The Effect of Long Short-Term Memory Forecasting with Varied Time Frames

Pongsakorn Teerarassamee[1], Ratiporn Chanklan, Kittisak Kerdprasop[1], and Nittaya Kerdprasop[1,*]

[1]School of Computer Engineering, Suranaree University of Technology, Thailand.
[2]Faculty of Engineering and Technology, School of Computer Engineering, Rajamangala University of Technology Isan, Nakhon Ratchasima, Thailand.
Email: psk.trrsm@gmail.com (P.T.); arc_angle@hotmail.com (R.C.); kerdpras@sut.ac.th (K.K.); nittaya@sut.ac.th (N.K.)
[*]Corresponding author
Manuscript received December 26, 2023; revised March 4, 2024; accepted March 17, 2024; published September 26, 2024

***Abstract*—This study explores the application of Long Short-Term Memory (LSTM) networks to predict the price of Bitcoin over various time periods. The performance of the model is evaluated using cross-validation methods and parameter selection techniques. The results show that the LSTM model is able to accurately predict the price of Bitcoin, with performance improving as the time period of the data increases. This suggests that LSTM networks are well-suited for modeling time series. Our research study contributes to the determination of the optimal parameters and cross-validation methods for LSTM models applied to financial market data.**

***Keywords*—bitcoin forecasting, long short-term memory, LSTM, deep learning, machine learning.**

I. Introduction

Machine learning has developed into a crucial tool for forecasting distinct outcomes across several fields, including finance. Currently, it has been widely accepted that deep neural network is one of the most powerful machine learning model for estimating future and unknown values. A neural network can refer to either a neural circuit of biological neurons or a network of artificial neurons. There are numerous ways to use a neural network. Prior to the acquisition of the deep neural network known as a long short-term memory (LSTM), the recurrent neural network (RNN) was one of the most acceptable model.

RNN is a class of neural networks where connections between nodes can create a cycle, allowing output from some nodes to affect subsequent input to the same nodes. This allows it to exhibit temporal dynamic behavior. Derived from feedforward neural networks, RNNs can use their internal state (memory) to process variable length sequences of inputs. This makes them applicable to tasks such as unsegmented, connected handwriting recognition or speech recognition.

LSTM is the next generation of RNN. Unlike standard feedforward neural networks, LSTM has feedback connections. Based on advancement of RNN structure, LSTM can process not only single data point (such as images), but also entire sequences of data (such as speech or video). This characteristic makes LSTM networks ideal for processing and predicting data. LSTM has proven to outperform traditional RNNs in many applications including speech recognition [1], natural language processing [2], and image captioning [3]. LSTM has also successfully predicted various types of time-series data such as stock prices [4], energy consumption [5], and weather forecast [6].

In this research, we apply LSTM to forecast Bitcoin, which is a decentralized digital currency also known as a cryptocurrency. A cryptocurrency is designed to work as a medium of exchange through a computer network that is not reliant on any central authority, such as a government or bank, to uphold or maintain it. It is a decentralized system for verifying that the parties to a transaction have the money they claim to have, eliminating the need for traditional intermediaries, such as banks, when funds are being transferred between two entities. Individual coin ownership records are stored in a digital ledger, which is a computerized database using strong cryptography to secure transaction records, control the creation of additional coins, and verify the transfer of coin ownership.

Despite their name, cryptocurrencies are not considered to be currencies in the traditional sense, and while varying treatments have been applied to them, including classification as commodities, securities, and currencies, cryptocurrencies are generally viewed as a distinct asset class in practice. Some crypto schemes use validators to maintain the cryptocurrency. In a proof-of-stake model, owners put up their tokens as collateral. In return, they get authority over the token in proportion to the amount they stake. Generally, these token stakers get additional ownership in the token over time via network fees, newly minted tokens, or other such reward mechanisms.

