Ilkogretim Online - Elementary Education Online, 2021; Vol 20 (Issue 3): pp. 4264-4272
http://ilkogretim-online.org
doi: 10.17051/ilkonline.2021.03.440

Deep Learning Approach For Predictive Analysis Of Crypto-Currency Price+

NOOR MOHD Department of Computer Science & Engineering, Graphic Era Deemed to be University, Dehradun, Uttarakhand India, 248002
noormohdcs@gmail.com

Vijay Singh Department of Computer Science & Engineering, Graphic Era Deemed to be University, Dehradun, Uttarakhand India, 248002
vijaysingh.cse@geu.ac.in

Poonam Verma Department of Computer Application Graphic Era Hill University, Dehradun, Uttarakhand India, 248002 pverma@gehu.ac.in

Abstract

Because cryptocurrencies are now decentralised, the degree of centralised control that they are susceptible to has significantly diminished. This has had an effect on the sector as well as on foreign relations. The bitcoin market is characterised by extremely volatile prices, making it urgently necessary to devise a mechanism that can accurately measure these prices. The long short-term memory (LSTM) and recurrent neural networks (RNN), both of whom are effective instructional models for testing phase and the LSTM is good at spotting longer-term contacts, are used in this investigation to offer novel technique for predicting the price of bitcoin by that take into account a number of features including selling price, amounts, reactor supply, and optimum supply. This research attempts to generate a method to predict the price movement by making use of these features. The methodology that has been suggested is implemented in Python and tested using sample datasets. The findings indicate that the strategy recommended for accurately predicting the price of bitcoin is a viable option.

Introduction

Time series prediction well-known issue to need help predicting or forecasting time series. Foreseeing markets like the stock market has been the subject of extensive research [1,2]. The concept of cryptocurrencies, which may be viewed as a type of virtual money meant to be used as a trade means, is intriguing because it can be viewed as a time series prediction problem. This issue is still in its early stages. As a result, the market is highly volatile [3], which presents opportunities for additional research on the cryptocurrency price prediction.

Additionally, the adoption of cryptocurrencies like Bitcoin is growing globally. Due to the cryptocurrency's open nature runs on a decentralized, peer-to-peer, and untraceable system that ensures that all transactions are recorded on the blockchain, an open ledger. In the world of traditional financial markets, there is no such transparency. Traditional methods for solving time series prediction problems, like Holt-Winters exponential smoothing, rely on linear presumptions. These methods call for segmenting data input into patterns [4] and are better suited for forecasting variables with seasonal impacts, such as sales. Since cryptocurrencies are highly volatile and have no seasonal effect, these methods could not help predict the price of cryptocurrencies. Given the intricacy of these issues, the deep learning methodology has grown in favour due to its success in resolving issues like naturally occurring language processing [5]. RNN and LSTM are quite well techniques in the field of machine learning. Due to their temporal nature, these methods could be superior to the established multilayer perceptron (MLP) [6]. Given the nonlinearities of cryptocurrency prices, this study suggests a framework for predicting pricing using deep learning techniques. The aspects of deep learning techniques are discussed in the section that follows.

Both reinforcement learning and semi-supervised learning fall under the umbrella of machine learning. Unsupervised learning lacks this requirement, whereas supervised learning involves modelling datasets with labelled examples. Each instance in supervised learning can be characterized as a set of characteristics and target classes. Target classes are assigned to these attributes. Supervised techniques include support vector machines and neural networks. Classification tasks are where the autoencoder (MLP), a straightforward feed-forward neural network, is most frequently applied. Examples supplied into the model are inputs in the context of neural networks, while predicted values are referred to as outputs. A layer is each modular subfunction. Inlet and outlet layers make up a model, and the hidden layers that lie in between are referred to as such. Each unit that emerges from one of these levels can be compared to neurons in the brain. The weight, comparable to a synapse in the brain, is the term for the connections between these units. Since the load is the property that is altered during model training, it establishes the purpose of the model. However, the vanishing-gradient problem restricts the MLP's usefulness. Derivatives here are vulnerable to exploding or disappearing gradients because the system's levels and time series are multiplicatively coupled to one another.

