BAREKENG: Journal of Mathematics and Its Applications
September 2023 Volume 17 Issue 3 Page 1575–1584
P-ISSN: 1978-7227 E-ISSN: 2615-3017
https://doi.org/10.30598/barekengvol17iss3pp1575-1584

CRYPTOCURRENCY PRICE PREDICTION: A HYBRID LONG SHORT-TERM MEMORY MODEL WITH GENERALIZED AUTOREGRESSIVE CONDITIONAL HETEROSCEDASTICITY

Indah Manfaati Nur[1*], Rifqi Nugrahanto[2], Fatkhurokhman Fauzi[3]

[1,2,3] *Department of Statistics, Faculty of Mathematics and Natural Sciences, Muhammadiyah Semarang University*
Kedungmundu Raya Street No 18, Semarang, 50273, Indonesia

*Corresponding author's e-mail: ** ***indahmnur@unimus.ac.id***

ABSTRACT

Article History:
Received: 04th April 2023
Revised: 7th August 2023
Accepted: 16th August 2023

Keywords:
Cryptocurrency;
Generalized autoregressive heterocedasticity;
Investment;
Long-short term memory;
MAPE

Cryptocurrency is a virtual payment instrument currently popular as an investment alternative. One type of cryptocurrency widely used as an investment is Bitcoin due to its high-profit potential and risk due to unstable exchange rate fluctuations. This high exchange rate fluctuation makes trading transactions in the crypto market speculative and highly volatile. To overcome this volatility factor, this research used the Generalized Autoregressive Conditional Heteroscedasticity forecasting method to describe the heteroscedasticity factor, as well as a Recurrent Neural Network (RNN) with long-short-term memory that has feedback in modeling sequential data for time series analysis. The two methods are combined to overcome the dependency of time series data in the long term and the heteroscedastic effect of the volatility of price changes. The results of the GARCH-LSTM hybrid model in this study show a Mean Absolute Percentage Error (MAPE) value of 15.69%. The accuracy value is obtained from the division of training data by 80% and testing data by 20%, with the number of neurons as many as three and epochs of 100 using the Adam optimizer. The MAPE accuracy results show a good prediction in predicting the value.

How to cite this article:

I. M. Nur and R. Nugrahanto., " CRYPTOCURRENCY PRICE PREDICTION: A HYBRID LONG SHORT-TERM MEMORY MODEL WITH GENERALIZED AUTOREGRESSIVE CONDITIONAL HETEROSCEDASTICITY," *BAREKENG: J. Math. & App.,* vol. 17, iss. 3, pp. 1575-1584, September, 2023.

Journal homepage: *https://ojs3.unpatti.ac.id/index.php/barekeng/*
Journal e-mail: *barekeng.math@yahoo.com;* barekeng.journal@mail.unpatti.ac.id

Research Article · Open Access

1. INTRODUCTION

Investment is placing several financial resources currently owned into an object, institution, or other party to obtain future profits [1]. Investment is divided into two types, namely real investment and financial investment. Actual investment is made by placing financial resources into assets such as metals, gold, or property. In contrast, financial investment is made by placing financial resources into intangible forms such as stocks, bonds, mutual funds, and *cryptocurrencies*. *Cryptocurrency* is a means of payment for transactions carried out virtually or via the Internet [2]. According to the Commodity Futures Trading and Supervisory Agency (BAPPETI), the number of *cryptocurrency* investors increased by 78% compared to the end of 2020, with a total number of investors reaching 2.5 million. The growth in the number of investors can be attributed to the high rate of return offered by *cryptocurrencies*. Although it offers a high rate of return, *cryptocurrency* investment also has a high risk of fluctuation.

