AN INDUSTRY ORIENTED MINI PROJECT REPORT ON

Bitcoin price prediction using machine learning: An approach to sample dimension engineering

in the partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

in

CSE (Data Science)

Submitted by

V KAVYA	**21B81A6780**
SANIA	**21B81A67B3**
VAISHNAVI CH	**21B81A67C2**

Under the guidance of
Dr. R. Raja
Associate Professor

DEPARTMENT OF CSE (Data Science)

CVR COLLEGE OF ENGINEERING

(*An Autonomous institution, NAAC Accredited and Affiliated to JNTUH, Hyderabad*)

Vastunagar, Mangalpalli (V), Ibrahimpatnam (M),
Rangareddy (D), Telangana- 501 510

NOVEMBER 2024

CVR COLLEGE OF ENGINEERING

(*An Autonomous institution, NAAC Accredited and Affiliated to JNTUH, Hyderabad*)
Vastunagar, Mangalpalli (V), Ibrahimpatnam (M),
Rangareddy (D), Telangana- 501 510

DEPARTMENT OF CSE (Data Science)

CERTIFICATE

This is to certify that the Industry Oriented Mini Project report entitled **"Bitcoin price prediction using machine learning: An approach to sample dimension engineering"** bonafide record of work carried out by **V KAVYA (21B81A6780), SANIA (21B81A67B3) and CH VAISHNAVI(21B81A67C2)** submitted to **Dr. R. Raja, Associate Professor** for the requirement of the award of **Bachelor of Technology** in **CSE (Data Science)** to the CVR College of Engineering, affiliated to Jawaharlal Nehru Technological University, Hyderabad during the year 2024-2025.

Project Guide	**Project Coordinator**
Dr. R. Raja	**Dr. R. Raja**
Associate Professor	Associate Professor
Department of CSE(CS)	Department of CSE(CS)
Head of the Department	**External Examiner**

CVR COLLEGE OF ENGINEERING

(UGC Autonomous Institution)
Affiliated to JNTU Hyderabad

Vastunagar, Mangalpalli (V), Ibrahimpatnam (M),

Ranga Reddy (Dist.), Hyderabad – 501510, Telangana State

DECLARATION

We hereby declare that the Industry Oriented Mini Project report entitled **“Bitcoin price prediction using machine learning: An approach to sample dimension engineering”** is an original work done and submitted to CSE (Data Science) Department, CVR College of Engineering, affiliated to Jawaharlal Nehru Technological University Hyderabad in partial fulfilment for the requirement of the award of Bachelor of Technology in CSE (Data Science) and it is a record of bonafide project work carried out by us under the guidance of **Dr. R. Raja, Associate Professor**, Department of CSE (Cyber Secuity).

We further declare that the work reported in this project has not been submitted, either in part or in full, for the award of any other degree in this Institute or any other Institute or University.

Signature of the Student

V KAVYA

Signature of the Student

SANIA

Signature of the Student

VAISHNAVI CH

Date:

Place:

ACKNOWLEDGEMENT

We are thankful for and fortunate enough to get constant encouragement, support, and guidance from all **Teaching staff of CSE (Data Science) Department** which helped us in successfully completing this Industry Oriented Mini Project.

We are extremely thankful to our internal guide **Mr. K. Harish Kumar,** Assistant Professor, Department of CSE (Data Science) for providing support and guidance, which made us to complete the Industry Oriented Mini Project on time.

We thank **Dr. R. Raja**, Project Coordinator and **Dr. P.A. Abdul Saleem**, **Dr. LNC Prakash K, Dr. R. Srilakshmi,** Project Review Committee members for their valuable guidance and support which helped us to complete the Industry Oriented Mini Project Work successfully.

We thank Professor and Head of the Department **Dr**. **S. V. Suryanarayana,** for giving us all the support and guidance, which made us to complete the Industry Oriented Mini Project duly.

We would like to express heartful thanks to **Dr. H. N. Lakshmi,** Associate Dean, ET Department for providing us an opportunity and extending support and guidance.

We thank our Vice-Principal **Prof. L. C. Siva Reddy** for providing excellent computing facilities and a disciplined atmosphere for doing our work.

We wish a deep sense of gratitude and heartfelt thanks to our Principal **Dr. Rama Mohan Reddy** and the **Management** for providing excellent lab facilities and tools.

TABLE OF CONTENTS

LIST OF FIGURES

SYMBOLS

t	Time index
vvv	Feature index
ggg	Granularity of Bitcoin price data
ccc	Current time
hhh	Prediction length
XgX_gXg	Matrix of training samples under granularity ggg
YgY_gYg	Vector of labels under granularity ggg
yyy	Label value
LgL_gLg	Prediction error for granularity ggg
ftf_tft	Forget gate in LSTM
iti_tit	Input gate in LSTM
oto_tot	Output gate in LSTM
CtC_tCt	Memory cell state in LSTM
www	Weight vector
bbb	Bias term in Support Vector Machine

ABBREVIATIONS

AI	Artificial Intelligence
ARIMA	Autoregressive Integrated Moving Average
CPU	Central Processing Unit
GBDT	Gradient Boosted Decision Trees
GPU	Graphics Processing Unit
IDE	Integrated Development Environment
LDA	Linear Discriminant Analysis
LR	Logistic Regression
LSTM	Long Short-Term Memory
NLP	Natural Language Processing
QDA	Quadratic Discriminant Analysis
RAM	Random Access Memory
RNN	Recurrent Neural Network
SVM	Support Vector Machine
VC	Vapnik–Chervonenkis (dimension theory)

ABSTRACT

It explores machine learning methods to predict Bitcoin prices at different intervals. Given Bitcoin's volatility and the increasing view of it as an investment asset, the paper seeks accurate price prediction methods to aid investors. The study uses two data granularities: daily and 5-minute intervals.

It presents a novel approach to Bitcoin price prediction by utilizing machine learning techniques with an emphasis on sample dimension engineering. Recognizing Bitcoin's unique market volatility and popularity as a speculative asset, the authors aim to improve predictive accuracy by adapting model complexity to data frequency and structure. The study introduces high-dimensional features, including network metrics, trading volume, investor attention indicators, and gold prices, for daily Bitcoin price prediction. For high-frequency predictions at 5-minute intervals, the authors employ fewer, essential trading features. Leveraging the principle of Occam's Razor, they use simpler statistical models for daily price prediction to prevent overfitting, while reserving complex machine learning algorithms like Random Forest and LSTM for high-frequency interval predictions.

The results indicate that simple statistical methods outperform machine learning models in daily predictions, achieving an accuracy of 66%, while machine learning models show superior performance for high-frequency predictions, with accuracy reaching 67.2%. This dual-model strategy demonstrates the significance of sample dimension in Bitcoin price prediction, providing a scalable framework for other asset classes with similar high volatility. The study contributes to the field by illustrating how machine learning models can be tailored to different frequency-based datasets, thereby enhancing prediction reliability. Future work is suggested to explore further feature sets and refine models for greater predictive power across various financial and asset markets.

CHAPTER 1
INTRODUCTION

1.1 MOTIVATION

The motivation for this project centers on the unique challenges posed by Bitcoin's volatility and the increased interest in cryptocurrency as a speculative investment. Traditional financial models struggle with predicting Bitcoin's price fluctuations due to its rapid, erratic shifts and high dependency on external factors like media attention and investor sentiment. The authors emphasize the need for improved predictive models, as investors and policymakers alike seek reliable forecasts to make informed decisions in the crypto market.

In response to this challenge, it explores the potential of machine learning to predict Bitcoin prices by addressing the limitations of previous models, which often overlook the importance of tailoring prediction techniques to specific data structures and sampling frequencies. By testing various machine learning models on high-dimensional daily data and lower-dimensional high-frequency data, the researchers aim to improve predictive accuracy while addressing the complexities of Bitcoin's price dynamics. This work is intended to serve as a foundation for applying machine learning to similar predictive tasks in other volatile markets.

1.2 PROBLEM STATEMENT

The problem statement addresses the need for accurate prediction of Bitcoin prices using machine learning techniques. Existing models tend to focus on improving prediction accuracy without considering the specific structure of Bitcoin price data, which varies significantly by frequency (e.g., daily versus high-frequency, minute-level data). This oversight can lead to issues such as overfitting or underfitting when complex models are applied indiscriminately across different data granularities.

