MINI PROJECT REPORT

Bitcoin Price Prediction Using Historical Data and LSTM

A Mini Project Report

Submitted By:

(21BT04014) Dhyey Bhatt

(21BT04064) Daksh Patel

(21BT04065) Devanshu Patel

Submitted To:

DR. Deepak Parashar

Abstract

This project focuses on predicting Bitcoin prices using historical data through a Long Short-Term Memory (LSTM) neural network model. Given the volatile nature of cryptocurrency markets, accurate price prediction is essential for investors and traders. The primary objective of this study is to implement an LSTM model that leverages past closing prices of Bitcoin from 2017 to 2023 to learn underlying patterns and trends. The dataset was preprocessed to ensure data integrity, and subsequently split into training and testing sets. The model's performance was evaluated using Mean Squared Error (MSE) to quantify prediction accuracy and Mean Absolute Percentage Error (MAPE) to assess the model's reliability in predicting price changes. The results indicate that the LSTM model effectively captures trends in the historical data, achieving satisfactory prediction accuracy. The findings highlight the potential of LSTM networks in financial forecasting and suggest avenues for future research, including the incorporation of additional market indicators.

Introduction

Cryptocurrencies have emerged as a revolutionary form of digital currency, fundamentally changing how we perceive and conduct financial transactions. Among these, Bitcoin stands out as the first and most well-known cryptocurrency, experiencing dramatic fluctuations in its price since its inception. As the cryptocurrency market continues to grow, the need for effective price prediction methods becomes increasingly critical for traders and investors looking to make informed decisions. Accurate predictions can help mitigate risks associated with this highly volatile market, potentially maximizing profits while minimizing losses.

Recent advancements in machine learning, particularly in deep learning techniques such as Long Short-Term Memory (LSTM) networks, offer promising solutions for time-series forecasting tasks, including financial predictions. LSTMs are particularly well-suited for sequential data due to their ability to capture long-term dependencies, making them an ideal choice for predicting Bitcoin prices based on historical data.

This project aims to address the following problem statement: **Can we accurately predict Bitcoin prices using historical data and LSTM networks?**

The objectives of this study are:

1. **Data Preprocessing:** To clean and preprocess historical Bitcoin price data to ensure its readiness for model training.
2. **Model Implementation:** To design and implement an LSTM model that learns from historical price patterns.
3. **Performance Evaluation:** To evaluate the model’s performance using Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE) metrics, ensuring the reliability of the predictions.
4. **Insights Generation:** To analyse the results and discuss the model's effectiveness in capturing price trends, as well as to propose potential future improvements.

Literature Review

Over the years, there has been a growing interest in predicting Bitcoin prices due to the cryptocurrency's popularity and volatility. Many researchers have explored various methods to forecast these prices, aiming to provide better insights for investors and traders.

Initially, traditional statistical methods, such as the Autoregressive Integrated Moving Average (ARIMA) model, were commonly used. These models analyze historical price data to make predictions. While ARIMA can work well for short-term forecasts, it often struggles with the unpredictable nature of Bitcoin prices over longer periods.

As the field evolved, machine learning techniques gained traction. Algorithms like Support Vector Machines (SVM) and Random Forests started to outperform traditional methods. For example, studies showed that machine learning models could learn from larger datasets and better capture the complex patterns in price movements. Researchers found that these advanced methods provided more accurate predictions, helping to improve trading strategies.

Among the machine learning techniques, Long Short-Term Memory (LSTM) networks have become particularly popular. LSTMs are a type of recurrent neural network designed to handle sequences of data, making them well-suited for time series forecasting like Bitcoin prices. Research has demonstrated that LSTM models can effectively identify patterns in historical data, leading to better predictions than simpler neural networks.

Furthermore, combining different models has emerged as a promising approach. Hybrid models that mix LSTMs with other techniques, such as Gated Recurrent Units (GRUs), have shown improved performance by leveraging the strengths of each method. Additionally, the inclusion of various features—like trading volume and social media sentiment—has been explored. Studies indicate that adding these features enhances the accuracy of predictions, as they provide a broader context for understanding market behaviour.