The high fluctuations in the *cryptocurrency* market cause trading to be classified as speculative and highly volatile [3]. Volatility prediction analysis is crucial for investors to determine which assets have high volatility and risk [4]. In addition, volatility can be used in price formation and risk management. Therefore, volatility analysis can help investors in making investment decisions. The *Generalized Autoregressive Conditional Heteroscedasticity* (GARCH) method can be used, which is suitable for calculating the volatility effect of price changes and explaining heteroscedasticity factors. This method considers the clustering of volatility in the time series when high and low volatility states are maintained up to a certain threshold. The *GARCH* method is suitable to factor in the volatility effect of a stock price change to explain the heteroscedasticity factor [5]. In addition to GARCH, Recurrent Neural Network (RNN) with feedback can also be used, effectively modeling sequential data in time series analysis. *Feedforward* neural networks with the *long short-term memory* (LSTM) method can also improve predictions on data in short-term conditions in recognizing data patterns [6]. Combining the two methods can overcome the problem of the dependency on time series data in the long term to form a pattern with the heteroscedastic influence of the volatility of price changes [7].

Research on the LSTM method that has been done [8] as a model to predict gold prices confirms that the RNN-LSTM model produces a small error value, so it is good at predicting. Research using the GARCH method has also been conducted [9] by producing a GARCH (1,1) model and a small AIC value. This research intends to model the prediction of bitcoin closing prices by integrating *Generalized Autoregressive Conditional Heteroscedasticity* as a factor with *Long Short-Term Memory* to determine the level of *accuracy* in the model so that it can be used as a consideration for investors in buying or selling *cryptocurrency* coins.

2. RESEARCH METHODS

In this study, researchers used a hybrid approach between *generalized autoregressive conditional heteroscedasticity* and *long-short-term memory* to analyze Bitcoin closing price data from September 1st, 2017, to October 6th, 2022. The data used in the research is secondary data obtained from the source https://id.investing.com/. This hybrid method was chosen because it can overcome the problem of time series data dependence over a long period and also consider the influence of heteroscedasticity from the volatility of price changes. In this study, researchers focused on developing data analysis techniques that effectively identify data patterns and forecast future Bitcoin price movements. Investors can make smarter investment decisions and better manage risks using the proper techniques.

2.1 Unit Root Test of Augmented Dicky-Fuller (ADF)

When testing using the Dickey-Fuller test, it is assumed that there is no correlation between ε_t and the stochastic error term. The Dickey-Fuller test was then developed into the Augmented Dickey-Fuller (ADF) test to estimate this assumption [10].

$$\Delta Y_t = \alpha_0 + \alpha_1 + \gamma Y_{t-1} + \beta_1 \sum_{i=1}^{p} \Delta Y_{t-i} + \varepsilon_t$$

Augmented Dickey-Fuller (ADF) test hypothesis:

H_0 : $\gamma = 0$ Data is not stationary (there is a unit root)
H_1 : $\gamma \neq 0$ Stationary data (no unit root)

In making a decision, the criterion is to compare the significance value (1-α) with 100%. If the p-value is smaller than α, then the null hypothesis (H_0) is rejected, so it can be concluded that the data is stationary.

2.2 ARCH-Lagrange Multiplier Test (ARCH_LM)

The Lagrange Multiplier test is used to see if there is heteroscedasticity in *time series* data. The concept of this test is that the residual variance is not only a function of the independent variable but depends on the square of the previous period's residuals [11].

$$LM = nR^2$$

H_0 : α1 = α2 = ... = αp = 0
H_1 : $\exists \alpha_i \neq 0$, i = 1,2,...,p

Where n is the number of observations, R^2 is the coefficient of determination, M is the number of lags tested. Test criteria: H_0 is rejected if the probability value of LM < or p-value < α

2.3 Ljung-Box Test

The assumption of *white noise* in the residuals is fulfilled, characterized by a process that does not show autocorrelation, or it can be said that the residuals do not have a specific pattern (zero mean and constant variance) [12].

$$Q = n\,(n+2)\sum_{i=1}^{k}\frac{\hat{\rho}_i^2}{n-1} \sim X^2_{k-np}$$

Where n refers to the sample size. The sample size indicates the number of individuals, objects, or cases observed in each treatment group, k defines the number of treatment groups being analyzed. Treatment groups refer to the types of considerations or treatments given to research subjects, p states the number of dimensions or data variables observed, $\hat{\rho}_i^2$ indicates that the value is estimated from the sample, rather than the actual value.