Bitcoin is the most well-known cryptocurrency, among thousands of others. Bitcoin was introduced to the public in 2009 by an anonymous developer or group of developers using the name Satoshi Nakamoto. It has since become the most well-known cryptocurrency in the world. Its popularity has inspired the development of many other cryptocurrencies. Bitcoin is opensource. The design is public, nobody owns or controls Bitcoin and everyone can take part. Through many of its unique properties, Bitcoin allows exciting uses that could not be covered by any previous payment system. Based on its fast peer-to-peer transactions, worldwide payments, and low processing fees, we are thus interested in studying a model to predict Bitcoin values using LSTM as a forecasting method.

II. Theory and Related Work

A common LSTM unit is composed of a cell, an input gate, an output gate, and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell.

The forget gate decides what information to discard from a previous state by assigning a previous state, compared to a current input, a value between 0 and 1. A value of 1 means to keep the information, and a value of 0 means to discard it. The input gate decides which pieces of new information to store in the current state, using the same system as forget gates. The output gate controls which pieces of information in the current state to output by assigning a value from 0 to 1 to the information, considering the previous and current states. These gates control the flow of information in and out of the memory cell or LSTM cell.

Forget gate is a gate that decides whether data entering the cell state should be discarded or stored. By evaluating from the information that enters that node and the previous node through the sigmoid function as in Eq. 1.

$$f_t = \sigma\,(\mathrm{W}_f \cdot [h_{t-1}, x_t] + b_f) \qquad (1)$$

where

f_t is the output from forget gate, σ is the sigmoid function, W_f is the weight value, h_{t-1} is the output of the previous cell state at t–1, x_t is the input that comes in at time t, and b_f is the bias value.

The result obtained from forget gate is between zero and one. If it is zero, then the value from the previous cell state is deleted. If non-zero, the value from the previous cell state will be passed on.

Input gate is a gate that helps to update the cell state. It is divided into two parts. The first part is the sigmoid function which controls how much value can be remembered as in Eq. 2. The second part is the tanh function which is used to generate candidate values as in Eq. 3.

$$i_t = \sigma\,(W_x * [h_{t-1}, x_t] + b_x) \qquad (2)$$

$$C_t = tanh\,(W_y * [h_{t-1}, xt] + b_y) \qquad (3)$$

where

i_t is an input gate, σ is the sigmoid function, C_t is the candidate value at time t, tanh is tanh function, W_x and W_y are the weight values, h_{t-1} is the output of the previous cell state at t–1, x_t is the input that comes in at time t, b_x and b_y are the bias values.

Output gate is a gate that prepares to return the output value. It is divided into two parts. The first part, the sigmoid function, controls how much the value will be memorized. The second part is the tanh function used to generate the candidate value by taking the latest cell state value to calculate through the tanh function. Finally, the output value will be generated from both parts, which are $o_t \times C_t$ obtained from Eqs. 4 and 5.

$$o_t = \sigma\,(\mathrm{W}_{out} * [\mathrm{h}_{t-1}, \mathrm{x}_t] + \mathrm{b}_{out}) \qquad (4)$$

$$C_t = tanh\,(W_o * [h_{t-1}, xt] + b_o) \qquad (5)$$

where

o_t is an input gate, σ is the sigmoid function, C_t is the candidate value at time t, tanh is tanh function, W_{out} and W_o are the weight values, h_{t-1} is the output of the previous cell state at t–1, x_t is the input that comes in at time t, b_{out} and b_o are the bias values.

The stock market and cryptocurrency are two different types of investment vehicles. The stock market is a marketplace where investors can buy and sell shares of publicly traded companies, while cryptocurrency is a digital or virtual currency that uses cryptography for security and operates independently of a central bank.

One key difference between the two is the type of assets traded. Stocks represent a share of ownership in a company, while cryptocurrencies are digital assets that are created and stored using blockchain technology. Stocks have real-world value, while cryptocurrencies have digital value. Another difference is the level of regulation. The stock market is heavily regulated, while the cryptocurrency market is largely self-regulated and has little interference from third-party intermediaries.

In terms of similarities, both the stock market and cryptocurrency market allow investors to trade assets with the goal of making a profit. Both markets can be volatile and subject to fluctuations in value. It is important to note that investing in either the stock market or cryptocurrency carries risks, and it is important to thoroughly research and understand the potential risks and rewards before making any investment decisions.

