Bitcoin Price Prediction using Deep Learning

Sulochana Devi
Department of Information Technology
Xavier Institute of Engineering
Mumbai, India
sulochana.d@xavier.ac.in

Sheldon Noronha
Department of Information Technology
Xavier Institute of Engineering
Mumbai, India
sheldonnoronha26@gmail.com

Aditya Jagtap
Department of Information Technology
Xavier Institute of Engineering
Mumbai, India
adityajagtap38@gmail.com

Sahil Desai
Department of Information Technology
Xavier Institute of Engineering
Mumbai, India
sahild2809@gmail.com

Abstract — Bitcoin is a decentralized digital currency created in January 2009..We will go deep into the datasets, do an EDA, feature extraction and predict the price of bitcoin using Stochastic, Machine Learning and Deep Learning models. In this particular work we will be visualising the Time Series data, handling the missing values with various imputation techniques. Feature extraction is performed before model building, the focus of study will be these four models ARIMA, Facebook prophet, XG boost and LSTM. Comparing the models and their evaluation metrics to see how each model have performed

Keywords: Blockchain, Bitcoin, Cryptocurrency, Neural Network.

I. INTRODUCTION

Bitcoin is a cryptographic currency which is utilized worldwide for advanced installment or basically for speculation purposes. Bitcoin is decentralized, for example it isn't possessed by anybody. Exchanges made by Bitcoins are simple as they are not attached to any nation. Speculation should be possible through different commercial centers known as "bitcoin trades". These enable individuals to sell/purchase Bitcoins utilizing various monetary forms.

Bitcoin emerged out of the 2008 global economic crisis when big banks were caught misusing borrowers' money, manipulating the system, and charging exorbitant fees. To address such issues, Bitcoin creators wanted to put the owners of bitcoins in-charge of the transactions, eliminate the middleman, cut high interest rates and transaction fees, and make transactions transparent. They created a distributed network system, where people could control their funds in a transparent way.

However, there are issues with bitcoins such as hackers breaking into accounts, high volatility of bitcoins, and long transaction delays. Considering the volatility it's always challenging to predict the bitcoin price.

In this study, we have used models such as ARIMA, LSTM, FB prophet, XG boost to predict the price

II. LITERATURE REVIEW

Here the data of BTC per minute was gathered and it was rearranged so that to reflect BTC price in hours, and a total of 56,832 points.24 hrs of data was taken as input and output for the BTC price of the next hour. For this the Multi-Layer Perceptron (MLP) was the unsuitable case for predicting price based on current trends whereas Long Short-Term Memory (LSTM) gave the best prediction when the past memory and Gated Recurrent Network (GRU) was included.[1]

So here to predict BTC price movement and prices in short and medium terms, High performance ML-based classification and regressions are demonstrated. In this the work goes beyond that by using machine learning-based models for one, seven, thirty and ninety days compared to the previous ML-based classification which has only one-day time frames. The models which are developed have 65% accuracy for the next-day forecast and 62-64% for seventh-ninetieth day forecast, while the error percentage is as low as 1.44% while it varies from 2.88-4.10% for horizons of seven to ninety days.[2]

So here firstly what affects the BTC value is taken into consideration. Here there are two phases in the first phase is to understand and identify the daily trends in the BTC market while gaining insight where the data sets consists of various features relating to the BTC price and payment network over the course of five years daily records and in the second phase using the available information it will be possible to predict the sign of daily price change with highest possible accuracy.[3]

This paper looks over user comments in online communities to predict the transactions taking place in crypto. It was predicted that the prices would fluctuate at low costs. The method used approved buying crypto currencies, and gave information on aspects influencing user decisions. Moreover, the simulated investment demonstrated the methods used are applicable while trading in crypto. [4]

