSTM hybrid model by using forex market case studies proves that this model is superior to the regular RNN and LSTM models. The results of this study were proven by using RMSE, MAE and MAPE metrics to produce lower values than the ordinary RNN-LSTM model. oleh Faru dkk., (2023)

The digital financial institution object using the RNN-LSTM hybrid method produces a higher f-score value and accuracy than the f-score value and accuracy of the traditional RNN and LSTM models [9]. Through the study of these studies, researchers can gain insight into effective model approaches and the optimization of appropriate parameters in the development of predictive models based on RNN-LSTM.

Prediction is the process of systematically predicting what is most likely to happen in the future based on past and present information that we have, so that errors (the difference between something happening and the result of the forecast) can be minimized [10]. Predictions do not have to provide an exact answer to the event that will occur, but rather try to find the answer as close as possible to what will happen [8].

Stocks were certificates or securities that show the ownership of a company, the certificate shows the value of the shares owned, the type of shares owned and the rights and obligations of each shareholder [2]. Stocks were proof of ownership in a company that goes public[11]. From the above two opinions, it can be concluded that shares are proof of ownership of the assets of the company that issued the stock. Stock are valuable documents as proof of ownership of a company [8]. Stocks are high-risk investment instruments that offer high return value. High return values also carry high risks due to the volatile nature of stocks [12]. Stocks can provide considerable profits. However, stock price movements are volatile which causes stock investment to be risky.

RNN is a type of artificial neural network architecture whose processing is repeatedly called upon to process sequential data input. RNN falls under the category of deep learning because data is processed through

many layers. RNN has come a long way and has revolutionized fields such as Natural Language Processing (NLP) [10]. The findings affirm the potential of utilizing advanced machine learning techniques like Random Forest in conjunction with SMA indicators to enhance the predictability of stock market trends, offering valuable insights for investors and financial analysts [13].

2. RESEARCH METHOD

Data collection using APIs from Yahoo Finance is done by accessing available historical and real-time data. Third-party APIs use the 'yfinance' library to access Yahoo Finance data more easily. 'yfinance' is a Python library that allows users to download stock market data. The yfinance API proves invaluable for accessing detailed financial data from Yahoo Finance directly into Python applications. It simplifies the process of fetching historical market data, financial statements, and stock actions for various securities such as stocks and indices. By leveraging yfinance, users can efficiently perform financial analyses and gain insights crucial for finance, investment, and trading decisions [14]. So no account is required, just install the Yahoo Finance API on the python environment. The data used from September 1, 2019 to September 30, 2024, totaled 1856 data.

The first stage in stock prediction is preprocessing, which is a step taken to prepare raw data before it is used for model development. The modeling stage includes: Data Collection The source of data in this study was taken from secondary data (yahoo.finance.com) using closing data in the form of CSV (Comma Separated Values); Preprocessing involves transforming raw data into a format suitable for further processing by an RNN or LSTM model. This process includes normalization and noise removal. After that, the data is divided into training set data and test data using an 80:20 ratio; Model development is carried out by building a model architecture with an input layer to receive sequential data in the form of historical data and then adding an RNN layer that aims to capture short-term patterns, after which two hidden layers are in the form of an LSTM layer that aims to capture long-term patterns in the dataset; Training Models are used to train a model that has been created by configuring epochs and batch size parameters for training. Furthermore, to measure prediction errors, a loss function in the form of MSE is used, then the model will be optimized using ADAM to speed up the training process; Model testing is carried out to ensure that the model has good performance in training and can generalize existing data; Model Evaluation is used to measure model performance, in this process it includes the use of evaluation metrics in the form of MSE, MAE and R^2 scores.

3. RESULTS AND DISCUSSION

The stock data that has been taken from the Yahoo Finance API is then used as input to the program to be created. In this input, it is done by using the 'yf.download' function and the start date 'start' and the end date 'end' to download historical data. After the data is successfully downloaded and stored in the variable 'saham_BCA', the 'to_csv' method is used to save the data in CSV form. The gross stock price data from September 2019 until September 2024 is presented in Table 1.

