A Report on

"A Web Application on Forecasting the Bitcoin Prices of Next day using ML"

Course Title: Machine Learning and Pattern Recognition Sessional

Course Code: CSE 470

Department: Computer Science and Engineering

Level -4 semester –II

Submitted by:

Md Iftekhar Hossain Tushar

Student ID: 1902061

Level-4, Semester-II

Azizur Rahman Maruf

Student ID: 1902029

Level-4, Semester-II

Project Supervisor

Dr. Md. Delowar Hossain

Professor

Dept. of Computer Science and Engineering.

Hajee Mohammad Danesh Science & Technology University, Dinajpur

Department of Computer Science and Engineering

Hajee Mohammad Danesh Science & Technology University, Dinajpur-5200

CERTIFICATE

This is certify that Md Iftekhar Hossain Tushar, Azizur Rahman Maruf submit this project work entitled **"A Web Application on Forecasting the Bitcoin Prices of Next day using ML**" is carried out in partial fulfillment for the award of the degree of bachelor of science (engineering) in Computer Science and Engineering. This is a record of their own work carried out by them under of supervision and guidance.

Supervisor

Dr. Md. Delowar Hossain

Professor

Dept. of Computer Science and Engineering.

Hajee Mohammad Danesh Science & Technology University, Dinajpur

Department of Computer Science and Engineering
Faculty of Computer Science and Engineering
Hajee Mohammad Danesh Science and Technology University
Dinajpur-5200, Bangladesh

DECLARATION

The work entitled "**A Web Application on Forecasting the Bitcoin Prices of Next day using ML**" has been carried out in the Department of Computer Science and Engineering, Hajee Mohammad Danesh Science and Technology University is original and conforms to the regulations of this University.

We understand the University's policy on plagiarism and declare that no part of this thesis has been copied from other sources or been previously submitted elsewhere for the award of any degree or diploma.

………………………………
(Md Iftekhar Hossain Tushar)
Student ID: 1902061
Level-4, Semester-I

………………………………
(Azizur Rahman Maruf)
Student ID: 1902029
Level-4, Semester-I

Acknowledgment

We would like to express our thanks of gratitude to Dr. Md. Delowar Hossain, Professor, department of Computer Science and Engineering who gave us a golden opportunity to do this project and also provided support in completing in our project. His heartiest & kind Cooperation during our project work makes the dream real & we succeed to complete our project. While we were preparing this project file, various information that we found helped us in chapter of profile adding and we are glad that we were able to complete this project and understand many things. Through preparation of project was an immense learning experience and we included many personal qualities during this process like responsibility, punctuality, confidence and others. We would like to thank to our supervisor who supported us all the time, cleared our doubts and to our parents who also played a big role in finalization of our project file. We are taking this opportunity to acknowledge their support and we wish that they keep supporting us like this in the future. A project is a bridge between theoretical and practical learning and with this thinking we worked on the project and made it successful due to timely support and efforts of all who helped us. Once again, we would like to thank our classmates and friends also for their encouragement and help in designing and making our project creative. We are in debt of all these. Only because of them we were able to create our project and make it good and enjoyable experience.

Dedication

To my family, whose unwavering support and endless encouragement have been my anchor throughout this academic journey. To my friends, who provided both inspiration and moments of respite. And to all the mentors, teachers, and scholars whose guidance and wisdom illuminated the path to this achievement. This thesis is dedicated to you with heartfelt gratitude.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

Abstract

"**A Web Application on Forecasting the Bitcoin Prices of Next Day using ML**," integrates machine learning with a web platform to predict the next day's Bitcoin prices. Using historical data from Yahoo Finance (2014-2024), linear regression models are trained on four key indicators: "Close," "Open," "High," and "Low." The web application allows users to input these values and view the predicted prices, powered by pre-trained models. The project evaluates model accuracy using metrics such as MAE, MSE, RMSE, and r^2 score, demonstrating a practical approach to financial forecasting through machine learning.

Chapter 1

Introduction

1.1 Introduction

The rapid growth and volatility of cryptocurrency markets, particularly Bitcoin [1], have sparked significant interest in the development of predictive models for price forecasting. Accurate predictions can aid traders and investors in making informed decisions, mitigating risks, and maximizing profits. This project, titled "A Web Application on Forecasting the Bitcoin Prices of Next Day using Machine Learning," aims to develop a robust system for predicting the next day's Bitcoin prices using advanced machine learning (ML) techniques. The web-based application is designed to provide users with accurate predictions of Bitcoin's closing price by analyzing historical price data. Using a Ridge regression model, which has been trained on time series data, the system forecasts prices based on four key features: 'Close,' 'Open,' 'High,' and 'Low' prices. The project leverages various ML models [2] and techniques, including regression analysis, to achieve optimal predictive performance.

The application integrates an easy-to-use interface where users can input price data and obtain predictions instantly. The underlying architecture of the system is built on Python and leverages powerful libraries for machine learning, including scikit-learn and other essential packages for handling data. The goal of this project is to provide a practical tool for Bitcoin price forecasting, making use of machine learning models that can adapt to the inherent volatility of the cryptocurrency market. Through this application, users will be able to explore the potential of machine learning in forecasting financial assets, specifically Bitcoin, and understand the impact of various predictive models on price forecasting accuracy. This report details the methodology, models, evaluation metrics, and the implementation process of the forecasting system.

1.2 Cryptocurrency

Analyzing historical data on cryptocurrency, particularly focusing on Bitcoin, offers valuable insights into the dynamic evolution of the digital asset. Bitcoin, introduced in 2009 [3] by an anonymous entity known as Satoshi Nakamoto, marked the inception of a decentralized financial system. Historical Bitcoin data provides a chronological record of its price movements, market trends, and adoption milestones. Examining price charts over time reveals periods of volatility, bull runs, and corrections, shedding light on the cryptocurrency's market dynamics.

Notable events, such as regulatory developments, technological upgrades, or macroeconomic factors, are often reflected in historical data, showcasing the interconnected nature of Bitcoin with broader financial landscapes. Additionally, historical data aids in understanding Bitcoin's role as a store of value, digital gold, and a hedge against inflation, contributing to its narrative in the financial markets. Studying transaction data unveils patterns of on-chain activity, wallet movements, and the overall health of the Bitcoin network. This historical [4] perspective helps investors, analysts, and enthusiasts make informed decisions based on past performance and market trends. It also contributes to the ongoing discourse on Bitcoin's potential as a disruptive force in traditional finance. As the cryptocurrency landscape continues to evolve, historical data on Bitcoin remains a valuable resource, providing a foundation for predicting future trends, refining investment strategies, and fostering a deeper understanding of the broader implications of this innovative digital currency.

