A comparison between Super Vector Regression, Random Forest Regressor, LSTM, and GRU in Forecasting Bitcoin Price

Rifando Panggabean[1], and Yohana Dewi Lulu Widyasari[2]

E-mail: *[3] rifando21mttk@mahasiswa.pcr.ac.id

Abstract: High bitcoin user volume results in high market volatility, and indicators commonly used in stock and forex transactions have low accuracy in handling bitcoin's highly volatile market. The present study aims to find out the most optimal machine learning algorithm for Bitcoin transactions by examining four algorithms: *Super vector regression(SVR),Random Forest Regressor(RF),Long short-term memory(LSTM),* and *Gated Recurrent Unit (GRU), examined using* four tests, namely *Root Mean Square Error (RMSE), Mean Square Error (MSE)* , *Mean Absolute Error (MAE)* and *R-Squared(R2)*. The test was performed using Bitcoin data between 2014 and 2022. The test result showed that LSTM+GRU algorithm exhibited the highest accuracy, indicated by a R-squared of 94%.

Keywords: Machine Learning, Bitcoin, Cryptocurrency, LSTM, SVR, GRU, RF

1. Introduction

Bitcoin is one of the cryptocurrencies developed by Satoshi Nakamoto in 2009 and is increasingly popular for alternative trading. A trader usually uses various strategies to determine the best time to take buy/sell actions in Cryptocurrency trading, such as [1] Weigted Moving Average and [2] Autoregressive Integrated Moving Average (ARIMA).

Deep learning algorithms have been applied to the financial sector in the last decade. One of the most popular algorithms for predicting Bitcoin price is Recurrent Neural Network (RNN). Long-short-term memory is one of the RNN algorithms capable of addressing issues of vanishing gradient in RNN. [3] A. Dharma Arisandi used LSTM for predicting Bitcoin with an 80% accuracy rate. Another application of LSTM in the financial sector was reported by [4] W. Hastomo, A. S. B. Karno, N. Kalbuana, E. Nisfiani, and L. Etp, in which they added more layers using the GRU algorithm and reported an accuracy improvement for several issuers [5]. In addition to LSTM and GRU, a number of algorithms have also been applied in financial sectors, including [6] Support Vector Regression (SVR) and Random Forest Regressor (RF).

Bitcoin's high volatility causes algorithms usually used in less volatile stock markets to be inapplicable to Bitcoin, resulting in various accuracy rates. Therefore, it is necessary to test several deep learning algorithms to see the most effective algorithm for Bitcoin trading.

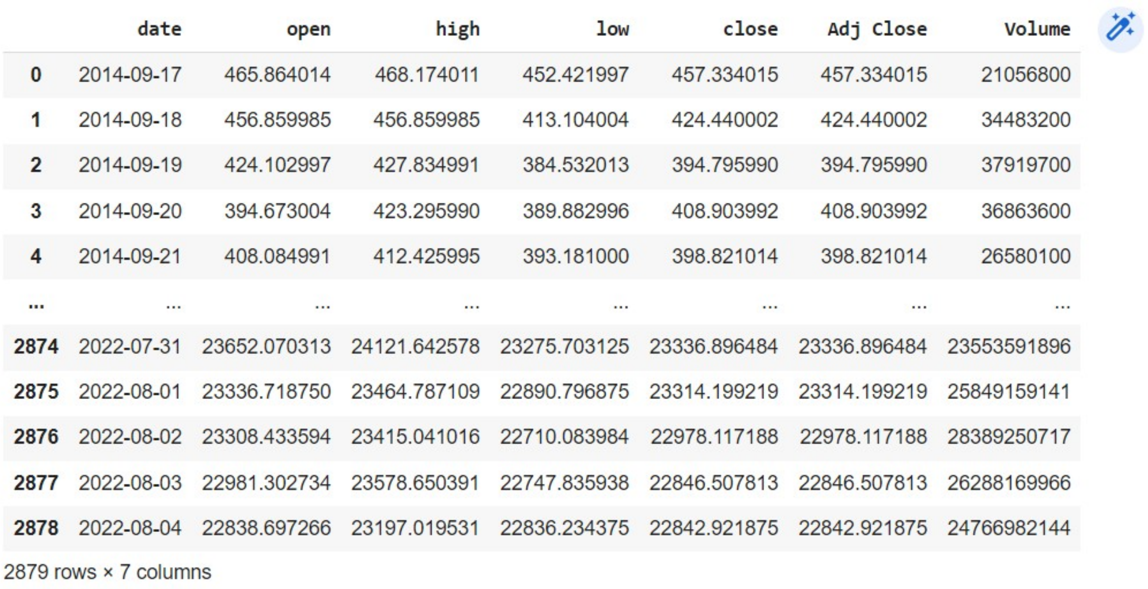
In this study, four deep learning algorithms were tested for bitcoin price forecasting: Super vector regression(SVR), Random Forest Regressor(RF), Long short-term memory(LSTM), and Gated Recurrent Unit (GRU).