Stock markets can often be one of the most volatile places to invest, and statistical analysis of past stock performance and external factors play a major role in the decision to buy or sell stocks. These factors are all used to maximize profits. Stock price index forecasting has been a subject of great research for many years, and several machine learning and deep learning algorithms have been proposed to simplify this complex task, but little success has been found so far. In order to forecast stocks accurately, it is crucial to understand the context-specific dependence of stock prices on their past values. The use of LSTM, which is capable of understanding long-term data dependencies, can help overcome this obstacle.

In 2023, Chenyu Han and Xiaoyu Fu [7] proposed a bi-directional long short-term memory (Bi-LSTM) model for predicting future stock prices based on historical prices. The model was applied to Apple Inc. stock price data and evaluated using mean squared error and visual inspection. The results showed that the Bi-LSTM model could make accurate predictions and capture trends and patterns in the data but may struggle with sudden market changes. The Bi-LSTM model is a promising tool for stock price prediction with potential applications in finance and investment.

Similarly, in 2023, Abdul Quadir *et al.* [8] proposed a novel optimization approach for stock price prediction based on a Multi-Layer Sequential Long Short Term Memory (MLS LSTM) model using the adam optimizer. The MLS LSTM algorithm used normalized time series data divided into time steps to determine the relationship between past and future values, and eliminated the vanishing gradient problem associated with simple recurrent neural networks. The stock price index was forecasted by taking into account past performance information along with past trends and patterns. The results showed that the MLS LSTM algorithm achieved 95.9% prediction accuracy on the training data set and 98.1% accuracy on the testing data set, outperforming other machine learning and deep learning algorithms. The mean absolute percentage error was 1.79% on the training set and 2.18% on the testing set, respectively. Moreover, the proposed model was able to estimate the stock price with a normalized root mean squared error of 0.019, thus giving an accurate forecast and making it a feasible real-world solution.

In 2023, Ramani *et al.* [9] used deep learning and machine learning algorithms, including LSTM, Autoregressive Integrated Moving Average, XGBoost, Prophet, and

Sentiment analysis, to perform predictions on Bitcoin data. The algorithms were trained on live streaming financial data and their results were compared based on predicted metrics such as Root Mean Square Error, Mean Absolute Error, and R2. The results showed that Sentiment analysis combined with LSTM provided better performance in Bitcoin price prediction than all other algorithms.

Despite its popularity, Bitcoin faces several challenges, including volatility, scalability, cyber theft, and price prediction. Among these challenges, this paper focuses on price prediction. The authors used deep learning and machine learning algorithms to predict Bitcoin prices and found that combining Sentiment analysis with LSTM provided the best performance.

III. Materials and Methods

This study used the LSTM model from Keras to analyze and make predictions based on the data. LSTM is a type of recurrent neural network (RNN) that is capable of learning long-term dependencies in data. This makes it well-suited for tasks such as time series prediction, where the goal is to predict future values based on past observations.

The data that will be used for analysis and prediction are the Bitcoin historical prices. These data have been obtained from a reputable and reliable source, specifically the Binance Exchange, which is one of the largest and most popular cryptocurrency exchanges in the world. Binance offers a wide range of trading pairs and has a high trading volume, making it a reliable source for obtaining accurate and up-to-date Bitcoin price data. By using data from Binance, we can ensure that the data are accurate by period, allowing for more reliable predictions and analysis. The data were carefully selected and pre-processed to ensure their quality and relevance to the study. This will involve cleaning the data to remove any errors or inconsistencies, as well as normalizing the data to ensure that it is in a format that can be easily analyzed.

