VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi – 590018

Project Work Phase2(18CSP83)
Report on

"CRYPTO-CURRENCY PRICE PREDICTION USING MACHINE LEARNING"

Submitted by

PRABHAT KUMAR	**(1KT19CS061)**
ROHAN YADAV C	**(1KT19CS072)**
V.RUDRATEJA REDDY	**(1KT19CS073)**
P.SANDEEP KUMAR	**(1KT19CS057)**

Under the guidance of

Internal Guide

Mr. Imran Ulla Khan
Assistant Professor
Department of CSE.

Head of The Department

Dr. Shantharam Nayak
Professor and Head
Department of CSE.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Sri Krishna Institute of Technology

CHIKKABANAVARA, BENGALURU-560090

May - June 2023(AY 2022-2023)

SRI KRISHNA INSTITUTE OF TECHNOLOGY

No.29, Hesaraghatta Main Road, Chimney Hills, Chikkabanavara Post

Bengaluru-560090.

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the project entitled **"CRYPTO-CURRENCY PRICE PREDICTION USING MACHINE LEARNING"** as a part of 18CSP85 Laboratory is a bonafide work carried out by **V.RUDRATEJA REDDY** bearing **USN:1KT19CS073, PRABHAT KUMAR** bearing **USN: 1KT19CS061, ROHAN YADAV C** bearing **USN: 1KT19CS072** and **P.SANDEEP KUMAR** bearing **USN: 1KT19CS057** in partial fulfilment for the award of degree in Bachelor of Engineering in Computer Science Engineering from Visvesvaraya Technological University, Belagavi during the academic year 2022-22. It is certified that all the corrections/suggestions indicated for internal assessment have been incorporated in the report submitted in the department Library. This project report has been approved as it satisfies the academic requirements in respect of project reports prescribed for award of said degree.

……..	……..	……
Signature of Internal Guide	Signature of HOD	Signature of Principal
[Mr. Imran Ulla Khan] Asst.Professor Dept. of CSE, SKIT	**[Dr. Shantharam Nayak]** Prof. and Head Dept. of CSE, SKIT	**[Dr. Mahesha K]** Principal SKIT, Bangalore

External Examination

Name of the Examiners	Signature with date
1……	……
2……	……

DECLARATION

We, do hereby solemnly declare that the contents of project report submitted by us (like the idea, concept, diagrams, figures, videos etc) to the department of CSE/B.E (final year) of SRI KRISHNA INSTITUTE OF TECHNOLOGY, Bangalore, affiliated to VTU Belgaum, Approved by AICTE – New Delhi & Approved by Government of Karnataka, is in majority the courtesy by the referred National /International Journals, Text Books, Papers Presented &Published by the learned authors in the area of their research and also available in the public domain. We sincerely acknowledge all of them.

This report has been submitted by me to satisfy the academic requirement by the affiliating university for the award of Bachelor's Degree in the discipline of my study.

Name:Prabhat Kumar
Signature:

Name: Rohan Yadav C
Signature:

Name:V.Rudrateja Reddy
Signature:

Name: P.Sandeep Kumar
Signature:

ACKNOWLEDGEMENT

We consider it a privilege to whole-heartedly express our gratitude and respect to each and everyone who guided and helped us in the successful completion of this project.

We are grateful to **Dr Mahesha K, Principal, SKIT, Bangalore** and all staff members of the Computer Science and Engineering Department for their kind cooperation. We are extremely grateful to **Dr Shantharam Nayak**, **Professor and HOD, Department of Computer Science and Engineering,** for his co-operation and encouragement. I thank him for providing us an opportunity to carry out the project at college.

We express my deepest gratitude and sincere thanks to **Prof. Imran Ulla Khan**, **Assistant Professor, Department of Computer Science and Engineering** for his valuable guidance during the course of this Project and continuous suggestions to make our Project successful. Finally it's a pleasure and happiness to the friendly co-operation showed by all the staff members of Computer science department.

PRABHAT KUMAR (1KT19CS061)

ROHAN YADAV C (1KT19CS072)

V.RUDRATEJA REDDY(1KT19CS073)

P.SANDEEP KUMAR (1KT19CS057)

ABSTRACT

We analyse the predictability of the Bitcoin market across prediction horizons ranging from 1 to 60 min. In doing so, we test various machine learning models and find that, while all models outperform a random classifier, recurrent neural networks and gradient boosting classifiers are especially well-suited for the examined prediction tasks. We use a comprehensive feature set, including technical, blockchain-based, sentiment-/interest-based, and asset-based features. Our results show that technical features remain most relevant for most methods, followed by selected blockchain-based and sentiment-/interest-based features.

we attempt to predict the Bitcoin price accurately taking into consideration various parameters that affect the Bitcoin value. For the first phase of our investigation, we aim to understand and identify daily trends in the Bitcoin market while gaining insight into optimal features surrounding Bitcoin price. Our data set consists of various features relating to the Bitcoin price and payment network over the course of five years, recorded daily. For the second phase of our investigation, using the available information, we will predict the sign of the daily price change with highest possible accuracy.

In this paper, we use the Linear Regression version of Recurrent Neural Networks, pricing for Bitcoin. To develop a better understanding of its price influence and a common view of this good invention, we first give a brief overview of Bitcoin again economics. After that, we define the database, including data from stock market indices, sentiment, and . in this investigation, we demonstrate the use of LSTM structures with the series of time mentioned above. In conclusion, we draw the Bitcoin pricing forecast results 30 and 60 days in advance.

CONTENTS

List of Figures

Chapter 1

INTRODUCTION

Chapter 1

INTRODUCTION

Cryptocurrency, like Bitcoin, is a digital currency secured by blockchain technology and cryptography, allowing for peer-to-peer transactions. The short-term predictability of the Bitcoin market is analyzed using various machine learning methods and a comprehensive set of predictive features. Machine learning is preferred in this domain due to its flexibility in selecting features and capturing complex relationships. However, there is a research gap in analyzing the short-term predictability of the Bitcoin market comprehensively and considering feature importance in machine learning models.

This study aims to address this gap by comparing different machine learning models to predict market movements of Bitcoin, the most relevant cryptocurrency. Bitcoin is a decentralized digital currency used globally for payments and investment purposes. It can be bought or sold on Bitcoin exchanges, and transactions are stored in a blockchain. The value of Bitcoin fluctuates, making it important to predict its value for investment decisions. Unlike the stock market, Bitcoin's value is not influenced by business events or governments.

Therefore, AI technology is proposed to predict its value accurately. Bitcoin, created in 2009, was the first cryptocurrency and gained popularity later. It operates on blockchain technology, enabling secure transactions without intermediaries or tax examination. Similar to gold in the 19th century, Bitcoin can be exchanged for money, goods, and used for transactions. Mining generates new coins, with a maximum of 21 million Bitcoins, and approximately 11 million already mined. Implementing a system for accurate price prediction has been a challenge. This innovative project utilizes machine learning algorithms to predict short-term changes in Bitcoin prices using historical quantitative data. The application processes real-time data and provides insights to assist investors in decision-making.

1.1 About Domain

Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values. Recommendation engines are

a common use case for machine learning. Other popular uses include fraud detection, spam filtering, malware threat detection, business process automation (BPA) and predictive maintenance.

1.2 Overview

Over $40 billion worth of cryptocurrencies are traded every day. They are among the most popular assets for speculation and investment, yet have proven wildly volatile. Fast-fluctuating prices have made millionaires of a lucky few, and delivered crushing losses to others. Could some of these price movements have been predicted in advance?

In this competition, you'll use your machine learning expertise to forecast short term returns in 14 popular cryptocurrencies. We have amassed a dataset of millions of rows of high-frequency market data dating back to 2018 which you can use to build your model. Once the submission deadline has passed, your final score will be calculated over the following 3 months using live crypto data as it is collected.

The simultaneous activity of thousands of traders ensures that most signals will be transitory, persistent alpha will be exceptionally difficult to find, and the danger of overfitting will be considerable. In addition, since 2018, interest in the crypto market has exploded, so the volatility and correlation structure in our data are likely to be highly non-stationary. The successful contestant will pay careful attention to these considerations, and in the process gain valuable insight into the art and science of financial forecasting.

G-Research is Europe’s leading quantitative finance research firm. We have long explored the extent of market prediction possibilities, making use of machine learning, big data, and some of the most advanced technology available

1.3 Objectives

Core Objectives**:**

- The objective of Crypto Prediction is to ascertain with what accuracy can the price of cryptocurrency be predicted using different machine learning and blockchain algorithm and compare their accuracy.
- Using the technical analysis like PCR ratio, 1\10---10 ratio, Moving Average Convergence/ Divergence(MACD), volume ratio etc. it will predict the direction of the market.
- Specially more useful for intraday traders, and option traders.

1.4 Problem Statement

To develop a model which can help us to predict the price of the crypto currency used (in this case: Bitcoin), with low error rate and a high precision of accuracy. The model will not tell the future, but it might forecast the general trend and the direction to expect the prices to move.

One of the main problem with the bitcoin is price volatility, which indicates the need for studying the "Time-series forecasting of Bitcoin".

The case study based on Time-series forecasting of Bitcoin. The paper considered bitcoin data from April 2015 to December 2022. It proposed this as both regression and classification problem. For regression the paper was predicting price and for classification task where an attempt was made to predict increase/decrease in price. It was performed to predict next day, 30th day & 90th day price.

In this case study, we attempted to predict next day prices based on features of previous day using machine learning & deep learning regression algorithm.

1.4.1 Existing System

There are many systems that are currently existing for crypto related projects and even for the price prediction model, many people have used all kinds of algorithms where they still didn't get to cover some drawbacks.

Some commonly found drawbacks in the previously dine project are:

- Existing models are made for only data analysis and prediction model was not build.
- The classification model was not discussed and performance matrix like accuracy are not calculated.
- Existing model don't provide the sufficient information like prediction, accuracy, market move for the intraday traders.

Drawbacks of existing system

- There was no proper implementation.
- Long process for filter the data
- Low redundancy to perform the prediction
- There was no proper implementation

1.4.2 Proposed System

Our analysis provides a comprehensive guide to sensitivity analysis of model parameters with regard to performance in the prediction. To propose a machine learning based model to additionally compare and discuss the performance of various machine learning algorithm for the given dataset we have used python programming language to feature the backend and front end id developed using flask.

The dataset that we use here is first being processed and then only it is sent to the testing phase for analysis, to check the readiness of the dataset for future phases (especially for next day). Then the model will be trained by splitting the data. Then in the splitter data, it takes a part of data to train itself, a kind of self learning based on the patterns in the data. Then it tests itself by performing predictions by using the data that is left for testing.

Advantages

- The model can be used to predict the crypto future. Performance metrics like accuracy, recall and precision can be calculated.
- Crypto currency future may be predicted and the investments can be made wisely.
- The main focus is on the prediction of crypto currency for intraday traders.
- It works the prediction by taking the coinMarkup cap.
- Quandl is used to filter the dataset by using the Matrix Laboratory (MAT Lab) properties.

Chapter 2

LITERATURE SURVEY

Chapter 2

LITERATURE SURVEY

Year **2021**

Author **: D. Shah and K. Zhang**

Title **: A Bayesian regression and Bitcoin,‖ in 52nd Annual Allerton Conference on Communication, Control, and Computing.**

We discuss the method of Bayesian regression and its efficiency for predicting price variation of Bitcoin, a recently popularized virtual, cryptographic currency. Bayesian regression refers to utilizing empirical data as proxy to perform Bayesian inference. We utilize Bayesian regression for the so-called "latent source model". The Bayesian regression for "latent source model" was introduced and discussed by Chen, Nikolov and Shah and Bresler, Chen and Shah for the purpose of binary classification. They established theoretical as well as empirical efficacy of the method for the setting of binary classification.

In this paper, instead we utilize it for predicting real-valued quantity, the price of Bitcoin. Based on this price prediction method, we devise a simple strategy for trading Bitcoin.

Year **:2020**

Author **: Huisu Jang and Jaewook Lee**

Title **: An Empirical Study on Modelling and Prediction of Bitcoin Prices with Bayesian Neural Networks based on Blockchain Information**

We proposed to predict the bitcoin price accurately taking into consideration various parameter that affect the Bitcoin value. By gathering information from different reference paper and applying in real time, I found the advantages and disadvantages of bitcoin price prediction. Each and every paper has it's own set of methodologies of Bitcoin price predictions. Many papers has accurate price but some other don't, but the time complexity is higher in those prediction, so to reduce the time complexity here in this paper we used an algorithm linked to artificial intelligence named least absolute shrinkage selection operator (LASSO).The other papers used different algorithms like Support vector machine (SVM) coinMarkupcap, Quandl, GNM, Convolutional neural network (CNN) and Recurrent neural network (RNN) etc...