In conclusion, the literature highlights the shift from traditional statistical methods to advanced machine learning techniques for predicting Bitcoin prices. LSTM networks, in particular, have proven effective due to their ability to learn from historical data patterns. This project aims to build on these insights by employing LSTM models to forecast Bitcoin prices, contributing to the ongoing efforts to improve prediction methods in the dynamic world of cryptocurrencies.

Methodology

4.1 Data Collection

The dataset utilized in this project comprises daily Bitcoin price data spanning from 2017 to 2023. The data was sourced from Kaggle, providing a comprehensive collection of historical Bitcoin prices, including essential features such as timestamps, opening prices, closing prices, highs, lows, and trading volume. The primary focus is on the closing prices, as they are critical for understanding market trends. For the specific dataset used, please refer to the link in the Appendix section.

4.2 Data Preprocessing

The data preprocessing phase involved several key steps to ensure the dataset's integrity and suitability for model training:

1. **Data Cleaning:**
    - Irrelevant columns such as trading volume and other metrics not pertinent to price prediction were removed from the dataset.
    - The dataset was examined for duplicates and null values, with necessary actions taken to clean the data.
2. **Date Formatting and Resampling:**
    - The timestamp column was converted into a datetime format to facilitate time-based operations.
    - The data was resampled to a daily frequency, taking the maximum value for each day to account for fluctuations within the trading day.
3. **Feature Scaling:**
    - Since LSTM networks are sensitive to the scale of the input data, the closing prices were normalized using MinMaxScaler from the Scikit-learn library. This scaler transforms the data to a range between 0 and 1, enhancing the training stability and convergence speed of the neural network.
4. **Windowing:**
    - To prepare the data for the LSTM model, a sliding window approach was employed. Each input sequence consisted of 60 previous closing prices, allowing the model to learn from historical trends. The output for each sequence was the closing price immediately following that window.

4.3 Model Development

The project implemented a Long Short-Term Memory (LSTM) model using TensorFlow and Keras. The model architecture consisted of:

1. **Input Layer:**
    - The model accepts input in the shape of (window_size, 1) to represent 60 previous closing prices.

2. **LSTM Layers:**
    - Three LSTM layers, each with 64 units, were employed. The first two layers return sequences, allowing the subsequent layer to process the full output from the previous layers.

3. **Dropout Layers:**
    - Dropout layers with a rate of 0.2 were added after each LSTM layer to reduce overfitting by randomly setting a fraction of the input units to zero during training.

4. **Dense Layers:**
    - A dense layer with 32 units and a **SoftMax** activation function was included. While **SoftMax** is typically used for classification tasks, it introduces non-linearity in this context. For regression, a linear activation would typically be used, but the choice here can be discussed in terms of model behaviour.
    - The final output layer consists of a single neuron with a linear activation function, providing the predicted closing price.

5. **Model Compilation:**
    - The model was compiled using **Mean Squared Error (MSE)** as the loss function to quantify the difference between predicted and actual prices. The **Adam optimizer** was chosen for its efficiency and adaptability in optimizing the model's learning process.

4.4 Model Training

The model was trained on the training dataset with the following specifications:

- **Epochs:** The model was trained for 15 epochs, allowing it to learn from the data over multiple iterations.
- **Batch Size:** A batch size of 32 was selected to balance training speed and model performance.
- **Validation Split:** 10% of the training data was set aside for validation to monitor the model's performance on unseen data during training.

4.5 Performance Evaluation

To assess the model's performance, the following metrics were used:

- **Mean Squared Error (MSE):** This metric measures the average of the squares of the errors between predicted and actual values, providing a clear indication of the model's accuracy.
- **Mean Absolute Percentage Error (MAPE):** This metric expresses the accuracy as a percentage, allowing for easier interpretation of the prediction's reliability.