H_0 : $\hat{p} = 0$ (there is no correlation between *lags* in the distribution)
H_1 : there is at least one i, where $\hat{p}_i \neq 0$ (there is a correlation between lags in the distribution).

If Q > $\chi^2_{(\alpha,\, K\text{-}1)}$, then H_0 is rejected, which means the *errors* have no serial correlation. Therefore, the model is suitable for forecasting.

2.4 Long-Short Term Memory

Long-Short Term Memory (LSTM) is an evolution of RNN. LSTM, introduced by Hochreiter and Schamidhuber in 1997, arose because of the missing gradient in RNN [13]. The main feature of the LSTM network is the hidden layer which consists of memory cells. Each memory cell has three gates: the Front Gate, the input gate, and the output gate [14]. RNN can process *time series* data, but the inability to accommodate a long memory results in RNN will experience a situation where the value for updating the weights will disappear (vanishing gradient). Therefore, RNN can only process *time series* data with short dependency values [15]. This complicated gate combination is called LSTM; the gates of the LSTM cell can be seen in Figure 1.

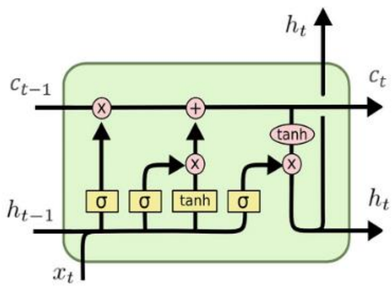

Figure 1. Long Short-Term Memory Architecture

Forget Gate is a gate that decides whether the input and output will be forwarded to the cell state.

$$f_t = \sigma\left(W_f.\lfloor h_{t-1}.y_t\rfloor + b_f\right)$$

Input Gate is an input gate with two activation functions (sigmoid and tanh) to select the part to be updated.

$$i_t = \sigma(W_i.\lfloor h_{t-1}.y_t \rfloor + bi)$$

$$\bar{C}_t = tanh(W_c.\lfloor h_{t-1}.y_t \rfloor + bc)$$

i_t refers to the input gate at timestep t that controls the flow of information into the cell state. The value of $i_t$is calculated using the sigmoid activation function (σ) on the product of the input gate weight matrix (W_i) and the product of the previous hidden state (h_{t-1}) and the current input (y_t), plus the input gate bias (bi). Meanwhile, $\bar{C}_t$ is the candidate for updating the cell state, which is calculated using the hyperbolic tangent activation function on the product of the cell state weight matrix (W_c) and the product of the hidden state and previous input, plus the cell state bias (bc)

Cell State Gate will update the old value C_{t-1} to the new C_t.

$$C_t = (f_t.C_{t-1} + I_t.\bar{C}_t)$$

C_t is calculated from the result of the summation operation of two components, namely the result of the multiplication between the forget gate f_t and the previous cellular state C_{t-1} and the result of the multiplication between the input gate I_t and the candidate updated cellular state $I_t.\bar{C}_t$. The forget gate f_t controls how much information in the previous cellular state will be ignored, while the input gate I_t controls how much new information will be added to the cellular state

The first Output Gate is the Gate that combines the old value and the new value, i.e.

$$\mathrm{o_t} = \sigma(\mathrm{W_0}.\lfloor \mathrm{h_{t-1}}.\mathrm{y_t} \rfloor + \mathrm{b_0})$$

$\mathrm{o_t}$ is the output gate value at time step t. The sigmoid function (σ) is applied to the sum of the weighted inputs, which consists of the previous hidden state $\mathrm{h_{t-1}}.\mathrm{y_t}$ and the current input $\mathrm{y_t}$. $\mathrm{W_0}$ represents the weights for these inputs. Additionally, $\mathrm{b_0}$ represents the bias term added to the weighted sum. The output gate $\mathrm{o_t}$ determines how much information is from the current input.