Year **2019**

Author **: Albert Bifet , Nino AntulovFantulin, Tian Guo**

Title **: Bitcoin Volatility Forecasting with a Glimpse into Buy and Sell Orders**

Bitcoins can also be exchanged for other currencies, products, and services. The exchange of the Bitcoins with other currencies is done in the exchange office, where ”buy” or ”sell” orders are stored on the order book . ” Buy” or ”bid” orders represent an intention to buy a certain amount of Bitcoins at some maximum price while ”sell” or ”ask” orders represent an intention to sell a certain amount of Bitcoins at some minimum . The exchange is done by matching orders by price from the order book into a trade transaction between buyers and sellers. we focus on studying the predictive performance of Bitcoin price short-term volatility using both volatility history and order book data. Though Bitcoin is the largest of its kind in terms of total market capitalization value, it still suffers from a volatile price. Volatility as a measure of price fluctuations ,has a significant impact on trade strategies, investment decisions as well as systemic risk . Meanwhile, order book data carrying fined-grained information about price movement and market intentions is proven to be closely related to volatility and influences Bitcoin market with variation over time . Therefore, it is of great interest to data mining and machine learning community to be able to develop predictive models for Bitcoin volatility.

Year **2014**

Author **: Vismay Pandit, Jinesh Doshi, Dhruv Mehta, Ashay Mhatre and Abhilash Janardhan.**

Title **: Predicting the Price of Bitcoin Using Machine Learning**

Time series prediction is not a new phenomenon. Prediction of mature financial markets such as the stock market has been researched at length . Bitcoin presents an interesting parallel to this as it is a time series prediction problem in a market still in its transient stage. As a result, there is high volatility in the market and this provides an opportunity in terms of prediction. In addition, Bitcoin is the leading cryptocurrency in the world with adoption growing consistently over time. Due to the open nature of Bitcoin it also poses another paradigm as opposed to traditional financial markets. It operates on a decentralised, peer-to-peer and thrustless system in which all transactions are posted to an open ledger called the Blockchain. This type of transparency is unheard of in other financial markets. Traditional time series prediction methods such as Holt-Winters exponential smoothing models rely on linear assumptions and require data that can be broken down into tree .

Year :2014

Author : N.Ahmed Surobhi and Abbas Jamalipour

Title : Predicting Bitcoin Prices using Deep Learning.

Project based learning is the methodology in which projects drive knowledge and is used in dedicated subjects without negotiating the coverage of the required technical material. This paper discusses the scheme and delivery of project based learning in computer science engineering as major project which adopts undergraduate creativities and emphasizes on real-world, open-ended projects. These projects foster a wide range of abilities, not only those related to content knowledge or technical skills, but also practical skills. The goal for this innovative undergrad project is to show how a trained machine model can predict the price of a cryptocurrency if we give the right amount of data and computational power. It displays a graph with the .

Year :2019

Author : M.S.kanikar, S.Prabhu, Rahul Chauhan, Akhileshbhat

Title : Predicting Bitcoin Prices using Deep Learning.

Time series prediction is not a new phenomenon. Prediction of mature financial markets such as the stock market has been researched at length . Bitcoin presents an interesting parallel to this as it is a time series prediction problem in a market still in its transient stage. As a result, there is high volatility in the market and this provides an opportunity in terms of prediction. In addition, Bitcoin is the leading cryptocurrency in the world with adoption growing consistently over time. Due to the open nature of Bitcoin it also poses another paradigm as opposed to traditional financial markets. It operates on a decentralised, peer-to-peer and thrustless system in which all transactions are posted to an open ledger called the Blockchain. This type of transparency is unheard of in other financial markets. Traditional time series prediction methods such as Holt-Winters exponential smoothing models rely on linear assumptions and require data that can be broken down into trend, seasonal and noise to be effective . This type of methodology is more suitable for a task such as forecasting sales where seasonal effects are present.

Year : 2019

Author : S. Yogeshwaran, PiyushMaheshwari, ManinderJeet Kaur

Title: : Predicting Bitcoin Prices using Deep Learning.

Project based learning is the methodology in which projects drive knowledge and is used in dedicated subjects without negotiating the coverage of the required technical material. This paper discusses the scheme and delivery of project based learning in computer science engineering as major project which adopts undergraduate creativities and emphasizes on real-world, open-ended projects. These projects foster a wide range of abilities, not only those related to content knowledge or technical skills, but also practical skills. The goal for this innovative undergrad project is to show how a trained machine model can predict the price of a cryptocurrency if we give the right amount of data and computational power. It displays a graph with the predicted values. The most popular technology is the kind of technological solution that could help mankind predict future events. Through the training results, we can get a bitcoin price prediction based on deep learning and get good results. The SDAE-B model is better than the LSSVM model and BP model, which have higher accuracy and lower error. It shows that this method can be used as a promising tool for bitcoin price prediction.

Year : 2019

Author : F. Andrade de Oliveira, L. Enrique ZÃ, and M. de Azevedo Reis, C. NeriNobre

Title : The use of artificial neural networks in the analysis and prediction of stock prices

Each and every paper has its own set of methodologies of bitcoin price prediction. Many papers has accurate price but some other don’t, but the time complexity is higher in those predictions, so to reduce the time complexity here in this paper we use an algorithm linked to artificial intelligence named LASSO(least absolute shrinkage selection operator. The other papers used different algorithms like SVM(support vector machine),coinMarkupcap, Quandl, GLM, CNN(Convolutional Neural Networks)and RNN(Recurrent neural networks) etc.. which do not have a great time management, but in LASSO finding of the results from a larger database is quick and fast..so for this purpose we draw a comparison between other algorithms and the LASSO algorithm, this survey paper helps the upcoming researchers to make an impact in the their papers.

The process happens in the paper is first moment of the research, we aim to understand and find daily trends in the Bitcoin market while gaining insight into optimal features surrounding Bitcoin price. Our data set consists of various features relating to the Bitcoin price and payment network over the course of every years, recorded daily. By preprocessing the dataset, we apply the some data mining techniques to reduce the noise of data. Then the second moment of our research, using the available information, we will predict the sign of the daily price change with highest possible accuracy. In recent years there has been a significant growth of interest in the incorporation of historical series of variables related to stock prediction into mathematical models or computational algorithms in order to generate predictions or indications about expected price movements.

Year : 2020

Author : M. Daniela and A. BUTOI,

Title : Data mining on Romanian stock market using neural networks for price prediction‖.

Stock markets are trading institutions where stocks (equity) and other financial instruments such as bonds are offered for trade. For stocks, the market generally operates a 'willing-buyer, willing-seller' trade, where buyers and sellers prices are matched for a fit. If there is no match, then no trade takes place and waits for a future match or expires. In most stock exchanges, the common and easily accessible market is the equity market (stocks), where the entry investment can be as low as USD1. Another method of stock market prediction is Time series method, which uses historical performance to predict on a time series scale. A time series is a sequence of sampled quantities from an observation out of which discoveries such as periodic distribution can be determined . Other methods in time series prediction are linear regression, auto-regression and Auto- regression Integrated Moving Average (ARIMA). An important characteristic of time series data is the dependence on time, hence current observations depend on past observation in time . In Reinforcement learning the agent's knowledge is reinforced by rewards and punishment. Finally, in Supervised learning, the agent is provided with data so that it can observe input and output pairs so as to formulate a function to map such pairs as applied in Genetic Algorithm (GA) and Artificial Neural Networks (ANN).

Year : 2019

Author : Kalpanasonika ,Sayasri S , Vinothini , SugaPriya

Title : Bitcoin Cost Prediction using Deep Neural Network Technique

One of the most popular cryptocurrency is Bit-coin,which is used worldwide for digital payment and for investment purposes. Bitcoin is decentralized i.e. it is not owned by anyone. Transactions made by Bitcoins are easy as they are not tied to any country.

In this paper, daily price changes for multiple cryptocurrencies are predicted. A few of them are NM, bitcoin, ripple etc., The proposed approach for the price prediction using one of the famous Deep learning neural network concept, i.e. multi-layer perceptron neural network. The plan starts with data pre-processing, in which cleaning the dataset, scaling and normalization are carried out. Next, the independent features in the dataset are extracted which help in predicting the highest price of the cryptocurrency in future. At last, it can predict the costs of cryptocurrencies with different time periods. In our study, we especially focused on a popular cryptocurrency, i.e., bitcoin. From many types of virtual currencies, bitcoin has a great acceptance by different bodies such as investors, researchers, traders, and policy-makers. To the best of our knowledge, our target is to implement the efficient deep learning-based prediction models specifically long short-term memory (LSTM) and gated recurrent unit (GRU) to handle the price volatility of bitcoin and to obtain high accuracy. Our study involves comparing these two time series deep learning techniques and proved the efficacy in forecasting the price of bitcoin.

Year:2018

Author: Juntao Wang, Xiaolong Su

Title: An improved K-Means clustering algorithm

Data Mining is justify technique used to extract, meaning full information from mountain of data and Clustering is an important task in Data Mining process which can be used for the purpose to make groups or clusters of the particular given data set which is based on the similarity between them. K Means clustering is a clustering procedure in which the given data set is divided into K i.e number of clusters. The impact factor of k-means is its simplicity, high efficiency and scalability.

However, is also comprises of number of limitations: random selection of initial centroids, number of cluster K need to be initialized and influence by outliers. In view of these deficiencies, this paper represents a survey of improvements done to traditional k-means to handle such limitations and we will compare K-means clustering algorithm with various clustering algorithm. The algorithm developed density-based detection methods based on characteristics of noise data where the discovery and processing steps of the noise data are added to the original algorithm. By preprocessing the data to exclude these noise data before clustering data set the cluster cohesion of the clustering results is improved significantly and the impact of noise data on K-means algorithm is decreased effectively and the clustering results are more accurate.

Year : 2018

Author : Xiaoxing Yang and Wushao Wen

Title : Ridge and Lasso Regression Models for CrossVersion Defect Prediction.

Sorting software modules in order of defect count can help testers to focus on software modules with more defects. One of the most popular methods for sorting modules is generalized linear regression. However, our previous study showed the poor performance of these regression models, which might be caused by severe multicollinearity. Ridge regression (RR) can improve the prediction performance for multicollinearity problems. Lasso regression (LAR) is a worthy competitor to RR. Therefore, we investigate both RR and LAR models for cross-version defect prediction. Cross-version defect prediction is an approximate to real applications. It constructs prediction models from a previous version of projects and predicts defects in the next version. Experimental results based on 11 projects from the PROMISE repository consisting of 41 different versions show that: 1) there exist severe multicollinearity problems in the experimental datasets; 2) both RR and LAR models perform better than linear regression and negative binomial regression for cross- version defect prediction; and 3) compared with two best methods in our previous study for sorting software modules according to the predicted number of defects, RR has comparable performance and less model construction time . Ridge and Lasso regression are powerful techniques generally used for creating parsimonious models in presence of a 'large' number of features. Here 'large' can typically mean either of two things: Large enough to enhance the tendency of a model to overfit (as low as 10 variables might cause overfitting) .

Year : 2015

Author: D. Shah and K. Zhang

Title : D. Shah and K. Zhang, :Bayesian regression and Bitcoin

Bayesian inference is foundational and use of empirical data as a proxy has been a well known approach that is potentially discovered and re-discovered in variety of contexts over decades, if not for centuries. For example, [5] provides a nice overview of such a method for a specific setting (including classification). The concrete form (4) that results due to the assumption of latent source model is closely related to the popular rule called the 'weighted majority voting' in the literature. It's asymptotic effectiveness is discussed in literature as well, for example [6]. The utilization of latent source model for the purpose of identifying precise sample complexity for Bayesian regression was first studied . In authors showed the efficacy of such an approach for predicting trends in social media Twitter. For the purpose of the specific application, authors had to utilize noise model that was different than Gaussian leading to minor change in the instead of using quadratic function, it was quadratic function applied to logarithm (component-wise) of the underlying vectors.In this paper, we discuss the method of Bayesian regression and its efficacy for predicting price variation of Bitcoin, a recently popularized virtual, cryptographic currency. Bayesian regression refers to utilizing empirical data as proxy to perform Bayesian inference. We utilize Bayesian regression for the so- called "latent source model". The Bayesian regression for "latent source model" was introduced and discussed by Chen, Nikolov and Shah [1] and Bresler, Chen and Shah [2] for the purpose of binary classification.

Year : 2018

Author : Huisu Jang and Jaewook Lee

Title : An Empirical Study on Modelling and Prediction of Bitcoin Prices with Bayesian Neural Networks based on Blockchain Information.

