A Hybrid Approach for Predicting Bitcoin Price Using Bi-LSTM and Bi-RNN Based Neural Network

Sunanda Das(✉), Md. Masum Billah, and Suraiya Akter Mumu

Department of Computer Science and Engineering, Khulna University of Engineering & Technology, Khulna 9203, Bangladesh
sunanda@cse.kuet.ac.bd

Abstract. Bitcoin is an electronic or digital currency. However, unlike government-issued currencies, there is no single entity that issues bitcoin or is in charge of processing transactions. That's why bitcoin has become popular in the recent era. As bitcoin's price fluctuates a lot in a short period, it is very challenging to predict the bitcoin price accurately. In this paper, we proposed a method by merging different highest-level building blocks in deep learning such as Convolutional Neural Networks (CNN), Long Short Term Memory (LSTM), Bi-LSTM, Recurrent Neural Network (RNN), and Bi-RNN to predict bitcoin price as accurately as possible. Though CNN, LSTM, Bi-LSTM, RNN, Bi-RNN or ARIMA independently produce an acceptable result, our proposed hybrid method is fairly reliable compared to individual building block of the network as the proposed method outperforms other individual models and achieves RMSE, MAE, MAPE, MedAE of 2.69%, 1.78%, 2.20%, and 1.23% respectively.

Keywords: Bitcoin · Prediction · Deep learning · CNN · LSTM · Bi-LSTM · RNN · Bi-RNN · Time series forecasting

1 Introduction

Bitcoin is the world's first and most valuable cryptocurrency, which is used worldwide for digital payment or investment purposes. Satoshi Nakamoto, who is a mysterious person or group of individuals, founded bitcoin in 2008 [1]. Bitcoin is a decentralized digital currency that used a blockchain-based network. Bitcoin transaction data is stored on its blockchain, and bitcoin miners generate new coins by adding new transaction data to the blockchain, which tends to keep the network running.

Bitcoin is open-source, its design is public, and no one can control it. Bitcoin is available to anyone who wants to transaction bitcoin and every confirmed transaction is incorporated in the blockchain [2]. Blockchain is a set of blocks where each block is a set of transactions. Bitcoin is a digital file deposited in a

H. Yin et al. (Eds.): IDEAL 2021, LNCS 13113, pp. 223–233, 2021.
https://doi.org/10.1007/978-3-030-91608-4_23

digital wallet similar to a virtual bank deposit. There have many bitcoin wallets for smartphones or computers. People can send or receive bitcoin by using their digital wallets. Bitcoin is not controlled by the government or central banks, or single administrators. Hence, bitcoin can be sent from user to user on the peer-to-peer technology allowing immediate payments without any central authority [3]. A bitcoin transaction is recorded publicly in the blockchain list. All computers running the blockchain have the same list of blocks and transactions, allowing them to see new blocks filled with new bitcoin transactions in real-time. So it is difficult to make any fake transaction of bitcoin, and it is very secure.

In this paper, a Bi-LSTM and Bi-RNN based hybrid model has been proposed, which predicts the closing price of bitcoin in a 1-day interval. For evaluating the performance of the proposed model, a comparison has been made with CNN, LSTM, Bi-LSTM, Bi-RNN, and several other models using different errors such as RMSE, MAE, MAPE, and MedAE.

2 Related Work

As bitcoins are becoming more popular day by day, many researchers have developed several methods to predict the bitcoin price. McNally et al. [4] proposed deep learning-based models such as recurrent neural network (RNN), LSTM for predicting the closing price of bitcoin where the simple moving average (SMA) is used for feature engineering. They applied a temporal window size of 20 days and concluded LSTM network produces better results compared to RNN and ARIMA model.

Ji et al. [5] demonstrated a comparative analysis employing deep learning-based networks such as deep neural network (DNN), LSTM, CNN, deep residual network, and their combinations for forecasting bitcoin price. They shared that for the regression task, i.e., for the prediction of bitcoin price, LSTM networks perform better and DNN-based networks are better suited for predicting the direction of price. Aggarwal et al. [6] investigated socio-economic factors like gold price, Twitter sentiment, and different cryptocurrencies for bitcoin price prediction and proposed root LSTM and gated recurrent unit (GRU) based network. They conferred gold price has less impact on the bitcoin price prediction while Twitter sentiment may give false information about the up-down of price. Also, the LSTM model performs more reliably compared to CNN or GRU-based models.