In contrast to bursting gradients, disappearing gradients are more problematic since the network may only be able to learn from them after they vanish. The MLP also has the drawback of having a network's static signals in the future. Its memory might be thought frozen in time, preventing it from effectively recognizing the temporal part of a time series task. The MLP can be seen as treating all inputs as a collection of items with no particular chronological sequence. As a result, all incoming data is given the same weight, which is a misguided strategy. Some of these constraints are addressed by the RNN, also called a dynamic neural network [6]. Although the construction of the RNN and the MLP are identical, the RNN allows for iterative forward and backward signalling. The context layer, a new layer introduced to the system, makes this easier. In addition to sending inputs across levels, each layer feeds its output to the context layer, which then feeds it into the subsequent layer along with the subsequent input. Every timestep in this situation overwrites the state. This advantage is that, unlike the MLP, it enables the network to provide specific weighting to events that happen in a sequence instead of all inputs having the same weight. As a result, the network becomes dynamic. The size of the network memory can be compared to the size of the temporal window. It is a suitable technique for a problem requiring time series prediction [5, 7]. Even though this fixes the temporal problem in a time series work, diminishing gradients can still be problematic.

In addition, despite the fact that the RNN is capable of handling long-term dependencies, a number of studies [8, 9] have demonstrated that, in actuality, it usually requires assistance in order to train because of backpropagation and long-term dependencies. Residual blocks [10] are able to address each of these difficulties well. They make it possible for weights to be maintained as they go across both forward and backward layers. In contrast to the RNN, which constantly modifies the state by replacing it with new data, this. LSTM units enables the infrastructure to keep learning across a large number of time steps while still maintaining a consistent error rate. This is made possible by maintaining the error rate. As a consequence of this, the network is able to identify long-term dependencies.

Forget/remember gates in an LSTM cell provide the cell the ability to choose which data to block or pass through by taking into account the importance and weight of the information being processed. As a result, it has the power to suppress strong signals and prevent the gradient from decreasing. The LSTM cell states are dependent on these three factors, which can be generalised as historic substantial number, prior concealed states, and current time steps respectively. The manipulation of this memory requires special gateways, and the variables that are manipulated are the ones that are accountable in memorizing. The output, input, and forget barriers are all included in these gates. Forget gates, as their title suggests, clear out any data that the LSTM does not require in order to function properly.

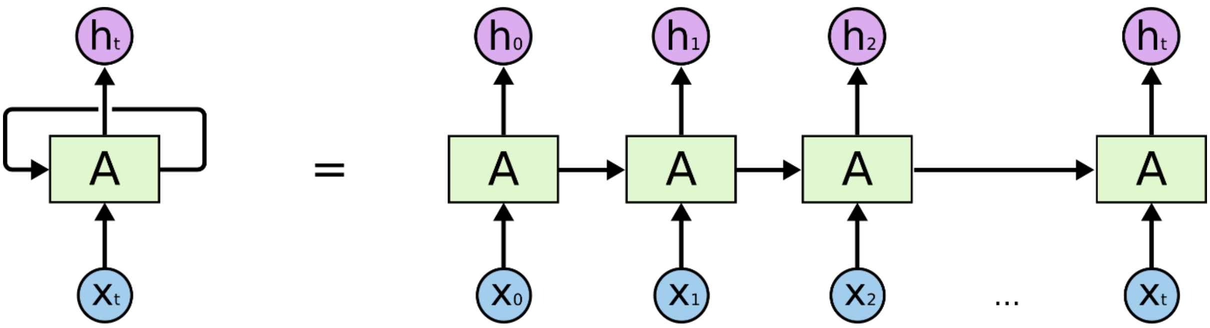

Figure 1. LSTM Process

The input gate is used for every new piece of information that is added to the cell state. The input gate uses the tanh function to produce an output range of -1 to +1 as shown in figure 1. The input gate ensures that only the most crucial information is present and that any unnecessary information is eliminated. The output gate's primary responsibility is to choose the most valuable information from the convolution layer and show it. Models like

ARIMA rely on linear data presumptions. These models might not produce helpful findings due to the bitcoin price's very nonlinear nature. In light of the highly nonlinear cryptocurrency prices and the benefits of deep learning algorithms, this research suggests using LSTM models, a subset of deep learning algorithms, to forecast cryptocurrency prices..

