E-ISSN: 2676-6167 | P-ISSN: 2538-5100

Journal of Applied Research on Industrial Engineering

www.journal-aprie.com

J. Appl. Res. Ind. Eng. Vol. 9, No. 1 (2022) 134–150.

Paper Type: Research Paper

A Hybrid Model for Predicting Bitcoin Price Using Machine Learning and Metaheuristic Algorithms

Aboosaleh Mohammad Sharifi[1], Kaveh Khalili Damghani[2,*], Farshid Abdi[2], Soheila Sardar[3]

[1] Department of Information Technology Management, North Tehran Branch, Islamic Azad University, Tehran, Iran; sharifi.absh@gmail.com.

[2] Department of Industrial Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran; kaveh.khalili@gmail.com; farshidabdi@azad.ac.ir.

[3] Department of Industrial Management, North Tehran Branch, Islamic Azad University, Tehran, Iran; s_sardar@iau-tnb.ac.ir.

Citation:

Sharifi, A. M., Khalili Damghani, K., Abdi, F., & Sardar, S. (2022). A hybrid model for predicting bitcoin price using machine learning and metaheuristic algorithms. *Journal of applied research on industrial engineering*, *9*(1), 134-150.

Received: 18/06/2021 Reviewed: 19/07/2021 Revised: 09/10/2021 Accepted: 30/10/2021

Abstract

Cryptocurrencies are considered as new financial and economic tools having special and innovative features, among which Bitcoin is the most popular. The contribution of the Bitcoin market continues to grow due to the special nature of Bitcoin. The investors' attention to Bitcoin has increased significantly in recent years due to significant growth in its prices. It is important to create a prediction system which works well for investment management and business strategies due to the high chaos and volatility of Bitcoin prices. In this study, in order to improve predictive accuracy, Bitcoin price dataset is first divided into a time interval through time window, then propose a new model based on Long Short-Term Memory (LSTM) neural networks and Metaheuristic algorithms. Chaotic Dolphin Swarm Optimization algorithm is used to optimize the LSTM. Performance evaluation indicated that the proposed model can have more effective predictions and improve prediction accuracy. In addition, the performance of the optimized model is better and more reliable than other models.

Keywords: Bitcoin, Prediction, Machine learning, Deep learning, Metaheuristic algorithms.

1 | Introduction

Today, cryptocurrencies have become a popular topic in the economics and investment areas [2]. It is considered as a digital or virtual currency which uses encryption technology for its security. Therefore, fraud and forgery are very difficult and is easily detectable and traceable even if they exist, due to this security feature. The organizational nature of the cryptocurrency is considered as another distinctive and certainly most attractive feature, which is not controlled by any government or central authority [3]. Bitcoin is considered as one of the types of cryptocurrencies which has attracted a lot of attention due to its simplicity, transparency and growing popularity. Bitcoin was first designed and introduced in 2008 by Nakamoto [1], then went into the online market in 2009 [2]. Bitcoin was the world's first cryptocurrency which was not well welcomed in the early years; however, it has attracted the attention of many investors, policymakers and the media in recent years.

Corresponding Author: kaveh.khalili@gmail.com

http://dx.doi.org/10.22105/jarie.2021.291175.1343

Not surprisingly, this is because its price has risen from an approximate zero value in 2009 to approximately $ 20,000 in December 2017. This was accompanied by a significant increase in the number of Bitcoins in circulation and the volume of the capital market, which was about 16.8 million Bitcoins and $ 300 billion, respectively [4].

Bitcoin can be transferred through a peer-to-peer network which is managed by Blockchain [5]. Blockchain is actually a growing list of records or blocks that are linked to each other through encryption. Each block contains the hash code of the timestamp of the previous block and the transaction information. Each block has a list of transaction data in which all members are equal and there is no central server to determine what needs to be done. This network enables financial transactions among users [6]. Each record in this chain is encrypted and the username and details of the owner are hidden during the transaction and only the wallet ID is displayed publicly. Bitcoin is the first application of the blockchain, and financial transaction information is stored in its blocks. In addition, each block has a specific capacity, and the financial transactions' data is recorded in a block in chronological order which is completely locked after its completion. Then, the recorded data is shared among all nodes in the network. Therefore, it is impossible to cheat or manipulate it. This is another appropriate feature of this technology for replacing current monetary and currency transfer systems [5]. Bitcoin is attractive in many economies cryptography and computer science fields due to its unique nature of combining cryptographic technology and currency units. Bitcoin will compete with legal instruments such as the US dollar if it is used primarily as a currency to buy goods and services and that will affect its value, and ultimately its monetary policy. However, it will compete with other assets such as government bonds, stocks and commodities if it is mainly used as capital, affecting the financial system [6]. The cryptocurrency market has a relatively short history compared to that of the stock market and has many new features [7]. In addition, it has higher fluctuations and many factors play a role such as hash rate, the relationship between cryptocurrencies, global financial market and public awareness [8]. As a measure for price volatility, these fluctuations have a significant impact on business strategies and investment decisions. Therefore, researchers in the fields of data mining and machine learning are interested in offering new ways to predict the price of Bitcoin and its fluctuations. The traditional and conventional prediction methods are very suitable for cases such as sales price under the influence of seasonal factors. Using traditional methods is not very effective due to the lack of seasonal capability in the cryptocurrency market and their high fluctuations [5]. New algorithms have been designed for prediction along with recent advances in computer computing power, and more importantly, the development of advanced machine learning algorithms and approaches such as deep learning. The prediction using conventional models may be appropriate; however, more advanced methods such as deep neural networks are necessary for higher prediction accuracy.

The structure of the article is as follows. In the second section, the theoretical foundations and the latest related studies are reviewed based on Bitcoin price prediction approaches. Then, the structure of the proposed model is introduced in the third section and then in the fourth section, the implementation steps of the model, the results and its evaluation are stated and finally, the fifth section presents the conclusions.

2 | Theoretical Foundations

2.1 | Long Short-Term Memory

Recurrent Neural Network (RNN) is a generalization of a feedforward neural network that has internal memory. Unlike conventional networks that use different parameters in each layer, an RNN network shares the same parameters between all-time steps. This means that RNNs perform the same operation at each time step, only the inputs are different. With this technique, the total number of parameters that the network must learn is greatly reduced. To predict, if the network cannot learn the relationships between inputs, it certainly cannot predict correctly. For more accurate predictions, we need to look at more information from the past. As this distance increases, the RNN reduces their ability to remember and use the information they have learned in the distant past, in other words, RNN is unable to use more distant past information. Since the RNN's memory and storage capabilities are limited, then it is prone to trapping into gradient vanishing [9].

The LSTM is considered as a type of RNN that has been proposed and improved based on its structure. This type of network is developed specifically to overcome common problems in RNNs. The LSTM network cell has three input gates, forget gate and output gate [10]. *Fig. 1* shows the basic structure of a LSTM network.

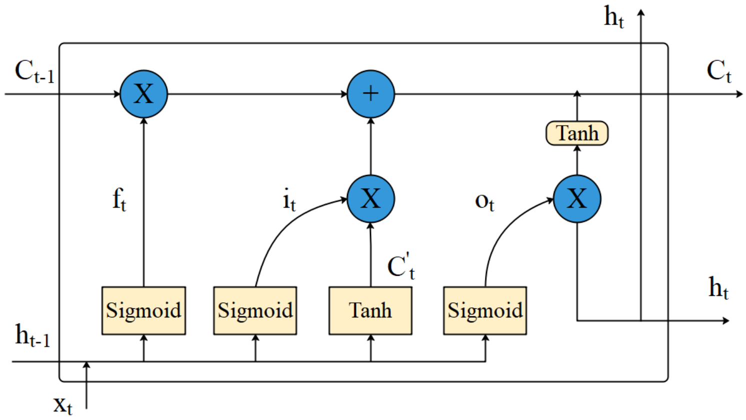

Fig. 1. Structure of a long short-term memory network.