Table 1. Gross stock price data for the period 2019 to 2024

Price Ticker Date	Close BBCA.JK	High BBCA.JK	Low BBCA.JK	Open BBCA.JK	Volume BBCA.JK
2-Sep-2019	5,204.09	5,290.83	5,204.09	5,290.83	42,429,000
3-Sep-2019	5,251.80	5,251.80	5,204.09	5,204.09	42,743,500
4-Sep-2019	5,221.44	5,256.14	5,208.43	5,256.14	41,979,000
5-Sep-2019	5,238.79	5,277.82	5,217.10	5,277.82	53,650,500
6-Sep-2019	5,225.78	5,256.14	5,225.78	5,256.14	54,349,000
:	:	:	:	:	:
:	:	:	:	:	:
25-Sep-2024	10,473.88	10,473.88	10,160.15	10,304.95	56,890,500
26-Sep-2024	10,329.08	10,498.01	10,304.95	10,498.01	56,705,500
27-Sep-2024	10,280.81	10,377.35	10,039.48	10,136.01	46,997,500

The data that has been successfully entered will go through a cleaning process that involves several activities, including checking duplicate data and checking for missing value. In this stage, two processes are carried out, namely Duplication Data Checking. If any data is identified as duplicated, it must be deleted either to ensure that the data has a different uniqueness. The following is a pseudocode for the duplicate data checking process:

```

Start
1. SET duplication = saham_bca[saham_bca.duplicated()]. #To identify duplicated rows.
2. SET amount_duplication = saham_bca.duplicated().sum(). #To calculate the number of duplicated rows.
3. PRINT "Amount of duplicates: " + amount_duplication. #To display the number of duplicates.
4. PRINT "Duplicate rows:" #To display duplicated rows.
End

```

Missing data or missing value is a condition where there are incomplete or empty values in one or several criteria [15]. Missing value is undesirable in machine learning and data mining because missing value will cause many problems. Furthermore, if there is a missing value, the missing value will be determined based on the existing data or by deleting rows and columns that are identified as having missing values. Net stock price data after filtering can be seen in Table 2.

Table 2. Net stock price data after filtering

Price Ticker Date	Close BBCA.JK	High BBCA.JK	Low BBCA.JK	Open BBCA.JK	Volume BBCA.JK
2-Sep-2019	5,204.09	5,290.83	5,204.09	5,290.83	42,429,000
3-Sep-2019	5,251.80	5,251.80	5,204.09	5,204.09	42,743,500
4-Sep-2019	5,221.44	5,256.14	5,208.43	5,256.14	41,979,000
5-Sep-2019	5,238.79	5,277.82	5,217.10	5,277.82	53,650,500
6-Sep-2019	5,225.78	5,256.14	5,225.78	5,256.14	54,349,000
:	:	:	:	:	:
:	:	:	:	:	:
3-Jun-2022	10,473.88	10,473.88	10,160.15	10,304.95	158,369,100
4-Jun-2022	10,329.08	10,498.01	10,304.95	10,498.01	84,960,100
5-Jun-2022	10,280.81	10,377.35	10,039.48	10,136.01	93,027,400

The following is a pseudocode for missing value checking:

```

Start
1. stock_bca.isna().sum(). #To calculate the sum of values that is missing in each column.
End

```

Data transformation involves the process of normalizing data with the min-max scaling method to convert the data into a range of 0 to 1. Normalized data is also easier to visualize, thus aiding in data exploration analysis and dataset patterns. Here is the pseudocode of the data transformation:

```

Start
1. IMPORT MinMaxScaler FROM sklearn.preprocessing. #To import the libraries required in the data transformation process.
2. CREATE scaler AS MinMaxScaler WITH feature_range (0.1). #To initiate the Min-Max Scaler with a range of 0 until 1.
3. SET close_data = stock_bca[['Close']]. #To retrieve the 'Close' column from the BCA stock dataset.
4. SET saham_bca_scaled = scaler.fit_transform(close_data). #To get the normalized value and store it in the 'stock_bca_scaled' variable.
5. PRINT stock_bca_scaled. #To display the normalization results that have been performed in the 'Close' column.
End

```

The division between training and test data uses a ratio of 80 : 20, with 985 training data and 247 test data. The 80:20 ratio is used because it can reduce overfitting and to save time and resources in the training and testing process so that it focuses on developing a good model. Here is the pseudocode of data sharing:

```

Start
1. SET total_data = LENGHT(stock_bca_sacled). #To get the total amount of data in the normalized dataset.
2. SET train_size = INTEGER(total_data * 0.8). #To take 80% of the total data and store it in the 'train_size' variable.
3. SET test_size = total_data – train_size. #To calculate the size of the test data by subtracting the 'train_size' from the 'total_data'.
4. PRINT "Training data quantity: " + train_size PRINT "Test data quantity: " + test_size. #To display training and test data sizes.
End