In [7], Chen et al. analyzed different statistical methods, i.e., logistic regression, linear discriminant analysis, and various machine learning models such as random forest, support vector machine (SVM), LSTM, etc., for bitcoin price prediction. For daily price prediction, they utilized the statistical methods while machine learning-based models are adopted for forecasting the price for 5-minutes intervals. It is observed from their results that for high-frequency data, machine learning models perform better than the statistical methods.

Velankar et al. [8] attempted to predict the daily price of bitcoin using bayesian regression and generalized linear model (GLM). Their study concentrates on finding the optimal features for predicting bitcoin price. However, they didn't provide any evaluations to compare the effectiveness of their models. Datta et al. [9] used the gated recurring unit (GRU) with recurrent dropout for estimating bitcoin price. They consider features that have financial linkage, such as Transaction Fees, Money Supply, US$ Index, etc., in their study. Although recurrent dropout enhances the performance capability of the GRU architecture, further investigation is required to verify its reliability.

3 Methodology

The proposed method is divided into several stages, including dataset collection, normalization, dataset splitting, and so on, as shown in Fig. 1.

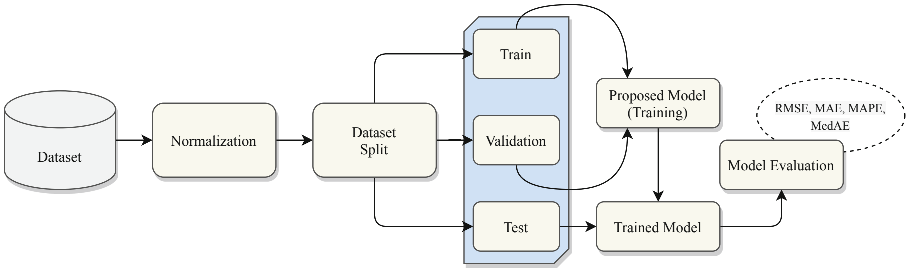


Fig. 1. Work flow of the proposed method.

3.1 Dataset Collection and Description

We collected the Bitcoin dataset from yahoo finance [10]. The dataset consists of seven attributes, namely Date, Open, High, Low, Close, Adj Close, and Volume. We have taken the 'Close' column, which includes bitcoin closing prices in USD from 1st January 2017 to 31st December 2019.

3.2 Normalization

The min-max normalization technique has been applied to the dataset to keep the close price between the range of 0 to 1. To normalize a range between an arbitrary set of values $[l, r]$, the min-max normalization formula is

$$p' = l + \frac{p - min(p)(r - l)}{max(p) - min(p)} \tag{1}$$

Where p' is the normalized price, p is the actual price.

3.3 The Proposed Model

The proposed hybrid model consists of an input layer, 1D-CNN layer, Bi-LSTM, LSTM, Bi-RNN, RNN, and Dense layer that work in a sequential manner. The proposed model is illustrated in Fig. 2 and the summary of the model is given in Table 1.

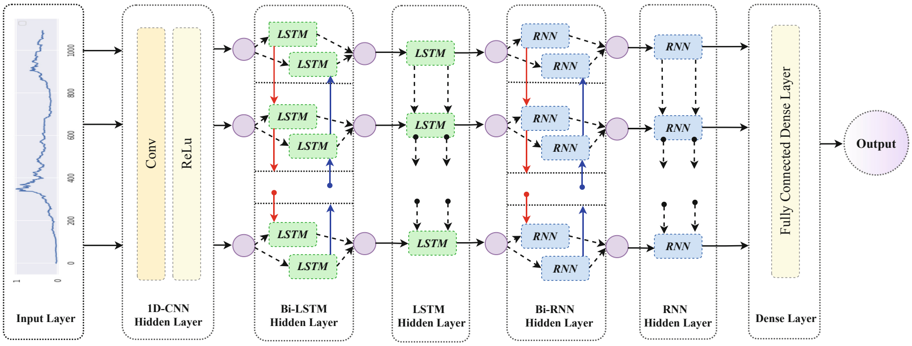


Fig. 2. The proposed model architecture.

Input Layer: The first layer of the model is the input layer which takes the normalized data and passes the information to the hidden layer.

Table 1. Summary of the proposed model.