In this paper a tree-based classifier is used to perform the predictionDecision Tree Classification method.In their analysis they implement decision tree learning based on cross-entropy impurity function optimization. They perform 1,000 independent rounds of training and prediction.In each of the rounds they select the last 510 sample points as test data and return the win ratio of each type of prediction and they try to clip the trading cutoff to check the real trading performance of the predictive model. [5]

The study revolves around forecasts of crypto currency particular bitcoin price using algorithms like

ARIMA-autoregressive integrated moving average and NEAR neural network autoregressive models.
Using the static forecast approach they have tried to forecast next day bitcoin prices both with and without re-estimation of the forecast model. [6]

III. PROPOSED WORK

In the system that we have implemented, we have taken the data from January, 2012. This data is within 1 minute intervals. Four types of ML models have been used: ARIMA, Facebook prophet, XG boost and LSTM.

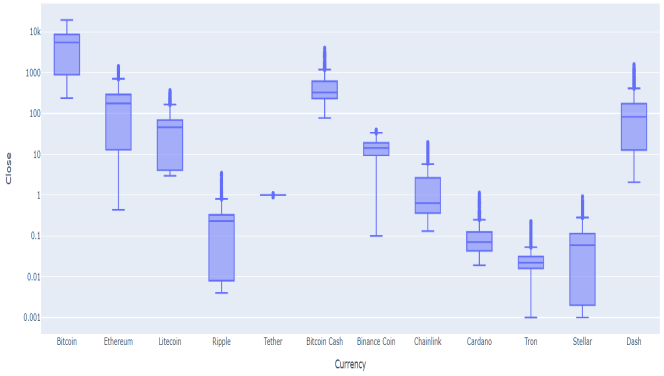


Fig 1: Comparison of various Cryptos

At first we compared various cryptos using a dataset that consisted of various coins like Bitcoin, Dash, Etherium, Cardano, etc which were mapped with each other alongside columns such as Date, Open, High, Close, Adj Close, Volume, etc. We then plotted various pyplots, box plots, pie charts, violin plots in order to analyze all the currencies and figured out that bitcoin would be ideal to work upon because of the volume that it is traded in and its stability.

Methodology:

Data Collection:
In this study, we are focusing on the time-series forecast of BTC prices using machine learning. A time-series is a set of data values with respect to successive moments in time. Time-series forecast is the forecast of future behavior by analyzing time-series data. First we collect all the Data from the "Bitcoin Historical Data" dataset which is available on Kaggle. Included here is historical bitcoin market data at 1-min intervals for select bitcoin exchanges where trading takes place. It consists of a time period of Jan 2012 to September 2020, with minute to minute updates of OHLC (Open, High, Low, Close), Volume in BTC and indicated currency, and weighted bitcoin price.The Open and Close columns indicate the opening and closing price on a particular day. The High and Low columns provide the highest and the lowest price on a particular day, respectively. The Volume column tells us the total volume of traded on a particular day.The Weighted price is a trading benchmark used by traders that gives the weighted price a security has traded at throughout the day, based on both volume and price. It is important because it provides traders with insight into both the trend and value of a security.

Missing Values:

We replaced the missing values by using three imputation techniques which are 'ffill' or 'pad', 'bfill' or 'backfill' and the linear interpolation method. In the 'ffill' method, the NaNs are replaced with the observed value. In the 'bfill' method, the NaNs are replaced with the next observed value. By using these two methods, a fair portion of missing values are filled. In order to fill the remaining values, we use the linear interpolation method. It is an imputation technique that assumes a linear relationship between data points and utilises non-missing values from adjacent data points to compute a value for a missing data point. Thus, using these three methods, we observe no null values in our dataset.

Exploratory Data Analysis:

1. Visualizing the weighted price using markers:
When working with time-series data, a lot can be revealed through visualizing it. It is possible to add markers in the plot to help emphasize the specific observations or specific events in the time series.

