CRYPTO-CURRENCY (BITCOIN) PRICE PREDICTION USING DEEP LEARNING

Submitted in partial fulfillment of the requirements

for the degree of

Bachelor of Engineering (SEM VIII)

Mini Project Synopsis Report - Stage-II

By

Shivprakash Vishwakarma
Roll No. 57

Afzal Siddique
Roll No. 48

Swaraj Kondlekar
Roll No. 32

Under the Supervision of

Prof. A.S. Kunte

DEPARTMENT OF INFORMATION TECHNOLOGY
KONKAN GYANPEETH COLLEGE OF ENGINEERING
KARJAT-410201

2020-2021

Certificate

This is to certify that the following students have submitted the synopsis for the project titled

Cryptocurrency (Bitcoin) Price Prediction Using Deep Learning

At Konkan Gyanpeeth College of Engineering, Mumbai as a partial fulfillment of the requirement for the award of the degree of Undergraduate in Department of Information Technology (Semester VIII) of University of Mumbai in the year 2020 – 2021.

Supervisor/Guide	**Head of Department**
Prof. A. S. Kunte	
Department of Information Technology	Department of Information Technology

Project Report Approval

This mini project report is entitled **Crypto Currency Price Prediction using Deep Learning** by **Shivprakash Vishwakarma(Roll No: 57), Afzla Siddique(Roll No:48), Swaraj Kondlekar(Roll No: 32)** is approved for the degree of **DEPARTMENT OF INFORMATION TECHNOLOGY.**

Examiners

1.

2.

Date:

Place:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data /fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Signature
Shivprakash Vishwakarma(Roll no:58)

Signature
Afzal Ali Niyaz Siddique(Roll no:49)

Signature
Swaraj Kondlekar(Roll no:33)

Date:

Abstract

The decentralization of cryptocurrencies has greatly reduced the level of central control over them, impacting international relations and trade. Further, wide fluctuations in crypto Currency price indicate an urgent need for an accurate way to forecast this price. This project proposes a method to predict crypto Currency price by considering various factors such as market cap, volume, circulating supply, and maximum supply based on deep learning techniques such as the recurrent neural network (RNN) and the long short-term memory (LSTM), which are effective learning models for training data, with the LSTM being better at recognizing longer-term associations. Building on this knowledge, we examine the price and predict the price of Bitcoin cryptocurrency available in the market. We examine the prediction of prices, or rather inability to do so, before introducing the Bitcoin Price Predictor web application developed as part of this project. Containing up to date prices, this web application is hosted on local machine and predicts prices of Bitcoin. Our web app tries to predict the price of bitcoin for near future. The research, planning methodologies, technologies, and design and evaluation of this application are described in detail in the penultimate chapter of this dissertation, followed by a concluding word on the process as a whole.

Acknowledgement

Success is nourished under the combination of perfect guidance, care blessing. Acknowledgement is the best way to convey. We express deep sense of gratitude brightness to the outstanding permutations associated with success. Last few years spend in this estimated institution has molded us into contend and aspiring Engineers.

We express our sense of gratitude towards our project guide Prof. A.S Kunte. It is because of his valuable guidance, analytical approach and encouragement that we could learn and work on the project. We will always cherish the great experience to work under their enthusiastic guidance.

We are also grateful to our principle Dr. M.J Lengare, who not only supporting us in our project but has also encouraging for every creative activity. We extend our special thanks to all teaching and non-teaching staff, friends and well-wishers who directly or indirectly contributing for the success of our maiden mission. Finally, how can we forget our parents whose loving support and faith in us remains our prime source of inspiration.

Lastly, we would like to thank all those who directly and indirectly helping to complete this project. We would also like to acknowledge with much appreciation the crucial role of the staff of Information Technology Department, who gave the permission to use the all required software/hardware and the necessary material to completing to the project.

Contents

LIST OF FIGURES

ABBREVIATIONS:

ARIMA: Autoregressive integrated moving average

LSTM: Long-Short Term Memory

RNN: Recurrent Neural Networks

KDE: Kernel Density Estimation

KPSS: Kwiatkowski-Phillips-Schmidt-Shin

ADF: Augmented Dickey Fuller

TPU: Tensor Processing Unit

GPU: Graphics Processing Unit

HTML : Hypertext Markup Language

CSS: Cascading Style Sheets

API: Application programming interface

Chapter 1

INTRODUCTION

1.1 Introduction

Crypto Currency, is a decentralized digital or virtual Currency. Use of cryptography for security makes it difficult to counterfeit. Cryptocurrencies started to gain attention in 2013 and since then witnessed a significant number of transactions and hence price fluctuations. The crypto Currency market is just similar to Crypto-Currency market. It has gained public attention and so effective prediction of price movement of crypto Currency will aid public to invest profitably in the system. This project tries to predict the price of Cryptocurrencies like Bitcoin. Deep learning techniques were implemented and the use of Long Short Term Memory (LSTM) network proved very efficient in predicting the prices of digital currencies.

1.2 Objectives

1. To implement LSTM model which will be train on available historical data.

2. To forecast the price of Crypto Currency(Bitcoin).

3. To integrate this model in order to develop simple system through which user can interact and perform predictions as per his requirements i.e no. of days for which he/she want to predict the future price of bitcoin.

1.3 Purpose, Scope & Applicability

1.3.1 Purpose

The Cryptocurrencies like Bitcoin are influenced by many uncertainties factor such as political issue, the economic issue at impacted to local or global levels. To interpretation key of success, factor to providing accurate predictions is complicated work. For the market, we can analyze with any techniques such as technical indicator, price movements, and market technical analysis. To solve the problem regarding the fluctuations there's a need automation tool for prediction to help investors decide for bitcoin or other crypto Currency market investment. Nowadays the automation tools are usually used in common Crypto Currency market predictions, and we can do the same works and strategy on this domain cryptocurrencies.

The purpose of this study is to predict the price of Bitcoin and changes therein using the LSTM model and to build a system which will help common people, investors or business analyst to invest in digital currencies as bitcoin by predicting their future values and to visualize and analyse the pattern of digital currencies.

1.3.2 Scope

Dataset Includes Historical bitcoin market data at 1-min intervals for select bitcoin exchanges where trading takes place. It consists time period of Jan 2012 to may 2021, with minute to minute updates of OHLC (Open, High, Low, Close), Volume in BTC and indicated Currency, and weighted bitcoin price.

Now that our data has been converted into the desired format, let's take a look at its various columns for further analysis.

- 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. To read more about how Weighted price is calculated.

Visualising the Time Series data

```
bitstamp.set_index("Timestamp").Weighted_Price.plot(figsize=(14,7), title="Bitcoin Weighte
d Price")
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f78a194b7d0>
```

FIG.1 Weighted Price Plot

Handling Missing Values in Time-series Data:

Imputations Techniques for non-Time Series Problems:

Imputation refers to replacing missing data with substituted values. There are a lot of ways in which the missing values can be imputed depending upon the nature of the problem and data. Depending upon the nature of the problem, imputation techniques can be broadly they can be classified as follows:

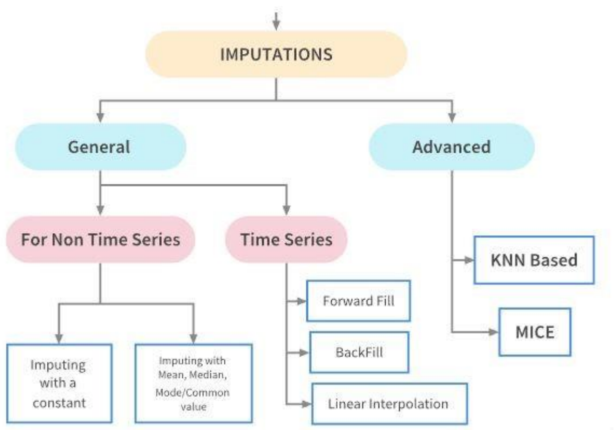


FIG.2 Imputation Techniques Classification

```
a['interp'] = a['Weighted_Price'].interpolate()   # Imputation using interpolation
```

Exploratory Data Analysis

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.

