A Deep Learning Approach to Forecast Cryptocurrency Prices

Shobhit Nigam

January 13, 2025

A deep learning approach to forecast cryptocurrency prices

Shobhit Nigam[1*]

[1*]Department of Mathematics, Pandit Deendayal Energy University, Raysan, Gandhinagar, 382007, Gujarat, India.

Corresponding author(s). E-mail(s): shobhitngm@gmail.com;

Abstract

This work aims to propose deep learning technique that combines convolutional neural network with single multiplicative neuron model to optimize delay value and improving forecasting efficiency in predicting cryptocurrency prices. This model is proposed with the intent of tackling high non-linearity present in the cryptocurrency prices. A uni-variate time series of daily price of two cryptocurrencies Bitcoin and Ethereum is considered to validate the proposed model. Multiple experiments have been performed to validate the proposed deep learning model and RMSE value is used as the error criteria. The least RMSE value is used in evaluating optimal delay value. The proposed model is **23%-33%** *is more accurate in forecasting compared to the single multiplicative neuron model. The results obtained can give valuable insights for decision making. This work will enable future research studies in time series prediction, as well as facilitate easy adaptation to various time series and with different scenarios.*

Keywords: Deep Learning, Forecasting, Cryptocurrency

1 Introduction

With an increasing demand in digital transactions [7], efforts in financial inclusion and empowerment of cryptocurrency is increasing day-by-day. Financial forecast is an essential part in taking effective decision in business planning. Cryptocurrency, which is not only used for digital transactions but is also traded actively by traders, financial organizations, etc. Efficient forecasting models can help in managing trade deficits and avoiding unexpected shocks. Researchers are putting lot of efforts in developing models which are efficient in producing more accurate predictions. Recent advances in deep learning have revolutionized arenas of computer vision, pattern recognition and natural language processing [1]. Artificial Intelligence and Machine learning are playing a key role in taking the development of efficient prediction models forefront, which is helping policymakers in expanding digital public infrastructure.

Efficient prediction results can significantly reduce potential risks of the investor and increase opportunities for profitable trades. Financial analysts and enthusiastic traders trading in cryptocurrencies, heavily rely on financial forecasting models for accurate price forecasts and are imperative for financial planning. Various deep learning models have been introduced in the past for forecasting cryptocurrency prices [13–15]. Also, sometimes due to delayed decisions of investors the response accuracy in time series gets affected and therefore the number of times the delayed factor impacts the precision of the recorded observations in time series. This could be tackled if the delayed factor could be considered while developing the forecasting model. In [4], an empirical study of lag selection for univariate time series forecasting using deel learning is provided in which they have shown the significant impact on excessive large and excessive small lag sizes. Therefore optimal delay value is one of the most important factor that influence the forecasting efficiency.

A single multiplicative neuron model was introduced in [8] and has been applied on several time series prediction tasks. The learning process can be made more quicker if the changing process can be delivered value upon value which drives it to make 'value multiplying' rather than summing it. In [5], importance of multiplicative operations specifically in motion perception and learning are discussed. Further, it is proved in [9], a network consist of one input and one hidden layer of multiplicative unit can represent any continuous function on a finite interval which was ensured by Weierstrass theorem.

Since cryptocurrency prices are highly non-linear and depends on multiple factors, deep learning techniques are more efficient in learning the changes more accurately. The learning process can be made more quicker if the changing process can be delivered value upon value which drives it to make 'value multiplying' rather than summing it. In [16, 17], the computational power of product units and a plausible neurobiological interpretation is explored, shows how learning operates with product units. Adapting the concept of product unit, a single multiplicative neuron model was introduced in [18] and has been applied on several time series prediction tasks. A fascinating aspect of this approach is that it doesn't require any pre-determined structures prior to training, also used in several time series prediction tasks. The learning methodology for multiplicative neural network is discussed in [3].

In [10], Physics informed neural networks (PINNs) is combined with multiplicative neuron model and demostrated computational efficiency compared to the real PINNs. Although numerous research have been undertaken in deep learning, yet the efficiency of other deep neural network never combined with multiplicative neuron in never undertaken. This work also evaluated the past dependencies of data sets using the proposed approach in forecasting tasks. The main contribution of the work can be summarized as follows:

- Introducing a single multiplicative neuron model in convolutional neural network framework.
- Determining the impact of time-delay in forecasting efficiency considering two data sets of cryptocurrencies Bitcoin and Ethereum using error criteria RMSE.
- A comparative study is performed with SMN model.

