DASINYA JOURNAL
for engineering and informatics

https://doi.org/10.65542/djei.v2i1.36
Dasinya Journal for Engineering and Informatics. **2026**, *2*, 1.

Article

Bitcoin Price Prediction Using Blockchain Transaction Data and Machine Learning Models

Ronak Hassan [1,*] and Jihan Ahmed [1]

[1] Computer Science Department, University of Duhok, Kurdistan Region, Iraq
* Correspondence: ronak.ahmed.uod@gmail.com

Abstract

In this work, we utilize the blockchain transactions and financial instruments to predict the Bitcoin price using machine learning. We use three models: Light Gradient Boosting Machine (LightGBM), Decision Tree Regressor and Random Forest Regressor applied on a feature set which includes lagged close prices, 14-day Simple Moving Average (SMA), Relative Strength Index (RSI) and daily confirmed Bitcoin transactions. The data is temporally aligned and pre-processed to maintain temporal coherence, as well as for conversational fluency. Through the results assessment by means of RMSE MAE, MAPE and R^2, we can found that Random Forest model has results closer to best performance with values of: 264.81 (RMSE); 175.41(MAE); for MAPE is 0.27% and R^2 equals to 0.9958. Our findings also lend strong support for the effectiveness of simultaneously considering not only blockchain-specific market variables but also traditional financial predictors towards improved model performance and generalization. Our findings underscore the importance of raw blockchain transaction data for predicting cryptocurrency prices, and present a new tool for data-based decision making in decentralized finance.

Keywords: Bitcoin Price Prediction; Blockchain Transaction Data; Machine Learning Models; Random Forest Regressor; Technical Indicators

1. Introduction

Bitcoin's transformation into a unique form of money has also increased its value as compared to traditional commodities like consumer goods and fiat money. Bitcoin's price fluctuates because it takes place on the speculative market. Outside of trading and investing, the act of interacting with crypto itself presents risks beyond losing/gaining money, such as potential losses from settlement appearing from an interface error or network interruption. These issues can not only facilitate, but also emphasize the significance of price forecast in Bitcoin for algorithmic trading, financial analysis and risk management [1].

While traditional currencies are created by central banks and stocks are issued by commercial banks, the network of computers that run the Bitcoin system operate on an as-a-service network. Being the decentralized ledger this ledger is maintained and validated by the network and captures and stores all Bitcoin transactions not yet recorded and thereby could be considered a treasure trove as a big data information source - as it captures the movement, mood and pulsating energy of the Bitcoin-sphere [2]. Classic Predictive systems use a certain amount of statistical quantitative analysis and high complicate math formulas to predict a possible price range for a financial instrument. Signs are helpful, but they don't reveal a great deal about the transactional dynamics of the broader ecosystem being played out on the blockchain itself. Thus, using transaction data on blockchain as leading indicators in prediction models of future transactions production is an efficient way to enhance prediction accuracy [3].

Received: December 17, 2025; Revised: January 8, 2026; Accepted: January 9, 2026; Published: January 12, 2026; Available online: January 12, 2026.

Machine learning (ML) has become increasingly popular in financial time series analysis in recent years. For example, supervised learning models have been broadly employed to describe non-linear structure and complex relationships in data [4]. Of these, decision trees, ensemble methods including Random Forest and boosting methods such as LightGBM have good predictive performance [5]. Structured data processing, the ability to capture nonlinearities and interpretability makes them suitable for use in financial modeling. Nevertheless, there are few comparative studies that feature blockchain transaction data in such models [6].

This paper seeks to fill this gap by proposing a hybrid feature set that combines traditional technical indicators with blockchain transaction metrics such as the number of confirmed transactions per day. We speculate that this composite feature representation is better at representing sequences of driving forces of the Bitcoin price. To this end, we use three models from tree-based machine learning: Light Gradient Boosting Machine (lightGBM), decision tree regressor, and random forest regressor, and conduct a backtesting analysis using a dataset that merges history of the Bitcoin closing price together with the block-chain transaction-level data.

We applied the 14-day Simple Moving Average and Relative Strength Index financial indicators in our test, which are commonly used in stock technical analysis to monitor trends and overbought or oversold conditions. Besides, we engineer lag features in the model in order to retain last low-price points and trends. The inclusion in our analysis of behaviour accompanied by blockchain-transaction data also allows us to measure transactions volume and speed of settlement, and more often than not it goes first to the price. Then, we describe a certain content selection and model introduction process. RMSE, MAE, MAPE and R^2 are the main evaluation metrics used in our experiments. These indicate the generalization and cross-validation performance of the model.