Bitcoin has recently attracted considerable attention in the fields of economics, cryptography, and computer science due to its inherent nature of combining encryption technology and monetary units. This study reveals the effect of Bayesian neural networks (BNNs) by analyzing the time series of Bitcoin process. We also select the most relevant features from Blockchain information that is deeply involved in Bitcoinb's supply and demand and use them to train models to improve the predictive performance of the latest Bitcoin pricing process.

We conduct the empirical study that compares the Bayesian neural network with other linear and non-linear benchmark models on modeling and predicting the Bitcoin process. Our empirical studies show that BNN performs well in predicting Bitcoin price time series and explaining the high volatility of the recent Bitcoin price.

Year : 2019

Author : Farokhmanesh, F., &Sadeghi, M. T

Title : Deep Feature Selection using an Enhanced Sparse Group Lasso Algorithm

Feature selection is an important method of data dimensionality reduction widely used in machine learning. In this framework, the sparse representation based feature selection methods are very attractive. This is because of the nature of these methods which try to represent a data with as less as possible non-zero coefficients. In deep neural networks, a very high dimensional feature space is usually existed. In such a situation, one can take advantages of the feature selection approaches into account. In this paper, first, three sparse feature selection methods are compared. The Sparse Group Lasso (SGL) algorithm is one of the adopted approaches. This method is theoretically very well-organized and leads to good results for man-made features. The most important property of this method is that it highly induces the sparsity to the data. A main step of the SGL method is the features grouping step. In this paper, a k-means clustering based method is applied for grouping of the features. Our experimental results show that this sparse representation based method leads to very successful results in deep neural networks. The implementation was conducted in the PyTorch framework. For LassoNet, we use a onehidden-layer feed-forward neural network with ReLU activation function. We also included a two-hidden-layer network in Section 7 for the matrix completion problem. The number of neurons in the hidden layer was varied within [d/3, 2d/3, d, 4d/3], where d is the total number of features, and the network with the highest validation accuracy was selected and measured on the test set.

Chapter 3

REQUIREMENT SPECIFICATION

Chapter 3:

REQUIREMENT SPECIFICATION

Software Requirements Specification(SRS) is a description of a software system to be developed, laying out functional and non-functional requirements, and may include a set of use cases that describe interactions the users will have with the software.

3.1 HARDWARE REQUIREMENTS

- Processor : Pentium IV
- System : intel i3/i5 2.4 GHz.
- Hard Disk : 40 GB
- Ram : 4/8 GB

3.2 SOFTWARE REQUIREMENTS

- Operating System : Windows 7 and Above, MacOs, Linux
- Software Module : Open CV Jupyter notebook, Linear regression, Dataset.
- Interfacing : The interfacing between the hardware
- Computer : A general purpose PC as a central processing unit for various tasks .

3.3 SYSTEM REQUIREMENTS

Whenever you purchase software or hardware for your computer, you should first make sure your computer supports the system requirements. These are the necessary specifications your computer must have in order to use the software or hardware.

3.3.1 FUNCTIONAL REQUIREMENTS

Functional requirements are those that refer to functionality of the system. That is what services it will provide to the users. A Functional Requirement (FR) is a description of the service that the software must offer. It describes a software system or its component. A function is nothing but inputs to the software system, its behaviour, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform.

Functional Requirements in Software Engineering are also called Functional Specification. In software engineering and systems engineering, a Functional Requirement can range from the high-level abstract statement of the sender's necessity to detailed mathematical functional requirement specifications. Functional software requirements help you to capture the intended behaviour of the system.

Functional Requirements of a system should include the following things : Details of operations conducted in every screen. Data handling logic should be entered into the system. It should have descriptions of system reports or other outputs. Complete information about the workflows performed by the system. It should clearly define who will be allowed to create/modify/delete the data in the system. How the system will fulfil applicable regulatory and compliance needs should be captured in the functional document. Here are some of the functional requirements listing of this project:-

- Using machine learning, this problem helps in forecasting cryptocurrency prices.
- In an effort to more accurately predict cryptocurrency prices quantitatively .
- This system uses the machine learning to forecast cryptocurrency prices based on the previous data .
- Bitcoin dataset (the dataset used is from the last 8 years [2015-2022]).

3.3.2 NON –FUNCTIONAL REQUIREMENTS

Non functional requirements are those that refer to the non –functionality of the system. That tells about how the system is benefit for the user.

The different non –functional requirements are listed below:

- **Performance Requirements**

The system is expected to have reasonable short time response. As the camera is continuously recording the video in traffic signal, once the ambulance is detected in fraction of second the way will be cleared immediately.

- **Reliability**

The system should be 99% reliable. Since it may need some maintenance or preparation for the some particular day, the system does not need to be reliable every time. So, 80% reliability is enough.

- **Efficiency**

Low redundancy to perform the prediction and most useful for intraday traders.

- **Availability**

Bitcoin, Ethereum dataset of few years (min 8 yr.) , and neural network class classifier are available all the time.

- **Maintainability**

The system should be optimized for supportability, or ease of maintenance as far as possible.

Chapter 4

DESIGN

Chapter 4

DESIGN AND IMPLEMENTATION

Design is the key phase of any project. It is the first step in moving from the problem domain. The input to the design phase is the specifications of the system to design. Before the implementation of package, this has to be carried out thoroughly to illuminate any bug, which may be present. The project has to be submitted for system design. The output of the top level designs such as architectural design, or the system design for software system to be built. A design should be very clear, verifiable, complete, efficient, and simple.

4.1 SYSTEM DESIGN

Here the focus is on deciding which modules are needed for the system, the specifications of these modules low and how the module should be interconnected. It includes the following types of design.

- **Database Design**: It describes structure that resides within the system.

- **Architecture Design**: It uses information flow characteristics and maps them into the program structure. Transformations mapping method is applied to exhibit distinct boundaries between incoming.

In the current scenario, all the information and activities relies either on paperwork or on in broke, distributed and automation such as use of Microsoft word. The problem that arises in such a scenario is that the process is too cumber some and demanding as well.

4.2 System Architecture

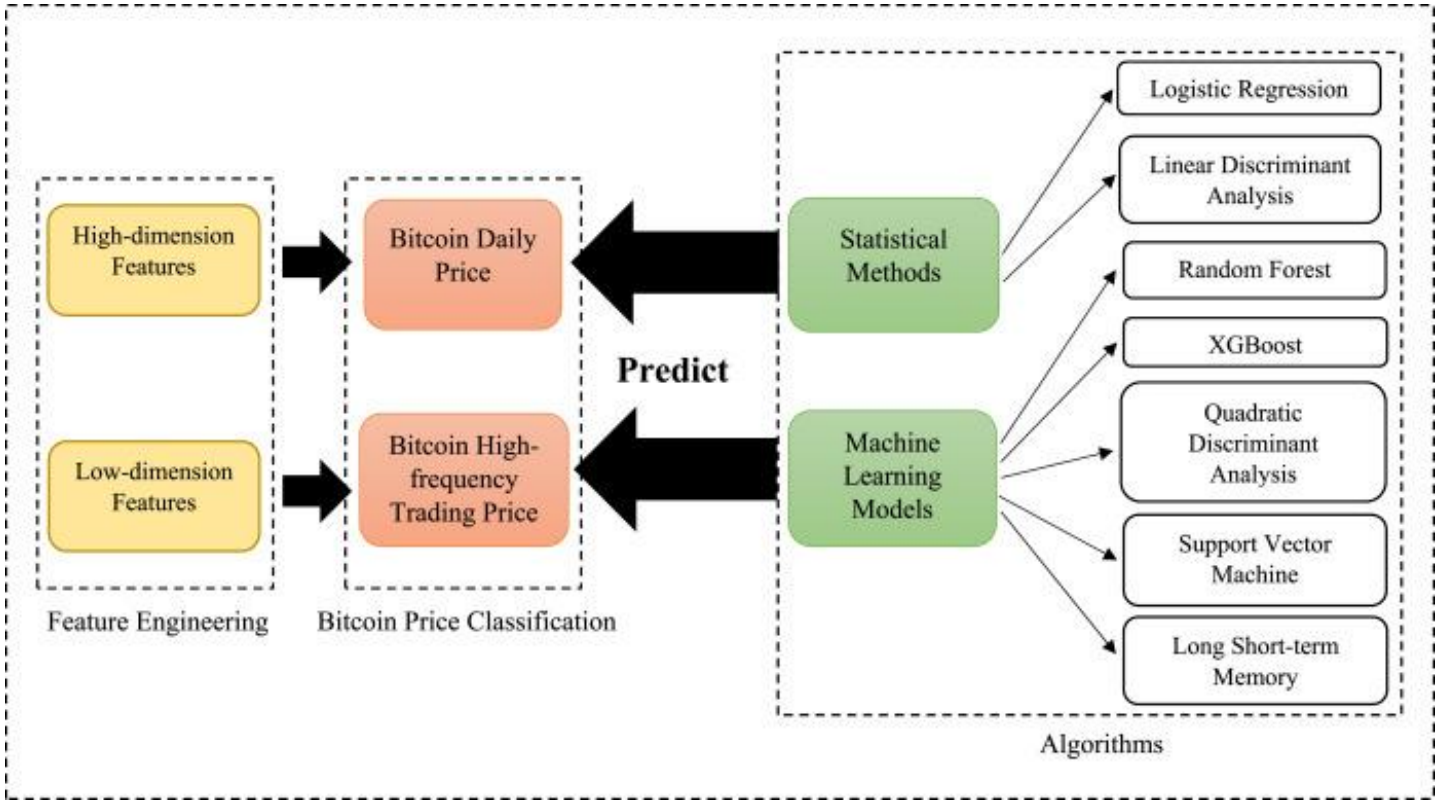

Figure 4.1 : System Architecture of Cryptocurrency price prediction

METHODOLOGY:

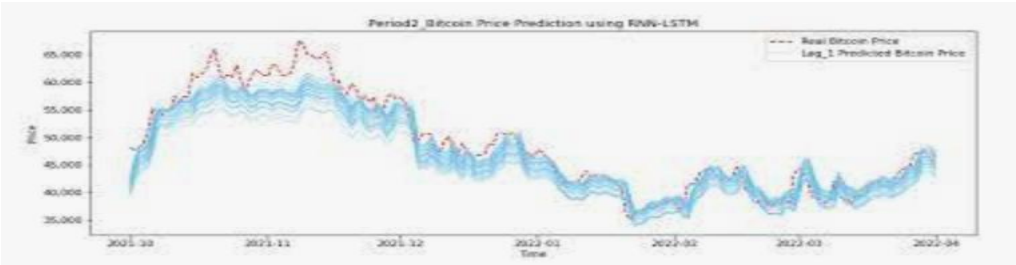

Figure 4.2 : Analysis Model of Cryptocurrency price prediction

PRICE DETECTION OPERATION: When there is uptrend, it will indicate the users that the market will go up for the specific time and now the market will be down.

Higher demand pushes prices up. If demand goes higher than the amount available, the price of that cryptocurrency increases. Some cryptocurrencies have a maximum supply and only increase by a fixed amount, such as bitcoin. Other cryptocurrencies – such as Ether – do not have supply limitations.

ALGORITHM USED

LSTM: Analyse previous data.

Linear Regression: Predict the Low & High with 99.3% accuracy.

RNN (Recurrent neural network) : Used to reduce the error rate.

LSTM: LSTM (Long Short-Term Memory) is a type of recurrent neural network (RNN) commonly used in cryptocurrency price prediction using machine learning (ML). It is effective due to its ability to capture long-term dependencies in sequential data, making it suitable for time series analysis. LSTM models can learn patterns from historical price and volume data, as well as other relevant factors, to make predictions about future cryptocurrency prices. By analyzing past trends and considering complex relationships, LSTM models can identify patterns and provide insights for investors and traders. This enables them to make more informed decisions about buying, selling, or holding cryptocurrencies. Ultimately, LSTM helps to enhance prediction accuracy and improve decision-making in the volatile and dynamic cryptocurrency market.

Linear Regression: Linear regression is used in cryptocurrency price prediction using machine learning because it helps identify relationships between independent variables (such as time, trading volume, and market sentiment) and the dependent variable (cryptocurrency price), allowing for the creation of a linear equation that can estimate future prices based on historical data.

RNN (Recurrent neural network): Recurrent neural networks (RNNs) play a crucial role in cryptocurrency price prediction using machine learning. By considering the sequential nature of price data, RNNs can capture temporal dependencies and patterns. They analyze historical price movements, trading volumes, and other relevant factors to forecast future price fluctuations. RNNs enable the model to learn from past data and make predictions based on this learned knowledge. Their ability to handle sequential data makes them valuable for predicting cryptocurrency prices, aiding in investment decisions and risk management strategies.