Following the test, four measurements were performed, i.e., Root Mean Square Error (RMSE), Mean Square Error (MSE), and Mean Absolute Error (MAE), to examine the margin of error of a machine learning model, which was followed up by R-square (R2) measurement to identify the model's accuracy. This study may contribute to the literature by identifying the most optimal algorithm in bitcoin trading using deep learning.

2. Method

2.1 Dataset

Bitcoin data source was obtained from finance.yahoo.com from 17 September 2014 to 4 August 2022, consisting of 2879 rows and seven columns. The following figure displays the data set.



	date	open	high	low	close	Adj Close	Volume
0	2014-09-17	465.864014	468.174011	452.421997	457.334015	457.334015	21056800
1	2014-09-18	456.859985	456.859985	413.104004	424.440002	424.440002	34483200
2	2014-09-19	424.102997	427.834991	384.532013	394.795990	394.795990	37919700
3	2014-09-20	394.673004	423.295990	389.882996	408.903992	408.903992	36863600
4	2014-09-21	408.084991	412.425995	393.181000	398.821014	398.821014	26580100
...	...	...	...	...	...	...	...
2874	2022-07-31	23652.070313	24121.642578	23275.703125	23336.896484	23336.896484	23553591896
2875	2022-08-01	23336.718750	23464.787109	22890.796875	23314.199219	23314.199219	25849159141
2876	2022-08-02	23308.433594	23415.041016	22710.083984	22978.117188	22978.117188	28389250717
2877	2022-08-03	22981.302734	23578.650391	22747.835938	22846.507813	22846.507813	26288169966
2878	2022-08-04	22838.697266	23197.019531	22836.234375	22842.921875	22842.921875	24766982144

2879 rows × 7 columns

Figure 1. Bitcoin Raw Data

The raw data consisted of six columns presenting High, *Low, Open, Close, Volume, and Adj Close.*

Of these six features, close was selected as the dataset for the next process (Figure 4).

2.2 Training and Testing Dataset

Machine can learn from time series dataset, and the dataset were divided into training and testing dataset with 75%:25% ratio. After that, the model training was performed using each algorithm.

2.2.1 Super Vector Regression (SVR)

The super vector regression model is one of the SVM development for regression purposes. It aims to find function f(x) as the hyperplane (separation line) between the regression function that fits all input data and error and to make the line as thinnest possible [6]. Support Vector Machine method is used in machine learning's classification case, while SVR is applied to regression cases that generate real numbers.

Super Vector Regression algorithm generates a good result in forecasting time series data as it is capable of addressing the overfitting issue — a condition where the model generated by training data exhibited an almost-perfect result. The purpose of SVR is to find the best possible hyperplane, which could be obtained by measuring the hyperplane margin. The SVR formula is presented in the following equation:

$F(x) = \sum i{=}1\ 1\ (\alpha j * - \alpha j)\ (K(xi,x) + \lambda 2\ )$

$\lambda 2$) Description:

Ei = Error value

yi= Normalized data value

$*$ = Lagrange multiplier

αi = Lagrange multiplier

Rij = Hessian matrix

$\delta\alpha i*$ = single variable, not the multiply of δ by $\alpha i*$

$\delta\alpha$i = single variable, not the multiply of δ by αi

γ = Learning rate value

= Epsilon parameter

C= Complexity parameter

The SVR parameter used in this study was as follow:

Kernel = 'rbf'

C = 0.001

Gamm = 0.1

Degree = 3

Epsilon = 0.1

2.2.2 Random Forest (RF)

Random forest is a development of the Classification and Regression Tree (CART) that applies bootstrap aggregation methods and random feature selection. Random forest performance is adopted from the Decision Tree algorithm, with each tree developed from bootstrap samples based on the training data.

