Forecasting Bitcoin Prices Using Deep Learning for Consumer-Centric Industrial Applications

Pradeep Kumar Roy, Abhinav Kumar, Ashish Singh, and Arun Kumar Sangaiah

***Abstract*—As cryptocurrencies become more popular as investment vehicles, bitcoin draws interest from businesses, consumers, and computer scientists all across the world. Bitcoin is a computer file stored in digital wallet applications where each transaction is secured using strong cryptographic algorithms. It was challenging to forecast the future price of bitcoin due to its nonlinearity and extreme volatility. Several recent classic parametric models have been found with limited accuracy. To address the limitations and fill the existing research gaps, there is a need for a good prediction model which will provide the desired accuracy in the case of uncertainty and dynamism. This research suggested a deep learning-based framework for predicting and forecasting Bitcoin price. The research will be helpful for worldwide consumers and industries to take their decision on whether to invest or not. The research utilizes Yahoo! finance dataset for the period of 01-03-2016 to 26-02-2021 having 1828 samples. The experimental outcomes of the proposed Long Short-Term Memory (LSTM) model outperformed similar deep learning models by securing minimum loss and confirming that it can be used for future price prediction of the cryptocurrencies, which is helpful for the buyer to take their decision.**

***Index Terms*—Deep learning, cryptocurrency, bitcoin, GRU, LSTM, machine learning.**

I. Introduction

NOWADAYS, the world is shifting to digital currency or tokens due to its faster, high returns, rapid growth, and cheaper transaction cost. One of the most important and attractive digital currencies is bitcoin [1], [2]. It is the first cryptocurrency invented in 2008 to solve the trust issues in currency-based transactions. It was initially developed for a peer-to-peer cash system in which a chain of valid transactions is distributed among all peers in the decentralized network [3], [4]. Now, it has become a popular currency, and its usage is a substantial rise in market capitalization in recent years. Cryptocurrencies are popular among many consumers and industries because of their highly secure transactions. A recent crypto survey conducted by 54% of fintech experts said bitcoin overtake the currencies issued by central banks in the global finance market by 2050.[1]

The recent studies says that the bitcoin price is very fluctuated and uncertain [4], [5]. As it plays a very important role in the cryptocurrency-based global economy and the price fluctuation may affect the economy of the global market [6]. The future price prediction of cryptocurrency (bitcoin) would help investors choose the best option to invest their money. It will also improve the economy of the bitcoin market. It will depend upon many factors, including market trends, transaction cost, popularity, mining, difficulty, price of alternate coins, stock markets, sentiments, and some legal factors [7]. Thus, the future price prediction of bitcoin will gain popularity in the financial field.

In recent years, financial experts have tried to tackle this problem by investigating various factors that affect the Bitcoin price and the patterns behind its fluctuations. The advancement in prediction methodology using artificial intelligence, machine learning, and deep learning approaches makes it easier to predict the price accurately [4], [8]. But, in the case of high volatile and stochastic financial time series data, the algorithm needs to find out the hidden pattern of the data. This is a still challenging issue. Some of the prediction models illustrated that hybrid deep learning models with decomposition algorithms yield more accurate prediction results than individual deep learning or machine learning models [9]. With the popularity of bitcoin, seasonality and high volatility in the bitcoin market may affect the accuracy of the prediction model. In such a case, deep learning-based hybrid models are interesting technical solutions. There is a need for a more robust and dynamic price prediction model which will also handle the cases of volatile, seasonality, and stochastic financial time series data [10].

To address the challenges and fill the research gap, this research suggested a future price prediction model for cryptocurrency. The model can extract hidden contextual patterns from the time series data. The deep learning frameworks are investigated under several settings to get the best-suited model to predict the accurate price of the cryptocurrency [11]. Deep neural networks (DNN), convolutional neural networks (CNN) [12], deep residual networks (ResNet), and recurrent

Manuscript received 15 March 2023; revised 28 May 2023; accepted 21 September 2023. Date of publication 2 October 2023; date of current version 26 April 2024. *(Corresponding authors: Ashish Singh; Arun Kumar Sangaiah.)*

Pradeep Kumar Roy is with the Department of Computer Science and Engineering, Indian Institute of Information Technology Surat, Surat 394190, India (e-mail: pkroynitp@gmail.com).

Abhinav Kumar is with the Department of Computer Science and Engineering, Motilal Nehru National Institute of Technology Allahabad, Prayagraj 211004, India (e-mail: abhinavanand05@gmail.com).

Ashish Singh is with the School of Computer Engineering, KIIT Deemed to be University, Bhubaneswar 751024, India (e-mail: ashishashish307@gmail.com).

Arun Kumar Sangaiah is with the International Graduate Institute of AI, National Yunlin University of Science and Technology, Douliou 64002, Taiwan, and also with the Department of Electrical and Computer Engineering, Lebanese American University, Byblos 1102-2801, Lebanon (e-mail: aksangaiah@ieee.org).

Digital Object Identifier 10.1109/TCE.2023.3321653

[1]https://tinyurl.com/bp7bcbbf, accessed on 18-12-2021.

neural networks (RNN) were the subjects of most earlier research. However, the existing models need improvements to predict the price of bitcoin more accurately. Hence, the objective of this research work is to find out a method that can accurately predict the bitcoin price. This research suggests a LSTM based model to achieve the objective. The proposed model performance was compared with other deep learning frameworks like Bi-LSTM and GRU. The proposed model performed comparatively better than other experimented deep learning models. The contribution of this research work are summarized as follows:

1) Long Short-Term Memory (LSTM) based model is proposed for Bitcoin price prediction.
2) Extensive analysis of the loss vs epoch and predicted vs actual price of the Bitcoin are discussed.
3) Compared the outcomes of the proposed LSTM model with another deep learning models like Bi-LSTM, and GRU on different input settings.