1.2.1 Historical Data as cryptocurrency

The historical data of cryptocurrencies serves as a valuable resource for understanding their evolution and market behavior over time. As the backbone of insightful analysis, historical data encompasses information such as price movements, trading volumes, and market capitalization. Studying the historical trajectory of individual cryptocurrencies and the broader market enables researchers, traders, and enthusiasts to identify patterns, trends, and potential correlations. This data is often visualized through charts and graphs, aiding in the interpretation of market dynamics. Moreover, historical data provides context for major events or developments within the cryptocurrency ecosystem, allowing stakeholders to analyze how external factors impact market performance. As the cryptocurrency market is known for its volatility, historical data becomes a critical tool [5] for risk assessment and decision-making. Researchers can glean insights into market cycles, identify potential support and resistance levels, and formulate strategies based on past performance. Overall, historical cryptocurrency data serves as a foundation for informed decision-making, strategic planning, and a deeper understanding of the complex and dynamic nature of the digital asset landscape.

1.3 Regression model on cryptocurrency

Regression models are widely used in financial markets to predict future prices or trends based on historical data. In the context of cryptocurrency, regression models analyze the relationship between various features such as past prices, trading volumes, and market sentiment to forecast future prices. These models help traders and investors understand potential price movements in highly volatile assets like Bitcoin, Ethereum, and Litecoin. One of the most common regression techniques applied in cryptocurrency forecasting is Linear Regression. Linear regression establishes a relationship between independent variables (e.g., 'Open,' 'High,' 'Low,' 'Volume') and the dependent variable (e.g., the 'Close' price). By fitting a straight line through the data points, the model minimizes the difference between predicted and actual prices, thereby providing a simple yet effective method for prediction. For example, in a linear regression model for Bitcoin price prediction, historical 'Close' prices could be regressed against the 'Open,' 'High,' and 'Low' prices from previous days. The model attempts to learn a linear relationship between these variables and the closing price, making it easier to predict future closing prices. Though simple, linear regression often serves as a baseline model for more complex techniques like Ridge regression or machine learning algorithms. Despite its simplicity, linear regression can capture basic price patterns and trends in cryptocurrency markets. However, given the high volatility of cryptocurrencies, more advanced models such as Ridge, Lasso, or ensemble methods like Random Forest and XGBoost are often used to improve prediction accuracy. These models account for non-linearity and complex relationships between market features, making them better suited for volatile environments like cryptocurrency markets.

In summary, regression models, especially linear regression, play a fundamental role in the early stages of cryptocurrency price forecasting, providing a foundation for more sophisticated techniques.

1.4 Aim and Objective of the thesis:

1.4.1 Aim

The aim of the project "A Web Application on Forecasting the Bitcoin Prices of Next Day using ML" is to develop a user-friendly web application that leverages machine learning models to accurately predict the next day's Bitcoin closing price based on historical data, enabling users to make informed trading decisions.

1.4.2 Objectives

The objectives of the project "A Web Application on Forecasting the Bitcoin Prices of Next Day using ML" is:

1. Develop a web application to predict the next day's Bitcoin prices using machine learning.
2. Train Linear Regression models on historical Bitcoin data for "Close," "Open," "High," and "Low" price datasets.
3. Allow users to input price values through a form and obtain real-time predictions via the web application.
4. Evaluate model performance using metrics like MAE, MAPE, SMAPE, MSE, RMSE, and R^2 score.

1.5 Contribution

Our contribution involves creating an integrated web application for predicting Bitcoin prices using machine learning. We collected and processed historical Bitcoin time series data from Yahoo Finance, covering "Close," "Open," "High," and "Low" prices. We applied Linear Regression models to these datasets and converted the models into pickle files for integration. The web application features two pages: one for inputting price data and another for displaying predictions. Our approach includes evaluating model performance with comprehensive metrics such as MAE, MAPE, SMAPE, MSE, RMSE, and R^2 score, providing users with accurate and actionable price forecasts.

1.6 Summary

This project has successfully developed a web application for Bitcoin price forecasting using machine learning. By analyzing historical data from Yahoo Finance with Linear Regression models, we created a system that predicts the next day's Bitcoin prices based on "Close," "Open," "High," and "Low" values. The user-friendly interface allows real-time predictions and decision-making support. Evaluating model performance with metrics like MAE, MAPE, SMAPE, MSE, RMSE, and R^2 score demonstrates the effectiveness of our approach. This work showcases the practical application of machine learning in financial forecasting and sets the stage for further improvements with advanced models.

Chapter 2

Literature Review

2.1 Introduction

The burgeoning interest in cryptocurrency markets, especially Bitcoin, has catalyzed extensive research into effective price prediction models using machine learning techniques. Traditional methods often rely on random data selection and basic correlation-based features, which can compromise accuracy. Recent advancements emphasize systematic data selection and the use of diverse machine learning models to enhance prediction precision. Studies have demonstrated that structured approaches and advanced techniques like Ridge Regression, LSTM, and ensemble methods outperform conventional models. These methods address the challenges posed by Bitcoin's volatility and nonlinearity, providing more reliable forecasts. This research explores the efficacy of various regression models, including Linear Regression, SVM, and gradient boosting techniques, in predicting Bitcoin prices. By leveraging precise time series data and evaluating performance through comprehensive metrics, this study aims to refine forecasting accuracy and contribute valuable insights into the evolving landscape of cryptocurrency prediction.

2.2 Related Works

Bitcoin, being a leading cryptocurrency, has attracted substantial research into price prediction using machine learning techniques. Traditional studies often involve feature selection based on correlation with Bitcoin prices and rely on randomly selected data chunks for training models. This randomness can sometimes compromise prediction accuracy. Recent research highlights the benefits of systematic data selection methods to improve model performance. For example, a study applying a structured approach to data selection demonstrated that training a linear regression model [6] with carefully chosen data chunks yielded a high accuracy rate of 96.97% in predicting Bitcoin prices over a 7-day period. This underscores the importance of methodical data handling in enhancing prediction reliability and suggests that refined data selection techniques can significantly boost model performance.

