International Journal of Engineering & Technology, 7 (3.27) (2018) 258-264

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Predictive Analysis of Cryptocurrency Price Using Deep Learning

Yecheng Yao[1*], Jungho Yi[2], Shengjun Zhai[3], Yuwen Lin[4], Taekseung Kim[5] Guihongxuan Zhang[6], Leonard Yoonjae Lee[7]

[1]University of Chicago, USA.
[2]Seoul National University, South Korea.
[3]University of California, Davis.
[4]University of California, Davis.
[5]Seoul Science High School, South Korea.
[6]Seoul International School, South Korea
[7]Seoul International School, South Korea.
**Corresponding author E-mail:yechengyao93@gmail.com*

Abstract

The decentralization of cryptocurrencies has greatly reduced the level of central control over them, impacting international relations and trade. Further, wide fluctuations in cryptocurrency price indicate an urgent need for an accurate way to forecast this price. This paper proposes a novel method to predict cryptocurrency price by considering various factors such as market cap, volume, circulating supply, and maximum supply based on deep learning techniques such as the recurrent neural network (RNN) and the long short-term memory (LSTM),which are effective learning models for training data, with the LSTM being better at recognizing longer-term associations. The proposed approach is implemented in Python and validated for benchmark datasets. The results verify the applicability of the proposed approach for the accurate prediction of cryptocurrency price.

1. Introduction

Time series forecasting or prediction is a well-known problem. Much research has been done for predicting markets such as the stock market [1,2]. Cryptocurrenciescan be considered a form of virtual currency intended to serve as a medium of exchange and presents an interesting topic since it can be treated as a time series prediction problem. This problem still remains in nascent stages. Consequently, there is high volatility in the market [3], and this offers opportunities for further research on the prediction of cryptocurrency price.

Moreover, cryptocurrencies such as the Bitcoin are increasingly adopted across the world. Because of the open nature of the cryptocurrency, it operates on a decentralized, peer-to-peer, and trustless system in which all transactions are passed to an open ledger known as the blockchain. Such transparency is unknown in the world of classical financial markets.

Classical approaches such as Holt-Winters exponential smoothing for time series prediction problems are dependent on linear assumptions. These approaches require segregating input data into trends [4] and are more suitable for predicting variables with seasonal effects, such as sales. However, these approaches may not be useful for predicting cryptocurrency price since there is no seasonal effect in cryptocurrencies, which are highly volatile in nature. Given the complexity of such problems, the deep learning paradigm has gained increasing popularity for its performance in solving similar problems such as natural language processing [5]. In the field of machine learning, the recurrent neural network (RNN) and the long short-term memory (LSTM) are well-known approaches. Such approaches havepotential advantages over the traditional multilayer perceptron (MLP) for its temporal nature [6]. This paper proposes a framework for predicting cryptocurrency price using deep learning techniques by considering the nonlinear nature of cryptocurrency price, and the following section discusses the features of deep learning techniques.

The rest of this paper is organized as follows: Section 2 describes deep learning techniques with their advantages and disadvantages. The suitability of deep learning techniquessuch as the LSTM in predicting highly volatile cryptocurrency price is justified. Section 3 examines previous studies in the field of cryptocurrencies, including Bitcoin price prediction and other types of time series prediction in financial markets using machine/deep learning techniques. Section 4 focuses on the proposal of approach to the prediction of cryptocurrency price. The workflow is described and depicted graphically. Section 5 describesthe experimental setup, including implementation methods, benchmark datasets, and performance metrics. Section 6 presents the results in tabular as well as graphical formats. Finally, Section 7 concludes with suggestions for future research.

2. Deep Learning Techniques

Machine learning falls into two categories: supervised learning and unsupervised learning. Supervised learning consists of modeling datasets with labelled instances, whereas unsupervised learning has no such requirement. In supervised learning, each instance can be represented as a set of attributes and target classes. These attributes are mapped into target classes. Examples of supervised methods include neural networks and support vector