The rest of the paper is organized as follows: Section II discusses the relevant works on bitcoin prediction. Section III describes the proposed methodology in detail. In Section IV, the experimental outcomes of the proposed models are discussed followed by discussion in Section V. Section VI concludes this work with possible future scope.

II. Related Work

The bitcoin price prediction has received much attention in recent years [13], [14], [15], [16]. The deep learning and time sequence-based hybrid algorithm was proposed by Rezaei et al. [17]. They claim CNN with LSTM and Complete Ensemble Empirical Mode Decomposition (CEEMD) gives more accurate predictions. Patel et al. [18] proposed a cryptocurrency price prediction system based on LSTM and Gated Recurrent Unit. Their hybrid model considered two different cryptocurrencies, namely Litecoin and Monero. A machine learning-based prediction technique has been proposed in [16]. They have classified the bitcoin price at daily and high-frequency prices at different frequencies. Their model achieved the accuracy value of 67.2% for the best suited model.

In [19], Generalized Autoregressive Conditional Heteroskedasticity and LSTM model has been composed to make a hybrid bitcoin price prediction model. An LSTM-based forecasting framework has been proposed in [20] to predict bitcoin price. A hybrid bidirectional deep learning model including Bi-GRU has been proposed in [21]. The outcome of the work illustrated that the proposed technique outperforms compared to pure ML and DL techniques.

Ji et al. [22] investigated several deep learning approaches for bitcoin price prediction. The comparative results demonstrated that LSTM-based prediction models outperformed compared to other techniques. A prediction model for cryptocurrencies like Litecoin and Zcash has been proposed in [23]. They use a hybrid model composition of GRU and LSTM. A hybrid deep learning model for the prediction of the bitcoin dollar rate has been proposed in [24]. In [25], three price prediction features are used for bitcoin price fluctuation. They have used a Denoising autoencoder, including LSTM and embedding network. Three most popular cryptocurrencies like Bitcoin, Digital Cash and Ripple price, are predicted in [26]. They used the LSTM model to implement the work and later compared it with generalized regression neural architecture. Machine learning (ML) and statistical methods are used in [16] to predict Bitcoin prices at different frequencies. They have achieved 66% accuracy, which is higher than other methods. In [27], authors use neural models to predict Bitcoin price.

LSTM is one of the popular deep learning frameworks the researcher uses for predicting the future price of bitcoin. In [11], authors predicted the forecasting of the cryptocurrencies using time series analysis with LSTM model and ML algorithms such as SVM and linear regression. Another LSTM and RNN-based deep learning model has been used in [28] to predict the future price of the cryptocurrencies. LSTM and GRU-based deep learning model was also used by [29] for future bitcoin price prediction. Hashish et al. [30] proposed a hybrid model for Bitcoin price prediction. They have combined hidden Markov models and optimized LSTM networks. The experiment was performed using Coinbase exchange market bitcoin data.

Many models have been proposed to date for forecasting the price of Bitcoin. Still, research is going on to develop a system which can be used with low resources. This research aims to build a lightweight system which can be utilized with low resources for future price prediction of cryptocurrencies.

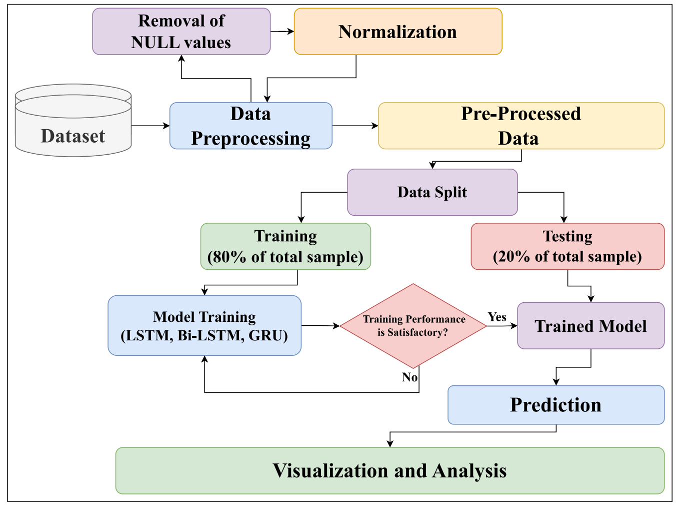

Fig. 1. Steps followed to develop Bitcoin price predictive system.

III. System Development

Bitcoin prediction is a challenging issue due to high market price fluctuations. This research aims to develop an automated model to forecast bitcoin prices with deep learning-based frameworks, which will be helpful for consumer-centric industrial applications. The goal will be achieved by utilizing the previous trends. This section highlights the methodology used for training the deep learning models for the said task. It also highlights the characteristics of the dataset used for the price prediction of Bitcoin. The detailed working steps of bitcoin price prediction model are shown in Figure 1.

