End-Term Report

Price Forecasting using Time Series Analysis on Cryptocurrency Data

Rajvardhan Verma (2330504) under Prof. Tushar Sandhan

July 12, 2023

Abstract

Virtual currencies have been declared as one of the financial assets that are widely recognized as exchange currencies. The cryptocurrency trades caught the attention of investors as cryptocurrencies can be considered as highly profitable investments. Cryptocurrency, being a novel technique for transaction systems, has led to a lot of confusion among investors and any rumours or news on social media has been claimed to significantly affect the prices of cryptocurrencies. The huge percentage increase/decrease in its price over a short period of time is an intriguing phenomenon that cannot be foreseen, hence cryptocurrency price prediction has been a hot topic of study.
I studied some basic regression models which use feature extraction, and implemented the popular ARIMA model. Further, I moved to deep learning models and chose the LSTM model since it is much less susceptible to vanishing gradient problem. From the results, it is clear that ARIMA model is quite efficient in making prediction in short span of time, but as the time grows, the precision rate would decrease. However, after training, the LSTM could make prediction more efficiently, and the precision rate is also higher. In general case, taking less previous data to make prediction in LSTM could lead to better result.

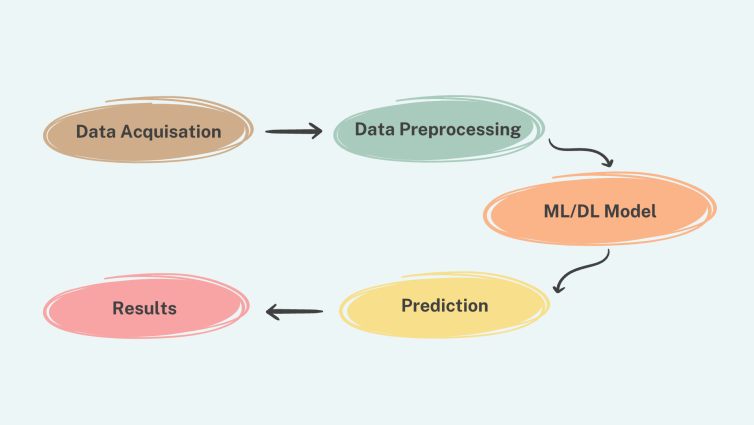

Figure 1: General Pipeline for Analysis of Prediction Models

1 Data Preprocessing

The data for the cryptocurrency Bitcoin was obtained through yahoo finance. The steps for preprocessing the data included Noise Minimization, checking for stationarity of data as well as for missing data.

- **Noise Minimization -** The average for the data of the past 20 days was taken and the current closing price was replaced with the said data.
- **Stationarity Test -** The stationarity of data was tested through Augmented Dicky Fuller Test.
  - Null Hypothesis: If failed to be rejected, it suggests that the time series data is not stationary.
  - Alternate Hypothesis: The null hypothesis is rejected, it suggests that the time series data is stationary.
- **Data Filling -** The missing data was filled using the KNN Imputer from scikit-learn.

2 Sentiment Analysis

The data for sentiment analysis was obtained through tweets data related to Bitcoin and other cryptocurrencies. The mwclient package was used to get this data. The data consisted of a dictionary consisting of 'revid', 'parentid', 'user', 'timestamp', and 'comment'. We imported the sentiment pipeline from the transformers package to perform the analysis. We created another dictionary where we store the average sentiment score for all the edits of each day. We also stored the number of tweets made during the day and the number of negative tweets. The sentiment pipeline uses the concept of transformers and self-attention mechanisms to classify the positive or negative sentiments of the text. Transformers are a type of deep-learning model architecture that has produced cutting-edge outcomes in applications such as machine translation, text categorization, and question-answering. The self-attention mechanism in the encoder allows it to capture the relationships between words in the input sequence. An encoder and a decoder comprise the transformer architecture. The encoder takes an input sequence and generates a set of contextualized representations for each word in the sequence. On the other hand, the decoder takes the encoded representations and generates the output sequence one word at a time.