Forecasting cryptocurrency prices [7], including Bitcoin, Litecoin, and Ethereum, remains a challenging but crucial task. Recent research has utilized various models to improve prediction accuracy, such as Simple Linear Regression (SLR), Multiple Linear Regression (MLR), and advanced neural networks like Multilayer Perceptron (MLP) and Long Short-Term Memory (LSTM). Studies have shown that partitioning data into shorter sequences—representing different price regimes—enhances forecast precision.

metrics like Mean Absolute Percentage Error (MAPE) and relative Root Mean Square Error (relative RMSE) indicate that using multiple prior prices leads to better predictions. This approach contrasts with the random walk theory often applied to cryptocurrency prices and demonstrates improved performance over traditional benchmarks and recent forecasting studies.

Predicting Bitcoin prices remains challenging [8] due to its nonlinear, nonstationary behavior influenced by various uncontrollable factors and its inherent volatility. Traditional machine learning methods have struggled to deliver accurate results. Recent research emphasizes the limitations of conventional performance metrics like Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE) in capturing the effectiveness of prediction models. To address these issues, recent studies propose evaluating multiple regression models, including CatBoost, gradient boosting, extra trees, AdaBoost, K-nearest neighbors, and Theil-Sen regressors, using comprehensive performance metrics. These metrics include Coefficient of Determination (R^2), Mean Absolute Error (MAE), Mean Square Error (MSE), Root Mean Square Error (RMSE), Root Mean Squared Logarithmic Error (RMSLE), and MAPE. Results show the extra trees regressor had the highest MAE, RMSLE, and MAPE, while the gradient boosting regressor showed the highest MSE and RMSE. The Theil-Sen regressor achieved the highest R^2 value, highlighting its robustness in modeling Bitcoin price dynamics.

The rise of cryptocurrencies has drawn significant attention due to their accessibility and investment potential. Numerous studies have explored predicting Bitcoin's value using various factors like the total supply of cryptocurrencies, daily trading volume, and market demand. These factors heavily influence Bitcoin’s price volatility. In one such study [9], researchers employed time series analysis techniques, such as moving averages and ARIMA, alongside machine learning models like Support Vector Machines (SVM), linear regression, Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) networks. The primary goal was to compare the accuracy of these models in forecasting Bitcoin prices using a dataset collected daily over three years. By evaluating these models based on their accuracy, the study demonstrated that advanced machine learning models, particularly LSTM and GRU, often outperform traditional methods like ARIMA in handling cryptocurrency price volatility. This comparative analysis provides a solid foundation for further exploration in selecting the best model for Bitcoin price prediction.

Recent studies [10] have explored Bitcoin price prediction using machine learning techniques, particularly Linear Regression (LR) and Support Vector Machines (SVM). One study focused on time series data of daily Bitcoin closing prices from 2012 to 2018, evaluating various parameter combinations and kernel functions for SVM, including linear and polynomial kernels. Different filters with varying weight coefficients were tested across multiple window lengths to improve prediction accuracy. The study employed 10-fold cross-validation during the training phase to ensure model robustness and generalization across datasets. Performance metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Pearson Correlation were used for evaluation. Results indicated that the SVM model significantly outperformed the LR model, demonstrating superior accuracy in predicting Bitcoin prices. This highlights the efficacy of SVM in handling complex time series data for financial forecasting.

In recent years, Bitcoin has emerged as the most valuable cryptocurrency, yet its highly volatile prices present significant challenges for prediction. This study [11] focuses on identifying the most efficient and accurate machine learning models for forecasting Bitcoin prices. Utilizing 1-minute interval trading data from Bitstamp, spanning January 1, 2012, to January 8, 2018, the research employed various regression models implemented through scikit-learn and Keras libraries. The findings revealed that the most effective models achieved a remarkably low Mean Squared Error (MSE) of 0.00002 and a high R-Square (R^2) value of 99.2%. This highlights the potential of advanced regression techniques in capturing Bitcoin's price dynamics and underscores the importance of precise data intervals and robust algorithms in enhancing prediction accuracy.

2.3 Summary

The recent advancements in machine learning have significantly improved Bitcoin price prediction accuracy. By employing systematic data selection and sophisticated models like Ridge Regression, LSTM, and ensemble methods, researchers have enhanced forecasting precision and addressed the challenges of Bitcoin's volatility. The comparative analysis of various regression models, including SVM and gradient boosting, underscores their effectiveness in capturing complex price dynamics. These advancements provide a more reliable foundation for forecasting and highlight the importance of refined methodologies in cryptocurrency prediction.

Chapter 3

Background

3.1 Introduction

Linear regression is a fundamental statistical technique widely used to model relationships between variables and predict outcomes. In various fields, including finance and economics, it serves as a powerful tool for understanding how changes in one or more independent variables influence a dependent variable. This method involves finding the best-fitting line that minimizes prediction errors, thereby allowing for effective forecasting and decision-making. In evaluating the performance of such predictive models, measurement scores play a critical role. These metrics, including Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE), provide insights into the accuracy and reliability of the predictions. By assessing how well a model performs in predicting outcomes, these scores help in fine-tuning algorithms and choosing the most suitable methods for specific tasks. Understanding these metrics is crucial for optimizing model performance and ensuring reliable results in predictive analysis.

3.2 Machine Learning models:

3.2.1 Linear Regression

Linear regression is a statistical method used to model the relationship between a dependent variable and one or more independent variables. The basic idea [26] is to find the best-fitting line (or hyperplane in higher dimensions) that minimizes the sum of squared differences between the observed values and the values predicted by the model. This line is often referred to as the regression line.

The equation of a simple linear regression model with one independent variable can be expressed as:

$$Y = B_0 + B_1X + e$$

Where,

Y is a dependent variable, X is an independent variable, B_0 is the y-intercept, representing the predicted value of Y when X is zero, B_1 is the slope of the regression line, indicating the change in Y for a one-unit change in X. e is the error term, representing the unobserved factors that affect Y but are not included in the model.

The goal of linear regression is to estimate the values of B_0 and B_1 that minimize the sum of squared residuals (differences between observed and predicted values). This is typically achieved through the method of least squares.