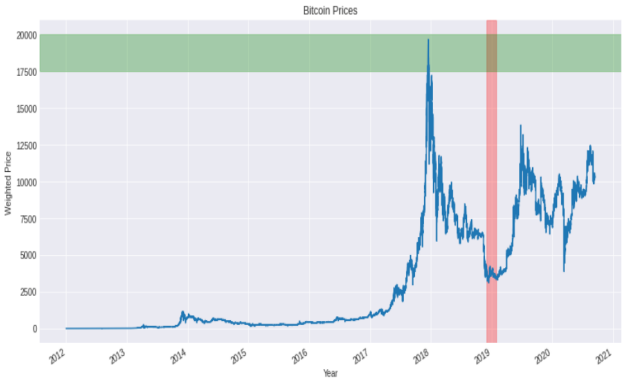


Fig 2: Weighted prices v years

2. Visualizing using KEDs:
Summarizing the data with Density plots to see where the mass of the data is located.

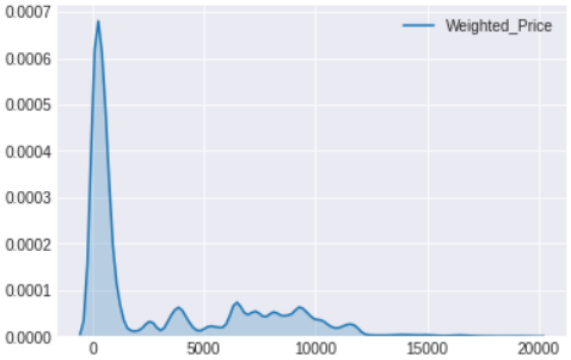


Fig 3: Visualizing KEDs

3. Visualizing using Lag Plots:
Lag plots are used to observe the autocorrelation. These are crucial when we try to correct the trend and stationarity and we have to use smoothing functions. Lag plot helps us to understand the data better.

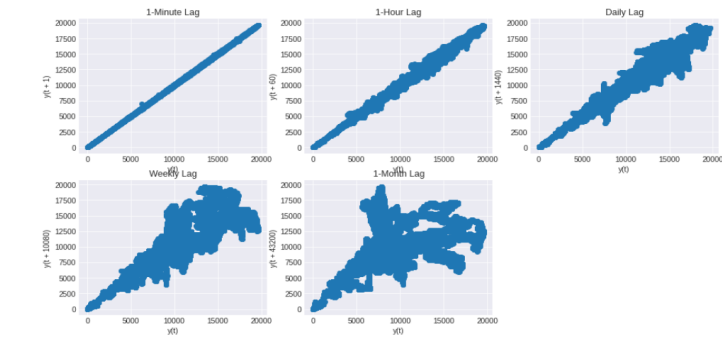


Fig 4: Lag plots

Thus, We can see that there is a positive correlation for minute, hour and daily lag plots. We observed absolutely no correlation for month lag plots.Thus, we resampled our data to utmost the daily level to preserve autocorrelation.

4. Time resampling:

As we noticed the correlation in the above diagram, we decided to resample the data in an Hourly format. Thus we replaced the price of the particular hour by the mean of all the minute prices in that hour.

5. Time Series Decomposition:

We can decompose a time series into trend, seasonal and remainder components. The series can be decomposed as an additive or multiplicative combination of the base level, trend, seasonal index and the residual.Then,we performed some statistical tests like KPSS and Augmented Dickey-Fuller tests to check stationarity.

Fig 5: Time series Decomposition

Post time series decomposition we don't observe any seasonality. Also, there is no constant mean, variance and covariance, hence the series is Non Stationary.

6. KPSS test:

The KPSS test, short for Kwiatkowski-Phillips-Schmidt-Shin (KPSS), is a type of Unit root test that tests for the stationarity of a given series around a deterministic trend.

```
Test Statistics : 0.9719743430417129
p-value : 0.01
Critical Values : {'10%': 0.119, '5%': 0.146, '2.5%': 0.176, '1%': 0.216}
Series is not Stationary
```

In this test, the null hypothesis is that the series is Stationary. As we got a p-value of 0.01(<0.05), we reject the null hypothesis. Hence, we came to the conclusion that the series is not stationary.

