iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** IX **Month of publication:** September 2023

DOI: https://doi.org/10.22214/ijraset.2023.55647

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue IX Sep 2023- Available at www.ijraset.com

Digital Currency Price Prediction using Machine Learning

Sahil Kadadekar[1], Sumaiya Shaikh[2], Isheeta Shahir[3], Hemantkumar Mali[4], Rupesh Jaiswal[5]
Department of Electronics and Telecommunication, Pune Institute of Computer Technology

Abstract: ***Coming into the 21st century, the views towards finance and investments have drastically shifted from traditional assets like gold, land, and property to assets in the digital domain, like investing in various stocks, mutual funds, currencies, etc. The more recent developments of technology and newer investment options have brought various block-chained digital currencies to the forefront. Digital currency options offer massive gains but are very dynamic and quick-moving investments, making predicting the future values for maximum gains difficult, thereby reducing the confidence of even seasoned investors. While digital currencies saw massive gains in the past decade, research in deep learning has seen equivalent growth in terms of efficiency, required computation, and prediction rate. In this project, we explored various digital currencies, using multiple machine learning models from extended trees to time series analysis, only to use Long Short-Term Memory (LSTM) model to get accurate predictions for the tested parameters. While there is a risk of overfitting the dataset, given that this project uses APIs to get real-time data, the risk is mitigated by the fact that it's giving precise outputs for unseen data.***
Keywords: Machine Learning, LSTM, Forecast, BTC, API

I. INTRODUCTION

The advent of bitcoin in 2008 as an individual cryptocurrency that slowly climbed to a global domain leading means for cryptocurrency transactions, lead to a spur of development in the field of blockchain. Initially, cryptocurrencies carried set value and were considered as a reward for an activity duly named mining. However, as the internet became the main source of communication between people all over the world, and small businesses and competitions started to acquire a global audience, the restriction of using a singular regulated currency, i.e., the US dollar, became cumbersome. This spurred rapid growth in the blockchain and crypto market, which made cryptocurrency the main means of transactions between people over the internet without the cumbersome nature of banking beyond international borders, acquiring a specific currency, and the fees associated with the process. The sudden boom in the crypto market also made it one of the premier investment opportunities with the caveat that it was also highly unstable given it was decentralized and offered no security for currency bonds. To counterbalance this volatility and lack of investor confidence in the money invested, the solution of forecasting future prices using machine learning algorithms was proposed. For the current project, the team plans to use the numerical data for the digital currency fetched from the API to train a recurrent neural network, namely Long Short-Term Memory (LSTM) to predict digital currency prices. LSTM is a recurrent neural network in which connections are made along the nodes in a temporal sequence to form sequential graphs where the individual nodes process both singular data points and remember data over arbitrary time intervals such that the model can process both individual data points and the total sequential data to account for the effect of individual points and the trends in the total data for forecasts for a required duration of time.

II. RELATED WORK

The analysis performed utilized multivariate and multi-step machine learning algorithms [30] for cryptocurrency volatility forecasting, this research highlighted that even for relatively small datasets the calculation time for a successful prediction was comparatively much higher than for any other comparable algorithm. The usage [1, 9, 4] of multivariate linear regression for predicting cryptocurrency prices was proposed which focused on multivariate linear regression, using the historic dataset for training, and then trying to compare future values for validation of the model created. However, after optimizing the machine learning models to fit multiple features mentioned above, the resultant accuracy hit its peak at 67 percent for the prediction of BTC price. While a comparative analysis of models such as linear regression, and polynomial regression was done with time series analysis which showed that traditional algorithms do not work as effectively for dynamic data such as cryptocurrencies.

Bc. Vojtech Pulec [26], published research that utilized Google Trends of respective cryptocurrencies as a parameter for the forecasting of cryptocurrency prices using univariate and multivariate time series analysis.

©IJRASET: All Rights are Reserved

The key feature is that the models utilized for price forecasting, namely, ARMA for univariate time series analysis and VAR and VECM for multivariate time series analysis. For bitcoin market analysis, Vladimir Puzyrev [27] created a deep convolutional autoencoder which looked at historical data from 40 different cryptocurrencies to extract the most important properties from each one. The key findings [23] were the fact that time series analysis, especially the ARIMA model promised a higher degree of accuracy for forecasting values of cryptocurrencies than AR, MA, or ARMA individually. The paper [19] utilized historical cryptocurrency data acquired from crypto compare public API while the social data was obtained in the form of raw tweets from Twitter with a series of parameters applied for sorting for tracking social media sentiment. This paper [31] highlighted a comparative analysis of MLP, SVM, and RF for four cryptocurrencies namely BTC, ETH, Ripple, and Litecoin. However, the accuracies for prediction using sentiment analysis as a feature for the selected models offered subpar accuracy ranging from 35 percent to 66 percent for a varied set of parameters. The dataset utilized was historic data of BTC collected via API integration. Sriramya [28] performed a comparative analysis of selective machine learning models such as KNN, Linear Regression, LASSO Regression, Polynomial Regression Ridge Regression, and Random Forests for prediction of BTC prices using historical data integrated from CoinMarketCap.com API. In a paper [25] published in IJACSA 2020, authors explored a different approach to the processing problem faced by large datasets generated by cryptocurrencies. Using IoT to push historical data to a remote server with sufficient processing capabilities for cloud computing yielded promising results. KNN and Linear Regression performed better than the other models tested for the same. The paper [2] concluded that the GRU model was more capable of predicting volatility in BTC. Students from IAE [8] , utilized Recurrent Neural Network to predict BTC prices. LSTM gave a high degree of accuracy for prediction on the testing set. This model was not utilized for deployment on realtime data, and for forecasting future values of BTC. In 2021, researchers [14] utilized multiple Recurrent Neural Network models such as GRU, LSTM, and tree-based models for Bitcoin market price prediction. This research had integrated APIs for minute-by-minute data but failed to build a model that offered considerable forecasting accuracy. CHRIST university [10] , published a comparative analysis of LR and LSTM models of machine learning, using various parameters as features for model training. In the current digital world, with the advent of Machine Learning and the increasing computational capabilities of systems, machine learning has found application in various fields, ranging from classification problems, forecast in the digital domain such as internet traffic classification, image processing, biological signals, stock market, digital currencies, classification, and predictions [11, 29, 5, 6, 7, 20, 24, 15, 22, 16, 12, 13, 18, 21, 3, 17]

III. METHODOLOGY

A. Dataset

API: - The main dataset for this project is integrated with the use of an API that offered data from 4 months ago since the point at which the code is run. This dataset was obtained from the API and was initially for USDT in two different periods, namely, minute-by-minute and daily. For the current iteration of model training, the dataset extracted was from 21st November 2021 to 28th November 2021 for the minute-by-minute dataset while for the daily dataset, the dataset extracted was from 4th May 2021 to 28th November 2021. The daily dataset has 209 entries overall, compared to 11752 in the minute-by-minute dataset. Using the same API, data was gathered for six more cryptocurrencies: Bitcoin (BTC), Ethereum (ETH), Polkadot (DOT), Litecoin (LTE), Dash Coin (DASH), and Dogecoin (DOGE) for intervals of 86400 seconds, 21600 seconds, 3600 seconds, 300 seconds, and 60 seconds each. These datasets have 6 main features which are DATE-TIME, LOW, HIGH, OPEN, CLOSE, and VOLUME out of which DATE-TIME is used as the index.

Passive Dataset: - For the model training to yield results with higher accuracy, a passive dataset consisting of historical data of the cryptocurrency [USDT] was utilized. This dataset had a total of 288562 entries for the minute-by-minute movement of the cryptocurrency, from 4th May 2021 to 22nd November 2021[11:14 AM]. Similarly, the dataset for the daily values has a total of 784 entries, from 18th September 2019 to 9th November 2021. This dataset has the same features as the API dataset except for an additional column added in the daily dataset, namely, ADJUSTED CLOSE, which accounted for any aftermarket transactions and changes in the value of the cryptocurrency.