3.3 Measurement Scores

Measurement scores are crucial metrics in evaluating the performance of machine learning algorithms, providing insights into their predictive accuracy and generalization capabilities. Mean Absolute Error (MAE) assesses the average absolute differences between predicted and actual values, offering a straightforward measure of model accuracy. Mean Squared Error (MSE) takes the average of squared differences, emphasizing larger errors and penalizing outliers more heavily. Root Mean Squared Error (RMSE) builds on MSE by taking the square root, providing a measure in the same unit as the target variable and offering a more interpretable metric. Mean Absolute Percentage Error (MAPE) gauges the average percentage difference between predicted and actual values, making it useful for assessing the relative accuracy of a model across different scales. The coefficient of determination (R-squared or D2) quantifies the proportion of the variance in the dependent variable that is predictable from the independent variables, indicating the goodness of fit of the model. Each measurement score caters to specific aspects of model performance, and their selection depends on the nature of the problem and the desired evaluation criteria. A comprehensive understanding of these metrics aids in fine-tuning models and selecting the most suitable algorithm for a given task.

3.4 Summary

linear regression is a vital statistical tool for modeling and predicting relationships between variables, offering valuable insights across various fields. The effectiveness of such models is rigorously evaluated using measurement scores like Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE), which assess the accuracy and reliability of predictions. These metrics are essential for optimizing model performance and making informed decisions. By understanding and applying these evaluation techniques, researchers and practitioners can refine their models to achieve more accurate and actionable results.

Chapter 4

Proposed System

4.1 Introduction

This project develops a web application designed to forecast Bitcoin prices using machine learning techniques. By harnessing data from Yahoo Finance, which covers Bitcoin price information from September 17, 2014, to July 31, 2024, the application aims to provide accurate predictions of key financial metrics, including “Close,” “Open,” “High,” and “Low” prices. The core of the project involves training Linear Regression models on these datasets, with each model specializing in predicting one of the price metrics. Once trained, these models are saved as pickle files, which are later utilized by the web application to generate forecasts. The web application itself is crafted to be user-friendly, allowing users to input price values and receive predictions in a clear and accessible format. The backend, powered by a Python script named app.py, integrates these models with the web interface, managing the flow of data and ensuring that predictions are accurately generated and displayed. Performance is rigorously evaluated using metrics like Mean Absolute Error (MAE) and R^2 score to ensure the reliability of the forecasts. The application is deployed on a web server, making it readily accessible to users for efficient and interactive Bitcoin price forecasting.

4.2 Working environment

To set up a working environment for your project, you'll need to consider both the software and hardware requirements. Here's a comprehensive list of what you'll need:

4.2.1 Hardware Requirements

1. **Computer**:
    - **CPU**: Modern multi-core processor (e.g., Intel i5/i7 or AMD Ryzen 5/7)
    - **RAM**: Minimum of 8 GB (16 GB recommended for larger datasets)
    - **Storage**: SSD with at least 256 GB of available space (for faster performance and ample space for data and models)
    - **GPU** (optional but recommended for deep learning models): NVIDIA GPU with CUDA support (e.g., GTX 1060 or higher) if using advanced ML models like MLP or LSTM
2. **Internet Connection**:
    - A stable internet connection for downloading datasets, libraries, and updates.

4.2.2 Software Requirements

1. **Operating System**:
   - Windows, macOS, or Linux (ensure compatibility with the required software)
2. **Python Environment**:
   - **Python**: Version 3.7 or higher
   - **Package Manager**: pip or conda (for managing Python packages)
3. **Python Libraries**:
   - **Data Manipulation**: pandas
   - **Numerical Computing**: numpy
   - **Machine Learning**: scikit-learn, statsmodels, keras, xgboost (for additional models)
   - **Web Framework**: Flask (for app.py)
   - **Model Serialization**: joblib or pickle (for saving and loading models)
   - **Visualization**: matplotlib, seaborn (optional, for data visualization)
   - **Others**: requests, beautifulsoup4 (if scraping data or handling HTTP requests)
4. **Development Tools**:
   - **IDE/Text Editor**: VSCode, PyCharm, or Jupyter Notebook (for developing and testing code)
   - **Version Control**: Git (optional, for version control and collaboration)
5. **Web Browser**:
   - Modern browser (e.g., Google Chrome, Mozilla Firefox) to visualize and interact with the web pages.
6. **Database** (optional, if needed for storing results):
   - SQLite or another lightweight database if you plan to store historical predictions or user inputs.

4.3 Dataset and preprocessing

Bitcoin time series data was meticulously gathered from Yahoo Finance, covering a substantial period from September 17, 2014, to July 31, 2024. This dataset was carefully chosen to capture a broad range of market conditions and trends over nearly a decade, providing a robust foundation for accurate price forecasting. The collected data comprises four critical financial indicators: "Close," "Open," "High," and "Low" prices. Each of these indicators was treated as a distinct dataset, with Date Time indexing to preserve the temporal order of the data.

Table 1. Features Description

Feature	**Description**	**Unit**
Open	Opening price of cryptocurrency for particular date	USD
Close	Closing price of cryptocurrency in particular date	USD
High	Highest price of cryptocurrency sold in particular date	USD
Low	Lowest price of cryptocurrency sold in particular date	USD
Adjusted close	The adjusted closing price amends a stock’s closing price to reflect that stock’s value after accounting for any corporate actions	USD
Volume	Volume indicates how many cryptocurrencies are being bought and sold on specific exchanges	USD

This approach allows for detailed analysis and prediction of each price category independently, ensuring that the models trained on this data can effectively capture the unique patterns and fluctuations associated with each type of price data. The "Close" price represents the final transaction price for each day, reflecting the overall market sentiment. The "Open" price denotes the initial transaction price, providing insight into the market's opening conditions. The "High" and "Low" prices indicate the highest and lowest prices reached during the trading day, respectively, highlighting volatility and price range.

Table 2. Dataset specification

Variable Name	Variable Description	Data Type
Date	Date of Observation	Date
Open	Opening price on the given day	Number
High	High price on the given day	Number
Low	Low price on the given day	Number
Close	Close price on the given day	Number

By segmenting the data in this manner, the project aims to leverage linear regression models to forecast future prices accurately, based on historical trends and patterns observed in these key financial metrics. This structured approach ensures that the models are well-calibrated to predict the next day's Bitcoin prices with a high degree of precision.