4.3 SYSTEM MODULE

A coherent set of tools that is designed to support related software process activities such as analysis and design workbenches support system modeling during both requirements engineering and system design.

4.3.1 DATAFLOW DIAGRAM

A dataflow diagram for cryptocurrency price prediction using machine learning (ML) with LSTM, Linear regression, and RNN consists of several key components. First, historical cryptocurrency price and volume data are collected and preprocessed. These data serve as inputs to the ML models.

Next, the preprocessed data flow into the LSTM model, which captures long-term dependencies and patterns in the sequential data. The LSTM model learns from the historical data to make predictions about future cryptocurrency prices.

Simultaneously, the preprocessed data is also fed into the Linear regression model, which identifies linear relationships between input features and target prices. It uses these relationships to make predictions about future cryptocurrency prices.

Additionally, the data is passed to the RNN model, which is a type of neural network that considers both historical data and the current input to predict future prices. The RNN model leverages its recurrent connections to capture temporal dependencies in the data.

Finally, the predictions from all three models are combined or compared to determine the final prediction for the cryptocurrency price. The dataflow diagram illustrates the flow of data between the models, showcasing the information processing and prediction steps in the cryptocurrency price prediction process.

In the prediction phase, the LSTM model is used to predict future prices based on the testing data. These predictions are evaluated against the actual prices to measure accuracy and make further improvements if necessary.

The overall goal of this dataflow diagram is to utilize LSTM, Linear Regression, and RNN models to analyze historical data, identify patterns, and predict future cryptocurrency prices. By comparing different models, it aims to determine the most effective approach for accurate price predictions.

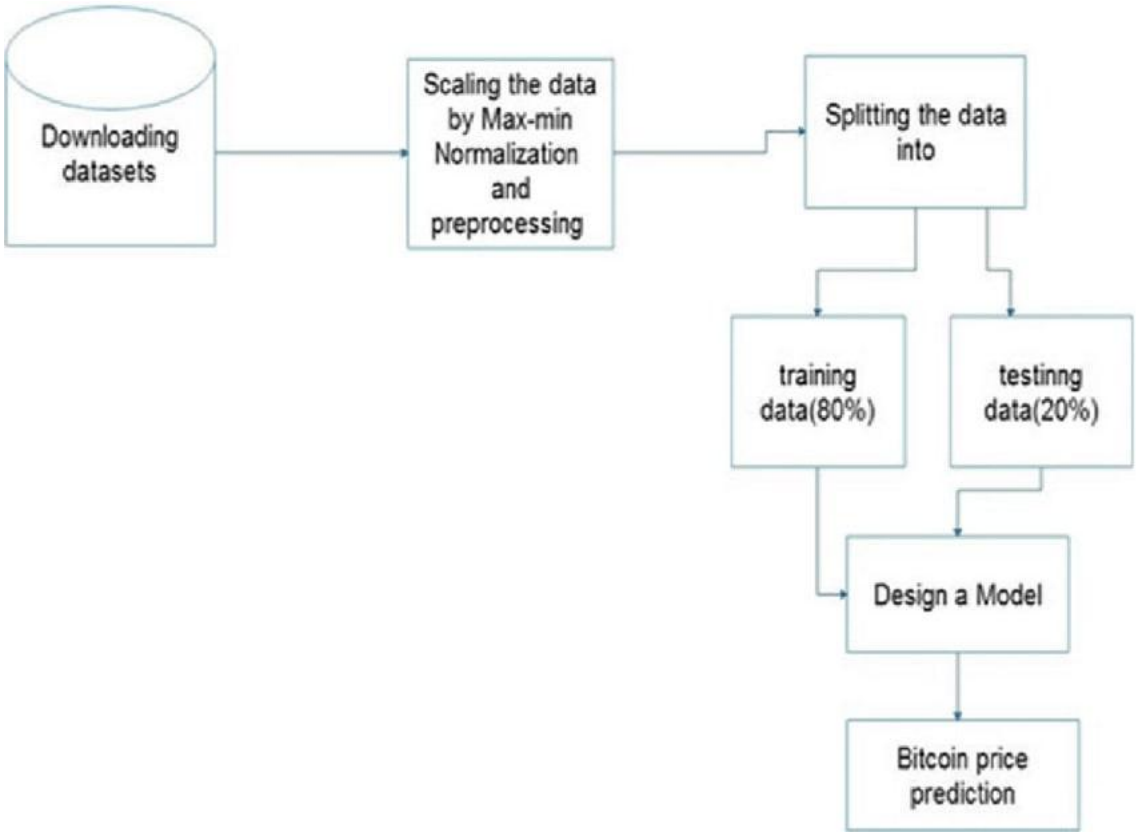

Figure. 4.3: Flow Chart of Cryptocurrency Price prediction

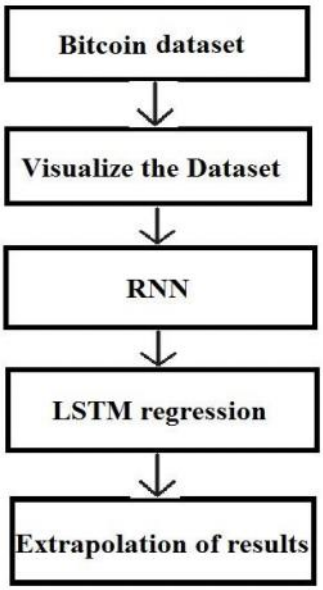

Figure 4.4: System diagram

Bitcoin Dataset will be given to the machine learning algorithm, and it will visualize the dataset given. After visualization, dataset will be send to the RNN which will reduce the error rate and LSTM will analyse and predict the price based on the previous data given through the dataset. And finally it will give the final result.

4.4 MODULE DECOMPOSITION

Module decomposition in cryptocurrency price prediction using ML involves breaking down the task into smaller modules or components, each serving a specific purpose. In this case, three key modules are LSTM, Linear Regression, and RNN.

1. LSTM (Long Short-Term Memory): LSTM is a type of RNN that focuses on capturing long-term dependencies in sequential data. It is employed to analyze historical price and volume data, as well as other relevant factors, to learn patterns and predict future cryptocurrency prices accurately.

2. Linear Regression: Linear regression is a traditional statistical modeling technique used to establish a linear relationship between input variables and the target variable. In cryptocurrency price prediction, linear regression may be employed to model simple relationships between independent variables (such as historical price trends, trading volumes, and market indicators) and the target variable (future cryptocurrency prices).

3. RNN (Recurrent Neural Network): RNN is a class of neural networks that can process sequential data by maintaining hidden states. It is utilized to capture temporal dependencies in cryptocurrency price data. RNN models can analyze the sequence of historical prices and identify patterns or trends that may influence future price movements.

4.4.1 MODULE 1: DETECTION

Cryptocurrency price prediction involves forecasting future values of cryptocurrencies based on historical data and market trends.

4.4.2 MODULE 2: SEGMENTATION

Segmentation in cryptocurrency price prediction involves dividing the historical price data into smaller sections to analyze and predict specific price trends within those segments.

4.4.3 MODULE 3: TRACKING AND CONTROLLING

Tracking and controlling cryptocurrency price prediction using linear regression involves monitoring and managing the predicted price trends of cryptocurrencies using a linear regression model, allowing for informed decision-making and potential risk mitigation.

1: Initialize the classifier with parameter set w_1

2: else

3: Transfer prior for animal representation.

4: Estimate x_t from t-th frame.

5: Store target observation corresponding to z_t.

6: if the number of target observations is equal to some predefined threshold **then**

7: Collect a number of negative samples in the current frame.

8: Use the target observations (positive samples) and negative samples to update w_t.

9: Clear the target observation set.

10**: else**

11: $w_t = w_{t-1}$

12: end if

13: end if

14**: end for**

4.5 Use Case Diagram

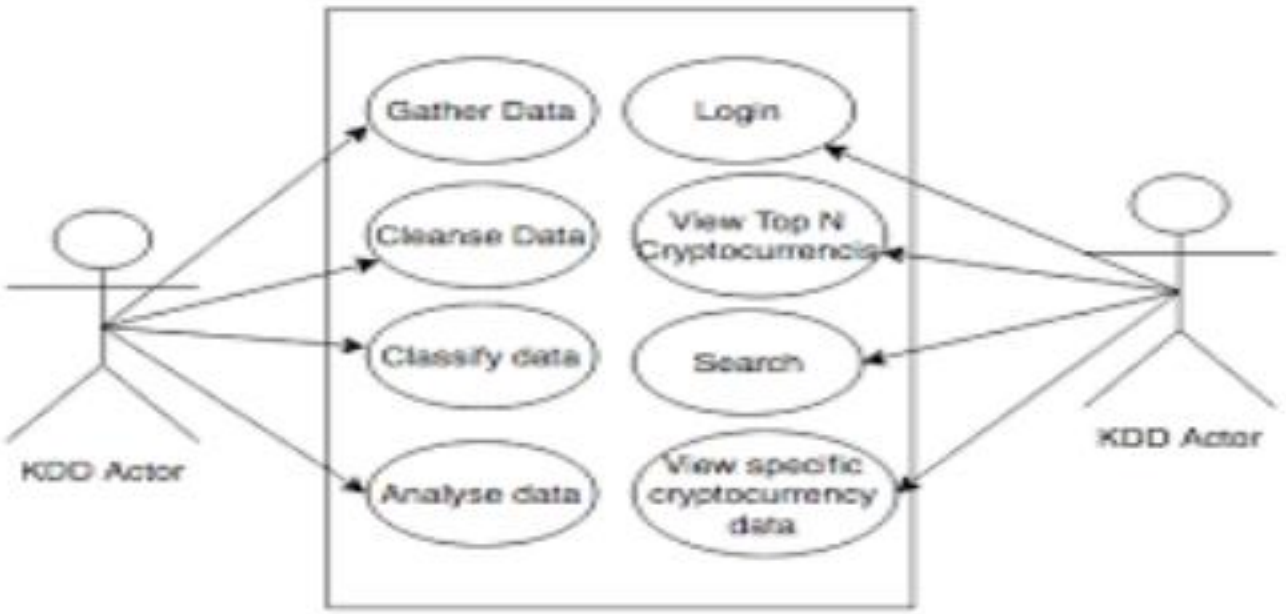

Fig. 4.5: Use Case Diagram of Cryptocurrency price prediction.

4.6 Class Diagram

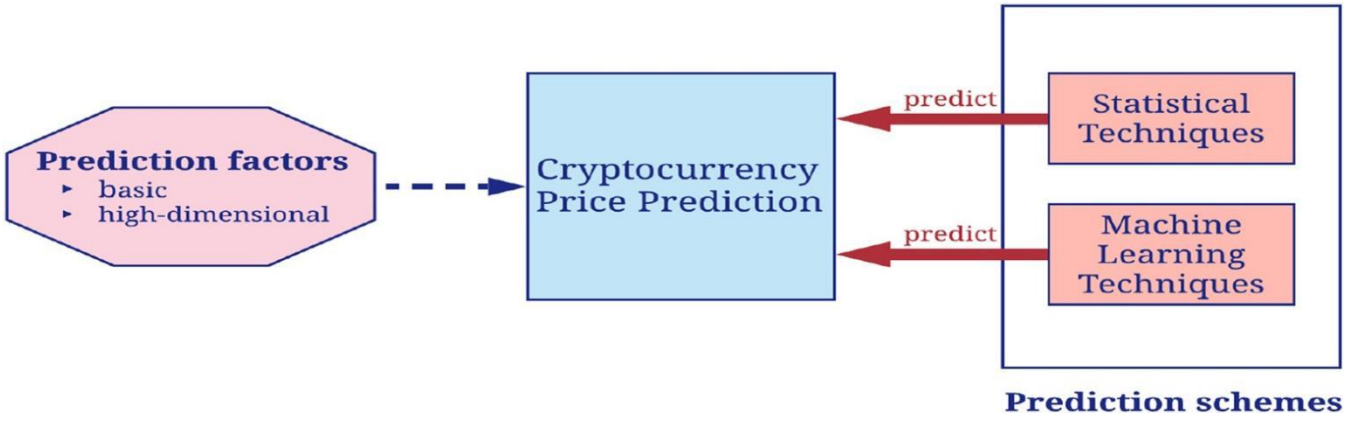

Fig. 4.6: Class Diagram of Cryptocurrency price prediction.