It defines two specific predictive tasks-Daily Price Prediction (GBP-D), High-Frequency Price Prediction (GBP-M). For both tasks, the goal is to apply sample-dimension engineering to ensure that the machine learning models used are appropriately aligned with the data structure, thus improving predictive accuracy and model reliability across varying time intervals.

1.3 PROJECT OBJECTIVES

The objectives of this project are designed to advance predictive accuracy in Bitcoin price forecasting through carefully tailored machine learning approaches. By developing and evaluating models specifically for Bitcoin's daily aggregated prices and high-frequency interval prices, the study aims to improve reliability in forecasting within the volatile cryptocurrency market. Key to this approach is sample-dimension engineering, which strategically aligns features and models to the unique frequency and complexity of each dataset, ensuring both efficiency and accuracy. The study further aims to compare traditional statistical models with more complex machine learning algorithms, assessing which models perform best across different data structures. These findings not only contribute to a more robust framework for investors and analysts seeking to make informed decisions but also lay a foundation for future research in predictive modeling for volatile assets. Ultimately, the project's goals are rooted in providing actionable insights for financial decision-making while expanding the practical applications of machine learning in financial markets.

1.4 PROJECT REPORT ORGANIZATION

The organization of the project report is well-structured, beginning with an **introduction** that outlines Bitcoin's rise as a volatile, speculative asset and the necessity for accurate price prediction methods. The authors justify the research by highlighting gaps in existing studies, specifically the lack of focus on sample dimensions in machine learning models for Bitcoin price prediction. This section sets up the foundation for their methodology, noting the distinctive approach of using high-dimensional features, and dividing prediction tasks into daily and 5-minute intervals to reflect different data frequencies and structures.

The **methodology** section presents a clear breakdown of the problem statement and solution framework. The authors discuss dimension engineering by categorizing prediction data based on frequency: high-dimension features are used for daily prices, and fewer features are selected for high-frequency, 5-minute interval prices. Their choice of machine learning and statistical methods reflects an alignment with Occam's Razor, aiming for simpler models when possible to minimize overfitting. They further elaborate on feature selection, including relevant market and network indicators like hash rate, Google Trends search volume, and gold spot price, integrating sentiment and traditional financial measures.

Finally, **implementation and results** are presented with detailed descriptions of machine learning algorithms and evaluation metrics used to assess model performance across the two frequency-based categories. Results show that simple statistical methods are effective for daily price predictions, while complex machine learning models perform better for high-frequency price predictions, demonstrating the significance of sample dimension engineering. The paper concludes with a discussion on the broader applicability of their approach, suggesting future work to expand feature sets and refine predictive models.

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING WORK

[1] S. McNally, J. Roche, S. Caton, Predicting the price of bitcoin using machine learning, in: 2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing, PDP, IEEE, 2018.

S. McNally investigates the use of machine learning models to predict Bitcoin's price, aiming to address the challenges of forecasting in the highly volatile cryptocurrency market. Given Bitcoin's susceptibility to rapid price fluctuations, traditional financial models often struggle to make accurate predictions. Machine learning offers a promising alternative by analyzing large datasets to detect patterns that may not be visible through conventional approaches. The study evaluates several machine learning techniques to determine their predictive power and accuracy in forecasting Bitcoin prices.

Key Features

1. **Focus on Cryptocurrency Volatility:** Directly addresses the volatility inherent in Bitcoin pricing, a key challenge in financial modeling.
2. **Comparative Model Analysis:** Compares several machine learning models (e.g., recurrent neural networks, support vector machines, and random forests) to find the best predictor for Bitcoin prices.
3. **Useof Time-Series Data:** Employs time-series analysis techniques that account for the sequential nature of cryptocurrency price data, making the predictions more robust.
4. **High-Frequency Data Utilization:** Incorporates high-frequency data, such as hourly or minute-based pricing, to capture short-term trends in the cryptocurrency market.
5. **Feature Extraction and Selection:** Carefully selects features, including historical price, trading volume, and market sentiment indicators, to improve model accuracy.

6. **Performance Metrics for Evaluation:** Uses metrics like Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) to evaluate model accuracy, providing a clear view of each model's effectiveness.
7. **Potential for Real-Time Application:** The study suggests that machine learning models can be adapted for real-time applications, useful for investors or traders looking to make quick decisions.

[2] E. Pagnotta, A. Buraschi, An equilibrium valuation of bitcoin and decentralized network assets, 2018.

An Empirical Assessment" (2018) explores whether cryptocurrencies, particularly Bitcoin, can be considered a viable asset class for investors. The study examines cryptocurrencies in comparison to traditional asset classes (e.g., stocks, bonds, commodities) and analyzes their risk-return profiles, correlations, and potential for diversification. With the rapid rise of digital assets, this paper provides insights into the investment potential of cryptocurrencies and their role in a diversified portfolio.

Key Features

1. **Asset Class Comparison:** The study compares cryptocurrencies to traditional asset classes, assessing their unique characteristics in terms of risk, return, and volatility.
2. **Diversification Potential:** Examines the potential of cryptocurrencies to act as diversification tools in investment portfolios, especially given their low correlation with other assets.
3. **Risk-Return Profile Analysis:** Provides a detailed analysis of the risk-return profiles of leading cryptocurrencies, highlighting their performance and risk factors.
4. **Market Volatility Exploration:** Focuses on the high volatility associated with cryptocurrencies and its implications for investors.
5. **Correlation with Other Assets:** Investigates how cryptocurrencies correlate with assets like stocks, bonds, and commodities, which is crucial for understanding their role in portfolio management.
6. **Data-Driven Approach:** Uses empirical data to support findings, making the analysis reliable for investors and researchers alike.

7. **Implications for Institutional Investors:** Addresses whether institutional investors can or should consider cryptocurrencies as part of their asset mix, given the distinct market dynamics.

[3] I. Madan, S. Saluja, A. Zhao, Automated bitcoin trading via machine learning algorithms, 2015.

The paper "Automated Bitcoin Trading via Machine Learning Algorithms" by Isaac Madan, Shaurya Saluja, and Aojia Zhao (2014) investigates the use of machine learning algorithms to develop an automated trading strategy for Bitcoin. The authors aim to capitalize on Bitcoin's price volatility by building models that predict short-term price movements, thereby creating profitable trading signals. Their study compares several machine learning techniques, including logistic regression, support vector machines, and random forests, to determine which performs best for automated Bitcoin trading.

Key Features

1. **Focus on Automated Trading:** The paper explores the automation of trading strategies for Bitcoin, catering to traders seeking to capitalize on short-term price movements.
2. **Algorithm Comparison:** Compares various machine learning models (e.g., logistic regression, support vector machines, random forests) to identify the most effective for trading predictions.
3. **High-Frequency Data Utilization:** Uses high-frequency Bitcoin trading data, which enhances the ability to detect and respond to rapid market changes.
4. **Backtesting for Validation:** Implements backtesting on historical data to validate the effectiveness of the algorithms, providing an empirical basis for assessing profitability.
5. **Feature Engineering:** Conducts feature engineering to identify the most relevant variables (e.g., price trends, volume, moving averages) for price prediction.
6. **Performance Evaluation Metrics:** Utilizes metrics such as prediction accuracy and profitability to evaluate each model's performance in a trading context.
7. **Potential for Real-Time Application:** Highlights the potential for real-time application, aiming to aid traders in making swift, data-driven trading decisions.

[4] L. Kristoufek, What are the main drivers of the Bitcoin price? evidence from wavelet coherence analysis(2015).

The paper "What Are the Main Drivers of the Bitcoin Price? Evidence from Wavelet Coherence Analysis" by L. Kristoufek, published in PLOS ONE in 2015, investigates the primary factors influencing Bitcoin's price using wavelet coherence analysis. This approach allows the study to capture relationships between Bitcoin prices and various potential drivers across different time scales. Kristoufek examines economic, financial, and technological factors, such as market interest, exchange rates, and macroeconomic indicators, to understand how each correlates with Bitcoin's price movements. The study provides insights into the complexity of Bitcoin's price dynamics and identifies which factors are most influential over time.