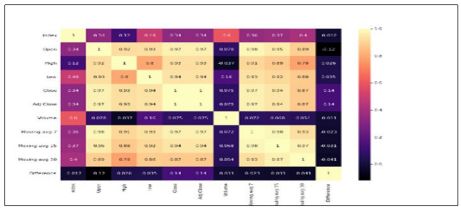


Fig. 1: Passive USDT dataset heatmap

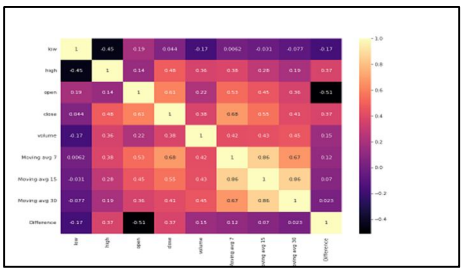


Fig. 2: API USDT dataset heatmap

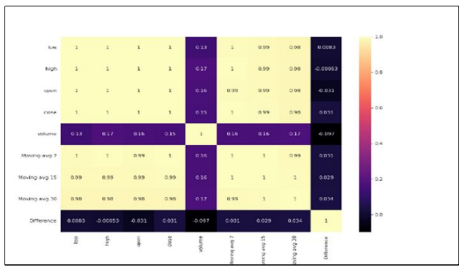


Fig. 3: API BTC dataset heatmap

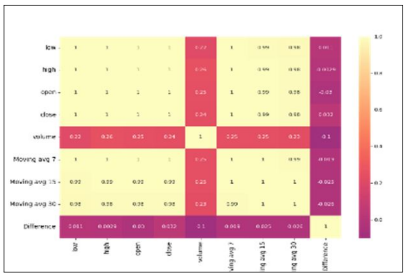


Fig. 4: API ETH dataset heatmap

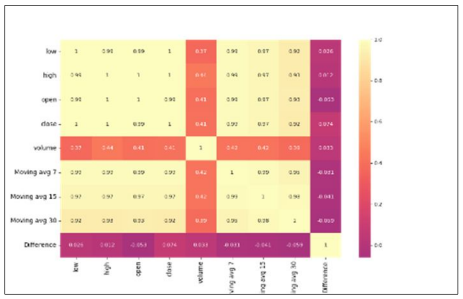


Fig. 5: API DOT dataset heatmap

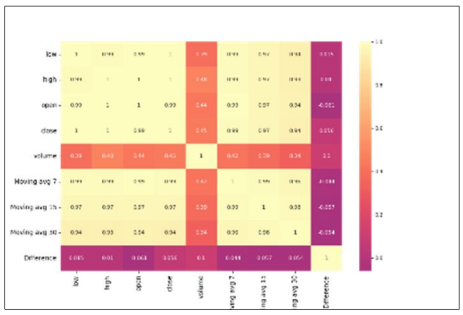


Fig. 6: API LTC dataset heatmap

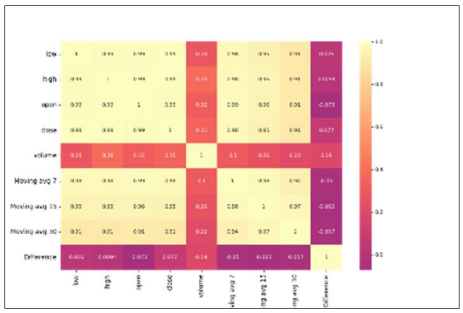


Fig. 7: API DASH dataset heatmap

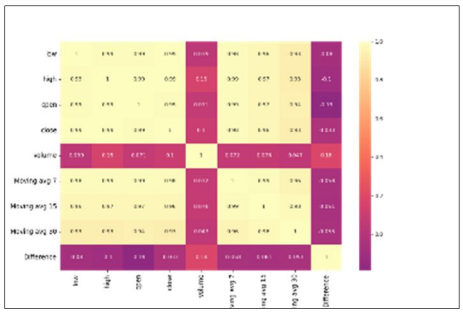


Fig. 8: API DOGE dataset heatmap

B. Data Pre-processing

Data cleaning: - After the initial fetching of data, the data needs to be processed for a multitude of factors before passing it on to the machine learning pipeline. After careful consideration, the API data was collected for maximum efficiency and minimum time loss in the data-gathering phase. Given that the API gives only the necessary data, no columns were dropped. The entire dataset is checked for missing or corrupted values in each training cycle. The nature of the dataset acquired from the API is dynamic and subject to corruption, causing missing values to appear in the dataset which in turn will cause the model to malfunction and predict forecasts incorrectly. To avoid this problem, the python library function of forward-fitting is utilized where the library takes the average of the preceding and succeeding value and fits it in place of the missing values, which helps the dataset avoid any 'NaN' values from appearing in the dataset to be used as the training set for the machine learning model. The formula for which is:

$$F = \frac{\left(\frac{SSE_{j} - SSE_{(j+X_{\propto})}}{DF_{X_{\propto}}}\right)}{MSE_{(j+X_{\propto})}} \quad (1)$$

Data Normalization: After the initial processing and data analysis, it is observed that the data is extremely volatile with large gradient changes, which ultimately result in a lack of convergence of the machine learning pipeline for the forecast. This problem is approached and solved by scaling the data using the MinMax scaler. MinMaxScaler is a library in the Scikit-learn module, which is used to scale the data. It scales all data in the range from 0 to 1 if the data contains only positive values and from -1 to 1 if the data contains negative values. M in

$$\text{MinMaxScaler} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2)$$

The initial observations after training the model on an unscaled dataset showed that the LSTM model was failing to acquire the trends the dataset took, and forecasts were slightly irregular and unable to follow the trends in the testing set. To avoid this, the dataset was scaled using the MinMax scaler of the sci-kit-learn library to scale the data before being passed to the LSTM model for training and testing.

Machine Learning pipeline: The nature of the data in any digital currency is primarily that, it is heterogeneous and continuous time-series data. For the current consideration, a variety of combinations in terms of values, period of the data considered, and intervals between individual data points were utilized.

Given that the model had to be updated dynamically for prediction and the test set were values that were to be predicted for the coming 15 days, the main challenge was to find the total span of data which was just enough to give consistently high accuracy with minimal computational resources and time.

Train-Test split: - For the sake of results and validation of the model, the data gathered from the API is passed on to the model with a split of the recent 15 days data being removed from the training data pool. These values are later plotted and visualized versus the predicted values forecasted by the LSTM model to validate the veracity and accuracy of the model predictions and check for various validation parameters.

Data Reshaping: - The key aspect of passing any data to LSTM is that it is supposed to be passed as a tensor, that is, it should be in a 3-dimensional vector. Based on the shape of this tensor, the input layer of the LSTM is designed. This layer must account for all necessary data that is in terms of window size, samples, and the number of features to be utilized during the training phase. Hence, the data is reshaped as per the requirements of the model and passed to the model for compilation.