The second output Gate consists of one cell, which is

$$h_t = o_t.* \tanh\ (C\ t\ )$$

h_t represents the current hidden state at time step t. The element-wise multiplication operator ($*$) is applied to the output gate value (o_t) and the hyperbolic tangent function (tanh) of the cell state (C_t). The cell state (C_t) represents the memory of the LSTM network at time step t. The output gate (o_t) controls how much of the cell state will be used to update the hidden state

LSTM cells can connect previous information with subsequent information, and this effectiveness in storing extended information is indispensable in processing time series data.

3 RESULTS AND DISCUSSION

3.1 Descriptive Analysis

The daily plot of bitcoin closing price from the period of September 1st, 2017, to October 6th, 2022, obtained 1851 data observations which are visualized in the following graph:

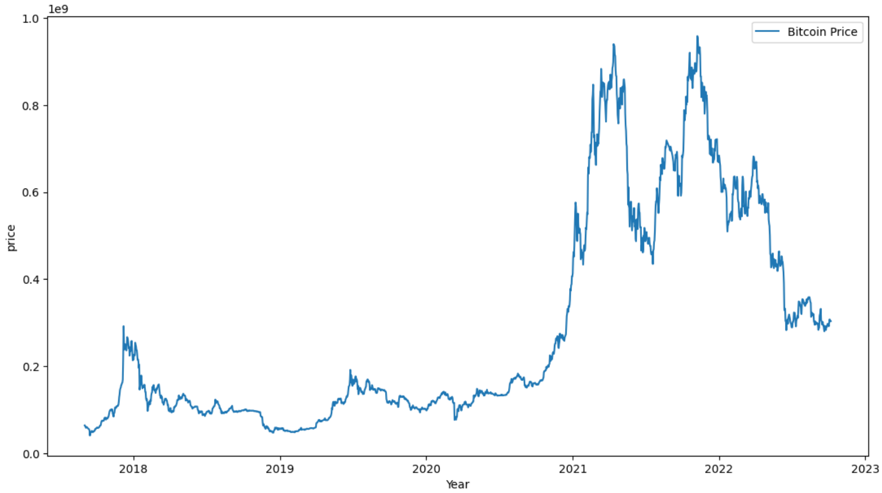

Figure 2. Bitcoin Price Movement

Figure 2 shows the high fluctuation of bitcoin price movement. These changes are due to several external factors, such as the economy, the decline in investors, and market activity. It can be confirmed that the time series data is not stationary because the movement of stock prices from September 1st, 2017, to October 6th, 2022, has increased and decreased every month. Therefore, the data is not around a constant average value. Furthermore, the closing price is analyzed again to get the data results from the *return*. Price changes are often the main factor in the price of bitcoin, so changes in the price of bitcoin can be seen in **Figure 3**.