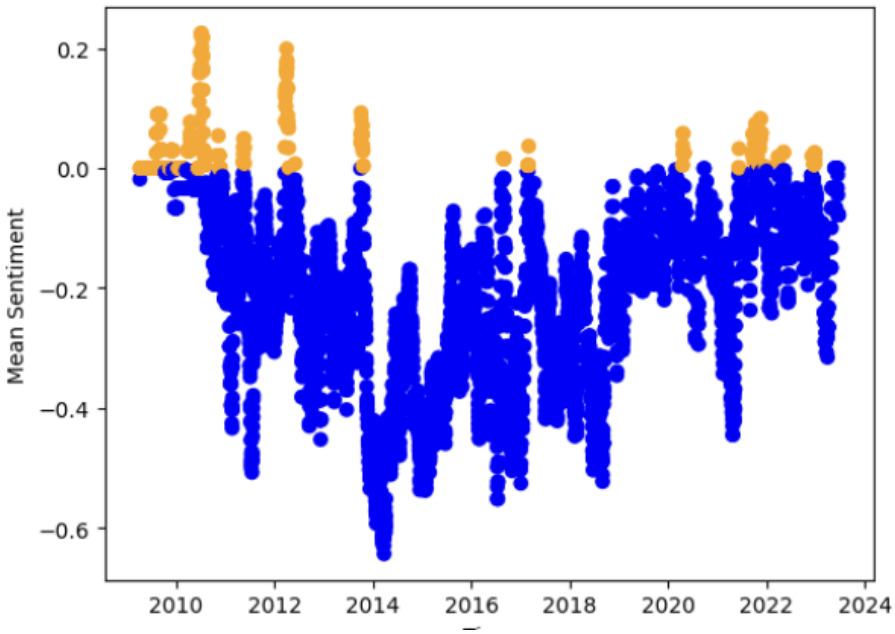

Figure 2: Plot of mean positive and negative sentiments

3 ARIMA Model

3.1 Data Collection & Preprocessing

The cryptocurrency datasets used in my study includes *Bitcoin* and *Ethereum* which contains historical prices for the past five years.
The datasets that I have downloaded had some missing points, which should be handled since many ML models can't handle missing data. To fill the missing points, I have taken first k instances closer to the missing value instance, and then get the mean of that attribute related to the k-nearest neighbors (KNN). This method is called **KNN Imputer**.

The ARIMA models are valid only for stationary dataset. These are the two methods that I have used for checking stationarity.

- **Rolling Statistics -** Plotted the moving average or moving standard deviation to see if it varies with time. It is a visual technique to check for stationarity.
- **ADF Test -** Augmented Dickey–Fuller test is used to gives us various values that can help in identifying stationarity. The test gives a p-value, if this value is less than 0.05, our data is stationary.

To make the datasets stationary, the seasonality and trends from the series were reduced by using the differencing techniques.

