ISSN 2278-3091
Volume 9, No.4, July – August 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse132942020.pdf
https://doi.org/10.30534/ijatcse/2020/132942020

An Approach to Cryptocurrency Price Prediction using Deep Learning

Gopalakrishnan T[1], Ruby D[2], Gayathri A[3]. Saai Mahesh[4], Ritesh Choudhary[5]
[1, 2,4] School of Computer Science and Engineering, Vellore Institute of Technology, Vellore
gopalakrishnan.t@vit.ac.in, ruby.d@vit.ac.in, srinivasansaai.mahesh2016@vitstudent.ac.in
[3]School of Information and Technology, Vellore Institute of Technology, Vellore.
gayathri.a@vit.ac.in
[5]School of Computing and Information Technology, Manipal University Jaipur,
ritesh.choudhary456@gmail.com

ABSTRACT

Cryptocurrency such as Bitcoin, Ethereum etc. is increasingly well known nowadays among cryptocurrency miners and enthusiasts. Our goal is to estimate the various cryptocurrency accurately considering various parameters that impact the cost. We use various deep learning and machine learning algorithms to be able to recognize the value pattern on the closing value thus giving us the predicted price. The point of this is to determine the precision of cryptocurrency values utilizing various AI algorithms and predict their estimated prices. The dataset that we will be using contains consolidated financial information for the top 10 cryptocurrencies sorted by Market Cap which also have various attributes for each type of cryptocurrency such as open, close, high, low, value and date. Therefore, we are going to use Linear Regression, SGD Regression, Support Vector Regression, and LSTM algorithms to predict the various cryptocurrencies prices.

Key words : Cryptocurrency, Deep Learning, Machine Learning, Long Short Term Memory, SGD Regression, Linear Regression, Support Vector Regression.

1. INTRODUCTION

There has been a huge boost in popularity of cryptocurrencies due to several months of vast exponential growth of their market capitalization of around $300 Billion in 2020. This has led to cryptocurrency receiving a lot of attention from cryptocurrency enthusiasts and media. However, with the decentralization and open sourced nature of cryptocurrencies, it has reduced the level of central control over them which led to various other types of cryptocurrencies being created and introduced into the market. Correspondingly, this causes a wide range of differences in the cryptocurrency prices for between cryptocurrencies which indicate a important requirement for price prediction. This paper proposes the numerous ways to predict cryptocurrency price by inducing various factors such open price, high price and close price through deep learning techniques such as recurrent neural network (RNN) - long short-term memory (LSTM), Linear Regression, Stochastic Gradient Descent (SGD), Random Forest Regression, Support Vector Regression which are known effective machine learning models for training data, with LSTM(Long Short Term Memory) being recognized for longer term predictions. This proposed approach implements Python as the programming language and uses dataset that contains consolidated financial information for the top 10

cryptocurrencies sorted by Market Cap which also have various attributes for each type of cryptocurrency such as open, close, high, low, value and date. The cryptocurrencies used are; Tezos which is a technology that deploys a blockchain capable of modifying its set of rules with minimal disruption to its network through an on-chain governance model; Bitcoin which works without a central bank or administrator that can be sent from user to user on its peer-to-peer Bitcoin network; Ethereum works through on an operating system featuring smart contract functionality; XRP (Ripple)works as a fast, cost efficient cryptocurrency develop for cross-border payments; Binance Coin (BNB) is the cryptocurrency of the Binance platform which is a global cryptocurrency exchange; EOS is designed to support large-scale applications; Tether is a cryptocurrency that has a value which matches the value of the U.S. dollar; Bitcoin Cash is cryptocurrency based on a fork of Bitcoin which is a spin-off that was created similar to Bitcoin SV which is a fork of Bitcoin Cash (BCH), it attempts to restore the original Bitcoin protocol as defined by version 0.1 of Bitcoin; Stellar cryptocurrency is used to fiat money transfers which allows for cross-border transactions; Litecoin cryptocurrency enables instant, near-zero cost payments; Cardano cryptocurrency is the first blockchain platform to evolve out of from a scientific philosophy and is based on a research driven approach. The results that we accomplish from the project help us understand and examine the various types of model applicability of accurate prediction of cryptocurrency prices further facilitating the performances of the proposed deep learning prediction models.

The aim of the proposed work is to use various deep learning, machine learning algorithms such as LSTM, Linear Regression, SGD Regression, Support Vector Regression, Random Forest Regression to predict the cryptocurrency prices in the dataset for a certain day and perform comparative analysis of the models to identify the best model for our application that include deriving the MSE values for the models.