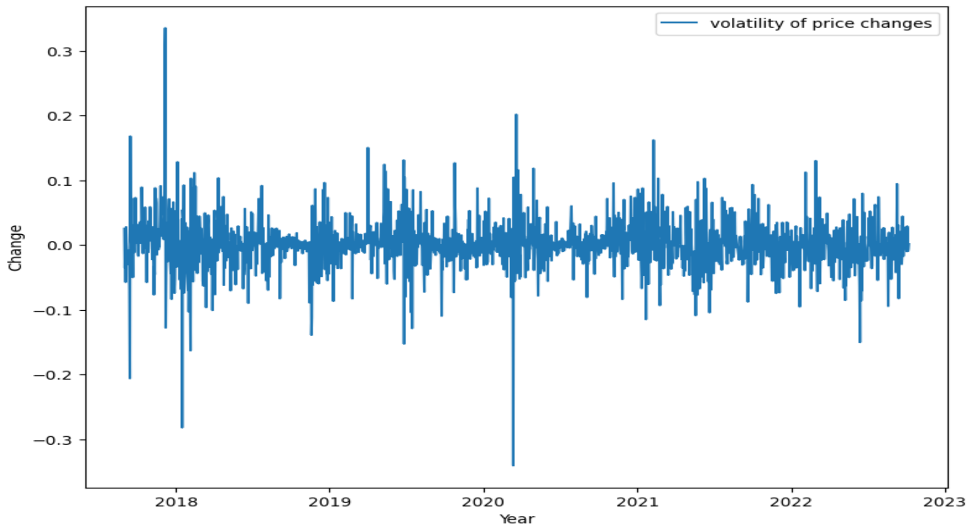

Figure 3. The Volatility of Bitcoin Price Change

Figure 3 plots the daily residual return on the Bitcoin closing price *time series*. The presence of significant changes followed by small changes in consecutive days indicates the presence of *conditional heteroscedasticity* characteristics in the *return series* variance process. This *time series* data uses the GARCH modeling specification to model the *return* volatility in the Bitcoin price *time series*.

3.3.1 Stationary Test

Stationary *time series* analysis requires constant mean and variance. The assumption test that can be done to see the average value is to use the *Augmented Dicky Fuller* (ADF) test while testing the variance using the Box-Cox test. The test results on the average and variance of bitcoin price changes can be seen in **Table 1**.

Table 1. Augmented Dickey-Fuller (ADF) test

	ADF test
Data	p-value
bitcoin price changes	0,000

The stationarity check on the average change in bitcoin price is done with the ADF test. The test results can be seen in **Table 1,** which shows the p-value of 0,000 is smaller than the critical point of 0,05. It can be concluded that the daily bitcoin price change data is stationary to the mean.

3.3.2 ARCH Effect Check

Before analyzing the formation of the GARCH model, it was necessary to check the existence of the ARCH effect using the Lagrange multiplier test. In contrast, the Ljung-box test was used to see the freedom on the variance of the residuals. The results of the ARCH effect check can be seen in **Table 2.**

Table 2. ARCH Model Residual Test

Data	***p-value***	
	Ljung-box	**Lagrange multiplier**
bitcoin price changes	1,32288x 10^{-2}	0,030071

The Ljung-box test used to look at the freedom of variance of the residuals shows a *p-value* of 1,32288 x 10^{-2}, more minor than the critical value $\alpha = 0{,}05$. This decision leads to rejecting H_0, which means there is a correlation between the observations of the residual values of the variances.

The influence of the ARCH effect was tested using the Lagrange multiplier test. The analysis results obtained mean a *p-value* of 0,030071 is smaller than the critical value $\alpha = 0{,}05$. It indicates the decision to reject H_0, which means that there is an ARCH effect on the residual variance. Therefore, it can be continued to the following analysis for GARCH modeling.

3.2 GARCH Modeling

The Generalized Autoregressive Conditional Heteroscedasticity parameter significance test estimation process used *Maximum Log-Likelihood* estimation [16]. *The* estimation process requires the *error* distribution to be normally distributed with constant variance. For this reason, to determine the existence of ARCH/GARCH effects, it was done by modeling the *return* data in the mean and variance functions. Parameter estimation of GARCH was done with *Maximum Log-Likelihood estimation*. GARCH parameter estimation with *return* data can be seen in **Table 3**.