4.7 Sequence Diagram

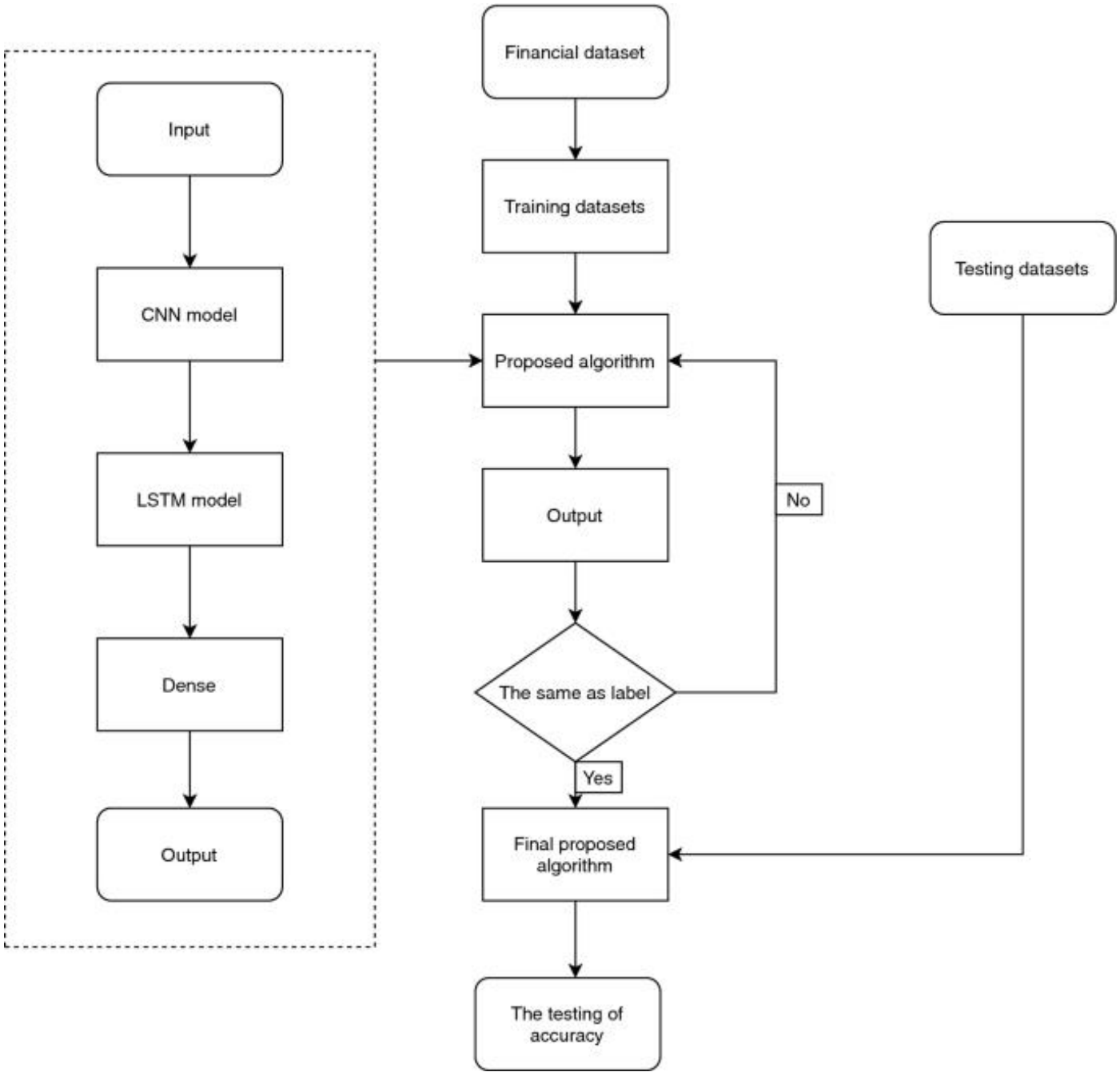

Figure. 4.7: Sequence Diagram of Cryptocurrency price prediction.

4.7 IMPLIMENTATION

In an Information Technology(IT) context, software or hardware implementation encompasses all the post-sale processes involves in something operating properly in its environment, including analyzing requirements, installation, configuration, customization, running, testing, system integrations, user training.[4]

4.7.1 OPEN CV

Open CV is an open source C++ library for image processing and computer vision originally developed by Intel and now supported by Willow Garage. It is free for both commercial and non-commercial use. Therefore it is not mandatory for your Open CV communication to open for free it is a library of many inbuilt functions mainly aimed at real time image processing. Now it has several hundreds of image processing and computer vision algorithms which make developing advanced computer vision applications easy and efficient if you are having any troubles with installing Open CV or configure your Visual Studio IDE for Open CV, please refer to Installing and Configuring with Visual Studio.

Key Features

- Optimized for real time image processing & computer vision applications
- Primary interface of Open CV is in C++
- There are also C, Python and JAVA full interfaces
- Open CV applications run on Windows, Android, Linux, Mac and iOS Optimized for Intel processors

4.7.1.1 OPENCV MODULES

Open CV has a modular structure. The main modules of Open CV are listed below. I have provided some links which are pointing to some example lessons under each module.

- **Core**

This is the basic module of Open CV. It includes basic data structures (e.g.- Mat data structure) and basic image processing functions. This module is also extensively used by other modules like highgui, etc.

- **Highgui**

This module provides simple user interface capabilities, several image and video codecs, image and video capturing capabilities, manipulating image windows, handling track bars and mouse events and etc. If you want more advanced UI capabilities,

- **Imgproc**

This module includes basic image processing algorithms including image filtering, image transformations, color space conversions and etc.

- **Video**

This is a video analysis module which includes object tracking algorithms, background subtraction algorithms and etc.

- **Objdetect**

This includes object detection and recognition algorithms for standard objects. Open CV is now extensively used for developing advanced image processing and computer vision applications. It has been a tool for students, engineers and researchers in every nook and corner of the world.

4.7.1.2 HISTORY

Officially launched in 1999, the OpenCV project was initially an Intel Research initiative to advance CPU-intensive applications, part of a series of projects including real-time ray tracing and 3D display walls. The main contributors to the project included a number of optimization experts in Intel Russia, as well as Intel's Performance Library Team. In the early days of OpenCV, the goals of the project were described as:

- Advance vision research by providing not only open but also optimized code for basic vision infrastructure. No more reinventing the wheel.
- Disseminate vision knowledge by providing a common infrastructure that developers could build on, so that code would be more readily readable and transferable.
- Advance vision-based commercial applications by making portable, performanceoptimized code available for free with a license that did not require to be open or free themselves.

The first alpha version of OpenCV was released to the public at the IEEE Conference on Computer Vision and Pattern Recognition in 2000, and five betas were released between 2001 and 2005. The first 1.0 version was released in 2006. In mid-2008, OpenCV obtained corporate support from Willow Garage, and is now again under active development. A version 1.1 "pre-release" was released in October 2008. The second major release of the OpenCV was on October 2009. OpenCV 2 includes major changes to the C++ interface, aiming at easier, more type-safe patterns, new functions.

And better implementations for existing ones in terms of performance (especially on multi-core systems). Official releases now occur every six months and development is now done by an independent Russian team supported by commercial corporations. In August 2012, support for OpenCV was taken over by a non-profit foundation OpenCV.org, which maintains a developer and user site.

4.7.1.3 APPLICATIONS OF OPENCV

- 2D and 3D feature toolkits
- Egomotion estimation
- Facial recognition system
- Gesture recognition
- Human–computer interaction (HCI)
- Mobile robotics
- Motion understanding
- Object identification
- Segmentation and recognition
- Stereo sis stereo vision: depth perception from 2 cameras
- Structure from motion (SFM)
- Motion tracking
- Augmented reality

To support some of the above areas, OpenCV includes a statistical machine learning library that contains:

- Boosting
- Decision tree learning
- Gradient boosting trees
- Expectation-maximization algorithm
- k-nearest neighbor algorithm
- Naive Byes classifier
- Artificial neural networks
- Random forest
- Support vector machine (SVM)

Programming language

OpenCV is written in C++ and its primary interface is in C++, but it still retains a less comprehensive though extensive older C interface. There are bindings in Python, Java and MATLAB/OCTAVE. The API for these interfaces can be found in the online documentation Wrappers in other languages such as C#,Perl,Ch,and Ruby have been developed to encourage adoption by a wider audience.

All of the new developments and algorithms in OpenCV are now developed in the C++ interface. **Hardware Acceleration**

If the library finds Intel's Integrated Performance Primitives on the system, it will use these proprietary optimized routines to accelerate itself. A CUDA-based GPU interface has been in progress since September 2010. An OpenCL-based GPU interface has been in progress since October 2012.documentation for version 2.4.9.0 can be found at docs.opencv.org.

OS support

OpenCV runs on a variety of platforms. Desktop Windows, Linux, OS X, FreeBSD, NetBSD OpenBSD, Mobile, Android, iOSMaemo, BlackBerry 10.The user can get official releases from Source Forge or take the latest sources from GitHub. OpenCV uses CMake.

4.7.2 SOURCECODE

RESULT VIEW

```
from django.shortcuts import render
from .forms import PredictionForm
from .model.data_collector import *
from .model.model import *
import datetime

def predict(request):
    if request.method == 'POST':
        form = PredictionForm(request.POST)
        if form.is_valid():
            date = form.cleaned_data['date']
            date_str = str(date)
            year, month, day = date_str.split('-')

            year = int(year)
            month = int(month)
            day = int(day)
            symbol = form.cleaned_data['coin']

start_date = datetime.datetime(2016, 1, 1)
            end_date = datetime.datetime(year, month, day)
            start_timestamp = int(start_date.timestamp())
            end_timestamp = int(end_date.timestamp())
```

```
            # Retrieve historical data and preprocess it
            data = get_historical_data(symbol, start_timestamp, end_timestamp)
            data = preprocess_data(data)
            data = add_features(data)

            # Train machine learning models
            actual_prices, model_price, model_return, X_test, y_price_test, y_return_test = train_model(data)

            # Make predictions
            input_data = data.tail(1)
            input_data = input_data.drop(['Price', 'Return'], axis=1)
            predicted_price = model_price.predict(input_data)[0]
            actual_price = model_price.predict(X_test)[-1]
            predicted_return = model_return.predict(input_data)[0]
            print(actual_price)
            return render(request, 'predict.html',
                          {'form': form, 'actual_price': actual_price, 'predicted_price': predicted_price,
                           'predicted_return': predicted_return})
    else:
        form = PredictionForm()

    return render(request, 'predict.html', {'form': form})
def news(request):
    news_data = get_crypto_news()
    for article in news_data:
        published_on_timestamp = article['published_on']
        published_on = datetime.datetime.fromtimestamp(published_on_timestamp)
        article['published_on'] = published_on
context = {'news_data': news_data}
    return render(request, 'news.html', context)
```

```
def home(request):
    return render(request, 'home.html')
def about(request):
    return render(request, 'about.html')
def contact(request):
    return render(request, 'contact.html')
```

apps.py

```
from django.apps import AppConfig
class PredictionConfig(AppConfig):
    default_auto_field = 'django.db.models.BigAutoField'
    name = 'prediction'
```

requirements.txt

```
asgiref==3.6.0
beautifulsoup4==4.11.2
certifi==2022.12.7
charset-normalizer==3.0.1
Django==4.1.7
django-bootstrap4==22.3
idna==3.4
joblib==1.2.0
numpy==1.24.2
```

```
pandas==1.5.3
python-dateutil==2.8.2
pytz==2022.7.1
requests==2.28.2
scikit-learn==1.2.1
scipy==1.10.1
six==1.16.0
soupsieve==2.4
sqlparse==0.4.3
threadpoolctl==3.1.0
tzdata==2022.7
urllib3==1.26.14
```

urls.py

```
from django.urls import path
from . import views

urlpatterns = [
    path('', views.home, name='home'),
    path('predict/', views.predict, name='predict'),
    path('news/', views.news, name='news'),
    path('about/', views.about, name='about'),
    path('contact/', views.contact, name='contact'),
]
```

Chapter 5

TESTING

CHAPTER 5

TESTING AND RESULTS

System should not be tested as a single, monolithic unit. The testing process should therefore proceed in stage where testing is carried out incrementally in conjunction with system implementation. Errors in program components may come to light at a later stage of the testing process. The process is therefore an iterative one with information being fed back from later stage to earlier part of the process.[5]

5.1 TYPES OF TESTS

The various stages that were used in testing this software were as follows:

1. Unit testing
2. Integration testing
3. System testing
    - Validation testing
    - Black Box testing
    - White Box testing
4. Acceptance testing