In addition to these three gates, there is a memory cell called C. LSTM can delete or add information to the cell through these gates. The f_t , forget gate are responsible for controlling the flow of information from the previous time step and determines which information is removed from the memory. i_t, input gate is responsible for controlling the flow of new information and identify the information which can be added to memory. The o_t, output gate specifies how much of the previous time step information will be transferred to the next time step with the current time information. In addition, the network utilizes an input of hidden memory h and an input x, and produces two outputs C_t and h_t. The network decides which information in the forget gate is removed from memory at each time steps t due to the memory transfer function. All gates use sigmoid as the activation function and will output a value between 0 and 1. When the gate is open (sigmoid output is 1), all information can pass. But when the gate is closed (sigmoid output is 0), any information cannot pass. The process of calculating the LSTM memory unit is described as follows:

$$
\begin{aligned}
&\mathrm{sigmoid} = \frac{1}{1+e^{-x}}. \\
&f_t = \mathrm{sigmoid}\left(w_f\left[x_t, h_{t-1}\right] + b_f\right). \\
&i_t = \mathrm{sigmoid}\left(w_i\left[x_t, h_{t-1}\right] + b_i\right). \\
&o_t = \mathrm{sigmoid}\left(w_o\left[x_t, h_{t-1}\right] + b_o\right). \\
&C'_t = \tanh\left(w_c\left[x_t, h_{t-1}\right] + b_c\right).
\end{aligned}
\tag{1}
$$

Where w_f, w_i, w_o, and w_c are the weight coefficient matrices of the forget gate, input gate, output gate and neuron state matrix, respectively. b_f, b_i, b_o, and b_c represent the corresponding biases, respectively. The bias can prevent a cell from sticking to zero for output, and it can speed up part of the process by reducing the number of neurons needed to solve the problem. According to the above equations, C_t and h_t are shown based on the following equation:

Sharifi et al.| J. Appl. Res. Ind. Eng. 9(1) (2022) 134-150

$$C_t = f_t C_{t-1} + i_t C'_t. \quad (2)$$
$$h_t = o_t \times \tanh(C_t).$$

This particular type of structure and forward calculation model process enables the LSTM neural network to learn long-term dependencies and is widely used in analyzing and predicting time series data.

2.2 | Chaotic Dolphin Swarm Algorithm (CDSA)

Wu et al. [11] proposed a new metaheuristic algorithm, called the Dolphin Swarm algorithm, through examining some of the dolphin's behaviors inspired by the Particle Swarm Optimization (PSO) algorithm. Dolphins use the sound echo position to explore their surroundings while searching its prey. In addition to using echoes, dolphins work together to catch prey and division of tasks. In addition, they exchange information for this purpose. These behaviors lead to detecting the prey by the dolphin and alerting the other ones. Then all of them surround the prey and hunt it. The Dolphin Swarm algorithm is divided into six stages as initialization, searching, call, reception, predation, and termination stage [12]. The commonality of dolphins hunting collectively is that they circle their prey and tighten the circle over time. Each individual reform its position with that of the other ones. If the prey escapes the dolphin's circle, they reorganize themselves to surround it. They choose prey and the group gradually approaches it, until it can hunt it. This feature is used in the dolphin swarm algorithm to find the optimal solution to the optimization problem. In this algorithm, each dolphin represents a solution to a specific problem and the prey indicates the optimal point. Each Dolphin's position related to its prey determines the chances of its hunting. Like other metaheuristic algorithms, this one has the problem of optimal balance between exploration and exploitation. Qiao and Yang [12] introduced a new edition called the CDSA in 2019 to solve this problem and increase the speed of convergence and the ability to achieve the optimal solution. In this algorithm, each iteration updates all individual dolphins, and selects the current optimal position, repeating the process until the end condition is satisfied. *Fig. 2* briefly shows the algorithm's flow steps.

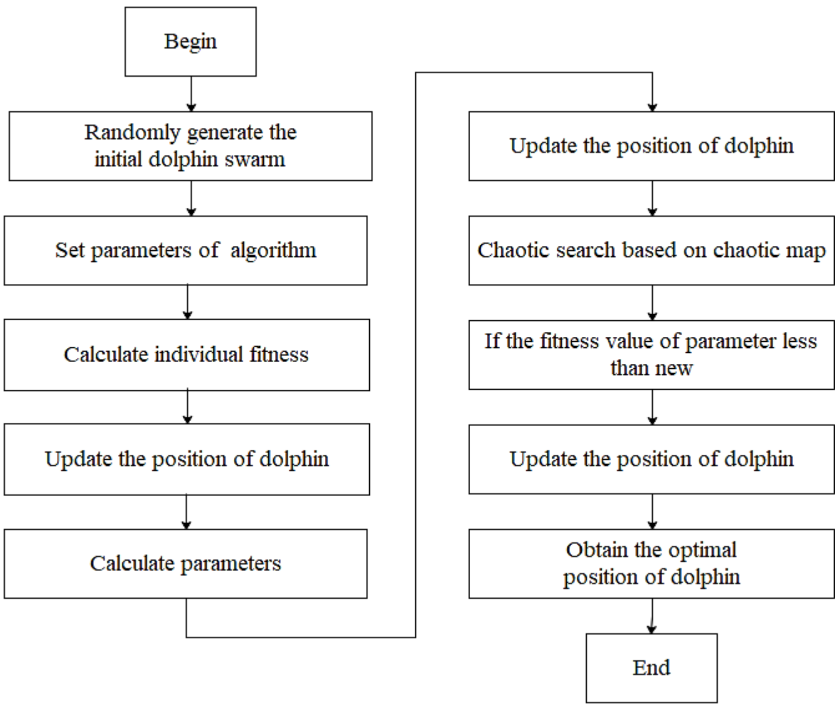

Fig. 2. The flow chart of CDSA [12].

Since the objective function slope is used to move forward of the evolution process of Dolphin Swarm algorithm, the individuals can move quickly toward the optimal solution, leading to rapid and therefore poor convergence of the algorithm. However, some individual may be trapped where the slope of the objective function becomes zero and reduce the search algorithm's efficiency. The CDSA updates the individual position of all dolphins in each iteration and selects the current optimal position and iterates this process until the optimal final condition is met. Therefore, the CDSA can improve the shortcomings of its basic algorithm, resulting in increased efficiency in finding the solution.

2.3 | Multilayer Perceptron

The most widely used neural network architecture is Feed-Forward multilayer networks, which are called MLP. The multilayer perceptron consists of a network of nodes arranged in layers. A typical MLP network consists of three or more layers of processing nodes which include an input layer that receives inputs, one or more hidden layers, and an output layer that produces the classification results [13]. The architecture of MLP is shown in *Fig. 3*.