A. Dataset

The Yahoo! Finance dataset from 01-03-2016 to 26-02-2021 was collected[2] for training and validating the proposed system. The dataset consists of 1,828 samples, i.e., every day's opening price, closing price, lower value of bitcoin, the highest value and other parameters are recorded. The recorded information is considered as features for building the automated predictive model. The dataset consists of seven columns-(i) Date, (ii) Opening Price, (iii) Highest price of the day, (iv) Lowest Price of the day, (v) Closing price, (vi) Adj. Close, having a similar value of (v), and (vii) the Volume- signifies the Volume of the amount of trade that day.

B. Data Preparation

The important steps used for data preparation to train the models are as follows:

1) Firstly, the dataset is cleaned by removing the rows and columns having NULL values.
2) The values of the parameters are very large; hence needed to fix them on a smaller scale. To do this, data normalization was performed using *MinMaxScaler* function. The *MinMaxScaler* function changes the values between 0 and 1. The smaller data values help the model converge faster than the model train with larger values.
3) To predict the price of the bitcoin, supervised models were used. The price was predicted with the help of current price and other factors.
4) The preprocessed dataset was split into two parts: Training and Testing, where training data consists of 80% of the total sample and the remaining 20% of samples used to test the trained models.
5) The predicted values are in the range of 0 and 1, which are further de-normalized into the respective actual values for better understanding.
6) To know the performance of the models, the evaluation metrics- RMSE (root mean square error), MAE (mean absolute error) and R2-score were used.

C. Deep Learning Frameworks

Deep learning frameworks such as LSTM, Bi-LSTM, and GRU are trained for six different output layer neurons, i.e., 25, 50, 75, 100, 125, and 150. The outputs of the first layer neurons passed to a dense layer having a single neuron. The single neuron present at the output layer is used for Bitcoin prediction. The MAE loss function was used with the Adam optimizer for all different settings during the training of the models. The batch size was fixed to 72, and the learning rate was 0.001. All models are trained for 50 epochs.

1) Long Short-Term Memory (LSTM): The LSTM network's capacity to remember lengthy input sequences is one of its main advantages over other deep learning models like CNN [31]. The LSTM network was made up of a variety of parts, as seen in Figure 2.

Input gate (i_t), output gate (o_t), memory cell (m_t), and forget gate (f_t) are the four main gates in the LSTM network. A

[2]https://in.finance.yahoo.com/quote/BTC-USD/history?p=BTC-USD

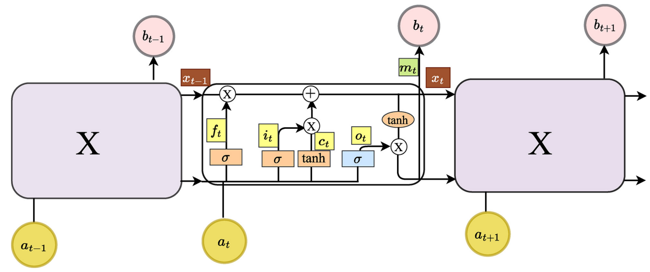

Fig. 2. Components of the Long Short-Term Memory Network.

word vector a_i is provided to the LSTM network at each time stamp t, where it is processed to produce an output b_i as seen in Figure 2. Finding irrelevant information from the cell state and discarding it is the first step of the LSTM network. The decision is taken care by the forget gate (f_t) as described in Eq. (1).

$$f_t = \sigma\left(w_f\left[x_{(t-1)}, a_t\right] + b_f\right) \tag{1}$$

where a_t is the new input message word, w_f is the weight, $x_{(t-1)}$ is the output from the previous time stamp, and b_f is the bias. The network's next step is to decide what data will be saved for later processing. The i_t, c_t, and new cell x_t are calculated using Eqs. (2), (3), and (4), respectively.

$$i_t = \sigma\left(w_i\left[x_{(t-1)}, a_t\right] + b_i\right) \tag{2}$$

$$c_t = tanh\left(w_c\left[x_{(t-1)}, a_t\right] + b_c\right) \tag{3}$$

$$x_t = f_t * x_{(t-1)} + i_t * c_t \tag{4}$$

The output o_t (Eq. (5)) of the network is finally determined with the aid of the sigmoid layer, and the cell state c_t is further passed through the function *tanh* and multiplied by the output of the sigmoid function.

$$o_t = \sigma\left(w_o\left[x_{(t-1)}, a_t\right] + b_o\right) \tag{5}$$

The price of Bitcoin was predicted using various settings of the LSTM network. Section IV discusses the experimental results produced using these models.

2) Bi-Directional LSTM (Bi-LSTM): The bi-directional LSTM model was created by combining two separate LSTM networks. At each time step, the Bi-LSTM network has access to the sequence's forward and backward information [32].

The inputs in the Bi-LSTM model will be processed in two different directions: from the start sequence to the end sequence and from the end sequence to the starting sequences. By combining the two hidden states, the current and future data can be preserved at any given time. This method differs from unidirectional in that information from the future is preserved in the LSTM that runs backwards. In Figure 3, the Bi-LSTM network is depicted graphically.

3) Gated Recurrent Unit (GRU): GRU (Gated Recurrent Unit) attempts to solve the vanishing gradient problem that plagues conventional recurrent neural networks. Due to the similarities in their designs and, in some cases, their equally impressive outcomes, GRU can also be seen as an adaptation of the LSTM. Figure 4 displays the GRU's fundamental parts.