Results from experimentation indicate that Random Forest outperforms in all parameter settings the LightGBM and decision tree with significant difference. It would appear achievable through fine-grained features of the ensemble collection of models to learn and extrapolate to predictable shifts in cryptocurrency values and behavior. And so to make a conclusion of my thesis: This paper is one of the growing papers in the field of machine learning and Blockchain analysis. However, the prediction indicators are generated by employing the true blockchain transaction data, instead of the STP-based market-driven data, for conducting forecast the predictions based on blockchain transaction data pave the way for more reliable forecasts for the price of Bitcoin in the future. And not just investors and traders, but also researchers and developers who are working to produce smarter financial technology in the first place.

For an overall view the structure of the remainder of this paper shall be such. Section 2 present related work and discusses existing methods for predicting Bitcoin prices by employing either traditional financial indicators or blockchain features. Section 3 details the methodology consisting of data collection, preprocessing, feature engineering and building the model. The experimental results on various metrics of the proposed models are reported and discussed in Section 4. Section 5 eventually closes the study by highlighting the main results and practical implications of the findings and by proposing avenues for future research.

2. Related works

Recent studies have applied machine learning and deep learning models to Bitcoin price prediction using features derived from blockchain data. Li and Wu developed a k-order transaction graph model to automatically learn structural transaction patterns, achieving improved predictive performance compared to hand-crafted TF-IDF features [7]. Wei et al. introduced DLForecast, employing temporal graph embeddings to model evolving Bitcoin account interactions, showing enhanced accuracy in short-term forecasting [8].

Palaiokrassas et al., in a systematic mapping study found that on-chain predictors like transaction volume, network activity, and fee rates significantly enhance financial prediction models [9]. In a different setting of feature engineering, Cohen and Aiche applied multi-network ensembles with traditional technical indicators which result in high returns, stressing the significance of normalization and trend-based generation of features [10]. Similarly, Hafid et al. utilized XGBoost including MACD, RSI, Bollinger Bands and other selected indicators for more than 92% directional-prediction accuracy [11].

From an economics modeling point of view, Erfanian et al. a comprehensive list of macro-and micro-economic, technical and blockchain variables under the frameworks of SVR and ensemble, concluding that SVR outperformed other models in short-and long-term prediction accuracy [12]. Alsini et al. detected buy signals with trees of bags learned with indicator-derived feature sets, which resulted on a more reliable decision support system [13].

Investigations by Bistarelli et al. demonstrated the utility of incorporating external factors (such as investor sentiment, etc.) with technical indicators for better predictability [14]. An earlier influential study by Madan et al. demonstrated that hybrid features using social sentiment and network metrics can achieve accuracy over 98% in daily movement prediction [15].

Newer models concentrate on features, such as enhancement and noise reduction. Mousa et al. incorporated wavelet denoising and hashrate-based features with a stacked deep learning architecture, which lowered MAPE down to around 0.58% in daily forecasts [16]. For blockchain fraud and behavioral analysis, Elmougy et al. presented a graph-based financial forensics dataset Elliptic++ with methods that address the issue of transaction-level profile extraction [17]. In the end, Sebastião and Godinho compared Random Forest and LSTM for Bitcoin prediction task showing that model selection as well as train-test split are important to error in predictions [18].

In concert these recent works consolidate several themes: First, on-chain blockchain related metrics (e.g., transaction volume, graph structure, hashrate and network activity) are available to help improve prediction accuracy; Second. technical indicators such as RSI, MACD, SMA and Bollinger Bands remain strong predictors when combined with ML models; and third, careful feature engineering and preprocessing – including graph embeddings, denoising techniques, feature selection normalization and time-lag windowing is necessary for model robustness. We extend this literature by combining verified transaction amounts with lagged price features and standard technical indicators examined for tree-based models, which have a well-defined train-test distinction.

3. Methodology

This section describes the overall approach from market and blockchain transaction metrics analysis to forecasting Bitcoin price. The methodology consists of data integration, preprocessing and feature engineering selected features based on domain-specific indicators, model construction by means of machine learning regressors, and validation through standard statistical metrics. Figure 1 illustrate our methodology.