machines. In the case of unsupervised learning, similar data instances are grouped into clusters. Examples of unsupervised learning include clustering techniques.
The multilayer perceptron (MLP) is a simple feed-forward neural network that is most commonly used in classification tasks. In terms of neural network terminology, examples fed to the model are known as inputs, and predicted values are known as outputs. Each modular subfunction is a layer. A model consists of input and output layers, with layers between these known as hidden layers. Each output of one of these layers is a unit that can be considered analogous to a neuron in the brain. Connections between these units are known as the weight, which is analogous to a synapse in the brain. The weight defines the function of the model since this weight is the parameter that is adjusted when training a model. However, the MLP's effectivenessis limited with the vanishing-gradient problem. Here, as layers and time steps of the network are related to each other through multiplication, derivatives are susceptible to exploding or vanishing gradients. Vanishing gradients are more of a concern as they can become too small for the network to learn, whereas exploding gradients can be limited using regularization. Another limitation of the MLP is that its signals only pass forward in a network in a static nature. As a result, it does not recognize the temporal element of a time series task in an effective manner since its memory can be considered frozen in time. The MLP can be considered to treat all inputs as a bucket of objects with no particular order in terms of time. As a result, the same weight is applied to all incoming data, which is a naive approach. The RNN, also known as a dynamic neural network, addresses some of these limitations [6].
The structure of the RNN is similar to that of the MLP, but signals can be both forwards and backwards in an iterative manner. To facilitate this, another layer known as the context layer is added. In addition to passing inputs between layers, the output of each layer is fed to the context layer to be fed into the next layer with the next input. In this context, the state is overwritten at each timestep. This offers the benefit of allowing the network to assign particular weights to events that occur in a series rather than the same weight to all inputs, as with the MLP. This results in a dynamic network. In one sense, the length of the temporal window is the length of the network memory. It is an appropriate technique for a time series prediction task [5, 7]. While this addresses the temporal issue in a time series task, vanishing gradient can still be an issue. In addition, some studies have found that, while the RNN can handle long-term dependencies, it often fails to learn in practice because of difficulties between gradient descent and long-term dependencies [8, 9].
LSTM units address both these issues [10]. They allow the preservation of weights that are forward and back-propagated through layers. This is in contrast tothe RNN, in which the state gets overwritten at each step. LSTM units also allow the network to continue learning over many time steps by maintaining a more constant error. This allows the network to learn long-term dependencies. An LSTM cell contains forget/remember gates that allow the cell to decide what information to block or pass based on information strength and importance. As a result, weak signals can be blocked, preventing the vanishing gradient. LSTM cell states have three dependencies that can be generalized as previous cell states, previous hidden states, and current time steps. These states are accountable for memorizing things, and special gates are used for manipulating this memory. These gates are forget gates, input gates, and output gates. As the name indicates, forget gates remove information that is no longer mandatory for the LSTM. Any addition of new information to the cell state is done using the input gate. The input gate makes use of the tanh function, which gives the output in the form of -1 to +1. The input gate ensures that all redundant information is removed and only the most important information is present. The selection of the most beneficial information from the cell state and its display are the main task of the output gate.
Models such as ARIMA depend on linear assumptions about data. Because of the highly nonlinear nature of cryptocurrency price, these models may not provide useful results. Therefore, retaining the nonlinear nature of cryptocurrency price and features of deep learning techniques, this paper proposesthe use of deep learning techniques, specifically LSTM models,to predict cryptocurrency price.

3. Literature survey

Research that applies deep learning techniques to the prediction of cryptocurrency price is in early stages. Some studies have predicted cryptocurrency pricebyusing machine learning techniques. However, there remains an urgent need to develop an effective framework for accurately predicting cryptocurrency price.
Tschorsch and Scheuermann[11] conducted a technical survey of decentralized digital currencies. They examined the building blocks and protocols related to the Bitcoin as a representative cryptocurrency, highlighting key points including centralized digital currencies, the proof of work, blockchains, transactions, scripts, recapitulation, and security. They provided all technical perceptions of the cryptocurrency (distributed currencies), and their elaborative findings can be easily mapped to other cryptocurrencies for a better understanding of their workings.
Mukhopadhyay et al. [12] also presented a survey of cryptocurrency systems. They presented various aspects of cryptocurrencies, includingthe proof of stake, the proof of pork, and their combination for use in data mining techniques. They highlighted that the proof of stake is not sufficient to act independently, whereas the proof of work is resourcedependent. Therefore, a combination of these aspects can result in more accurate results. They also highlighted that most algorithms used for cryptocurrencies are CPU and memory intensive.
Phillips and Gorse [13] proposed the prediction of cryptocurrency price bubbles using social media data and epidemic modeling. Theydemonstrated the use of epidemic modeling and social media data for predicting cryptocurrency price. They used the hidden Markov model(HMM) to detect bubble-like behaviors in the time series and concluded that social media data can play an important role in forecasting cryptocurrency movements.
Deng et al. [14] proposed a deep direct reinforcement learning framework for financial signal representation and trading. They focused on training the computer to beat experienced financial traders in predicting accurate results for financial trading. They proposed to combine reinforcement learning (RL), deep learning (DL), and the recurrent deep neural network (NN) to generate precise prediction results. They validated the proposed approach using commodity future markets as well as stock market data.
Zhao et al. [15] suggested a deep learning ensemble approach called stacked denoising autoencoders (SDAE) for forecasting crude oil price. They used a model based on ensemble learning and deep learningto forecast the price of crude oil. They employed advanced deep neural networks to find the relationship between factors affecting this price. To validate the proposed approach, they considered 198 exogenous variables and concluded that a bagging approach along with the SDAE provides better results in terms of the accuracy of the predicted price of crude oil. They also verified the results through a statistical test.
Shehhi et al. [16] investigated the factors behind choosing a cryptocurrency. They investigated two types of challenges related to cryptocurrencies: the exploration of various factors influencing users to use mine cryptocurrencies and the factors influencing the popularity and price of cryptocurrencies. They explored eight types of cryptocurrencies and conducted an online survey. They concluded that the logo and name of the cryptocurrency are the dominating factors motivating buyers to choose the cryptocurrency. Other factors such as the cryptocurrency community, ease of mining, privacy, and anonymity are also