GRU uses the *update* and *reset* gate to fix the vanishing gradient problem that a standard RNN. These two vectors

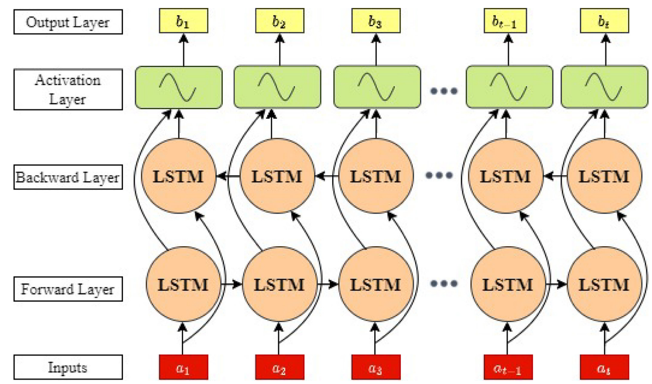

Fig. 3. A Bi-LSTM network with their working components.

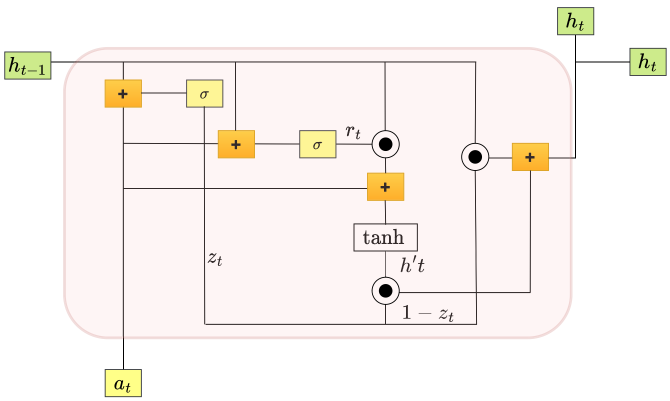

Fig. 4. A GRU model with their working components.

essentially decide what information should be sent to the output. It is possible to teach them to remember information from the past without having it vanish over time or to ignore information that is unrelated to the prediction, which gives them a special advantage.

The *update* gate z_t for time stamp t calculated using the mathematical formula (Eq. (6)):

$$z_t = \sigma\left(W^{(z)} a_t + U^{(z)} h_{t-1}\right) \tag{6}$$

The weight of x_t is multiplied by W once it is connected to the network unit. The same is true for $h_{(t-1)}$, which contains data for the previous $(t-1)$ units (z) and is multiplied by its own weight U. The result is squeezed between 0 and 1 using a sigmoid activation (σ) function after the two results are added collectively.

The *update* gate helps the model choose how much data with a previous time stamp should be sent into the future. That makes it less likely that there will be problems with vanishing gradients.

The *reset* gate is used in the model to decide how much of the previous data should be forgotten. It is calculated using the formula Eq. (7):

$$r_t = \sigma\left(W^{(a)} a_t + U^{(r)} h_{t-1}\right) \tag{7}$$

The formula for the *update* gate is the same as the *reset* gate. The gates and weights' usages are the only differences.

Let's now investigate the specific effect of the gates on the output. The *reset* gate is activated first. We introduce new memory content that stores the required historical data in the reset gate. Eq. (8) discusses the formula.

$$h_t = tanh((W * a_t) + (r_t \odot (U * h_{t-1}))) \tag{8}$$

In this case, the multiplication operations with the variables $W * a_t$ and $U * h_{(t-1)}$ were done first, followed by the Hadamard (element-wise) product $(\odot)$ between the reset gate r_t and $U * h_{(t-1)}$. This operation assists the model in determining which data from the earlier steps will not be taken into account going forward. The results of the first two steps will then be combined and given to the non-linear activation function *tanh*.

The last step is for the network to compute h_t, a vector that stores data for the current unit and transmits it throughout the network. This necessitates the *update* gate, which is required. It chooses what should be collected from the preceding steps $(h_{(t-1)})$ and what should be collected from the contents of the current memory (h_t'). Eq. (9) gives a description of the procedure.

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot h_t \tag{9}$$

Here we have performed element-wise multiplication between $z_t \odot h_{(t-1)}$ and $(1 - z_t) \odot h_t$ the summed up their outcomes for the final outcomes.

It uses data from Yahoo! Finance as its input and produces predictions in the form of MSE, RMSE, MAE, and R2-score. First, the NULL values from the dataset were removed in preprocessing. The features, i.e., column values, were normalized from the processed dataset using the MinMaxScaler normalization method. The reason behind doing the normalization is the incomparable range of the feature values, which may lead to the low performance of the model training. The normalized features are divided into two proportions: (i) training: 80% of the total sample and (ii) testing: 20% of the total sample. The training samples were used to train the model. A 125-dimensional output vector space with the *tanh* activation function is created for the LSTM model.

Further, the outcomes of the LSTM model is combined with a dense layer having a single neuron and linear activation function as shown in Algorithm 1. The model is compiled using MAE as a loss function and Adam optimizer. The model is trained using the training data samples. The trained model is employed in prediction. The testing data sample was used to obtain predictions from the trained model. The performance of the model was evaluated with *MSE*, *RMSE*, *MAE*, and $R2 - score$.

IV. Results

This section discussed the experimental outcomes of LSTM, Bi-LSTM, and GRU models in detail. Keras, Tensorflow, and Sklearn libraries were utilised to implement the proposed models. The proposed system is built entirely in Python and runs on a Google Colab platform. The metrics used to evaluate the performances of the experimented models are discussed in Equations (10), (11), and (12), respectively.