LSTM: - Long Short-Term Memory (LSTM) is a machine learning algorithm that relies on feedforward neural networks with feedback connections. Normally, feedforward networks try to not hold any memory of the previous iterations in the next step of the learning process. However, in LSTM, the feedback connections allow the model to hold memory from the previous iterations, allowing it to process entire continuous sequences of datapoints. The key components of LSTM are a cell, an input gate, an output gate and a forget gate.

$$\text{Input gate } (i_t) = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (3)$$

$$\text{Forget gate } (f_t) = \sigma\left(W_f[h_{t-1}, x_t] + b_f\right) \quad (4)$$

$$\text{Output gate } (o_t) = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (5)$$

$$\text{Intermediate Cell State}(C^{\sim}) = \frac{1}{1+e^{-x}}(W_c[h_{t-1}, x_t] + W_c X_t) \quad (6)$$

$$\text{Cell State}(\ c_t\ ) = (i_t * C^{\sim}t) + (f_t * c_{t-1}) \quad (7)$$

$$\text{Calculating new state } (h_t\ ) = o_t * \frac{1}{1+e^{-x}}\ (c_t) \quad (8)$$

The cell's job is to remember values over a random arbitrary interval, while the forget gate is used to remove the values, that is, forget values that are not a part of the learning process for acquiring trends that are necessary for forecasting future values. Given that this processing is done in a loop, it allows the LSTM network to remember relevant values into cells and forget irrelevant values using the forget gate. Thus, LSTM is a supervised learning algorithm with an artificial Recurrent Neural Network (RNN) structure, whose algorithm is as follows:

1) Start
2) Process the data to account for missing values, Normalization, Scaling, and finally turn the data into a tensor by rescaling.
3) Pass the tensor to the LSTM. This decides the input layer shape for the LSTM.
4) Using an activation function (Sigmoidal in this iteration), set the number of units in the input layer.

$$\text{Sigmoidal function} = \frac{1}{1+e^{-x}} \quad (9)$$

5) Insert the output dense layer with a select number of units. Given that only one value is being forecast, a single unit is enough for this iteration.
6) The model is compiled using an optimizer (Adam) and a selected loss factor (Mean squared error here.)

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(P_i - E_i)^2 \quad (10)$$

7) Start the training process by adding the data, selecting batch size and the number of epochs, that is, the number of iterations for which the model should be trained.
8) Check the loss per epoch to check the validity of the training process.
9) Use the window method to use previous values to forecast future values using the trained LSTM model.
10) Validate the forecasted values using the validation parameters such as MAE, MSE, RMSE, and R-square score.

11) If the model overfits the training dataset, add dropout between the input and output layers so that an optimum accuracy can be observed.

$$O_i^h = \sigma(S_i^h) = \sigma\left(\sum_{l<h} \sum_j w_{ij}^{hl}\, \delta_j^l O_j^l\right) \text{ with } O_j^0 = I_j \quad (11)$$

12) Repeat steps 4-10 until optimum accuracy is achieved.
13) End

IV. RESULTS

For this research, a multitude of datasets were utilized for the training process to establish a comparative analysis for computational efficiency and accuracy. Given the sheer size of the datasets, the time required for the initial training process was rather unacceptable for dynamic adjustments of the forecast as per requirements. Thus, the data was trimmed for daily values until only 300 entries were considered. Also, the data was validated for future forecasts instead of a training and testing split of the already available data which reinforced the fact that the model performed well in real-world applications. The validation parameters selected for checking the accuracy of the forecasts are as follows:

1) Mean Absolute Error (MAE): - MAE gives the absolute error between two data points at the same time instance and averages the total error of the time series forecasted by the model versus the actual time series data. MAE is given as:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |(P_i - E_i)| \quad (12)$$

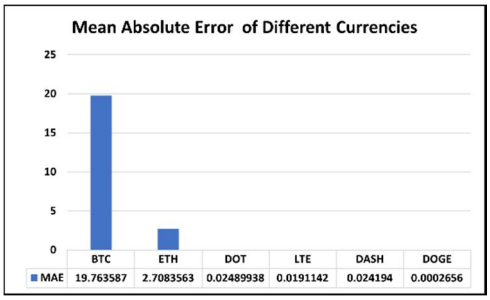


Fig. 9: Mean Absolute Error

2) Mean Square Error (MSE): - MSE squares off all error values, thereby turning all error values strictly into positive and giving the mean of the squared error. This validation parameter serves as the risk assessment of the model and is given is as:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (P_i - E_i)^2 \quad (13)$$

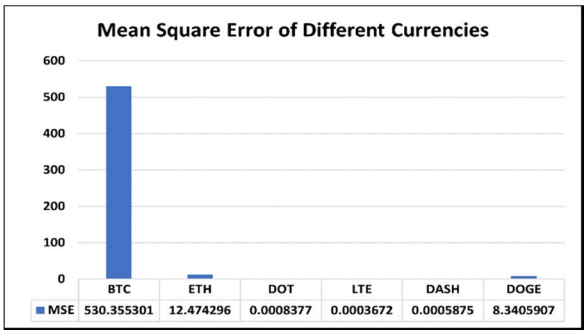


Fig. 10:Mean Squared Error

3) Root Mean Square Error (RMSE): - RMSE takes the squared error at each data point and finds the root of it, to give the absolute positive error of forecast at each time interval. This serves as the validation parameter which gives the numerical accuracy of the model which is given as:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (P_i - E_i)^2} \quad (14)$$

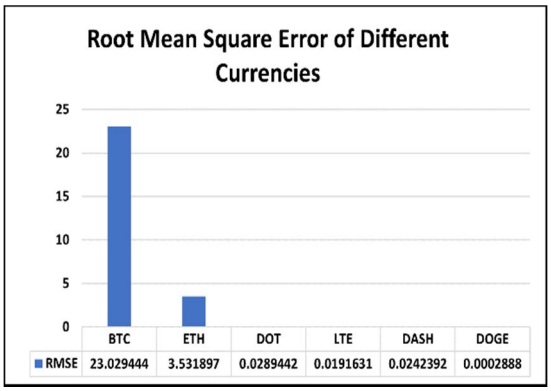


Fig. 11:Root Mean Squared Error

4) R-Squared score: - R-squared score gives how accurately the model observes and replicates the trends and outcomes of the given dataset. This validation pattern, simply, is the proportion of the variation of the dependent data point from the independent data point.

$$\text{R-Squared score} = 1 - \frac{Unexplained\ Variation}{Total\ variation} \quad (15)$$

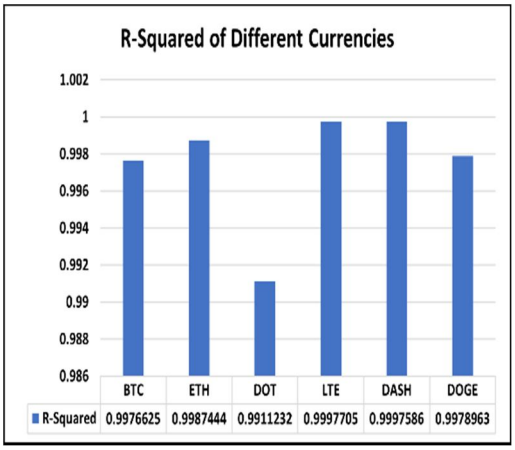


Fig. 12:Comparative Analysis of R-Squared Values

5) Mean Absolute Percent Error (MAPE): - MAPE is a measure of the accuracy of the forecasted values generated by the model versus the actual values and is given as:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right| \quad (16)$$

Table 1:
Comparative Analysis of Validation Parameters for Various Datasets

Dataset	Validation Parameters			
	MAE	MSE	RMSE	R-Squared
Passive – Daily	0.026232	0.0010377	0.0322144	0.9903814
Passive - Min by Min	2.243226e-05	5.032575e-10	2.243340e-05	0.34119011
API – Daily	0.028197	0.0009952	0.03154797	0.9907753
API - Min by Min	1.935327e-06	7.1587463e-12	2.675583e-06	0.9971194
BTC	40.42163	2646.81303	51.447186	0.9995420

Table 2:
Analysis of Validation Parameters for Different Datasets with 86400 Seconds of Interval

Dataset (1 day interval data)	Validation Parameters 84600 sec			
	MAE	MSE	RMSE	R-Squared
BTC	19.763587	530.355301	23.029444	0.9976625
ETH	2.7083563	12.474296	3.5318970	0.9987444
DOT	0.02489938	0.0008377	0.0289442	0.9911232
LTC	0.0191142	0.0003672	0.0191631	0.9997705
DASH	0.0241940	0.0005875	0.0242392	0.9997586
DOGE	0.0002656	8.3405907	0.0002888	0.9978963

Table 3:
Analysis of Validation Parameters for Different Datasets with 21600 Seconds of Interval