4.4 Proposed approach

The project focuses on developing an intuitive web application that leverages machine learning for predicting Bitcoin prices. The project is divided into two main stages: data preparation and model training, and web application development.

4.4.1 Data Preparation and Model Training:

The project involves collecting a comprehensive dataset of Bitcoin price time series from Yahoo Finance. This dataset spans from September 17, 2014, to July 31, 2024, providing a robust historical record for analysis. The dataset is organized into four distinct columns— "Close," "Open," "High," and "Low"—each representing a different aspect of Bitcoin's market activity. To manage these data points effectively, each column is treated as a separate dataset with DateTime indexing, which facilitates detailed and organized time series analysis.

Table 3. Train test Dataset

Data Splitting	**Time span**
Train	September 17, 2014 to August 10, 2022
Test	August 11, 2022 to July 31, 2024

To forecast Bitcoin prices, Linear Regression, a widely-used and straightforward machine learning model, is applied to each of these four datasets. Linear Regression is chosen for its simplicity and effectiveness in identifying trends and making predictions based on historical data. After training the models on these datasets, each trained model is saved as a pickle file. This results in four separate pickle files, each dedicated to predicting one of the price metrics: closing price, opening price, highest price, and lowest price.

4.4.2 Model Performance Evaluation:

To ensure the accuracy and effectiveness of the Linear Regression models, several performance metrics are utilized. These metrics include:

- **Elapsed Time:** Measures the time taken to train and make predictions with the model.
- **Mean Absolute Error (MAE):** Indicates the average magnitude of errors in predictions, providing a straightforward measure of model accuracy.
- **Mean Absolute Percentage Error (MAPE):** Reflects the accuracy of the predictions in percentage terms, making it easier to interpret the model's performance.
- **Symmetric Mean Absolute Percentage Error (SMAPE):** A variation of MAPE that accounts for both under- and over-predictions, offering a balanced view of accuracy.

- **Mean Squared Error (MSE):** Calculates the average of the squares of the errors, giving more weight to larger errors.
- **Root Mean Squared Error (RMSE):** Provides the square root of MSE, offering a more interpretable measure of prediction error.
- **Mean Pairwise Distance:** Measures the average distance between predicted and actual values.
- **R^2 Score:** Indicates the proportion of variance in the dependent variable that is predictable from the independent variables, assessing the goodness of fit of the model.

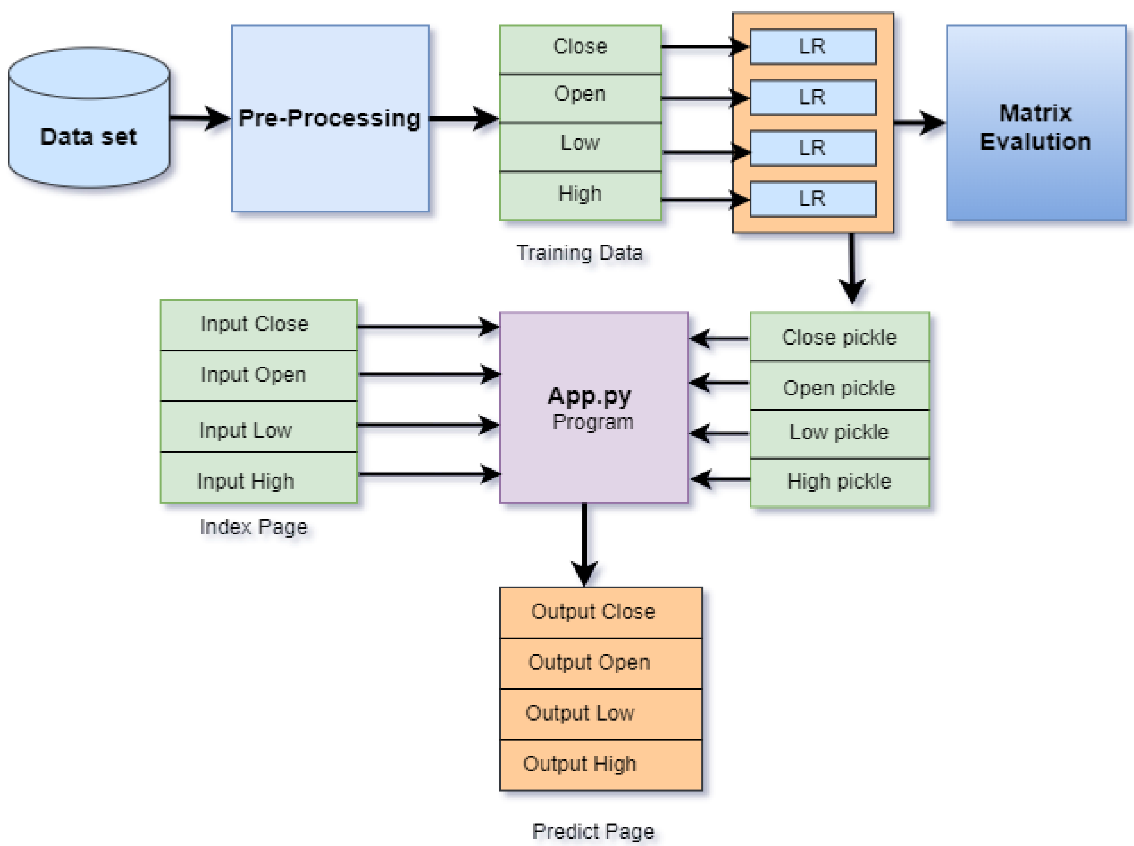

Figure 1: Proposed System

4.4.3 Web Application Development:

The web application component of the project is designed to offer a user-friendly interface for interacting with the Bitcoin price forecasting models. It serves as a bridge between users and the machine learning models trained on the Bitcoin price data.

The development of this web application involves creating two distinct pages and integrating them through backend Python code.