In this work, the remaining sections are described in the following way: Section 2 provides the background of the related work. Further, Section 3 describes the proposed architecture and its implementation. In Section 4, performance evaluation criteria used in this work are defined. Experimental setup is explained in Section 5 and results are analyzed in Section 6. Finally, in Section 7, concluding remarks and future aspects of the proposed approach are given.

2 Background

In this Section, a brief explanation of Single Multiplicative Neuron Model (SMN) and CNN is provided.

2.1 Single Multiplicative Neuron Model

A generalized structure of a single multiplicative neuron model is described in [16]. There is a single neuron, which calculates the weighted product of inputs with bias added.

The multiplicative operation shows that how the weights and inputs are multiplied with the bias added, gives y_{prod} defined in Eq. 1 as

$$y_{prod}(\mathbf{x}, \theta) = \prod_{\mathbf{i=1}}^{\mathbf{n}} (\mathbf{w_i x_i} + \mathbf{b_i}) \tag{1}$$

Then y_{prod} is then passed to sigmoid function giving output $\hat{Y}$ defined in defined in Eq. 2,

$$\hat{Y}_i = \frac{1}{1 + e^{-y_{prod}}} \tag{2}$$

2.2 Convolutional Neural Network

A Convolutional Neural Network comprises of convolution layer, Relu and max-pooling layer used for image classification and object recognition tasks [2]. In addition, a set of weights which are called kernel are connected with the inputs and during the convolution operation, the kernel share the weights of the local dependencies present in the data set. The weight sharing is an important property in CNNs and effectively reduce the dimensions of the inputs [11]. CNN are basically used for image classification and object recognition tasks. It has been used, directly or indirectly, in several computer vision problems in reducing the dimensions of the images before presenting it is presented to the neural networks. Also, CNN is introduced for time series prediction tasks [12].

3 Architecture of the Proposed Model

This Section begins with a brief overview of data representation, activation functions and selection of delay. Later backpropagation equations for training the proposed model is presented.

The proposed model consider a uni-variate time series which is in this work considered as the closing price of cryptocurrency. In this work, the original time series data is reconstructed before presenting it to the proposed network. The features of the input are extracted through the uni-variate time series, through which the time delay factor is analyzed. A state at time t which is denoted by Y_t is dependent on the delayed value. The delayed value in the proposed model will serve as the number of dependent variables at a state Y_t. Therefore, given an observed times series of cryptocurrency price $x(t)$, a state $Y_t = [x(t), x(t-1), \cdots, x(t-d)]$ can be generated, where d is the time delay. Figure 1 depicts the architecture of the proposed model.

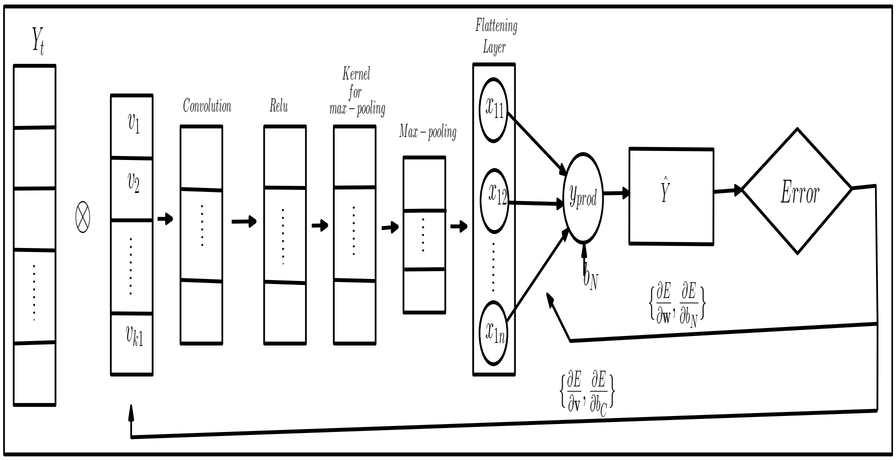

Fig. 1 Architecture of the Proposed Model

The objective of the proposed model is to minimize the loss of the cost function defined in Eq. 3 .