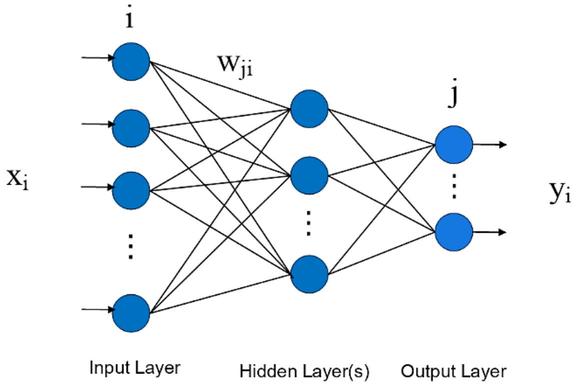

Fig. 3. Architecture of a MLP.

The connections among the layers should be characterized by some weights that are located inside [−1,1]. Each node in the MLP can perform two functions: summation and activation. The product of inputs, weights, and bias are summed using the summation function in the following equations [14].

$$v_j = \sum_{i=1}^{p} w_{ji} x_i + \theta_j. \tag{3}$$

$$y_j = f_j\left(v_j\right).$$

Where v_j is the linear combination of inputs $x_1; x_2; x_3; \ldots; x_p$, θ_j is the bias, w_{ji} is the connection weight between the input x_i and the neuron j, and f_j is the activation function (sigmoid) of the jth neuron, and y_j is the output.

2.4 | Support Vector Machine

The Support Vector Machine (SVM) is one of the machine learning methods based on statistical learning in the 90s by Vapnik et al. The basic principle of the SVM method is finding a separator hyperplane through nonlinear mapping in features to classify data samples in two classes. The SVM method of classification is similar to supervised learning that involves feature extraction and generates desirable outputs and can process large volumes of data robustly, without the occurrence of overfitting. The performance of the SVM classifier depends on three aspects, the penalty parameter C of SVM classifier, the type of the kernel function, and its parameters. SVM in high dimensional spaces and wherever the quantity of dimensions is higher than the number of data Samples, can efficient. Also, SVM uses memory efficiently, and different kernel functions can be used for the decision function [15].

Sharifi et al.| J. Appl. Res. Ind. Eng. 9(1) (2022) 134-150

2.5 | Literature Review

Studies of Bitcoin price prediction using except methods than LSTM are summarized as follows:

Albariqi and Winarko [16] used a multi-layered perceptron neural network and RNN to predict short-term and long-term changes in Bitcoin price. They investigated 1,300 examples of Bitcoin data and found that their model performed more accurately in long-term prediction. Their results show that the multi-layered perceptron neural network model in the long-term period (60 days) has an accuracy of 81.3% and the return neural network in the 56-day period has an accuracy of 77.3%. Hitam et al. [17] proposed an SVM-PSO hybrid model to predict the price of six major cryptocurrencies. They proved that the SVM-PSO performs well compared to the other classifiers with an accuracy of 97%. Their results show that the hybrid model has higher accuracy than the separate models for all cryptocurrencies studied. Also, the SVM and SVM-PSO were more accurate for Ethereum price prediction compared to other cryptocurrencies. Kurbucz [18] used single-hidden layer feedforward neural networks for Bitcoin price prediction. Their findings show this neural network has high accuracy (60.05%) for the price movement classifications compared to other studies such as McNally [38]. For predicting Bitcoin price, Chen et al. [19] compared the ARIMA model with the backpropagation neural network considering the relationships linear and non-linear on the dataset. They used Bitcoin historical data from 2014 to 2018 and presented the results as a combination of different years. Their results show that the RMSE in BPNN is less than ARIMA for all years, especially in 2017. However, in 2014 and 2018 when the price fluctuation was relatively stable, the ARIMA model had a low deviation.

Heo et al. [20] collected and processed cryptocurrency price data through the Application Programming Interface (API). Then, they used a machine learning model called gradient boosting. This method allows the model to learn changes in price data. They found that the maximum model's accuracy is about 63%.

They did not use a deep learning model for prediction and did not provide any results of a comparison between the gradient boosting and deep learning models. Jang and Lee [7] proposed a Bayesian neural network model to predict Bitcoin prices based on Blockchain data on Bitcoin's supply and demand. Their model performs well in predicting the latest Bitcoin price. However, their model is limited to Bitcoin because it is not easy to access blockchain data for other cryptocurrencies. Xiong and Lu [21] have combined the Autoregressive Integrated Moving Average (ARIMA) model with the backpropagation neural network. They implemented this model to predict four stocks in the capital market. They found that the proposed method for the hybrid model has an accuracy of 78.79% and a higher accuracy than that of the non-hybrid model. Katsiampa [22] reported the best model by examining the types of Generalized Autoregressive Conditional Heteroscedasticity (GARCH) models on Bitcoin data. They found that AR-GARCH model was more appropriate for predicting Bitcoin price information.

Radityo et al. [23] studied several multilayer perceptron neural networks predict the Bitcoin's price. Their studies open up new opportunities for in-depth learning related to currency cryptocurrency trends in the future. Sin and Wang [24] provided a combined model based on neural network and genetic algorithm by examining the relationship between Bitcoin features and the price changes in its subsequent day. To better understand its effectiveness and use in the real world, they used an ensemble model to predict the price of Bitcoin the next day. Their results showed that over a period of 50 days, a hybrid model-based trading strategy versus a previous day's trading-based trading strategy showed an 85% return. However, the trading strategy based on the price of the previous day has yielded only 38%. Their model obtained an accuracy of 64% in predicting the trend of price movement. Greaves and Au [25] analyzed Bitcoin's Blockchain network to predict its price through a SVM and an Artificial Neural Network (ANN). Their results indicated a 55% accuracy in the ANN to predict and determine the direction of price change. Madan et al. [26] investigated the features of the Bitcoin network and analyzed its price based on historical data in the intervals of 30, 60 and 120 minutes. They predicted the Bitcoin price situation with an approximation accuracy of 55% in the next 10 minutes through random forest algorithms, SVM and binary Logistic Regression (LR).

Studies using LSTM are summarized as follows:

Chen et al. [27] used two statistical methods, LR and Linear Discriminant Analysis (LDA) and five machine learning models including random forest, XGBoost, Quadratic Discriminant Analysis (QDA), SVM, and LSTM for Bitcoin price prediction. Their results show that the average accuracy of statistical methods (65.0%) was higher than the average accuracy of machine learning models (55.3%) for the daily price of Bitcoin. Also, for Bitcoin 5-minute interval price, the machine learning models have better accuracy than the statistical methods, which was 62.2% and 53.0%, respectively and LSTM obtained the best accuracy (67.2%) compared to other than. Dutta et al. [28] proposed a new framework using a set of advanced machine learning models to predict the daily price of Bitcoin. Their results show that returned neural network models perform better than traditional machine learning models and that GRU neural network architecture performs better than LSTM in data analysis. Yamak et al. [29] compared the performance of various deep learning models with the ARIMA model for predicting Bitcoin price. They used the LSTM and GRU to compare with the traditional model. Their results show that the ARIMA model has better results than others. On the other hand, their research shows that the GRU performs better than the LSTM. Hashish et al. [30] proposed a novel hybrid model using Hidden Markov Model (HMM) and optimized LSTM. Their results showed that the HMM-LSTM has the lowest MSE, RMSE and MAE compare to other models. For example, in the MAE metric value, HMM-LSTM, LSTM and ARIMA show 2.510, 2.652, and 112.060, respectively, which proves that the hybrid model can be more effective for Bitcoin price prediction. Li et al. [31] introduced a combined LSTM-based and Black–Scholes models to predict the Bitcoin price. Their model predicted the future Bitcoin price by examining Bitcoin's data for the next 30 days, indicating that the proposed model reduces the Root Mean Square Error (RMSE) by 46.2%.