Table 3. Significance Test of GARCH Parameter Estimation

No.	**Model**	**Parameter**	**Parameter Estimation**	***P-Value***
1	**GARCH(1,1)**	**α1**	**0,7493**	**4,56E-03**
		β1	**0,8141**	**1,98E-21**
2	GARCH (2,1)	α1	0,1110	1,08E-02
		α2	0,0000	1,000
		β1	0,8457	3,41E-10
3	GARCH (1,2)	α1	0,1000	1,93E-01
		β1	0,4000	2,20E-02
		β2	0,4000	7,80E-02
4	GARCH (1,2)	α1	0,1310	0,423
		α2	3,02E-12	1,000

	β1	0,5303	0,814
	β2	0,2812	0,867

Based on **Table 3** above, the significance value of the parameter estimation can be seen in the *p-value*. If the p-value is smaller than the significant level (0,05), it can be concluded that the parameter estimation is accepted. However, if the *p-value is* greater than the significant level (0,05), it can be concluded that the parameter estimation cannot be accepted. From the GARCH (1,1) model experiment, it has a parameter estimation with a value smaller than 0,05; it can be concluded that the GARCH (1,1) model has an acceptable parameter estimation.

3.3 Analysis of Hybrid LSTM-GARCH Model

Researchers used the bitcoin closing price variable from each experiment by combining the Long-Short Term Memory method with Generalized Autoregressive Conditional Heteroscedasticity in Hybrid analysis.

3.3.1 Hybrid Model Prediction Results

Parameters are an influential factor in getting higher performance when training LSTM models. The experimental model in the study included neurons and epochs. Determining the number of neurons in the hidden layer and epoch is essential in Long-Short Term Memory. A neuron is an operation process on input data that is performed repeatedly. Each layer of neurons can only perform one operation. However, groups of neurons can perform up to four different operations on the input data in LSTM. The results were then combined sequentially. The determination of the number of neurons depended on the researcher in this study. The number of neurons used was 3, 10, 20, and 30. Starting from a small number like 3 provides a simple baseline model to compare performances against more complex models. This helps identify any potential for improved prediction with increased neurons.

Epoch is the number of iterations used in the study. One epoch means that an algorithm processes all training data at one time. The more the number of *epochs* increases, the more the weighting in the neural network changes. The number of epochs cannot be determined precisely. However, it is adjusted to the number of datasets owned. This research used the number of epochs 50, 75, and 100. It can be seen from the smallest loss value to determine whether the number of epochs is appropriate.

Researchers used Adam Optimizer to process and operate the LSTM model on data samples. The use of parameters in this optimizer used learning rate, beta1, and beta 2 **[17]**. The learning rate is a parameter to calculate the accuracy of the weight value during the training process. After going through the training process to see the results of evaluating bitcoin price predictions using the GARCH-LSTM hybrid model and using Adam's optimizer, the accuracy results are obtained using Mean Absolute Percentage Error. The results of several experiments conducted can be seen in **Table 4**.

Table 4. Neuron and *Epoch* Experiment Results

Neuron	*Epoch*	MAPE
3	50	18,14%
10		17,23%
20		21,93%
30		23,47%
3	75	17,07%
10		18,68%
20		19,17%
30		20,15%
3	**100**	**15,69%**
10		18,21%
20		19,40%

30	25,65%

From the *accuracy* MAPE results, the slightest *error* value is 15,69%, with three neurons and 100 *epochs* parameters. In the research experiment, the most *significant error value* was obtained in parameters with 30 neurons and 100 epochs with an *accuracy* MAPE value of 25,65%. From the research experiment results, using parameters 3 neurons and 100 epochs is classified in the excellent category in producing predictive values due to the low *error value.*

3.4 Bitcoin Price Prediction

After undergoing the LSTM process to predict the hybrid model using 80% *training* data and 20% *testing* data as well as using 3 neurons and 100 epochs, prediction results were obtained compared between actual price data and Bitcoin price prediction data shown in **Table 5**.

