Bitcoin Price Prediction at Minute Resolution Based on enhanced LSTM Models

Kaifu Xiao

April 29, 2024

Abstract

The long short-term memory (LSTM) network, recognized as a state-of-the-art model for time series forecasting like stock or cryptocurrency markets, has seen less exploration regarding minute-level prediction of cryptocurrency markets. Three different models—LSTM, stacked LSTM, and CNN-LSTM—were proposed to forecast the closing price of cryptocurrency based on a small amount of previous sequential data. The models were trained on minute bitcoin price data from 2019/4/1 to 2019/5/2. It was found that the proposed CNN-LSTM outperformed the other models under various evaluation indicators such as mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE). The reasons why CNN-LSTM was more effective than others in minute-level prediction were also explained. The study highlights the potential for automatic transaction robots in the Bitcoin market.

1 Introduction

Bitcoin (BTC), a decentralized digital currency, stands as one of the most famous examples of cryptocurrencies. From the history depicted in the bitcoin diagram (Figure 1), it can be observed that there was a steep increase in the price of bitcoin starting from 2018. This fluctuation has attracted people to participate in the digital currency market. On the other hand, the instability and high uncertainty of the cryptocurrency market have interested us in predicting the price at a minute level.

The LSTM model, proposed by Sepp Hochreiter and Jürgen Schmidhuber, is recognized as the state-of-the-art model for time series prediction in neural networks. However, while the LSTM model is effective for daily level price prediction, its performance at the minute level price prediction does not appear promising. It was found that the original LSTM model necessitated numerous trial-and-error processes and parameter tuning during the experiments. The poor performance was attributed to the high fluctuation of minute-level data. Consequently, the use of a stacked LSTM model was proposed, which incorporates multiple LSTM layers and dropout layers to mitigate oscillation and overfitting. Inspired by the model stacking idea of CNN-LSTM, the CNN-LSTM model was also proposed in the hope of capturing critical characteristics from the highly fluctuating data. The approach involved converting the raw dataset into a ready-to-train dataset. After employing grid search to tune the best model parameters and hyperparameters, the performance of different models was compared using comprehensive evaluation indicators.

2 Literature Review

Over the past decade, Bitcoin has been studied on multiple levels, including price formation, volatility, system dynamics, and economic value. A study by D. G. Baur et.al. helped clarify the fundamental and speculative value of Bitcoin from an economic perspective. It provided ample support for the claim that