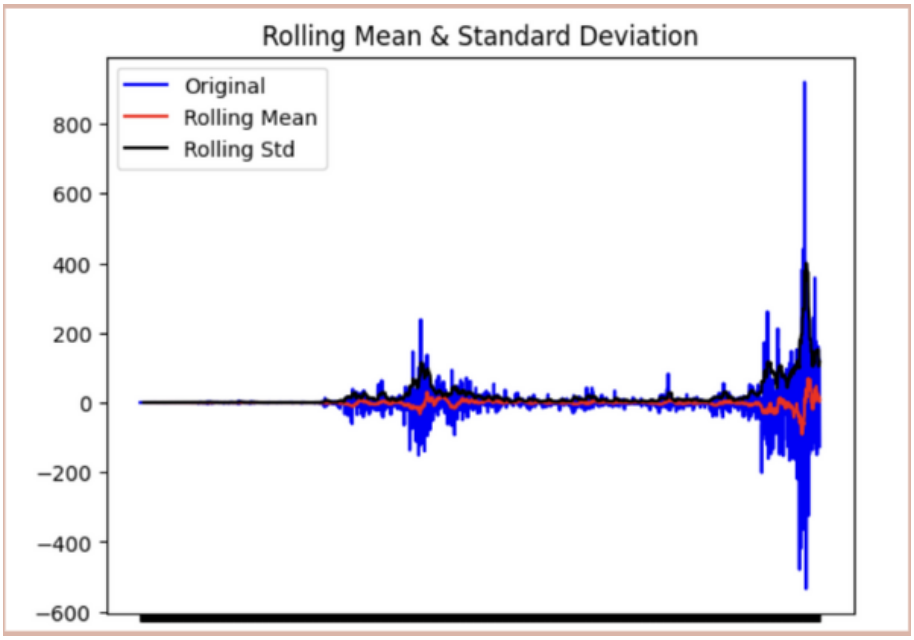

Figure 3: Plot of Rolling Statistics

3.2 Forecasting & Results

After preprocessing the data, model was trained and tested for both the crytocurrencies, Bitcoin and Ethereum. The RMSE was found to lower for Ethereum since there is less sudden variation in Ethereum as compared to Bitcoin.
The original and predicted data is plotted to observe how well the model predicts the test data.

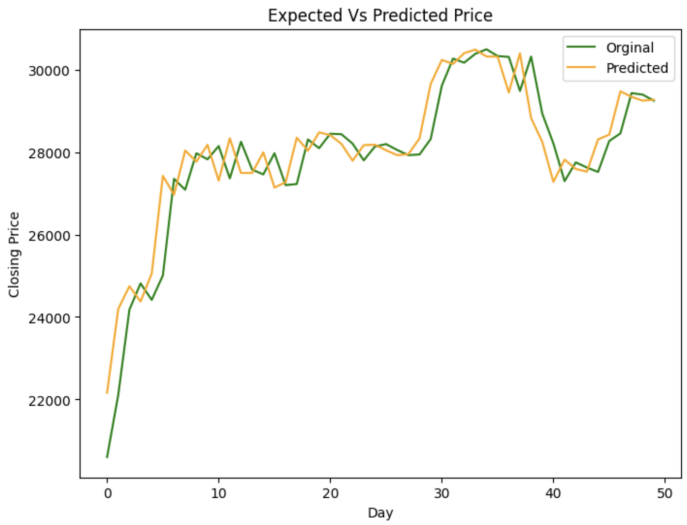

Figure 4: Plot of original and predicted values by ARIMA

4 LSTM Model

4.1 Data Collection & Preprocessing

The datasets used are similar to ones used in ARIMA model to draw a comparision between the two models. Also, the noise is handled by removing the *outliers* in dataset and the missing points are filled with suitable values using KNN Imputer method of gap filling.

To help the LSTM model to converge faster, it is important to scale the data. We have used **Min-Max scaler**, which is one of the most common scalers and refers to scaling the data between a predefined range (usually between 0 and 1). This method is beneficial for Neural Networks since they don't assume any data distribution.

$$\text{Scaled Value} = \frac{\text{Value} - \text{Min Value}}{\text{Max Value} - \text{Min Value}}$$

4.2 Forecasting & Results

After normalization, dataset was divided into train and test data. Then we train the LSTM model, which contains 2 LSTM layers and 2 dense layers.
The RMSE is calculated for the predicted values, and we found that the model is quite accurate for Ethereum and fairly accurate for Bitcoin as well.