To predict the price of several popular cryptocurrencies, Zhengyang et al. [32] used two machine learning models, fully-connected ANN and the LSTM. In general, their results show, ANN outperforms LSTM. Aggarwal et al. [33] used different deep learning models such as a Convolutional Neural Network (CNN), LSTM and GRU for Bitcoin price prediction and effect of Gold price on that. Their results show, all models were effective for Bitcoin price prediction, and LSTM showed the lowest RMSE (47.91) compared to other networks. Felizardo et al. [34] compared various methods such as ARIMA, random forest, SVM, LSTM and WaveNets for predicting the future price of Bitcoin. They considered different prediction windows based on 1, 5, 10 and 30 days. Their results show the MAPE for LSTM in 1day prediction window is better than other. However, the SVM and ARIMA models show a lower error value in almost all prediction windows. Yiying and Yeze [35] used two distinct artificial intelligence frameworks, i.e., fully connected ANN and LSTM, to analyze and predict the Bitcoin, Ethereum, and Ripple prices. They found that ANN and LSTM models are comparable in price prediction, and although they differ in internal structure, they are logically good enough. In addition, LSTM's efficiency is stronger than that of ANN for using useful hidden information in historical memory. Karakoyun and Cibikdiken [36] compared the ARIMA time series model with LSTM in predicting Bitcoin prices and used Mean Absolute Percentage Error (MAPE) to evaluate the model, indicating that MAPE was approximately 11.86% and 1.40% for ARIMA and LSTM, respectively. They found that the LSTM model performed better than the ARIMA. Lv et al. [37] improved the LSTM neural network based on the PSO optimization algorithm to predict the close price. Their model uses PSO to optimize the LSTM neural network weight and reduce predictive error. They found that LSTM performed better in reliability and the improved PSO-LSTM model has better accuracy. McNally et al. [38] implemented the RNN and LSTM neural networks to predict the Bitcoin price and compare it with the ARIMA model. Although the accuracy of all models is almost the same, LSTM has better accuracy than others. However, in RMSE, RNN has better value. According to their results, the LSTM takes considerably longer to train.

Other studies using combined neural network and metaheuristic algorithms for prediction are summarized as follows:

Sharifi et al.| J. Appl. Res. Ind. Eng. 9(1) (2022) 134-150

Jafarian-Namin et al. [39] used Box–Jenkins modeling and the neural network hybrid modeling approaches to forecast wind power generation. They combined ANNs with two common metaheuristic algorithms including genetic algorithm and PSO to predict. The comparison of their results indicated that among the tested ANN model and its improved model, ANN-GA with RMSE of 0.4213 and R2 of 0.9212 gains the best performance in the prediction. Finally, the comparison between ANN-GA and ARIMA methods shows the ARIMA is better performance with RMSE 0.3443 and R2 0.9480 for prediction. Goli et al. [40] provided an integrated framework using statistical tests, time series prediction, and artificial intelligence with the Runner-Root Algorithm (RRA) as a novel metaheuristic algorithm to obtain the best demand for dairy products. They implemented and improved artificial intelligence including MLP, ANFIS, and LSTM using RRA. Then they compared their proposed model with gray wolf optimization, invasive weed optimization, and PSO algorithms. Their results showed that the proposed model has the highest ability to improve using metaheuristic algorithms. Also, RRA showed better results than other tested algorithms. Their accurate results confirm that hybrid methods are able to better predict different subjects. Ansari et al. [41] presented a new method called MOA-PSO, to obtain more accurate results in forecasting by combining two Meta heuristic algorithms, including Magnetic Optimization Algorithm (MOA) and PSO. Their used MOA-PSO for training an ANN to improve its performance in bankruptcy prediction. Finally, they compared the performance of the combined MOA-PSO with four commonly available algorithms. Their study showed that the proposed MOA-PSO hybrid algorithm achieved hopeful results with faster and more accurate predictions, with an accuracy of 99.7. This study also showed that hybrid algorithms are able to solve optimization problems faster and more accurately.

According to the above studies, it can be observed that ANNs are usually used in most of them. One of the biggest advantages of ANNs is the ability to accurately predict with a limited number of layers. However, this property can cause overfitting in the neural network. In this case, the neural network detects a significant amount of unrealistic relationships between the input data, which can lead to false increase accuracy. Also, when using long series datasets with high fluctuations, other methods can cause various problems, for example, forgetting long-term relationships in datasets. On the contrary, the application of LSTM to predict cryptocurrency prices, are still in development. Since LSTM more appropriate for time sequence prediction tasks, therefore, a more accurate framework for price prediction using LSTM can be valuable and contributing. On the other hand, the use of metaheuristic algorithms in the training process of neural networks, can also help in better and more accurate predictions.

3 | Proposed Model

ANNs can approximately obtain the true logarithm of data distribution in the predictive process, however, more complex methods and deep networks are finally needed to achieve high prediction accuracy. This study aimed to provide a framework for predicting Bitcoin prices through this type of network. The proposed model is a combined LSTM and CDSA. The LSTM neural network can learn a complex relationship between features and labels, but the learning process is prone to time, number of hidden layers, the number of nodes per hidden layer and other features. However, CDSA has an appropriate convergence rate and has a significant effect on solving complex optimization problems. Therefore, the CDSA algorithm is used in this model to optimize the learning process of the LSTM to select the appropriate feature. The architecture of the proposed model is shown in *Fig. 4*.

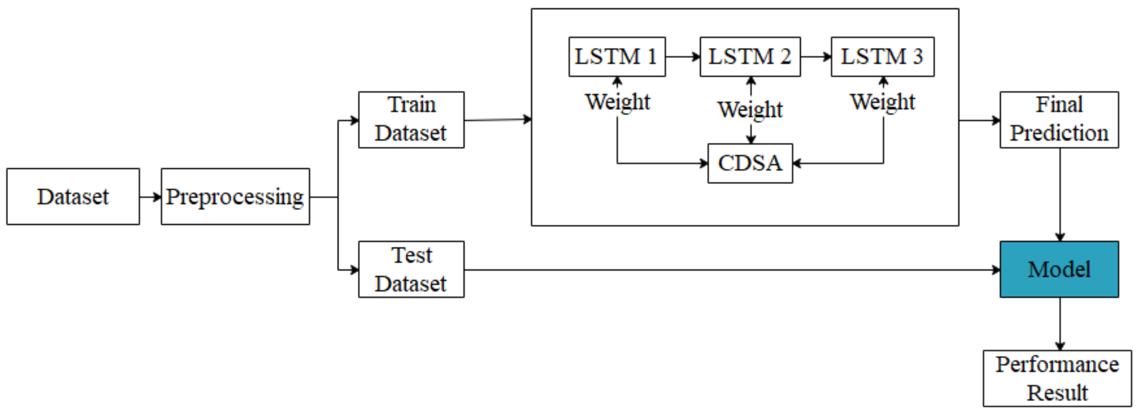

Fig. 4. Framework of proposed model.