Layers	Output shape	Parameters
Conv1D	1×32	1,632
Bi-LSTM	1×64	16,640
LSTM	1×32	12,416
Bi-RNN	1×64	4,160
RNN	32	3,104
Dense	1	33
Total parameters		**37,985**

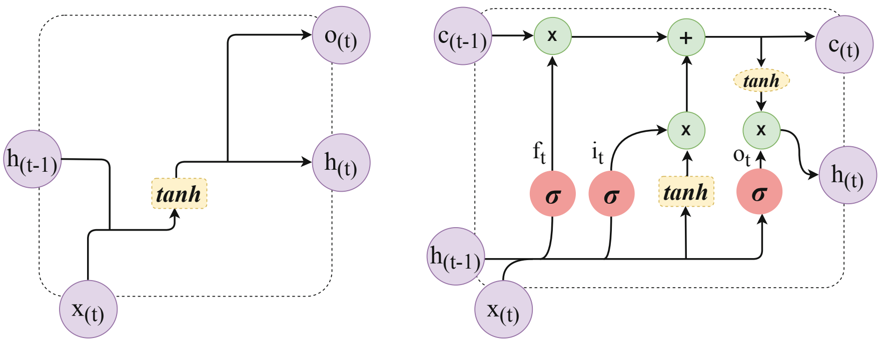
Convolutional Neural Network (CNN): 1D-CNN [11] has shown higher performance in applications with small labeled data and high signal variations gained from various sources. We applied 1D-CNN in the proposed model with Rectified Linear Unit (ReLU) as an activation function that decides when the neurons of the network have to be activated. The mathematical formula for 1D-CNN can be presented as

$$H(j) = \sum_{n=1}^{N_k} I(j+n) \times K(n) \tag{2}$$

where $1 \leq j \leq N_i$, I and H denote the input and output feature map with N_i dimension and K is the convolutional kernel with a dimension size of N_k.

Recurrent Neural Network (RNN): RNN [12] is a type of neural network that makes decisions based on the current input as well as what it has learned from prior inputs. With the help of internal memory, it can loop back the produced output back into the network. RNN follows a precise temporal order when processing inputs that mean the current input is only contextualized by previous inputs, but not by future inputs. The RNN processing chain is duplicated in bidirectional RNN (Bi-RNN) [13], which processes inputs in both forward and reverse time order thus enabling the network to consider the future context as well.

Long Short Term Memory (LSTM): The LSTM layer [14], which is typically a gated Recurrent Neural Network, is quite effective for solving the vanishing and exploding gradient problem. By keeping a more stable error rate, LSTM allows the network to understand several time stages. Again the LSTM contains forget and remember gates, which allow the neuron to determine which information to forget or distribute depending on its significance. The necessary equations for the LSTM network are defined in Eq. (3)–Eq. (8). A bidirectional LSTM (Bi-LSTM) is made up of two LSTMs, one of which takes input in forward direction while the other takes it in a backward direction. The standard architecture of RNN and LSTM are shown in Fig. 3.



(a) RNN **(b)** LSTM

Fig. 3. Standard architecture of RNN, and LSTM.

$$f_t = \sigma(W_f x_t + W_f h_{t-1} + b_f) \tag{3}$$
$$i_t = \sigma(W_i + W_i h_{t-1} + b_i) \tag{4}$$
$$\tilde{c}_t = tanh(W_c x_t + W_c h_{t-1} + b_c) \tag{5}$$
$$o_t = \sigma(W_\circ x_t + W_\circ h_{t-1} + b_\circ) \tag{6}$$
$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t \tag{7}$$
$$h_t = o_t * tanh(c_t) \tag{8}$$

here, f_t symbolises the activation vector for forget gate, x_t is the LSTM unit's input vector, i_t signifies the activation vector of the input/update gate, o_t indicates the activation vector of the output gate, h_t indicates the vector of hidden states, $\tilde{c}_t$ denotes the activation vector for cell input, c_t denotes a vector that represents the current state of a cell. Here W_x denotes weight matrices of respective gate(x), b_x denotes bias vectors for gate(x) and subscript t denotes timestamps.