The decision tree is a part of Random Forest algorithm, a logical tree that distinguishes data. For instance, in the data of two number 1s and five number 0s, where each number is distinguished in terms of color, the grouped data may form a tree, as presented in Figure 2. Previous studies report that Random forest could deliver a fairly good forecasting result [7].

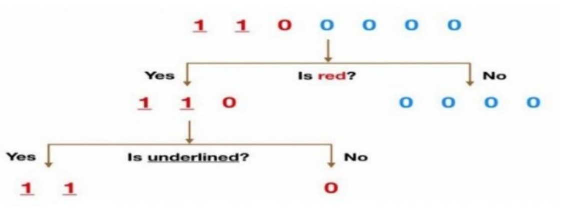


Figure 2. Decision tree procedure

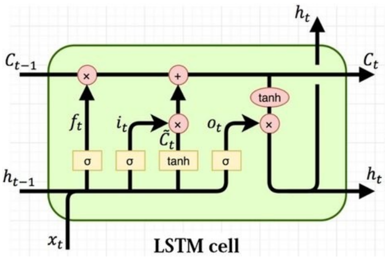
Parameter used in Random Forest in this study was as follows:

n_estimator = 100

random_state =0

2.2.3 Long Short Term Memory (LSTM)

Recurrent Neural Network (RNN) is the initial model of Neural Network Machine Learning. RNN has a limitation in using long information sequences, leading to vanishing gradient problems in the recurrence process. In order to address the problem, the Long Short-Term Memory (LSTM) model is developed. The model is a cell containing sequences consisting four gates and five complex activation functions.



$$i_t = \sigma\left(x_t U^i + h_{t-1} W^i\right)$$
$$f_t = \sigma\left(x_t U^f + h_{t-1} W^f\right)$$
$$o_t = \sigma\left(x_t U^o + h_{t-1} W^o\right)$$
$$\tilde{C}_t = \tanh\left(x_t U^g + h_{t-1} W^g\right)$$
$$C_t = \sigma\left(f_t * C_{t-1} + i_t * \tilde{C}_t\right)$$
$$h_t = \tanh(C_t) * o_t$$

Figure 3. LSTM procedure

The LSTM parameter used in this study was as follow:

loss= ‘mse’

optimizer='adam'

3 optimizer layers with 32 nodes

epoch=10

batch=5

2.2.4 LTSM + GRU (Gated Recurrent Unit)

Although LSTM has a good performance in storing long time series information, its complex sequence results in extensive time required for data processing. In order to address this issue, a more simple model called Gated Recurrent Unit (GRU) is developed. It is an RNN cell comprising two gates and three activation functions. The smaller number of gates expedites the processing of larger data size.

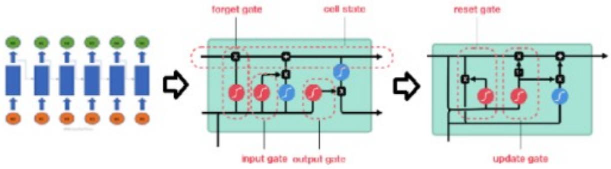


Figure 4. GRU procedure

The LSTM+ GRU parameter used in this study was as follow:

loss= ‘mse’

optimizer='adam'

2 GRU layers with 32 nodes

2 optimizer layers with 32 nodes
epoch=10

batch=5

2.3 Tests

The test was divided into two sections, the first section uses *Root Mean Square Error (RMSE), Mean Square Error (MSE) and Mean absolute Error (MAE)* to test the error of each model, while the second section used R-squared (R2) to assess the model effectiveness.

2.3.1 *Root Mean Square Error (RMSE), Mean Square Error (MSE), and Mean Absolute Error (MAE)*

Root Mean Square Error (RMSE), Mean Square Error (MSE) and Mean absolute Error (MAE) are commonly used algorithms to evaluate machine learning algorithms. The values of RMSE, MSE, and MAE were used to distinguish the model performance within a calibration period from the validation period, and were used to compare the individual model performance to other prediction model [8].

1. RMSE formula

$$RMSE = \sqrt{\frac{\sum_{j=i}^{n}(y'-y)^2}{n}}$$

Description:

= Total data

= Error rate.

y' = Output value (prediction).

y = actual value

2. MSE formula

$$MSE = \sum \frac{(Y' - Y)^2}{n}$$

Description

Y ' = Prediction value

Y = Actual Value

n = Total data

3. MAE formula

$$MAE = \sum \frac{|Y' - Y|}{n}$$

Description:

Y ' = Prediction value

Y = Actual Value

n = Total data

2.3.2 *R-squared (R2)*

R-squared is a number between 0 and 1 indicating the simultaneous effect size of independent variables on the dependent variable. A value closer to 1 represent a better model. R-squared formula:

$$\text{Predicted R}^2 = [1 - (\frac{PRESS}{SST})] x\ 100$$

3. Result and Discussion

The following chart presents the dataset training result using 4 algorithms and 4 test methods.