1. **Index Page:** The first page of the web application is a data entry form. This page is designed to collect user inputs for the four key Bitcoin price metrics: “Close,” “Open,” “High,” and “Low.” The form includes input fields where users can enter these values manually. The inputs provided by users are critical as they serve as the basis for generating forecasts using the trained models.
    - **User Interface (UI):** The form page is crafted to be intuitive and easy to navigate. It typically features text boxes or dropdown menus for users to input or select values for each of the four price metrics. Proper labels and instructions are included to guide users through the data entry process.
    - **Data Validation:** To ensure the accuracy and integrity of the data submitted, validation checks are implemented. These checks confirm that the input values are in the correct format and fall within expected ranges, reducing the likelihood of errors or invalid data affecting the predictions.
2. **Predict Page:** The second page of the web application is dedicated to displaying the prediction results based on the user inputs from the form page. After the user submits their data, the application processes this information and generates forecasts for each of the price metrics.
    - **Display of Results:** This page is designed to clearly present the predicted values for "Close," "Open," "High," and "Low" prices. The results are typically displayed in a structured format, such as a table or a set of labels, making it easy for users to interpret the predictions.
    - **Visualization (Optional):** Depending on the requirements, additional visualizations, such as graphs or charts, may be included to provide a visual representation of the forecasted values compared to historical data trends.

3. **Backend Integration with app.py:** At the core of the web application's functionality is the Python script app.py, which acts as the backend logic that integrates the front-end web pages with the trained machine learning models.
    - **Model Loading:** When the web application is accessed, app.py handles the loading of the trained pickle files that contain the Linear Regression models for each of the price metrics. This step is crucial as it ensures that the models are available for making predictions.
    - **Prediction Generation:** Upon receiving user inputs from the form page, app.py processes this data and utilizes the loaded models to generate predictions. For each input value, the corresponding model is used to forecast the Bitcoin price for that metric.
    - **Data Handling:** The backend script manages the flow of data between the user interface and the models. It captures user inputs, passes them to the appropriate model, retrieves the predictions, and then sends this information to the results page for display.
    - **Error Handling and Feedback:** app.py also includes error handling mechanisms to manage any issues that arise during data processing or model prediction. It provides feedback to users if there are errors in their input or if the models encounter any problems.
4. **Deployment and Hosting:**
    - **Web Server Setup:** The final step in web application development involves deploying the application to a web server, which makes it accessible to users via a web browser. This process includes configuring the server environment, ensuring that all dependencies are properly installed, and making sure the application is secure and scalable.
    - **User Access:** Once deployed, users can access the application through a web address. They can interact with the form to input data and view the prediction results on the results page.

Overall, the web application provides a seamless and interactive experience for users, allowing them to forecast Bitcoin prices using the trained models with ease. The integration of user inputs, model predictions, and result displays ensures that the application is both functional and user-friendly.

4.5 Summary

The project successfully develops a web application that forecasts Bitcoin prices using machine learning. By training Linear Regression models on comprehensive Bitcoin price data from Yahoo Finance, the application provides accurate predictions for key metrics: “Close,” “Open,” “High,” and “Low.” The user-friendly web interface, integrated with a Python backend script (app.py), enables users to input values and view forecasts easily. Performance metrics like MAE and R^2 score ensure the reliability of predictions, while deployment on a web server ensures accessibility.

Chapter 5

Experimental Result

5.1 Introduction

the experimental results of our linear regression models designed for Bitcoin price forecasting. We developed and trained four distinct models, each focusing on a specific market indicator: Close, Open, High, and Low prices. The objective was to evaluate and compare the performance of these models in predicting Bitcoin's future prices using various performance metrics. To assess model efficacy, we measured several indicators including Elapsed Time, Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Symmetric Mean Absolute Percentage Error (SMAPE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Pairwise Distance (MPD), and R^2 score. These metrics provided a detailed evaluation of the models' accuracy and effectiveness in capturing the nuances of Bitcoin price movements.

Additionally, we created a web application to facilitate user interaction with the models. This application, built using Flask, includes an index page where users can input the Close, Open, High, and Low prices, and a predict page where the forecasts for the next day are displayed. The app.py file acts as the central integrator, managing the interaction between the user interface and the trained models. By running the application from the command prompt, users can access the form, submit data, and view the results in real time. This setup ensures a practical and accessible way to leverage our forecasting models.

5.2 Training Linear Regression Model

While training a Machine Learning model we have used Linear Regression model. We have trained the model four times by fitting the segmented distinct training datasets— "Close," "Open," "High," and "Low"—each representing a different aspect of Bitcoin's market activity. We have evaluated the results measuring Elapsed time, MAE, MAPE, SMAPE, MSE, RMSE, MPD, R2 on the test datasets. Table 3 shows the results for Linear regression model.

Table 3. Measurement Scores

Measurement Scales	Close	High	Low	Open
Elapsed time	0.002989	0.002997	0.002687	0.018995
MAE	653.8944	556.6622	589.2676	656.0244
MAPE	1.726743	1.497406	1.581253	1.73588
SMAPE	1.72981	1.50332	1.57989	1.73889
MSE	1149013	836404.4	952702.5	1149550
RMSE	1071.92	914.5515	976.0648	1072.171
MPD	0.90945	0.77422	0.81957	0.91241
R2	0.996164	0.99732	0.996645	0.996148

We also compared the forecasted results in graphical representation for datasets— "Close," "Open," "High," and "Low"—to visualize the linear models performance on each dataset.

(a)

(b)

(c)

(d)

Figure 2: Plotting models performance graphically. (a) Actual vs Predicted Close price, (b) Actual vs Predicted Open price, (c) Actual vs Predicted High price, (d) Actual vs Predicted Low price

At the end of this training process we converted the trained models into pickle file. So we have got pickle files for four different training dataset on linear regression model.

5.3 App.py

We can say the App.py file is the communicator or integrator of two web page named as index page and predict page. App.py file gets the inputs from the form of index page and finds four predicted values of next day using the four pickle files of trained models, and passes all values to the predict web page.

To perform the web pages synchronously, it is mandatory to run the App.py file from the command prompt or terminal. It gives the service of flask app and activates the debugging mode. The terminal shows the hyperlink on which the synchronized index page is running. The figure shows the result of command prompt.