The objectives of the proposed work is to identify the parameters such as high, open and close within the dataset for various types of cryptocurrencies and use it accurately to compute the cryptocurrency price through various deep learning algorithms of LSTM, Linear Regression, SGD Regression, Support Vector Regression, Random Forest Regression. Once the price prediction is achieved, comparative analysis can be performed between all the models to identify the most accurate model that best fits the practicality of using it within our application. This provides us the feasibility to understand the requirements of the best model and it's working for the of real-world cryptocurrency price prediction. We use RMSE which helps to measure the differences between values predicted by a model and the values observed. The predicted output of the price can be evaluated within the bound of open and high values to get us a suitable prediction.

2. LITERATURE SURVEY

Detailed Gorse and Phillips [1] worked on the prediction of cryptocurrency prices bubbles by using social media data along with epidemic modeling. Through using Hidden Markov model (HMM) to detect their bubble-like characteristics in the time series they concluded that the social media data plays an important role in the prediction of cryptocurrency prices.

Zheshi Chen, Wenjun Sun and Chunhong Li [2] proposed to predict Bitcoin prices by classifying by their respective everyday price and high frequency based prices SnehaGullapalli [3] proposed the perform prediction based on the trained temporal neural networks such as time-delay neural networks and recurrent neural networks on prices of Bitcoin over few years. Parameters such as the opening price, highest price, lowest price, closing price and volume of its currency were taken into consideration so as to predict the highest and closing price of the next day.

Scheuermann and Tschorsch [4] carried out a technical survey based on decentralized cryptocurrencies. Their research was based on the building blocks and protocols that correspond to the cryptocurrency known as Bitcoin; this highlights the characteristics of centralized cryptocurrencies, and various other findings. They have given in-depth insights of Bitcoin cryptocurrency. Their detailed work can be linked to understand various cryptocurrencies.

Mukhopadhyay [5] proposed a study based on cryptocurrency systems. The study consisted of several characteristics of cryptocurrencies, that comprises proof of stake, the proof of pork, and their combination that was used in data mining techniques. They elaborated on how the proof of stake is not self-dependent, while the proof of work is resource dependent. Hence, the combination of this can result in precise results. Bruno Spilak [6] proposed the usage of Multilayer Perceptron, Recurrent Neural Network and Long Short-Term Memory models to predict the output of priced directions of cryptocurrencies that uses rolling window regression method. They constructed a classification problem that predicts if the price of the cryptocurrency will either increase or decrease based on a three-month trading strategy which will then compared to its performance with a passive index investment that relies on cryptocurrency market which follows CRIX

The gaps that have been identified after reviewing the aforementioned research/work is that there was either one or two types of cryptocurrency being used for their price prediction. This limits the model's efficiency and accuracy for cryptocurrency price prediction to the unique parameters of that specific cryptocurrency type. So, in our project, we propose to find common parameters (open, high and close values) for ten different types of cryptocurrency train the models to these parameters and then predict their price. This facilitates to produce the efficiency of the deep learning[10] and machine learning models by performing analysis such as using MSE on the price predictions byperforming analysis such as using RMSE on the price predictions by the various models implemented.

3. OVERVIEW OF PROPOSED SYSTEM

3.1 Introduction and Related Concepts

Cryptocurrency is a virtual currency created to serve for online monetary needs. Crypto is prefix that originates from the method of using cryptography methodologies to verify and secure transactions which help produce new cryptocurrency units. The principle of cryptography is of that it can make information decipherable with a key than without one. Blockchain is composed of transactions that are made by cryptocurrencies on a peer to peer network. This consists of many different nodes which ensure that counterfeiting does not take place. Transactions between accounts and cryptocurrency wallets can be identified easily, thus ensuring security. There are a lot of advantages for cryptocurrencies due to its control over transactions and inflation. Investors can be seen using cryptocurrencies as assets for their portfolios. Cryptocurrencies are placed in the market of non-correlation that creates it as a potential hedge against risk for example, valuable metals such as platinum, diamond etc.

Django is used as a web based framework that is used for rapid deployment and development. It's free and open source and Django was designed to assist developers to take applications from ideas to execution quickly.

Django is also highly secure and helps developers to avoid common security risks. Popular websites around the world use

Django for their platform. Framework, Architecture or Module for the Proposed System.

3.2 Linear Regression

The Linear Regression models are used to predict the relationship between two variables. The variable that is being predicted is called the as dependent variable. The variables that are used to predict the value of the dependent variable are called the independent variables. In Linear Regression, each observation consists of two values. One value is for the dependent variable and one value is for the independent variable. In this simple model, a straight line approximates the relationship between the dependent variable and the independent variable. We have used the sklearn.linear_model to import Linear Regression algorithm and implement it for our application. For our approach, we fit the Linear Regression algorithm with the scaled trained X and Y data and predict the value for the X test scaled data. Using the inverse transform function, we transform the predicted value and the Y test scaled values. We generate a loss function for Linear Regression using the Mean Squared Error for the predicted value and the Y test scaled value and save the generated model through pickle. Figure 3 shows the mathematical model of Linear Regression.