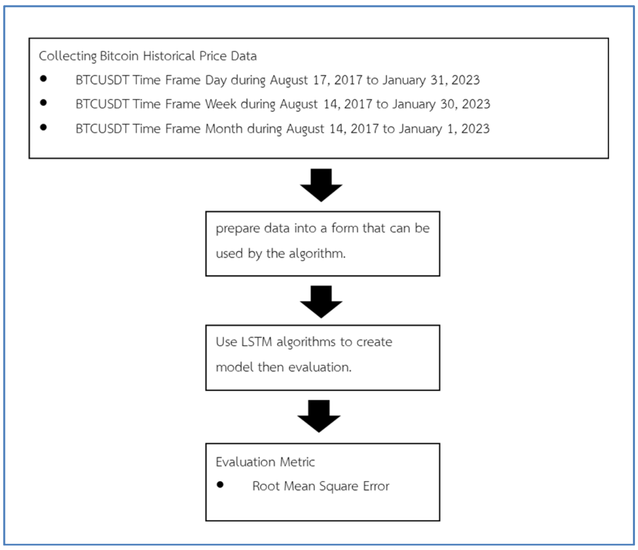


Fig. 1. Data preparation and modeling with LSTM.

Data were collected across three different time frames: daily, weekly, and monthly. This allowed for examination of changes in Bitcoin's price at different levels of granularity and identification of both short-term and long-term trends.

For the daily time frame, data was collected for the period between August 17, 2017 and January 31, 2023. This provided a detailed view of how Bitcoin's price changed on a day-to-day basis over a period of several years.

For the weekly time frame, data was collected for the period between August 14, 2017 and January 30, 2023. This allowed for examination of changes in Bitcoin's price on a weekly basis, providing a slightly broader view of trends and patterns.

For the monthly time frame, data was collected for the period between August 14, 2017 and January 1, 2023. This provided an even broader view of changes in Bitcoin's price over time, allowing for identification of long-term trends and patterns. Modeling concept is shown in Fig. 1.

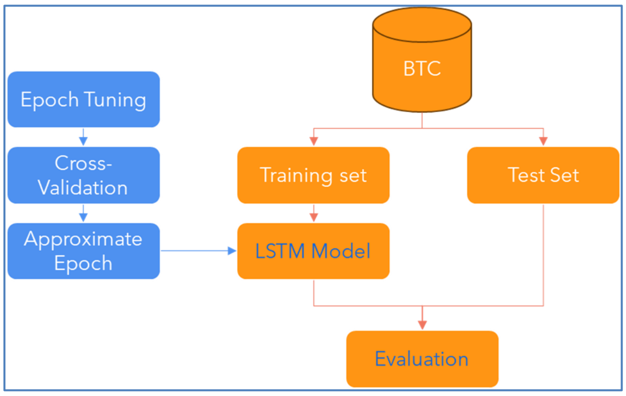


Fig. 2. Modeling process and parameter tuning.

The steps in data preparation and modeling (Fig. 2) are as follows:

(1) The first step in this process is to download historical Bitcoin price data from the Binance Exchange, which is one of the largest and most popular cryptocurrency exchanges in the world.

(2) The next step is to prepare the data for appropriate use with the LSTM model from Keras. LSTM model has many parameters that need to be defined, so it is important to carefully select the best values for these parameters in order to achieve optimal performance. To aid in this decision-making process, we will use a technique called cross-validation. This involves dividing the data into multiple subsets and training the model on each subset, then evaluating its performance on the remaining data. By doing this, we can determine which parameter values resulting in the best performance and use those values when training our final model.

(3) Once the data has been prepared and the optimal parameter values have been determined, we can move on to training our LSTM model. To do this, we will divide our Bitcoin price data into two sets: a training set and a test set. The Training set will be used to train our LSTM model, while the test set will be used to evaluate its performance. By training our model on the training set and then testing its performance on the test set, we can determine how well our model is able to predict future Bitcoin prices.