Algorithm 1 Proposed Model for Bitcoin Price Prediction

Input: D: Yahoo! Finance Dataset
Output: *Model performance: MSE, RMSE, MAE, & R2-score*
begin
  Data Preparation:;
  $D_c \leftarrow$ *(D) Removed NULL values*;
  $D_N \leftarrow$ *MixMaxScaler*(D_c): *Data normalization in range (0,1)*;
  $D_{Training}(Xtrain, Ytrain) \leftarrow D_N \times 0.80$;
  $D_{Testing}(Xtest, Ytest) \leftarrow D_N \times 0.20$;
  Model Creation:;
  $Model \leftarrow LSTM(units = 125, activation = \text{“}tanh\text{”})$;
  $Model \leftarrow Dense(neurons = 1, activation = \text{“}linear\text{”})(Model)$;
  Model Compilation:;
  $Model.compile(loss = \text{‘}mae', optimizer = \text{‘}adam')$;
  Model Training:;
  $Model.fit(Xtrain, Ytrain)$;
  Model Prediction:;
  $Ypredicted(0-1) \leftarrow Model.predict(Xtest)$;
  $Ypredicted \leftarrow MinMaxScaler.inverse\_transform(Ypredicted(0-1))$;
  Model Evaluation:;
  $MSE \leftarrow MSE(Ytest, Ypredicted)$;
  $MAE \leftarrow MAE(Ytest, Ypredicted)$;
  $RMSE \leftarrow RMSE(Ytest, Ypredicted)$;
  $R2-score \leftarrow R2-score(Ytest, Ypredicted)$;
end

- Root Mean Squared Error (RMSE) [33], [34]:

$$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(Actual_i - Predicted_i)^2} \tag{10}$$

- Mean Absolute Error (MAE):

$$\frac{1}{N}\sum_{i=1}^{N}|Actual_i - Predicted_i| \tag{11}$$

- R2-score:

$$R2 = 1 - \frac{RSS}{TSS} \tag{12}$$

In Eqs. (10) and (11), N represent the number of samples present in the test data, the predicted and actual value of the i^{th} test sample is presented as $Actual_i$ and $Predicted_i$. In Eq. (12), TSS represents the total sum of squares, and RSS represents the residual sum of squares.

Total of six settings of all mentioned models are experimented with by changing the number of neurons at the input layer. However, parameters like an optimizer, loss function, batch size, and others remain the same for all models. The performance of the models is evaluated using RMSE, MAE, and R2-score, which are mathematically defined using the formula mentioned in the Equations (10), (11), and (12), respectively.

The experiment started with a basic LSTM model having 25 neurons at the first layer; the value of performance metrics- MAE, RMSE, and R2- is 374.0088, 918.9602481 0.993590816, respectively, as shown in Table I. The obtained RMSE value indicates the performance of the model is not good. Hence, by increasing the number of neurons from 25 to 50, the LSTM model was re-executed. This time, the value of performance metrics- MAE, RMSE, and R2 are 462.02933, 892.0088635, and 0.993961244. The RMSE value decreased from 918.9602481 to 892.0088635 by increasing the number of neurons at the input layer. This motivated us to further check the model performance by increasing the input neurons of the LSTM model. Hence, another four models are executed by fixing the number of neurons as 75, 100, 125, and 150. The complete outcomes of these models are presented in Table I. The best outcomes in terms of the RMSE values were obtained using the LSTM model having 125 input neurons, where the RMSE value was 409.4120021.

A total of six different settings have been experimented with the LSTM model, and the best RMSE value obtained was 409.4120021, which is high and has scope to minimize it. Therefore, we have opted for another DL model called Bi-LSTM. The Bi-LSTM model was also executed for all six settings as of the LSTM model. The outcomes of the Bi-LSTM model on different settings are presented in Table I. The best outcome of the Bi-LSTM model in terms of RMSE value was 690.4841825, which was poorer than the LSTM model.

Lastly, the GRU model has been experimented for the same. The performance of the GRU model is better than the Bi-LSTM model; however, lower than the LSTM model, as shown in Table I. The best RMSE value of the GRU model was 509.6778394, which was obtained with 125 input neurons.

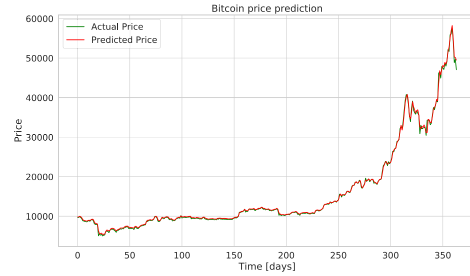

Fig. 5. Plot obtained using LSTM model for actual vs predicted closing bitcoin price prediction.

A. Analysis

This section analyses the performances of the experimented outcomes in terms of losses obtained concerning epochs and the errors in the price prediction concerning the time intervals (days). Firstly, the closing price predicted by the LSTM model was compared with the actual price of Bitcoin. Figure 5 shows the actual vs. predicted plot. The actual and predicted prices are almost the same and have fewer errors. The predicted price was almost matched till 300 days of the data; after that, errors were identified. The curves are slightly separated, indicating that the actual and predicted values differ. The loss obtained during the training of the LSTM model can be seen in Figure 6.