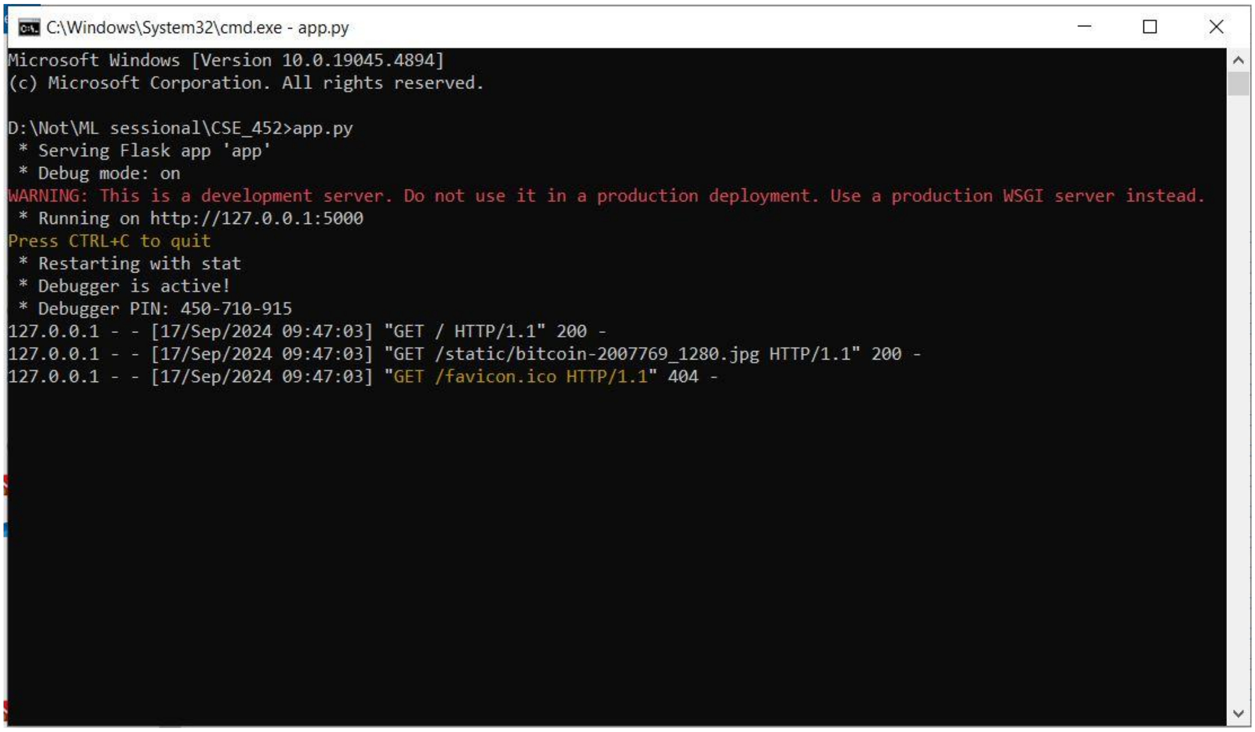

Figure 3: Running the App.py on command prompt.

5.4 Index Page

We copied the hyperlink from the command prompt and browse with it in a browser. It gives the index web page on the browser which contains mainly a form to give input of Close, Open, High, Low prices of three previous values of predicting day and there is a predict button which submits the values to App.py to find the next day predicted values.

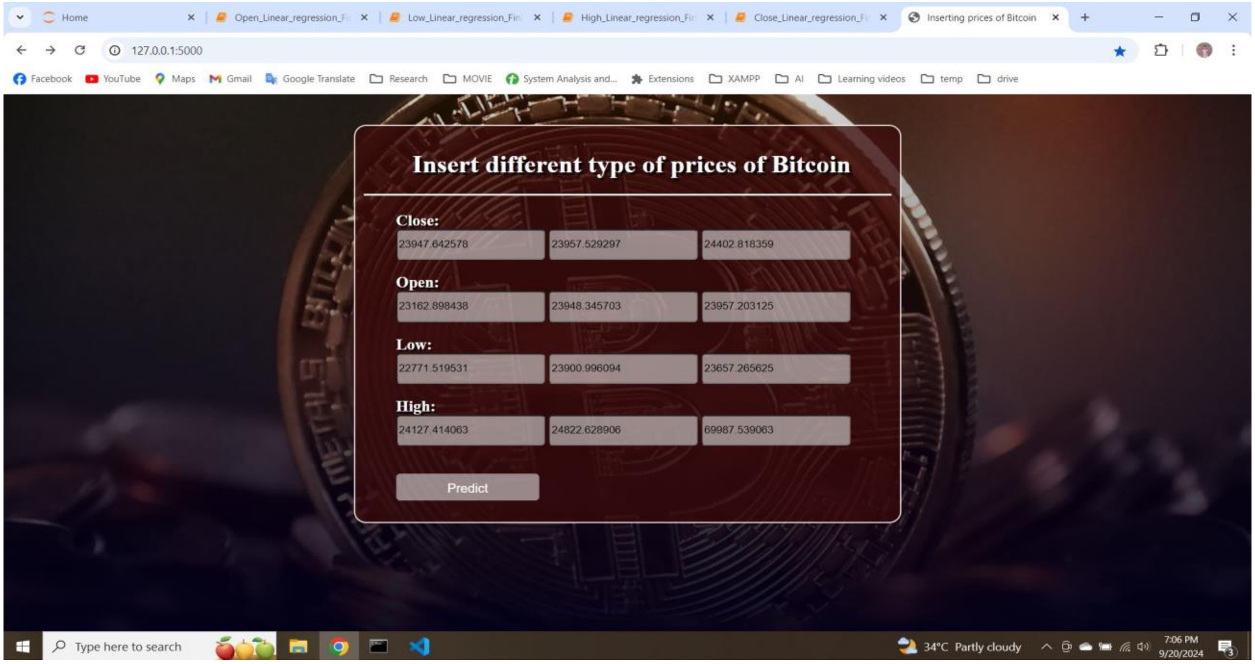

Figure 4: Index Page

5.5 Predict Page

The predict page show the results for the following day of Close, Open, High, Low.