$$Error = \frac{1}{m}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{3}$$

The development of the proposed prediction model is described as follows:

3.1 Data Normalization

Most importantly, how the data set is presented to a network means a lot in terms of reducing the error and computational time while learning any neural network model. Therefore, before the training of the proposed model the time series has been normalized. Data normalization is often performed before the training process begins. The closing price of both Bicoin and Ethereum are normalized in the range [0.1 0.7]. The normalization of the data set is performed using the Eq. 4

$$D_{norm} = \frac{q - Min}{Max - Min}(b - a) + a \tag{4}$$

where D_{norm} denotes the normalized value; q denotes the value to be normalized, Min and Max represents the minimum and maximum value, respectively, of the time series which is to be normalized, a denotes the minimum value of the range and b denotes the maximum value of the range.

For the proposed model, backpropagation equations are derived which are described below. The steepest descent gradient approach and the chain rule for computing the partial derivative is used to update the weights and biases of the proposed model.

$$\begin{aligned}
w_j{}^{(epoch+1)} &= w_j{}^{(epoch)} - \eta \frac{\partial E}{\partial w_j} \\
b_N^{(epoch+1)} &= b_N^{(epoch)} - \eta \frac{\partial E}{\partial b_N} \\
v_k{}^{(epoch+1)} &= v_k{}^{(epoch)} - \eta \frac{\partial E}{\partial v_k}, \\
k &= 1, 2, \cdots \lfloor \frac{d - k_1}{s_1} \rfloor + 1 \\
b_C^{(epoch+1)} &= b_C^{(epoch)} - \eta \frac{\partial E}{\partial b_C}
\end{aligned}$$

4 Evaluation of Error Metrics

The root mean squared error (RMSE) is calculated for the evaluation of the proposed model. The root mean square error (RMSE) is the square root of the average of squared differences between the predicted and observed values. The metric's squared aids in the delivery of more reliable findings by preventing the cancellation of positive and negative error values. It avoids the use of absolute error numbers in mathematical calculations [6] The root mean (RMSE) is descibed below in the Eq. 5

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (Y_i - \hat{Y}_i)^2} \tag{5}$$

where N is the total observations and Y_i and $\hat{Y}_i$ represents the original and predicted cryptocuurency price, respectively.

5 Experimental Setup

This work consider two cryptocurrencies to evaluate the performance of the proposed model, which are Bitcoin and Ethereum. The observations of data sets are considered from the time period of 1st January, 2020 to 20th December 2023, which comprises of 1449 observations . The closing price of cryptocurrencies for two data sets has been collected from site yahoofinance.com. Also, the learning rate to update the weights and biases η is considered as 0.04. The delay value has been varied from 2 to 5. A MATLAB code is generated for the proposed model. To avoid overfitting, a single convolutional layer, pooling layer and a flattening layer is considered which is then followed by a single multiplicative unit.

6 Result and Discussion

In this section, an attempt has been made to focus on those results that are relevant to our particular proposed model. The graphs in Figure 2 and Figure 4 shows the historical data and future predictions of the proposed model which are both plotted together.

The proposed model is trained for 70% and the rest 30% is used for prediction. The batch size for training in both data sets is considered as 950 observations and the rest 500 observations are used for testing. During the training, with various configurations, the error was reduced to 0.001 in Bitcoin and 0.0013 in Ethereum, which are shown in Figure 3 and Figure 5, respectively. The proposed model is also compared with a single multiplicative neuron model. The Table 1 shows the RMSE value for Bitcoin and Ethereum for delay values varied from 2 to 5 with both CNN-SMN and only SMN model. The least RMSE value has been observed in both Bitcoin and Ethereum. It is clear from the Table 2 that the optimal value is 3 in both data sets. Moreover, the proposed model is compared with SMN model.