Literature review

A technological study of decentralized digital currency was carried out by Tschorsch and Scheuermann [11]. They focused on important issues, including centralized cryptocurrency transactions, the proof of work, blockchain-based transactions, scripts, epilogue, and security, as they looked at the fundamentals and protocols relating to Bitcoin as a representative cryptocurrency. They explained every technical aspect of the cryptocurrency (also known as distributed currencies), and these illuminating conclusions may be applied to other cryptocurrencies to understand better how they operate. An overview of cryptocurrency systems was also published by Mukhopadhyay et al. [12]. They discussed several facets of cryptocurrencies, such as the proof of stake and the proof of pork and how to combine them for data mining approaches. They emphasized that, in contrast to the proof of work, which is resource-dependent, the proof of stake is insufficient to act freely. Therefore, combining these factors can produce more accurate outcomes.

Additionally, they emphasized how CPU and memory expensive the majority of cryptographic techniques are. Using data from social media and epidemic modelling, Phillips and Gorse [13] suggested predicting bitcoin price bubbles. They demonstrated how to forecast cryptocurrency prices using social media data and epidemic modelling. They concluded that social media data could be crucial in predicting cryptocurrency movements after using the hidden Markov model (HMM) to find bubble-like tendencies in the time series. Deng et al. [14] suggested a deep direct, reinforced learning system for the encoding and trading financial signals. They concentrated on teaching the computer to outperform seasoned financial traders at accurately predicting trading outcomes. They suggested combining deep learning, recurrent deep neural networks, and reinforcement learning to produce accurate predictions. They used data from the stock market and commodity futures markets to validate the suggested methodology. To predict crude oil prices, Zhao et al. [15] proposed a deep learning ensemble method termed stacked denoising autoencoders (SDAE). They employed a deep learning and ensemble learning-based model to predict the crude oil price.

Deep neural networks, which are on the leading edge of technology, were utilised in order to ascertain how various aspects that decide this price relate to one another. To validate the suggested approach, they examined 198 independent variable and came to the conclusion that a stuffing operational planning with the SDAE produces superior findings in terms of a precision of the expected crude oil price. This was the conclusion reached after they assessed the suggested approach. They ran a statistical analysis to provide additional confirmation of the results. Shehhi et al. [16] investigated the factors that should be considered while choosing a cryptocurrency. They investigated two distinct categories of issues that are associated with cryptocurrencies: first, they investigated the many variables that motivate customers to use cryptocurrency mining, and second, they investigated the factors that influence the value and acceptability of cryptocurrencies. They researched eight distinct forms of cryptocurrencies and put out an online questionnaire. They came to the conclusion that the coin's name and emblem are the most important factors in determining its level of adoption.

Proposed System

This section outlines the suggested method for utilizing deep learning to anticipate cryptocurrency prices. The suggested method's workflow entails four LSTM input layer units for modelling and an activated sigmoid for managing information flow and remembering all patterns discovered in cryptocurrency data. In the section, it is also suggested that network weights be iteratively updated for training purposes using the Adam optimizer. To improve the model's accuracy, the dense layer is passed. The four LSTM layers incorporated into the suggested method make the model better suited for acquiring higher-level representations. The dense layer reduces the input sequence to a single vector while the LSTM stages return their complete output sequences. In this method, the relationship between the distinctive fields in the data set is determined using the square of the Pearson correlation. This makes it easier for the dominant parameters to determine values for the other variables. The price of cryptocurrencies is then calculated using linear and exponential forecasting. The suggested method forecasts bitcoin prices using the LSTM model. The following are descriptions of the various steps of the suggested strategy:

Step 1: Data Analysis This phase processes the information and its characteristics to look for redundant data values, which could impact the accuracy of the predictions. Unimportant parameters are deleted from datasets if they are present. This phase also analyses data for potential data merging to increase the model's predictability.