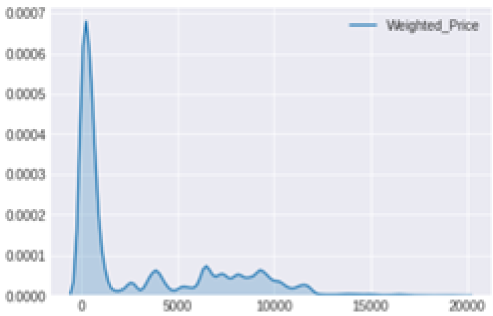


FIG.3 KDEs Plot

So there is a downward trend in Crypto Currency prices from Dec-17 onwards till March 2019. We will investigate it further by investigation and with some findings during that period.

Visualizing using Lag Plots

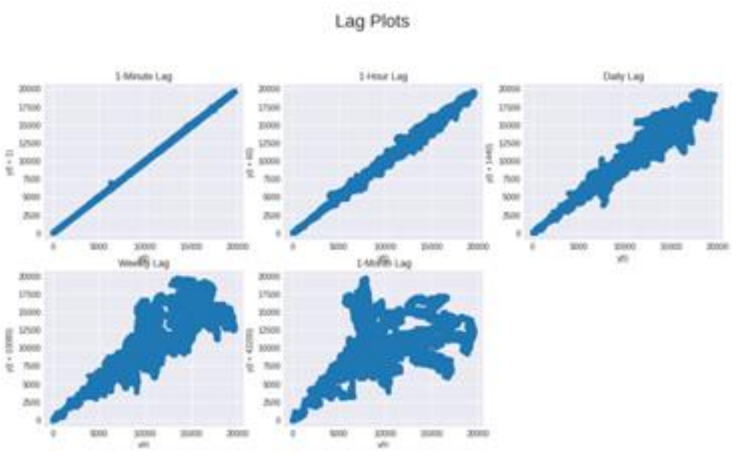


FIG.4 LAG PLOTS

We can see that there is a positive correlation for minute, hour and daily lag plots. We observe absolutely no correlation for month lag plots.

It makes sense to re-sample our data at most at the Daily level, thereby preserving the autocorrelation as well.

Time resampling

- data.resample() is used to resample the Crypto Currency data.
- The '1H' stands for hourly frequency, and denotes the offset values by which we want to resample the data.
- mean() indicates that we want the average Crypto Currency price during this period.

KPSS and ADF Test Conclusion

KPSS says series is not stationary and ADF says series is stationary. It means series is difference stationary, we will use differencing to make series stationary.

Feature Extraction

Rolling windows

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.

Benefits : The key benefits of calculating a moving average or using this rolling method is that 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 split the time series into a training period and a validation period. This is called fixed partitioning.

we will opt for a ***hold-out based validation***. Hold-out is used very frequently with time-series data. In this case, we will select all the data for 2020 as a hold-out and train our model on all the data from 2012 to 2019.

We are further building our LSTM model as follows:

```
# Initialising the LSTM
regressor = Sequential()

# Adding the first LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True, input_shape = (X_train.shape[1], 1)))
regressor.add(Dropout(0.2))

# Adding a second LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True))
regressor.add(Dropout(0.2))

# Adding a third LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50, return_sequences = True))
regressor.add(Dropout(0.2))

# Adding a fourth LSTM layer and some Dropout regularisation
regressor.add(LSTM(units = 50))
regressor.add(Dropout(0.2))

# Adding the output layer
regressor.add(Dense(units = 1))

# Compiling the RNN
regressor.compile(optimizer = 'adam', loss = 'mean_squared_error')
```

User Interaction Building

To make User Interaction easy we have a made a simple prediction system in which the User must have logged in to access the feature of prediction. The User have to choose the prediction period i.e how many days the price to be predicted. By submitting the prediction period, the user gets the desired result/Prediction.

FIG.5 Website User Interface

1.3.2.1 Major Challenges

The objective of the project deals with a lot of ideas that need to be researched to be able to execute properly. The project deals with a lot of research areas in Machine Learning and Deep Learning which is in a preliminary stage. This becomes the core obstacles of the project which we need to overcome to be able to execute the project.

The following are the major challenges which need to be tackled for the project lifecycle.

➢ **Database Modelling**:

Finding the datasets for popular currencies like Bitcoin was easy to find. However no clear well- defined dataset was available for all the ten cryptocurrencies. Hence dealing with the dataset was an initial challenge.

➢ **Lack of proper communication:**

The ongoing pandemic made communication between team members difficult as initially not various technologies were available to communicate and work on the project. Proper communication is an important factor to drive the project to completion.

- **Accuracy:**

Some of the traditional ML and Deep Learning algorithms are not very accurate when t comes to predicting cryptocurrency prices. Hence LSTM model was adopted. However even with LSTM the maximum accuracy that could be achieved is approximately 70 percent.

1.3.3 Applicability

Different organization may consider it as investment portal anyone who wants to buy Crypto Currency i.e Bitcoin can take look at this system. This will help them to properly invest money in with digital Currency lesser chances of losing it. It will majorly be beneficial for the digital Currency brokerage institution as they will get an idea whether the price will rise or fall and as a result will provide a better understanding of the price pattern.

1.4 Achievements

We have successfully build a deep learning Bitcoin prediction model which predicts the bitcoin prices for upto 30 upcoming days using LSTM. We have also build a website for implementing the model.

1.5 Organisation of the Report

Report Overview

Chapter 2 explores the different existing system. their technologies, motivation, value proposition, their advantages, and drawbacks. It also highlights upon how our system will try to improve upon any shortcomings. The literature survey covers the tools used by the systems and the reasons for using those tools. The aim behind the literature survey is to give us a reference for creating our product and to make the most feasible and appropriate decision in every stage of the product development.

Chapter 3 explains our proposed methodology for the project. The chapter moves onto the proposed approach. Finally, the chapter ends with the benefits of our proposed solution.

Chapter 4 focuses on the management of our project. The way our project was scheduled, the task network diagram of our project, project timeline chart. Next is the feasibility study of our project. It focuses on different kinds of feasibilities like technical feasibility, operational feasibility, economic feasibility. It focuses on the project resources used, mainly the hardware, software and operating requirements. It also focuses on project estimation, function point analysis and risk management mitigation planning.

Chapter 5 focuses on design system of our project. Design diagrams, ER Model of our system, DFD (Data Flow Diagram) of our system, the UML diagrams of our system, our system architecture, the algorithms and the technologies used for our project.

Chapter 6 focuses about working of our project. Here we have explained each module of our system in detail, the algorithms and tools used, and our user interface design.

Chapter 7 is all about the demonstration and results of our project.

Chapter 8 is the conclusion, where we have concluded what our project does.

Chapter 2

LITERATURE SURVEY

2.1 Literature Survey

1. A review of missing values handling methods on Time-Series data:

(Authors: Irfan Pratama, Rini Indrayani, Adhistya Erna Permanasari, October 2016)

In this paper, the description of several previous studies about missing values handling methods or approach on time series data is explained. This paper also discuss some plausible option of methods to estimate missing values to be used by other researchers in this field of study. The discussion's aim is to help figure out what method is commonly used now along with its advantages and drawbacks.

Missing values becomes one of the problems that frequently occur in the data observation or data recording process. The needs of data completeness of the

observation data for the uses of advanced analysis becomes important to be solved. Conventional method such as mean and mode imputation, deletion, and other methods are not good enough to handle missing values as those

method can caused bias to the data. Estimation or imputation to the missing data with the values produced by some procedures or algorithms can be the best possible solution to minimized the bias effect of the conventional

method of the data. So that at last, the data will be completed and ready to use for another step of analysis.

This paper figures missing values handling method from the conventional one to the modern one. Methods such as deletion, mean imputation, and hot decking are considered as conventional method which can be used for more general dataset, while estimation techniques is modern method that specifically picked to handle missing values in time series data.

2. Resampling strategies for imbalanced time series:

(Authors: Nuno Moniz, Luis Torgo, Paula Branco, October 2016)

The objective of this paper is to provide solutions capable of significantly improving the predictive accuracy of rare cases in forecasting tasks using imbalanced time series data. The Paper Extent application of resampling strategies to the time series context and introduce the concept of temporal

and relevance bias in the case selection process of such strategies, presenting new proposals. This paper evaluate the results of standard regression tools and the use of resampling strategies, with and without bias over 24 time series data sets from 6 different sources. Results show a significant increase in predictive accuracy of rare cases associated with the use of resampling strategies, and the use of biased strategies further increases accuracy over the nonbiased strategies.