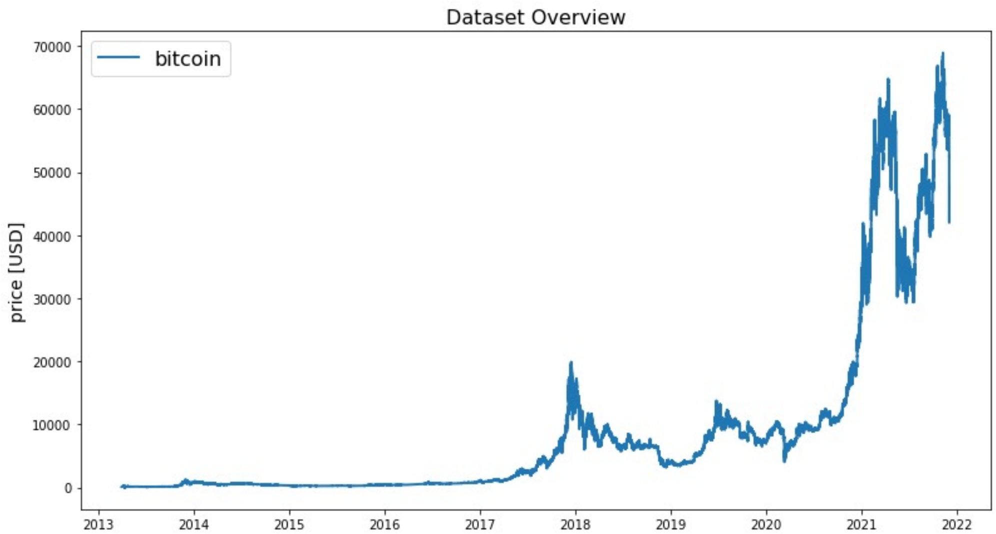

Figure 1: The price of Bitcoin to USD from 2013 to 2021

Bitcoin is a financial asset that is readily susceptible to dynamic market activities. [1] Due to the resemblance of cryptocurrency to other investment properties such as stocks [1], the price of cryptocurrencies is likely to be predicted using similar forecasting methods for the stock market. Traditional analysis methods and machine learning methods are two types of forecasting methods for the noisy stock market.[8] However, traditional analysis involves econometric methods or equations, both of which are not suitable for analyzing series data in a highly dynamic and complicated market system.[6] Therefore, neural networks have come into the play when a powerful and dynamic forecasting method is required. Neural networks are capable of extracting patterns from a large volume of data without requiring any knowledge beforehand.

In 2018, [3] compares the ARIMA Time Series Model and LSTM Deep Learning Algorithm to determine the future price of Bitcoin. The daily prices were estimated with the obtained models, with a result of MAPE 11.86% with ARIMA and MAPE 1.40% with LSTM. Combining other results of the accuracy test, the research supported that the LSTM is a better model for predicting time series prices. However, the price forecasting method for daily price may perform differently from price data in minute level, since the price fluctuates rapidly and may lead to serious overfitting. In response to this issue, [9] has proposed a solution that reduces overfitting by randomly omitting half of the feature detectors in each training.

Different from the monotonous neural network mentioned above, mixture models of LSTM have been proposed in studies by [2] and [4]. The mixture models proposed in [4] are described as having better-defined data flows and architecture to capture the time-varying effect in data compared to methods based on recurrent neural networks. In this paper, the aim is to compare the performance of mixture models with other variants of LSTM (CNN-LSTM, LSTM, stacked-LSTM) and to find a possible data preprocessing method to enhance the performance of mixture or vanilla models when dealing with minute-level time series data.

3 Models

3.1 LSTM model

LSTM was first introduced by Hochreiter & Schmidhuber (1997), which is explicitly designed to avoid the long-term dependency problem based on RNN. RNNs have feedback from the loops in the recurrent layer, which will save the parameters in the memory over time. However, it does not perform well in learning long-term temporal dependencies because of a limited range of contextual information. Also, RNN faces the problem of a vanishing gradient. The Long Short-Term Memory (LSTM) network is a type of recurrent neural network capable of learning order dependence in sequence prediction problems, which is designed to address the vanishing gradient problem. LSTM units have a memory cell that can save the parameters for a long period of time by manipulating 4 gates. The forget gate, input gate, cell gate, output gate. The details of each LSTM cell are illustrated in Figure 2.