(4) The last step is parameter tuning. LSTM has many parameters that need appropriate tuning including number of neurons, batch size, epoch. In the context

of neural networks, the number of neurons in the input layer is typically equal to the number of features in the input data. This is because each neuron in the input layer represents a single feature and is responsible for processing the information contained in that feature. For example, if data set has 10 features, the neural network typically has 10 neurons in the input layer. Each neuron would receive the value of one of the features as its input and would process that information before passing it on to the next layer of the network. It is important to note that this is just a general guideline and that the number of neurons in the input layer can vary depending on the specific architecture of the neural network and the problem being solved. For this study, the dataset contains only close price feature of Bitcoin, so number of neuron is 1. The batch size refers to the number of samples that are processed at once by the model during each iteration of training. In this study, the initial value for the batch size has been set to 100. This means that during each iteration of training, the model will process 100 samples at a time. The number of epochs refers to the number of times the entire training dataset is passed through the model during training. In this study, cross-validation is used to approximate the optimal number of epochs for training the model. Cross-validation involves dividing the data into multiple subsets and training the model on each subset, then evaluating its performance on the remaining data. By doing this, we can determine how many epochs are needed to achieve optimal performance from the model.

IV. Result and Discussion

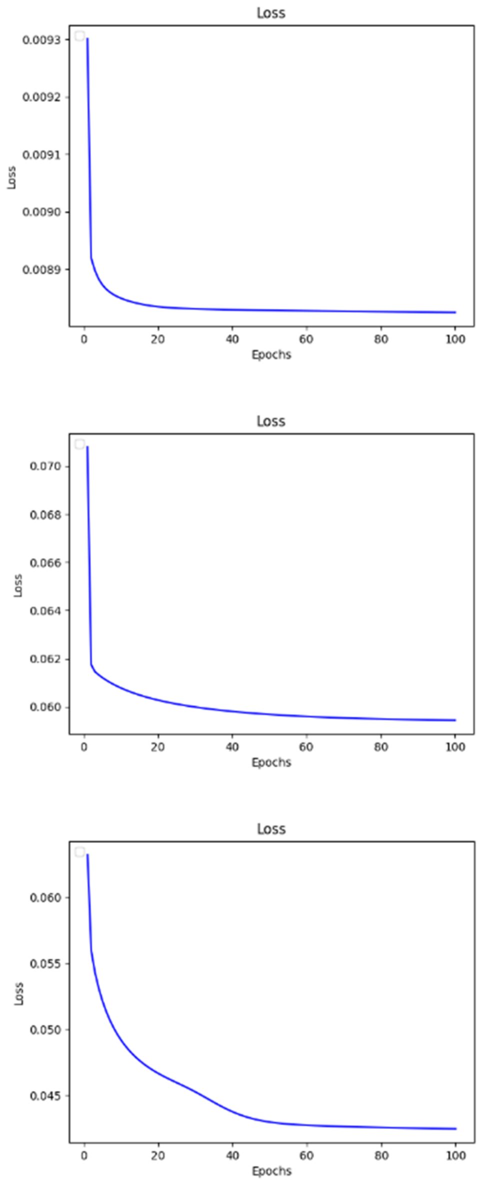
In this research, the LSTM algorithm was used to analyze and make predictions based on historical data. The results of the research were divided into three parts according to the nature of the data, with separate analyses conducted for daily, weekly, and monthly time frames for Bitcoin forecasted prices.

To prepare the most accurate LSTM model, this study focused on estimating the optimal number of epochs for training the model. This was done using a technique called cross-validation with number of folds to be 3 (or k=3). Cross-validation involves dividing the data into multiple subsets and training the model on each subset, then evaluating its performance on the remaining data. By doing this, this study was able to determine the optimal number of epochs for training the LSTM model and achieve the best possible performance.

Cross-validation is a technique used to assess the performance of a machine learning model and to select the best parameters for the model. In the context of time series data, cross-validation can be used to determine how well a model is able to make predictions based on past observations. One common approach to cross-validation for time series data is the "rolling window" method. In this approach, the data is divided into multiple subsets, with each subset consisting of a fixed number of observations. The model is then trained on each subset and its performance is evaluated on the next observation in the series.

For example, suppose there is a time series with 10 observations and the cross-validation is applied with a rolling window of size 3. In this case, the data are divided into subsets, with each subset consisting of 3 observations. The model is then trained on the first subset (observations 1–3) and evaluate its performance on the next observation (observation 4). This process is repeated for each of the remaining subsets, that is, training the model on observations 2–4 and evaluating its performance on observation 5, training the model on observations 3–5 and evaluating its performance on observation 6, and so on. By using cross-validation in this way, we can assess how well our model is able to make predictions based on past observations and select the best parameters for our model.