3.3 Stochastic Gradient Descent (SGD) Regression

Stochastic Gradient Descent (SGD) is an easy and methodical way of studying linear classifiers under convex loss functions like the Support Vector Machines and Logistic Regression. The SGD Regressor integrates a stochastic slope learning routine that supports different loss functions and setbacks in an appropriate linear regression model. SGD Regressor is accustomed in such a way that it has many training samples for regression problems. We use sklearn.linear_model to import SGD Regressor algorithm and implement it for our application. For our approach, we fit the SGD algorithm with the scaled trained X and Y data and predict the value for the X test scaled data. Using the inverse transform function, we transform the predicted value and the Y test scaled values. We generate a loss function for SGD Regression using the Mean Squared Error for the predicted value and the Y test scaled value and save the generated model through pickle. Figure 4 shows the mathematical model of SGD Regression.

3.4 Random Forest Regression

A Random Forest is a method which has the capacity of executing both regression and classification problems by the use of several decision trees. It has an ability called Bootstrap Aggregation, which is also recognized as Bagging. Bagging incorporates numerous decision trees for deciding a definitive final output which relies on singular decision trees. We use sklearn.ensemble to import Random Forest Regressor algorithm for our application. For our approach, we fit the Random Forest Regression algorithm with the scaled trained X and Y data and predict the value for the X test scaled data. Using the inverse transform function, we transform the predicted value and the Y test scaled values. We generate a loss function for Random Forest Regression using the Mean Squared Error for the predicted value and the Y test scaled value and save the generated model through pickle. Figure 5 shows the mathematical model of Random Forest Regression.

3.5 Support Vector Regression:

This is a type of support vector machine which assists in linear and non-linear regression. SVR requires the training data's X and Y values where in it generates a correlation matrix using the X and Y values and then trains the model. We get a contraction coefficient which gives us the predicted value. For our approach, we fit the Support Vector Regression algorithm with the scaled trained X and Y data and predict the value for the X test scaled data. Using the inverse transform function, we transform the predicted value and the Y test scaled values. We generate a loss function for Support Vector Regression using the Mean Squared Error for the predicted value and the Y test scaled value and save the generated model through pickle. Figure 6 shows the mathematical model of Support Vector Regression.

3.6 Long Short-Term Memory

Long Short-Term Memory is a division of a recurrent neural network (RNN). In RNN, the resultant product of the last step is used as the input in the present step[8]. This is used to solve the problem of long-term dependencies of its usage. As time increases, RNN does not provide systematic performance however LSTM can by default preserve the desired information for a long time. It is used for processing, estimating and organizing by arranging it on the basis of time series data. We use tensorflow.keras.layers to import LSTM, Dense algorithms and tensorflow.keras.models to import Sequential, load_model, model_from_json from where our approach consists of two layers of LSTM input layers for representing an elliptical function for influencing the flow of information and memorization of patterns created inside a cryptocurrency data. The Adam optimizer is used to repeatedly upgrade the network weights for the purpose of training it. The dense layer is used for making the model more accurate. This is done by using one LSTM layer and one dense layer. LSTM[16] layers give their complete output sequences back, and the input sequence is converted into a single vector by the dense layer. We generate a loss function for LSTM using the Mean Squared Error for the predicted value and the Y test scaled value and save the generated model in .json and .h5 file formats. Figure 7 shows the mathematical model of LSTM.

When you submit your final version, after your paper has been accepted, prepare it in two-column format, including figures and tables.

4. DATASET DESCRIPTION

For our dataset, we use the consolidated financial information for the top ten cryptocurrencies that was pulled from the site www.CoinMarketCap.com[7]. Their attributes include: Currency name (e.g. Bitcoin), Date, Open, High, Low, Close, Volume, Market Cap[9,11]. The descriptions of the attributes are:

a. **Currency**: Name of cryptocurrency.
b. **Date**: Date refers to the calendar date for the particular row i.e. 24 hours from midnight to midnight.
c. **Open**: Open is what the price was at the beginning of the day.
d. **High**: Highest recorded trading price of the day.
e. **Low**: Lowest recorded trading price of the day.
f. **Close**: Close is what the price was at the end of the day.
g. **Volume**: Volume represents the monetary value of the currency traded in a 24 hours period, denoted in USD.
h. **Market Cap**: Market cap is circulating supply 'x' price of the coin.