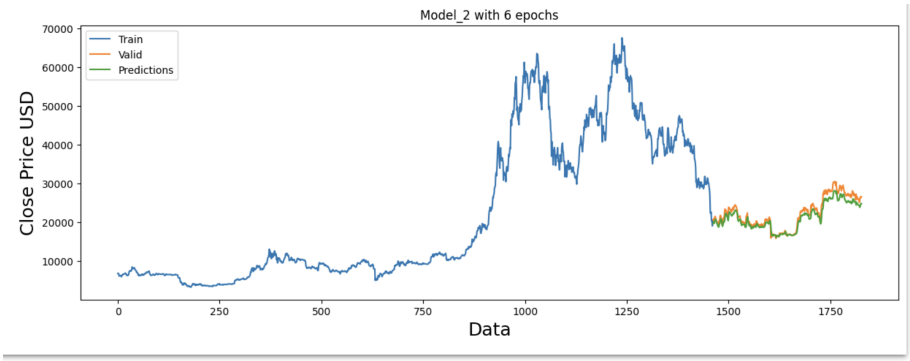

Figure 5: Plot of original and predicted values by LSTM

5 Reinforcement Learning Model

We have attempted to use Q-Learning as a training model for reinforcement learning. For this, we have first defined our Environment. A constructor, a reset() function to reset the environment, and a step function make up the environment. Buy and sell are the step function's actions. When we buy, we store those days' closing prices. When we sell, we calculate the total profit. We reward positively if profit is positive and negatively otherwise. The code implements a strategy for training a Deep Q-Network (DQN) agent using the Chainer library. The Q network architecture consists of three linear layers and was implemented using the chainer library. During the training of the DQN network, the model first begins by taking random actions to explore the environment we have defined. It gradually shifts towards exploiting its learned knowledge by selecting actions based on the maximum Q- value predicted by the DQN. This strategy is used to enable the DQN agent to learn an optimal policy for making decisions in the given environment.

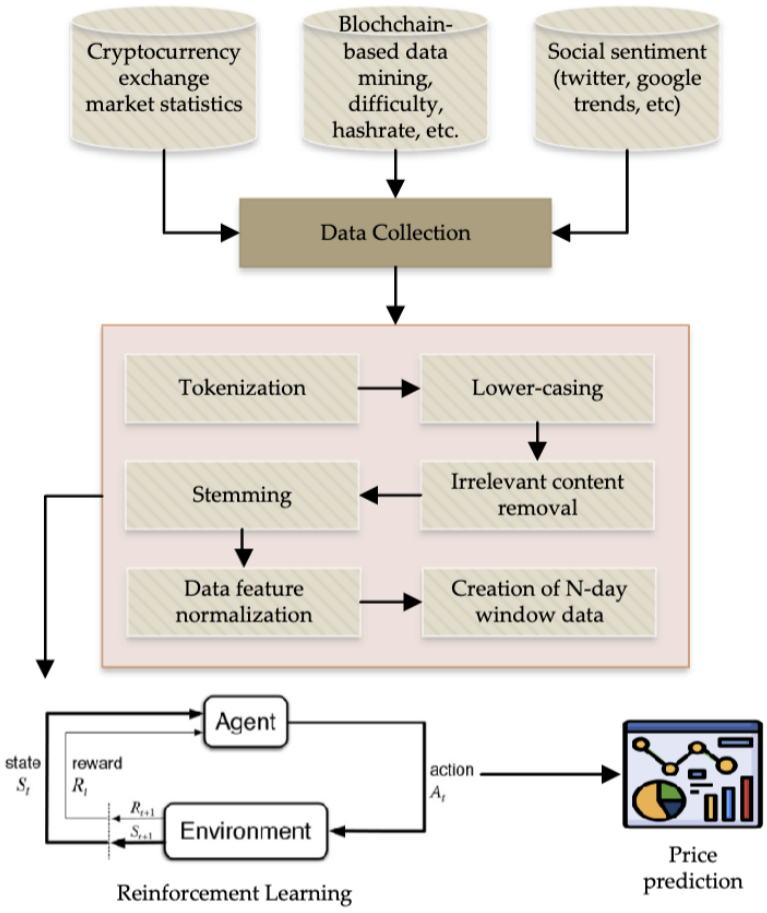

Figure 6: Reinforcement Learning based Prediction Model

Results compiled (RMSE): The formula for Root Mean Square Error(RMSE) is given by:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Model	Bitcoin	Ethereum
ARIMA	661.3740	193.3512
LSTM	127.1942	5.0199

Hence, the deep learning model LSTM outperforms the basic machine learning model ARIMA in terms of accuracy of prediction.

6 Future Works

- To improve the accuracy of LSTM model by adding other types of layers in our model.
- To improve the performance of the reinforcement learning model.

7 References

- M. Fernandes, S. Khanna, L. Monteiro, A. Thomas and G. Tripathi, "Bitcoin Price Prediction," IEEE journal.
- Chen, J. Analysis of Bitcoin Price Prediction Using Machine Learning. J. Risk Financial Manag. 2023, 16, 51.

- Special Thanks to my mentor **Mr. Pranjal Dubey** who regularly guided me and helped me reach this far in this research project.

- Please note that the links to my work are accessible through clickable coloured links.