TABLE I
RESULT OF EXPERIMENTED MODELS WITH DIFFERENT SETTINGS

Models	Neurons	MAE	MSE	RMSE	R2
LSTM	25	374.00880	844487.94	918.9602481	0.993590816
	50	462.02933	795679.80	892.0088635	0.993961244
	75	296.76453	284371.25	533.2647091	0.997841784
	100	249.27089	227724.64	477.2050300	0.998271699
	125	253.30370	167618.19	409.4120021	0.998727873
	150	284.89514	220898.60	469.9985040	0.998323505
Bi-LSTM	25	308.35110	476768.40	690.4841825	0.996381599
	50	517.43945	1225689.10	1107.108452	0.990697716
	75	359.29028	519036.34	720.4417699	0.996060809
	100	513.31070	772436.50	878.8836669	0.994137647
	125	462.98816	699355.80	836.2749623	0.994692287
	150	617.02590	1204126.80	1097.327093	0.990861362
GRU	25	875.57160	2577728.80	1605.530676	0.980436504
	50	407.83160	452600.12	672.7556206	0.996565022
	75	434.25244	633291.90	795.7963779	0.995193675
	100	545.90840	919881.75	959.1046606	0.993018620
	125	291.62380	259771.50	509.6778394	0.998028482
	150	479.74030	708876.25	841.9478903	0.994620032

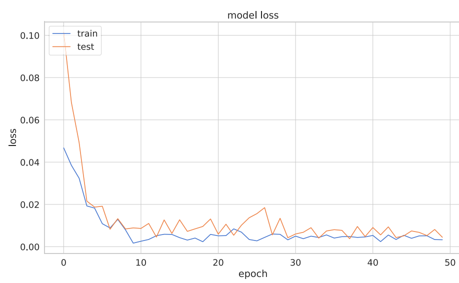

Fig. 6. Plot obtained using LSTM model for the loss vs epoch.

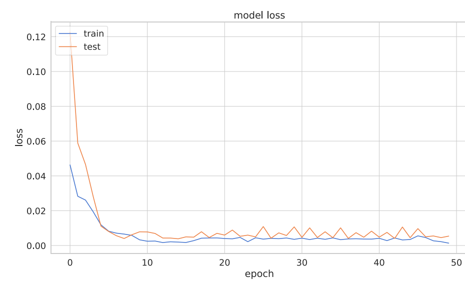

Fig. 8. Plot obtained using Bi-LSTM model for the loss vs epoch.

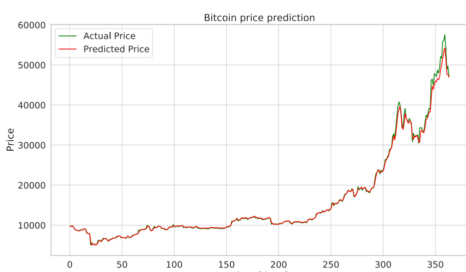

Fig. 7. Plot obtained using Bi-LSTM model for actual vs predicted closing bitcoin price prediction.

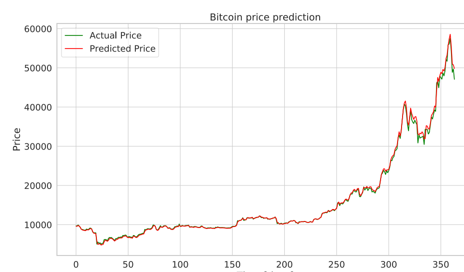

Fig. 9. Plot obtained using GRU model for actual vs predicted closing bitcoin price prediction.

Next, the performance of the other experimented models, i.e., Bi-LSTM and GRU model, was evaluated. The Bi-LSTM model yielded a similar performance as the LSTM model, where the price of Bitcoin was not matched after 300 days and showed minor gaps between them. However, before that, the errors were less, and both the predicted and actual values are together, as shown in Figure 7. The loss obtained during the training of the LSTM model can be seen in Figure 8. Further, the performance of the GRU model was plotted in Figure 9. GRU model performance is also similar to previously implemented LSTM and Bi-LSTM models (Figures 5, 7), where the inconsistency in the price prediction was obtained after 300 days. The training loss obtained on trained and test samples of the GRU model is shown in Figure 10.

Among all experimented models, the LSTM model achieved the best performance by securing minimum losses and other metrics values, confirming that the proposed LSTM is better for building a predictive model with a time-series dataset.

V. DISCUSSION

This research investigates deep learning models such as LSTM, Bi-LSTM, and GRU with different hyperparameter settings to predict the bitcoin price. The performance of these models was recorded in terms of MAE, RMSE, R2, and others.

TABLE II
PERFORMANCE COMPARISON OF THE PROPOSED MODEL WITH EXISTING MODELS

Models		MAE	MSE	RMSE	R2
GRNN [26]		318.2547	210451.16	458.7495613	0.995365456
ANN [35]		410.2545	473814.21	688.3416375	0.996871420
Proposed	Bi-LSTM	308.3511	476768.40	690.4841825	0.996381599
	GRU	407.8316	452600.12	672.7556206	0.996565022
	LSTM	253.3037	167618.19	409.4120021	0.998727873

Fig. 10. Plot obtained using GRU model for the loss vs epoch.