Table 1: Attribute description for the cryptocurrency dataset

	Currency	Date	Open	High	Low	Close	Volume	Market Cap
count	28944	28944	28944	28944	28944	28944	28944	28944
unique	12	2412	12307	12057	12803	12294	16349	16058
top	stellar	Feb 26, 2015	1	1	1	1	0	4,51,600
freq	2412	14	1725	1511	1367	1729	2916	394

5. PROPOSED SYSTEM MODEL

If the proposed system model can be understood through the following phases:

a. **Data Analysis Phase**: This step examines the data and its parameters wherein it checks for any redundancy in the values of the data which might affect the predicted results. In case the dataset has any insignificant parameters, then the values of the data are erased. We improve the model predictability by merging the data.

b. **Data Filtration Phase**: The data is filtered so that it can get rid of all unimportant values.

c. **Train Test Split Phase**: In this phase the data is divided into subsets of training and testing. The division of these training and testing subsets are in the ratio of 70% and 30% respectively.

d. **Data Scaling Phase**: The data is scaled accordingly based on the parameters needed in the model. The data scaling makes it compatible for its use in the model.

e. **Model Building Phase**: We have implemented this paper using Python and its libraries such as sklearn, keras that contain the required models in Python that can be imported directly such as the sklearn.linear_model package that gives Linear and SGD regression, sklearn.ensemble gives the Random Forest Regression, sklearn.svm gives Support Vector Regression and keras gives LSTM. We cannot we use these models directly to build a LSTM model. Hence we have used Keras which implements tensorflow as a backend library that makes the model accurate. The model of Keras is composed of LSTM layer and dense layer. The data is processed in these layers to identify and form patterns present in the dataset so that the model becomes more accurate. The model then takes in this data for training.

f. **Model Learning and Evaluation Phase**: In this phase, the algorithm is fitted with the X and Y trained scaled data that is open, high and close values and it is formatted to get a better output.

g. **Prediction Phase**: In this phase we implement the prediction based on the model that has been generated in which the input values are fed into the model to give the cryptocurrency price prediction. The accuracy and losses are calculated by comparing the test data to the resultant output and losses through the MSE function.

1) Mathematical Models: This is used to elaborate a system's mathematical model by its formulation and theories. This method of creating a mathematical model is called as mathematical modeling. The following are the mathematical models of the algorithms that have been discussed in the earlier section.

a. Linear Regression Mathematical Model

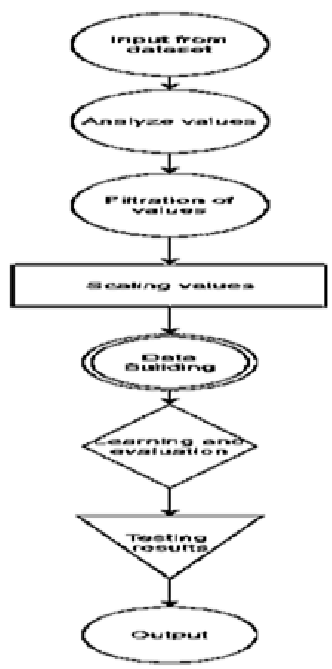

Figure 1: Workflow of proposed approach

X and Y are the variables that are used in Linear Regression. Linear Regression's equation elaborates about the relation between X and Y and is called as the regression model. It is represented by the equation y = β0 + β1x+ ε, where, the y-intercept is known by β0, the slope is known as β1 and the mean is known as E(y). The Linear Regression models represented 'ε'. β0 and β1 are used to represent the parameters for the population studies[12,15].

b. SGD Regression Mathematical Model

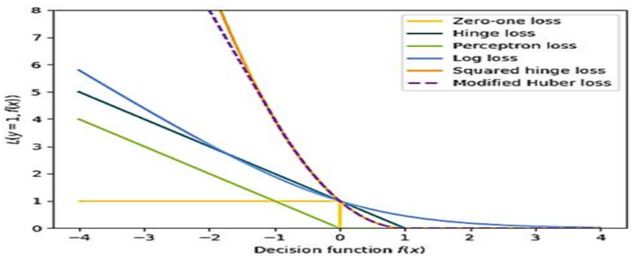

Figure 2: Mathematical model for SGD Regression [13]

The SGD Regression is based on various gradient descent functions as shown in the figure above.