relevant for preparing the mindset of the user to purchase of a particular type of cryptocurrency.
Chen and Wang [17] used the LSTM algorithm for supervised speech separation. They suggested some improvements by focusing on a model that can perform well for unseen noise and speakers. They found that the DNN was not effective while working with unseen and seen speakers and thus was is not a good option for modeling. To address such gaps, they proposed the use of the LSTM for improved speaker generalization. By modeling temporal dynamics of speech, the LSTM utilizes previous inputs to characterize and memorize a target speaker. Therefore, mask estimation depends on both the current input and LSTM internal states. By visualizing temporal patterns of LSTM memory cells, the authors found a correlation between cell values and speech patterns. These memory cells capture different contexts to improve mask estimation at a current frame.In contrast to the DNN, LSTM performance increases with more seen training speakers. Their work provides a good direction for speech separation.
Sun et al. [18] proposed the use of the lattice long short-term memory algorithm (L^2STM) for the recognition of human actions. They extended the LSTM by exploring memory cells'hidden transition states and performed all operations for discrete spatial locations. They also worked on improving the two-stream architecture for training the network. Their work mutually trained both forget gates and input gates relative to the traditional approach in which both are treated as distinct entities. They applied their proposed approach to benchmark human action recognition datasets HMDB-51 and UCF-101.

4. The Proposed Approach

This section presents the proposed approach for predicting cryptocurrency price using deep learning techniques, as shown in Figs. 1 and 2. The workflow of the proposed approach consists of four units of LSTM input layersfor modeling and a sigmoid activation function for controlling the flow of information and memorizing all patterns formed in cryptocurrency data. The section also proposes the use of theadam optimizer to iteratively update network weightsfor training purposes. The dense layer is passed to make the model more precise. The four LSTM layers used in the proposed approach help make the model more suitable for learning higher-level representations. LSTM layers return their full output sequences, and the dense layer converts the input sequence into a single vector.
This approach uses the square of the correlation coefficient to find the relationship between characteristic fields in the data set. This helps the dominating parameters to derive the values for remaining fields.Then the cryptocurrency price is determined using linear forecast and exponential forecasting. The proposed approach uses the LSTM model to forecast cryptocurrency price. Different phases of the proposed approach are described as follows:

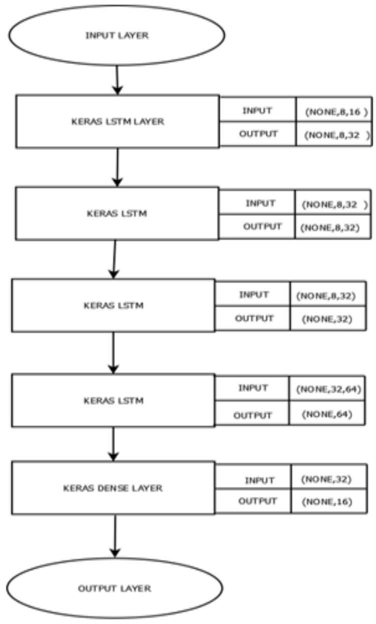

Fig. 1: Workings ofvarious layers of the LSTM **Fig. 2:** Workflow of the proposed approach

1. Data Analysis Phase: This phase analyzes data and its parameters to check any redundancy in data values that may affect prediction results. If a dataset contains any irrelevant parameters, then those data values are removed. This phase alsoanalyze data for the possible merging of data for improved model predictability.
2. Data FiltrationPhase: This phase filters data to remove all empty/redundant values.
3. Train-Test Split Phase: This phase splits data into training and testing data subsets. For example, data are divided into two parts per a ratio of 70% training data and 30% test data.
4. Data-Scaling Phase: Before data are passed to the model, the data are scaled according to model requirements. In this way, this phase reshapes data to make them more suitable for the model.
5. Model-Building Phase: The proposed approach is implemented in Python. For any machine learning models, there are two most powerful libraries in Python: Theano and tensorflow. However, these libraries are difficult to use directly for building a model. Therefore, Keras with tensorflowis used as the backend library to make the model more accurate.The Keras sequential model consists of two layers named LSTM and dense layers. These layers process data in depth to analyze all kindsof patterns formed in the datasetto make the model more precise. Then the data are passed to that model for training.
6. Model Learning and Evaluation Phase: Data are trained using various LSTM units. This consists of four gates: a memory cell, an input gate, an output gate, and a forget gate. These gates are used to let information pass through. They consist of activation layers such as a sigmoid that outputs numbers between 0 and 1. Here 0 means "let nothing through," and 1 means "let everything through." These gates are used to protect and control the cell state.
7. Prediction Phase: Prediction is made using the saved model. Input values are passed to the model to give predicted values as the output. Then that output is compared with testing data to calculate accuracy and losses.