Step 2. This phase filters the data to eliminate redundant or empty values.

Step 3: Train-Test Split: Data is divided into subgroups for training and testing in this phase. As an illustration, data are split 70% training examples and 30% testing dataset.

Step 4. This phase involves scaling the raw information to meet the model's needs before providing it to the model. In this manner, the data are reshaped during this phase to fit the model better.

Step 5. Model-Building Phase: Python is used to implement the suggested strategy. Theano and tensorflow are the two most potent libraries for Python when it comes to machine learning models. However, using these libraries to create a model takes time and effort. Keras with TensorFlow is employed as the blackbox library to increase the model's accuracy. The LSTM and dense layers are the two layers that make up the Keras sequential model. These layers do in-depth data processing to examine all patterns that have emerged in the dataset and increase the model's accuracy. The data are then sent to the model so it can be trained.

Step 6:Data is trained using a variety of LSTM units in the sixth phase of the model learning and evaluation. The recall cell, input gate, output gate, and forget gate are the four gates that make up this. To allow information to pass through, these gates are used. They are made up of activation layers like a sigmoid, which produces outputs of values between 0 and 1. Let everything through in this case, while 1 denotes letting nothing through. These gates are employed to safeguard and regulate the state of the cell.

Step 7. Prediction Phase: The saved model is used to make predictions. The model receives input values and outputs predicted values. The accurateness and losses are then determined by associating the output to the testing data.

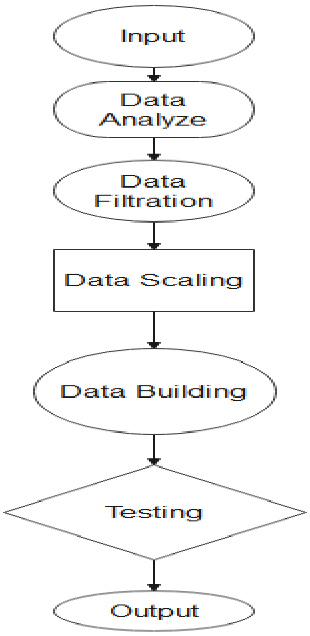

Figure 2. Flow chart of the proposed system

Experimental Setup

Python is used to carry out the suggested strategy. The adopted model was developed using 10000 training epochs and 5 batch sizes. The multi-core CPU described in Table 1's specifications is used to carry out the suggested method. The suggested solution is tested on a single machine to validate and simplify debugging.

Performance Metrics

The effectiveness of the proposed system was evaluated and use the most frequently used criterion:

1. Number of epochs: The term "number of epochs" refers to the total quantity of data that the computer must learn in a single training iteration (see figure 3).

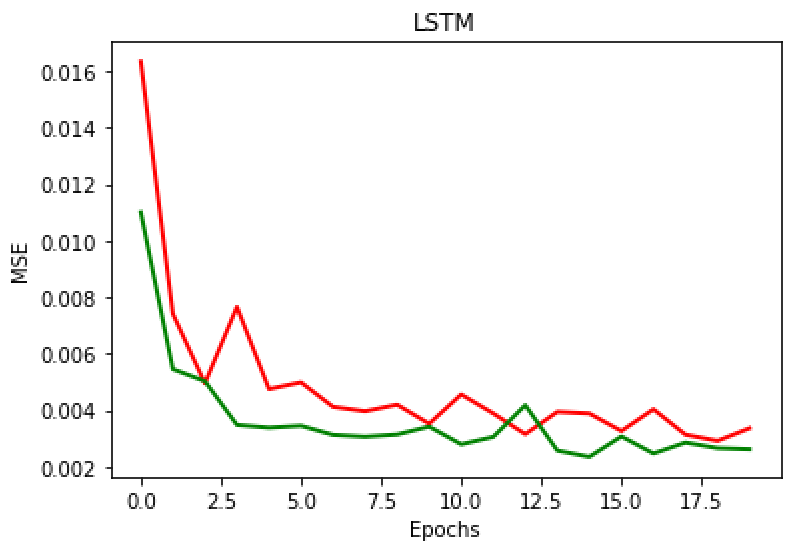

Figure 3. Number of epochs

2. Amount of losses: As depicted in figure 4, this is the accuracy loss brought on by the prediction model's inefficiency. The prediction model's improper tuning and insufficient data are two potential causes.