```

Model development is carried out by building a model architecture with an input layer to receive sequential data in the form of historical data and then adding two hidden layers in the form of an LSTM layer that aims to capture long-term patterns in the dataset, after which it adds an RNN layer that aims to capture short-term patterns. The development architecture of the merger of the 2 models can be seen in Figure 1.

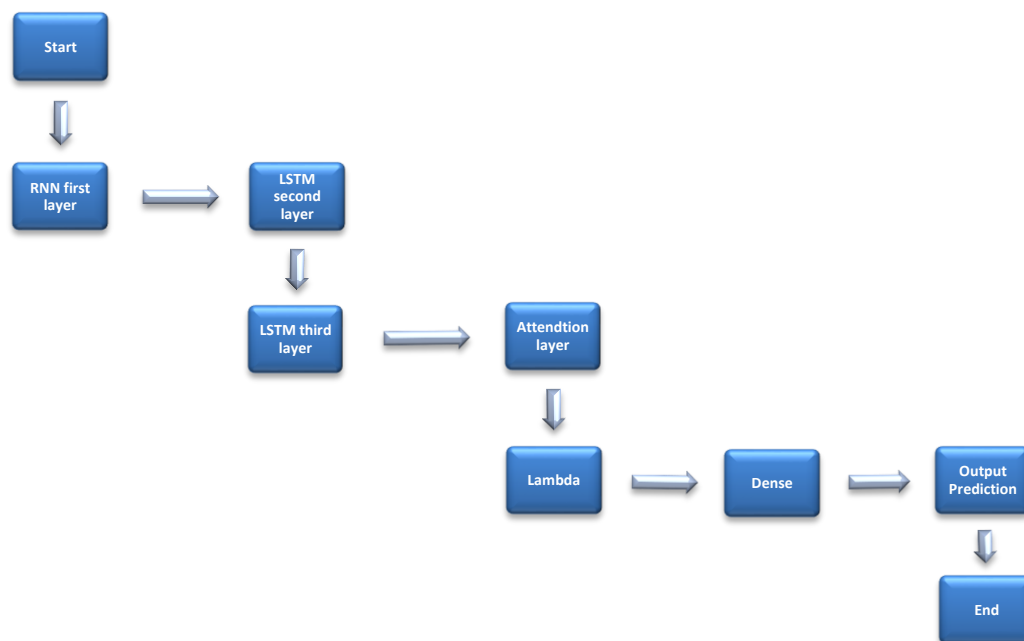


Figure 1. The development architecture of combination of RNN-LSTM models

The first RNN layer with a dropout, the RNN layer is built simply using 64 units for the 'return_sequences=True' function signifying that this layer will return the output for each step of time, not just the last output. Meanwhile, a dropout with a rate of 0.2 is used to deactivate 20% of neurons to prevent overfitting. The first LSTM layer with dropout, the RNN layer was built simply using 32 units. The LSTM layer has a better function of capturing long-term dependencies compared to regular RNNs. The second LSTM layer with dropout, the RNN layer is built simply using 64 units. This second LSTM layer is used to restore the order of output. The Attention Layer allows the model to focus on specific parts of the sequential input that are more relevant to the prediction task. With the mechanism of attention, the model can give more weight to important information, improving accuracy and efficiency in sequential data processing. Lambda is an R2 regularization parameter that controls the weight size to avoid overfitting. The dense layer is a layer where each neuron in that layer is connected to every neuron in the previous layer. This means that the Dense layer has a direct relationship between all inputs and outputs. Output Prediction: The final stage in which the model generates an output based on information that has been processed through previous layers. This output can be a classification, a numerical value prediction, or another form according to the model's objectives.