Key Features

1. **Wavelet Coherence Analysis:** The study employs wavelet coherence analysis, which captures both the strength and time-dependent nature of relationships, making it ideal for studying dynamic price drivers.
2. **Focus on Multiple Drivers:** Considers a broad range of factors (e.g., economic indicators, trading volume, search trends) to provide a comprehensive view of potential influences on Bitcoin's price.
3. **Time-Scale Analysis:** Analyzes relationships over short, medium, and long-term horizons, offering insights into how different factors impact Bitcoin prices over time.
4. **Exploration of Market Sentiment:** Uses Google Trends data as a proxy for public interest and sentiment, revealing the importance of attention-driven influences on Bitcoin's price.
5. **Analysis of Financial Factors:** Examines the role of financial variables, such as the U.S. dollar exchange rate, in influencing Bitcoin's price, emphasizing Bitcoin's interaction with traditional financial markets.
6. **Nonlinear Dynamics:** Acknowledges the non-linear and complex nature of Bitcoin's price drivers, moving beyond linear correlation methods to capture intricate relationships.

7. **Insight into Speculative and Fundamental Drivers:** Differentiates between speculative interest and fundamental economic drivers, helping to clarify Bitcoin’s role as both an investment and a speculative asset.

[5] P. Ciaian, M. Rajcaniova, d.A. Kancs, The economics of bitcoin price formation(2016).

The paper "The Economics of Bitcoin Price Formation" by P. Ciaian, M. Rajcaniova, and d.A. Kancs, published in Applied Economics in 2016, explores the key economic factors influencing Bitcoin's price formation. The authors develop an economic framework to understand Bitcoin’s unique position as both a currency and an investment asset, analyzing how supply-and-demand dynamics, macroeconomic indicators, and market forces affect its value. By examining Bitcoin's economic characteristics, the study aims to shed light on the drivers that differentiate it from traditional fiat currencies and financial assets.

Key Features

1 **Supply and Demand Analysis:** The paper examines how Bitcoin’s fixed supply structure and demand variability influence price formation, highlighting Bitcoin’s scarcity value.

2 **Economic Model for Price Formation:** Provides an economic model specifically tailored to Bitcoin, accounting for factors like adoption rates, transaction volume, and speculative demand.

3 **Comparison with Traditional Assets:** Analyzes Bitcoin’s similarities and differences with traditional assets, assessing how these differences impact its market behavior and volatility.

4 **Macroeconomic Influence:** Investigates how traditional macroeconomic variables (e.g., inflation rates, exchange rates) affect Bitcoin prices, drawing parallels with fiat currency behaviors.

5 **Investor Demand and Speculation:** Acknowledges the significant role of speculative demand in Bitcoin's price, reflecting its appeal as a high-risk, high-reward investment.

6 **Network Effects:** Considers network effects where increased adoption and usage drive up value—as a unique factor in Bitcoin price formation, particularly important for digital assets.

7 **Market Efficiency Exploration:** Explores whether Bitcoin markets exhibit efficient price formation or are prone to bubbles, giving insights into market maturity and stability.

2.2 LIMITATIONS

[1] S. McNally, J. Roche, S. Caton, Predicting the price of bitcoin using machine learning, in: 2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing, PDP, IEEE, 2018.

1 **Model Interpretability:** Some of the models, particularly deep learning methods, can be difficult to interpret, limiting their transparency and usability for investors.

2 **Overfitting Risks:** Given the volatility of Bitcoin, models may overfit to short-term trends and fail to generalize well to new data.

3 **Dependency on Feature Quality:** Effective predictions are heavily reliant on high-quality, well-chosen features, and irrelevant or noisy features can degrade model performance.

4 **Data Availability Constraints:** Models require large amounts of historical data, which can be problematic due to Bitcoin's relatively short trading history compared to other assets.

5 **Lack of Long-Term Predictive Power:** Machine learning models are often better suited for short-term predictions and may struggle to accurately forecast long-term price movements.

6 **Computational Cost:** Some models, particularly those involving deep learning, are computationally intensive, which can limit their practicality for real-time forecasting in low-resource environments.

[2] E. Pagnotta, A. Buraschi, An equilibrium valuation of bitcoin and decentralized network assets, 2018.

1. **Volatile Nature of Data:** The study's findings are sensitive to the high volatility and rapid changes in cryptocurrency markets, which may reduce the generalizability over time.
2. **Limited Historical Data:** Cryptocurrencies have a relatively short trading history compared to traditional assets, potentially affecting the robustness of conclusions.
3. **Exclusion of Emerging Cryptocurrencies:** The analysis may focus on a limited number of major cryptocurrencies, overlooking the potential of emerging ones.
4. **Regulatory Uncertainty:** Regulatory developments in cryptocurrency markets are rapidly evolving, which could impact their behavior as an asset class and alter the study's conclusions.
5. **Market Manipulation Risks:** Cryptocurrency markets have been subject to manipulation, which could distort data and affect the validity of investment insights.
6. **Potential for High Drawdowns:** The study may not fully capture the risks associated with extreme market drawdowns, which are common in cryptocurrency markets.
7. **Narrow Focus on Investment Value:** Concentrates mainly on the investment aspects of cryptocurrencies, potentially underestimating their utility as decentralized transaction mechanisms or their role in digital finance beyond investment.

[3] I. Madan, S. Saluja, A. Zhao, Automated bitcoin trading via machine learning algorithms, 2015.

1. **Limited Market Context:** The study focuses solely on Bitcoin and does not consider other cryptocurrencies or asset classes, potentially limiting its generalizability.
2. **Dependency on High-Quality Data:** The success of machine learning models depends on the availability of high-quality, real-time data, which can be challenging to obtain consistently.
3. **Overfitting Risk:** Some models, especially those with complex feature engineering, may overfit historical data, reducing their effectiveness in live trading.

4 **Lack of Consideration for Transaction Costs:** Transaction fees, slippage, and other trading costs are not fully considered, which may overestimate the profitability of the strategies.

5 **Sensitivity to Market Changes:** Bitcoin's price is affected by various external factors (e.g., regulations, market sentiment) that are not included in the models, limiting their robustness.

6 **Computational Demand:** Real-time automated trading requires substantial computational power, which could be a limitation for implementation in resource-constrained environments.

7 **Market Manipulation Risks:** Bitcoin markets are prone to manipulation and illiquidity issues, which can significantly impact trading strategies but are not accounted for in the models.

[4] L. Kristoufek, What are the main drivers of the Bitcoin price? evidence from wavelet coherence analysis.

1. **Short Data History:** At the time of study, Bitcoin had a relatively short trading history, which may limit the robustness and generalizability of long-term findings.
2. **Data Quality and Availability:** The study relies on proxies (e.g., Google Trends) to measure public interest, which may not fully capture sentiment or real market dynamics.
3. **Limited to Correlation:** Wavelet coherence analysis highlights correlations but does not imply causation, so the findings cannot definitively determine cause-and-effect relationships.
4. **High Sensitivity to External Events:** Bitcoin is highly responsive to unpredictable events (e.g., regulations, market crashes), which are difficult to account for in the analysis.
5. **Exclusion of Emerging Drivers:** As Bitcoin has evolved, newer drivers of price (such as institutional investment and regulatory changes) may not be fully captured in this 2015 study.
6. **Complexity of Methodology:** Wavelet coherence analysis can be challenging to interpret for non-technical readers, limiting accessibility for broader audiences.

7. **Dependency on Market Sentiment:** While sentiment is examined, it can fluctuate rapidly and unpredictably, making it difficult to reliably incorporate into long-term models or strategies.

[5] P. Ciaian, M. Rajcaniova, d.A. Kancs, The economics of bitcoin price formation.

1 **High Volatility Impact:** Bitcoin's extreme volatility complicates the analysis of its economic drivers, as rapid price changes may obscure stable, long-term patterns.

2 **Speculative Nature Not Fully Quantifiable:** While speculative demand is identified, its exact impact on Bitcoin prices is difficult to measure precisely due to unpredictable investor sentiment.

3 **Limited Data Set:** The analysis is based on data available only up to 2016, which may limit the relevance of findings given Bitcoin's significant market evolution since then.

4 **Simplified Economic Model:** The economic model may oversimplify certain aspects of Bitcoin's complex dynamics, such as regulatory impacts or technological developments.

5 **Potential Overemphasis on Macro Factors:** The study may overemphasize traditional macroeconomic indicators, which could have a less direct effect on Bitcoin than on fiat currencies.

6 **Network Effects Measurement Challenges:** Network effects, though crucial for digital assets, are challenging to quantify accurately, limiting the reliability of conclusions drawn.