7. ADF test:

The only difference here is the Null hypothesis which is just opposite of KPSS. The null hypothesis of the test is the presence of unit root, that is, the series is non-stationary.

```
Results of Dickey-Fuller Test:
Test Statistic                  -1.257922
p-value                          0.648197
#Lags Used                      29.000000
Number of Observations Used    3151.000000
Critical Value (1%)              -3.432427
Critical Value (5%)              -2.862458
Critical Value (10%)             -2.567259
dtype: float64
Series is Stationary
```

Fig 6: Result of Dickey-fuller test

Thus ADF says that the series is stationary. As KPSS says it is not stationary, we came to the conclusion that the series is difference stationary.

Feature Extraction:

Time series data can be noisy due to high fluctuations in the market. As a result, it becomes difficult to gauge a trend or pattern in the data.As we're looking at daily data, there's quite a bit of noise present. It would be nice if we could average this out by a week, which is where a rolling mean comes in.A rolling mean, or moving average, is a transformation method which helps average out noise from data. It works by simply splitting and aggregating the data into windows according to function, such as mean(), median(), count(), etc. For this example, we'll use a rolling mean for 3, 7 and 30 days.Our data becomes a lot less noisy and more reflective of the trend than the data itself.

Model building:

To measure the performance of our forecasting model, We typically want to split the time series into a training period and a validation period. This is called fixed partitioning.

We'll train our model during the training period, we'll evaluate it during the validation period. Here's where you can experiment to find the right architecture for training. And work on it and your hyper parameters, until you get the desired performance, measured using the validation set. Often, once you've done that, you can retrain using both the training and validation data.And then test during the test(or forecast) period to see if your model will perform just as well.

And if it does, then you could take the unusual step of retraining again, using also the test data. The test data is the closest data you have to the current point in time. And as such it's often the strongest signal in determining future values. If your model is not trained using that data, too, then it may not be optimal.

1. ARIMA model:

An autoregressive integrated moving average, or ARIMA, is a statistical analysis model that uses time series data to either better understand the data set or to predict future trends[7].ARIMA is an acronym that stands for AutoRegressive Integrated Moving Average. It is a class of models that captures a suite of different standard temporal

structures in time series data.This acronym is descriptive, capturing the key aspects of the model itself. Briefly, they are:

- AR: Autoregression A model that uses the dependent relationship between an observation and some number of lagged observations.
- I: Integrated The use of differencing of raw observations (e.g. subtracting an observation from an observation at the previous time step) in order to make the time series stationary.
- MA: Moving Average A model that uses the dependency between an observation and a residual error from a moving average model applied to lagged observations.

2. Facebook Prophet:
Facebook Prophet is an open-source algorithm for generating time-series models that uses a few old ideas with some new twists. It is particularly good at modeling time series that have multiple seasonalities.[8]Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects.

It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well.

3. XG Boost:
XGBoost is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. In prediction problems involving unstructured data (images, text, etc.) artificial neural networks tend to outperform all other algorithms or frameworks.[8]

4. LSTM:
Long short-term memory (LSTM) is an artificial recurrent neural network (RNN) architecture used in the field of deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. It can process not only single data points (such as images), but also entire sequences of data (such as speech or video). For example, LSTM is applicable to tasks such as unsegmented, connected handwriting recognition, speech recognition and anomaly detection in network traffic or IDSs (intrusion detection systems).[9]

V. RESULTS

In this section, we present the results of the models used to predict the bitcoin prices.

1. ARIMA model:

The Auto ARIMAX model did a fairly good job in predicting the bitcoin price given data till the previous day. After applying this model, we got an RMSE score of 322.432 and a MAE score of 227.302.