Dataset (6 hours interval data)	Validation Parameters 21600 sec			
	MAE	MSE	RMSE	R-Squared
BTC	19.274200	375.05678	19.366382	0.9994073
ETH	4.4695699	20.374614	4.5138248	0.9873639
DOT	0.0154734	0.0002888	0.0169943	0.9941181
LTC	0.0716798	0.0052181	0.0722368	0.9980351
DASH	0.0818824	0.0102443	0.10121457	0.9958372
DOGE	0.0002872	1.0374806	0.0003220	0.9926873

Table 4:
Analysis of Validation Parameters for Different Datasets with 3600 Seconds of Interval

Dataset (1 hour interval data)	Validation Parameters 3600 sec			
	MAE	MSE	RMSE	R-Squared
BTC	38.361219	1471.5854	38.3612488	0.9851676
ETH	2.5272584	6.3969549	2.5292202	0.9854929
DOT	0.0115135	0.00014432	0.0120135	0.9915584
LTC	0.2640865	0.0697764	0.2641522	0.9627080
DASH	0.0597323	0.0035707	0.0597561	0.9959766
DOGE	0.0001138	1.3448379	0.0001159	0.9962996

Table 5:
Analysis of Validation Parameters for Different Datasets with 300 Seconds of Interval

Dataset (5 mins interval data)	Validation Parameters 300 sec			
	MAE	MSE	RMSE	R-Squared
BTC	0.4207271	0.2678939	0.5175846	0.9996836
ETH	0.4652877	0.2887961	0.5373976	0.9781287
DOT	0.0008691	7.6910161	0.0008769	0.9991398
LTC	0.0255787	0.0006782	0.0260433	0.9806422
DASH	0.0038931	2.0514891	0.0045293	0.9984147
DOGE	5.1424962	3.5011014	5.9170105	0.9936249

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

Table 6:
Analysis of Validation Parameters for Different Datasets with 60 Seconds of Interval

Dataset (1 min interval data)	Validation Parameters 60 sec			
	MAE	MSE	RMSE	R-Squared
BTC	0.1091526	0.0121904	0.1104105	0.9999621
ETH	0.5686339	0.32355065	0.5688151	0.9926898
DOT	0.0011558	1.3442484	0.0011581	0.9916389
LTC	0.0070686	4.9973064	0.0070691	0.9897357
DASH	0.0096338	9.3528512	0.0096710	0.9875834
DOGE	2.3092207	5.3443182	2.3117781	0.9708234

From the table, it is observed that the MAE remains minimal between the predicted and real values of the digital currency. The difference is consistent for both the daily datasets and the minute-by-minute datasets acquired from the API and historical data. The same inference is validated by the MSE which highlights the fact that there is minimal risk associated with the estimates and predictions generated by the model. The low MSE value highlights that the mean error is very low for the model. The RMSE further validates the inference that the residuals from the estimator are in fact comparatively very small. As RMSE is a measure of the accuracy of the model, given the low values in RMSE, can highlight the fact that the model is very accurate, closing in on pretty much overfitting the dataset. Finally, the R-square score of the models further validates the visual representation of the results with the only outlier being the minute-by-minute dataset with a low r-square score of 0.34. However, the more significant datasets show a consistently high r-square score of over 0.99. Thus, lowering confidence in the designed LSTM model by the virtue of its overfitting the dataset.