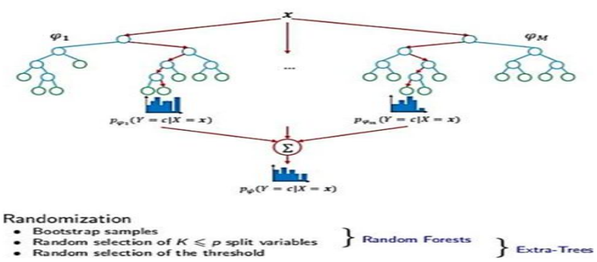

Figure 3: Mathematical model for Random Forest [14]

Random Forest Regression works by selecting random K data points(K) from the training data subset which is then used to build the decision tree based on these data points, we then choose 'N' number of trees that we require to build and the previous steps are repeated. When there is a newly created data point we create each one of the N data trees which are used to predict the value of Y for the given data point. The new data point is then assigned to represent as an average for all the Y predicted values.

d. Mean Squared Error (MSE) Mathematical Model

MSE is a mathematical model which is the average squared error that occurs in a loss function in minimal squares regression. It is a combination of all data points such that it is also the square of the difference between the variables of predicte a points are split and used. It is thr determine the loss of the model.

$$\sum_{i=1}^{n} \frac{\left(w^T x(i) - y(i)\right)^2}{n} \quad (1)$$

e. Pearson's Correlation Coefficient Mathematical Model

The linear correlation between the two variables X and Y is given by the Pearson's Correlation Coefficient.

$$r = \frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\sqrt{[n\Sigma x^2 - (\Sigma x)^2][n\Sigma y^2 - (\Sigma y)^2]}} \quad (2)$$

Use Case Representation

In the below Use Case Diagram, the User launches the Django Server or command prompt window to run the interface of the application. The user is it to then input the cryptocurrency name, date, open value, high value and should then chose the preferred algorithm. Taking in all these input parameters the system computes the given data to the model and presents the predicted value for the given cryptocurrency.

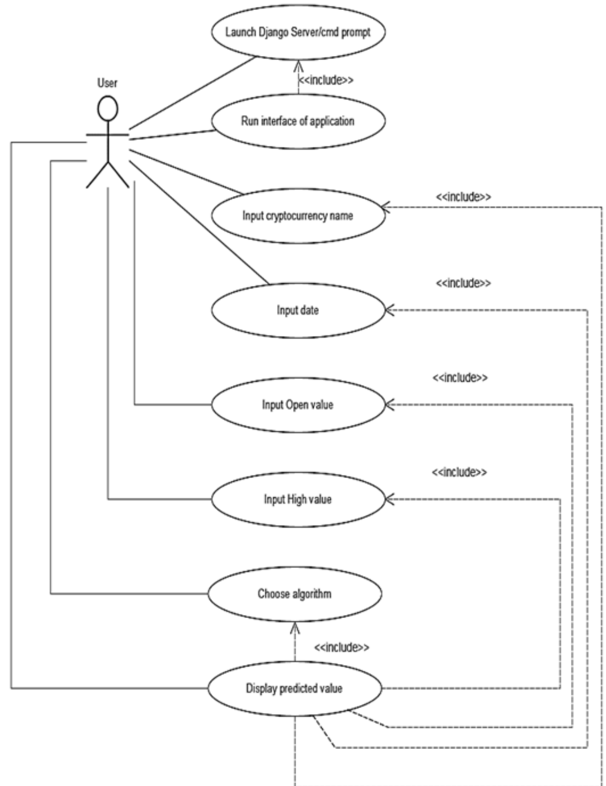

Figure 4: Use Case Representation for proposed system

6. RESULTS AND DISCUSSION

In our results, the model that we have implemented using our datasets has given us our predicted cryptocurrency prices. The Pearson correlation coefficient was used to find the features from the dataset where correlations were calculated between Open - Low, Open - Low, Open - High values that show the impact as the deciding factors of our input and output values. We also find the correlation between all the parameters in the dataset as seen in figure 6.

Table 2: Attribute Correlation for Cryptocurrency

	Open	High	Low	Close	Volume	Market Cap
Open	1.000000	0.999268	0.998868	0.998551	0.560012	0.953655
High	0.999268	1.000000	0.998588	0.999403	0.561073	0.954372
Low	0.998868	0.998588	1.000000	0.999205	0.559649	0.954388
Close	0.998551	0.999403	0.999205	1.000000	0.560458	0.955007
Volume	0.560012	0.561073	0.559649	0.560458	1.000000	0.591818
Market Cap	0.953655	0.954372	0.954388	0.955007	0.591818	1.000000

In Table 2, the attribute correlation gives us the amount of linear correlation between two types of parameters, X and Y. Here, the higher the correlation coefficient, the greater is their linear correlation.