7 **Exclusion of Regulatory Impact:** Regulatory developments, which can significantly affect Bitcoin prices, are not thoroughly considered, potentially underestimating their influence on market behavior.

CHAPTER 3

SOFTWARE & HARDWARE SPECIFICATIONS

3.1 SOFTWARE REQUIREMENTS

Operating system

- Windows, macOS, or Linux

Programing Language

- Python (version 3.6 or later)

Python Libraries

- Keras and TensorFlow
- Scikit-learn (Sklearn)
- XGBoost Library
- Matplotlib/Seaborn

Data Sources and APIs

- Access to CoinMarketCap API

Integrated Environment Development(IDE)

- Jupyter Notebook or Google Colab
- VS Code

3.2 HARDWARE REQUIREMENTS

Processor (CPU)

- Intel Core i5

Memory (RAM)

- Minimum: 8 GB RAM

Storage:

- Minimum: 256 GB SSD

CHAPTER 4

PROPOSED SYSTEM DESIGN

4.1 PROPOSED METHODS

1.Granularity-based Model Selection:

- **Daily Data Prediction (GBP-D):** The model leverages higher-dimensional feature sets for predicting Bitcoin's daily price. Due to the smaller dataset size, simpler statistical models such as Logistic Regression (LR) and Linear Discriminant Analysis (LDA) are selected, which are less prone to overfitting.
- **High-frequency Data Prediction (GBP-M):** This model uses data from 5-minute intervals with fewer features due to the volume of data and greater price fluctuation at high frequency. Complex machine learning models like Random Forest, XGBoost, Quadratic Discriminant Analysis (QDA), Support Vector Machine (SVM), and Long Short-Term Memory (LSTM) are employed, which can capture more intricate patterns and dependencies in the high-frequency data.

2.Feature Engineering:

- **Daily Data:** High-dimensional features are engineered, including network properties, market data, and investor attention.
- **High-frequency Data:** Features are selected directly from trading data provided by Binance, focusing on metrics such as price, trading volume, and open/close points.

3.Model Selection and Training:

- **Statistical Models for Daily Price:** Simpler models like LR and LDA are used on high-dimensional feature sets for daily prediction to avoid overfitting.
- **Machine Learning Models for High-Frequency Price:** More complex models, including Random Forest, XGBoost, QDA, SVM, and LSTM, are implemented for the 5-minute interval predictions, capitalizing on their ability to handle large datasets and more complex data structures.

4.Evaluation Metrics:

- The study evaluates model performance using accuracy, precision, recall, and F1-score to assess how well each model predicts the correct price direction across the two granularity-based datasets.

4.2 CLASS DIAGRAM

The class diagram outlines a pipeline for Bitcoin price prediction using machine learning. It starts with Data Acquisition for collecting and validating data, followed by Data Preprocessing to clean and structure it. Then, Feature Engineering transforms the data and generates useful features. In Sample Dimension Engineering, the dataset is adjusted for optimal performance.

Next, Model Selection and Training picks and trains the appropriate machine learning model. The Prediction Evaluation module makes predictions and assesses the model's performance. After that, Deployment and Monitoring puts the model into production and tracks its performance. Finally, the Reporting module generates reports and shares the results with stakeholders.

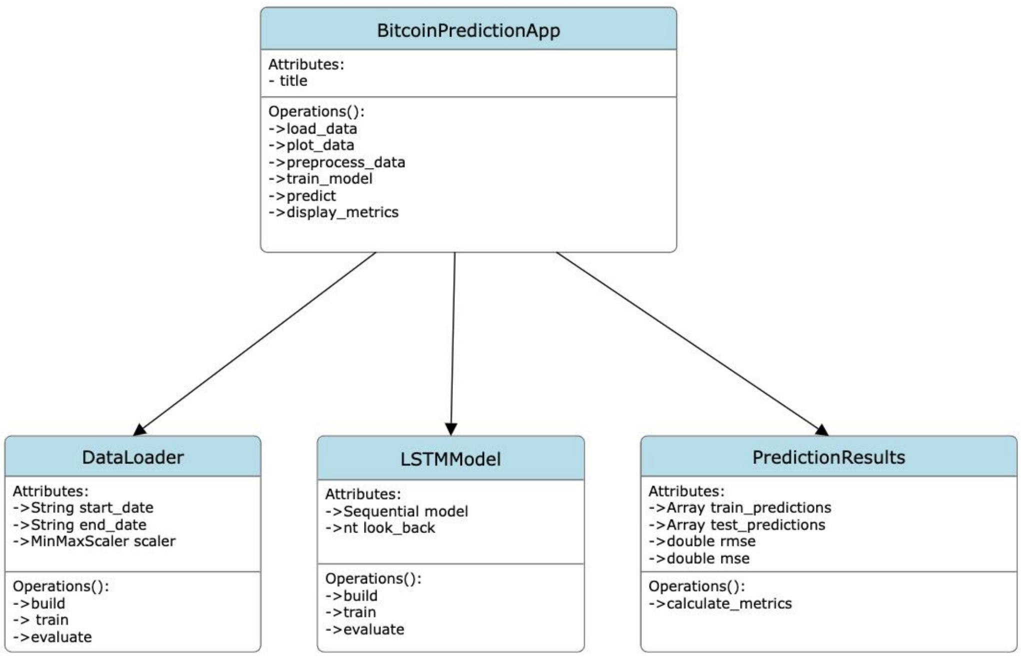


Figure 4.1 Class Diagram

4.3 USE CASE DIAGRAM

The use case diagram outlines a process flow for predicting Bitcoin prices using an LSTM model. It begins with the **Bitcoin Dataset** acquisition, followed by **Data Preprocessing** to prepare the data. The processed data is then split into **Training** and **Testing** sets. The **Long Short-Term Memory (LSTM)** model is trained with the training data and tested on the testing set. The model's performance is assessed in the **Evaluation of Accuracy** step. If accuracy meets the desired threshold, the process proceeds to **Compare Predicted Results** with actual values to validate the model's effectiveness, marking the **End** of the prediction workflow.