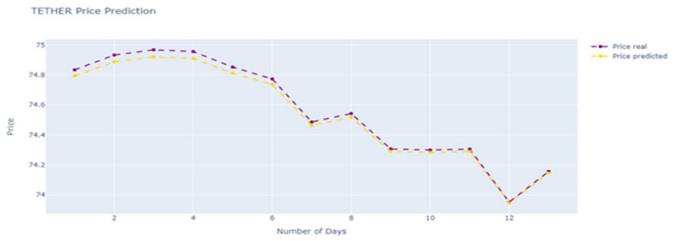


Fig. 13: Passive USDT data-Daily Prediction Graph

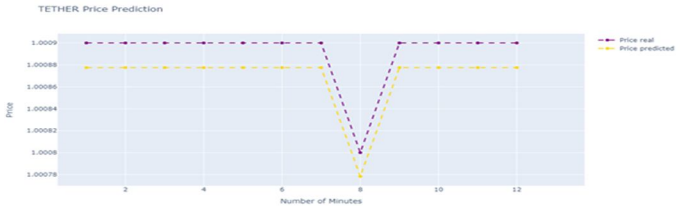


Fig. 14:Passive USDT data-Min by Min Prediction Graph

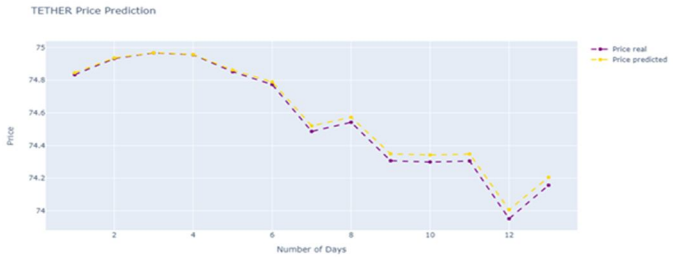


Fig. 15: API USDT data-Daily Prediction Graph

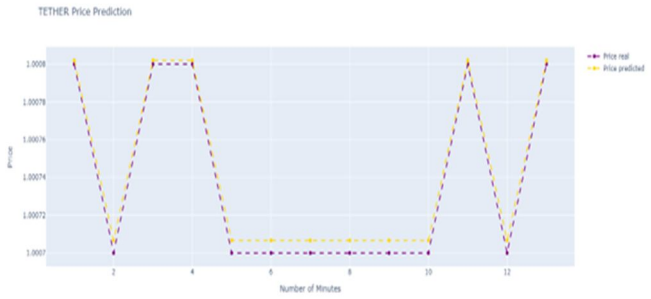


Fig. 16: API USDT data-Min by Min Prediction Graph

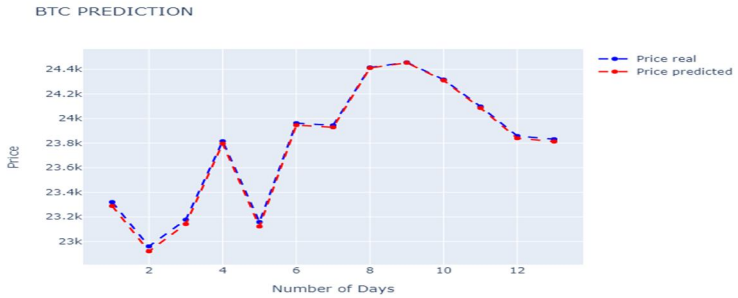


Fig. 17: API BTC data- Daily Prediction Graph

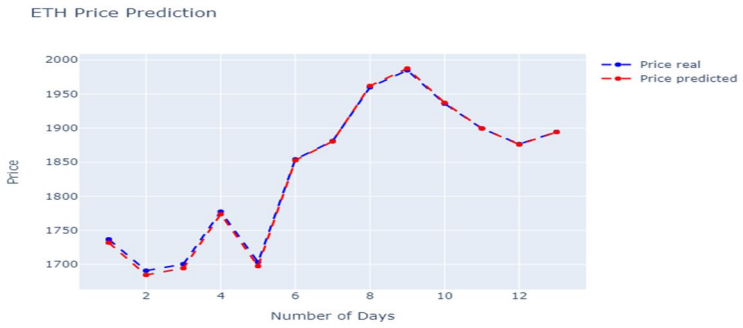


Fig. 18: API ETH data- Daily Prediction Graph

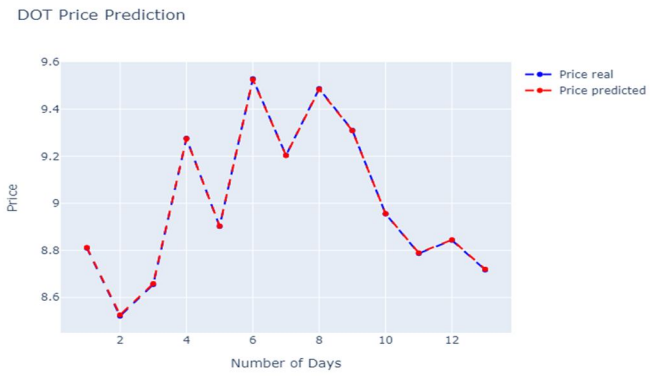


Fig. 19: API DOT data- Daily Prediction Graph

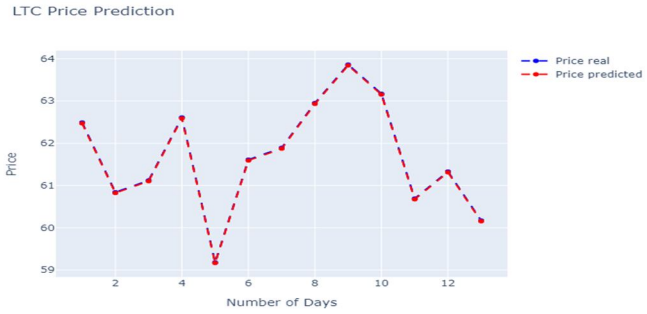


Fig. 20: API LTC data- Daily Prediction Graph

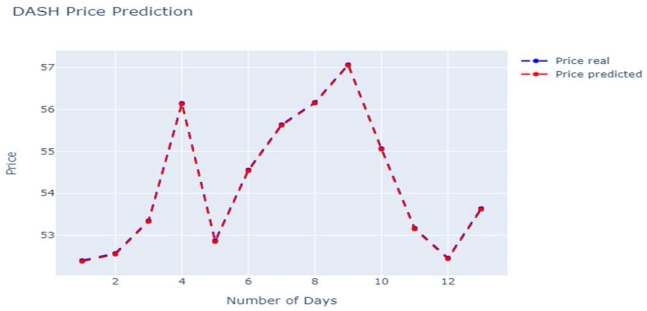


Fig. 21: API DASH data- Daily Prediction Graph

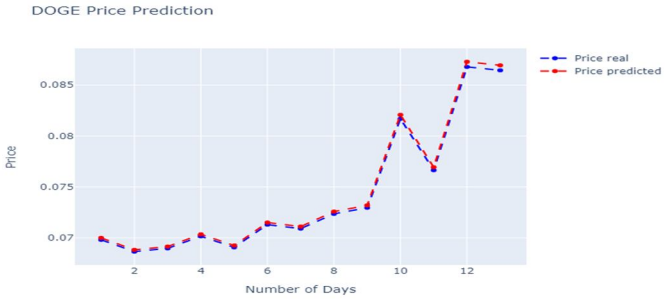


Fig. 22: API DOGE data- Daily Prediction Graph



Fig. 23: API BTC data: 6 Hour Prediction Graph

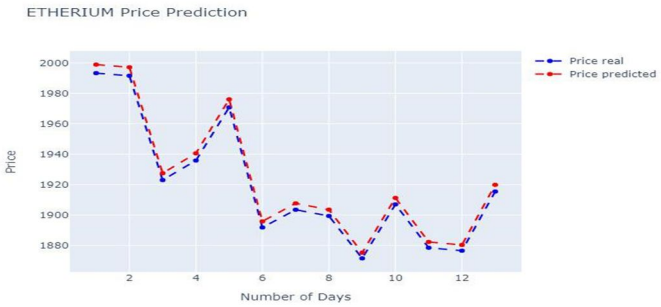


Fig. 24: API ETH data: 6 Hour Prediction Graph

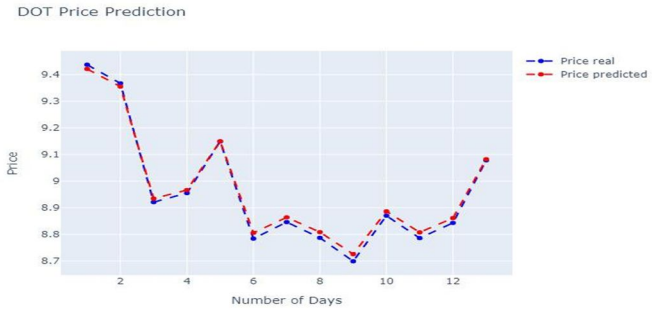


Fig. 25:API DOT data: 6 Hour Prediction Graph

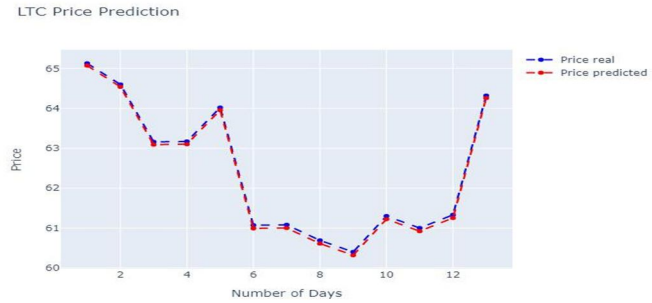


Fig. 26: API LTC data: 6 Hour Prediction Graph



Fig. 27: API DASH data: 6 Hour Prediction Graph



Fig. 28: API DOGE data: 6 Hour Prediction Graph

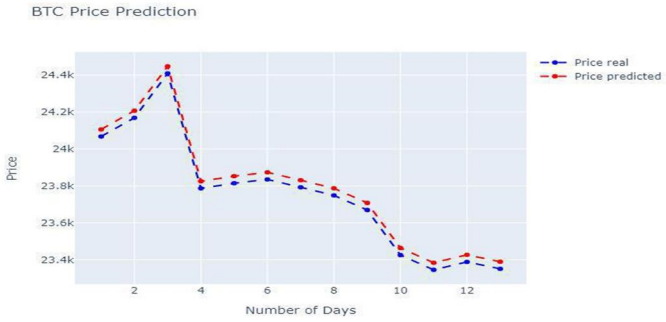


Fig. 29: API BTC data: 1 Hour Prediction Graph

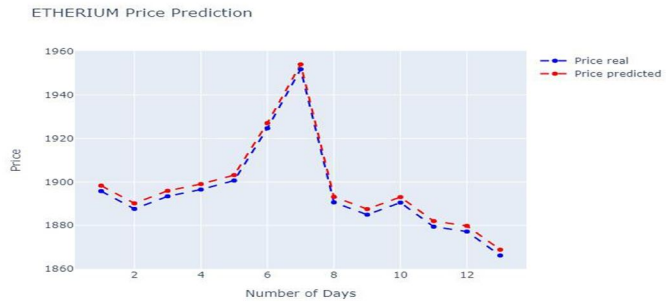


Fig. 30:API ETH data: 1 Hour Prediction Graph

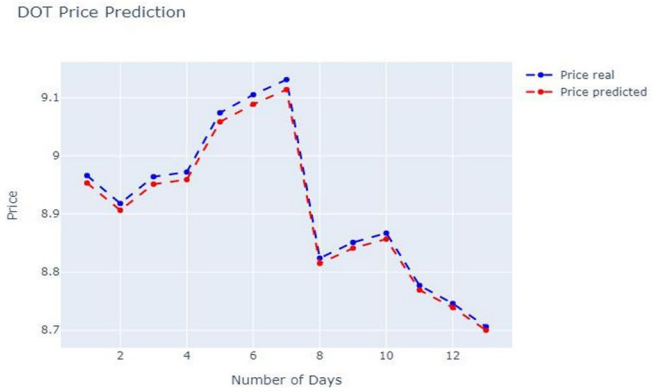