3.4 Performance Evaluation

To compare the proposed model with other models, we have used four types of errors, i.e., Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Median Absolute Error (MedAE). The formulas for calculating the errors are given below:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - \hat{p}_i)^2} \tag{9}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|p_i - \hat{p}_i| \tag{10}$$

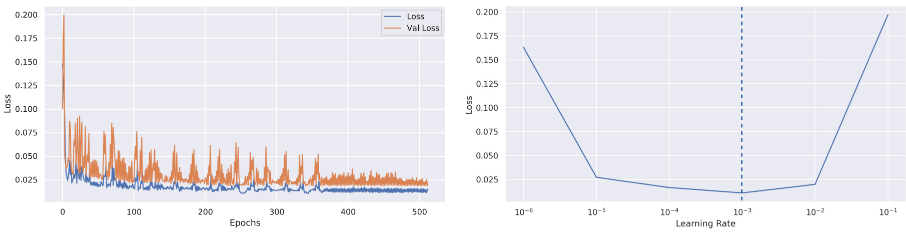
$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|p_i - \hat{p}_i|}{|p_i|} \tag{11}$$

$$MedAE = median(|p_1 - \hat{p}_1|, ..., |p_n - \hat{p}_n|) \tag{12}$$

where p is an actual value, $\hat{p}$ is the predicted value and n is the sample size.

4 Result Analysis

After developing the model, we applied the Adam as the optimizer and Mean Absolute Error as the loss function for compiling the model. The model is trained for 512 epochs with a batch size of 64. The loss vs epoch curve, as shown in Fig. 4a, indicates that the model is free from overfitting. For estimating the best

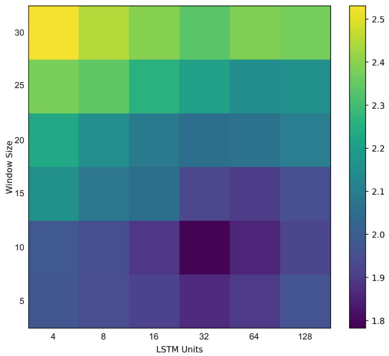


(a) Loss vs Epochs Curve **(b)** Loss vs Learning Rate Curve

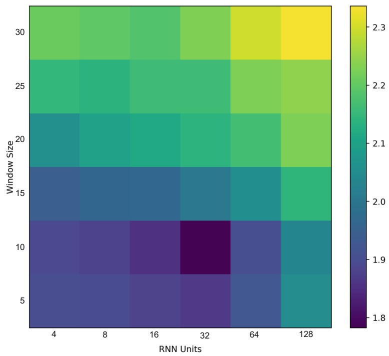
Fig. 4. Loss vs epochs and loss vs learning rate curve of the proposed model.

learning rate for the Adam optimizer, we vary the learning rate and calculate the respective minimum loss of that rate as indicated by Fig. 4b. From the plot, it is apparent that for learning rate 10^{-3}, the model produced the lowest loss.

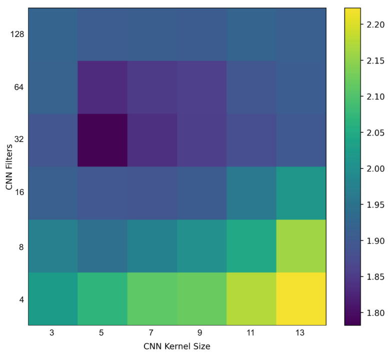
In our proposed model, we vary different hyper-parameters such as window size, CNN filters, CNN kernel size, LSTM units, and RNN units to analyze how MAE dependent on them. The details of this experiment can be found in Fig. 5. If we take a closer look at the plot, according to Fig. 5a and Fig. 5b, the model produces adequate results when the window size varies between 5–10, and both LSTM units and RNN units varies between 16–64. Similarly, we can determine the hyper-parameter ranges from the other subplots. From Fig. 5, it is evident that the value of MAE is lowest when the window size is 10, CNN filters is 32, CNN kernel size is 5, and both the LSTM units and RNN units are 32.



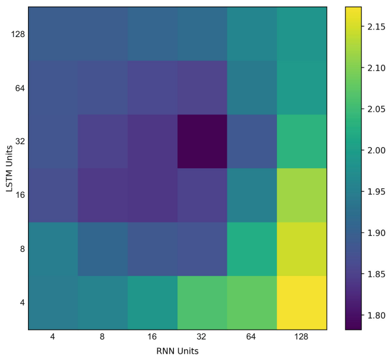
(a) MAE dependent on Window Size and LSTM Units.



(b) MAE dependent on Window Size and RNN Units.



(c) MAE dependent on CNN Filters and CNN Kernel Size.



(d) MAE dependent on LSTM Units and RNN Units.