As shown in *Fig. 4*, the proposed model consists of three LSTM neural networks called LSTM1, LSTM2, and LSTM3. LSTM1 is trained to extract the pattern between selected inputs and output variables. Then, LSTM1 sends both weights and output variable prediction to the subsequent neural network i.e., LSTM2. All of these networks have a similar structure. Therefore, the weight of each neural network is used by another. The knowledge gained from the prediction process is transferred from LSTM1 to LSTM2 through transferring the weight gained from LSTM1 to LSTM2. In addition, LSTM2 can better learn and predict the behavior of the output variable by sending the LSTM1 output variable prediction to the subsequent neural network since an initial prediction with close correlation with the output variable is transmitted to it as an input. Similarly, LSTM2 transmits the obtained weight and predicts its output variable for LSTM3. Therefore, LSTM3 can more effectively learn the input/output pattern. On the other hand, we can increase the number of neural networks by increasing the cost of computational load, but with an increasing number of networks, the prediction accuracy obtained by the last LSTM network will be negligible. Finally, predict is obtained by LSTM 3 which is the most accurate neural network in the proposed model.

All three neural networks in the proposed model are taught. Although this type of neural network is efficient and fast to learn, the slope of the objective function may become zero and the model may be trapped into local minima. In this method, whenever the evaluation error criterion increases, it is detected that the model trapped into local minima. Therefore, the obtained weights are given to the CDSA optimization algorithm after training each LSTM, to optimize most weights and release the neural network training phase from potential local optimization. The training phase of each neural network is terminated based on the initial stop criterion to prevent over-problems. Then, the results of the training phase, i.e. the weights of the neural network, are transferred to the CDSA algorithm. The algorithm searches the solution space repeatedly until the stopping condition is not met. If the best individual (with the least amount of evaluation error among all individuals) does not change in three consecutive iterations, its search process is terminated. The solution is returned in an optimal weight vector as the final weight of the LSTM1 neural network. At this stage, the training process in the neural network is terminated. This process is iterated for all three LSTM networks. Therefore, if the model is trapped into local minima, the CDSA creates an opportunity to escape from. The proposed model hereinafter called to as CDSA-LSTM.

3.1 | Training Phases of LSTMs

The main idea for combining the CDSA and LSTM is to use the initial connection weights between the LSTM layers and initial thresholds between the neural nodes to found the optimal weights and thresholds of the LSTM with a rapid convergence rate. After that, the initial weights and thresholds obtained by LSTM can be used for training and testing. *Fig. 5* shows the CDSA-LSTM training process flowchart.

Sharifi et al.| J. Appl. Res. Ind. Eng. 9(1) (2022) 134-150

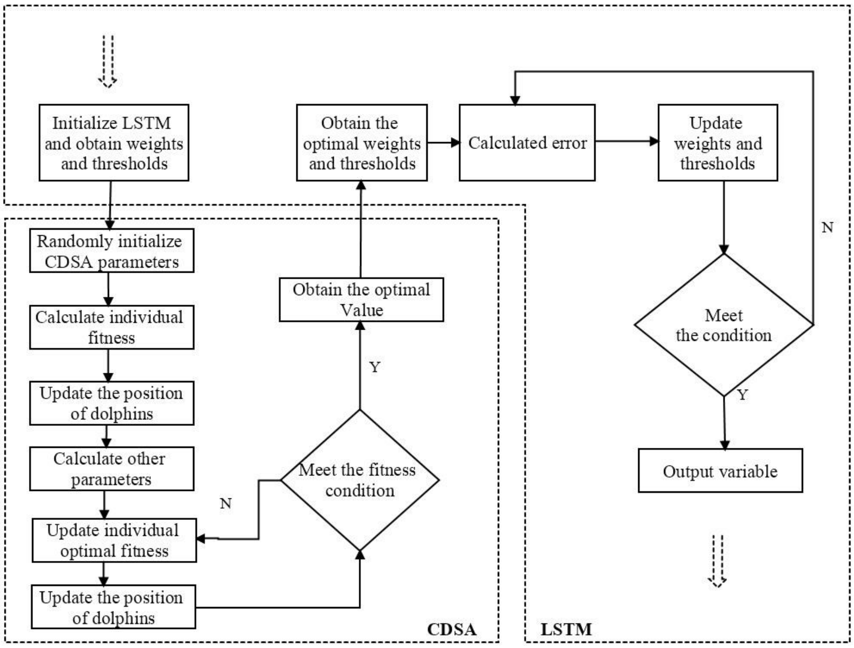

Fig. 5. The CDSA-LSTM algorithm flowchart.

The training process of the LSTMs are as follows:

As *Fig. 5* shows, we first Initialize the LSTM network and obtain the weights and thresholds of the network. We use CDSA to obtain the optimal solution for the initial weights and thresholds. In the next step, we enter the training sample into the LSTM and calculate the network learning error, next, correct the weights and thresholds for connecting the layers using CDSA output optimal values. Finally, we check the error to meet the conditions and whether the number of iterations has reached the set training limit. If both conditions are met, the training ends. Otherwise, the learning process continues. All of these steps are repeated for each LSTM network in the proposed model. The main advantage of the proposed prediction model is that it can predict the Bitcoin price with high accuracy (according to the results and discussion section). The hybrid method preserves the advantages of both techniques and avoids their drawbacks. Also, the relative simplicity of the CDSA algorithm in training LSTMs is one of the most important advantages of using it over other optimization algorithms. On the other hand, using a chain from the LSTM network can achieve more useful results for the time series dataset, because the input of each network is the optimized output by the previous network and finally, the next network was better trained and provided better results.

4 | Implementation

4.1 | Data Description and Preprocessing

Cryptocurrencies are the latest financial innovations, which can be considered as potential risks and opportunities in the financial industry. There are hundreds of cryptocurrencies with various designs in the market. Cryptocurrencies market is segmented based on the market capitalization of cryptocurrencies. For instance, in May of 2017, the top five cryptocurrencies in terms of market capitalization were Bitcoin, Ethereum, XRP, Litecoin, and NEM. This list continuously changes by increasing the number of investors to invest. However, in May of 2020, Bitcoin, Ethereum, Tether, XRP, and Bitcoin Cash were top of market capitalization. High investment in this market, well indicates that investors are looking for cryptocurrencies to make more profit. Therefore, understanding the future status of the market, with detailed analysis and discover risks and opportunities are deemed essential. On the other way, Bitcoin prices are the main driver of the cryptocurrency market, so that Bitcoin transactions alone have grown at nearly 60% per annum over the past 5 years [42].

Fig. 6. Status of train and test data (part of data).

First, the data are divided into smaller time windows to train the model. The window size is determined by the number of sample data within it, and the window step is specified by the number of samples between the ends of two consecutive windows. The prediction process is performed for each window, so it is possible to simulate the prediction process over time. A large portion of the consecutive data is allocated at the beginning of each window for the model's training in each prediction period and the smaller remaining part is used to test the model and calculate the prediction error. In other words, the weighted coefficients are calculated and the neural network model is constructed through the training data in each prediction period and then the model's performance is evaluated by the test's data. This process is repeated in each window and the prediction error is calculated and recorded in each window.

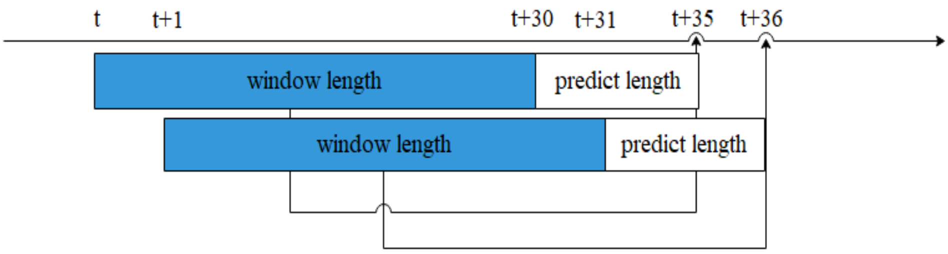

Fig. 7. Window length and predict length settings.