5. Experimental Setup

The proposed approach is implemented in Python. The implemented model is trained on 10000 epochs with 5 batch sizes. The proposed approach is executed on a multi-core CPU with the specifications mentioned in Table 1. For the validation and ease of debugging, the proposed approach is verified on a single system. Because of the flexibly of the programming language, the same work can be easily extended to the GPU.

Table 1: Description of the Proposed Approach

Parameter Name	Description
Processor	Intel Core i7 processor, up to 3.8 GHz
Operating System	Ubuntu
RAM	16 GB
Graphics Processor:	NVIDIA GeForce 930M

Benchmark Data Set

The proposed approach was validated using well-known and oldest cryptocurrencies, namely the Bitcoin(BTC) andthe Litecoin. The BTC dataset consisted of exchanges for the period January 2012 to March 2018, with minute-to-minute updates of OHLC (Open, High, Low, Close), the volume of BTC and the indicated currency, and weighted Bitcoin prices. The dataset wasfreely available for use on the Internet [19]. Two Bitcoin datasets corresponded to the U.S. dollar (USD) asUSD_Small and USD_Large,and one Bitcoin dataset corresponded to the Japanese yen (JPY). The Litecoin dataset was taken from the website [20]. The corresponding dates of the dataset varied from December 27, 2013, toJune 2, 2018. Salient characteristics of the dataset are shown in Tables 2 and 3.

Table 2: Bitcoin Datasets in different Currencies

Parameter	Value/Description	Value/Description	Value/Description
Dataset Details	USD (Large in number named as L1)	USD (Lesser in number named as L2)	JPY (Named as L3)
Memory usage	199.8 MB	99.9 MB	23.1 MB
RangeIndex	3273377 entries, 0 to 3273376	16865 entries 0 to 16865	37909 entries 0 to 37909
Total Data Columns	8		
Timestamp	int64		
Open	float64		
High	float64		
Low	float64		
Close	float64		
Volume_(BTC)	float64		
Volume_(Currency)	float64		
Weighted Price	float64		
Dtypes	float64(7), int64(1)		

Table 3: Different Fields for the Litecoin Dataset

Date	Open*	High	Low	Close**	Volume	Market Cap
Jun 02, 2018	119.8	123.9	119	123.3	31,08,00,000	6,80,54,30,000

Performance Metrics

The performance of the proposed approach was measured using the most commonly used metrics:

1. Number of epochs: This is defined as a complete amount of data that must be learned by the machine in a single iteration during the training stage.
2. Amount of losses: Thisis defined as the loss of accuracy due to the inefficiency of the prediction model. The reason can include insufficient data and the improper tuning of the prediction model.
3. Correlation coefficient:The measurement indicating how strong the relationship between two variables. The Person correlation coefficient is the most commonly used method, and its formula for any two relationships x and y is given by

$$r = \frac{n(\sum xy)-(\sum x)(\sum y)}{\sqrt{[n\sum x^2-(\sum x)^2]\,[n\sum y^2-(\sum y)^2]}}.$$

4. Mean absolute scaled error(MASE): This is a measure of the accuracy of forecasts. Forecast errors for a given period are represented by the numerator, and its value is calculated by subtracting the forecast value(F_t) from the actual value(Y_t) as $e_t = Y_t - F_t$. For a nonseasonal time series,the denominator is the representation of the mean absolute error of one step using the “naive forecast method” for the given dataset. Its value is calculated by using the prior period as the new forecast represented by $F_t = Y_{t-1}$ [21]:

$$\text{MASE} = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{|e_t|}{\frac{1}{T-1}\sum_{t=2}^{T}|Y_t-Y_{t-1}|}\right) = \frac{\sum_{t=1}^{T}|e_t|}{\frac{T}{T-1}\sum_{t=2}^{T}|Y_t-Y_{t-1}|}.$$