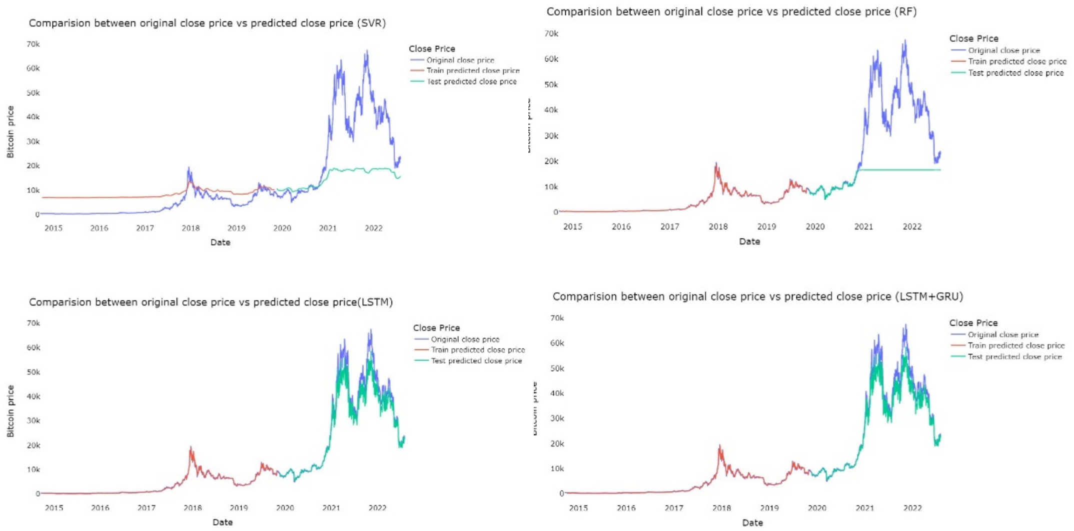
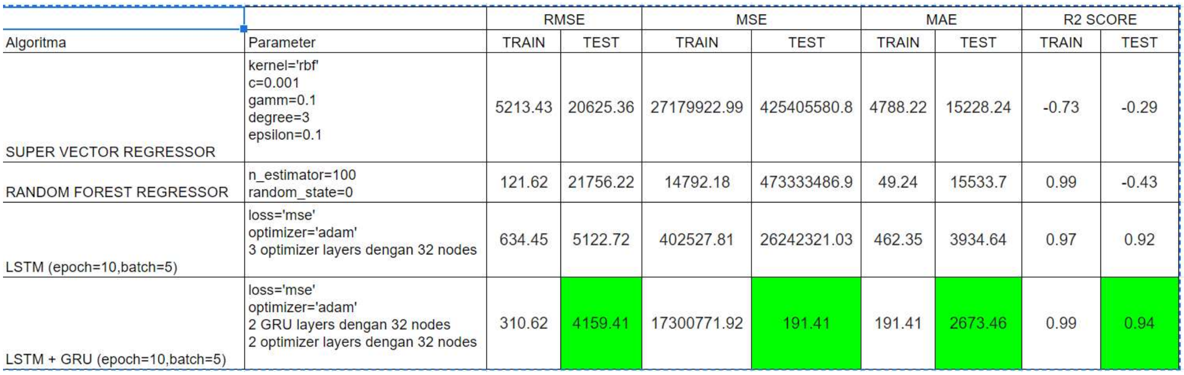


Figure 4. Model plotting result and the original chart.

As displayed in figure 4, LSTM and LSTM+GRU exhibited higher accuracy than RF and SVR. The following table presents the test result using RMSE, MSE, MAE, and R2.