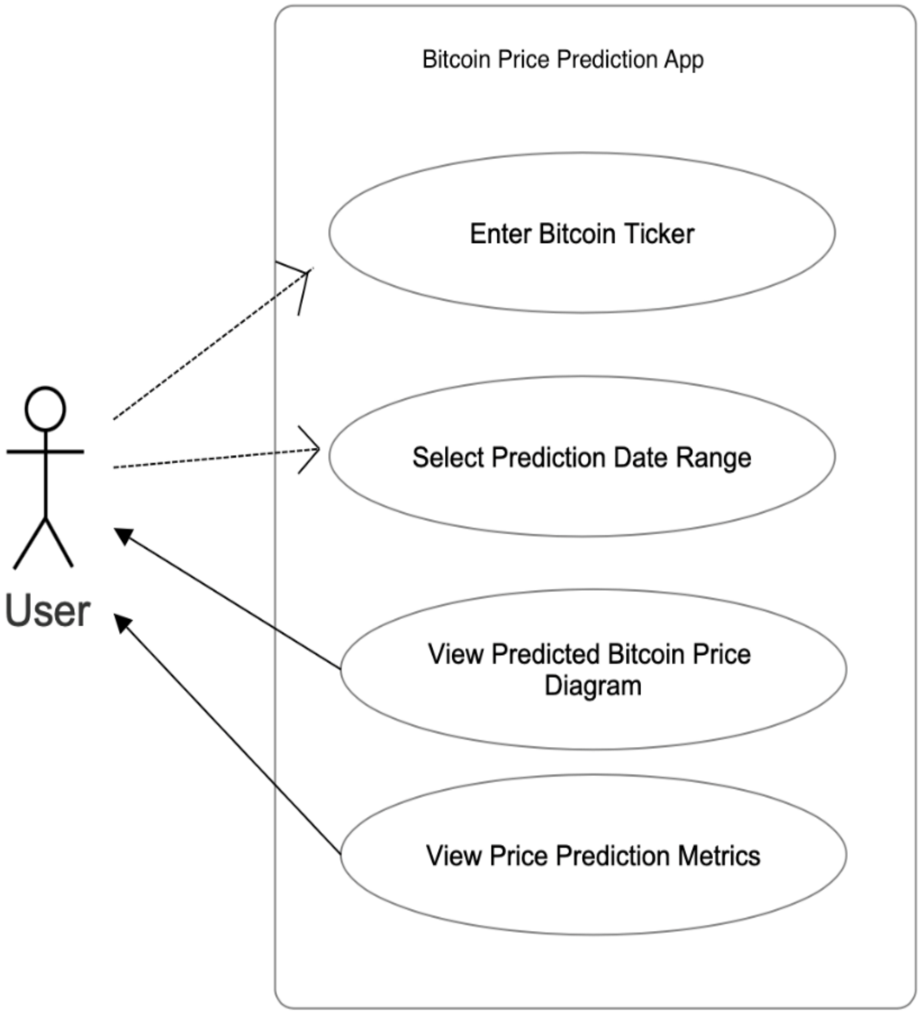


Figure 4.2 Use Case Diagram

4.4 ACTIVITY DIAGRAM

This activity diagram outlines a typical workflow for a machine learning or data science project. It begins with data acquisition, where relevant data is gathered, followed by preprocessing to clean and format the data. Next, feature engineering is performed to create useful input variables, and sample dimension engineering modifies the dataset structure if needed. Model selection and training come next, choosing and training the best model for the task. The model is then evaluated and tested in the prediction and evaluation module. Once ready, it moves to deployment and monitoring, making it accessible for end-users and monitoring its performance over time. Finally, results are compiled and reported in the reporting module, concluding the process.

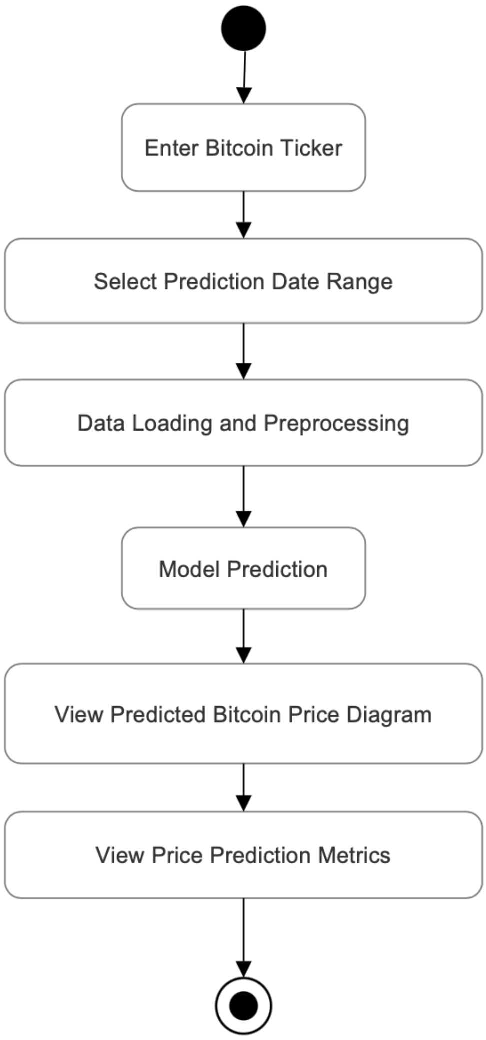


Figure 4.3 Activity Diagram

4.5 SEQUENCE DIAGRAM

This sequence diagram illustrates a process for predicting Bitcoin prices using a machine learning workflow. It begins with the user providing Bitcoin data, which then undergoes preprocessing to clean and prepare it for analysis. The cleaned data is used to train a model, and the trained model is subsequently tested. An LSTM (Long Short-Term Memory) model, specialized for time series data, is applied to make predictions. The model's accuracy is then evaluated, and the predicted results are compared to actual values to assess performance. Finally, the final prediction output is provided to the user.

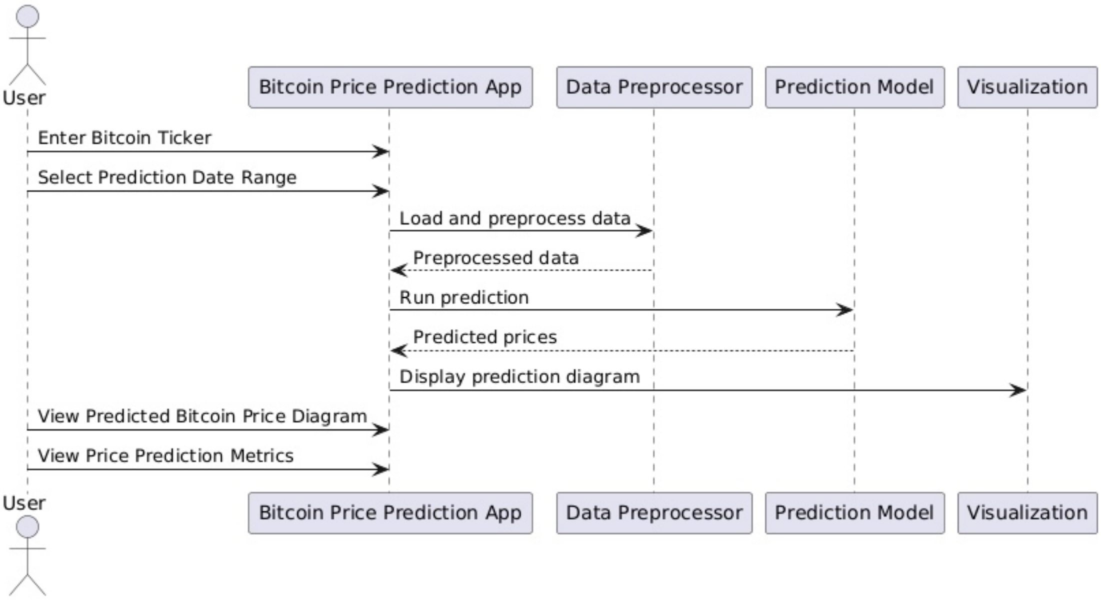


Figure 4.4 Sequence Diagram

4.6 SYSTEM ARCHITECTURE :

The system architecture for Bitcoin price prediction using LSTM networks starts by processing a historical Bitcoin dataset, which is split into training and testing sets. The LSTM model is trained on the data to capture long-term patterns in price movements. After training, the model's accuracy is evaluated using metrics like precision and recall. If the predictions are accurate, they are compared with real-world price data for validation. This approach uses LSTM's strength in handling time-series data to forecast Bitcoin price trends.

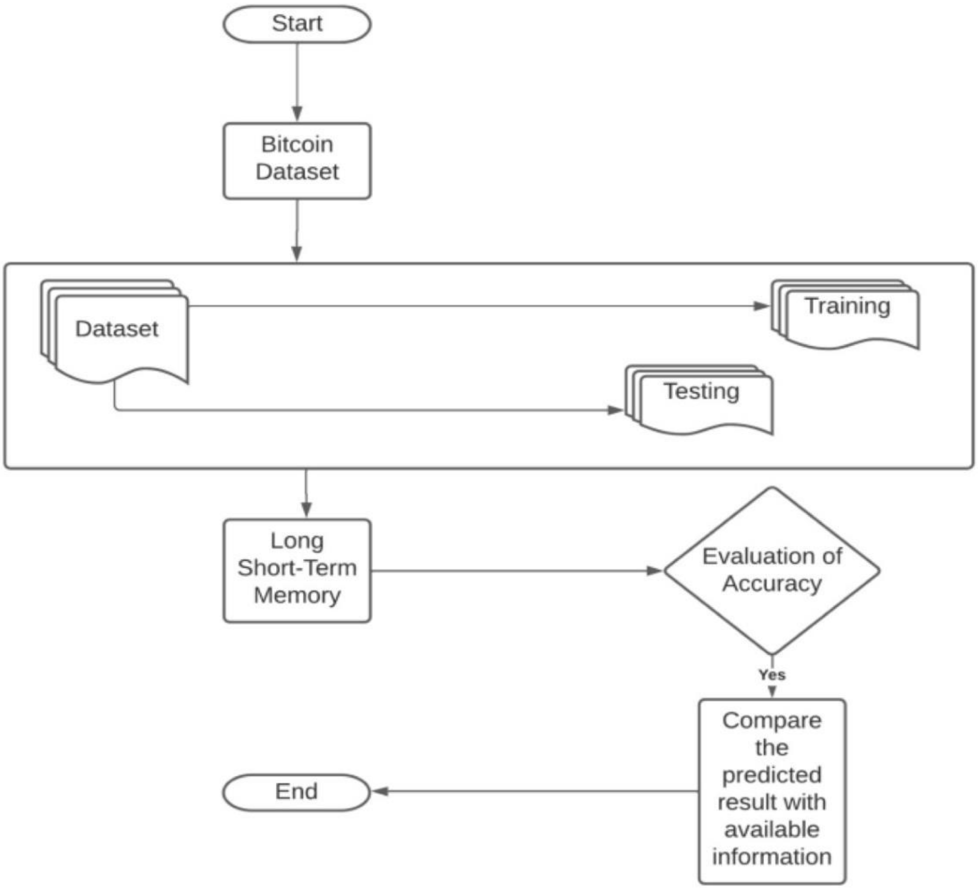


Figure 4.5 system architecture

4.7 TECHNOLOGY DESCRIPTION

1. Programming Language: Python

- Python is the main programming language used due to its extensive ecosystem of libraries for machine learning, data analysis, and scientific computing.
- Python offers powerful tools for data manipulation, statistical analysis, and visualization, crucial for tasks such as price trend prediction.