Time series forecasting is a challenging task, where the non-stationary characteristics of the data portrays a hard setting for predictive tasks. A common issue is the imbalanced distribution of the target variable, where some intervals are very important to the user but severely underrepresented. Standard regression tools focus on the average behaviour of the data. However, the objective is the opposite in many forecasting tasks involving time series: predicting rare values. A common solution to forecasting tasks with imbalanced data is the use of resampling strategies, which operate on the learning data by changing its distribution in favour of a given bias.

3. Crypto Currency Price Prediction Using Long-Short Term Memory Model:

(Authors: Prashanth J R, Vineetha S Das, July 2018)

This paper tries to predict the price of Cryptocurrencies. Machine learning techniques were implemented and the use of Adam optimizer and Long Short Term Memory (LSTM) network proved very efficient in predicting the prices of digital currencies.

There are lot of cryptocurrencies in the market and in this paper the following cryptocurrencies are selected for study and price prediction, Bitcoin, Ethereum, Ripple, Monero, Litecoin and Dash. The historical data required for price prediction of cryptocurrencies are collected from https://coinmarketcap.com. The methodology of the work consists of several steps data collection, data processing, feature extraction, training Long Short – Term Memory network and predictions using the trained network. The pre-processing involves data reduction, data normalization and data cleaning to get the required dataset. Then it is divided into test dataset and train dataset. Feature extraction selects the features that are to be fed to the LSTM network. In the current case it includes opening, high, low and closing price. Training of LSTM network involves feeding the neural network with data and training the same. The prediction involves assigning random biases and weights. The proposed LSTM model is composed of a sequential input layer an LSTM layer and a dense output layer with linear activation function. The prediction from the model is taken and the mean absolute error is used to ascertain its effectiveness.

4. Prediction of Bitcoin using Recurrent Neural Network:

(Authors: Pratik Mehta, E. Sasikala, march 2020)

The main objective of the work is to predict the Bitcoin prices, one of the most popular and widely used crypto Currency which is a source of attraction for many investors as a source of profit or investment. But the market for the cryptocurrencies been volatile since the day it was first introduced. So, the approach towards the survey is to use LSTM RNN and use the available dataset and train the model to give the highest possible accuracy and to provide a real-time price of the Bitcoin for the following days.

Using Neural Networking Systems the connection between the performance of Bitcoin and the next day's price change of Bitcoin using an Artificial Neural Network Ensemble solution called the Selective Neural Network Ensemble Dependent Genetic Algorithm was discussed, then the neural network using Multi-Layered Perceptron was constructed. The organization was used to predict the price of Bitcoin's next day course through a sequence of nearly 200 blockchain apps over a 2-year cycle to better understand Bitcoin's practicality and utility of real-world applications. With a series of almost 200 blockchain apps over a period of 2 years, the firm was used to predict the next day path of Bitcoin's price to better understand the practicality and utility of it in real world applications. The program was used to predict the next day's market path for Bitcoin with a selection of about 200 blockchain apps over a 2-year span to better understand the practicality and efficacy of real-world applications. An ensemble-based trading methodology was compared over a span of 50 days with a previous day trend following a back-test trading strategy. The former trading strategy produced about 85 percent returns, outperforming the previous day trend following a special trading strategy that yielded about 38 percent returns and a trading strategy that follows the one-time, best MLP (Multilayer Perceptron) model in the ensemble that yielded around about 53 percent. Provides multi-layered perceptron for estimating bitcoin level. Jang and Lee's work which predicts the bitcoin price using Bayesian Neural Network and blockchain information. There are multiple online platforms nowadays, such as CoinTracking, BitcoinCharts, Bitcoinity, and BitcoinWisdom, that enable traders to use many technological analytics tools to identify trends and market feelings that are useful for entering a trade. Several Regression models which was based on Linear Regressions, Random Forests , Gradient Boosting Technology, and Basic Neural Networks. Kim et al. and Li et al found the Bitcoin price volatility to be expected by social evidence.

5. Bitcoin Price Prediction using ARIMA Model:

(Authors: Dr. Jinan Fiaidhi, Ahmer Sabah, Mahpara Anwer Ansari, Zeba Ayaz, preprint 2020)

In this research work, we have investigated bitcoin closing price prediction by using an ARIMA model. Towards this end, at first, we have preprocessed time series data to make it stationary, and then, have searched over feasible (p, q, d) parameters for finding the ARIMA model which minimizes the MSE (Mean Squared Error) of prediction. The results we get indicates that the bitcoin price

prediction using the value "closing price" history could result in large MSE values since bitcoin's price is vulnerable to high jumps and fall-downs. On the other hand, the results also confirm that the ARIMA model could be still used for price prediction in sub-periods of the timespan, which is by dividing the

timespan to several timespans over which, dataset has a unique trend. Furthermore, we have investigated the effect of the different parameter p, q and d value on the achieved MSE in price prediction. We have elaborated

this work by creating web service and high chart for better resolution of graph obtained.

To predict the BTC prices, we have modeled time series using ARIMA algorithm. Models with lower MSE are considered to be the ideal ones. First, we fit an ARIMA (2,1,0) model. This sets the lag value to 2 for autoregression, uses a difference order of 1 to make the time series stationary, and uses a

moving average model of 0. Second, we try to fit an ARIMA (0,1,18) model. This sets the lag value to 0 for autoregression, uses a difference order of 1 to make the time series stationary, and uses a moving average model of 18. We tried many combinations of ARIMA for obtaining the lesser MSE value so that we can find the best fit for our model. After fitting the model by passing its

parameters we have loaded the testing dataset from the drive again in the same way how we read training dataset. In training we have stored the BTC data collected for January 1st until January 7th, 2020. Finally, we are going to predict the bitcoin closing price and display the Mean Squared error which is the evaluation metric for our predicted Model. We assign the timestamp of

the dates to the data frame dates and predict the bitcoin price for seven days using forecast function.

Chapter 3

SURVEY OF METHODOLOGY

3.1 Deep Learning:

Deep learning (also known as deep structured learning) is part of a broader family of machine learning methods based on artificial neural networks with representation learning. Learning can be supervised, semi-supervised or unsupervised.

Deep-learning architectures such as deep neural networks, deep belief networks, recurrent neural networks and convolutional neural networks have been applied to fields including computer vision, machine vision, speech recognition, natural language processing, audio recognition, social network filtering, machine translation, bioinformatics, drug design, medical image analysis, material inspection and board game programs, where they have produced results comparable to and in some cases surpassing human expert performance.

3.2 Tensorflow:

TensorFlow is an end-to-end, open-source machine learning platform. You can think of it as an infrastructure layer for differentiable programming. It combines four key abilities:

- Efficiently executing low-level tensor operations on CPU, GPU, or TPU.
- Computing the gradient of arbitrary differentiable expressions.
- Scaling computation to many devices (e.g. the Summit supercomputer at Oak Ridge National Lab, which spans 27,000 GPUs).
- Exporting programs ("graphs") to external runtimes such as servers, browsers, mobile and embedded devices.

3.3 Keras:

Keras is the high-level API of TensorFlow 2.0. It is an approchable, highly-productive interface for solving machine learning problems, with a focus on modern deep learning. It provides essential abstractions and building blocks for developing and shipping machine learning solutions with high iteration velocity.Keras empowers engineers and researchers to take full advantage of the scalability and cross-platform capabilities of TensorFlow. You can run Keras on TPU or on large clusters of GPUs, and you can export your Keras models to run in the browser or on a mobile device.

3.4 Time Series:

A time series is a series of data points indexed (or listed or graphed) in time order Time series analysis comprises methods for analyzing time series data in order to extract meaningful statistics and other characteristics of the data. Time series forecasting is the use of a model to predict future values based on previously observed values. While regression analysis is often employed in such a way as to test theories that the current values of one or more independent time series affect the current value of another time series, this type of analysis of time series is not called "time series analysis", which focuses on comparing values of a single time series or multiple dependent time series at different points in time. Interrupted time series analysis is the analysis of interventions on a single time series.

3.5 Long-Short Term Memory:

Long short-term memory (LSTM) is an artificial recurrent neural network (RNN)

architecture used in the field of deep learning. LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series. LSTMs have an edge over conventional feed-forward neural networks and RNN in many ways. This is because of their property of selectively remembering patterns for long durations of time. The purpose of this article is to explain LSTM and enable you to use it in real life problems .