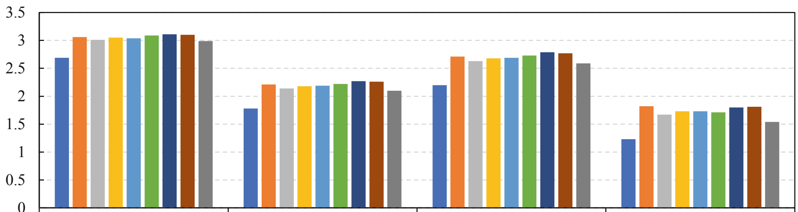
Fig. 5. MAE dependent on hyper-parameters.

The summary of the fine-tuned hyper-parameters used in our proposed hybrid model can be found in Table 2.

Table 2. Hyper-parameters of the proposed model.

Parameters	Value
Window size	10
CNN filters	32
CNN kernel size	5
LSTM units	32
RNN units	32
Learning rate	10^{-3}
Epochs	512
Batch size	64

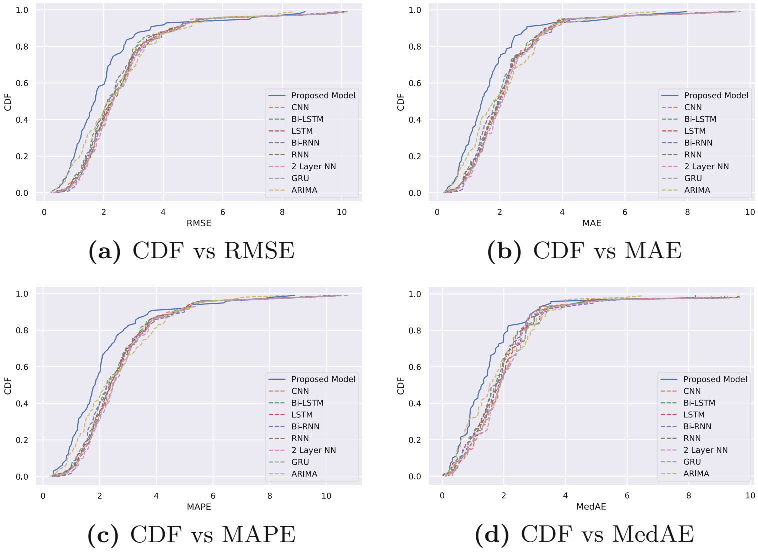
In Fig. 6, a comparative analysis has been illustrated between different models with respect to several errors, i.e., RMSE, MAE, MAPE, and MedAE. From the illustration, it is comprehensible that the proposed hybrid model offers the lowest error in every category compared to other models as the proposed model achieves RMSE, MAE, MAPE, and MedAE of 2.69%, 1.78%, 2.20%, and 1.23% respectively and successfully surpasses the other individual models in performances especially the popular ARIMA model as the ARIMA model delivers the RMSE, MAE, MAPE, and MedAE of 2.99%, 2.10%, 2.59%, and 1.54% respectively.



	RMSE (%)	MAE (%)	MAPE (%)	MedAE (%)
Proposed Model	2.69	1.78	2.2	1.23
CNN	3.06	2.21	2.71	1.82
Bi-LSTM	3.01	2.14	2.63	1.67
LSTM	3.05	2.18	2.68	1.73
Bi-RNN	3.04	2.19	2.69	1.73
RNN	3.09	2.22	2.73	1.71
2 Layer NN	3.11	2.27	2.79	1.8
GRU	3.1	2.26	2.77	1.81
ARIMA	2.99	2.1	2.59	1.54

Fig. 6. Comparative analysis of different models.

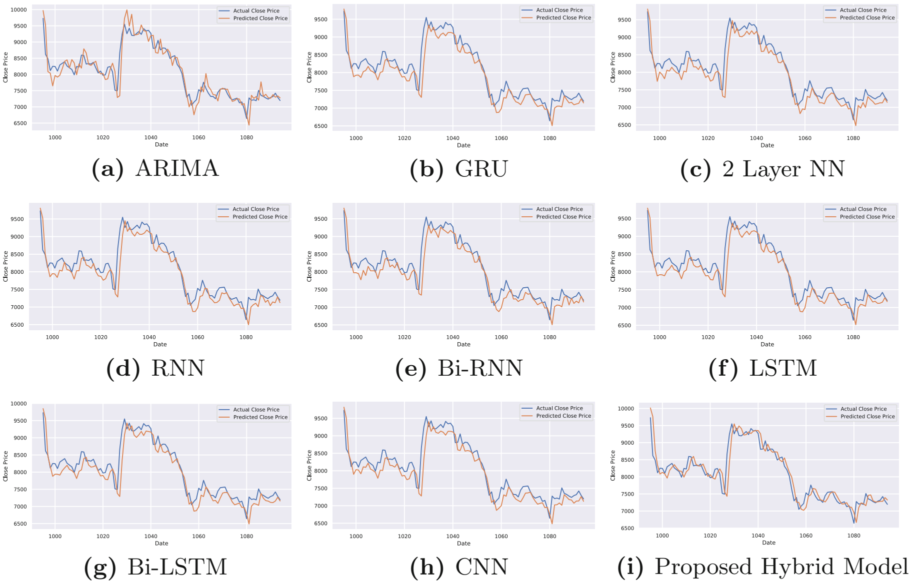
Cumulative distribution function (CDF) plots with respect to the different errors for all models can be inspected in Fig. 7. In comparison to other individual models, the proposed model has a larger area under the curve for each error, as