Table II shows the comparative performance of the proposed model with baseline models. The LSTM, Bi-LSTM, and GRU models are treated as baseline models in this research. The LSTM model achieved the best RMSE value of 409.4120021 with 125 units of LSTM cells. The Bi-LSTM model yielded the RMSE score of 690.4841825, with 25 units. The Bi-LSTM achieved a higher RMSE value than the LSTM baseline model for the given case. GRU model also experimented for the same by changing the number of units to 25, 50, 75, 100, 125, and 150. The best outcomes of the GRU model were obtained with 125 units, where the RMSE value was 509.6778394. The GRU model also performed low as compared to the LSTM model. As shown in Table II, the proposed LSTM model outperformed the models proposed by [26], [35] by securing the lowest RMSE value; the number of units used to achieve the performance was 125.

A. Theoretical and Practical Implications

One of the suggested model's theoretical implications is creating a deep learning-based bitcoin price forecasting model that helps worldwide consumer bases and industries decide about investment in digital currency. The model predicts future prices by extracting salient characteristics in their hidden layer for bitcoin price prediction. As a result, the suggested system can handle feature engineering, which is a labour-intensive operation. The suggested model predicts bitcoin price predictions in real-time, allowing practitioners to make informed investments and sells. People are increasingly using cell phones to buy and sell bitcoins. As a result, the current system may be used to create an Android-based application that allows users to receive the future price of bitcoins in real time while on the move. Since this study employed the LSTM model and publically available dataset for bitcoin price prediction, academics and practitioners can use the findings of this study to construct a more robust system for bitcoin price prediction in the future.

VI. CONCLUSION AND FUTURE RESEARCH SCOPE

Forecasting the prices of digital currencies is one of the current challenging tasks, attracting researchers globally to address this issue. This paper experimented with several deep learning models to find the best price prediction model. The DL models- LSTM, Bi-LSTM, and GRU– experimented with different settings. The outcomes of these models are compared with the proposed LSTM model in terms of RMSE and other metrics. The proposed LSTM model's performance outperformed other DL models. The best performing LSTM model outcomes in terms of RMSE are 409.4120021, MAE is 253.3037, and R2 is 0.9987. The RMSE value obtained was 409.4120021, which is high; hence, further improvement is possible.

The future of Bitcoin price prediction systems has a number of exciting possibilities. We can expect the following potential developments: (i) Prediction models can leverage cutting-edge machine learning methods like deep learning, recurrent neural networks, and ensemble approaches to detect complex relationships and trends in Bitcoin price data. (ii) In addition to typical market data, prediction models can also look at other data sources, such as sentiment analysis from news articles and social media. Improving the predictability and interpretability of models is a crucial field of research. As a result, a more trustworthy, understandable AI-based model might be developed in the future.

REFERENCES

[1] A. Urquhart, "The inefficiency of Bitcoin," *Econom. Lett.*, vol. 148, pp. 80–82, Nov. 2016.

[2] A. Singh, A. Kumar, and Z. Akhtar, "Bitcoin price prediction: A deep learning approach," in *Proc. 8th Int. Conf. Signal Process. Integr. Netw. (SPIN)*, 2021, pp. 1053–1058.

[3] S. Nakamoto and A. Bitcoin. "A peer-to-peer electronic cash system." 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[4] J. H. Yu, J. Kang, and S. Park, "Information availability and return volatility in the Bitcoin market: Analyzing differences of user opinion and interest," *Inf. Process. Manag.*, vol. 56, no. 3, pp. 721–732, 2019.

[5] T. Hu et al., "Transaction-based classification and detection approach for Ethereum smart contract," *Inf. Process. Manag.*, vol. 58, no. 2, 2021, Art. no. 102462.

[6] D. G. Baur and T. Dimpfl, "The volatility of Bitcoin and its role as a medium of exchange and a store of value," *Empir. Econ.*, vol. 61, pp. 2663–2683, Nov. 2021.

[7] Y. Sovbetov, "Factors influencing cryptocurrency prices: Evidence from Bitcoin, Ethereum, Dash, Litcoin, and Monero," *J. Econ. Financ. Anal.*, vol. 2, no. 2, pp. 1–27, 2018.

[8] M. F. M. Jalali and H. Heidari, "Predicting changes in Bitcoin price using grey system theory," *Financ. Innov.*, vol. 6, no. 1, pp. 1–12, 2020.

[9] P. Lv, Y. Shu, J. Xu, and Q. Wu, "Modal decomposition-based hybrid model for stock index prediction," *Expert Syst. Appl.*, vol. 202, Sep. 2022, Art. no. 117252.

[10] F. Casino, T. K. Dasaklis, and C. Patsakis, "A systematic literature review of blockchain-based applications: Current status, classification and open issues," *Telematics Informat.*, vol. 36, pp. 55–81, Mar. 2019.

[11] A. Gupta and H. Nain, "Bitcoin price prediction using time series analysis and machine learning techniques," in *Machine Learning for Predictive Analysis* (Lecture Notes in Networks and Systems), vol. 141, A. Joshi, M. Khosravy, and N. Gupta, Eds. Singapore: Springer, 2021, pp. 551–560. [Online]. Available: https://doi.org/10.1007/978-981-15-7106-0_54

[12] P. Kumar Roy and A. Kumar, "Convolutional neural network for text: A stepwise working guidance," in *Proc. Yukthi*, Dec. 2021, pp. 1–6.

[13] A. De Vries, "Cryptocurrencies on the road to sustainability: Ethereum paving the way for Bitcoin," *Patterns*, vol. 4, no. 1, pp. 1–5, 2023.