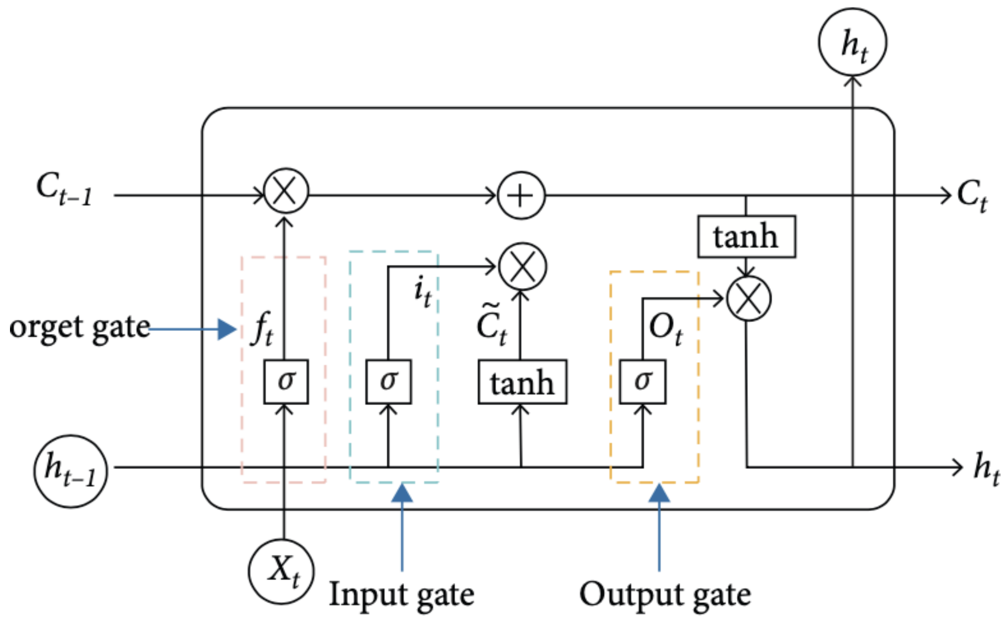

Figure 2: The illustration of a LSTM cell, cited from Lu, W., Li, J., Li, Y., Sun [6]

3.2 Stacked LSTM model

Since the training of the bitcoin prediction model is aimed at learning a hierarchical representation of the time-series data, a deeper hidden layer in the model is desired to improve performance. By vertically stacking the LSTM architecture, each LSTM layer is made to output a sequence of vectors, which is then used as an input to a subsequent LSTM layer. This approach potentially allows the hidden state at each level to operate at different timescales. Between LSTM layers, dropout layers are included to prevent overfitting.

3.3 CNN-LSTM model

The CNN-LSTM [6] was designed for the sequence prediction problem, like the image description, video description, and textual description. The CNN-LSTM includes applying Convolutional Neural layers for feature extraction on input data and then passing the results to the LSTM to perform sequence perdition. Using the CNN network to get the most important features and apply these to improve the performance in the LSTM model. The CNN-LSTM contains the input layer, conv1d, maxpool1d, LSTM hidden layer, and dense layer.

4 Experiment

4.1 Data

The data utilized in this research concern historical data from 1 April 2019 to 2 March 2019 of BTC in USD at the minute level, which constitute the cryptocurrencies with the highest market capitalization. Data for this research were collected from www.kaggle.com. For evaluation purposes, the data set was divided into the training set, validation set, and testing set. The training data set contains the BTC minute level data from 1 April 2019 to 25 April 2019 (26394 data points), the validation data set contains the minute level data from 26 April 2019 to 29 April 2019 (6440 data points), and the test data set contains the minute level data from 30 April 2019 to 2 March 2019 (6440 data points).