Fig. 7 shows the adjustment of window length and predict length throughout the periods. The window length, rolling window and predict length are set to 30 days, one day and the next five days, respectively.

In this study for preprocessing, we choose open, high, low, close and volume features as inputs (which shows window length) and the close price to output (which shows predict length). The dimension of window length as input data are 29x5. To reduce the noise, we convert all data into logarithmic returns based on the following equation.

$$R_t = \log \frac{P_t}{P_{t-1}}. \tag{4}$$

Where R_t is the return, P_t and P_{t-1} are Bitcoin prices (O,H,L,C) and volume (V) at time t and t-1. Preprocess input and output data are shown in *Fig. 8*.

Sharifi et al.| J. Appl. Res. Ind. Eng. 9(1) (2022) 134-150

$$\begin{bmatrix} \log\frac{O_t}{O_{t-1}}, \log\frac{O_{t+1}}{O_t}, \cdots\cdots, \log\frac{O_{t+28}}{O_{t+27}} \\ \log\frac{H_t}{H_{t-1}}, \log\frac{H_{t+1}}{H_t}, \cdots\cdots, \log\frac{H_{t+28}}{H_{t+27}} \\ \vdots \\ \vdots \\ \log\frac{V_t}{V_{t-1}}, \log\frac{V_{t+1}}{V_t}, \cdots\cdots, \log\frac{V_{t+28}}{V_{t+27}} \end{bmatrix} \rightarrow \begin{bmatrix} \log\frac{C_{t+33}}{C_{t+28}} \end{bmatrix}$$
$$\vdots \qquad\qquad\qquad\qquad \vdots$$

Fig. 8. Logarithmic return based on the window length and predict length.

4.2 | Evaluation Indicator

A confusion matrix is created among the evaluation methods and the results are divided into four categories. The confusion matrix consists of statistics on classifying actual data and the model-generated predictions. The performance of each method is evaluated through the results of this matrix. The rows and columns in the confusion matrix indicate the actual and predicted classes.

Table 1. Confusion matrix.

Confusion Matrix		**Predict**	
		1	**0**
Actual	1	True Positive (TP)	False Negative (FN)
	0	False Positive (FP)	True Negative (TN)

Table 1 indicates that if the label is Actual and prediction is one, the result will be TP. In addition, if the Actual label is one and prediction is zero, the result will be FN, and if the Actual label is zero and the prediction is one, the result will be FP. Further, is the Actual label is zero and the prediction is zero, the result will be TN. Accordingly, the indicators of accuracy, precision, recall and F1-score is calculated as follows:

$$\text{Accuracy} = (\text{tp}+\text{fn}) / (\text{tp}+\text{tn}+\text{fp}+\text{fn}), \tag{5}$$

$$\text{Precision} = \text{tp} / (\text{tp}+\text{fp}),$$

$$\text{Recall} = \text{tp} / (\text{tp}+\text{fn}),$$

$$\text{F1}-\text{score} = 2 \times \text{Precision} \times \text{Re call} / (\text{Precision}+\text{Recall}).$$

4.3 | Parameter Determination

To compare the results, in addition to the CDSA-LSTM, four models have been implemented, including LSTM (3), LSTM, MLP and SVM. As mentioned in Section 3, the proposed model includes a chain of three LSTM networks. In order to better understand the impact of the optimization algorithm, this part of the proposed model was implemented separately. Like the proposed model, the input data is first entered into LSTM1, and after the training and prediction process, the prediction results are sent to LSTM2. Also, LSTM2 performs the corresponding process. The prediction results of LSTM2 (which can be more efficient than the previous output from LSTM1) are sent to LSTM3. The final prediction is obtained from

LSTM3. Finally, the results are evaluated and recorded. The difference between this model and the proposed model is that, the weights of each network are not controlled and optimized by the CDSA.

Therefore, the effect of the CDSA can be observed by comparing the evaluation indicators. Also, LSTM, MLP, and SVM models are implemented separately. In this case, we can more comprehensively compare, the results of the proposed model with other conventional prediction methods.

To implement models, the parameters of each model should be set. *Table 2* indicates the specifications of the most important parameters of models.

Table 2. Parameters and configuration for models.

Model	Main Parameter
LSTM	Epoch=100; Dropout=0.2; Loss=MSE; N=32; Activation=Tanh; Metrics= ACC
CDSA	P=20; Iteration =100; K=20; Speed=2 A=2, e=4, T=20, u=2
MLP	Epoch=100; Hidden Layers=1; N=16; Activation Function= Sigmoid
SVM	C=1.0; Kernel= RBF; Degree=3; Gamma= 0; Coef=0.0; Tol=0.001

According to *Table 2*, for LSTM neural networks, 32 neurons were selected in each layer and the epoch was set at 100. The value of 0.2 was used for the dropout parameter to prevent overfitting and its regulation in the training process. Other parameters were set by default. For the CDSA, the value of 20 was set for the population parameter (p) and optimal neighborhood solution (k) and the value 2 for the speed, respectively. Other parameters were set between 2-20.

The architecture 5-16-1 of the MLP has been chosen. In particular, we have considered 5 input neurons because we have 5 input data and one output neuron because we predict only one Close price. 16 neurons of the hidden layer should be enough to provide predicting. Also, the sigmoid activation function was used in the hidden and output layer. For SVM, parameters C and degree were set to 1.0 and 3, and SVM kernel to RBF and gamma value was determined to 0.

4.4 | Results and Discussion

In the following, the results of the models are presented.

Fig. 9. CDSA-LSTM model prediction (part of data).

Fig. 9 indicates the CDSA-LSTM model prediction and actual data based on the close price in the two-month intervals. The fit of the chart is relatively precise. The overlap of the colored lines indicates the accuracy of the proposed model. The F1-score is used to evaluate the model's efficiency since this criterion represents the relation between precision and recall. The training of the models continues until the best mode with the highest F1-score was obtained in the training data set and reported better performance in the test set. *Table 3* summarizes the performance of all models for predicting Bitcoin's daily price. In addition, the accuracy, precision and recall have been reported. The higher values indicate better performance.

Table 3. Performance of models.

Model	Accuracy	Precision	Recall	F1- Score
CDSA-LSTM	0.87	0.90	0.79	0.84
LSTM (3)	0.71	0.85	0.78	0.82
LSTM	0.63	0.89	0.65	0.75
MLP	0.66	0.64	0.55	0.59
SVM	0.68	0.66	0.56	0.61

The results indicated that all five models tried to train data effectively; however, the CDSA algorithm effectively improves the accuracy of LSTM neural network prediction. As expected, the results of CDSA-LSTM are significantly better than all other models in all evaluation criteria. The F1-score values in LSTM (3) and CDSA-LSTM are 0.82 and 0.84, respectively. On the other hand, the proposed model achieved 0.87 for accuracy, which significantly has improved the prediction. As shown in *Table 3*, the prediction accuracy of the MLP, SVM and LSTM were 0.66, 0.68 and 0.63, respectively. This result show SVM performance was better than LSTM and MLP. In general, the proposed model has improved the prediction against other models.