Model training is a process by which the model that has been created will be trained so that it can get good accuracy results. Once these layers are arranged, the model can be created with the Keras function by

defining the inputs and outputs. Furthermore, the model will be compiled with the Adam optimizer and the MSE loss function for regression problems. Here is a pseudocode for model development:

```

Start
1. SET input_layer. #Berfungsi to set up the model input layer.
2. SET rnn_layer = SimpleRNN(units=32, return_sequences = True) (input_layer) SET rnn_layer =
Droupot(0.2)(rnn_layer).
3. SET lstm_layer1 = LSTM(units=32, return_sequences=True)(rnn_layer) SET lstm_layer1 =
Dropout(0.2)(lstm_layer1).
4. SET lstm_layer2 = LSTM(units=32, return_sequences=True)(rnn_layer1) SET lstm_layer2 =
Dropout(0.2)(lstm_layer2)
5. SET attention_layer = Attention()(lstm_layer_2, lstm_layer_2))
6. SET merged_layer = tf.keras.layers.Lambda(lambda x: tf.reduce_sum(x, axis=1))(attention_layer)
7. SET output_layer = Dense(units=1)(merged_layer)
8. SET model = Model(inputs=input_layer, outputs=output_layer)
9. CALL model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=0.001),
loss='mean_squared_error')
End

```

After creating the model, the next is the training of the RNN-LSTM model. This exercise aims for the model to understand the data to be predicted, by training the model, the model will know the patterns of data movement in a certain time and it is likely that the prediction produced will be the higher the level of accuracy. The following is a pseudocode in model training:

```

Start
1. SET x_train, y_train. #To provide training data and targets for training.
2. SET x_train, x_val, y_train, y_val = train_test_split(x_train, y_train,
test_size=0.2, random_state=42). #To divide training data into training data and validation data. The
parameter 'test_size = 0.2' indicates that 20% of the data will be used for validation, while 'random_state = 42'
is used to ensure consistent results.
3. SET history = model.fit (x_train, y_train, epochs = 10, batch_size = 64, validation_data = (x_val,
y_val), verbose=1). #To train models with training data.
End

```

In this training the function 'validation_data=(x_val, y_val)' is used to monitor the performance of the model during training, it can also help in detecting overfitting, where the model performs well on training data but poorly on invisible data. While the function 'verbose=1' is to determine the level of detail displayed during training including the losses in each epoch. The last stage in this training is to keep a training history that can be used to analyze the model's performance after training in the form of a graphic plot (Figure 2).

Model testing is carried out to ensure that the model has good performance in training and can generalize the existing data. In this stage, a holdout method is used where the dataset is divided into two parts, between the training data and the test data randomly. Denormalization is a step to return normalized data to its original range. The purpose of denormalization is so that the prediction results can be interpreted and compared with their original values.

In this stage, model testing is carried out using test data to evaluate the accuracy of predictions such as the use of MAE, MSE and R^2 score metrics. Once the model has been evaluated, the final step is to visualize the prediction results compared to the actual data for further analysis processes. Here is the pseudocode of the model evaluation:

```

Start
1. SET y_tested_rescaled, SET predictions_rescaled. #To prepare the necessary data.
2. SET mse = mean_squared_error (y_test_rescaled, predictions_rescaled) #To calculate MSE
between target and prediction data.
3. MAE SET = mean_absolute_error(y_test_rescaled, predictions_rescaled) #To calculate MAE
between target and prediction data.
4. SET r2 = r2_score(y_test_rescaled, predictions_rescaled). #To calculate the r2 between the target
data and the prediction.
End