Table 5. Bitcoin Price Prediction Results

No.	Date	Actual	Prediction
1	01/09/2017	63.999.000	94.023.404
2	02/09/2017	61.749.600	91.437.024
3	03/09/2017	61.791.800	90.900.858
⋮	⋮	⋮	⋮
1848	04/10/2022	304.748.000	328.325.785
1849	05/10/2022	302.888.992	325.156.067
1850	06/10/2022	303.392.992	323.004.636

Table 5 shows the prediction results and the actual Bitcoin closing price data; It can be seen that the actual data and predicted data have a small error. The comparison plot between actual data and Bitcoin price predictions can be seen in **Figure 4.**

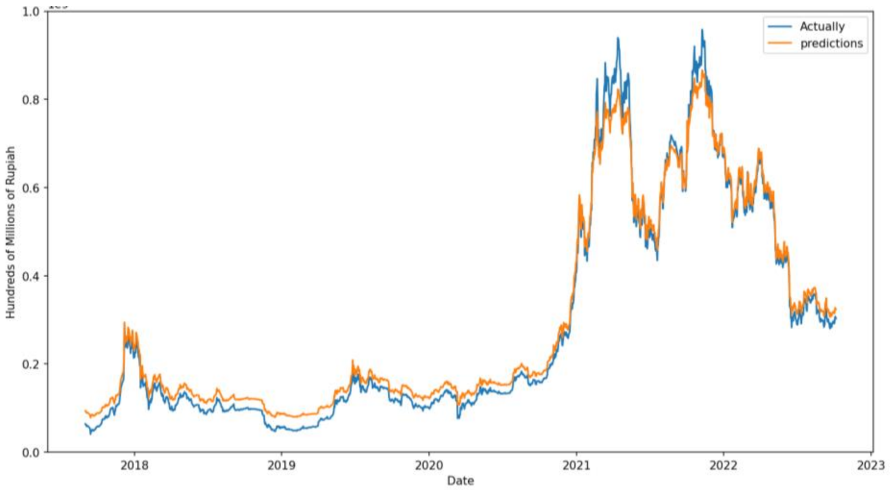

Figure 4. Comparison Graph of Actual Data and Bitcoin Price Prediction

Figure 4 above is a prediction graph of the *hybrid* model results. Part of the graph consists of the actual line (Bitcoin closing price), denoted in blue, and the predicted line (prediction result), denoted in orange. The graph shows good accuracy, marked by the difference between the actual line with a distance or error not too far from the prediction line with a MAPE value of 15,69%.

4. CONCLUSIONS

The results concluded that the prediction of the closing price of bitcoin tends to increase with a predicted value of 323.004.636 against the actual value at the previous price of 303.392.992 The GARCH (1,1) integration model was formed, and the *Long Short Term Memory Model* using 3 neurons and 100 *epochs* produced a *hybrid* model *accuracy* level integrating Generalized Autoregressive Conditional Heteroscedasticity with Long Short Term Memory in the good category using MAPE value of 15,69%. It can be used as a consideration for investors in buying or selling *cryptocurrency* coins against volatility.

This study proves that combining the two models by integrating GARCH into the LSTM model can produce an accuracy level in the good category. It can be used as a reference for further research to integrate other volatility models, such as asymmetric GARCH (EGARCH, GJR-GARCH).

ACKNOWLEDGMENT

The researcher would like to thank all those who helped complete this research journal.

REFERENCES

[1] A. Agrawal and K. Hockerts, "Impact investing: review and research agenda," *J. Small Bus. Entrep.*, vol. 33, no. 2, pp. 153–181, 2021, doi: 10.1080/08276331.2018.1551457.

[2] A. Faturahman, V. Agarwal, and C. Lukita, "Blockchain Technology - The Use Of Cryptocurrencies In Digital Revolution," *IAIC Trans. Sustain. Digit. Innov.*, vol. 3, no. 1, pp. 53–59, 2021, doi: 10.34306/itsdi.v3i1.523.

[3] K. H. Al-Yahyaee, W. Mensi, H. U. Ko, S. M. Yoon, and S. H. Kang, "Why cryptocurrency markets are inefficient: The impact of liquidity and volatility," *North Am. J. Econ. Financ.*, vol. 52, no. February, p. 101168, 2020, doi: 10.1016/j.najef.2020.101168.