2. Machine Learning and Deep Learning Libraries

- **Scikit-Learn:**
  - This library provides a broad range of simple and efficient tools for data mining and data analysis, making it ideal for implementing machine learning algorithms like Logistic Regression, Linear Discriminant

Analysis (LDA), Quadratic Discriminant Analysis (QDA), Support Vector Machine (SVM), and Random Forest.

- **XGBoost:**
  - XGBoost is an advanced gradient boosting framework optimized for speed and performance, essential for high-frequency, high-dimensional data like Bitcoin price data.
  - It provides faster computation than traditional boosting algorithms and can handle missing data, which is common in real-world datasets.
- **Keras (or TensorFlow backend):**
  - Keras is a high-level neural networks API that enables quick development of deep learning models.
  - Keras is typically run on a TensorFlow backend, which provides optimized operations for deep learning tasks, particularly when working with GPUs.
  - TensorFlow's automatic differentiation and Keras's intuitive layer structure help in rapidly prototyping, training, and tuning LSTM models for time-sensitive tasks.

3. Data Manipulation and Numerical Computation Libraries

- **Pandas:**
  - Pandas is essential for data preprocessing and manipulation, offering data structures like DataFrames for managing and cleaning data.
- **NumPy:**
  - NumPy is used to handle the numerical calculations and array manipulations that underlie machine learning algorithms, making them faster and more memory-efficient.

4. Visualization Libraries

- **Matplotlib and Seaborn:**
  - Matplotlib is a plotting library in Python used for creating static, animated, and interactive visualizations. It's essential for exploring data and model performance.

- Seaborn, built on top of Matplotlib, provides a more accessible interface and enables the creation of visually appealing statistical graphics.

5. Development Environment

- **Jupyter Notebook:**
  - Jupyter Notebook is an interactive computing environment that enables users to write and execute code in a cell-based format. It is particularly useful for data science tasks as it allows for exploratory data analysis, visualization, and iterative model development in a single interface.

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 Implementation of Frontend

In the Bitcoin price prediction project, Streamlit was used to create an interactive and intuitive web application for the frontend. The purpose of using Streamlit was to provide an easy-to-use interface for users to interact with the predictive model without needing to understand the underlying code.

Here’s how Streamlit was employed to build the webapp frontend:

1. **Input Forms:** Streamlit allows for easy creation of input forms that users can fill out. In the Bitcoin price prediction webapp, users are prompted to enter stock tickers (e.g., AAPL, MSFT), select a time range for prediction, and choose other relevant parameters. Using the st.text_input() or st.sidebar() functions, users can easily enter and select the options needed for predictions. This provides a clean and simple interface for non-technical users to interact with the model.
2. **Displaying Data:** The historical bitcoin data fetched from YFinance was displayed in a clear, tabular format using Streamlit’s built-in st.dataframe() function. This helps users visualize the raw bitcoin data they are working with before making any predictions. The dataframe is updated dynamically based on the user’s input, making the webapp responsive and interactive.
3. **Graphical Visualization:** Streamlit was used to generate interactive charts and plots that showcase both historical and predicted bitcoin prices. Using libraries such as Matplotlib or Plotly, the bitcoin data and the forecasted values were plotted. Streamlit's st.line_chart() or st.pyplot() functions allow these graphs to be embedded directly into the webapp, providing users with visual insights into the bitcoin trends and predictions.
4. **Displaying Prediction Results:** Once the user inputs their query and the bitcoin prediction model is run, the results—predicted stock prices—are displayed on the webapp. Streamlit’s st.write() function is used to present the output in a

readable format, whether as text or visual elements. This ensures that users can see the model's predictions and analyze the performance directly on the app.

5. **Live Updates and Real-time Interaction:** One of Streamlit's core features is its ability to refresh and update content dynamically. As soon as the user modifies any input (such as stock ticker or time frame), the webapp triggers the model to run again, updating the displayed data and predictions in real-time. This makes the app highly responsive and interactive.
6. **Customization and Styling:** Streamlit also allows customization of the layout, making the webapp visually appealing. For instance, the sidebar can be used to contain user inputs, while the main area of the webapp can display the results, graphs, and additional insights, providing an organized and structured look.

In summary, Streamlit played a crucial role in transforming the machine learning model for bitcoin prediction into a user-friendly, interactive web application. It allowed for seamless integration of data input, real-time updates, and visualizations, making the tool accessible to both technical and non-technical users.

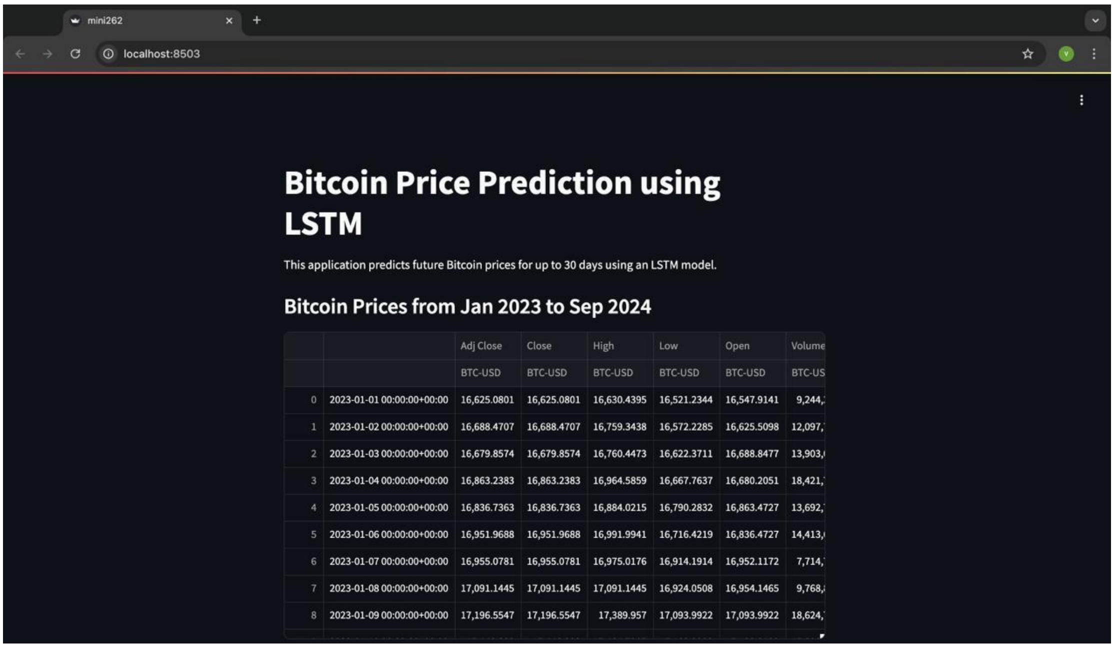


Figure 5.1 Home Page

The "Home Page" of the Bitcoin price prediction application displays a user interface that predicts future Bitcoin prices for up to 30 days using an LSTM model. It provides a table listing historical Bitcoin prices, including metrics like "Close," "High," "Low," "Open," and "Volume," spanning from January 2023 to September

2024. This page is designed to be user-friendly, making the tool accessible for both technical and non-technical users interested in analyzing Bitcoin's price trends.