3.6 Recurrent Neural Network:

A recurrent neural network (RNN) is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behavior. Derived from feedforward neural networks, RNNs can use their internal state (memory) to process variable length sequences of inputs. This makes them applicable to tasks such as unsegmented, connected handwriting recognition or speech recognition.

The term “recurrent neural network” is used indiscriminately to refer to two broad classes of networks with a similar general structure, where one is finite impulse and the other is infinite impulse. Both classes of networks exhibit temporal dynamic behavior. A finite impulse recurrent network is a directed acyclic graph that can be unrolled and replaced with a strictly feedforward neural network, while an infinite impulse recurrent network is a directed cyclic graph that can not be unrolled.

3.7 Pandas:

pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language. Its library highlights are as follows:

- A fast and efficient DataFrame object for data manipulation with integrated indexing;
- Tools for reading and writing data between in-memory data structures and different formats: CSV and text files, Microsoft Excel, SQL databases, and the fast HDF5 format;
- Intelligent data alignment and integrated handling of missing data: gain automatic label-based alignment in computations and easily manipulate messy data into an orderly form;
- Flexible reshaping and pivoting of data sets;
- Intelligent label-based slicing, fancy indexing, and subsetting of large data sets;
- Columns can be inserted and deleted from data structures for size mutability.
- High performance merging and joining of data sets;
- Highly optimized for performance, with critical code paths written in Python or C.
- Python with *pandas* is in use in a wide variety of academic and commercial domains, including Finance, Neuroscience, Economics, Statistics, Advertising, Web Analytics, and more.

Chapter 4

REQUIREMENTS AND ANALYSIS

4.1 Problem Definition

The Cryptocurrencies like Bitcoin are influenced by many uncertainties factor such as political issue, the economic issue at impacted to local or global levels One reason why **bitcoin** may **fluctuate** against fiat currencies is the perceived store of **value** versus the fiat Currency. Cryptocurrencies like **Bitcoin** has properties that make it similar to gold. It is governed by a design decision by the developers of the core technology to limit its production to a fixed quantity of 21 million **BTC**. To solve the problem regarding the fluctuations there's a need automation tool for prediction to help investors decide for bitcoin or other crypto Currency market investment.

4.2 Requirements Specification

For implementation in software we will require the following software and hardware specifications:

4.2.1 Software Specification

For implementation we will require applications such as Jupiter Notebook ,Python3.6 or better, Spyder IDE,MS-Excel etc. That can be used to build an environment based on neural network and to train machine based on it.

4.2.2 Hardware Specification

To implement the project we will require a computer with specification such as multicore CPU, graphics card, hard disk upto 500GB, upto 8GB RAM.
Input devices such as keyboard, optical mouse.
Output device: LCD monitor.

4.2.3 Operation Requirements

The underlying operating requirement for the computing environment is listed.

- Active Internet Connection
- Windows 10

4.3 Planning and Scheduling

4.3.1 Project Schedule

Project schedule of “Cryptocurrency Analyzer and Predictor” is discussed in terms of a task network diagram and a timeline chart with hierarchy of tasks and sub-tasks scheduled and divided in modular activities.

4.3.2 Task Network Diagram

Here, each task or activity related to the project is named as in terms of T_N and A-Z. Each task may comprise of one or more activity node, as well as each task may require previous task as prerequisite activities.

4.3.3 Timeline Chart

In this chart, a breakdown of up to 13 weeks in terms of deliverables is given.

Cryptocurrency Price Prediction		**2021**														
		Jan			***Feb***				***March***				***April***			
Deliverables	**Time**	**W1**	**W2**	**W3**	**W4**	**W5**	**W6**	**W7**	**W8**	**W9**	**W10**	**W11**	**W12**	**W13**	**W14**	**W15**
UML Diagram	**1 w**															
Refining System Architecture and UML Diagrams	1 w															
Specifications Survey	**1 w**															
Hardware and Software Specification Survey	1 w															

Training the LSTM model	**2 w**
Training set of data	1 w
	1 w
Test the trained model	**2 w**
Test using test dataset	1 w
	1 w
Feedback	2 w
Defining primitives	1 w
Defining Constructs	2 w
Designing UI	2 w
Evaluator	1 w
Grader	1 w
Implementation	1 w
Evaluator	1 w
Testing	1 w
Testing of the system	1 w
Documentation	1 w

4.4 Preliminary Product Description

For making it a hard-core product we will be making a website, which will be using API's of with tech stack of frameworks as,

Frontend – Reactjs, context API/redux, html5, css3, bootstrap5, materialcss.

Backend – django/flask

Database – Mongodb

Tools – GIT, visualstudio code, etc

Our website have functionalities like crypto Currency price prediction for specific period of time & visualizing/analysing pattern of bitcoin data.

Chapter 5

SYSTEM DESIGN

In this chapter, we have defined the specific problem(s) our system strives to solve and till what extent it'll be solved. Proposed the approach we will be using for implementation of the solution using various UML diagrams and diagrammatic depictions of the system.

5.1 Design Diagrams

5.1.1 UML Diagrams (Component and Deployment diagrams)

- **Use Case Diagram:**

  The system has 2 actors: The user, and the system administrator. Here the user can simply visit the website and check for daily prices of Bitcoin. Users can be of two types: new user and a registered user. Only Registered Users can view and Predict Bitcoin prices .

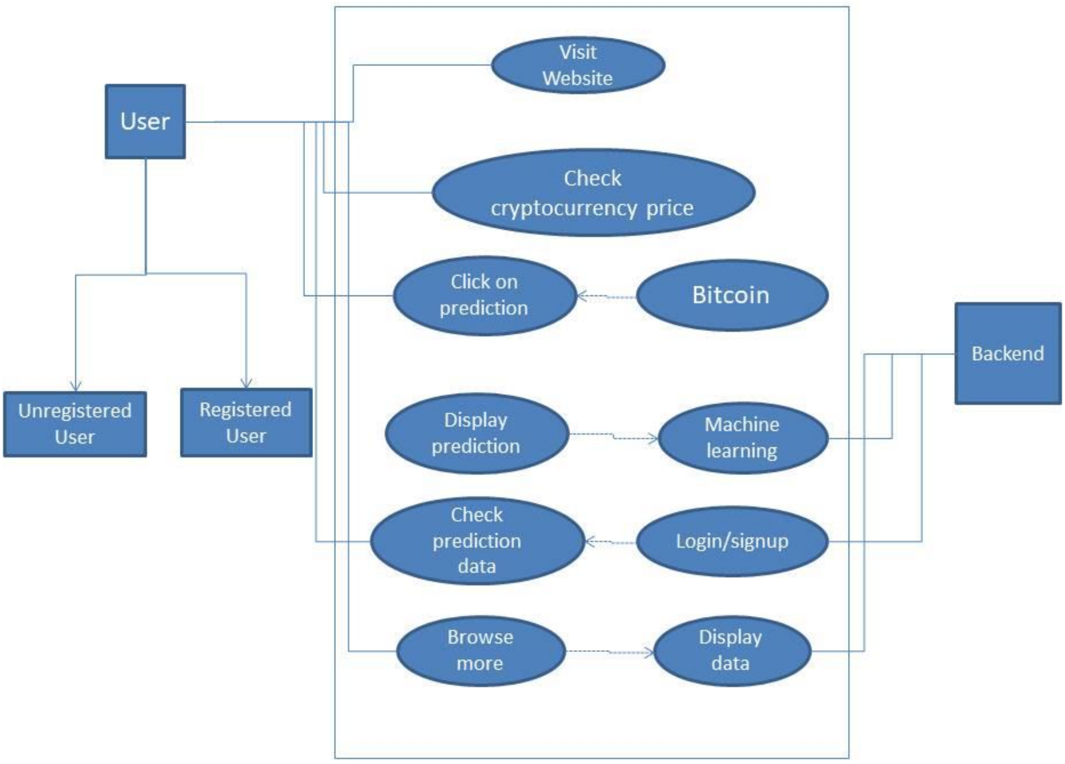


Figure 6: Use Case Diagram

➢ **Activity Diagram:**

The input is in the form of choices i.e how many days prediction does user wants. The input just provides a value to model that how many days of prediction is to be done. First day prediction is done and it is given as an input for predicting second day and so on. The diagram conveys the different use cases available for respective users to interact with or perform on the system. It is depicted in below activity diagram.