Fig. 31:API DOT data: 1 Hour Prediction Graph

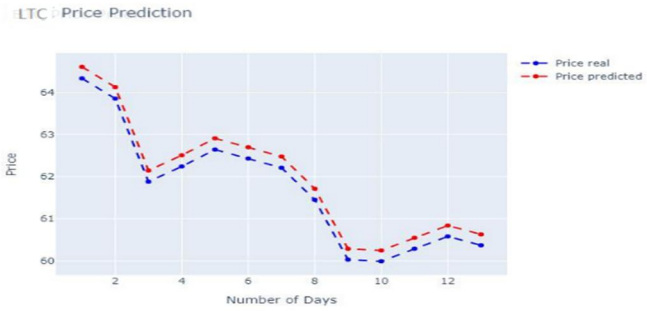


Fig. 32:API LTC data: 1 Hour Prediction Graph

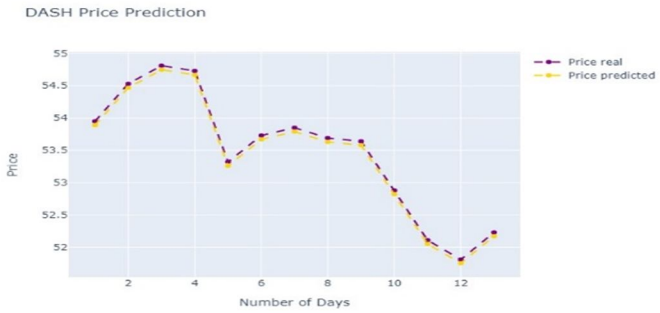


Fig. 33: API DASH data: 1 Hour Prediction Graph

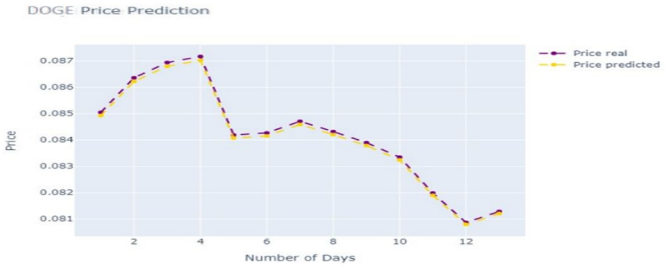


Fig. 34: API DOGE data: 1 Hour Prediction Graph

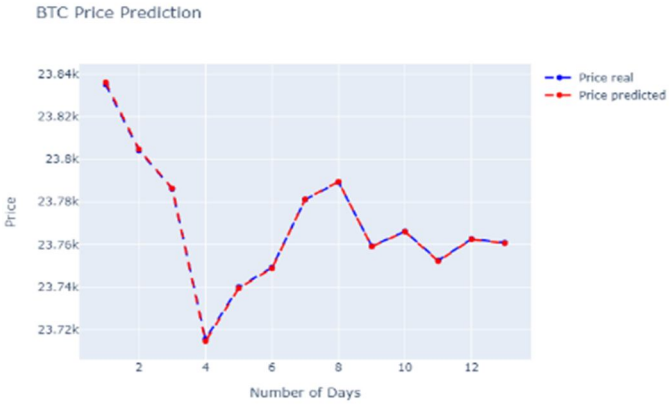


Fig. 35: API BTC data: 5 Mins Prediction Graph

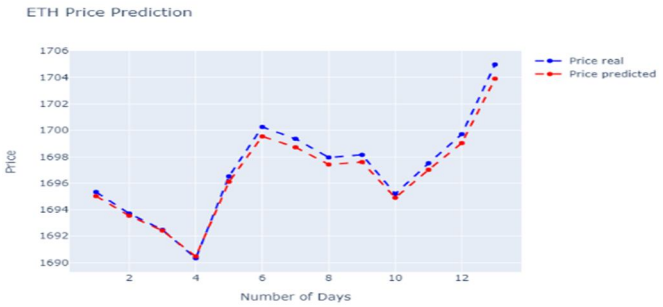


Fig. 35: API ETH data: 5 Mins Prediction Graph

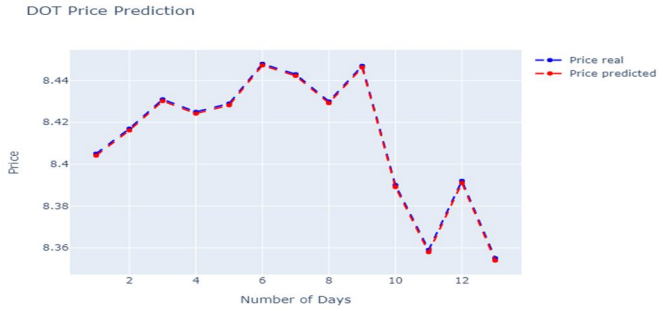


Fig. 367: API DOT data: 5 Mins Prediction Graph

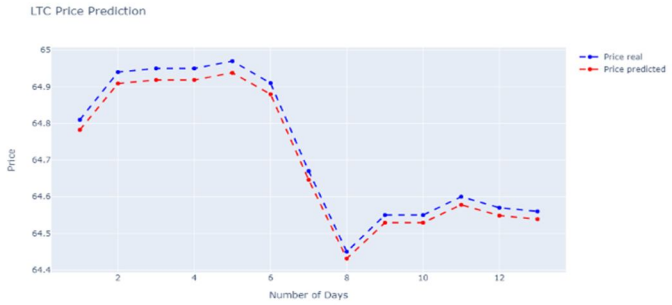


Fig. 38: API LTC data: 5 Mins Prediction Graph

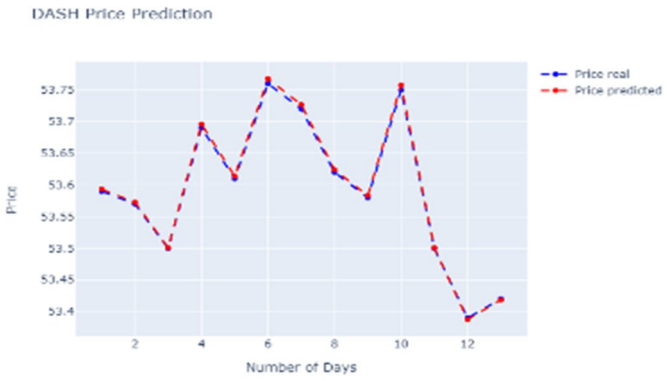


Fig. 39 API DASH data: 5 Mins Prediction Graph

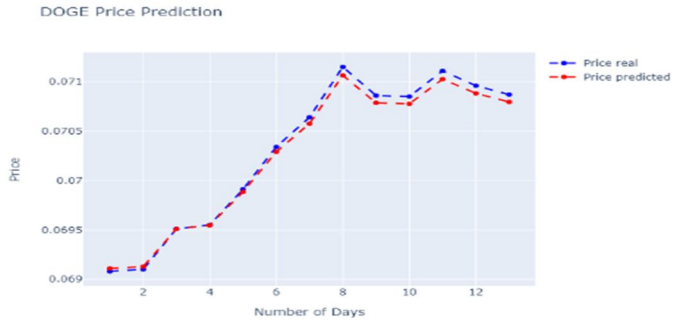


Fig. 40: API DOGE data: 5 Mins Prediction Graph

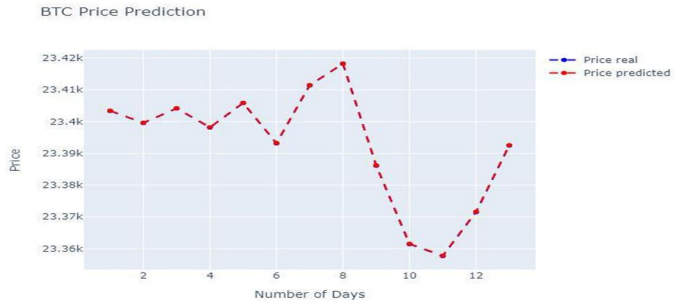


Fig. 41: API BTC data: 1 Min Prediction Graph

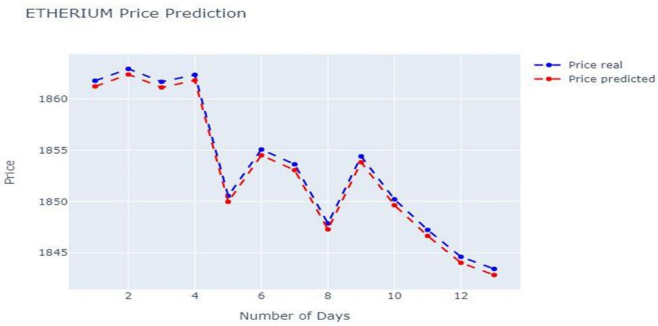


Fig.42: API ETH data: 1 Min Prediction Graph

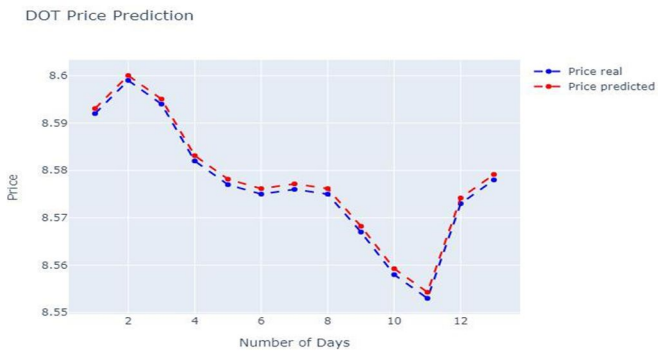


Fig. 43: API DOT data: 1 Min Prediction Graph

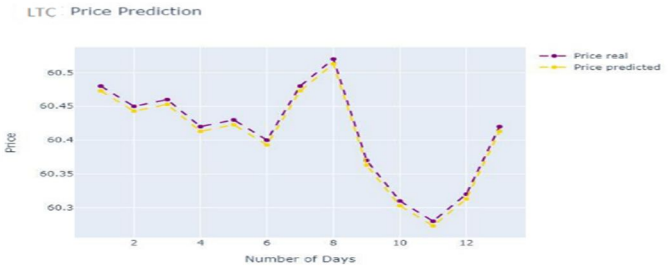


Fig. 44: API LTC data: 1 Min Prediction Graph

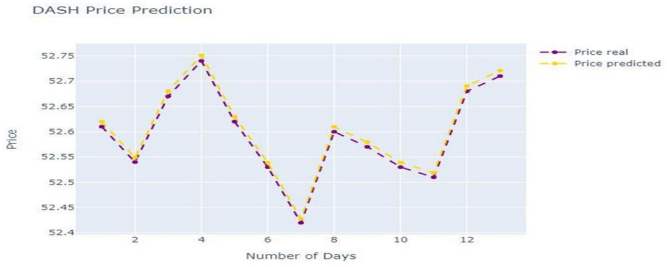


Fig. 45: API DASH data: 1 Min Prediction Graph

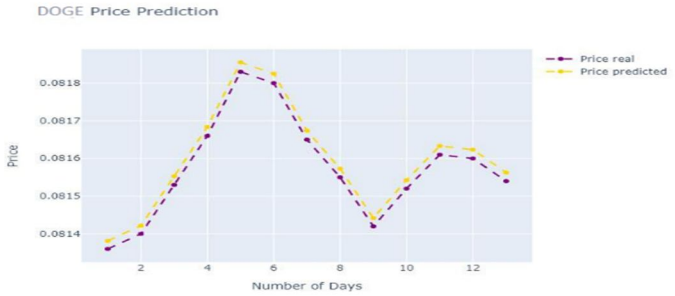


Fig. 46: API DOGE data: 1 Min Prediction Graph