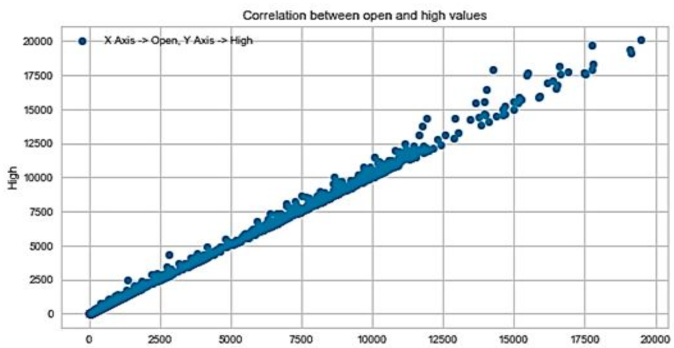

Figure 5: Correlation between open and high values

In figure 5, the correlations between Open and High values are observed to be highly correlated with noticeable dispersion after the 12500 and 12500 of Open and High values respectively. The Pearson correlation coefficient was found to be 0.999268.

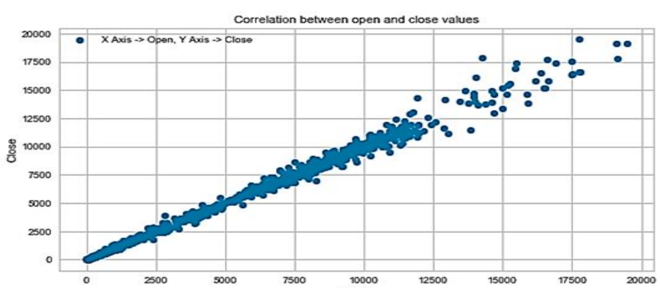

Figure 6 : Correlation between open and close values

In figure 6, the correlations between Open and Close values are observed to be highly correlated with high dispersion after the 15000 and 15000 of Open and Close values respectively. The Pearson correlation coefficient was found to be 0.998551.

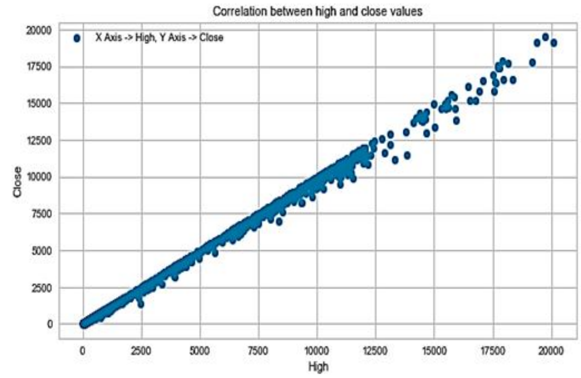

Figure 7: Correlation between high and close values

In figure 7, the correlations between High and Close values are observed to be highly correlated with high dispersion after the 12500 and 12500 of High and Close values respectively. The Pearson correlation coefficient was found to be 0.999403.

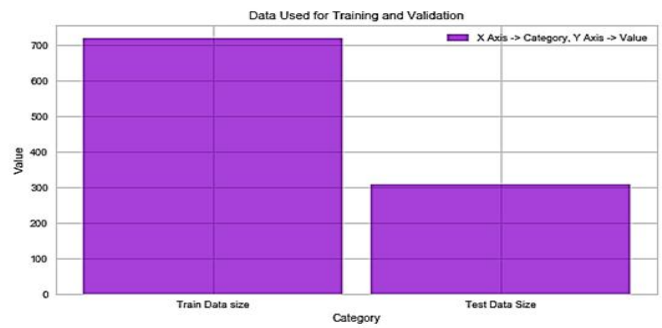

Figure 8: Train and test data sizes

In figure 8, for our implementation, we have split the data for our training and testing to the ratio of 70% and 30% respectively. We've used the random state of 1 in which an internal number has been randomly generated using the attribute of random state which decides how the training data and test data gets split.

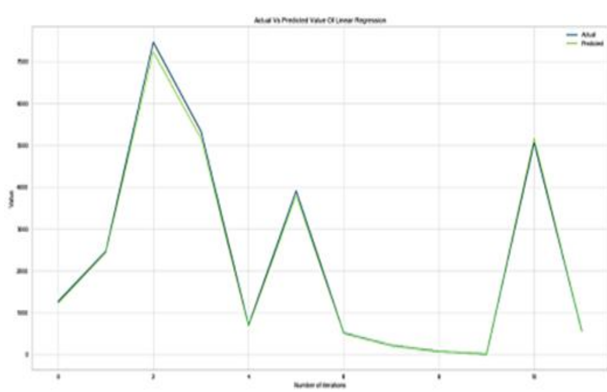

Figure 9: Actual vs predicted values of Linear Regression Model

In the figure 9, we have used a blue line to depict the actual value and green line to depict the predicted values where it can be seen that the graph is uniform this giving us nearly approximate predicted closing values and can be far off from the actual value if the values given are higher. Mean Squared Error loss of Linear Regression was found to be 0.00125.