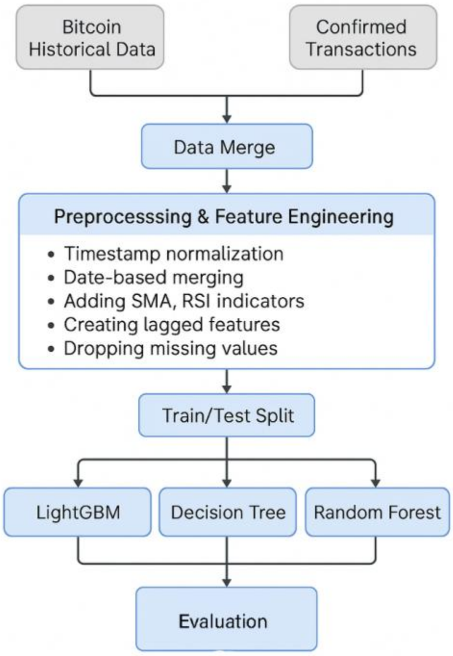
Bitcoin Historical Data
Confirmed Transactions
Data Merge
Preprocesssing & Feature Engineering
• Timestamp normalization
• Date-based merging
• Adding SMA, RSI indicators
• Creating lagged features
• Dropping missing values
Train/Test Split
LightGBM
Decision Tree
Random Forest
Evaluation

Figure 1. Flowchart for the proposed approach.

3.1. Data Sources

This study was based on two different datasets. The first dataset named Bitcoin Historical Data, gives timestamped data about the Bitcoin market. Where timestamp is a unix timestamp, close is the price in USD (closing), volume represents the currency and marketCap refers to market cap. These features are common in financial time series analysis and essential to measure the sentiment of the market and capitalization evolution. The second data set: Confirmed Transactions, is another blockchain level variable containing daily counts of confirmed_tx which is the number of Bitcoin transactions that have been confirmed on the blockchain. This measure is often treated as a proxy for network load and can indicate when levels of user activity are higher or the network is more congested.

3.2. Timestamp Harmonization and Dataset Merging

To integrate the two datasets for joint analysis, they were combined using their timestamp columns. The original time formats weren't always precisely equal and didn't always have the same precision or possible timezone offsets. To align these, timestamps were converted into a shared datetime format and stripped of time-of-day information (length discounting for Y-M) so that all datetimes would have daily granularity (YYYY-MM-DD). This operation allows the price and transaction data to be matched on a day granularity. The two datasets were merged by an inner join on the column of timestamp after normalization. This ensure that only dates found in both sources were kept in order not to propagate null values which could then biased the learning.

Let D_p represent the date set in the price dataset and D_t the date set in the transaction dataset. The merged dataset D is defined as:

$$D = D_p \cap D_t = \{ t \in Date \mid t \in D_p \wedge t \in D_t \} \quad (1)$$

This results in a combined dataset that includes: daily close prices, volume, marketCap, and confirmed_tx—all synchronized by date.

3.3. Feature Engineering

In order to enhance the predictability, some additional features were created from the combined ones. These can be lagging indicators, such as moving averages or momentum-based ones that are often seen in financial predictions. One important issue while modelling time series is the presence of autocorrelation in the data, the dependency of the variable on its past values. Thus, five lags of the close variable were derived, each representing closing prices one to five days prior to each observation. Formally, the lagged feature for day t with lag k is defined as:

$$CloseLage_k\,(t) = close(t-k), k \in \{1,2,3,4,5\} \quad (2)$$

These lag features encapsulate short-term historical trends and are particularly useful for tree-based learning algorithms which partition feature space based on value thresholds. In addition to lags, two domain-specific indicators from technical analysis were computed: the 14-day Simple Moving Average (SMA) and the 14-day Relative Strength Index (RSI).

The 14-day SMA is defined as:

$$SMA_{14}(t) = \frac{1}{14}\sum_{i=0}^{13} close(t-i) \quad (3)$$

This metric smooths price volatility and captures trend direction, assisting the model in distinguishing between bullish and bearish phases.

The 14-day RSI is a momentum oscillator that quantifies the magnitude of recent price changes. The RSI for day t is calculated as:

$$RSI_{14}(t) = 100 - \left(\frac{100}{1+RS_t}\right) \quad (4)$$

Where RS_t is the ratio of average gains to average losses over the past 14 days:

$$RS_t = \frac{\bar{G}_{14}}{\bar{L}_{14}}, with\ \bar{G}_{14} = \frac{1}{14}\sum_{i=1}^{14} G_{t-1}\,, \bar{L}_{14} = \frac{1}{14}\sum_{i=1}^{14} L_{t-1} \quad (5)$$

Here, $G_t = \max(\Delta P_t, 0)\,, L_t = \max(-\Delta P_t, 0)$, and $\Delta P_t = close(t) - close(t-1)$.