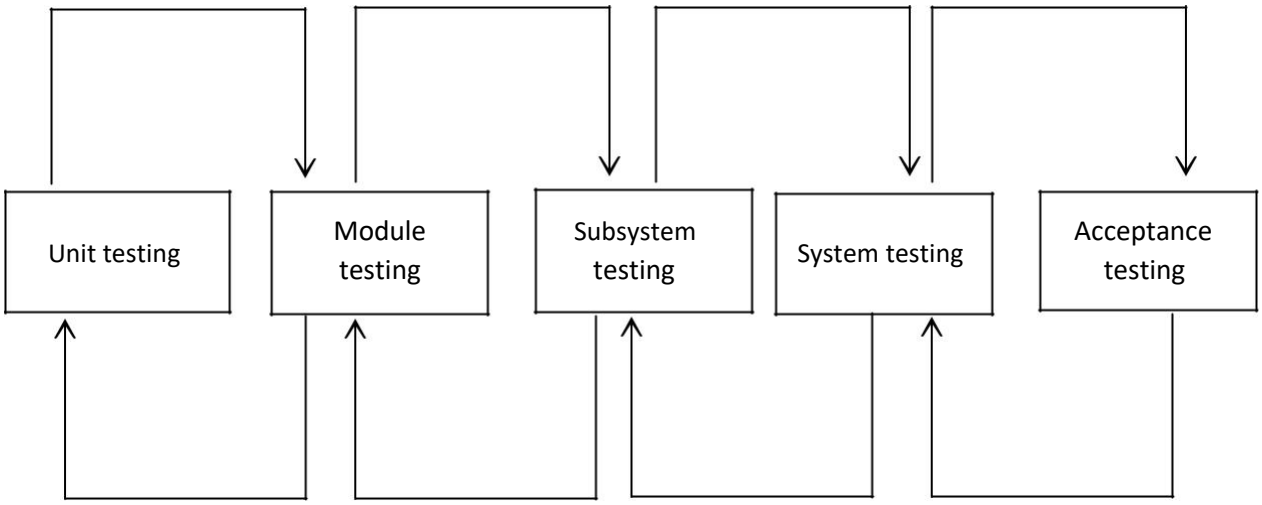

Figure 5.1: Types of testing

5.2 UNIT TESTING

Individual components are to ensure that they operate correctly. Each component is tested independently, without other system components. This system was tested with the set of proper test data for each module and the results were checked with the expected output. Unit testing focuses on verification effort on the smallest unit of the software design module. This is also known as MODULE TESTING. This testing is carried out during phases, each module is found to be working satisfactory as regards to the expected output from the module.

The positioning of camera will plays the major role here, the camera will placed in three different ways such as at left side and right side of the pave ways and at the centre in columns.

The camera will capture the images of the ambulance and the same will be stored in the microcontroller.

5.3 INTEGRATION TESTING

Integration testing is another aspect of testing that is generally done in order to uncover errors associated with flow of data across interfaces. The tested modules are grouped together and tested in small segments, which makes it easier to isolate and correct errors. This approach is continued unit we have integrated all modules to form the system as a whole.

The camera is going to capture the video continuously, the video is divided into number of frames and stored in micro controller and micro controller is going to compare stored image with the ambulance.

5.4 SYSTEM TESTING

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. The following are the system tests that were carried out for the system.

Validation testing:

The validation testing can be defined in many ways, but a simple definition that, validation succeeds when the software functions in a manner that can be reasonably expected by the end user.

- **Black Box testing:**

  Black Box testing is done to find the followings.

- ➢ Incorrect or missing functions.
- ➢ Interface errors.
- ➢ Error in external database access.
- ➢ Performance error.
- ➢ Initialization and termination error.

- **White Box testing:**

  This allows the tester to

  - ➢ Check whatever all independent paths within a module have been exercised at least once.
  - ➢ Exercise all logical decisions on their and false sides.
  - ➢ Execute all loops and their boundaries and within their bounds.
  - ➢ Exercise the internal data structure to ensure their validity.
  - ➢ Ensure whether all the possible validity checks and validity lookups have been provided to validate data entry.

5.5 ACCEPTANCE TESTING

Acceptance testing is a test conducted to determine if the requirements if the specification or contract are met. It may involve physical or performance test. Software developers often distinguish acceptance testing by the system provider from acceptance testing by the customer prior accepting transfer of ownership. In case of software, acceptance testing is performed by the customer is known as User Acceptance Testing (UAT), end user testing, site testing (Acceptance) or field testing (Acceptance).

In Acceptance testing the microcontroller is going to compare, if compared image is ambulance then the signal is toggled else density is calculated and traffic is controlled.

5.6 ERROR HANDLING

An exception is any error condition or unexpected behavior encountered by executing programs. Exception can be raised because of fault in the code or in code called, operating system resources not being available, unexpected conditions and common language run time encounters and so on. Your application can recover from some of these conditions, but not other. While you cannot recover from most run time exceptions.

5.7 RESULTS

- Traffic congestion can be solved.
- Emergency vehicles can reach the destination earliest.
- Traffic density is continuously monitor by video processing and converted into frames.
- The frames are analyzed by various techniques of image processing they are segmented for distinguish between ambulance and other vehicles.
- Traffic signals continuously glow to green as long as emergency vehicle is passed through the traffic and it is allowed to reach its destination.

SNAPSHOTS

Chapter 6

SNAPSHOTS

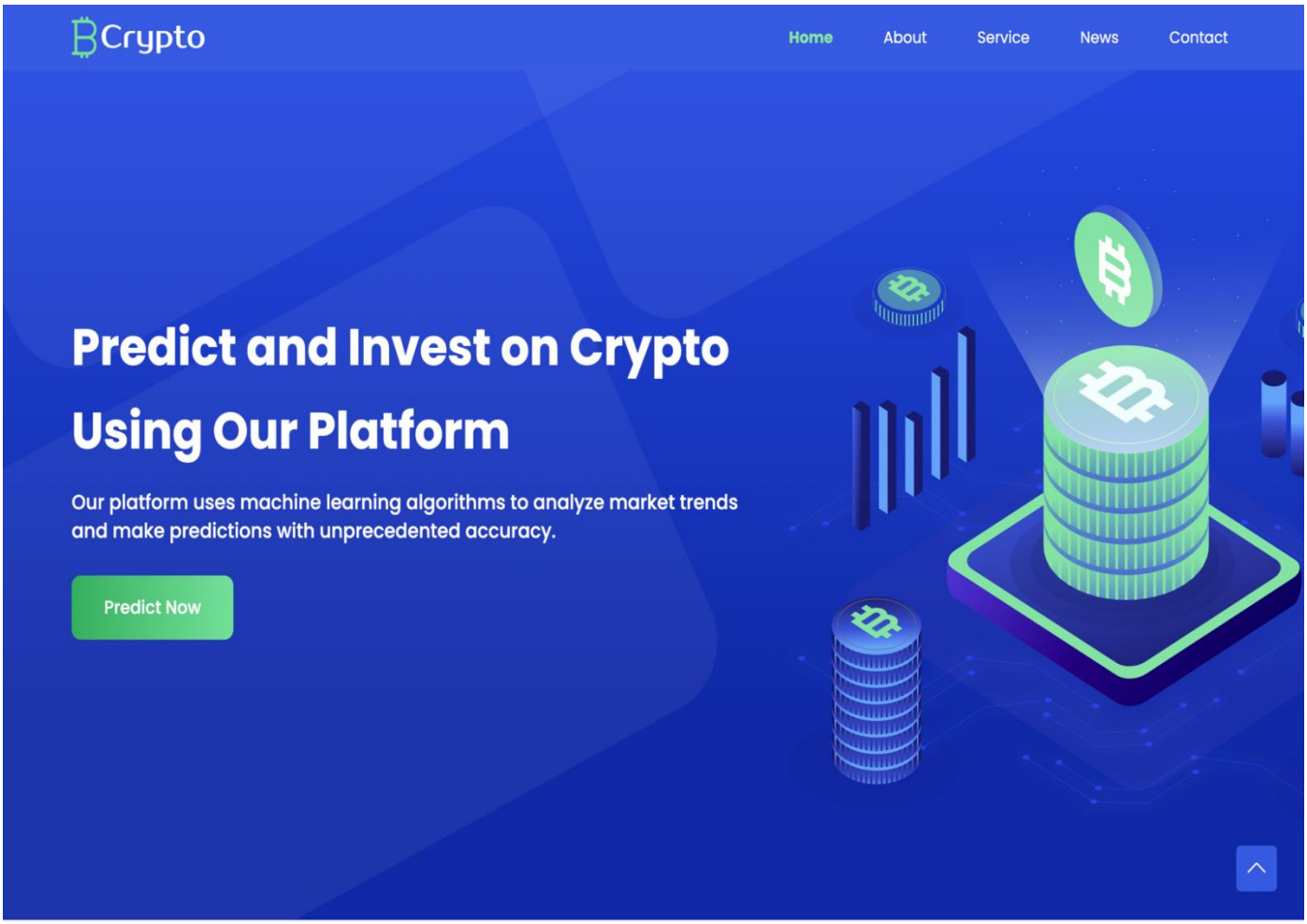

Figure 6.1: Home page.

This is the home page containing total five sections named Home, About, Service, News, Contact.

By clicking on Predict now user will be able to predict the crypto price by entering the date.

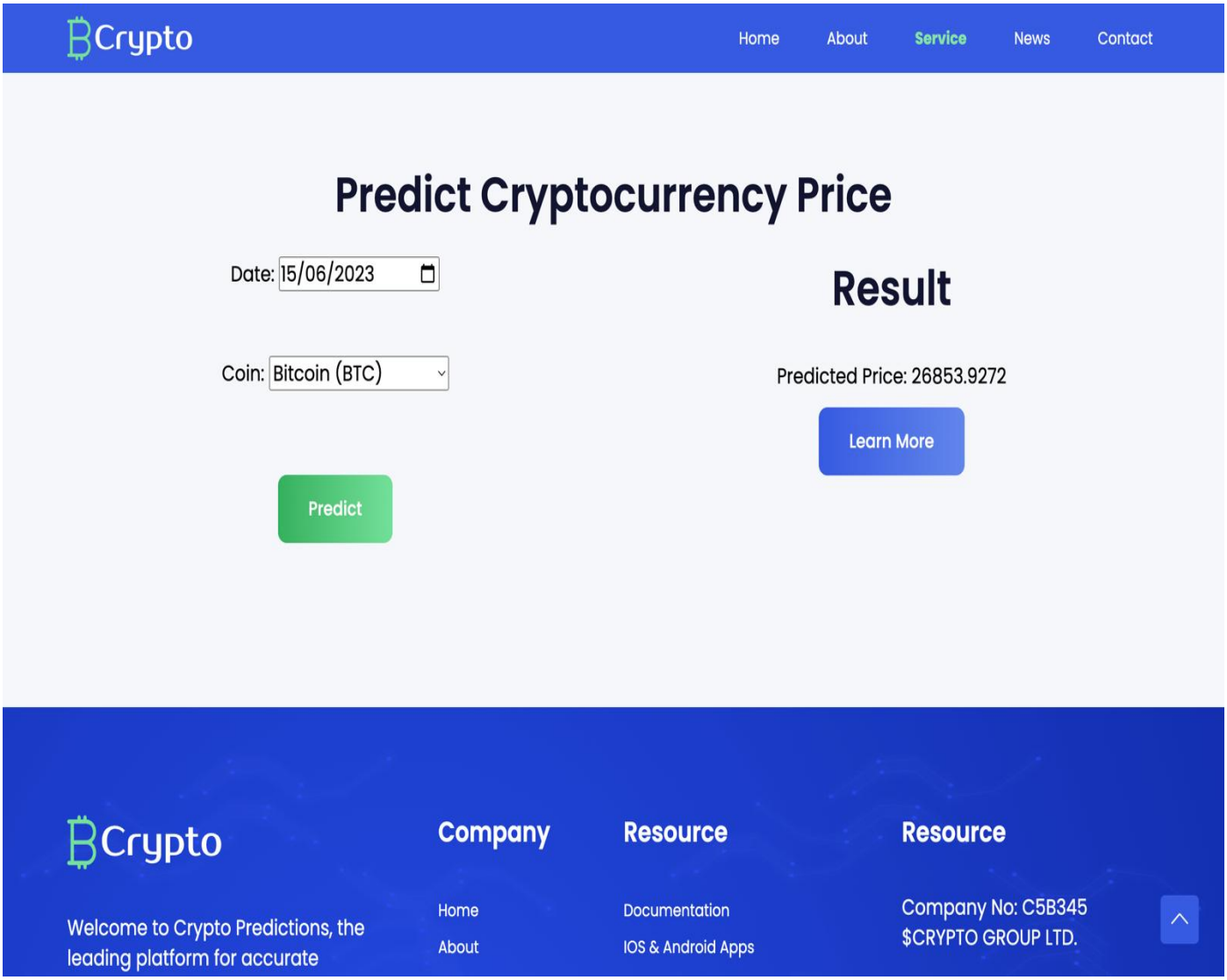

Figure 6.2: Result Predicted

User will enter the date and choose the cryptocurrencies based on their choice and can predict the cryptocurrency price. And by clicking on Learn more button they will be able to access the charts, their profile section, watchlist, portfolio and many more things.