5. For a seasonal time series, the seasonal naïve forecast method can be used to findthe mean absolute error for the training dataset. The actual value of the prior season serves as the new forecast $F_t = Y_{t-m}$:

$$\text{MASE} = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{|e_t|}{\frac{1}{T-m}\sum_{t=m+1}^{T}|Y_t-Y_{t-m}|}\right) = \frac{\sum_{t=1}^{T}|e_t|}{\frac{T}{T-m}\sum_{t=m+1}^{T}|Y_t-Y_{t-m}|}.$$

6. Systematic mean absolute percentage error: This is the measurement of accuracy based on the percentage error. Here the actual value is represented by At, and Ft represents the forecasted value. The absolute difference between F_t and A_t is divided by theirsum, followed by the further summation of all fitted points divided by the count n of fitted points:

$$\text{SMAPE} = \frac{100\%}{n}\sum_{t=1}^{n}\frac{|F_t-A_t|}{(|A_t|+|F_t|)/2}.$$

7. Mean absolute error (MAE): This measures the difference between two continuous variables. For a given scatter plot with n points, any point i is represented by coordinates (xi, yi), and the MAE is the average vertical distance between each point and the line Y=X. This line is called a one-to-one line [22]:

$$\text{MAE} = \frac{\sum_{i=1}^{n}|y_i-x_i|}{n} = \frac{\sum_{i=1}^{n}|e_i|}{n}.$$

8. Root mean square error: This measures the error between two datasets.This is calculated as the sum of all observations calculated as the differencebetween the predicted value(Pi) and the observed value(Oi) divided by the number of observations(n):

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(P_i-O_i)^2}{n}}.$$

6. Results and Discussion

The model wasexecuted and implemented for benchmark datasets. The square of the correlation coefficient was used to find a dominating feature from the complete dataset, and then correlationswere calculated between Market Open and Market High, Market Open and Market Low, Market Open and Market Close, and Market Open and Market Volume. Correlations between all kinds market data are shown in Fig. 3.

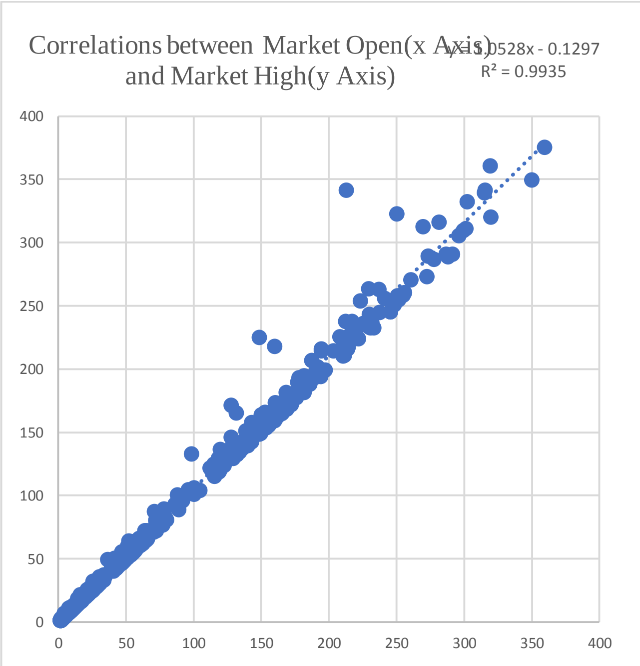

Fig. 3: Correlationsbetween market open (x-axis) and market high (y-axis)

The value of the square of the correlation coefficient of 0.9935 implies a high correlation between Market Open and MarketHigh, indicating an increase in Market High if the market opens at a high value and a decrease if the market opens at a low value. Market Open and Market High followed a linear relationship. Any value of Market High can be found using the following equation:

Market High= 1.0528 * Market Open – 0.1297.

The relationship strength of two variables were measured using the correlation coefficient.

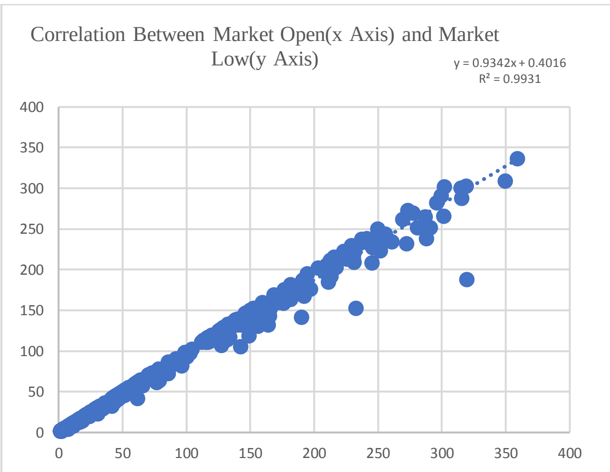

Fig. 4: Correlation between market open (x-axis) and market low (y-axis)

The squared correlation coefficient of .9931 indicates a high correlation between Market Open and Market Low, indicating an increase in the market if the market opens at a high value and a decrease if the market opens at a low value.