Results

5.1 Model Performance Metrics

After training the Long Short-Term Memory (LSTM) model on the Bitcoin price dataset, the model's performance was evaluated using two key metrics: **Mean Squared Error (MSE)** and **Mean Absolute Percentage Error (MAPE)**.

The evaluation results are as follows:

- **Test Loss (MSE):** The model achieved a test loss of **0.000356**.
- **Test MAPE:** The Mean Absolute Percentage Error calculated on the test dataset was **4.34%**.
- **Test Accuracy:** The model achieved an accuracy of **95.66%**.

These metrics indicate that the model performed exceptionally well, with a low MSE suggesting that the predicted prices were very close to the actual closing prices. The MAPE value of 4.34% further highlights the accuracy of the predictions, demonstrating the model's effectiveness in forecasting Bitcoin prices.

5.2 Visualizing Predictions

To better understand the model's performance, the predicted Bitcoin closing prices were compared against the actual prices over the test period. The following key observations were made from the visualization:

1. **Overall Trend Matching:**
    - The predicted prices closely followed the trend of actual prices, demonstrating the model's ability to capture underlying patterns in the time series data.
    - The LSTM model successfully identified periods of price increases and decreases, reflecting the volatility of the Bitcoin market.

2. **Accuracy During Fluctuations:**
    - During certain volatile periods, the model was able to approximate price movements accurately, although some deviations were noted. This behavior is typical in financial time series forecasting, where sudden market changes can lead to prediction errors.

3. **Plotting Results:**

The results were visualized using Matplotlib, as shown in Figure 1. The plot includes:

- **Training Data:** Represented in black, illustrating the historical closing prices used for training the model.

- **Actual Test Data:** Shown in blue, depicting the actual closing prices during the test period.
- **Predicted Test Data:** Illustrated in red, showing the closing prices predicted by the LSTM model.

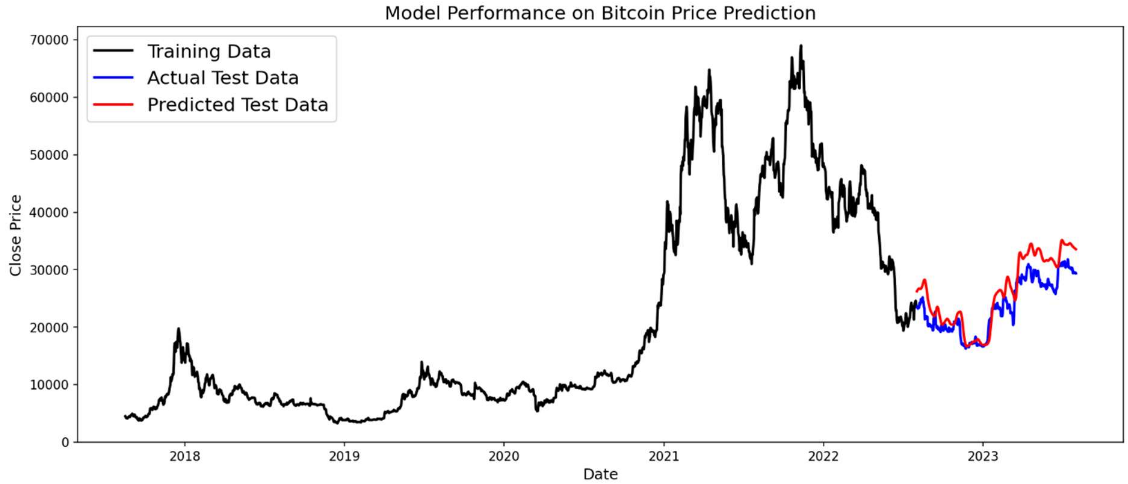

Figure – 5.1: Actual Price vs Predicted Price

5.3 Interpretation of Results

The analysis of the results yields several important insights:

- **Model Robustness:** The LSTM model's architecture, specifically its use of multiple layers and dropout for regularization, contributed to its robustness and ability to generalize well on unseen data. The ability to process sequences of data allowed the model to learn temporal dependencies effectively.
- **Implications for Traders:** The findings highlight the potential of machine learning models, particularly LSTM networks, in predicting Bitcoin prices. Traders and investors could leverage such models for better decision-making and risk management in volatile markets.
- **Limitations and Considerations:** Despite the model's strengths, it is crucial to note that predictions in financial markets can be inherently uncertain. External factors such as market sentiment, regulatory changes, and macroeconomic trends can significantly impact prices and may not be captured fully by the model. Thus, while the model can provide insights, it should be used in conjunction with other analysis tools and market knowledge.