The results of epoch calculation using the cross-validation method are shown in Fig. 3. From the knee point, the evaluation for selection will be taken into considering. It can be seen from the results that epoch = 20 is the best parameter selection.



(a) BTC daily timeframe

(b) BTC weekly timeframe

(c) BTC monthly timeframe

Fig. 3. Bitcoin price loss values from cross-validation technique.

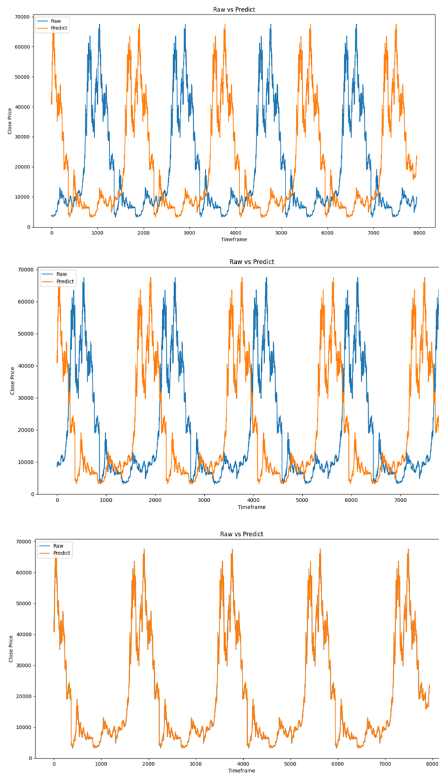
After completing the cross-validation process, it was determined that the optimal number of epochs for the LSTM model was 20. With this information, the model building and testing could proceed using the LSTM with the specified epoch setting.

In this study, the LSTM model was configured with three layers, each containing a different number of neurons. The first layer had 128 neurons, the second layer had 256 neurons, and the third layer had 512 neurons. The performance of the model was then evaluated based on the root mean square error (RMSE) metric by comparing the results obtained when using only the first layer, when using the first and second layers, and when using all three layers. The results are summarized and shown in Table 1.

Table 1. LSTM model evaluation results

Time series at different timeframes	**L1 RMSE**	**L1+L2 RMSE**	**L1+L2+L3 RMSE**
BTCUSDT_TFDAY	1148.26	1159.63	1130.08
BTCUSDT_TFWEEK	12595.56	6297.50	3464.55
BTCUSDT_TFMONTH	13507.77	11414.10	6900.87

The results in Table 1 reveal that LSTM trained from 20 epochs using all three layers can forecast Bitcoin prices with minimal errors. Forecasting results based on daily, weekly, and monthly timeframes using one (L1), one and two (L1+L2), and all three layers (L1+L2+L3) of the networks are displayed in Figs. 4–6, respectively.

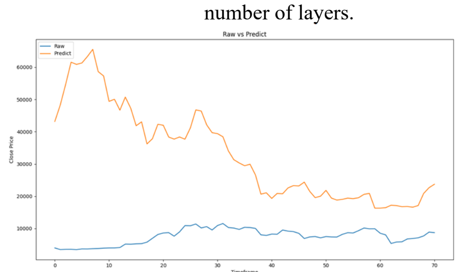


(a) L1

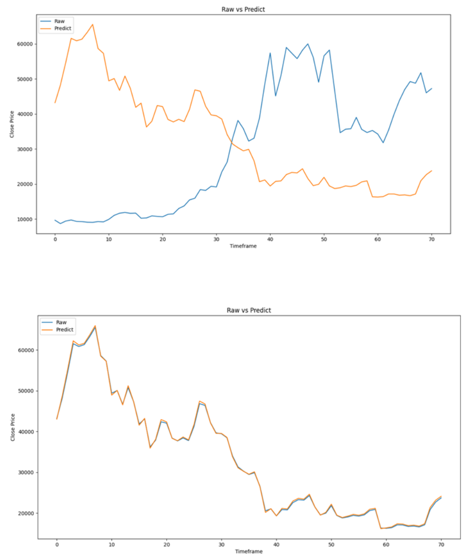
(b) L1+L2

(c) L1+L2+L3

Fig. 4. Forecasting Bitcoin prices on a daily timeframe data with different number of layers.