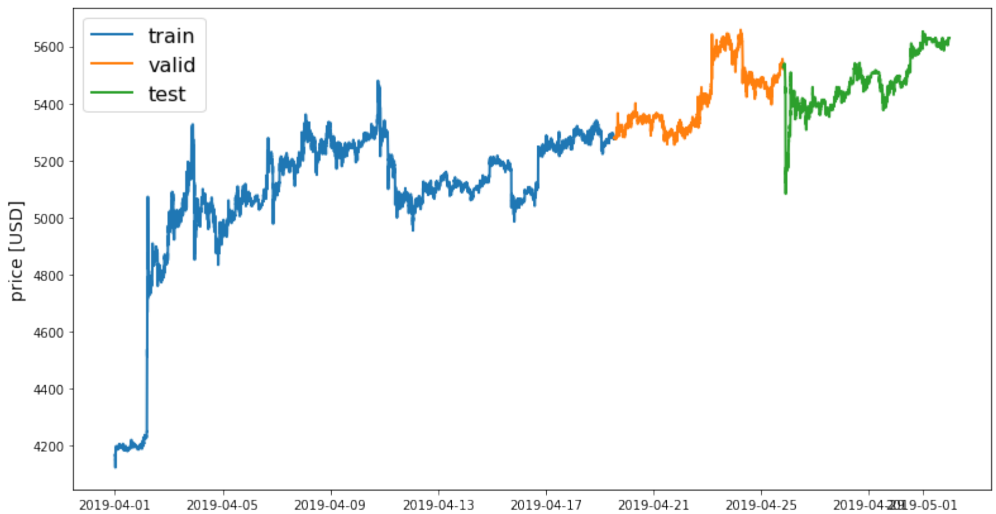

Figure 3: The overview of the train, valid, test data

The results show that the BTC value at the minute level fluctuated significantly during the time period of data collection (Figure 3). These fluctuations were most prominent from 26 April to 28 April 2019. An observation is that the Bitcoin market is in considerable volatility during this period and deviations from the regular behavior from the normal stock market (Figure 3). Though outdated, this period of data is still a great representation of the regularity of uncertainty of the Bitcoin market, which made traditional time series prediction models less likely to catch the essence of the data.

To fully understand the data, we include the results of the t-statistics and the associated p-values of the augmented Dickey-Fuller (ADF) test performed on our data set (Table 1). We selected the statistical significance at a 5% critical level (see Table 1). The interpretation of the Table 1 shows that BTC time-series implies that these series are non-stationary.

Table 1: Augmented Dickey-Fuller (ADF) unit root test of the Bitcoin time-series

Time-Series	t-Statistic	p-Value
Train	-4.00218	0.011820
Valid	-2.05183	0.213406
Test	-2.05202	0.228939

4.2 Data Preprocessing

To prepare the data for training, two main actions were taken on the original dataset. One was the normalization of the dataset, which involved changing the values of numeric columns to a common scale without distorting differences in the ranges of values. The other was the use of the sliding window algorithm, which may be suitable for the prediction of highly nonlinear stock data.

For normalization, zero-based normalization is used in preprocessing. Zero-based normalization reflects changes with respect to the first entry. This method involves, given a vector V with a size of n, constructing the new vector V by the following equation:

$$V' = V/V[0] - 1 \tag{1}$$

After constructing V',the minimum value V_{min} and maximum value V_{max} among all transformed V' vectors are recorded. Then, the final result V'' is constructed through:

$$V'' = (V' - V_{min})/(V_{max} - V_{min}) \tag{2}$$

Following this processing, the ordering of data in the original dataset is preserved, making it easier for the neural network to train.

Besides normalization, the sliding window algorithm is also introduced, a common technique for manipulating time series data. This algorithm works by selecting a specific point t in time along with a fixed window length win_len of the subsequent data. The corresponding y label is then created from data at the position $t + win\_len + 1$ of the original data set (see figure x). The sliding window maximizes the usage of time series data while maintaining the inherent sequence of the dataset. The window length chosen for the project is $win\_len = 5$. The batch size selected is $batch = 5$.Thus, the resulting input that is ready for training has the dimension of $(x, 5, 5)$, where the first omitted value x represents the size of the dataset, the first 5 represents the batch size $batch$, and the second 5 is the chosen window length win_len.

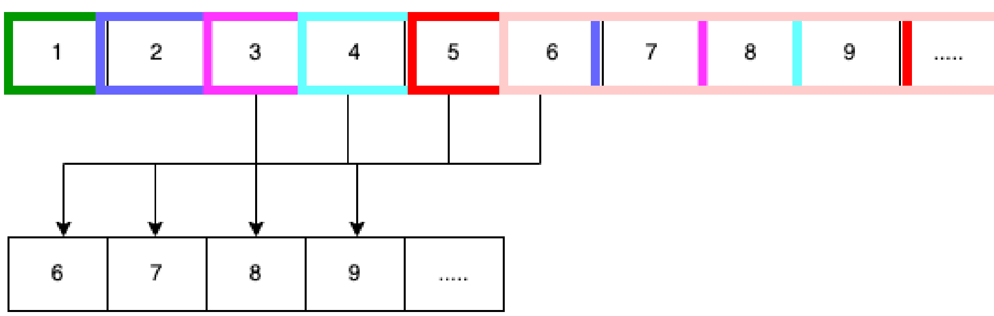

Figure 4: The illustration of sliding window for time series data

4.3 Parameters

Hyperparameters are tuned through the grid search technique. The parameters chosen for each model are summarized in Table 2 for reproducibility.