Market Open and Market Low followed a linear relationship. Any value of Market Low can be found by putting the value ofMarket Open in the following equation:

Market Low= .9342* Market Open + .4016.

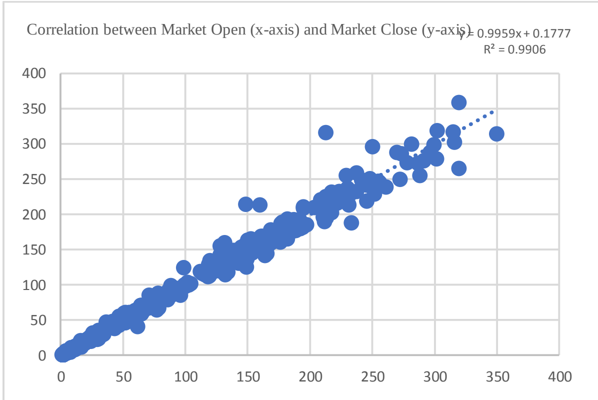

Fig. 5: Correlation between market open (x-axis) and market close (y-axis)

The squared correlation coefficient of .9906 indicates a high correlation between Market Open and Market Close, indicating an increase in Market Close if the market opens at a high value and a decrease if the market opens at a low value.

Market Open and Market Low followed a linear relationship. Any value of Market Close can be found by putting the value ofMarket Open in the following equation:

Market Close= 0.9959* Market Open + 0.1777.

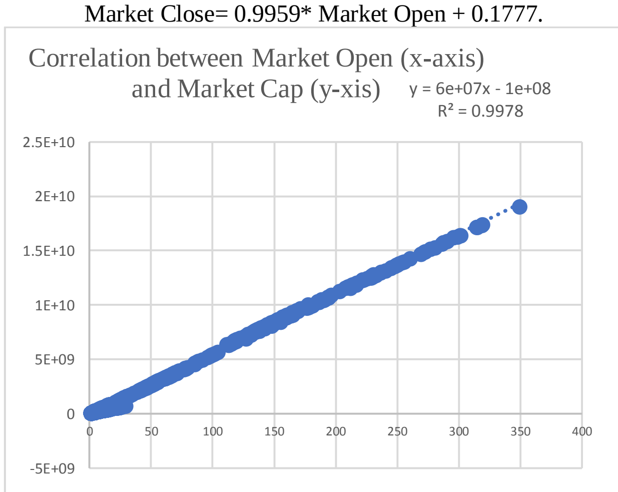

Fig. 6: Correlation between market open (x-axis) and market cap (y-axis)

The squared correlation coefficient of 0.9978 indicates a high correlation between Market Open and Market Cap, indicating an increase in Market Cap if the market opens at a high value and a decrease if the market opens at a low value.

Market Open andMarket Cap followed a linear relationship. Any value of Market Cap can be found by putting the value of Market Open in the following equation:

Market Cap= (6e+07) * Market Open+ (1e+08).

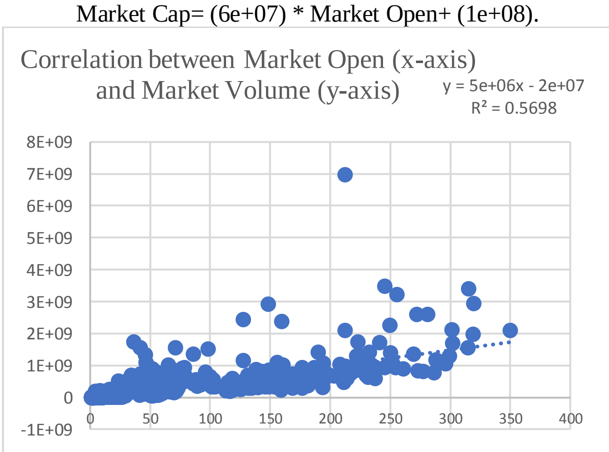

Fig. 7: Correlation between Market Open (x-axis) and Market Volume (y-axis)

The squared correlation coefficientwas quite low for Market Open and Market Volume. This means that Market Volume was not entirely dependent on Market Open. Market Open was projected using linear forecast and exponential forecast method. The forecasted area is marked on the graph.

Table 4: Relationships between different Fields and Derived Equations

Value Field 1	Value field 2	Relationship	Sqared correlation coefficient	Equation
Market Open	Market High	Linear	.9935	Market High= 1.0528 * Market Open – 0.1297
Market Open	Market Low	Linear	.9934	Market Low= .9342*Market Open + .4016
Market Open	Market Close	Linear	.9906	Market Close= .9959* Market Open+ .1777
Market Open	Market Cap	Linear	.9978	Market Cap= (6e+07) * Market Open+ (1e+08)
Market Open	Market Volume	Random	.5698	NA

Forecasted values for Market Open based on exponential forecasting are shown in Fig. 7.