[4] R. Bhowmik and S. Wang, "Stock market volatility and return analysis: A systematic literature review," *Entropy*, vol. 22, no. 5, pp. 1–18, 2020, doi: 10.3390/E22050522.

[5] A. P. Raneo and F. Muthia, "Penerapan Model GARCH Dalam Peramalan Volatilitas di Bursa Efek Indonesia," *J. Manaj. Dan Bisnis Sriwij.*, vol. 16, no. 3, pp. 194–202, 2019, doi: 10.29259/jmbs.v16i3.7462.

[6] A. Auti, D. Patil, O. Zagade, P. Bhosale, and P. Ahire, "Bitcoin Price Prediction Using Svm," *Int. J. Eng. Appl. Sci. Technol.*, vol. 6, no. 11, pp. 226–229, 2022, doi: 10.33564/ijeast.2022.v06i11.042.

[7] H. Young and C. Hyun, "Forecasting the volatility of stock price index : A hybrid model integrating LSTM with multiple GARCH-type models," *Expert Syst. Appl.*, vol. 103, pp. 25–37, 2018, doi: 10.1016/j.eswa.2018.03.002.

[8] S. N. Alawiyah, "Pemodelan Menggunakan Pendekatan Recurrent Neural Network Long Short Term Memory (Rnnlstm) Pada Harga Emas Dunia," 2021, [Online]. Available: http://repository.unimus.ac.id/id/eprint/4855

[9] M. Al Haris, "Peramalan Harga Emas Dengan Model Generalized Autoregressive Conditional Heteroscedasticity (Garch)," *J. Saintika Unpam J. Sains dan Mat. Unpam*, vol. 3, no. 1, p. 19, 2020, doi: 10.32493/jsmu.v3i1.5263.

[10] K. Worden, I. Iakovidis, and E. J. Cross, "On stationarity and the interpretation of the ADF statistic," *Conf. Proc. Soc. Exp. Mech. Ser.*, vol. 2, pp. 29–38, 2019, doi: 10.1007/978-3-319-74421-6_5.

[11] W. Rahayu and B. Sumargo, "Forecasting the Number of Foreign Tourist Using Intervention and ARCH Analysis," *AIP Conf. Proc.*, vol. 2588, no. October 2018, 2023, doi: 10.1063/5.0113461.

[12] H. Hassani and M. R. Yeganegi, "Sum of squared ACF and the Ljung–Box statistics," *Phys. A Stat. Mech. its Appl.*, vol. 520, pp. 81–86, 2019, doi: 10.1016/j.physa.2018.12.028.

[13] G. Van Houdt, C. Mosquera, and G. Nápoles, "A review on the long short-term memory model," *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5929–5955, 2020, doi: 10.1007/s10462-020-09838-1.

[14] H. Salman, J. Grover, and T. Shankar, "Hierarchical Reinforcement Learning for Sequencing Behaviors," vol. 2733, no. March, pp. 2709–2733, 2018, doi: 10.1162/NECO.

[15] M. W. P. Aldi, Jondri, and A. Aditsania, "Analisis dan Implementasi Long Short Term Memory Neural Network untuk Prediksi Harga Bitcoin," *J. Inform.*, vol. 5, No, no. 2, p. 3548, 2018, [Online]. Available: http://openlibrarypublications.telkomniversity.ac.id

[16] S.-M. Liu and B. W. Brorsen, "Maximum Likelihood Estimation of a Garch-Stable Model," *J. Appl. Econom.*, vol. 10, no. 3, pp. 273–285, Aug. 1995, [Online]. Available: http://www.jstor.org/stable/2285166

[17] S. J. Reddi, S. Kale, and S. Kumar, "ON THE CONVERGENCE OF ADAM AND BEYOND," *Int. Conf. Learn. Represent.*, vol. 28, no. 1, pp. 54–56, 2018, doi: 10.5505/agri.2014.32559.