Table 2: Hyperparameters and parameters choices for each models

Models	window_len	hidden_dim	learning_rate	num_epoch	drop_out_prob [a]
LSTM(Baseline)	5	5	1e-5	5	-
Stacked LSTM	5	5	1e-5	5	0.2
CNN-LSTM	5	5	5e-6	4	-

[a] Note that the dropout probability was set to same acorss all dropout layers

4.4 Method

To evaluate the performance, the mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE) are used as performance indicators. The equations are:

$$MSE = \frac{1}{T}\sum_{t=1}^{T}(P_t - P_t')^2 \tag{3}$$

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(P_t - P_t')^2} \tag{4}$$

$$MAE = \frac{1}{T}\sum_{t=1}^{T}|(P_t - P_t')| \tag{5}$$

$$MAPE = \frac{100}{T}\sum_{t=1}^{T}|\frac{P_t - P_t'}{P_t}| \tag{6}$$

where P_t is the acutal price of Bitcoin, P_t' is the predicted price of Bitcoin.

4.5 Results

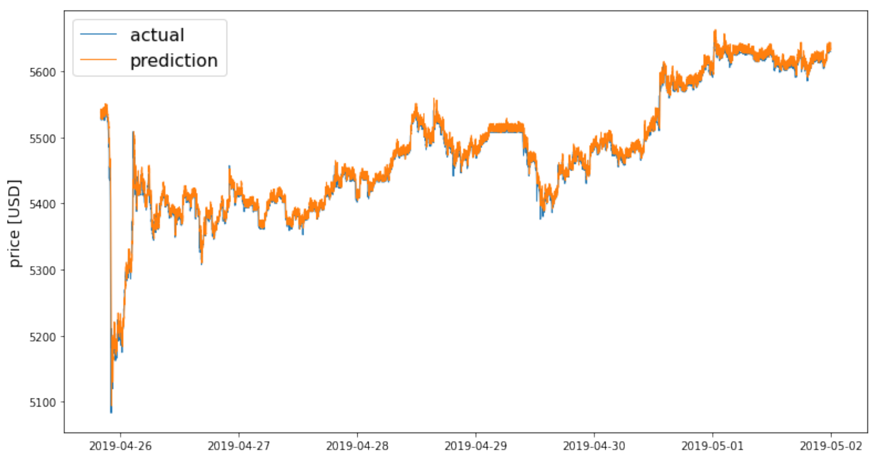

Figure 5: Comparison of the predicted value and the real value for LSTM(baseline)

Table 3: Precision results of the models under different estimation indicators.

	Precision			
Models	MSE[a]	RMSE[b]	MAE[c]	MAPE[d]
LSTM(Baseline)	27.89	52.82	50.76	1.56
Stacked LSTM	25.86	50.85	48.79	1.60
CNN-LSTM	7.041	26.53	22.41	8.07

[a] MSE Unit: 10^{-6} [b] RMSE Unit: 10^{-4} [c] MAE Unit: 10^{-4} [d] MAPE Unit: Percentage %

Note: these values were obtained by average the outcome running ten times.

The processed training dataset was used to train the LSTM, Stacked LSTM, and CNN-LSTM models, respectively. Upon completion of the training, the trained models were utilized to predict the test dataset, and the real values of the test dataset were compared. The comparison of the predicted values and the real values