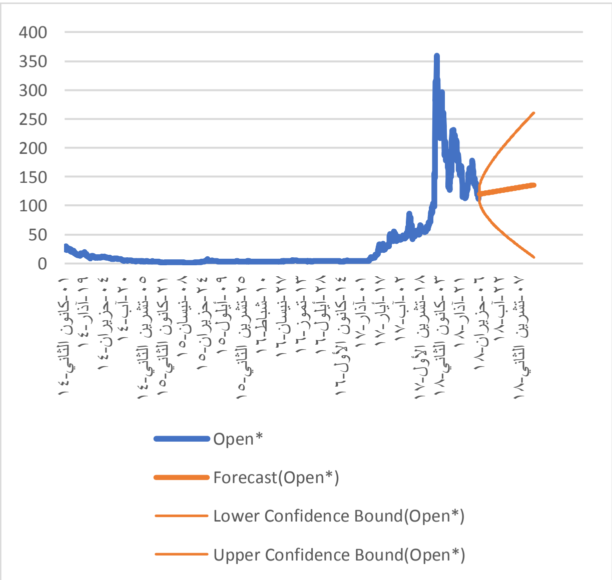

Fig. 8: Forecast values for litecoin lying in lower and upper bounds over the interval

Table 5: Values of different statistics for errors and their values

Statistic	Value
Mean absolute scaled error	30.36
Systematic mean absolute percent error	0.07
Mean absolute error	9.48
Root mean square error	15.84

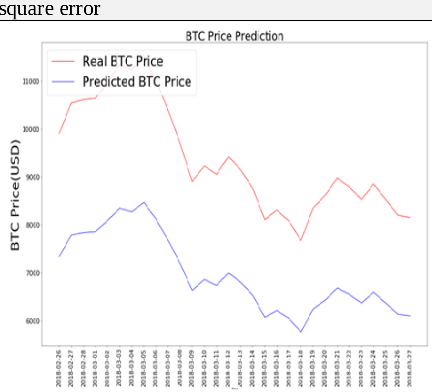

Fig. 9: BTC prediction values corresponding to dataset L2

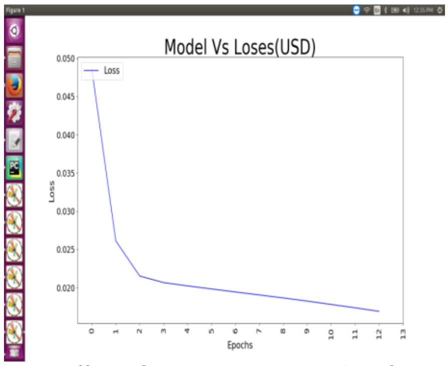

Fig. 10: Effects of an increase in the number of epochs

Figs. 9 and 10 illustrate prediction values in USD for 100M data. Herethe red line depicts actual values, and the blue line, predicted values. These figures show improved results based on a large decrease in the difference between actual and predicted prices. The proposed approach achieved78% accuracy. The model generateda maximum of 45% loss on 13 epochs.

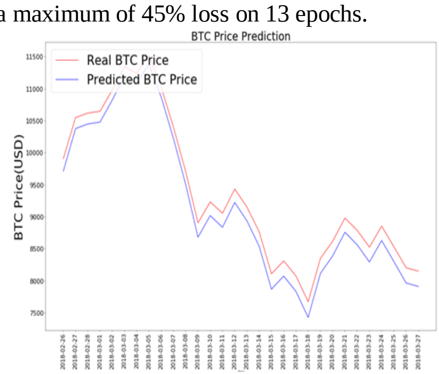

Fig. 11: BTC prediction for data set L1

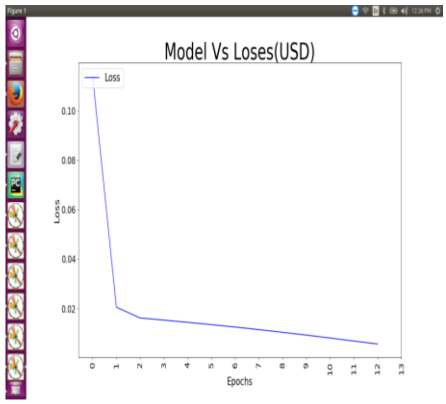

Fig. 12: Effects of an increase in the number of epochs

Figs 11 and 12 show the prediction values in USD for 200M data. Herethe red line depicts actual values, and the blue line, predicted values. These figures show improved results based on a large decrease in the difference between actual and predicted prices. The above scenario showed87% accuracy, and the reason behind the improved resultswas better model training based on the huge dataset. The model generateda maximum of 10% loss on 12 epochs, which is good for a prediction model.