(a) CDF vs RMSE **(b)** CDF vs MAE

(c) CDF vs MAPE **(d)** CDF vs MedAE

Fig. 7. Analysis of CDF plots for different errors.



(a) ARIMA **(b)** GRU **(c)** 2 Layer NN

(d) RNN **(e)** Bi-RNN **(f)** LSTM

(g) Bi-LSTM **(h)** CNN **(i)** Proposed Hybrid Model

Fig. 8. Prediction curve of various models.

seen in Fig. 7. Since a greater area under the curve indicates that the model has a lesser error, it is understandable that our proposed hybrid model outperforms the other individual models.

The prediction curve for various models is exhibited in Fig. 8 that represents the actual close price and the predicted close price for specific models. The visual illustration verifies our claim that the proposed hybrid model is considerably reliable in predicting bitcoin price compared to other independent models.

5 Conclusion

Because of its recent price surge and crash, bitcoin has experienced a lot of press and public interest. This paper aimed to develop an adequate method for forecasting the price of bitcoin using the hybrid deep learning model while reducing risks for investors and customers. In this paper, 1D CNN, Bi-LSTM, LSTM, Bi-RNN and RNN networks are combined with a dense layer to form a hybrid model. Other state-of-the-art predictive deep learning networks such as CNN, LSTM, RNN, GRU etc. and the traditional ARIMA model were also developed for comparison purposes. The detailed result analysis confirms the reliability and stability of the proposed hybrid model compared to other individual models. As a result, the proposed hybrid model for forecasting bitcoin price would aid consumer growth while also significantly reducing risk for potential investors.

References

1. Satoshi nakamoto. https://www.coindesk.com/people/satoshi-nakamoto
2. How does bitcoin work? https://bitcoin.org/en/how-it-works
3. Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system. Technical report, Manubot (2019)
4. McNally, S., Roche, J., Caton, S.: Predicting the price of bitcoin using machine learning. In: 2018 26th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), pp. 339–343. IEEE (2018)
5. Ji, S., Kim, J., Im, H.: A comparative study of bitcoin price prediction using deep learning. Mathematics **7**(10), 898 (2019)
6. Aggarwal, A., Gupta, I., Garg, N., Goel, A.: Deep learning approach to determine the impact of socio economic factors on bitcoin price prediction. In: 2019 Twelfth International Conference on Contemporary Computing (IC3), pp. 1–5. IEEE (2019)
7. Chen, Z., Li, C., Sun, W.: Bitcoin price prediction using machine learning: an approach to sample dimension engineering. J. Comput. Appl. Math. **365**, 112,395 (2020)
8. Velankar, S., Valecha, S., Maji, S.: Bitcoin price prediction using machine learning. In: 2018 20th International Conference on Advanced Communication Technology (ICACT), pp. 144–147. IEEE (2018)
9. Dutta, A., Kumar, S., Basu, M.: A gated recurrent unit approach to bitcoin price prediction. J. Risk Financ. Manage. **13**(2), 23 (2020)
10. Bitcoin usd (btc-usd). https://finance.yahoo.com/quote/BTC-USD/history?p=BTC-USD

11. Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., Inman, D.J.: 1d convolutional neural networks and applications: a survey. Mech. Syst. Signal Process. **151**, 107,398 (2021)
12. Mikolov, T., Karafiát, M., Burget, L., Černockỳ, J., Khudanpur, S.: Recurrent neural network based language model. In: Eleventh Annual Conference of the International Speech Communication Association (2010)
13. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE Trans. Signal Process. **45**(11), 2673–2681 (1997)
14. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)