Table 1. Test Result

BITCOIN FORECASTING RESULT (BASED ON CLOSE PRICE)



		RMSE		MSE		MAE		R2 SCORE	
Algoritma	Parameter	TRAIN	TEST	TRAIN	TEST	TRAIN	TEST	TRAIN	TEST
SUPER VECTOR REGRESSOR	kernel='rbf' c=0.001 gamm=0.1 degree=3 epsilon=0.1	5213.43	20625.36	27179922.99	425405580.8	4788.22	15228.24	-0.73	-0.29
RANDOM FOREST REGRESSOR	n_estimator=100 random_state=0	121.62	21756.22	14792.18	473333486.9	49.24	15533.7	0.99	-0.43
LSTM (epoch=10,batch=5)	loss='mse' optimizer='adam' 3 optimizer layers dengan 32 nodes	634.45	5122.72	402527.81	26242321.03	462.35	3934.64	0.97	0.92
LSTM + GRU (epoch=10,batch=5)	loss='mse' optimizer='adam' 2 GRU layers dengan 32 nodes 2 optimizer layers dengan 32 nodes	310.62	4159.41	17300771.92	191.41	191.41	2673.46	0.99	0.94

The table above shows that LSTM+GRU was superior in almost all test methods with R-square of 94%, higher than LSTM (r-square of 92%).

4. Conclusion and Recommendation

The test result demonstrates that LSTM+GRU exhibited better results than other three algorithms. It indicates that LSTM+GRU algorithm model could be used as a reference for bitcoin transaction. The LSTM-GRU algorithm in this model also has a higher accuracy than the previous research of 94%, where the previous study had a model accuracy of 80%.

Development:

1. It is recommended to test the model to other cryptocurrencies.

2. It is recommended to use different epoch-layer combination to improve the accuracy.

References

[1] N. A. Bakar and S. Rosbi, “Weighted Moving Average of Forecasting Method for Predicting Bitcoin Share Price using High Frequency Data: A Statistical Method in Financial Cryptocurrency Technology,” *International Journal of Advanced Engineering Research and Science*, vol. 5, no. 1, pp. 64–69, 2018, doi: 10.22161/ijaers.5.1.11

[2] N. Salwa, N. Tatsara, R. Amalia, and A. F. Zohra, “Peramalan Harga Bitcoin Menggunakan Metode ARIMA (Autoregressive Integrated Moving Average),” *Journal of Data Analysis*, vol. 1, no. 1, pp. 21–31, Sep. 2018, doi: 10.24815/jda.v1i1.11874.

[3] A. Dharma Arisandi, “Prediksi Mata Uang Bitcoin Menggunakan LSTM Dan Sentimen Analisis Pada Sosial Media,” *Jurnal Ilmiah Komputasi*, vol. 19, no. 4, Dec. 2020, doi: 10.32409/jikstik.19.4.370.

[4] W. Hastomo, A. S. B. Karno, N. Kalbuana, E. Nisfiani, and L. Etp, “Optimasi Deep Learning untuk Prediksi Saham di Masa Pandemi Covid-19,” Jurnal Edukasi dan Penelitian Informatika (JEPIN), vol. 7, no. 2, p. 133, Aug. 2021, doi: 10.26418/jp.v7i2.47411.

[5] T. Rustendi, “Hubungan Kinerja Keuangan Dengan Tanggung jawab Sosial Perusahaan (Studi pada Emiten Manufaktur Di Bursa Efek Indonesia),” Center for Open Science, May 2019. Accessed: Jul. 27, 2022. [Online]. Available: http://dx.doi.org/10.31227/osf.io/ntcz6

[6] “Support Vector Machines,” in *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press, 2000, pp. 93–124. Accessed: Aug. 04, 2022. [Online]. Available: http://dx.doi.org/10.1017/cbo9780511801389.008

[7] H. Supriyanto, “Perbandingan Metode Supervised Learning Untuk Peramalan Time Series Pada Kunjungan Pasien Rawat Jalan,” *Perbandingan Metode Supervised Learning Untuk Peramalan Time Series Pada Kunjungan Pasien Rawat Jalan*, vol. Vol. 10, no. 2088–2130, Jun. 2022.

[8] M. Hosseini, A. Javaherian, and B. Movahed, “Determination of permeability index using Stoneley slowness analysis, NMR models, and formation evaluations: a case study from a gas reservoir, south of Iran,” *Journal of Applied Geophysics*, vol. 109, pp. 80–87, Oct. 2014, doi: 10.1016/j.jappgeo.2014.07.016.

[9] L. An and B. Bhanu, “Improved image super-resolution by Support Vector Regression,” Jul. 2011. Accessed: Jul. 27, 2022. [Online]. Available: http://dx.doi.org/10.1109/ijcnn.2011.6033289