Similar prediction analyses were performed in JPY for 20M data. The corresponding accuracy was merely 59% for 13 epochs. This clearly shows the role of the amount of available information for making predictions. Larger datasets provide better training of the neural network for more efficient results.

Figure 12 depicts the variation in accuracy with respect to the size of the dataset. The results show 59% accuracy for 20M data, 75% for 100M data, and 90% for 200M data.

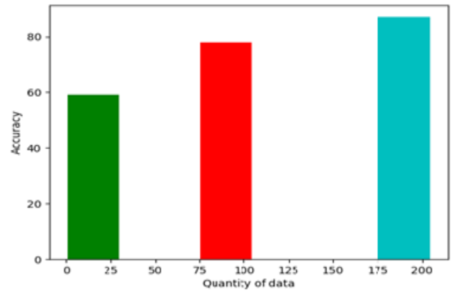

Fig. 13: Accuracy vs. quality

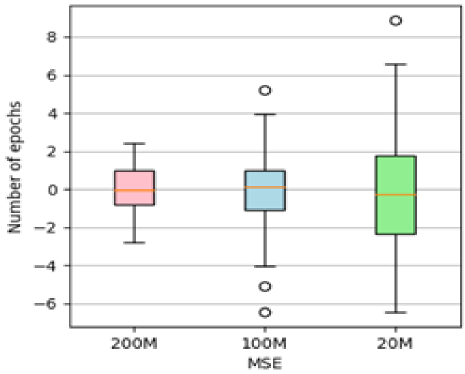

Fig. 14: MSE values for different data sets

Fig. 13 shows the variation in the MSE rate for different datasets for the number of epochs. The error rate was inversely related to the size of the dataset. That is, the error rate decreasedwith an increase in the size of the dataset and vice versa [24]. Fig. 14 provides an overall comparison in accuracy, error rates, and loss rates for different datasets.

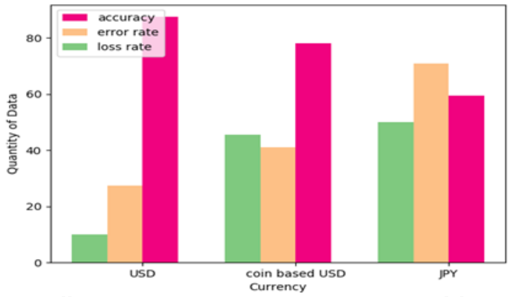

Fig.15: Overall comparison in accuracy, error rates, and loss rates for different datasets

In sum, the results highlight that Market Open may play a key role in influencing all other parameters. In addition, the size of the dataset may influence future predictions, as indicated with the results for the proposed model trained using large datasets.[25]

7. Conclusions and Future Research

The decentralization of cryptocurrencieshas sharply weakened central control. Further, wide fluctuations in cryptocurrency price indicate an urgent need formethods to accurately forecast cryptocurrency price. This paper proposes a novel method to predict cryptocurrency price by considering various factors such as market cap, volume, circulating supply, and maximum supply based on deep learning techniques such as the RNN and theLSTM. The proposed approach is implemented in Python and validated for benchmark data sets. The results verify the applicability of the proposed approach for the accurate prediction of cryptocurrency price. Future research should extend the proposed approach by considering additional parameters such as the political environment, human relations, and regulations, which vary across countries.

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.

[17] Chen J & Wang D, “Long short-term memory for speaker generalization in supervised speech separation”, *The Journal of the Acoustical Society of America*, Vol.141, No.6, (2017), pp.4705-4714.

[18] Sun L, Jia K, Chen K, Yeung DY, Shi BE & Savarese S, “Lattice long short-term memory for human action recognition”, *IEEE International Conference on Computer Vision*, (2017).

[19] www.kaggle.com [https://www.kaggle.com/mczielinski/bitcoin-historical-data/data]

[20] https://coinmarketcap.com/currencies/litecoin/

[21] Hyndman RJ & Koehler AB, “Another look at measures of forecast accuracy”, *International journal of forecasting*, Vol.22, No.4, (2006), pp.679-688.

[22] Willmott CJ & Matsuura K, “Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance”, Climate research, Vol.30, No.1, (2005), pp.79-82.

[23] G, Abikhanova, A Ahmetbekova, E Bayat, A Donbaeva, G Burkitbay (2018). International motifs and plots in the Kazakh epics in China (on the materials of the Kazakh epics in China), Opción, Año 33, No. 85. 20-43.

[24] A Mukanbetkaliyev, S Amandykova, Y Zhambayev, Z Duskaziyeva, A Alimbetova (2018). The aspects of legal regulation on staffing of procuratorial authorities of the Russian Federation and the Republic of Kazakhstan Opción, Año 33. 187-216.