5.2 Implementation of the Bitcoin Price Prediction

The implementation of the stock price prediction model involves several key steps, including data Load, preprocessing, model development (using the LSTM architecture), training, and deployment via Streamlit. Here’s a breakdown of each stage of the implementation:

1. **Data Collection:**

   The data for the Bitcoin price prediction model is sourced from Yahoo Finance using the YFinance Python library. This data includes historical bitcoin prices such as open, close, Adj close, high, low, and volume . YFinance enables easy extraction of this data over custom time periods, which serves as input for the machine learning model.

   Code :

```
def load_data():
    data = yf.download('BTC-USD', start='2023-01-01', end='2024-09-30')
    data.reset_index(inplace=True)
    return data

data = load_data()

# Display last few rows of data
st.subheader("Bitcoin Prices from Jan 2023 to Sep 2024")
st.write(data)
```

2. **Historical Bitcoin Price:**

   The "Historical Bitcoin Price" section shows a code snippet for plotting Bitcoin's historical closing prices using Matplotlib in a Streamlit application. It generates a line chart with date on the x-axis and price in USD on the y-axis, providing a visual representation of Bitcoin's past performance.

```
# Plot historical prices
st.subheader("Historical Bitcoin Price")
fig, ax = plt.subplots()
ax.plot(data['Date'], data['Close'], label='Close Price')
ax.set_xlabel('Date')
ax.set_ylabel('Price in USD')
ax.legend()
st.pyplot(fig)
```

3. **Data Preprocessing:**

Before training the model, the data must be preprocessed to ensure it's in a suitable format.

- **Normalization:** The bitcoin price data is scaled using techniques like Min-Max Scaling to bring all values into a range between 0 and 1, improving the efficiency of the LSTM model.

Code :

```
scaler = MinMaxScaler(feature_range=(0, 1))
scaled_data = scaler.fit_transform(data['Close'].values.reshape(-1, 1))
```

- **Data Splitting:** The dataset is split into training and testing sets, usually with 80% used for training and the remaining 20% for testing.

Code :

```
train_size = int(len(scaled_data) * 0.8)
train_data, test_data = scaled_data[:train_size], scaled_data[train_size:]

# Function to create dataset with look_back
def create_dataset(data, look_back=60):
    x, y = [], []
    for i in range(look_back, len(data)):
        x.append(data[i - look_back:i, 0])
        y.append(data[i, 0])
    return np.array(x), np.array(y)

look_back = 60
x_train, y_train = create_dataset(train_data, look_back)
x_test, y_test = create_dataset(test_data, look_back)
```

4. **Model Development and Training - LSTM Architecture:**

The heart of the bitcoin price prediction system is the LSTM (Long Short-Term Memory) network. LSTM is a specialized type of Recurrent Neural Network (RNN) that excels in learning from time-series data due to its ability to retain long-term dependencies. The LSTM model is built using libraries like TensorFlow or Keras.

Code :

```
# Reshape data for LSTM model
x_train = np.reshape(x_train, (x_train.shape[0], x_train.shape[1], 1))
x_test = np.reshape(x_test, (x_test.shape[0], x_test.shape[1], 1))

# Build the LSTM model
model = tf.keras.Sequential([
    tf.keras.layers.LSTM(50, return_sequences=True, input_shape=(x_train.shape[1], 1)),
    tf.keras.layers.LSTM(50, return_sequences=False),
    tf.keras.layers.Dense(25),
    tf.keras.layers.Dense(1)
])

# Compile the model
model.compile(optimizer='adam', loss='mean_squared_error')

# Train the model
model.fit(x_train, y_train, batch_size=1, epochs=50)
```

5. Model Testing and Evaluation:

Once trained, the model is tested on the unseen test data to evaluate its performance. The predicted values are then compared to the actual stock prices to assess the model's accuracy. The predictions from the LSTM model are then transformed back to the original scale using the inverse of the scaler applied earlier.

Code :

```
# Make predictions on the test data
train_predictions = model.predict(x_train)
test_predictions = model.predict(x_test)
```

5.3 Testing

In the testing phase, the model's performance in predicting Bitcoin prices was evaluated by comparing its predictions with actual historical data. This phase involved splitting the data into training and testing sets to observe how well the model generalizes to unseen data. Key metrics, including accuracy, RMSE, and MSE, were used to quantify the model's prediction accuracy and error rates. The results, as visualized through figures, highlight the model's ability to capture historical price trends and make reliable future predictions, providing insights into its effectiveness for forecasting Bitcoin prices.

Input : Historical Bitcoin Price Data (used to train the prediction model).The model likely uses a machine learning technique to predict future Bitcoin prices based on this historical data.

Expected Output : Predictions of Bitcoin prices for a future period, compared against the actual test data.The model should provide accuracy metrics to evaluate performance.

Result : The model achieved a high accuracy of 97.59%, with RMSE of 13,707.47 and MSE of 187,756,537.9, indicating strong predictive performance.

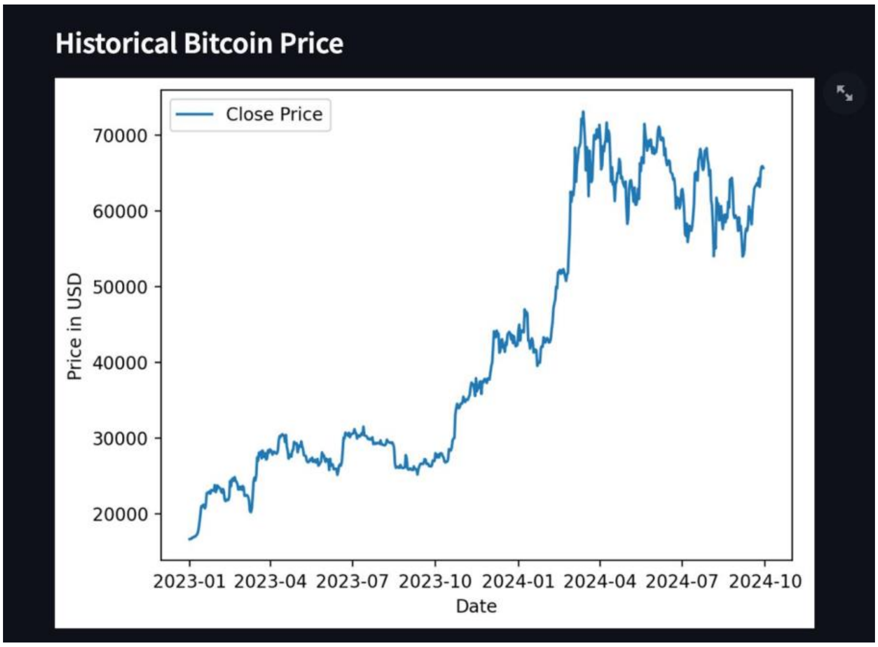


Figure 5.2 Historical Bitcoin Price

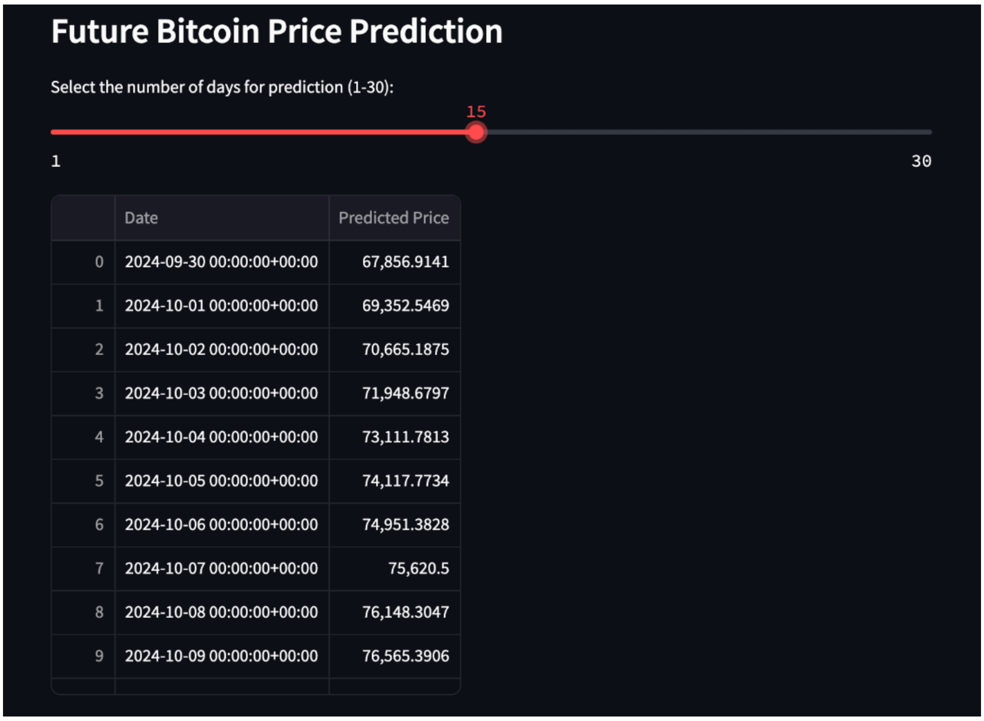
This plot shows the historical closing prices of Bitcoin from January 2023 to November 2024, illustrating trends and fluctuations in Bitcoin's value over time.