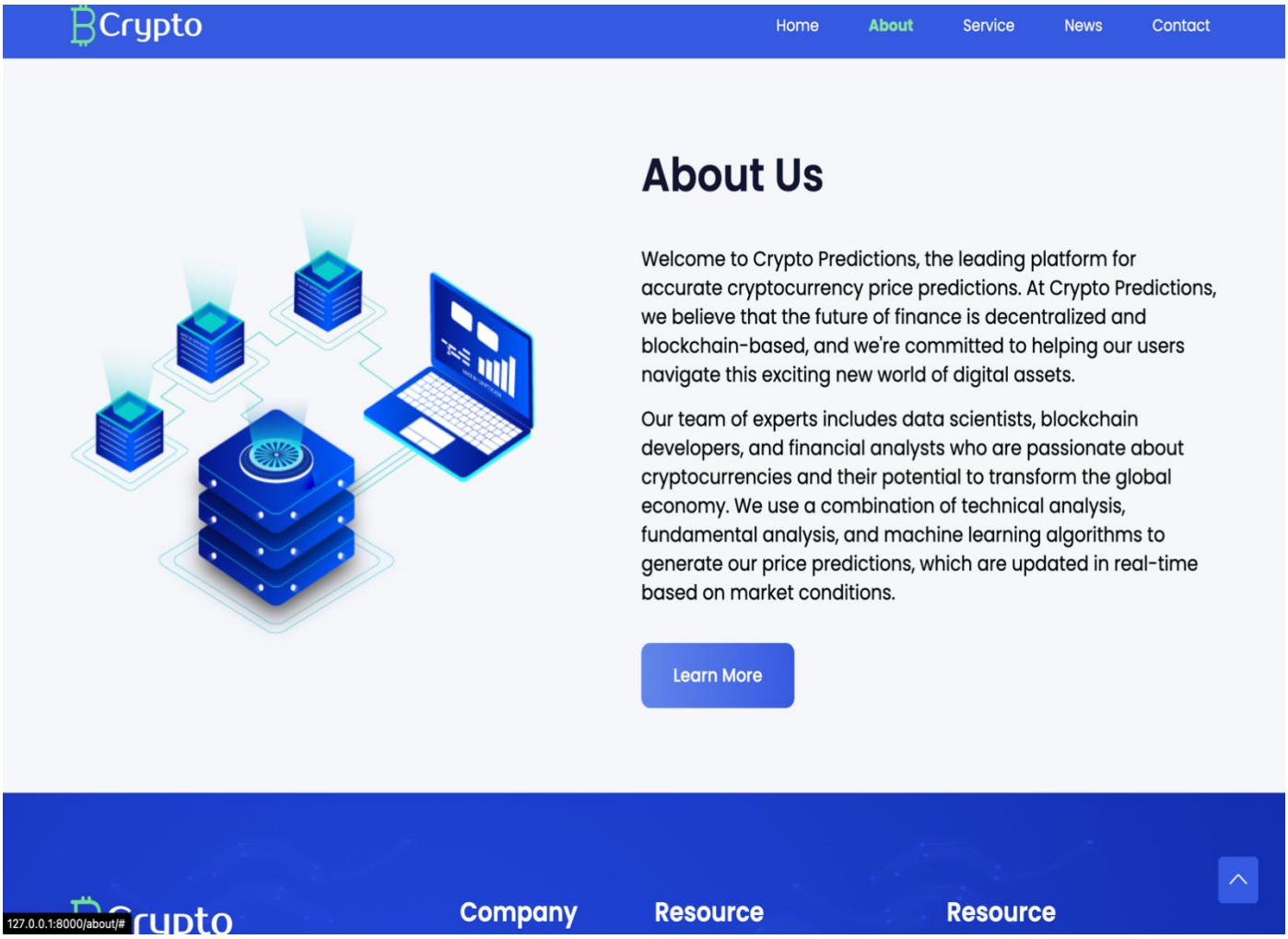

Figure 6.3: About us

This section contains the information about the admin along with the cryptocurrencies. By clicking on Learn More button user can explore more information about the cryptocurrency price prediction using Machine Learning.

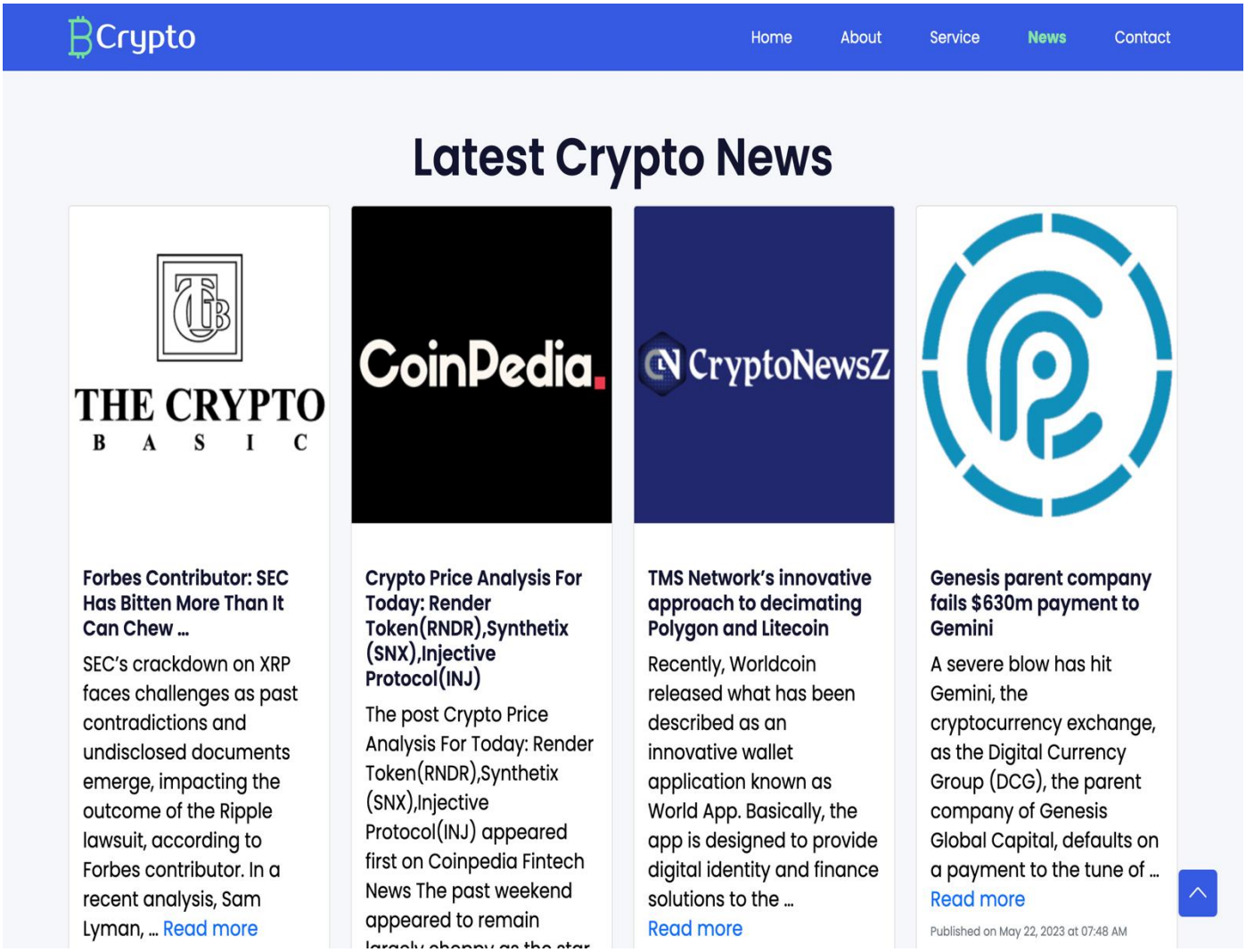

Figure 6.4: News section

This section contains the news related to cryptocurrency . Whenever there will be any news updated related to cryptocurrency market it will be automatically updated here. It provides the latest information about the crypto market which will help the Investors and specially to Intraday Traders.

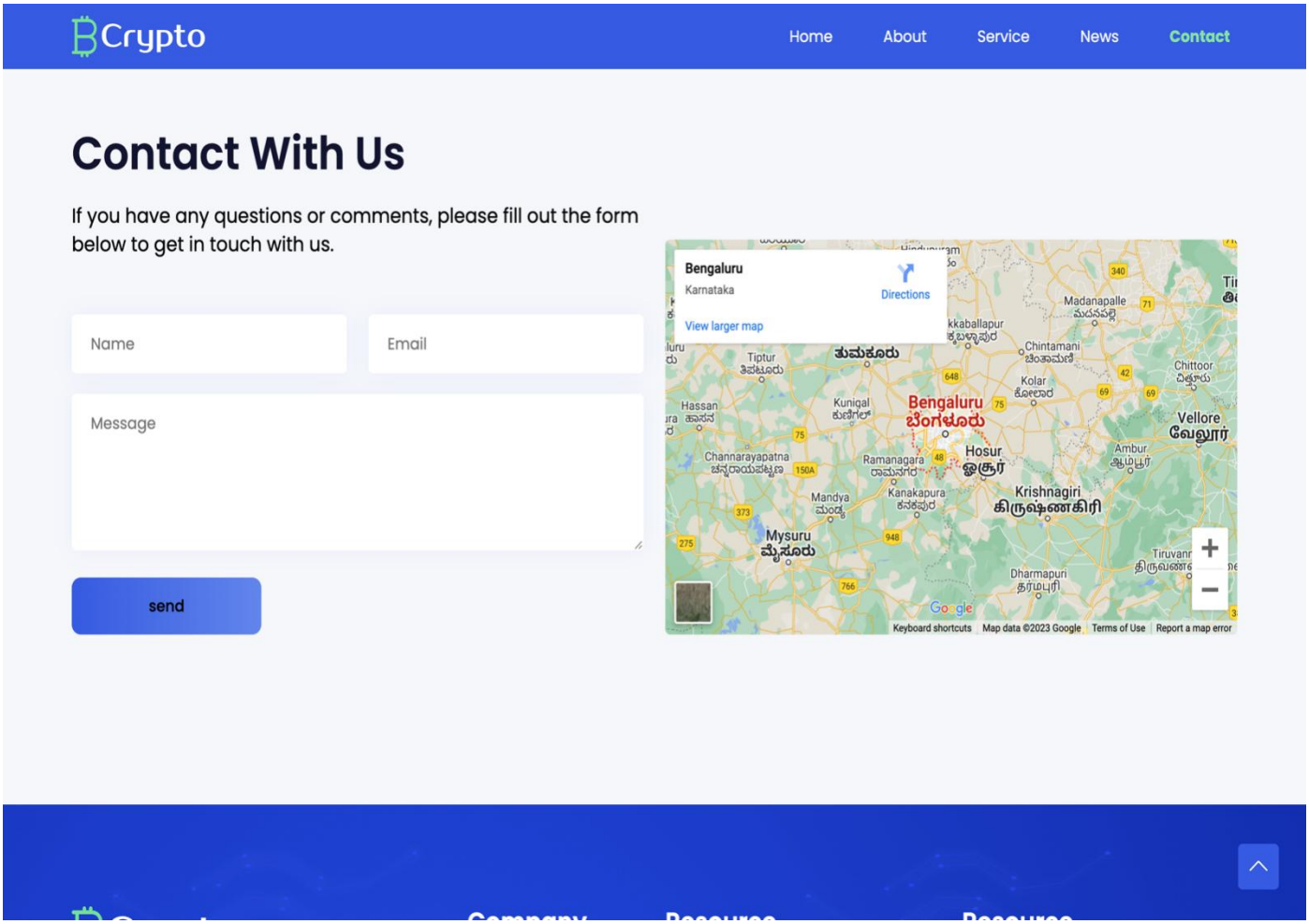

Figure 6.5: Contact us

This section will establish the connection between the admin and user. Users can write the feedback, issues, and if they found any bug in this project.