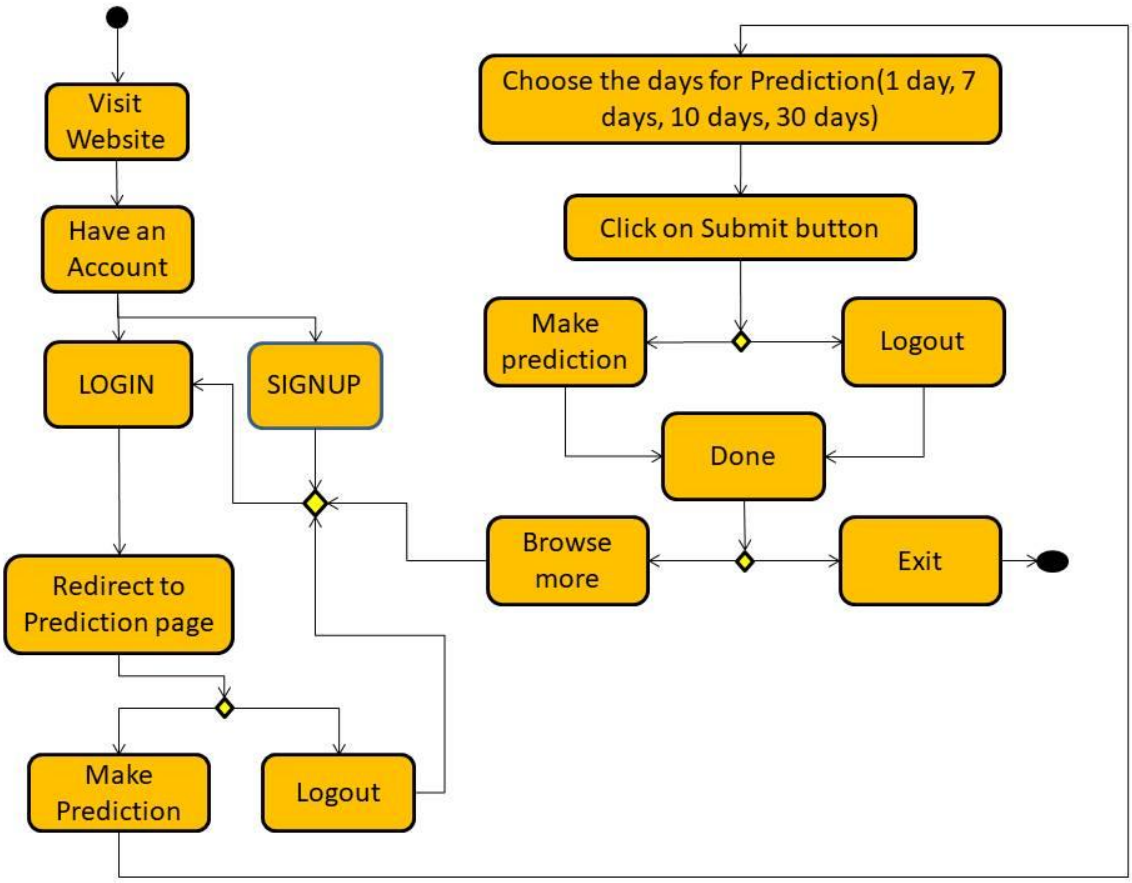


Figure 7: Activity Diagram

5.2 System Architecture

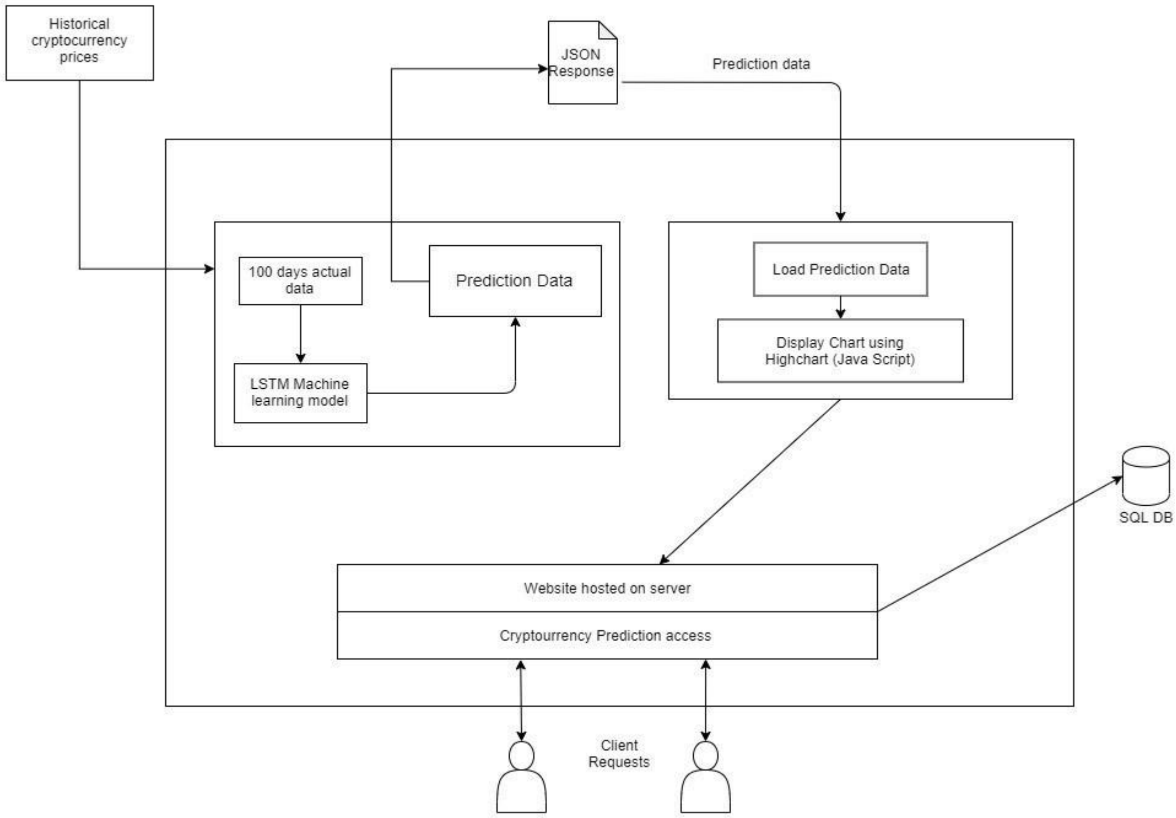


Figure 8: System Architecture

5.3 Tools

- The model will first be trained with historical cryptocurrency data.
- LSTM deep learning algorithm will then be deployed to make prediction for future prices of cryptocurrency using previous ten days data.
- The prediction result will then be passed on to the backend in the form of a JSON response as shown in the system architecture.
- Now to display the prediction result on the screen we make use of appropriate JavaScript library like Vue.js or D3.js. The output is in the form of a simple easy to understand graph.