After all features were constructed, the dataset included the following variables:

Lag features: close_lag_1 to close_lag_5

Original features: volume, marketCap, confirmed_tx

Technical indicators: SMA_14, RSI_14

Rows containing any missing values, typically introduced by rolling calculations or lag operations, were removed. This ensured a clean input matrix for machine learning models.

The final feature matrix $\mathbf{X} \in \mathbb{R}^{n*m}$, where n is the number of valid observations and $m = 10$ is the number of input variables, was used to predict the target variable $y = close(t)$, the actual Bitcoin closing price on day t.

3.4. Model Development and Training

Three supervised machine learning models were developed and compared: LightGBM, Decision Tree, and Random Forest. Each model was selected based on its ability to handle tabular data, support non-linear relationships, and maintain interpretability or ensemble learning advantages.

LightGBM is a very fast, high-performance gradient boosting framework based on decision trees. It was set with 200 estimators, a learning rate 0.02, and a maximum depth 8. LightGBM model is built in a forward manner, with each new tree attempting to correct the mistakes of previous trees. This is a very fast and efficient method, particularly if the data has a large number of cases and/or predictors. Nevertheless, its performance is highly sensitive on hyperparameter tuning, and gradient boosting mechanism in the basis of its boosting approach can be overfitted on noisy data.

The second model, Decision Tree Regressor, was fitted with a maximum depth of 8 aa the depth of 8 was able to prevent overfitting better and generalize as well. This model works by learning Decision Rules hierarchy based from the input features in order to predict. Being a single-tree model, it is easily interpretable and visualizable and is suitable for knowing which of the variables are the most influential on the target price. However, it tends to have large variance and lack robustness in particular with more complex places and non-linear interactions in data.

The third model, Random Forest Regressor, is an ensemble of 100 decision trees fitted on random subsets of the training data with the bootstrap aggregation bagging). The maximum depth of each tree was applied to eight. Since Random Forests average across many trees, they are good at mitigating overfitting and at giving you more accurate predictions. These are potent tools for financial forecasting problems, which often involve complex or noisy relationships between the variables. In this paper, Random Forest always outperformed in terms of evaluation indicators.

We trained both models with the same training set and same feature matrix and check their prediction on the held-out testing set. The training step was done by learning each model from the historical features and, tuning of the internal splits in order to minimize prediction error observation in the target variable.

3.5. Evaluation Metrics

In order to evaluate and compare the performance of the models, four well-known evaluation metrics were considered: RMSE, MAE, MAPE and R^2 (R-squared). RMSE also tells us how much error there is on average, and RMSE penalizes larger errors more than does MAE. The MAE gives a more intuitive average of the absolute predictions and differences, and yields a more meaningful metric for descriptive forecast purposes. MAPE is error as a percentage of the actual, so it can be compared across scales. Lastly, R^2 measures the ratio of the variance explained by the model and the target variable, the closer to 1 the better.

All these metrics offer complementary characteristics of the behavior of the model: RMSE and MAE say something on the magnitude of the error, MAPE accounts for relative errors to estimate R^2 is a general indicator of adjustment about that line. We evaluated all models against these criteria to gauge the ability of a model in predicting Bitcoin price using aggregated features.

4. Results and Discussion

This section presents a comprehensive analysis of the experimental outcomes. The three selected machine learning models—LightGBM, Decision Tree, and Random Forest—were trained on the same dataset derived from merged Bitcoin price and blockchain transaction data. Each model was evaluated using four standard performance metrics: RMSE, MAE, MAPE, and R^2.

4.1. Quantitative Performance Comparison

Table 1 provides quantitative performance comparison for all models and metrics used in this research.

Table 1 Quantitative Performance Comparison

Model	RMSE	MAE	MAPE	R^2 Score
LightGBM	**482.56**	**345.73**	**0.54%**	**0.9862**
Decision Tree	**357.85**	**246.51**	**0.32%**	**0.9924**
Random Forest	**264.81**	**175.41**	**0.27%**	**0.9958**

RMSE (Root Mean Squared Error): The Random Forest model gave the best prediction for RMSE and its RMSE values were much lower than that of RForest and Decision Tree models with only 264.81 USD error rate, in contrast to 357.85 USD and 482.56 USD by Decision Tree and LightGBM respectively. A smaller RMSE corresponds to better general model fit and less variability in prediction errors.