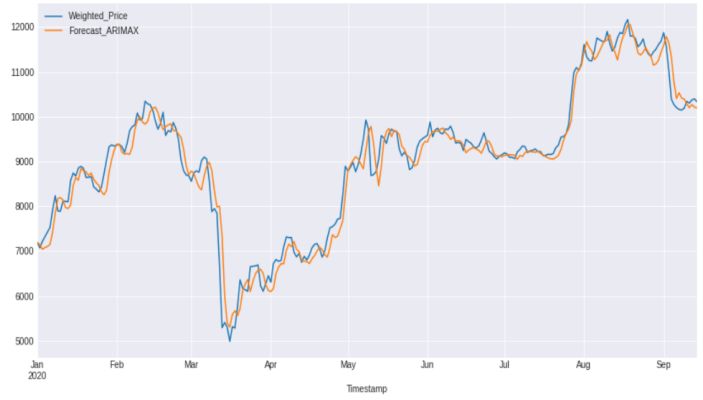


Fig 7: Arima prediction subplot

2. FB PROPHET model:

Our FB prophet model gave an RMSE score of 323.009 and an MAE score of 229.302.

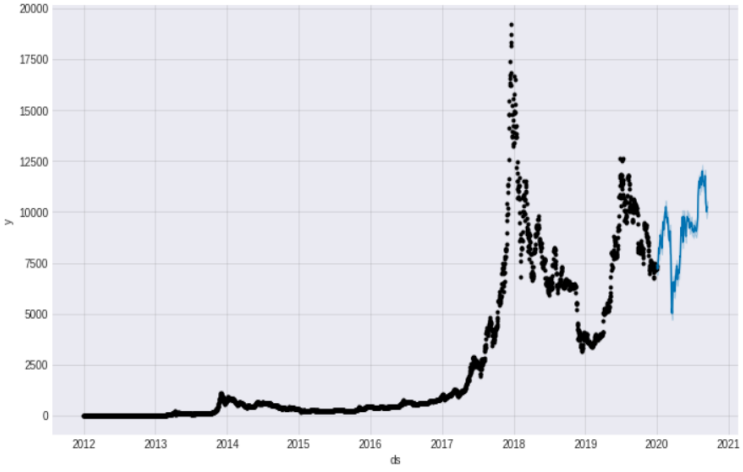
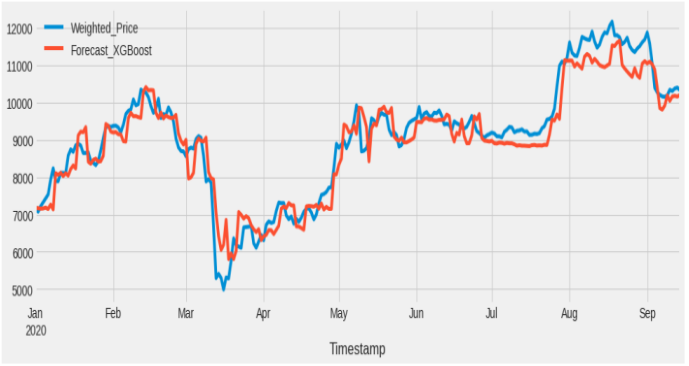


Fig 8: Fb prophet prediction subplot

3. XG BOOST model:

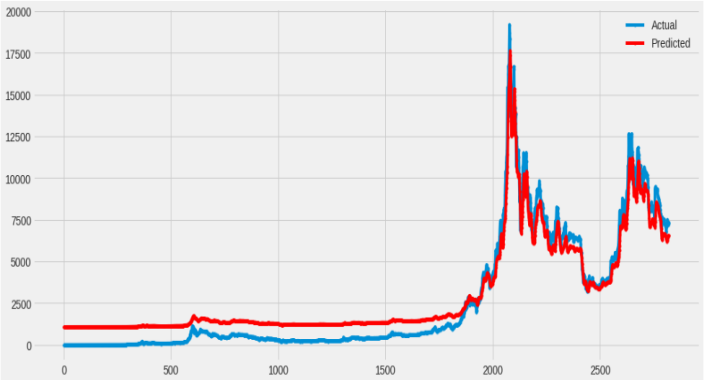
For this model, we opted for a hold-out based validation which means we selected data after 2019 as holdout and trained our model on data from 2012 to 2019. We obtained an RMSE score of 470.260 and a MAE score of 369.104.