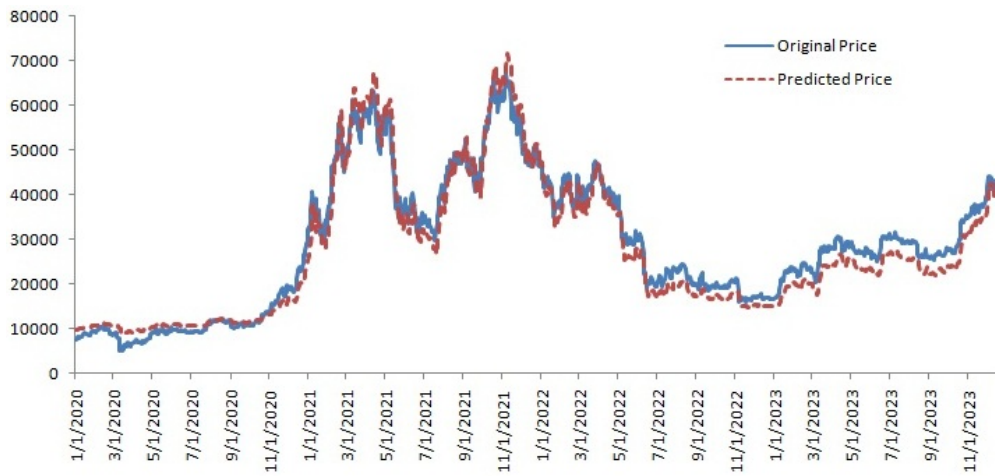

Fig. 2 Predicted Prices of Bitcoin

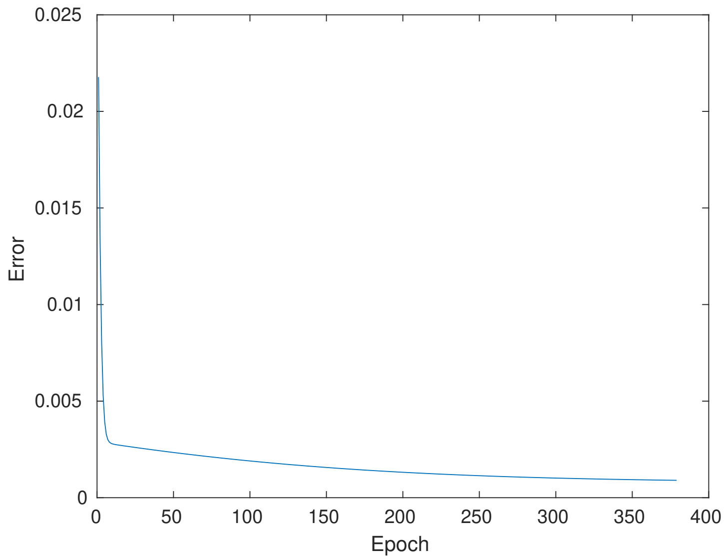

Fig. 3 Error graph of Bitcoin during training

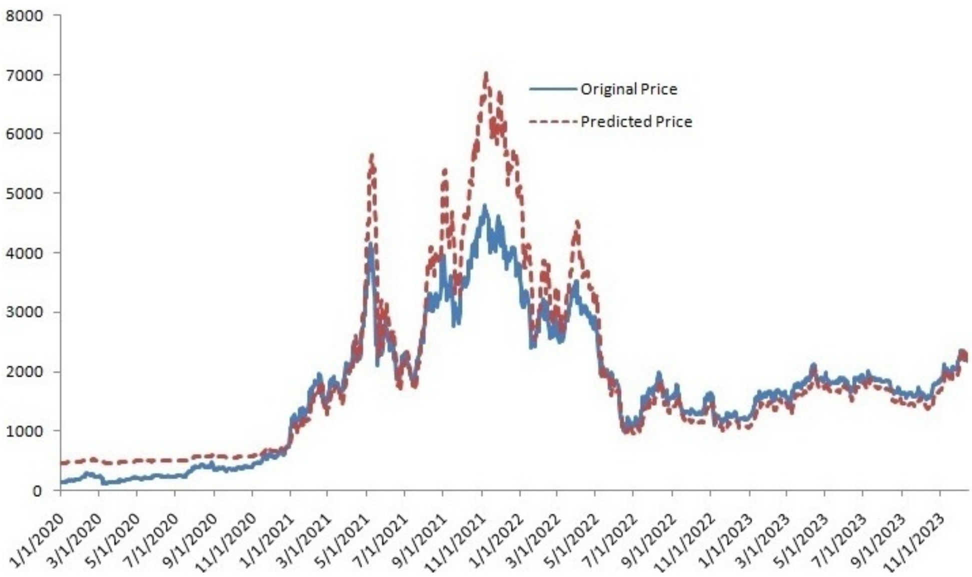

Fig. 4 Predicted Prices of Ethereum

For Bitcoin at delay value 3, the RMSE value is 0.0009 and for only SMN model is 0.0012 which is a significant difference in terms of forecasting efficiency. Similarly, for data set Ethereum at delay value 3, the RMSE value for the proposed approach is

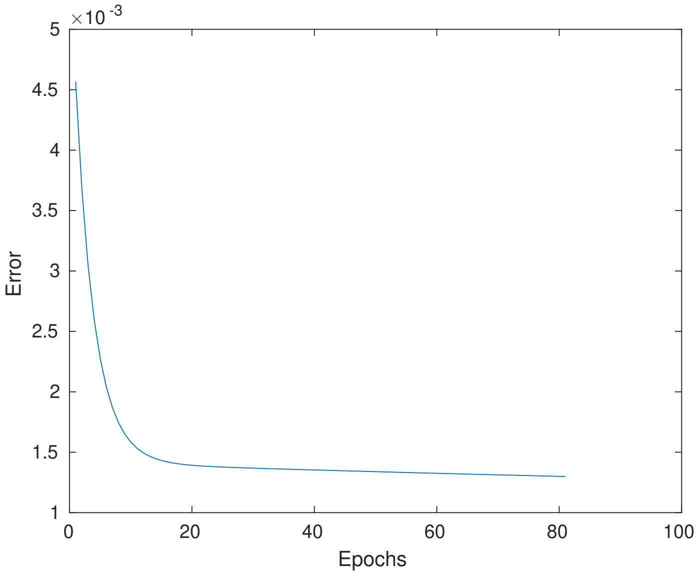

Fig. 5 Error graph of Ethereum during training

	Bitcoin (RMSE)		Ethereum (RMSE)	
Time Delay	CNN-SMN	SMN	CNN-SMN	SMN
2	0.0011	0.0014	0.0018	0.0020
3	0.0009	0.0012	0.0013	0.0016
4	0.0019	0.0023	0.0021	0.0023
5	0.0022	0.0026	0.0024	0.0027

Table 1 RMSE values for CNN-SMN and SMN model)

0.0013 and for SMN model it is 0.0016. Therefore, the proposed model is $23\% - 33\%$ is more accurate in forecasting compared to the SMN model. Figure 2 represents the predicted prices of Bitcoin.

The prediction results are showing strength and even during the last few years the predicted prices are bit far from the original price but the patterns are very much similar. In Figure 4 , the prediction results are showing strength in terms of patterns and prices. However, during the year January 2021 to January 2022, the results shows some sudden spike and the prediction results are bit far from the original price which clearly suggests that a model might be unsuccessful in capturing the volatility during the sudden changes of the prices, which might happen due to various reasons affecting the prices of cryptocurrencies. However, the short term predictions can still giving reliable results and moreover with more experimental settings the results could be improved. The code for the proposed model is developed in MATLAB. However, the short term predictions can still giving reliable results and moreover with more experimental settings the results could be improved.

The proposed approach is implemented on a univariate time series and a single step ahead prediction is performed, it would be encouraging if this could be analyzed for multivariate time series and for multistep ahead prediction. The proposed approach can also be compared with classical CNN which utilizes multilayer perceptron for prediction.

7 Conclusion and Future Direction

In this work, a deep neural network is proposed for forecasting cryptocurrency prices and illustrated how the proposed model can be implemented for other time series forecasting tasks. In addition, backpropagation equations are derived for the learning of the proposed model. The proposed approach is used to analyze dependencies present in the data sets and has been varied from 2 to 5. The time delay value corresponding to the lowest RMSE value is considered as the best forecasting result. The proposed approach anticipates in obtaining reasonable forecasts. As future direction, more example and data analysis are needed to investigate the proposed model. A study needs to be done to determine optimal values for d in order to find best approximation of data sets used in the proposed model. Therefore in future, various time series can be used and the time delay value can be optimized by experimenting with the the varying size of input.