```

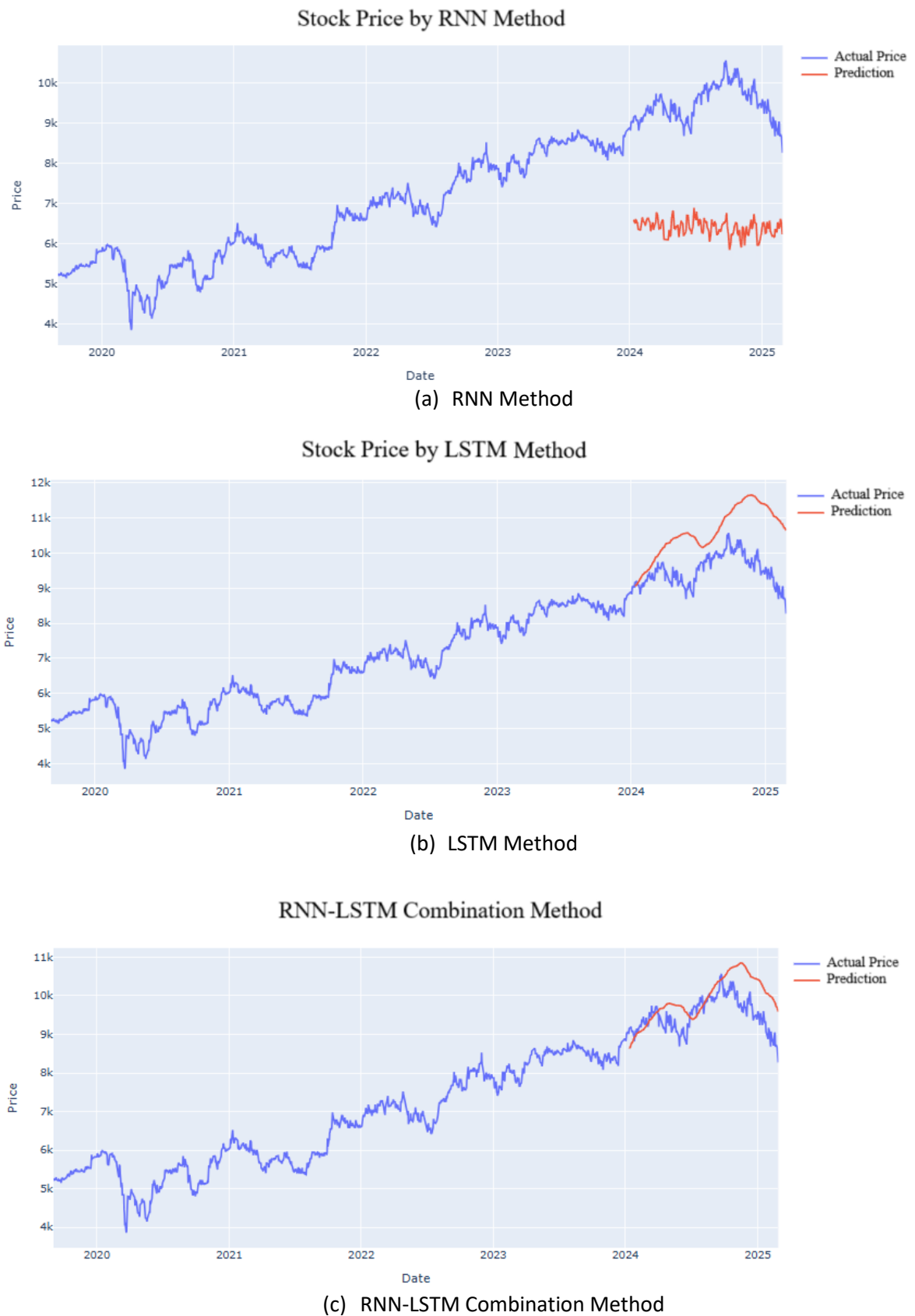


Figure 2. The relationship between the stock price and the prediction of the three methods

4. CONCLUSION

The results of stock prediction using the RNN model got poor results with epoch 10 obtaining an accuracy of 61.3%, while using the LSTM model obtained quite good results with epoch 10 obtaining an accuracy of 87.7%. Stock predictions using the combined RNN-LSTM model were able to get good results with epoch 10 obtaining an accuracy of 93.3%. The combined RNN-LSTM model has the best prediction results close to the actual stock price. Thus, this model can be used to predict the future price of stocks.

ACKNOWLEDGEMENTS

The authors are would like to express their gratitude for the head of research and community service institution of Sahid university in many ways of its contributions such as sponsorship and funding.