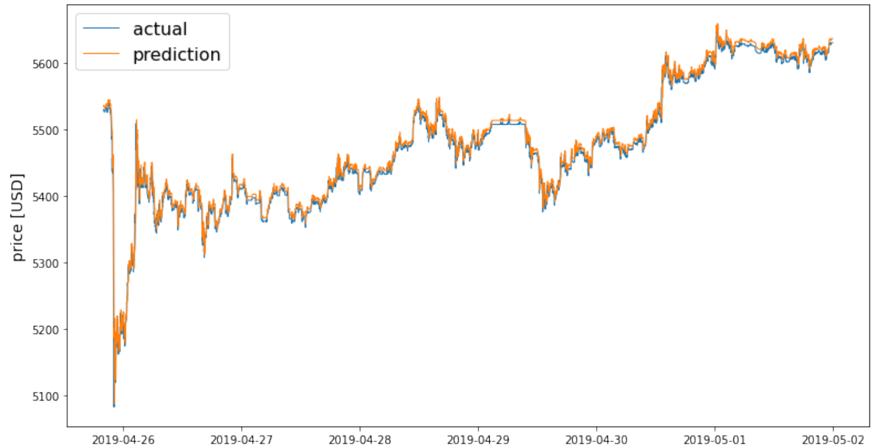

Figure 6: Comparison of the predicted value and the real value for Stacked LSTM

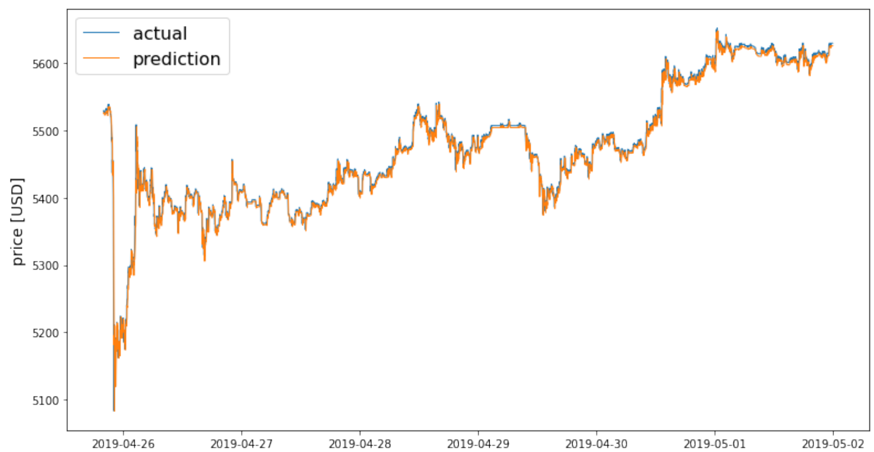

Figure 7: Comparison of the predicted value and the real value (all test data set) for CNN-LSTM

was shown in Figures 5-8, corresponding to LSTM, Stacked LSTM, and CNN-LSTM, respectively. Figures 8-10, which include only 500 data points, were also included to help visually estimate the performance of the models. After several experiments, the appropriate hyperparameters and model parameters for each model on the validation dataset were found. Each model was run several times to obtain the average of the different estimation indicators. The results are included in Table 3.

4.6 Analysis

From Table 2, it is noted that the MSE, RMSE, and MAE of the CNN-LSTM are the smallest among the three models. When comparing the Stacked LSTM with the baseline LSTM, it is observed that the MAE of Stacked LSTM decreased from 27.89 to 25.86, and the RMSE decreased from 52.83 to 50.85. This decrease aligns with expectations since dropout layers were added to each LSTM layer, and these layers prevent the model from overfitting, thereby reducing the likelihood of data oscillation. The results indicate that the performance of CNN-LSTM is superior among the three models, with the lowest MSE, RMSE, and MAE recorded at 7.041, 26.53, and 22.41, respectively. The significant increase in accuracy was attributed to the newly added CNN layers. The CNN layer functioned as expected, selecting and categorizing important features from the data, making the model less sensitive to minor changes in the original dataset. In Figure 10, the predicted curve and the actual curve almost exactly match, demonstrating that the model can accurately predict the closing price of the next minute. This accuracy offers potential for the development of automatic transaction robots.