4. LSTM model:

We obtained the best results from our LSTM model and thus decided to use it for our prediction of the bitcoin price for the next 30 days. The RMSE for our LSTM model obtained is

0.04792 and it's MAE result is 0.20979 for our train datasets. After using this model on our test dataset we got an incredible accuracy and the RMSE and MAE values for this test dataset are 0.05444 and 0.22614 respectively.



Comparing various models:

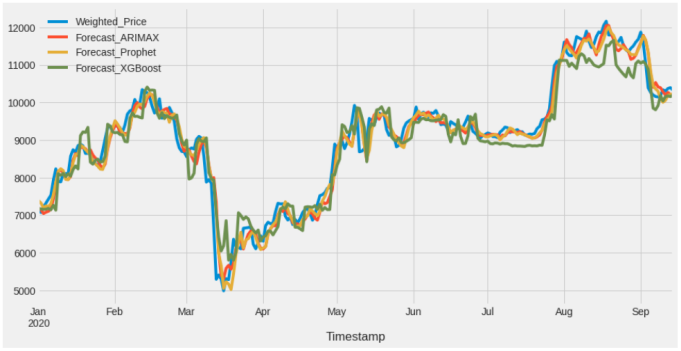
Below is show the RMSE and MAE score which are the validation metrics for the models performance

Model Name	RMSE Score	MAE Score
ARIMAX	322.423201	227.302623
FB PROPHET	323.009792	229.302623
XGBOOST	470.260663	369.104698

LSTM METRIC	TRAIN	TEST
RMSE	0.047928	0.054448
MAE	0.209794	0.226148

Fig 8: Validation metrics

The subplot below shows the comparison between the actual weighted price along with the predicted weighted price by the ARIMA, Fb prophet and XG boost model.



V. CONCLUSION & FUTURE WORK

We observed remarkable results using LSTMs. They really work a lot better for most sequence tasks. Also the other model's performance was not as good as LSTM which is evident from the RMSE score.

We can increase the number of epochs to refine our model performance, we can increase epochs to 100 and see the results. Also, the number of lag features can be increased beyond 100 to help in model learning.

VI. REFERENCES

[1] Bitcoin Price Prediction Based on Deep Learning Methods Xiangxi Jiang Published: February 11, 2020

[2] Time-series forecasting of Bitcoin prices using high-dimensional features: a machine learning approach Mohammed Mudassir, Shada Bennbaia, Devrim Unal & Mohammad Hammoudeh Published: 04 July 2020

[3] Bitcoin Price Prediction using Machine Learning Siddhi Velankar, Sakshi Valecha, Shreya Maji Published: 11 February 2018

[4] Predicting Fluctuations in Cryptocurrency Transactions Based on User Comments and Replies Young Bin Kim,Jun Gi Kim,Wook Kim,Jae Ho Im,Tae Hyeong Kim,Shin Jin Kang,Chang Hun Kim Published: August 17, 2016

[5] Predicting Bitcoin Returns Using High-Dimensional Technical Indicators Jing-Zhi Huang, William Huang, Jun Ni Received Date: 12 October 2018Accepted Date: 29 October 2018

[6] Next-Day Bitcoin Price Forecast Ziaul Haque Munim ,Mohammad Hassan Shakil and Ilan Alon Published: 20 June 2019

[7] Investopedia/ARIMA

[8] Towards data science

[9] Wikipedia