5.4 User Interface Design

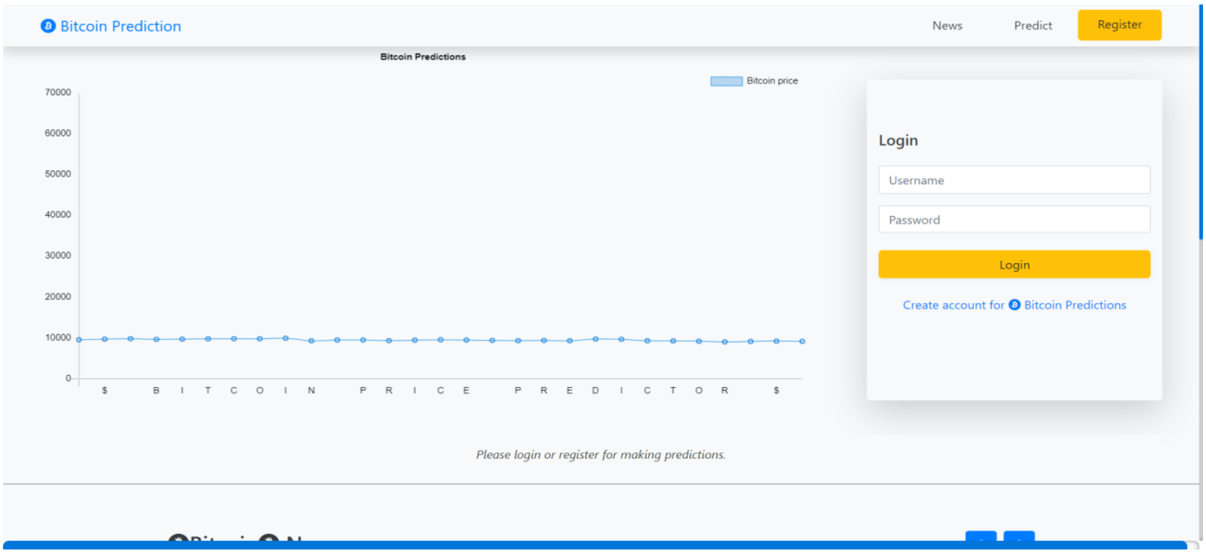


Figure 9: Landing Page

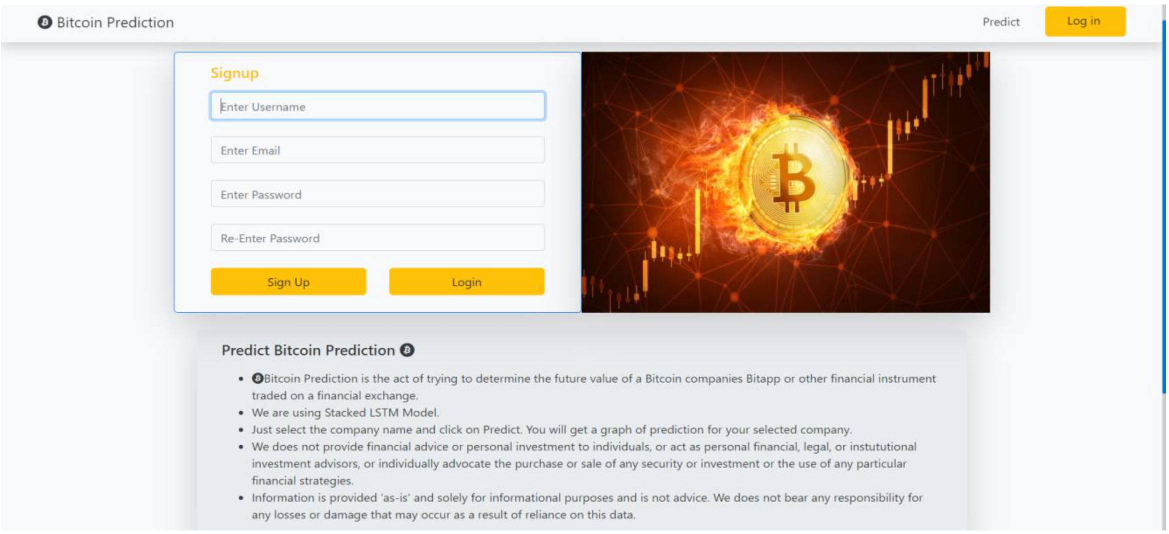


Figure 10: Sign-Up Page

Figure 11: Predict Section

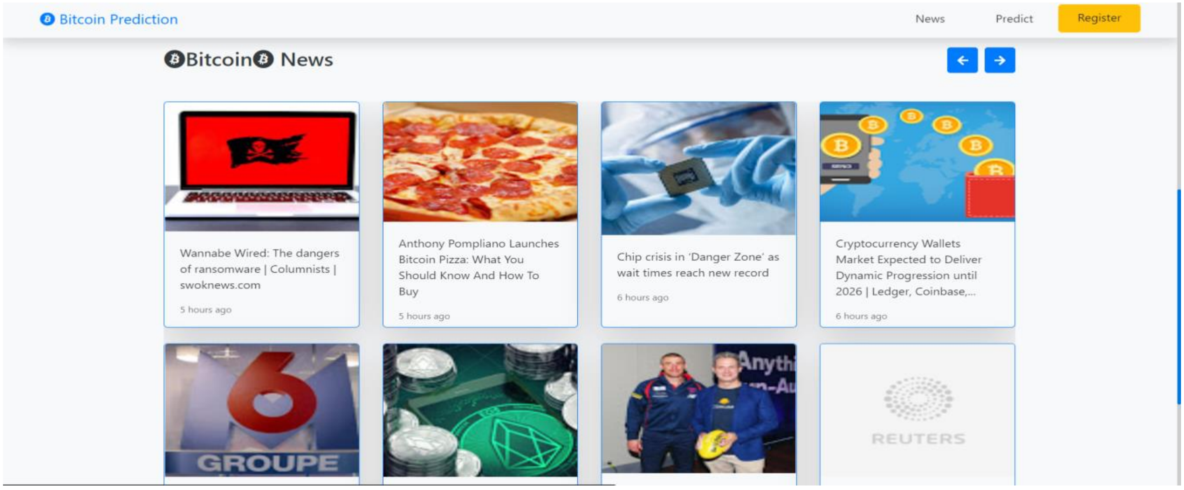


Figure 12: News Section

Chapter 6

IMPLEMENTATION AND TESTING

In this chapter, the implementation details including the working of the implemented system is shown. This chapter also includes the in-depth view of algorithms and interfaces designed.

6.1 Working of System:

- **Storing Historical Data:**

In this step the historical cryptocurrency data is stored in database like MongoDB or sol using a Python script.

This historical data consists of past 2000 days dataset which is used to train the model

The dataset consists of a total of six features: close price, open price, high price, low price, volume and an important parameter called timestamp.

- **Training the LSTM model:**

In this step the LSTM model is trained first using a training dataset. The model learns from this dataset

The model learns from a 100 days dataset. LSTM algorithm is deployed to predict the future prices of cryptocurrency and it is a very popular deep learning algorithm used specifically for temporal datasets.

LSTM works well for time series data because our dataset involves 6 parameters one of which is timestamp. Hence this algorithm.

- **Price Prediction & Display Predicted Result:**

In this step the trained LSTM model is fed with the test dataset to predict the closing price for the day of various cryptocurrencies.

The output of the model is the prediction data. This prediction data is passed to the etc. backend in the form a JSON response as shown in the system architecture.

The JSON response is then converted into an easy to interpret graph using an appropriate JavaScript library.

6.2 TESTING

6.2.1 Purpose of the test plan

The test plan includes and tracks the necessary information required to effectively define the approach to be used in the testing of the project's product. The Test Plan document is created during the Planning Phase of the project. Its intended audience is the project manager, project team, and testing team. Some portions of this document may be shared on occasion be shared with the client/user and other stakeholders whose input/approval into the testing process is needed.

6.2.2 Items to be tested/not tested

Item to Test	Test description	Test date
TI settings testing	**Tune the different technical indicators and check whether they are modified to be best suited for predictions of the underlying currency pair**	**30/1/2021**
TI weights testing	**Test whether assigned weights produced balanced results and observed values are not skewed**	**10/2/2021**

LSTM efficiency	**Observe the forecasts of LSTM to determine whether requirements of forecasts are being met**	**20/2/2021**
Functionin g of expert system	**Test whether the individual results of modules are being optimally incorporated into the expert system decision**	**10/3/2021**
Testing of GUI	**Make sure that the navigation and other required actions run smoothly on the app**	**30/4/2021**

6.2.3 Test Approach(s)

Module	Test Approach	Remark
Technical Analysis	**Follow the popular testing paths defined by the analyst community and ensure the approach is suitable for the forex market**	**Testing straightforward**

LSTM	**Use recommendations from literature review and understanding of the algorithm to correctly interpret test results.**	**Caution to be employed while mapping LSTM results to temporal data**

6.3 TEST CASES:

- **User Interface Test Cases:**

Test Case ID	Test Scenario	Test Steps	Test Data	Expecte Results	Actual Reslts	Pass/Fail
UT01	Ascertain that the app's navigation and other necessary activities run smoothly.	1. Perform all possible navigation paths and check backward/forward movability	-	No glitches/errors	Observed minute errors	Fail
UT02	Test whether user inputs from app are reflected properly in the database	1. Perform the user input actions	-	Data reflected within 5 minutes	Data reflected within 2 minutes	Pass

➢ **User Acceptance Test Cases**

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Reslts	Pass/Fail
UAT01	Test whether app is user friendly	1. Use app for defined time period 2. Evaluate the level of ease	-	No undesirable traits	No undesirable traits	Pass
UAT02	Test whether designed logo is clearly visible	1. Open app and check visibility	-	Clearly visible	Clearly visible	Pass

6.4 Testing Methods Used:

Classification of tests	Testing Method	Points to remember
Technical Analysis tests	Follow the popular testing paths defined by the analyst community and ensure the approach is suitable for the forex market	-
LSTM tests	Use recommendations from literature review and understanding of the algorithm to correctly interpret test results.	Caution to be employed while mapping LSTM results to temporal data

UI tests	**The method that is best suitable for these tests is straightforward- run the system as intended assuming yourself to be the end user.**	**-**

Chapter 7

RESULT AND DISCUSSION

We observed remarkable results using LSTMs. They really work a lot better for most sequence tasks.