From the results that the progress report of the project, it can be said that the model overfits the dataset. There is an argument to be made that the model overfits the forecasting parameters, with an accuracy of nearly 99.46 percent for different intervals of data using the dataset acquired from the API for BTC, 98.80 percent for ETH, 99.21 percent for DOT, 98.61 percent for LTE, 99.54 percent for DASH, and 99.02 percent for DOGE. However, if we observe the initial dataset itself, it fluctuates within a fine mesh of values, showing a very minimal change in trend and value over an extended period. This can point towards the fact that the dataset itself is very much in range for minimal movement and hence, the changes in values are periodic and pretty much follow a pattern that can easily be picked up by the designed LSTM model.

V. CONCLUSION

After a cohesive analysis of the results, it is established that the deployed LSTM model gives an extremely high degree of accuracy while being efficient in terms of computational resource requirement and time for real-time forecasts when deployed for real-time data analysis and subsequent forecasts. The low degree of error observed for predictions in terms of Mean Absolute Error, Mean Square Error, Root Mean Square Error, and the high degree of accuracy established by the R-square score of 99.97 percent for Tether (USDT), 99.63 percent for Bitcoin (BTC), 99.91 percent for Ethereum (ETH), 98.51 percent for Polkadot (DOT), 99.96 percent for Litecoin (LTE), 98.62 percent for Dash Coin (DASH), 99.65 percent for Dogecoin (DOGE) for daily intervals means that the selected algorithm and machine learning model can be used with modifications for real-time applications as per the requirements of the end-user.