Conclusion

This project successfully implemented a Long Short-Term Memory (LSTM) model to predict Bitcoin closing prices using historical data from 2017 to 2023. The model was trained and evaluated, achieving a test Mean Squared Error (MSE) of 0.000356 and a Mean Absolute Percentage Error (MAPE) of 4.34%. These metrics indicate that the model performed exceptionally well, reflecting its ability to closely approximate the actual Bitcoin prices, with a test accuracy of 95.66%.

The visual analysis of the predicted prices versus actual prices demonstrated that the model effectively captured the trends and fluctuations in the Bitcoin market. This ability to learn from sequential data underscores the potential of LSTM networks in time series forecasting, particularly in the volatile cryptocurrency domain.

The successful implementation and evaluation of the LSTM model highlight the growing relevance of machine learning techniques in financial forecasting. The ability to accurately predict Bitcoin prices based on historical data offers significant implications for traders and investors. By leveraging such models, stakeholders can enhance their decision-making processes and improve risk management strategies in a market characterized by high volatility.

This project contributes to the broader understanding of how advanced neural network architectures can be utilized to forecast prices in the cryptocurrency sector, providing a foundation for further research and application.

6.1 Future Scope

While our model has performed well, there are several ways to enhance it:

- **Adding More Data Features:** Incorporate additional information like trading volume and social media trends to improve predictions.
- **Trying Different Models:** Explore other models such as Gated Recurrent Units (GRU) or Transformers for potentially better results.
- **Analysing Longer Time Periods:** Extend the analysis to longer time frames to assess reliability across various market conditions.
- **Adding Risk Assessment Tools:** Develop tools to evaluate risks associated with predictions, helping traders make informed decisions.

References

1) Sibel Kervanci, M. Fatih Akay, and Eren Ozceylan, "Bitcoin Price Prediction Using LSTM, GRU and Hybrid LSTM-GRU with Bayesian Optimization, Random Search, and Grid Search for the Next Days," *Journal of Industrial and Management Optimization*, vol. 20, no. 2, pp. 570-588, February 2024. doi:10.3934/jimo.2023091. Available: https://doi.org/10.3934/jimo.2023091
2) Vijaya Kanaparthi, "Robustness Evaluation of LSTM-based Deep Learning Models for Bitcoin Price Prediction in the Presence of Random Disturbances," *International Journal of Innovations in Scientific Engineering and Management (IJISME)*, 2024. doi:10.35940/ijisme.B1313.12020224. Available: https://www.ijisme.org/wp-content/uploads/papers/v12i2/B131312020224.pdf
3) N. Latif, J. D. Selvam, M. Kapse, V. Sharma, and V. Mahajan, "Comparative Performance of LSTM and ARIMA for the Short-Term Prediction of Bitcoin Prices," *Australasian Accounting, Business and Finance Journal*, vol. 17, no. 1, pp. 256-276, 2023. doi:10.14453/aabfj.v17i1.15. Available: http://dx.doi.org/10.14453/aabfj.v17i1.15

Appendix

1) **Google Colab Notebook**: The complete implementation of the Bitcoin price prediction project can be accessed through the following link:
   Bitcoin Price Prediction Colab Notebook

2) **Dataset**: The historical Bitcoin price data used for this project is available on Kaggle. You can download it from the following link:
   Bitcoin Price Dataset on Kaggle

3) **GitHub Repository**: The source code, along with additional documentation, can be found in the GitHub repository:
   GitHub Repository for Bitcoin Price Prediction