Figure 5.3 Train and Test Predictions

This plot compares the actual Bitcoin prices with the model's predicted prices for the training and test data, showcasing the model’s accuracy in learning historical patterns.

Figure 5.4 Accuracy of the model

This figure displays the model's performance metrics, including RMSE, MSE, and overall accuracy percentage, to evaluate prediction reliability.



	Date	Predicted Price
0	2024-09-30 00:00:00+00:00	67,856.9141
1	2024-10-01 00:00:00+00:00	69,352.5469
2	2024-10-02 00:00:00+00:00	70,665.1875
3	2024-10-03 00:00:00+00:00	71,948.6797
4	2024-10-04 00:00:00+00:00	73,111.7813
5	2024-10-05 00:00:00+00:00	74,117.7734
6	2024-10-06 00:00:00+00:00	74,951.3828
7	2024-10-07 00:00:00+00:00	75,620.5
8	2024-10-08 00:00:00+00:00	76,148.3047
9	2024-10-09 00:00:00+00:00	76,565.3906

Figure 5.5 Future Bitcoin Price Prediction

This table presents the predicted Bitcoin prices for a user-defined number of future days, allowing users to view projected prices.

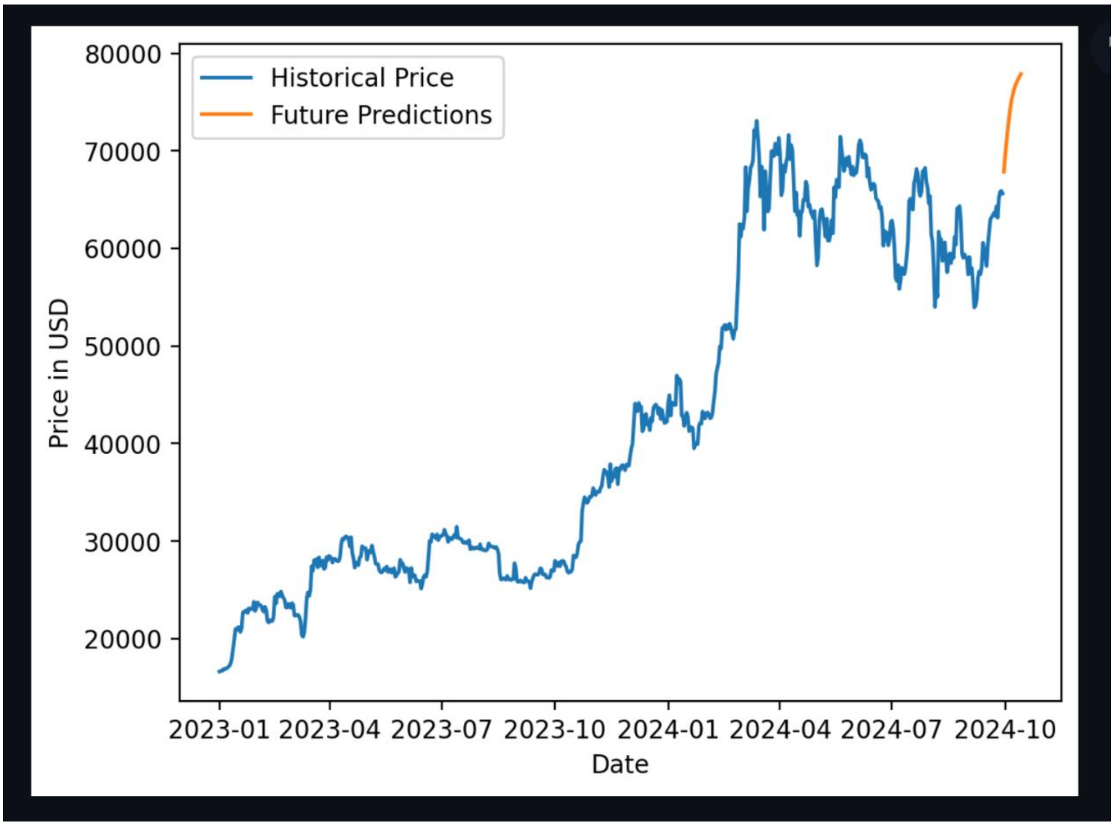


Figure 5.6 Comparing Historic data with Future Predictions

This plot overlays historical Bitcoin prices with the predicted future prices, giving a visual forecast for Bitcoin’s future trend based on the model's predictions.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

It provides a significant advancement in Bitcoin price prediction by aligning machine learning model complexity with the specific characteristics of data samples, namely sample granularity and feature dimensionality. Through an approach based on Occam's razor, the researchers demonstrate that simpler statistical models like Logistic Regression and Linear Discriminant Analysis can perform effectively on high-dimensional, low-frequency daily data. Conversely, high-frequency Bitcoin data benefits from complex models such as LSTM and SVM, which are better suited for capturing rapid, nuanced market fluctuations. It provides a scalable framework for predictive modeling in dynamic financial environments, marking a step forward for machine learning applications in finance and other domains that require tailored approaches to complex data.

6.2 FUTURE SCOPE

Future research could expand Bitcoin price prediction by incorporating macroeconomic indicators, regulatory changes, and advanced sentiment analysis from social media, improving predictions based on broader influences and market psychology. Exploring sophisticated models like Transformer-based or reinforcement learning algorithms could capture complex patterns in high-frequency trading data. Additionally, applying this approach to other volatile assets like altcoins, stocks, or commodities would test its adaptability, potentially creating more versatile predictive frameworks across various financial markets. This study opens doors for refined, data-informed models tailored to diverse and dynamic financial environments.

REFERENCES

1. S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system, 2008.

2. D. Yermack, Is bitcoin a real currency? an economic appraisal, in: Handbook of Digital Currency, Elsevier, 2015, pp. 31–43.

3. F. Mai, et al., How does social media impact bitcoin value? a test of the silent majority hypothesis, J. Manage. Inf. Syst. 35 (1) (2018) 19–52.

4. I. Madan, S. Saluja, A. Zhao, Automated bitcoin trading via machine learning algorithms, vol. 20. URL: http://cs229.stanford.edu/proj2014/Isaac% 20Madan, 2015.

5. P.G. Nieto, et al., A comparison of several machine learning techniques for the centerline segregation prediction in continuous cast steel slabs and evaluation of ts performance, J. Comput. Appl. Math. 330 (2018) 877–895.

6. B.M. Brentan, et al., Hybrid regression model for near real-time urban water demand forecasting, J. Comput. Appl. Math. 309 (2017) 532–541.

7. C. Ordóñez, et al., A hybrid ARIMA–SVM model for the study of the remaining useful life of aircraft engines, J. Comput. Appl. Math. 346 (2019) 184–191.

8. I. Georgoula, et al., Using time-series and sentiment analysis to detect the determinants of bitcoin prices, SSRN 2607167, 2015..

9. A. Greaves, B. Au, Using the bitcoin transaction graph to predict the price of bitcoin, stanford.edu, 2015.

PLAGIARISM REPORT

Chapter No.	Chapter Name	Number of Words	Plagiarism
	Abstract	203	0%
1	Introduction	914	0%
2	Literature Survey	464	0%
3	Software and Hardware Specificcations	48	0%
4	Proposed System Design	977	0%
5	Implementation and Testing	1231	0%
6	Conclusion	299	0%
Total		**2,848**	**0%**

Note: The “Duplichecker” plagiarism tool is used for checking chapter-wise plagiarism. Duplichecker --- https://www.duplichecker.com/