(a) L1



(b) L1+L2

(c) L1+L2+L3

Fig. 5. Forecasting Bitcoin prices on a weekly timeframe data with different number of layers.

(a) L1

(b) L1+L2

(c) L1+L2+L3

Fig. 6. Forecasting Bitcoin prices on a monthly timeframe data with different number of layers.

V. Conclusion

This research performs the modeling and parameter tuning process of Bitcoin price forecasting using the long short-term memory (LSTM) algorithm. LSTM is a type of recurrent neural network that is capable of learning long-term dependencies in data. This makes it well-suited for tasks such as time series prediction, where the goal is to predict future values based on past observations. In this work historical Bitcoin prices were obtained from the Binance Exchange, which is one of the largest and most popular cryptocurrency exchanges in the world. Binance offers a wide range of trading pairs and has a high trading volume, making it a reliable source for obtaining accurate and up-to-date Bitcoin price data.

Obtained data between the year 2017 to 2023 were prepared to be of three different time frames: daily, weekly, and monthly. These various timeframe data are to identify LSTM characteristics best suit the Bitcoin forecasting model. On the modeling phase, three kinds of LSTM networks are also examined. That are the network with one layer, two layers, and three layers.

Based on the results of the performance comparison, it was observed that an increase in the number of layers led to a reduction in the error value. Furthermore, an analysis of all three datasets revealed that the smallest timeframe resulted in the lowest overall error. It should be noted, however, that this study only utilized root mean square error (RMSE) as a measure of error. This study is unique in its focus on the optimal selection of epoch parameters and the use of a single performance measure, RMSE. Future research could expand on this by considering other parameters and incorporating additional performance measures, including the number of layers used. Different parameter settings could also be explored.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

The first author is responsible for data collection, experimentation and manuscript preparation. The second author helps revising the research idea and experimental results. The third author reviews and confirms correctness of the design and experimentation. The fourth author contributes the idea, research design improvement, and manuscript submission.

References

[1] A. Graves, A. R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, Canada, May 26–31, 2013, pp. 6645–6649.

[2] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. 27th International Conference on Neural Information Processing Systems*, Montreal, Canada, December 8, 2014, pp. 3104–3112.

[3] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Boston, U.S.A., June 7–12, 2015, pp. 3156–3164.

[4] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[5] M. Brown, F. de Hoog, M. Verlage, and T. Grüning, "LSTM neural networks for short-term load forecasting: A review," *Renewable and Sustainable Energy Reviews*, vol. 81, pp. 1548–1568, 2018.

[6] X. Shi, Z. Chen, H. Wang, D. Y. Yeung, W. K. Wong, and W. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. 28th International Conference on Neural Information Processing Systems*, Montreal, Canada, December 7–12, 2015, pp. 802–810.

[7] C. Han and X. Fu, "Challenge and opportunity: Deep learning-based stock price prediction by using bi-directional LSTM model," *Frontiers in Business, Economics and Management*, vol. 8, no. 2, pp. 51–54, 2023.

[8] A. Q. Md, S. Kapoor, C. Junni, A. K. Sivaraman, K. F. Tee, H. Sabireen, N. Janakiraman, "Novel optimization approach for stock price forecasting using multi-layered sequential LSTM," *Applied Soft Computing*, vol. 134, article 109830, 2023.

[9] K. Ramani, M. Jahnavi, P. J. Reddy, P. VenkataChakravarthi, P. Meghanath, and S. K. Imran, "Prediction of Bitcoin price through LSTM, ARIMA, XGBoost, Prophet and sentiment analysis on dynamic streaming data," in *Proc. 9th International Conference on Advanced Computing and Communication Systems*, Coimbatore, India, March 17–18, 2023, pp. 1514–1518.