References

[1] Fengxiang He, Dacheng Tao, *Recent advances in deep learning theory*, arXiv:2012.10931.

[2] LeCun Y, Boser B, Denker JS, Henderson D, Howard RE, Hubbard W and Jackel LD. (1989), *Backpropagation applied to handwritten zip code recognition*, Neural Computing, Vol. 1, 541-551.

[3] D. Li, K. Hirasawa, J. Hu, J. Murata, *Multiplication units in feedforward neural networks and its training*, in: Proceedings of Ninth InternationalConference on Neural Information Processing, ICONIP'02, Singapore, 2002.

[4] Leites, Jos´e, Vitor Cerqueira, and Carlos Soares. (2024) *"Lag Selection for Univariate Time Series Forecasting using Deep Learning: An Empirical Study"*, DOI:10.48550/arXiv.2405.11237

[5] Koch, C., T. Poggio, in, T. McKenna, J. Davis, and S.F. Zornetzer (Eds.). (1992) "Multiplying with Synapses and Neurons in Single Neuron Computation", Academic Press Inc., San Diego, CA, USA.

[6] Ioannis E Livieris and Stavros Stavroyiannis, Emmanuel G. Pintelas, Theodore Kotsilieris and and Panagiotis Pintelas, , A dropout weight-constrained recurrent neural network model for forecasting the price of major cryptocurrencies and CCi30 index, Evolving Systems, Springer, Pg. 1-16, 2021.

[7] M.F. Shahzad and S. Xu, W.M. Lim, M.F. Hasnain, S. Nusrat *Cryptocurrency awareness, acceptance, and adoption: the role of trust as a cornerstone*, Humanities and Social Sciences Communications, Nature, 2024.

[8] YadavR.N. Yadav, P .K. Kalra, and J. John. (2006) "Time series prediction with single multiplicative neuron model", Applied Soft Computing, Vol. 7, 1157-1163

[9] Poggio, T. (1975) "On optimal nonlinear associative recall", Biol. Cybernet. Vol. 19, 201-209

[10] Melih Agraz, *Evaluating single multiplicative neuron models in physics-informed neural networks for differential equations*, Scientific Reports, Nature, 14 (2024), 19073, Doi: 10.1038/s41598-024-67483-y.

[11] Yann LeCun, *Generalization and Network Design Strategies. Technical Report CRG-TR-89-4, Connectionism in Perspective,* University of Toronto Connectionist Research Group, Toronto, ON, 1989.

[12] Yann LeCun, Y Bengio, *Convolutional networks for images, speech, and time series*, The handbook of brain theory and neural networks 3361 (10), 1995

[13] Yin, W., Z. Chen, X. Luo, and B. Kirkulak-Uludag. (2024) "Forecasting cryptocurrencies price with the financial stress index: a graph neural network prediction strategy", Applied Economics Letters,, Vol. 31, 630-639.

[14]] Zhanga, Zhuorui, Hong-Ning Daia, Junhao Zhoua, Subrota Kumar Mondala, Miguel Mart´ınez Garc´ıa, and Hao Wangc. (2021) "Cryptocurrency Price Using Convolutional Neural Networks with Weighted and Attentive Memory Channels", Expert Systems with Applications, Vol. 183, 115378.

[15] Zhang, Xinchen , Linghao Zhang, Qincheng Zhou, and Xu Jin. (2022) "A Novel Bitcoin and Gold Prices Prediction Method Using an LSTM-P Neural Network Model, Computational Intelligence and Neuroscience, https://doi.org/10.1155/2022/1643413.

[16] Richard Durbin, David E. Rumelhart: Product units: A computationally powerful and biologically plausible extension to back propagation networks. Neural Computation 1, (1989), p.p. 133–142.

[17] D.E. Rumelhart, G.E. Hinton, R.J. Williams, *Learning internal representations by error back propagations, in: Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, MIT Press, 1986.

[18] R.N. Yadav, P.K. Kalra, J. John, *Time series prediction with single multiplicative neuron model*, Applied Soft Computing, 7 (2006), 1157-1163.