Table 4 shows compare the results with three best-related studies in the literature. All of our results outperformed the others in both accuracy and precision. In general, the proposed model outperformed the other models, indicating that this model is acceptable for Bitcoin price prediction. Comparing the results indicates, our results in LSTM have higher accuracy than [38] and [27]. Also, MLP shows a lower value than [16] and SVM better than [27].

Table 4. Performance comparison against literature studies.

Model	Accuracy	Precision
CDSA-LSTM	0.87	0.90
LSTM	0.63	0.89
LSTM [37]	0.53	0.36
LSTM [26]	0.57	0.55
MLP	0.66	0.64
MLP [15]	0.81	0.81
SVM	0.68	0.66
SVM [26]	0.65	0.71

Table 5. CPU time of train process.

Model	CPU Time
CDSA-LSTM	2874.18s
LSTM (3)	2117.59s
LSTM	1141s

To evaluate the training of the models, *Table 5* compared CPU time. The CPU utilized was an Intel Core i5 2.6GHz. Due to increased computing, CDSA-LSTM is longer than others, in terms of training time. It can be justified due to the increase in the accuracy of prediction and its effect on the decision of investors in the cryptocurrencies market.

5 | Conclusion

After booming and flourishing the price of cryptocurrencies in recent years, Bitcoin is increasingly considered an asset for investment. Therefore, its price prediction is required for many investors. This study systematically demonstrates the process of application of machine learning algorithms and combined them for predicting bitcoin price. It can be concluded that the whole process can be categorized into four phases: the selection of algorithms, the determination of optimum hyper parameters, and the improvement in model robustness by metaheuristic algorithm, and sensitivity analysis. Our proposed model, based on the LSTM neural network and the CDSA algorithm. We optimized the training process at LSTM using CDSA. It was found that the proposed model can successfully predict the price of Bitcoin with high accuracy. Since CDSA can prevent problems of trapping into local minima, then the hybrid model indicated better results. We compared the results of the proposed model with other models such as LSTM, MLP, SVM and literature studies. It shows that the proposed model has the highest accuracy and F1-score which proved, CDSA-LSTM can be effective in predicting Bitcoin's price. Our results showed that it is possible to achieve better performance by optimizing the training process in deep neural networks. The use of metaheuristic algorithms in this process can lead to improved prediction accuracy. In order to conduct a more comprehensive study of Bitcoin price prediction in the future, it is necessary to collect data at different time intervals. Also, we plan to collect data on other cryptocurrencies in the market and evaluated the performance of the proposed model on it, to provide a suitable model to predict their future price. Finally, for other researchers, it is suggested that other metaheuristic algorithms be used to improve the performance of this model and compare the results. Furthermore, other deep neural networks such as CNN or GRU can also be used in the model and compare with statistical models such as ARIMA and the like. However, it needs to investigate further to enhance the accuracy of the deep learning-based prediction models by considering different parameters in addition to the previous one. Features such as the political system, public relations, and market policy of a country can affect and determine the price volatility of cryptocurrency.

References

[1] Nakamoto, S. (2008). Bitcoin: a peer-to-peer electronic cash system. *Decentralized business review*. https://www.debr.io/article/21260-bitcoin-a-peer-to-peer-electronic-cash-system

[2] Sovbetov, Y. (2018). Factors influencing cryptocurrency prices: Evidence from bitcoin, ethereum, dash, litcoin, and monero. *Journal of economics and financial analysis*, *2*(2), 1-27. DOI: 10.1991/jefa. v2i2.a16

[3] Velankar, S., Valecha, S., & Maji, S. (2018, February). Bitcoin price prediction using machine learning. *20th international conference on advanced communication technology (ICACT)* (pp. 144-147). IEEE. DOI: 10.23919/ICACT.2018.8323676

[4] Atsalakis, G. S., Atsalaki, I. G., Pasiouras, F., & Zopounidis, C. (2019). Bitcoin price forecasting with neuro-fuzzy techniques. *European journal of operational research*, *276*(2), 770-780. https://doi.org/10.1016/j.ejor.2019.01.040

[5] Guo, T., Bifet, A., & Antulov-Fantulin, N. (2018, November). Bitcoin volatility forecasting with a glimpse into buy and sell orders. *2018 IEEE international conference on data mining (ICDM)* (pp. 989-994). IEEE.

[6] Crosby, M., Pattanayak, P., Verma, S., & Kalyanaraman, V. (2016). Blockchain technology: Beyond bitcoin. *Applied innovation*, *2*(6-10), 71. https://j2-capital.com/wp-content/uploads/2017/11/AIR-2016-Blockchain.pdf

[7] Jang, H., & Lee, J. (2017). An empirical study on modeling and prediction of bitcoin prices with bayesian neural networks based on blockchain information. *IEEE access*, *6*, 5427-5437. DOI: 10.1109/ACCESS.2017.2779181

[8] Yang, L., Liu, X. Y., Li, X., & Li, Y. (2019, October). Price prediction of cryptocurrency: an empirical study. *International conference on smart blockchain* (pp. 130-139). Springer, Cham. DOI: 10.1007/978-3-030-34083-4_13

[9] Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European journal of operational research*, *270*(2), 654-669. https://doi.org/10.1016/j.ejor.2017.11.054

[10] Rashid, T. A., Fattah, P., & Awla, D. K. (2018). Using accuracy measure for improving the training of LSTM with metaheuristic algorithms. *Procedia computer science*, *140*, 324-333.

[11] Wu, T. Q., Yao, M., & Yang, J. H. (2016). Dolphin swarm algorithm. *Frontiers of information technology & electronic engineering*, *17*(8), 717-729. DOI: 10.1631/FITEE.1500287

Sharifi et al.| J. Appl. Res. Ind. Eng. 9(1) (2022) 134-150

[12] Qiao, W., & Yang, Z. (2019). Modified dolphin swarm algorithm based on chaotic maps for solving high-dimensional function optimization problems. *IEEE access*, *7*, 110472-110486. DOI: 10.1109/ACCESS.2019.2931910

[13] Ewees, A. A., Abd Elaziz, M., Alameer, Z., Ye, H., & Jianhua, Z. (2020). Improving multilayer perceptron neural network using chaotic grasshopper optimization algorithm to forecast iron ore price volatility. *Resources policy*, *65*, 101555. https://doi.org/10.1016/j.resourpol.2019.101555

[14] Heidari, A. A., Faris, H., Aljarah, I., & Mirjalili, S. (2019). An efficient hybrid multilayer perceptron neural network with grasshopper optimization. *Soft computing*, *23*(17), 7941-7958. https://link.springer.com/article/10.1007/s00500-018-3424-2

[15] Zhou, C., Yin, K., Cao, Y., & Ahmed, B. (2016). Application of time series analysis and PSO–SVM model in predicting the Bazimen landslide in the three gorges reservoir, China. *Engineering geology*, *204*, 108-120. DOI: 10.1016/j.enggeo.2016.02.009

[16] Albariqi, R., & Winarko, E. (2020, February). Prediction of bitcoin price change using neural networks. *2020 international conference on smart technology and applications (ICoSTA)* (pp. 1-4). IEEE. DOI: 10.1109/ICoSTA48221.2020.1570610936

[17] Hitam, N. A., Ismail, A. R., & Saeed, F. (2019). An optimized support vector machine (SVM) based on particle swarm optimization (PSO) for cryptocurrency forecasting. *Procedia computer science*, *163*, 427-433. DOI: 10.1016/j.procs.2019.12.125

[18] Kurbucz, M. T. (2019). Predicting the price of bitcoin by the most frequent edges of its transaction network. *Economics letters*, *184*, 108655. DOI: 10.1016/j.econlet.2019.108655