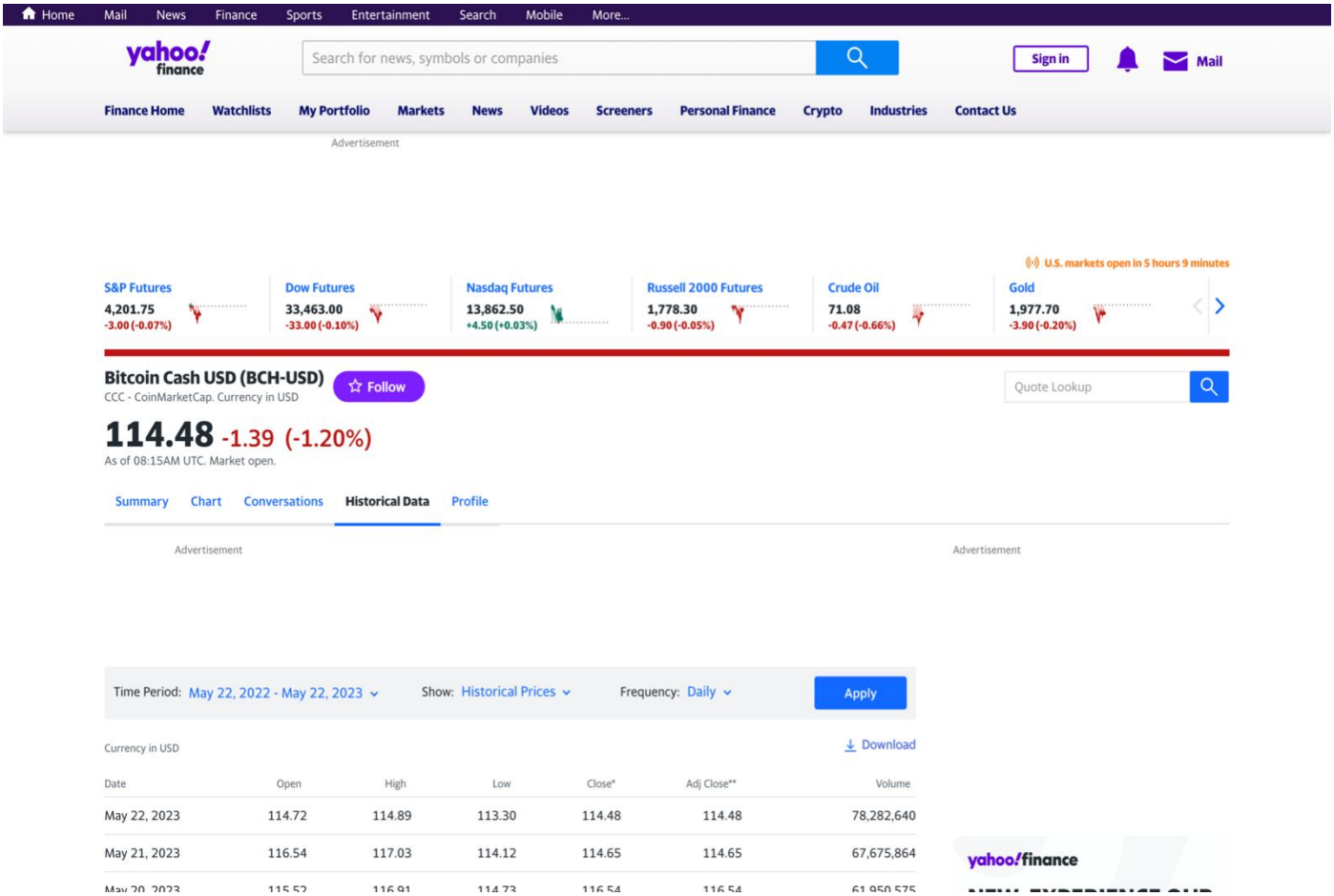

Figure 6.6: Service Page

The service page contains the more details about the crypto market (see figure 6.6) as mentioned in figure 6.2.

The user will be able to see the screener which provides the information of each particular coins which gets updated every minutes so that they can follow the market trend and make the profit with 96.8% accuracy.

User can view the market trend and movement of the USD, Gold, Crude Oil, Dow Features etc. which affect the cryptocurrency market.

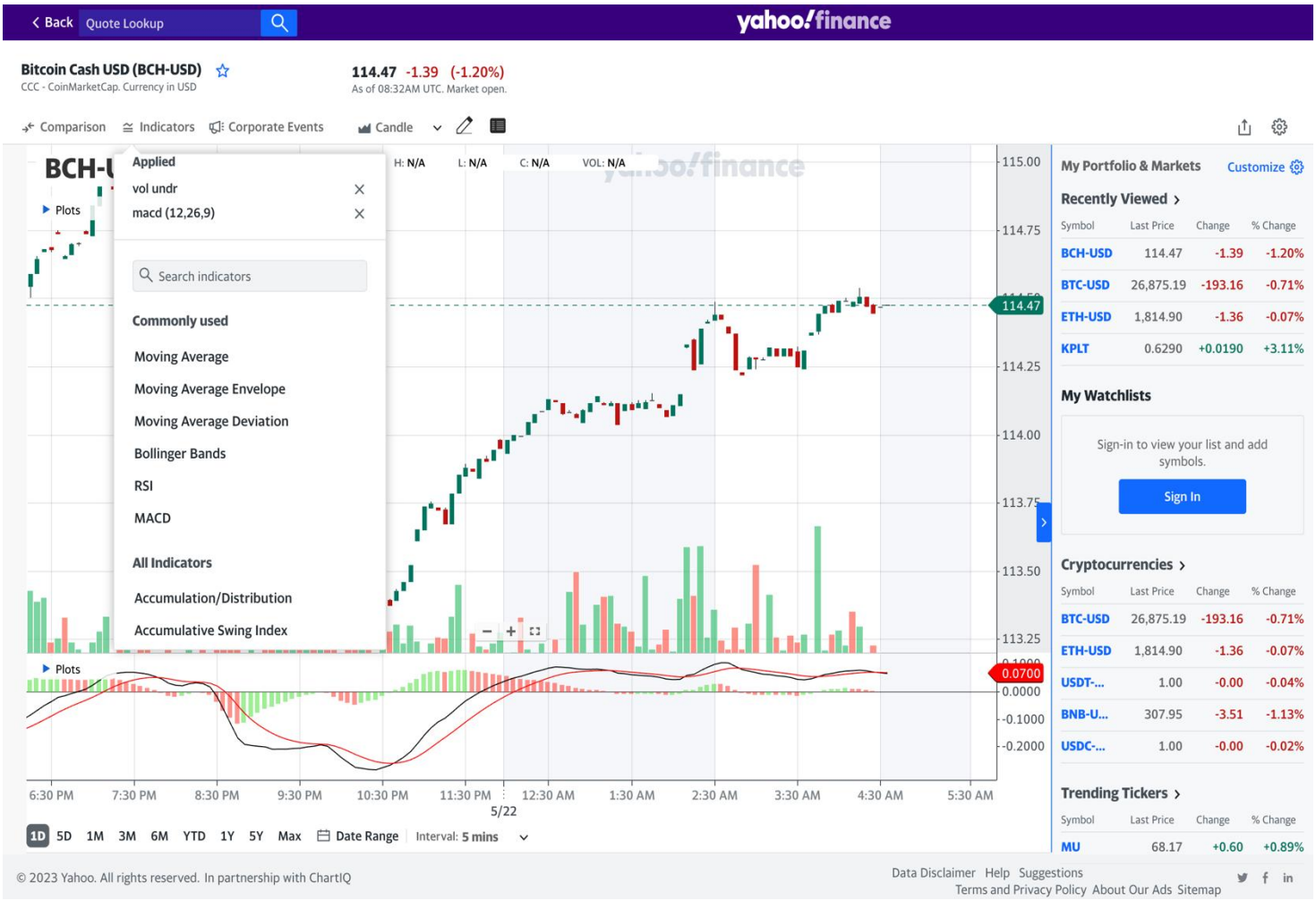

Figure 6.7: Cryptocurrency chart

Traders can trade the cryptocurrency using all the necessary tools like Technical indicators, candle sticks, Time frame of minimum 1 minute, date range, volume, etc.

CONCLUSION

Machine learning techniques such as LSTM, RNN, and linear regression have been applied to predict cryptocurrency prices. LSTM and RNN models are effective in capturing the sequential dependencies in cryptocurrency price data, allowing them to potentially make accurate predictions. These models consider historical price patterns, market sentiment, and other relevant factors to forecast future prices. Linear regression, on the other hand, assumes a linear relationship between the predictors and the target variable, making it simpler but potentially less accurate for cryptocurrency price prediction.

However, it is important to note that cryptocurrency markets are highly volatile and influenced by various external factors, including regulatory changes, market sentiment, and technological advancements. These factors can lead to sudden price fluctuations that are difficult to predict accurately using historical data alone. Therefore, while machine learning models can provide insights and potential trends, they should be used cautiously and in conjunction with other fundamental and technical analysis methods. Additionally, continuous monitoring and retraining of the models are necessary to adapt to evolving market conditions.

REFERENCES

[1] D. Shah and K. Zhang, ―Bayesian regression and Bitcoin,‖ in 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton), 2015, pp.

[2] Huisu Jang and Jaewook Lee, ―An Empirical Study on Modelling and Prediction of Bitcoin Prices with Bayesian
Neural Networks based on Blockchain Information,‖ in IEEE Early Access Articles, 2017, vol. 99, pp. 1-1.

[3] Bitcoin Volatility Forecasting with a Glimpse into Buy and Sell Orders‖ Tian Guo ; Albert Bifet ; Nino AntulovFantulin ; IEEE 2018.

[4] Predicting the Price of Bitcoin Using MachineLearning‖ Sean McNally ; Jason Roche ; Simon Caton; Ireland, Dublin , IEEE 2018.

[5] Project Based Learning: Predicting Bitcoin Prices using Deep Learning‖ S. Yogeshwaran ; PiyushMaheshwari; ManinderJeet Kaur ; Amity University Dubai Dubai, UAE; IEEE 2019.

[6] F. Andrade de Oliveira, L. Enrique ZÃ¡rate and M. de Azevedo Reis; C. NeriNobre, ―The use of artificial neural networks in the analysis and prediction of stock prices,‖ in IEEE International Conference .

[7] M. Daniela and A. BUTOI, ―Data mining on Romanian stock market using neural networks for price prediction‖. informatica Economica, 17,2013.

[8] Bitcoin Cost Prediction using Deep Neural Network Technique‖ Kalpanasonika ,Sayasri S , Vinothini , SugaPriya , IEEE 2018.

[9] An improved K-Means clustering algorithm‖ ;Juntao Wang, Xiaolong Su, IEEE 2017.

[10] Application of Random Forest Algorithm on Feature Subset Selection and Classification and Regression‖ ;Jitendra Kumar Jaiswal, Rita Samikannu IEEE 2017. .

[11] Improved Random Forest for Classification‖ ; Angshuman Paul, Dipti Prasad Mukherjee, Senior Member, IEEE, Prasun Das, AbhinandanGangopadhyay, Appa Rao Chintha and SaurabhKundu

[12] Ridge and Lasso Regression Models for CrossVersion Defect Prediction‖ Xiaoxing Yang and Wushao Wen IEEE 2018.

[13] Farokhmanesh, F., &Sadeghi, M. T. (2019). Deep Feature Selection using an Enhanced Sparse Group Lasso Algorithm. 2019 27th Iranian Conference on Electrical Engineering (ICEE) 2019.

[14] Robust Regression and Lasso‖ HuanXu ; Constantine Caramanis ; ShieMannor IEEE2018.

[15] D. Shah and K. Zhang, ―Bayesian regression and Bitcoin,‖ in 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton), 2015, pp. 409-415. [2]

[16] Huisu Jang and Jaewook Lee, ―An Empirical Study on Modelling and Prediction of Bitcoin Prices with Bayesian Neural Networks based on Blockchain Information,‖ in IEEE Early Access Articles, 2017, vol. 99, pp. 1-1. [3] F.

[17] Andrade de Oliveira, L. Enrique ZÃ¡rate and M. de Azevedo Reis; C. NeriNobre, ―The use of artificial neural networks in the analysis and prediction of stock prices,‖ in IEEE International Conference on Systems, Man, and Cybernetics, 2011, pp. 2151-2155.

[18] M. Daniela and A. BUTOI, ―Data mining on Romanian stock market using neural networks for price prediction‖. Informatica Economica, 17,2013.

[19] Paper on enhancing the price of bitcoin's fluctuation prediction using attentive LSTM and embedding network by Yang Li, Zibin Zheng and Hong-Ning Dai (2020) [2] published by applied journal as an open access article.

[20] Paper about the prediction of bitcoin price using ML algorithms by Shivam Pandey, Mr.Anil Chavan, [4] (2021), published by IRJET, in volume8-issue4.

BOOKS REFERED :-

1. Artificial Intelligence: A Modern Approach (Fourth edition, 2020) by Stuart Russell and Peter Norwig.

2. Understanding Machine Learning: From Theory to Algorithms c 2014 by Shai ShalevShwartz and Shai Ben-David.

WEBSITE VISITED :-

1. https://searchenterpriseai.techtarget.com

2. https://www.geeksforgeeks.org/