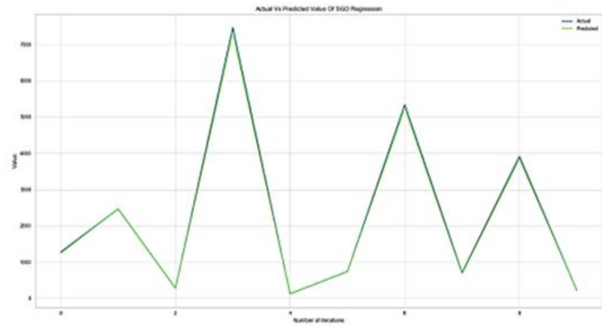

Figure 10: Actual vs predicted values of SGD Regression

In the figure 10, we have used a blue line to depict the actual value and green line to depict the predicted values where it can be seen that the graph is uniform this giving us nearly approximate predicted closing. Mean Squared Error loss of SGD Regression was found to be 0.00148.

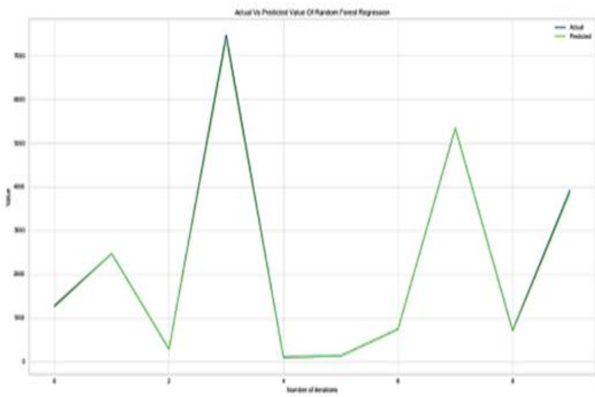

Figure 11: Actual vs predicted values of Random Forest Regression

In the figure 11, we have used a blue line to depict the actual value and green line to depict the predicted values where it can be seen that the graph is uniform this giving us nearly approximate predicted closing. Mean Squared Error loss of Random Forest Regression was found to be 0.00139.

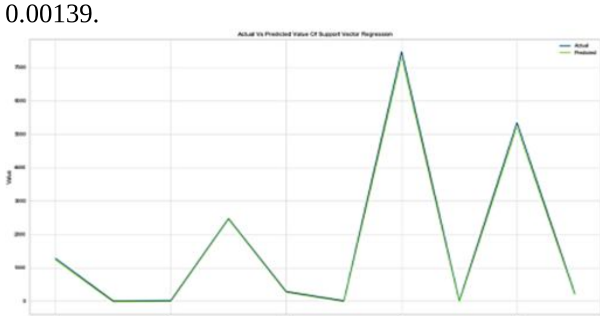

Figure 12: Actual vs predicted values of Support Vector Regression

In the figure 12, we have used a blue line to depict the actual value and green line to depict the predicted values where it can be seen that the graph is uniform this giving us nearly approximate predicted closing. Mean Squared Error loss of Support Vector Regression was found to be 0.00160.

In the figure 13, we have used a blue line to depict the actual value and green line to depict the predicted values where it can be seen that the graph is uniform this giving us nearly approximate predicted closing values and can be far slightly off from the actual value due to the sudden change in the values which can happen randomly based on how LSTM works. Mean Squared Error loss of LSTM was found to be 0.00111, which makes it the lowest compared to all the other models.