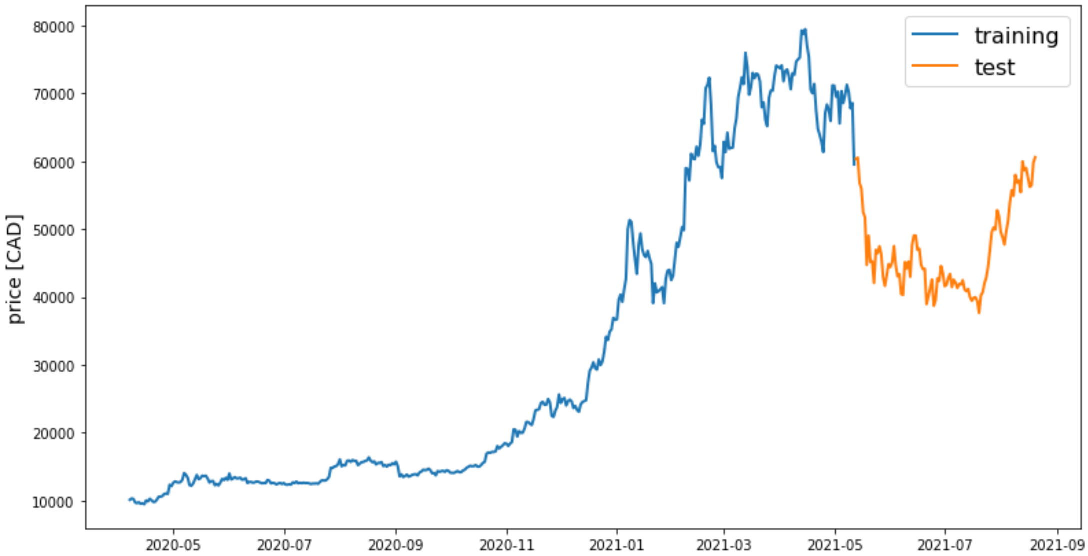

Figure 4. Amount of losses

3. Correlation coefficient: The quantification indicating how closely two variables are related is called a correlation coefficient. The Person coefficient of correlation is by far the most frequently used approach.