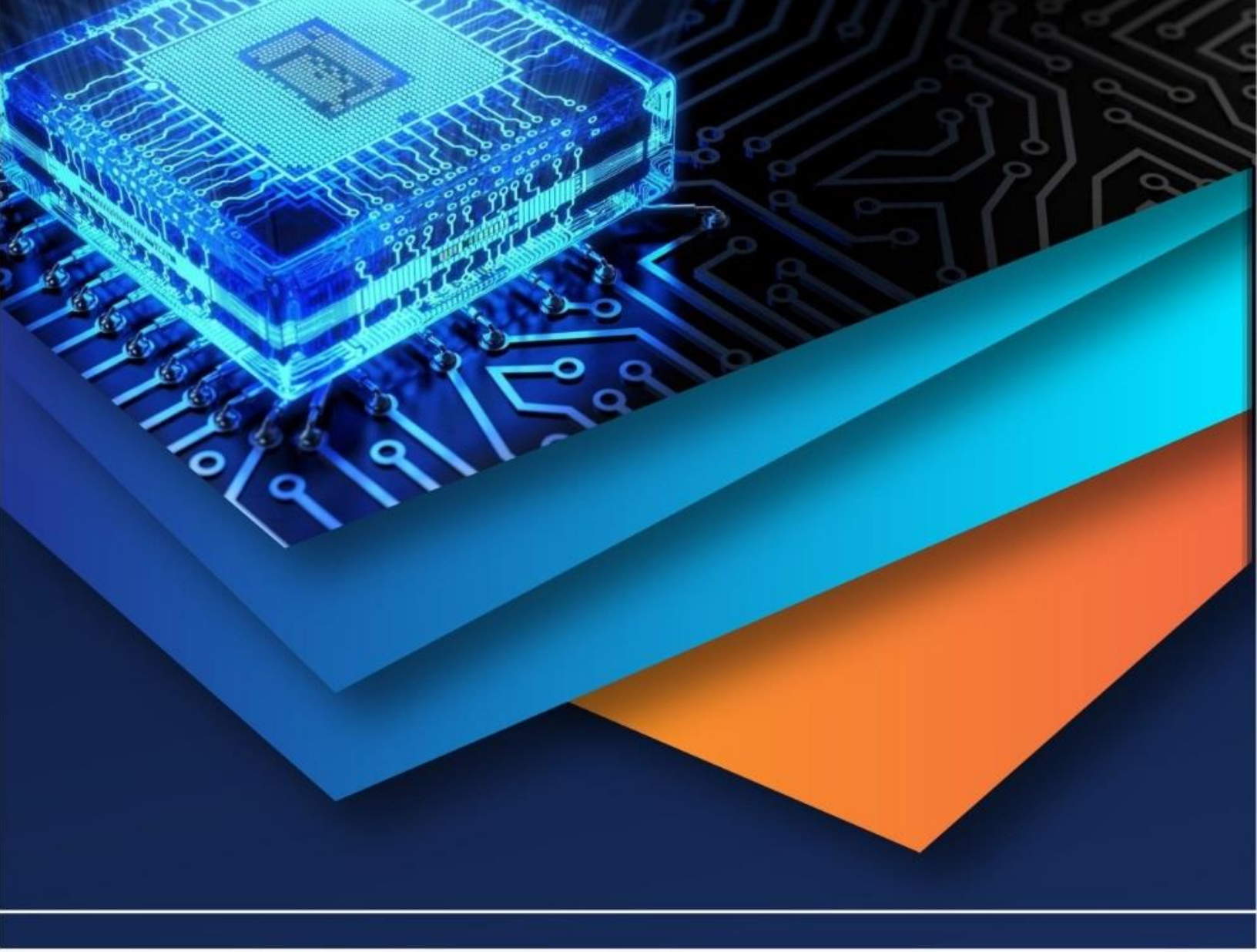
©IJRASET: All Rights are Reserved

We also received excellent results when adjusting the dataset's time intervals: 86400 seconds, or one day, 21600 seconds, or six hours, 3600 seconds, or an hour, 300 seconds, or five minutes, and 60 seconds, or one minute, with an R-square score of 95 to 99 percent high accuracy. Through rigorous testing, it can be concluded that for achieving such a high score in terms of validation for daily closing value forecasts, the LSTM model requires a minimum of 1500 data points for training, which can be adjusted dynamically throughout which the model needs to forecast as and when the model is run. This research can be further extended to various other datasets such as the stock market, other digital currencies in the financial domain. With the advent of newer machine learning models, a comparative study of computation time, accuracy, and size of data required for higher accuracy for forecasts can be made.

REFERENCES

[1] Shefali Arora, Ruchi Mittal, and M P S Bhatia. Automated cryptocurrencies prices prediction using machine learning collaborative approach for trend analysis using clustering mechanisms and big data technologies. Article in International Journal of Soft Computing, page 4, 2018.
[2] Temesgen Awoke, Minakhi Rout, Lipika Mohanty, and Suresh Chandra Satapathy. Bitcoin price prediction and analysis using deep learning models. volume 134, pages 631–640. Springer Science and Business Media Deutschland GmbH, 2021.
[3] Seema A Bhalegaonkar et al. Automated metaphase chromosome image selection techniques for karyotyping: Current status and future prospects. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(6):3258–3266, 2021. [4]
[4] Krishna Chakravarty, Manjusha Pandey, and Siddharth Routaray. Bitcoin prediction and time series analysis, 2020. [5]
[5] Jaiswal Rupesh Chandrakant and D Lokhande. Comparative analysis using bagging, logitboost and rotation forest machine learning algorithms for real time internet traffic classification. [6]
[6] Jaiswal Rupesh Chandrakant and D Lokhande. Statistical features processing based real time internet traffic recognition and comparative study of six machine learning techniques. [7]
[7] Jaiswal Rupesh Chandrakant and D. Lokhande Shashikant. Analysis of early traffic processing and comparison of machine learning algorithms for real time internet traffic identification using statistical approach. volume 28, pages 577–587. Springer Science and Business Media Deutschland GmbH, 2014. [8]
[8] Reshma Sundari Gadey, Nikita Thakur, Naveen Charan, and R Obulakonda Reddy. Price prediction of bitcoin using machine learning, 2020. [9]
[9] Guus Van Heijningen. Making predictions in highly volatile cryptocurrency markets using web scraping, 2017. [10]
[10] Alvin Ho, Ramesh Vatambeti, and Sathish Kumar Ravichandran. Indian journal of science and technology bitcoin price prediction using machine learning and artificial neural network model. Indian Journal of Science and Technology, 14:2300, 2021. [11]
[11] Rupesh Jaiswal and Shashikant Lokhande. Rupesh jaiswal and shashikant lokhande: A novel approach for real time internet traffic classification a novel approach for real time internet traffic classification. [12]
[12] Rupesh Jaiswal, Shashikant Lokhande, Aashiq Ahmed, and Prateek Mahajan. Performance evaluation of clustering algorithms for ip traffic recognition, 2013. [13]
[13] Rupesh Jaiswal, Shashikant Lokhande, and Aditya Gulavani. Implementation and analysis of dos attack detection algorithms, 2013. [14]
[14] Patrick Jaquart, David Dann, and Christof Weinhardt. Short-term bitcoin market prediction via machine learning. The Journal of Finance and Data Science, 7:45–66, 11 2021. [15]
[15] Danish Khan and Rupesh C Jaiswal. Issue 11 www.jetir.org (issn-2349-5162), 2020.
[16] Shreya Mondhe, Mayank Mukundam, and R C Jaiswal. Issue 6 www.jetir.org (issn-2349- 5162), 2019. [17]
[17] M Munot, M Joshi, and Nikhil Sharma. Automated karyotyping of metaphase cells with touching chromosomes. Int J Comput Appl, 29(12):14–20, 2011. [18]
[18] Mousami V Munot, Jayanta Mukherjee, and Madhuri Joshi. A novel approach for efficient extrication of overlapping chromosomes in automated karyotyping. Medical & biological engineering & computing, 51(12):1325–1338, 2013. [19]
[19] Shilpa Nair. Cryptocurrencies price movement prediction using machine learning, 2021. [20]
[20] Institute of Electrical, Electronics Engineers. India Council, Institute of Electrical, Electronics Engineers. Bombay Section, Annual IEEE India Conference 10 2013.12.13-15 Mumbai, Annual Conference of the IEEE India Council 10 2013.12.13-15 Mumbai, and INDICON 10 2013.12.13-15 Mumbai. Annual IEEE India conference (INDICON), 2013 13-15 Dec. 2013, Mumbai, India. [21]
[21] Sarika A Panwar, Mousami V Munot, Suraj Gawande, and Pallavi S Deshpande. A reliable and an efficient approach for diagnosis of brain tumor using transfer learning. Biomed Pharmacol J, 14:283–294, 2021. [22]
[22] Aashay Pawar and R C Jaiswal. Stock market study using supervised machine learning, 2020. [23]
[23] Chen Peng and Guo Yichao. Isss608 visual analytics and applications cryptocurrency price analysis and time series forecasting group 7, 2020. [24]
[24] Prajwal Pitlehra, R C Jaiswal, and Associate Professor. Credit analysis using k-nearest neighbours model, 2021. [25]
[25] Ajith Premarathne, Malka N., R. Samarakody, and Ampalavanapillai Nirmalathas. Real-time cryptocurrency price prediction by exploiting iot concept and beyond: Cloud computing, data parallelism and deep learning. International Journal of Advanced Computer Science and Applications, 11, 2020. [26]
[26] Vojtech Pulec. Cryptocurrency returns: short-term forecast using google trends, 2019. [27]
[27] Vladimir Puzyrev. Deep convolutional autoencoder for cryptocurrency market analysis. 10 2019. [28]
[28] Lekkala Sreekanth Reddy and DrP Sriramya. A research on bitcoin price prediction using machine learning algorithms. [29]
[29] Jaiswal Rupesh and Lokhande D Shashikant. Measurement, modeling and analysis of http web traffic. [30]
[30] Jacopo De Stefani, Olivier Caelen, Dalila Hattab, Yann Aël LeÂ Borgne, and Gianluca Bontempi. A multivariate and multi-step ahead machine learning approach to traditional and cryptocurrencies volatility forecasting. volume 11054 LNAI, pages 7–22. Springer Verlag, 2019. [31]
[31] Franco Valencia, Alfonso Gómez-Espinosa, and Benjamín Valdés-Aguirre. Price movement prediction of cryptocurrencies using sentiment analysis and machine learning. Entropy, 21:589, 6 2019.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)