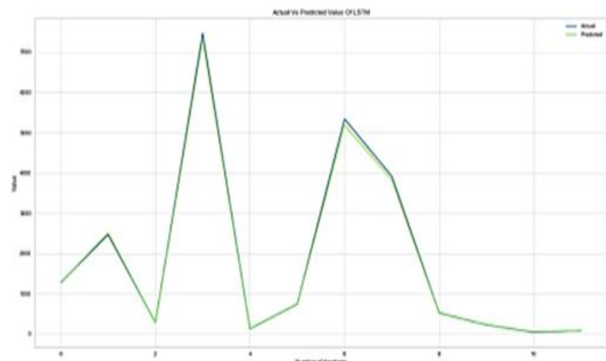

Figure 13: Actual vs predicted values of LSTM

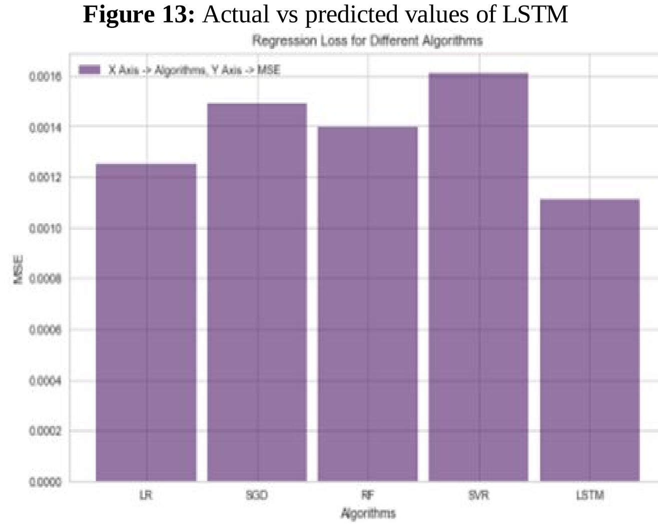

Figure 14 : Regression loss for all to MSE

Therefore, in comparison to all the models the figure 14 shows the difference between all the implemented models. As shown, LSTM proves to have the lowest MSE making it the most accurate.

In the figure 15, we see the user's interface that has been developed through Django from which they can enter the input values of currency name, date, open and high price and the preferred algorithm.

Figure 15: User interface through Django

In figure 16, the output from the workings of figure 16 is displayed for a selected algorithm.

Figure 16: Predicted cryptocurrency price

We have additionally added the feature of predicting the cryptocurrency price directly through the command line prompt which doesn't need the use of a web browser. Furthermore, since we can't implement .sav files for LSTM which makes it compatible for Django, we should use the command line prompt for predicting using LSTM.

7. CONCLUSION

Cryptocurrencies has weakened the central control over them by the years. Alongside with this, there has been variations and drastic changes in their prices which require an important need to predict cryptocurrency prices. This paper proposes several methods which are used to predict cryptocurrency prices by utilizing several parameters such as open, high and low values from the benchmark dataset. We have implemented several algorithms such as Linear Regression, SGD Regression, Support Vector Regression, Random Forest Regression and LSTM in Python. The results show the optimality and practicality of the proposed approach to reach the expected cryptocurrency price prediction as we train the models for several types of cryptocurrencies thus proving its applicability. Future research can extend to implementing a specific date from which the cryptocurrency price gets predicted.

REFERENCES

1. Phillips, RC; Gorse, D, **Predicting cryptocurrency price bubbles using social media data and epidemic modelling**. *In: Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (SSCI).* pp. 394-400, 2018
2. Zheshi C, Chunhong L, Wenjun S, **Bitcoin price prediction using machine learning: An approach to sample dimension engineering**, *Journal of Computational and Applied Mathematics*, Volume 365, Issue 2020(02), 2019
3. Sneha Gullapalli, **Learning to predict cryptocurrency price using artificial neural network models of time series**,https://krex.kstate.edu/dspace/bitstream/handle/2097/38867/SnehaGullapalli2018.pdf
4. F. Tschorsch and B. Scheuermann, **Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies**, in IEEE Communications Surveys & Tutorials, vol. 18, no. 3, pp. 2084-2123, thirdquarter 2016, doi: 10.1109/COMST.2016.2535718.
5. 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*, Auckland, 2016, pp. 745-752, doi: 10.1109/PST.2016.7906988.
6. Bruno Spilak, **Deep neural networks for cryptocurrencies price prediction** Retrieved from https://d-nb.info/1185667245/34, 2018
7. Yecheng Y, Jungho Y, Shengjun Z, Yuwen L, Taekseung K, Guihongxuan Z, Leonard Y, **Predictive Analysis of Cryptocurrency Price Using Deep Learning**, International Journal of Engineering & Technology, 7 (3.27) (2018). https://doi.org/10.14419/ijet.v7i3.27.17889
8. Https://www.geeksforgeeks.org/deep-learning-introduction-to - long-short-term-memory/
9. Https://www.kaggle.com [https://www.kaggle.com/philmohun/cryptocurrency-financial-data].
10. AbhinavSagar, **Cryptocurrency Price Prediction Using DeepLearning**, [https://towardsdatascience.com/cryptocurrency-price-prediction-using-deep-learning-70cfca50dd3a
11. Marco Santos, **Predicting Bitcoin Prices with Deep Learning**" [https://towardsdatascience.com/predicting-bitcoin-prices-with-deep-learning-438bc3cf9a6f].
12. [Https://madhureshkumar.wordpress.com/2015/07/21/regression-analysis-linear-regression-sse-assumption-of-linear-regression-error-term-and-best-fit-line].
13. [Https://scikit-learn.org/stable/modules/sgd.html]
14. [https://www.kdnuggets.com/2017/10/random-forests-plained.html]
15. [Https://stats.stackexchange.com/questions/13194/support-vector-machines-and-regression]
16. [https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory.