Jurnal Mantik
Volume 4, Number 2, Aug 2020, pp. 1090-1095
E-ISSN 2685-4236
https://iocscience.org/ejournal/index.php/mantik

# Bitcoin Price Prediction Using Long Short Term Memory (LSTM)

**Ihyak Ulumuddin [1]*, Sunardi[2], Abdul Fadlil[3]**

[1]Program Studi Magister Teknik Informatika,
[2,3]Electrical Engineering Study Program
[1,2,3]Universitas Ahmad Dahlan, Yogyakarta, Jl. Kapas No.9, Semaki, Kec. Umbulharjo, Kota Yogyakarta, Daerah Istimewa Yogyakarta 55166

E-mail: ihyak.data@gmail.com*

ARTICLE INFO

Article history:
Received: 10-7-2020
Revised: 29 -7- 2020
Accepted: 01-8-2020

**Keywords**:
Visitor Satisfaction,
Library,
Data Mining,
C4.5

ABSTRACT

Price fluctuation is a necessity in every business, including the price of Bitcoin. Therefore humans need a device or application that can assist in making predictions quickly and accurately. Deep Learning can be used to make predictions which include setting the parameters used during training, choosing the best parameters for prediction, and choosing the prediction results with the smallest error level in the actual situation. This study performs a Bitcoin price prediction using Long Short Term Memory (LSTM). This study uses quantitative calculations on the prediction results. Measurement of accuracy is done by testing the previous price (back testing) and calculating the average error using RMSE and MAPE. The method for generating predictions is LSTM as a type of Recurrent Neural Network (RNN) which has the advantage of using long-term memory in handling time series data. LSTM implementation using Python is used for price forecasting with stages starting from data collection and normalization, input and output modeling. The result of this research is a prediction of Bitcoin price with an accuracy rate of 97.48% based on the best model with input layer 2, the number of epochs 100, the number of hidden layers 100, and activation using Softmax. These price predictions can be used as a reference and consideration for traders to create trading strategies and run them automatically on the digital currency market.

Copyright © 2020 Jurnal Mantik.
All rights reserved.

## 1. Introduction

Bitcoin is a peer-to-peer electronic cash system (Nakamoto, 2008). In his article entitled Bitcoin: A Peer-to-Peer Electronic Cash System explains that Bitcoin is a solution for transactions without third parties. Transactions via the Bitcoin network can be carried out directly by one party to another without involving financial institutions / agencies so that double-spending does not occur. Like the official currency of a country that is transacted and exchanged, Bitcoin can also be used as a means of payment transactions. Some businesses in cities around the world accept Bitcoin payments. According to Blystone (2020) there are at least 10 leading cities since 2020 that accept Bitcoin payments and even cities provide Automatic Teller Machines (ATMs) for digital currencies.

Deep learning is an important part of various kinds of machine learning using Artificial Neural Networks (ANN). Types of learning in deep learning can be supervised, semi-supervised, and unsupervised. In this study, deep learning with Long Sort Term Memory (LSTM) was implemented to predict the price of Bitcoin. Bitcoin can also be traded on the digital currency market (crypto exchange) like stocks. In Indonesia, buying and selling (trading) Bitcoin with IDR currency can be done through the INDODAX market, based on data released on the https://coinmarketcap.com/ page INDODAX is currently in the 36th position of the Top Cryptocurrency Spot Exchanges in the world.

One of the biggest challenges for Bitcoin traders is predicting prices quickly and accurately for profit. Bitcoin traders often make buying or selling decisions based on the flow of information (trending topics). Bitcoin is believed to be the currency of the future so it is predicted that the price will continue to rise (bullish). The price of Bitcoin in December 2018 fell to IDR 46 million from the previous price of IDR 280 million. If the flow of information describes a lot of Bitcoin theft or the currency is considered an illegal asset, it is predicted that the price will fall (bearish) and immediately take the decision to sell. But the facts on the ground



Accredited "Rank 4"(Sinta 4), DIKTI, No. 36/E/KPT/2019, December 13th 2019.

Jurnal Mantik is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

are not like that, when the Nice Hash of the Bitcoin mining center and the Binance market was hacked losing some Bitcoin it turned out that the price continued to increase. That is the reason for the need to make an accurate and fast Bitcoin price prediction method. Accurate predictions certainly require complex and comprehensive calculations so that it is very difficult to do them manually. Digital currency trading can be done manually but it is inefficient and prone to errors due to human factors such as psychological conditions, emotions, and greed

## 2. Research Method

This study uses quantitative calculation techniques on the results of predictions, accuracy measurements are carried out by testing the previous price (Back Testing) and calculating the average eroor using RMSE and MAPE. The stages of this research are as follows:

### A. Dataset Collection

Data collection in this study uses Bitcoin transaction data from the website https://www.kaggle.com/khalilbrick/bitcoin-data-from-2014-to-2020 trade rate data 1 day interval in USD from 17 September 2014 to 7 July 2020.

### B. Data Pre-Processing

At this stage, cleaning, normalizing, indexing, classifying, determining the length of the data are used as input to the neural network. The process of reading data uses the Pandas Library which is available in the Python 3.8.7 package / library. This stage performs the process of scaling the data in such a way that the entire data is in the same range.

### C. Input Layer

The input model is defined as a layer sequence, namely a sequential model by adding layers one by one to the last layer. The first thing to do is to make sure the input layer has the right number of inputs, while the output quantity is a price prediction one day after that.

### D. Training and Testing

After the input model is defined, the next step is to determine the training parameters. The model is made sequentially using the Keras library available in the Python 3.8.7 package / library with several parameters of LSTM units, number of outputs, and activation between units.

## 3. Result and Discussion

### A. Dataset Collection

**Table 1.**
Summary of BTC Dataset

| | Open | High | Low | Close |
|---|---|---|---|---|
| **count** | 2.123,0 | 2.123,0 | 2.123,0 | 2.123,0 |
| **mean** | 4.185,9 | 4.294,7 | 4.069,4 | 4.190,0 |
| **std** | 4.029,8 | 4.152,4 | 3.886,7 | 4.030,5 |
| **min** | 176,9 | 211,7 | 171,5 | 178,1 |
| **25%** | 425,3 | 432,4 | 420,6 | 424,7 |
| **50%** | 3.341,8 | 3.453,4 | 3.247,7 | 3.378,9 |
| **75%** | 7.509,3 | 7.696,6 | 7.374,9 | 7.531,8 |
| **max** | 19.475,8 | 20.089,0 | 18.974,1 | 19.497,4 |

This study uses a dataset of Bitcoin prices in the range 17 September 2014 to 7 July 2020 (Brick, 2020). The dataset is 2,123 records with the column arrangement of Date, Open, High, Low, Close, and Volume. The average price is USD 4,190.0, the lowest is USD 178.1, and the highest is USD 19,497.4 which are summarized in Table 1. Furthermore, the data in the "Close" column of the Bitcoin dataset in Table 1 is used for training needs with the LSTM method so that the dataset can be seen in Figure 6. The horizontal axis is the unit of data amount while the vertical axis is the price of Bitcoin in USD.



**Accredited "Rank 4"(Sinta 4),** DIKTI, No. 36/E/KPT/2019, December 13th 2019.

**Jurnal Mantik** is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

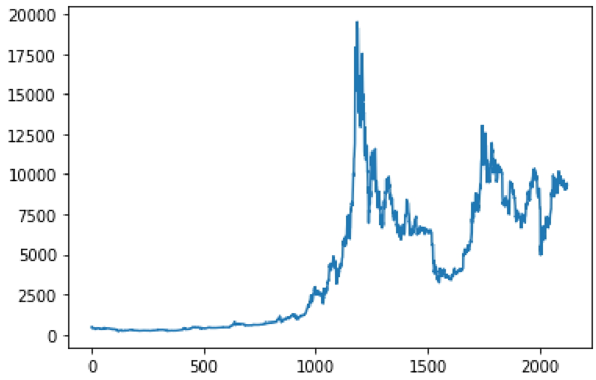


**Fig 1.** BTC Price Movement Chart

B. Data Pre-Processing

The dataset shown in Figure 6 in series format needs to be converted into an array format so that it can be normalized to a scale of 0-1. The pre-pocessing process to get a 0-1 scala using the Scikit-learn library available in the Python package / library with equation (9) where x is the dataset to be scaled (Pane & Rahcmadani, 2020).

$$X_{sc} = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{1.3}$$

X = Dataset to be scaled (sc)

**C. Input Layer**

The normalized dataset is divided by 90% of the training dataset and 10% of the testing dataset. The training data is divided into input (hereinafter referred to as X-train) and output (hereinafter referred to as Y-train) on the LSTM network. The LSTM training process is made with two models. The first model, the X-train and Y-train forms each consist of one price data as can be seen in Figure 2. The second model, the X-train form consists of two price data and the Y-train still consists of one price data as can seen in Figure 3. The process of making the training data uses the Libabry Numpy which is available in the Python 3.8.2 package / library.


**Fig 2.** Model 1

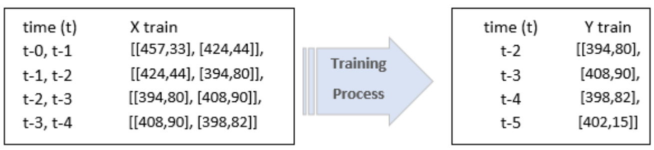


**Fig 3.** Model 2

**D. Training and Testing**

Determining the number of units will affect the number of parameters to be calculated during training. The number of parameters is determined by the following equation:

Number of parameters = 4 * ((input * output) + (output ^ 2) + output)

The more total parameters, the more time it takes to complete the training process. Then several tests were carried out as described below.

This test regulates the amount of data on the input layer used in the LSTM network, namely by using two models as described with the input parameters as in Table 2. The best results using the amount of data in the input layer as in Table 4 is input data 2 with an error value MAPE of 2.55 and RMSE of 331.18.Table 3. Test Results with Epoch


**Jurnal Mantik** is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Table 3.
Test Results with Epoch

| No | Number of Hidden Layer | MAPE | RMSE |
|---|---|---|---|
| 1 | 4 | 2.55 | 331.93 |
| 2 | 8 | 2.68 | 339.05 |
| 3 | 16 | 2.59 | 335.13 |
| 4 | 32 | 2.52 | 329.15 |
| 5 | 64 | 2.57 | 331.28 |

**Table 4.**
testing input layer

| Data Input | MAPE | RMSE | Jumlah Data |
|---|---|---|---|
| 1 | 2,73 | 340,07 | 211 |
| 2 | 2,55 | 331,18 | 210 |

**E. Epoch testing**

The second test was carried out on the LTSM network with a different epoch model. According to Brownlee (2017), one epoch is one time passing all samples in the training data and updating the tissue weights. LSTM can be trained for tens, hundreds, or thousands of epochs, while a batch is one time passing through a part / group of data from the traning dataset, after which the network weights are updated, so that one epoch can consist of one or more batches.

**Table 5.**
Testing Parameters Number of Epoch.

| Parameter | Value |
|---|---|
| X-train | 2 |
| Y-train | 1 |
| Hindden Layer | 50 |
| Activation Function | Softmax |
| Panjang Data Traning | 1.913 |
| Panjang Data Testing | 210 |
| Batch Size | 1 |

**Table 6.**
Number of Epoch Testing Results

| No | Epoch | MAPE | RMSE |
|---|---|---|---|
| 1 | 50 | 2,682 | 339,680 |
| 2 | 100 | 2,595 | 335,300 |
| 3 | 500 | 2,675 | 336,619 |

The test parameters for the number of epochs in this study can be seen in Table 5. The test results of using epoch as in Table 6 show that the number of epochs has no effect on the accuracy of the prediction results. The lowest error occurs when using the number of epochs of 100 with a MAPE value of 2.675%, which is better than the number of epochs of 50 and 500. Using less epochs can reduce the need for time during training.

**F. Hidden layer testing**

This test regulates the number of hidden layer inputs used in the LSTM network. The hidden layer determines the number of lines used in the LSTM network. The parameters and results of hidden layer testing can be seen in Table 7 and Table 8.

**Table 7.**
Testing parameters for the number of hidden layers.

| Parameter | Value |
|---|---|
| *X-train* | 2 |
| *Y-train* | 1 |
| *Activation Function* | Softmax |
| *Panjang Data Traning* | 1.913 |
| *Panjang Data Testing* | 210 |
| *Batch Size* | 1 |
| *Epoch* | 100 |

**Table 8.**
Results of Testing the Number of Hidden Layer.

| Total Hidden Layer | MAPE | RMSE |
|---|---|---|
| 4 | 2,55 | 331,93 |
| 8 | 2,68 | 339,05 |



**Accredited "Rank 4"(Sinta 4),** DIKTI, No. 36/E/KPT/2019, December 13th 2019.

**Jurnal Mantik** is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

| | | |
|---|---|---|
| 16 | 2,59 | 335,13 |
| 32 | 2,52 | 329,15 |
| 64 | 2,57 | 331,28 |

The number of hidden layers in LSTM does not have much effect on prediction error. The best value for using the number of Hidden Layer is 32 with a MAPE error value of 2.52% and an RMSE value of 329.15.