5 Conclusion

To predict the closing price of cryptocurrency for the next minute based on the previous minutes' closing prices, three different models were proposed: LSTM, stacked-LSTM, and CNN-LSTM. It was found that the CNN-LSTM model was most suitable for forecasting the closing price of the Bitcoin market on a minute level. The performance of this model was then tested on the Bitcoin market, revealing that the forecast MSE and MAE scores of the CNN-LSTM model were statistically significantly better than those of the original LSTM, which is considered the state-of-the-art model for time series data prediction.

The study contributed primarily in the following areas. First, it provided more perspective views on minute-level cryptocurrency price prediction. The minute-level cryptocurrency dataset differed from daily-level data, as it fluctuated rapidly within minutes. Additionally, due to the nature of cryptocurrency, extremely sudden price changes occurred frequently and were common, which posed challenges for the models to distinguish between normal price fluctuations and sudden changes. Such characteristics made many statistical tests required by traditional time series models like ARMA fail, and even transformations did not yield satisfactory results. This also posed a challenge to traditional ML models like LSTM. The study explained how to overcome the specificity of minute-level price prediction with different architectures of LSTM and CNN on a minute-level cryptocurrency dataset. Second, the performance of three different neural network models—LSTM, stacked-LSTM, and CNN-LSTM—was compared in different estimations. It was found that CNN-LSTM was more suitable for predicting cryptocurrency closing prices than LSTM in this case. The results led to further interpretations about why one model worked better than another. Third, possible explanations for the better performance of CNN-LSTM models were added. It is believed that CNN can capture the key characteristics from the highly fluctuating data. CNN transformed the highly unstable data into relatively stable data, which is why it was easier for CNN-LSTM to accurately predict the closing prices with an extremely small amount of previous data. This research could serve as a foundational step towards developing a personal trading bot from scratch.

References

[1] D. G. Baur, K. J. Hong, and A. D. Lee, ”Bitcoin-Currency or Asset?,” presented at the 2016 Financial Institutions, Regulation Corporate Governance (FIRCG) Conference, Melbourne Business School, 2015.

[2] T. Guo and N. Antulov-Fantulin, ”An experimental study of Bitcoin fluctuation using machine learning methods,” in arXiv preprintarXiv:1802.04065,https://arxiv.org/abs/1802.04065, 2018.

[3] E. Ş. Karakoyun and A. O. Çibikdiken, ”Comparison of ARIMA TimeSeries Model and LSTM Deep Learning Algorithm for Bitcoin Price Forecasting,” in The 13th Multidisciplinary Academic Conference in Prague 2018 (The 13th MAC 2018), ed. Czech Republic, 2018, pp. 171-180.

[4] Wu, Chih-HungMa,Yu-FengLu, Chih-ChiangLu, Ruei-Shan“A New Forecasting Framework for Bitcoin Price with LSTM’ in the 2018 IEEE International Conference on Data Mining Workshops (ICDMW) International Journal of Computer Applications (0975 – 8887) Volume 31– No.8, October 2011

[5] Haşim Sak, Andrew Senior, Françoise Beaufays,”Long Short-Term Memory-Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition, arxiv.org, 05 Feb. 2014. ”

[6] Lu, W., Li, J., Li, Y., Sun, A., amp; Wang, J. (2020, November 24). A CNN-LSTM-based model to forecast stock prices. Complexity. Retrieved December 10, 2021, from https://www.hindawi.com/journals/complexity/2020/6622927/.

[7] Akyildirim, E., Goncu, A., Sensoy, A. (2020, April 7). Prediction of cryptocurrency returns using machine learning. Annals of Operations Research. https://link.springer.com/article/10.1007/s10479-020-03575-y.

[8] J. Li, S. Pan, L. Huang, and X. Zhu, “A machine learning based method for customer behavior prediction,” Tehnicki VjesnikTechnical Gazette, vol. 26, no. 6, pp. 1670–1676, 2019.

[9] Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R. (2012). ”Improving neural networks by preventing co-adaptation of feature detectors, ” ArXiv, abs/1207.0580.