[14] A. H. Elsayed, G. Gozgor, and C. K. M. Lau, "Risk transmissions between Bitcoin and traditional financial assets during the COVID-19 era: The role of global uncertainties," *Int. Rev. Financ. Anal.*, vol. 81, May 2022, Art. no. 102069.

[15] K. John, M. O'Hara, and F. Saleh, "Bitcoin and beyond," *Annu. Rev. Financ. Econ.*, vol. 14, pp. 95–115, Nov. 2022.

[16] Z. Chen, C. Li, and W. Sun, "Bitcoin price prediction using machine learning: An approach to sample dimension engineering," *J. Comput. Appl. Math.*, vol. 365, Feb. 2020, Art. no. 112395.

[17] H. Rezaei, H. Faaljou, and G. Mansourfar, "Stock price prediction using deep learning and frequency decomposition," *Expert Syst. Appl.*, vol. 169, May 2021, Art. no. 114332.

[18] M. M. Patel, S. Tanwar, R. Gupta, and N. Kumar, "A deep learning-based cryptocurrency price prediction scheme for financial institutions," *J. Inf. Security Appl.*, vol. 55, Dec. 2020, Art. no. 102583.

[19] Z. Gao, Y. He, and E. E. Kuruoglu, "A hybrid model integrating LSTM and garch for Bitcoin price prediction," in *Proc. IEEE 31st Int. Workshop Mach. Learn. Signal Process. (MLSP)*, 2021, pp. 1–6.

[20] C.-H. Wu, C.-C. Lu, Y.-F. Ma, and R.-S. Lu, "A new forecasting framework for Bitcoin price with LSTM," in *Proc. IEEE Int. Conf. Data Mining Workshops (ICDMW)*, 2018, pp. 168–175.

[21] Y. Li, S. Jiang, X. Li, and S. Wang, "Hybrid data decomposition-based deep learning for Bitcoin prediction and algorithm trading," *Financ. Innov.*, vol. 8, p. 31, Apr. 2022.

[22] S. Ji, J. Kim, and H. Im, "A comparative study of Bitcoin price prediction using deep learning," *Mathematics*, vol. 7, no. 10, p. 898, 2019.

[23] S. Tanwar, N. P. Patel, S. N. Patel, J. R. Patel, G. Sharma, and I. E. Davidson, "Deep learning-based cryptocurrency price prediction scheme with inter-dependent relations," *IEEE Access*, vol. 9, pp. 138633–138646, 2021.

[24] H. Kilimci, M. Yıldırım, and Z. H. Kilimci, "The prediction of short-term Bitcoin dollar rate (BTC/USDT) using deep and hybrid deep learning techniques," in *Proc. 5th Int. Symp. Multidiscipl. Stud. Innovat. Technol. (ISMSIT)*, 2021, pp. 633–637.

[25] Y. Li, Z. Zheng, and H.-N. Dai, "Enhancing Bitcoin price fluctuation prediction using attentive LSTM and embedding network," *Appl. Sci.*, vol. 10, no. 14, p. 4872, 2020.

[26] S. Lahmiri and S. Bekiros, "Cryptocurrency forecasting with deep learning chaotic neural networks," *Chaos Solitons Fract.*, vol. 118, pp. 35–40, Jan. 2019.

[27] G. Cheuque Cerda and J. L. Reutter, "Bitcoin price prediction through opinion mining," in *Proc. World Wide Web Conf.*, 2019, pp. 755–762.

[28] K. Mahar, S. Narejo, and M. A. Zaki, "Bitcoin price prediction app using deep learning algorithm," in *Proc. 2nd Int. Conf. Comput. Sci. Technol.*, 2020, pp. 56–60.

[29] T. Awoke, M. Rout, L. Mohanty, and S. C. Satapathy, "Bitcoin price prediction and analysis using deep learning models," in *Communication Software and Networks* (Lecture Notes in Networks and Systems), vol. 134, S. C. Satapathy, V. Bhateja, M. Ramakrishna Murty, N. Gia Nhu, and J. Kotti, Eds. Singapore: Springer, 2021, pp. 631–640. [Online]. Available: https://doi.org/10.1007/978-981-15-5397-4_63

[30] I. A. Hashish, F. Forni, G. Andreotti, T. Facchinetti, and S. Darjani, "A hybrid model for Bitcoin prices prediction using hidden Markov models and optimized LSTM networks," in *Proc. 24th IEEE Int. Conf. Emerg. Technol. Fact. Autom. (ETFA)*, 2019, pp. 721–728.

[31] P. K. Roy, J. P. Singh, and S. Banerjee, "Deep learning to filter SMS spam," *Future Gener. Comput. Syst.*, vol. 102, pp. 524–533, Jan. 2020.

[32] S. Jain and P. K. Roy, "E-commerce review sentiment score prediction considering misspelled words: A deep learning approach," *Electron. Commer. Res.*, pp. 1–25, Jul. 2022. [Online]. Available: https://doi.org/10.1007/s10660-022-095824-

[33] T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)?—Arguments against avoiding RMSE in the literature," *Geosci. Model Develop.*, vol. 7, no. 3, pp. 1247–1250, 2014.

[34] P. K. Roy, "Deep neural network to predict answer votes on community question answering sites," *Neural Process. Lett.*, vol. 53, no. 2, pp. 1633–1646, 2021.

[35] M. Khashei and M. Bijari, "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 2664–2675, 2011.