### G. Activation Testing

This test is carried out by entering several LSTM unit activation parameters. Activation consists of Softmax, Relu, Sigmoid, and Tanh. Activation uses the Keras Library available in the Python 3.8.2 package / library with parameters and results as in Table 9 and Table 10.

**Table 9.**
Activation Type Testing Parameters.

| Parameter | Value |
|---|---|
| *X-train* | 2 |
| *Y-train* | 1 |
| *Hidden Layer* | 32 |
| *Panjang Data Traning* | 1.913 |
| *Panjang Data Testing* | 210 |
| *Batch Size* | 1 |
| *Epoch* | 100 |

**Table 10.**
Activation Type Testing Results

| No | Activation | MAPE | RMSE |
|---|---|---|---|
| 1 | Softmax | 2.52 | 329.15 |
| 2 | Relu | 2.62 | 336.27 |
| 3 | Sigmoid | 2.60 | 335.79 |
| 4 | Tanh | 2.58 | 331.29 |

After testing with several activation parameters on the network, the Softmax activation model is used which produces the smallest error, namely MAPE of 2.52% and RMSE of 329.15.

## 4. Conclusion

The price of Bitcoin can be predicted with the deep learning method using Long Short Term Memory (LSTM) with an accuracy rate of 97.48%, MAPE error 2.52% and RMSE 329.15 with several input parameters, namely the number of input layer 2, the number of epochs 100, the number of hidden layer 32, and activation using Softmax. Prediction using deep learning method using LSTM is a regression approach. The stages begin with dataset preparation, normalization, modeling, training, testing, and evaluation of predicted results. Evaluation of model selection is most done by calculating the average error during the testing process.

## 5. References


[1] Bowerman, B. L. & O'Connell, R.T. (1987). Time Series Forecasting. Boston: Duxbury Press.
[2] Dimas A. W. & Oscar D. (2017), Blockchain Dari Bitcoin untuk Dunia. Jakarta: Jasakom.
[3] Ivan V., Daniel S., Gianmario S., Peter R. & Valientino Z. (2019), Pithon Deep Learning. Second Edition. UK: Packt Publishing Ltd.
[4] Jason B. (2016), Deep Learning With Python, Edition: v1.7
[5] Oscar D. dan Sintha R. (2017). Bitcoin Trading For Z Generation Cara Gaul Mengenal dan Trading Bitcoin.Jakarta: Jasakom.
[6] Ian H.Witten.,Eibe Frank.,Mark A.Hall.,Christoper J.Pal. (2017), Data Mining Practical Machine Learning Tools and Techniques. fourth Edition. United State: Morgan Kaufmann.
[7] Jan Wira G.P. (2018), Pengenalan Konsep Pembelajaran Mesin dan Deep Learning., Edisi 12. Jepang
[8] Annalyn Ng.,Kenneth Soo.,(2 017). Numsense ! Data Science for The Layman.,Kindle Edition
[9] Foster P., Tom F., Data Science For Bussines :What you Need to Know about Data Mining and data Analytic Thinking., Kindle Edition
[10] Gareth J., Daniela W., Trevor H., Robert T., (2017), An Introduction to Statistical Learning : with Applications in R., New York:Springer
[11] Michael N., (2020), Artificial Intelligence: A Guide to Intelligent Systems (3rd Edition).
[12] Ritzkal. 2018. Manajemen jaringan untuk pemula. Bogor: UIKA Press
[13] Ritzkal. 2019. Keamanan Jaringan Cyber. Bogor: UIKA Press
[14] Ritzkal. (2020). Tick Waste Application in Houses With Warning of Microcontroller Assistant Social Media. Jurnal MANTIK, Vol. 3 , no. 4, pp. 559-568.




**Accredited "Rank 4"(Sinta 4),** DIKTI, No. 36/E/KPT/2019, December 13th 2019.

**Jurnal Mantik** is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).