[19] Chen, C. C., Chang, J. H., Lin, F. C., Hung, J. C., Lin, C. S., & Wang, Y. H. (2019, December). Comparison of forcasting ability between backpropagation network and ARIMA in the prediction of bitcoin price. *2019 international symposium on intelligent signal processing and communication systems (ISPACS)* (pp. 1-2). IEEE. DOI: 10.1109/ISPACS48206.2019.8986297

[20] Heo, J. S., Kwon, D. H., Kim, J. B., Han, Y. H., & An, C. H. (2018). Prediction of cryptocurrency price trend using gradient boosting. *KIPS transactions on software and data engineering*, *7*(10), 387-396. DOI: 10.3745/KTSDE.2018.7.10.387

[21] Xiong, L., & Lu, Y. (2017, April). Hybrid ARIMA-BPNN model for time series prediction of the Chinese stock market. *3rd international conference on information management (ICIM)* (pp. 93-97). IEEE. DOI: 10.1109/INFOMAN.2017.7950353

[22] Katsiampa, P. (2017). Volatility estimation for bitcoin: a comparison of GARCH models. *Economics letters*, *158*, 3-6. DOI: 10.1016/j.econlet.2017.06.023

[23] Radityo, A., Munajat, Q., & Budi, I. (2017, October). Prediction of bitcoin exchange rate to American dollar using artificial neural network methods. *International conference on advanced computer science and information systems (ICACSIS)* (pp. 433-438). IEEE. DOI: 10.1109/ICACSIS.2017.8355070

[24] Sin, E., & Wang, L. (2017, July). Bitcoin price prediction using ensembles of neural networks. *13th international conference on natural computation, fuzzy systems and knowledge discovery (ICNC-FSKD)* (pp. 666-671). IEEE. DOI: 10.1109/FSKD.2017.8393351

[25] Greaves, A., & Au, B. (2015). *Using the bitcoin transaction graph to predict the price of bitcoin*. Retrieved from http://snap.stanford.edu/class/cs224w-2015/projects_2015/Using_the_Bitcoin_Transaction_Graph_to_Predict_the_Price_of_Bitcoin.pdf

[26] Madan, I., Saluja, S., & Zhao, A. (2015). *Automated bitcoin trading via machine learning algorithms*. Retrieved from http://cs229.stanford.edu/proj2014/Isaac%20Madan,%20Shaurya%20Saluja,%20Aojia%20Zhao,Automated%20Bitcoin%20Trading%20via%20Machine%20Learning%20Algorithms.pdf

[27] Chen, Zh., Li, C., & Sun, W. (2020). Bitcoin price prediction using machine learning: an approach to sample dimension engineering. *Journal of computational and applied mathematics*, *365*, 112395. DOI: 10.1016/j.cam.2019.112395

[28] Dutta, A., Kumar, S., & Basu, M. (2020). A gated recurrent unit approach to bitcoin price prediction. *Journal of risk and financial management*, *13*(2), 23. DOI: 10.3390/jrfm13020023

[29] Yamak, P. T., Yujian, L., & Gadosey, P. K. (2019, December). A comparison between arima, lstm, and gru for time series forecasting. *Proceedings of the 2nd international conference on algorithms, computing and artificial intelligence* (pp. 49-55). *Association for computing machinery*. New York, NY, United States. DOI: 10.1145/3377713.3377722

[30] Hashish, I. A., Forni, F., Andreotti, G., Facchinetti, T., & Darjani, S. (2019, September). A hybrid model for bitcoin prices prediction using hidden Markov models and optimized LSTM networks. *24th IEEE international conference on emerging technologies and factory automation (ETFA)* (pp. 721-728). IEEE. DOI: 10.1109/ETFA.2019.8869094

[31] Li, L., Arab, A., Liu, J., Liu, J., & Han, Z. (2019, July). Bitcoin options pricing using LSTM-based prediction model and blockchain statistics. *2019 IEEE international conference on Blockchain (Blockchain)* (pp. 67-74). IEEE. DOI: 10.1109/Blockchain.2019.00018

[32] Zhengyang, W., Xingzhou, L., Jinjin, R., & Jiaqing, K. (2019, June). Prediction of cryptocurrency price dynamics with multiple machine learning techniques. *Proceedings of the 4th international conference on machine learning technologies* (pp. 15-19). Association for Computing Machinery, New York, NY, United States. DOI: 10.1145/3340997.3341008

[33] Aggarwal, A., Gupta, I., Garg, N., & Goel, A. (2019, August). Deep learning approach to determine the impact of socio economic factors on bitcoin price prediction. *2019 twelfth international conference on contemporary computing (IC3)* (pp. 1-5). IEEE. DOI: 10.1109/IC3.2019.8844928

[34] Felizardo, L., Oliveira, R., Del-Moral-Hernandez, E., & Cozman, F. (2019, October). Comparative study of Bitcoin price prediction using wavenets, recurrent neural networks and other machine learning methods. *6th international conference on behavioral, economic and socio-cultural computing (BESC)* (pp. 1-6). IEEE. DOI: 10.1109/BESC48373.2019.8963009

[35] Yiying, W., & Yeze, Z. (2019, March). Cryptocurrency price analysis with artificial intelligence. *5th international conference on information management (ICIM)* (pp. 97-101). IEEE. DOI: 10.1109/INFOMAN.2019.8714700

[36] Karakoyun, E. S., & Cibikdiken, A. O. (2018, May). Comparison of arima time series model and lstm deep learning algorithm for bitcoin price forecasting. *The 13th multidisciplinary academic conference in Prague* (Vol. 2018, pp. 171-180). MAC Prague Consulting s.r. o.

[37] Lv, L., Kong, W., Qi, J., & Zhang, J. (2018). An improved long short-term memory neural network for stock forecast. *2nd international conference on electronic information technology and computer engineering (EITCE 2018)* (Vol. 232, p. 01024). EDP Sciences. DOI: 10.1051/matecconf/201823201024

[38] McNally, S., Roche, J., & Caton, S. (2018, March). Predicting the price of bitcoin using machine learning. *26th euromicro international conference on parallel, distributed and network-based processing (PDP)* (pp. 339-343). IEEE. DOI: 10.1109/PDP2018.2018.00060

[39] Jafarian-Namin, S., Goli, A., Qolipour, M., Mostafaeipour, A., & Golmohammadi, A. M. (2019). Forecasting the wind power generation using Box–Jenkins and hybrid artificial intelligence: a case study. *International journal of energy sector management*, *13*(4), 1038-1062. DOI: 10.1108/IJESM-06-2018-0002

[40] Goli, A., Moeini, E., Shafiee, A. M., Zamani, M., & Touti, E. (2020). Application of improved artificial intelligence with runner-root meta-heuristic algorithm for dairy products industry: a case study. *International journal on artificial intelligence tools*, *29*(05), 2050008. https://doi.org/10.1142/S0218213020500086

[41] Ansari, A., Ahmad, I. S., Bakar, A. A., & Yaakub, M. R. (2020). A hybrid metaheuristic method in training artificial neural network for bankruptcy prediction. *IEEE access*, *8*, 176640-176650. DOI: 10.1109/ACCESS.2020.3026529

[42] Saiedi, E., Broström, A., & Ruiz, F. (2021). Global drivers of cryptocurrency infrastructure adoption. *Small business economics*, *57*(1), 353-406. DOI: 10.1007/s11187-019-00309-8