MAE (Mean Absolute Error) MAE, which represents the average absolute difference between predicted and actual values, was minimal for Random Forest (175.41 USD). This also supports its robustness to predictitude in the range of actual bitcoin prices. LightGBM also exhibited highest MAE among others, suggesting it was less accurate.

MAPE (Mean Absolute Percentage Error): MAPEs were remarkably low— well below 1% for all three models— which suggested that the models were forecasting very accurately in proportion to the range of Bitcoin prices ($56,000-$72,000). Remarkably, the Random Forest with a MAPE of 0.27% depicts that the model is capable of predicting percentage value accurately.

R^2 Score: The proportion of the variance in the dependent variable that is predictable from the independent variables. Random Forest remained on top with $R^2 = 0.9958$, which implies it explained 99.58% of the variance in closing price. The Decision Tree came close, with LightGBM (which is a gradient boosting algorithm) slightly lower compared in this respect.

4.2. Visual Inspection of Prediction Curves

For each model, we plotted the predicted values against the actual closing prices. The plots revealed the following:

LightGBM: As shown in Figure 2, the prediction line followed the actual values with noticeable fluctuations at the beginning and some lag in capturing local peaks. Its predictions were generally smoother but lacked responsiveness during rapid price movements.

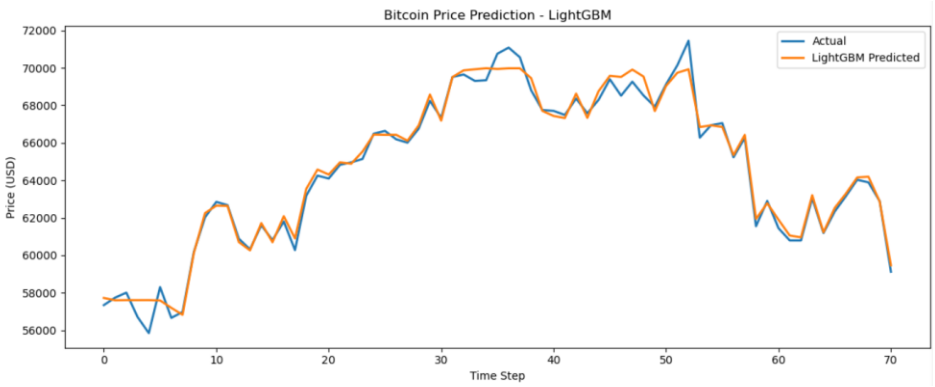

Figure 2. Visual Inspection of Prediction Curves of LightGBM.

Decision Tree: As shown in Figure 3, the model captured the trend reasonably well but exhibited slight underfitting around sharp uptrends or downtrends. Although the overall directionality was consistent, precision was lost in volatile segments.

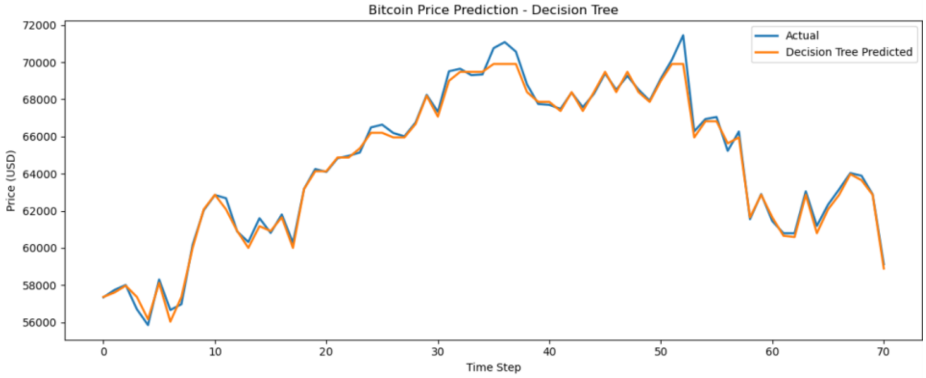

Figure 3. Visual Inspection of Prediction Curves of Decision Tree.

Random Forest: As shown in Figure 4, provided the most visually accurate fit, tracking both general trends and short-term fluctuations with minimal deviation. It effectively modeled non-linear patterns and responded well to abrupt changes in the price curve.