LSTM model Prediction Result plot

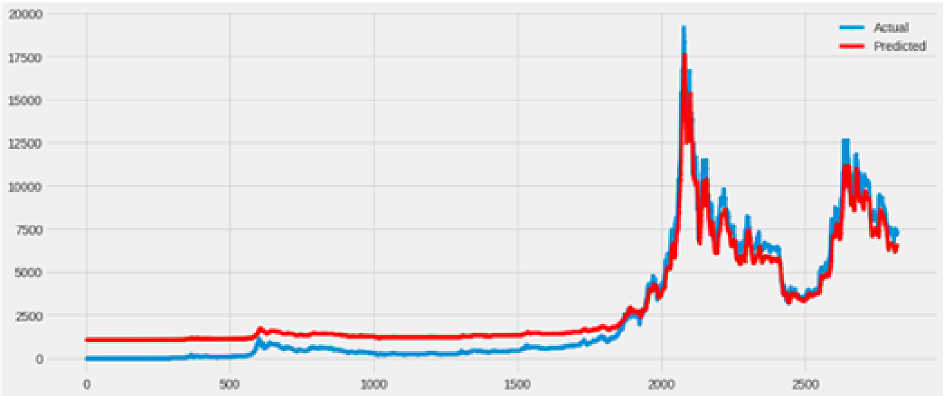


FIG.13.1 LSTM MODEL RESULT PLOT

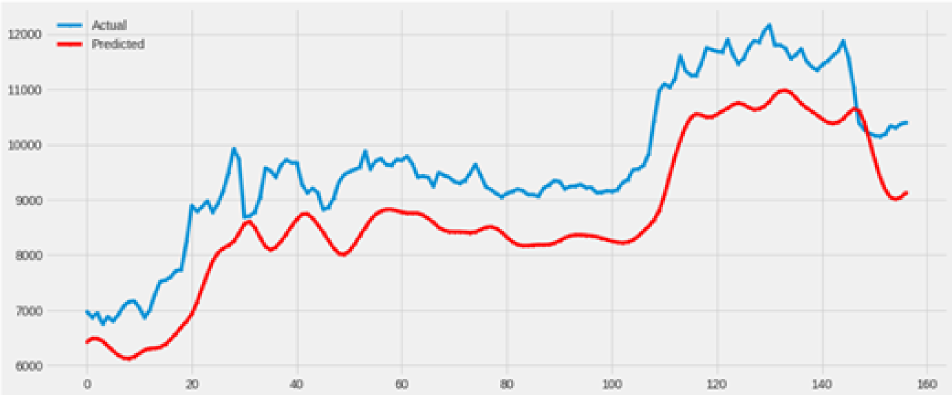


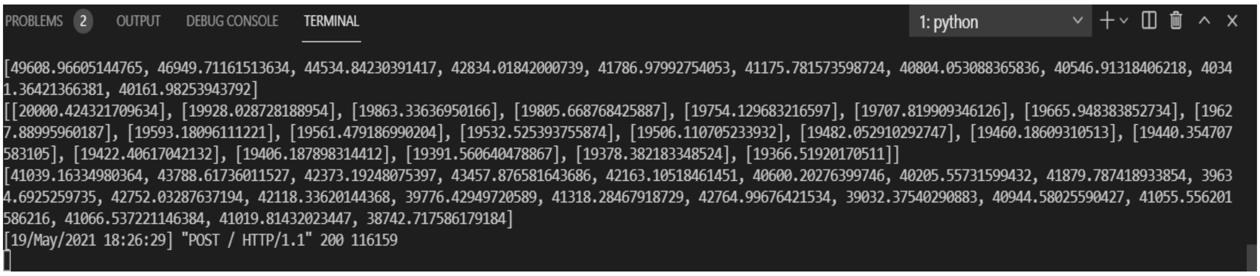
FIG.13.2 LSTM MODEL RESULT PLOT

- from sklearn.metrics import mean_absolute_error, mean_squared_error

```
Train RMSE: 0.04792889021418374
Train MAE: 0.20979423661732005
Test RMSE: 0.0544482676350978
Test MAE: 0.22614883920646772
```

Prediction Result:

Figure 13.3: Prediction Result



Chapter 8

CONCLUSIONS

8.1 Conclusion

In building our model, the steps involved are as follows:

During visualization of our historical dataset we encountered a quite no of missing values.

	Total Missing Values	Missing %
Timestamp	0	0.000000
Open	1241716	27.157616
High	1241716	27.157616
Low	1241716	27.157616
Close	1241716	27.157616
Volume_(BTC)	1241716	27.157616
Volume_(Currency)	1241716	27.157616
Weighted_Price	1241716	27.157616

These Missing values are imputed using Linear Interpolation Imputation Technique. As we can see the results after imputation in below fig.

```
   Volume_(Currency)  Weighted_Price
0           2.000000            4.39
1           2.436653            4.39
2           2.873305            4.39
3           3.309958            4.39
4           3.746611            4.39
Timestamp            0
Open                 0
High                 0
Low                  0
Close                0
Volume_(BTC)         0
Volume_(Currency)    0
Weighted_Price       0
dtype: int64
```

Then we did the Exploratory data analysis in which we have visualized the dataset using KDE’s density plot which concluded that there is a downward trend in Crypto Currency prices from Dec-17 onwards till March 2019.

Further we did Time resampling as Examining Crypto Currency price data for every single day isn’t of much use to financial institutions, who are more interested in spotting market trends. To make it easier, we use a process called time resampling to aggregate data into a defined time period, such as by month or by quarter. Institutions can then see an overview of Crypto Currency prices and make decisions according to these trends. Further we did 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.** We performed statistical tests like KPSS and ADF to confirm our understanding.

The output of the KPSS test contains 4 things:

- The KPSS statistic
- p-value
- Number of lags used by the test
- Critical values

The **p-value** reported by the test is the probability score based on which you can decide whether to reject the null hypothesis or not. If the p-value is less than a predefined alpha level (typically 0.05), we reject the null hypothesis.

The **KPSS statistic** is the actual test statistic that is computed while performing the test.

The number of **lags** reported is the number of lags of the series that was actually used by the model equation of the kpss test.

In order to reject the null hypothesis, the test statistic should be greater than the provided critical values. If it is in fact higher than the target critical value, then that should automatically reflect in a low p-value. That is, if the p-value is less than 0.05, the kpss statistic will be greater than the 5% critical value.

In **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**.

The KPSS and ADF test concluded that KPSS says series is not stationary and ADF says series is stationary. which means series is **difference stationary**, we used **differencing** to make series stationary.

After that we used Rolling Windows Method during our Feature Extraction as 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. 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. In this, we used a rolling mean for 3, 7 and 30 days which resulted into that Our data becomes a lot less noisy and more reflective of the trend than the data itself.

Further we did our Model Building Using LSTM, In which we did cross validation i.e we measure the performance of our forecasting model, We typically split the time series into a training period and a validation period which is called fixed partitioning. we opted for a ***hold-out based validation***.

Hold-out is used very frequently with time-series data. In this case, we select all the data for 2020 as a hold-out and train our model on all the data from 2012 to 2019.

We used the LSTM model where Initialized the LSTM and added Four LSTM layer and some Dropout regularisation and then we compiled the RNN. Further we Fitted the RNN to Training set and Finally we did the prediction and performance checking and We observed remarkable results using LSTMs.

Through the use of Deep learning, estimating the following day's closing price of Bitcoin has been achieved and implemented within the application. Due to the limitations of the system's prediction model, the estimate of close price is only applicable of upto 30 days and cannot be used for long term predictions, thus the system is retrained once per day on updated currency data. The predictions are delivered to the user via a graph, displaying previous predictions versus actual close prices.