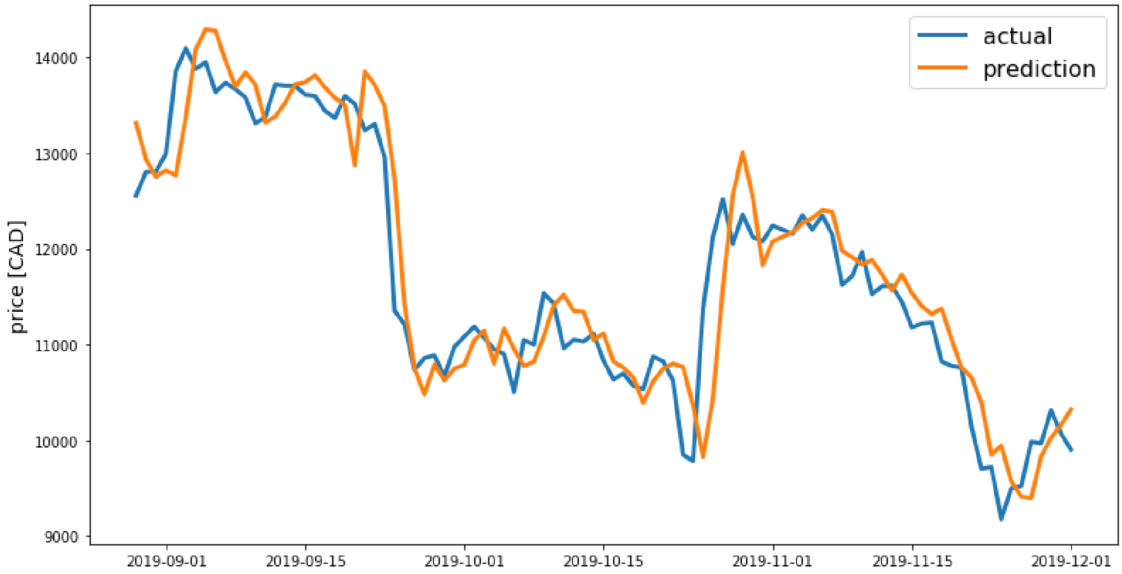

Figure 5. Actual vs Prediction

Conclusion

Central control has significantly diminished as a result of cryptocurrency decentralization. Furthermore, the price of cryptocurrencies has been subject to significant volatility, highlighting the urgent need for precise forecasting techniques. This study suggests a novel approach to predicting cryptocurrency prices that uses RNN and the LSTM to consider variables like market valuation, traffic, circulating supply, and maximal supply. The suggested strategy is developed in Python and tested on standard data sets. The outcomes demonstrate the viability of the suggested strategy for precise bitcoin price prediction. Future studies should expand on the suggested strategy by considering extra variables, including the political climate, interpersonal relationships, and legal frameworks, which differ between nations.

References

[1] Kaastra I & Boyd M, "Designing a neural network for forecasting financial and economic time series", Neurocomputing, Vol.10, No.3, (1996), pp.215-236.

[2] White H, "Economic prediction using neural networks: The case of IBM daily stock returns", IEEE International Conference on Neural Networks, (1988), pp.451-458.

[3] Briere M, Oosterlinck K & Szafarz A, "Virtual currency, tangible return: Portfolio diversification with bitcoin", Journal of Asset Management, Vol.16, No.6), (2015), pp.365-373.

[4] Chatfield C & Yar M, "Holt-Winters forecasting: some practical issues", The Statistician, (1988), 129-140.

[5] Karpathy A, "The unreasonable effectiveness of recurrent neural networks", Andrej Karpathy blog, (2015).

[6] Elman JL, "Finding structure in time", Cognitive science, Vol.14, No.2, (1990), pp.179-211.

[7] Gers FA, Eck D & Schmidhuber J, "Applying lstm to time series predictable through time-window approaches", Neural Nets WIRN Vietri-01, (2001), pp.669-676.
[8] Bengio Y, Simard P & Frasconi P, "Learning long-term dependencies with gradient descent is difficult", IEEE transactions on neural networks, Vol.5, No.2, (1994), pp.157-166. [9] Bengio Y, "Learning deep architectures for AI", Foundations and trends® in Machine Learning, Vol.2, No.1, (2009), pp.1-127.
[10] Hochreiter S & Schmidhuber J, "LSTM can solve hard long time lag problems", Advances in neural information processing systems, (1997), pp. 473-479.
[11] Tschorsch F & Scheuermann B, "Bitcoin and beyond: A technical survey on decentralized digital currencies", IEEE Communications Surveys & Tutorials, Vol.18, No.3, (2016), pp.2084-2123.
[12] Mukhopadhyay U, Skjellum A, Hambolu O, Oakley J, Yu L & Brooks R, "A brief survey of cryptocurrency systems", IEEE 14th Annual Conference on Privacy, Security and Trust (PST), (2016), pp.745-752.
[13] Phillips RC & Gorse D, "Predicting cryptocurrency price bubbles using social media data and epidemic modeling", IEEE Symposium Series on Computational Intelligence (SSCI), (2017), pp.1-7.
[14] Deng Y, Bao F, Kong Y, Ren Z & Dai Q, "Deep direct reinforcement learning for financial signal representation and trading", IEEE transactions on neural networks and learning systems, Vol.28, No.3, (2017), pp.653-664.
[15] Zhao Y, Li J & Yu L, "A deep learning ensemble approach for crude oil price forecasting", Energy Economics, Vol.66, (2017), pp.9-16.
[16] Al Shehhi A, Oudah M & Aung Z, "Investigating factors behind choosing a cryptocurrency", IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), (2014), pp.1443-1447.