REFERENCES

- [1] A. Imelda, S. A. C. Sihono, and D. R. Anggarini, "Pengaruh Likuiditas, Profitabilitas, dan Rasio Pasar Terhadap Harga Saham (Studi Kasus Pada Perusahaan Indeks LQ45 Yang Terdaftar Di Bursa Efek Indonesia Periode 2017-2021)," *Journals of Economics and Business*, vol. 2, no. 2, pp. 17–25, Sep. 2022, doi: 10.33365/jeb.v2i2.114.
- [2] T. J. Prasetya, V. A. Dina Safitri, and F. Yufantria, "Analisis Faktor-Faktor Yang Mempengaruhi Harga Saham," *Journals of Economics and Business*, vol. 2, no. 1, pp. 37–46, May 2022, doi: 10.33365/jeb.v2i1.75.
- [3] ksei, "Statistik Pasar Modal Indonesia," https://www.ksei.co.id/files/Statistik_Publik_Januari_2024_v3.pdf.
- [4] L. Rahmawati, H. M. K. Sari, and W. P. Setiyono, "Hebohnya U-23 Membuat Warna Harga Saham," *Journal of Economic, Bussines and Accounting (COSTING)*, vol. 8, no. 1, pp. 476–488, Feb. 2025, doi: 10.31539/costing.v8i1.13208.
- [5] M. Waqas and U. W. Humphries, "A critical review of RNN and LSTM variants in hydrological time series predictions," *MethodsX*, vol. 13, p. 102946, Dec. 2024, doi: 10.1016/j.mex.2024.102946.
- [6] S. J. Pipin, R. Purba, and H. Kurniawan, "Prediksi Saham Menggunakan Recurrent Neural Network (RNN-LSTM) dengan Optimasi Adaptive Moment Estimation," *Journal of Computer System and Informatics (JoSYC)*, vol. 4, no. 4, pp. 806–815, Aug. 2023, doi: 10.47065/josyc.v4i4.4014.
- [7] Ghufroon Tamami and M. Arifin, "Penggunaan LSTM dalam Membangun Prediksi Penjualan untuk Aplikasi Laptop Lens," *JURNAL FASILKOM*, vol. 14, no. 2, pp. 301–308, Aug. 2024, doi: 10.37859/jf.v14i2.7372.
- [8] M. N. Wathani, K. Kusrini, and K. Kusnawi, "Prediksi Tren Pergerakan Harga Saham PT Bank Central Asia Tbk, Dengan Menggunakan Algoritma Long Shot Term Memory (LSTM)," *Infotek: Jurnal Informatika dan Teknologi*, vol. 6, no. 2, pp. 502–512, Jul. 2023, doi: 10.29408/jit.v6i2.19824.
- [9] G. Musyoka, A. Waititu, and H. Imboga, "Credit Risk Modelling Using RNN-LSTM Hybrid Model for Digital Financial Institutions," *International Journal of Statistical Distributions and Applications*, vol. 10, no. 2, pp. 16–24, Apr. 2024, doi: 10.11648/j.ijds.20241002.11.
- [10] M. R. Firmansyah, R. Ilyas, and F. Kasyidi, "Klasifikasi Kalimat Ilmiah Menggunakan Recurrent Neural Network," *Prosiding 11th Industrial Research Workshop and National Seminar (IRWNS)*, vol. 11, no. 1, pp. 488–495, Sep. 2020.
- [11] H. Zreik, "Stock Certificates: The Art of Recognizing Ownership," <https://fastercapital.com/content/Stock-Certificates--Stock-Certificates--The-Art-of-Recognizing-Ownership.html>.
- [12] A. D. Milniadi and N. O. Adiwijaya, "Analisis Perbandingan Model Arima dan LSTM Dalam Peramalan Harga Penutupan Saham (Studi Kasus : 6 Kriteria Kategori Saham Menurut Peter Lynch)," *SIBATIK JOURNAL: Jurnal Ilmiah Bidang Sosial, Ekonomi, Budaya, Teknologi, dan Pendidikan*, vol. 2, no. 6, pp. 1683–1692, May 2023, doi: 10.54443/sibatik.v2i6.798.
- [13] A. M. Priyatno, L. Ningsih, and M. Noor, "Harnessing Machine Learning for Stock Price Prediction with Random Forest and Simple Moving Average Techniques," *Journal of Engineering and Science Application*, vol. 1, no. 1, pp. 1–8, Mar. 2024, doi: 10.69693/jesa.v1i1.1.
- [14] D. Chung, "How to Use yfinance API with Python," <https://www.geeksforgeeks.org/how-to-use-yfinance-api-with-python/>.
- [15] W. Sudrajat and I. Cholid, "K-NEAREST NEIGHBOR (K-NN) Untuk Penanganan Missing Value Pada Data UMKM," *Jurnal Rekayasa Sistem Informasi dan Teknologi*, vol. 1, no. 2, pp. 54–63, Nov. 2023, doi: 10.59407/jrsit.v1i2.77.