Throughout the process as a whole there have been many new experiences, owing particularly to working as a team on such an important project. There were of course disagreements along the way, all fortunately being resolved in a professional and calm manner. The experience gained throughout the development process both in terms of applied knowledge and interpersonal skills is invaluable, and will of course be reflected upon for future projects, academic or career-based.

With the above in mind, we can conclude that the initial objectives for this project were met in some capacity or another. Both the extensive research outlined in this dissertation and observation of the Bitcoin Predictor web application has emphasized that one cannot predict cryptocurrency prices. One can however make an educated guess as to what future prices might be. While our machine learning model is not intended to provide very accurate result, it is close to accurate as long as there is no sudden major shift in the market. Of course, this is the volatility of cryptocurrency yet again reiterating the impossibility of predicting prices.

Summary:

- We are visualising historical bitcoin data.
- The Missing values in the dataset is handled by using Imputation Technique i.e Interpolation which imputes the missing values in time-series.
- Exploratory data analysis is done by visualising using KDEs, using lag plots & further time resampling i.e aggregate data into a defined time period, such as by month or by quarter.
- Feature extraction is done using rolling windows technique.
- Model Selection and Model Building.
- Prediction using LSTMs
- Prediction is done using Web application of next 30 days.

Hence we overall Concluded that,

Our Proposed model has been succeeded to provide the result prediction bitcoin from Historical bitcoin dataset. Our model with time series techniques can build produce the results and the results can predict the price for the next 30 days with split the data to train and test that we mention above. Afterward, as we mentioned before in the article, the Crypto Currency market is influenced by many uncertainty factors. The Cryptocurrencies are influenced by many uncertainties factor such as political issue, the economic issue at impacted to local or global levels. So prediction price bitcoin using LSTM can't good enough to make the decision to invest in bitcoin, it is another side for taking the decisions.

8.2 Limitations of the System

For evaluating the performance of the project, we use validation split of i.e after training we validate before testing about the bias as

```
history = regressor.fit(X_train, y_train, validation_split=0.1, epochs = 50, batch_size = 32, verbose=1, shuffle=False)
plt.plot(history.history["loss"], label= "train loss")
plt.plot(history.history["val_loss"], label= "validation loss")
plt.legend()
```

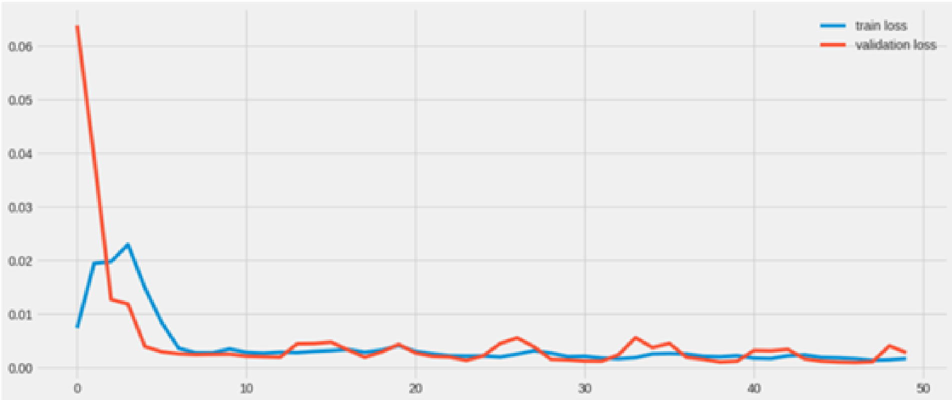


FIG 14 : Train and Validation Loss Plot

The lower the **loss,** the better a model (unless the model has over-fitted to the training data). The loss is calculated on **training** and **validation** and its interpretation is how well the model is doing for these two sets.

So, after increasing the number of epochs to 50 and beyond we notice that loss gradually decreases hence we optimized it a bit & we will try to optimize it further by maybe improving number of lag features can be increased beyond 100 to help learning the model.

8.3 Future Scope of the Project

The expected learning outcomes for this project included applying appropriate research and development methodologies, demonstrating awareness of innovative technologies and incorporating them into the project where relevant, and the ability to critically evaluate the work and any potential for future work. Our model is pretty generic and can be used for other datasets.

One interesting direction of future investigation might be analysing the correlation between different cryptocurrencies and how would that affect the performance of our model. While most of the project relies on faithful technologies such as Python, an effort can be made to integrate newer technologies such as TensorFlow for Machine Learning, where it was deemed important to take advantage of the newest innovations in the field.

While the project achieved its initial objectives in some capacity or another, the point at which any project is complete and unable to be improved upon is difficult to pinpoint. The preliminary planning of this project inevitably meant most of the technologies chosen were chosen for the right reasons, and if not were quickly replaced by more appropriate technologies. While the team is content with the technologies implemented in the final solution, there is of course opportunity for improvement with regard to extra features in the web application.

Wider Variety of Cryptocurrencies: Of course, it would be ideal to have all the major cryptocurrencies that are available in the market model is and the recently developed Bitcoin Cash implemented in and being predicted by this application. Due to time constraints and memory constraints with Heroku, it was deemed suitable to focus our efforts on the price prediction of only Bitcoin cryptocurrency.

Natural Language Processing: As mentioned in the initial Objectives, natural language processing can and has previously been used to determine fluctuations in prices of cryptocurrency. This project could greatly benefit from such technology, through implementing a component which would search given sites for negative or positive discussions on cryptocurrencies. This information could be used to predict an incoming change of price, or simply display the changing popularity of given cryptocurrencies from day to day

Encourage trading and exchange:

Allow the user to do cryptocurrency exchange and trading on our platform.

Open an account with us:

As of now Bitcoin Predictor web application only acts as an informative website and user can only create an account to predict Bitcoin price. We are planning to expand this so that the user can open his Demat and other account to sell or purchase cryptocurrencies.

Wider varieties of cryptocurrencies: Of course, it would be ideal to have all the major cryptocurrencies such as Ethereum, Litecoin, Ripple and the recently developed Bitcoin Cash implemented in and being predicted by this application.

Long term predictions: In addition to natural language processing, a neural network model for long-term prediction could also be implemented. Unfortunately, the current application only estimates short-term values effectively, and the addition of a long-term estimate using both machine learning and natural language processing would make this application more attractive to those considering long-term investment in cryptocurrency.

REFERENCES:

[1]https://www.researchgate.net/publication/340567768_Bitcoin_Price_Prediction_using_ARIMA_Model

[2]https://www.researchgate.net/publication/307598782_Resampling_Strategies_for_Imbalanced_Time_Series

[3] https://en.wikipedia.org/wiki/Long_short-term_memory

[4]https://www.researchgate.net/publication/271978892_Comparison_of_Linear_Interpolation_Method_and_Mean_Method_to_Replace_the_Missing_Values_in_Environmental_Data_Set

[5]https://www.researchgate.net/publication/313867740_A_review_of_missing_values_handling_methods_on_time-series_data

[6]https://www.researchgate.net/publication/336061476_A_Comparative_Study_of_Bitcoin_Price_Prediction_Using_Deep_Learning

[7] http://colah.github.io/posts/2015-08-Understanding-LSTMs/

[8] F. Mai, Z. Shan, Q. Bai, X. Wang, R. Chiang, "How does social media impact bitcoin value? a test of the silent majority hypothesis," Journal of Management Information Systems, vol. 35, no. 1, pp. 19–52, 2018.

[9] D. Chaim, "Blind signatures for untraceable payments," in Advances in Cryptology, pp. 199–203, Springer, Boston, MA, 1983.

[10] L. Hanyecz, "Bitcoin forum post: Pizza for bitcoins?," May 2010. Last accessed 9 Apr 2018. URL: https://bitcointalk.org/index.php? topic=137.0.

[11] Various, "Bitcoin core source code." Last accessed on 8 Apr 2018. URL: https://github.com/bitcoin/bitcoin.

[12] O. Bowcott and A. Hern, "Facebook and cambridge analytica face class action lawsuit," Apr 2018. Last accessed on 8 Apr 2018. URL: https://www.theguardian.com/news/2018/apr/10/ cambridge- analytica-and-facebook-face-class-action-lawsuit.

[13] Merriam-Webster Incorporated, “Bubble; noun.” Last accessed on 11 Apr 2018. URL: https://www.merriam-webster.com/dictionary/ bubble.

[14] Investopedia LLC., “The great recession.” Last accessed on 11 Apr 2018. URL: https://www.investopedia.com/terms/g/great-recession