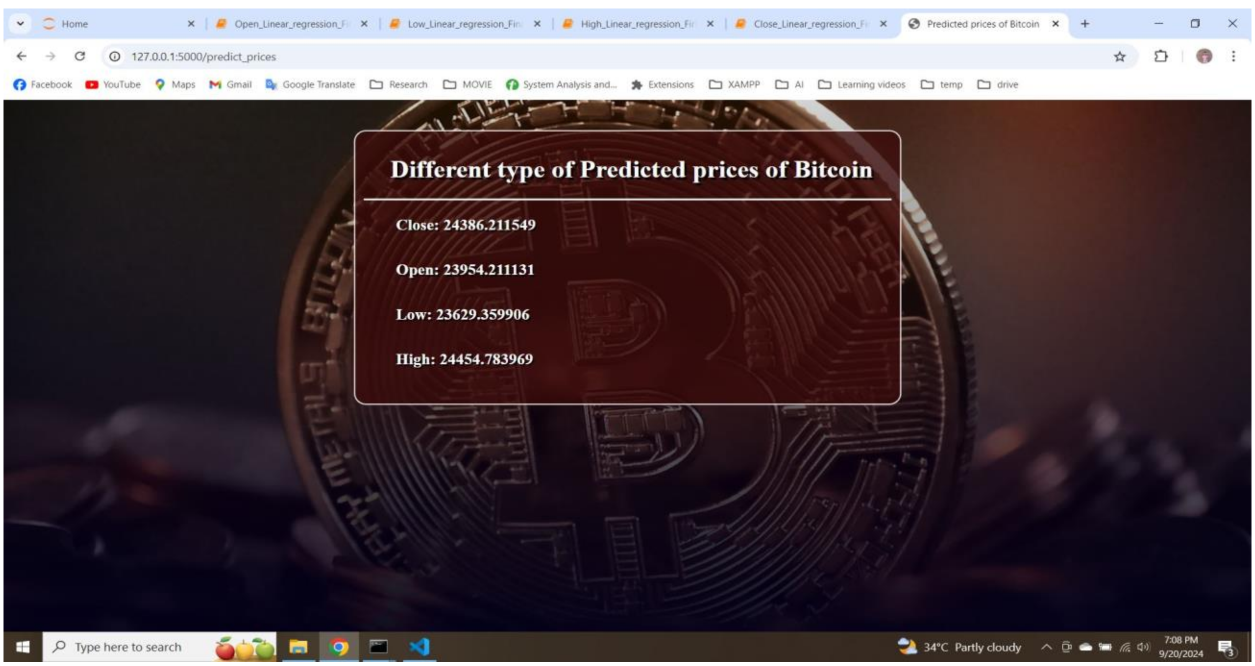

Figure 4: Predict Page

5.6 Summary

We examined the performance of our linear regression models developed for Bitcoin price forecasting. By focusing on distinct market indicators—Close, Open, High, and Low prices—we aimed to evaluate and compare the effectiveness of these models in predicting future Bitcoin prices. The use of multiple performance metrics, including Elapsed Time, Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Symmetric Mean Absolute Percentage Error (SMAPE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Pairwise Distance (MPD), and R^2 score, provided a comprehensive assessment of each model's accuracy and capability in capturing price trends.

The results from these evaluations demonstrated the models' proficiency in forecasting Bitcoin prices with varying degrees of accuracy. To enhance usability, we developed a web application using Flask, which allows users to interact with the models effectively. The application features an index page for data input and a predict page for displaying forecasted values. The app.py file ensures seamless integration between the user interface and the trained models, facilitating real-time data submission and result retrieval.

Chapter 6

Conclusion

And

Future work

6.1 Conclusion

The application leverages linear regression to predict Bitcoin prices for the following day, utilizing historical price data. By training models on datasets including 'Close,' 'Open,' 'High,' and 'Low' prices, the approach effectively forecasts future trends. Performance metrics such as MAE, MAPE, and RMSE are used to evaluate accuracy. The integration of these models into a web application facilitates user interaction by allowing input of price data and displaying predictions. Overall, this method provides a practical tool for anticipating Bitcoin price movements, demonstrating the utility of linear regression in financial forecasting.

6.2 Future work

In future we will find the best model that can perform more accurately then linear model in every fluctuation of time series.

References:

[1] Iqbal, M., Iqbal, M., Jaskani, F., Iqbal, K., & Hassan, A. (2021). Time-series prediction of cryptocurrency market using machine learning techniques. EAI Endorsed Transactions on Creative Technologies, 8(28).

[2] Yamak, P. T., Yujian, L., & Gadosey, P. K. (2019, December). A comparison between arima, lstm, and gru for time series forecasting. In Proceedings of the 2019 2nd international conference on algorithms, computing and artificial intelligence (pp. 49-55).

[3] Oyewola, D. O., Dada, E. G., & Ndunagu, J. N. (2022). A novel hybrid walk-forward ensemble optimization for time series cryptocurrency prediction. Heliyon, 8(11).

[4] Chowdhury, R., Rahman, M. A., Rahman, M. S., & Mahdy, M. R. C. (2020). An approach to predict and forecast the price of constituents and index of cryptocurrency using machine learning. Physica A: Statistical Mechanics and its Applications, 551, 124569.

[5] Indulkar, Y. (2021, March). Time series analysis of cryptocurrencies using deep learning & fbprophet. In 2021 International Conference on Emerging Smart Computing and Informatics (ESCI) (pp. 306-311). IEEE.

[6] Ali, M., & Shatabda, S. (2020, November). A data selection methodology to train linear regression model to predict bitcoin price. In 2020 2nd International Conference on Advanced Information and Communication Technology (ICAICT) (pp. 330-335). IEEE.

[7] Uras, N., Marchesi, L., Marchesi, M., & Tonelli, R. (2020). Forecasting Bitcoin closing price series using linear regression and neural networks models. PeerJ Computer Science, 6, e279.

[8] Saheed, Y. K., Ayobami, R. M., & Orje-Ishegh, T. (2022). A comparative study of regression analysis for modelling and prediction of bitcoin price. In Blockchain Applications in the Smart Era (pp. 187-209). Cham: Springer International Publishing.

[9] Gupta, A., & Nain, H. (2021). Bitcoin price prediction using time series analysis and machine learning techniques. In Machine Learning for Predictive Analysis: Proceedings of ICTIS 2020 (pp. 551-560). Springer Singapore.

[10] Karasu, S., Altan, A., Saraç, Z., & Hacioğlu, R. (2018, May). Prediction of Bitcoin prices with machine learning methods using time series data. In 2018 26th signal processing and communications applications conference (SIU) (pp. 1-4). IEEE.

[11] Phaladisailoed, T., & Numnonda, T. (2018, July). Machine learning models comparison for bitcoin price prediction. In 2018 10th international conference on information technology and electrical engineering (ICITEE) (pp. 506-511). IEEE.