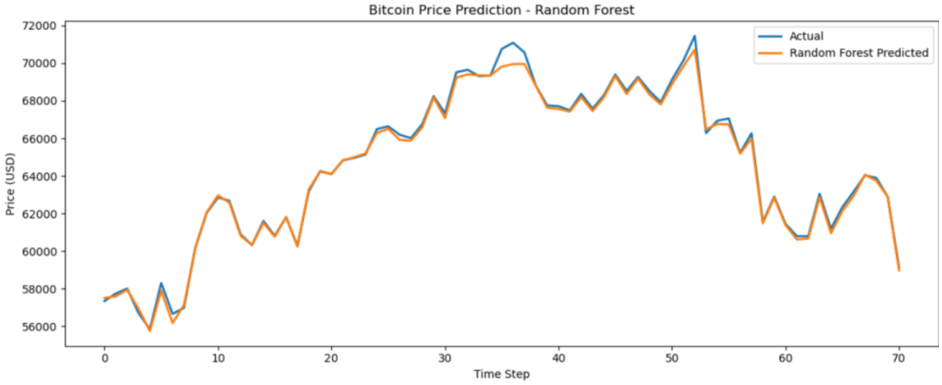

Figure 4. Visual Inspection of Prediction Curves of Random Forest.

4.3. Model Behavior and Interpretability

Though the Decision Tree is very interpretable and does little computation, it cannot capture complex patterns with a single-tree structure. Depending on its depth, it risks either overfitting or underfitting, but here with a max_depth=8 proved to be just about the sweet spot.

It is possible that LightGBM, which is designed to work well on large datasets, does not harness the comparatively small spatio-temporal nature of this dataset. It is a feature-sensitive learner and more in-depth tuning may be needed to fully exploit its capacities.

Random Forest, benefiting from ensemble averaging across multiple deep trees, mitigated variance and bias effectively. It accounted for non-linearity both, which made it more resilient for our time series problem with the engineered lagged prices and RSI/SMA features.

4.4. Role of Feature Engineering

Technical indicators (SMA_14 and RSI_14) along with lag scores were the dominant predictors of the models:

Lag Features (close_lag_1 to close_lag_5): The temporal aspect of the data can be recorded by the model.

RSI & SMA: As trend and momentum indicators were helping to provide useful information for models to make better predictions.

confirmed_tx -As a blood-chain-native metric, it added a novel, non-price signal that was discovered to improve forecasting particularly in periods of high transaction volume.

4.5. Implications and Practical Relevance

Our findings suggest that somewhat small, domain-specific models are at least a good starting point for predicting the value of cryptocurrencies. The success of Random Forest has also suggested that for short-horizon forecasts, we may not require sophisticated neural networks with care feature engineering. From the perspective as a crypto-financial practitioner, these findings provide a proof of concept of incorporating transaction-based blockchain metrics for modeling.

5. Conclusions

This study confirms the efficacy of utilizing blockchain transaction data, that is truly daily confirmed transactions in machine learning models, for predicting bitcoin price. It is in this setting that industry knowledge and expertise come to the fore and where we leveraged blockchain-native features, technical indicators, along with lags of historical prices to form a full-fledged feature set pertaining to market behaviour and network activity. Across all models considered, Random Forest performed best in terms of evaluation scores and outperformed LightGBM and Decision Tree for each metric, justifying the effectiveness of ensemble learning on time series prediction. The low RMSE, MAE and MAPE values and the high R^2 value indicate that the model is able to predict accurately. Our results reinforce the importance of tailored feature engineering and the potential of transaction data as a powerful forecasting signal. Future work will investigate longer forecasting horizons, real-time deployment, and the inclusion of additional blockchain metrics such as mempool size, average transaction fees, and network hashrate to further enhance forecasting performance.

Author Contributions: Both authors contributed to the conceptualization, methodology, analysis, writing, and revision of the manuscript. Both authors have read and approved the final version.

Funding: No external funding was received to support this research.

Data Availability Statement: The data used in this study are publicly available and can be accessed from official blockchain and financial data repositories. Further details are available upon request.

Acknowledgments: The authors would like to thank all institutions and individuals who supported the completion of this work.

Conflicts of Interest: The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Abbreviations

The following abbreviations are used in this manuscript:

AI	**Artificial Intelligence**
BTC	**Bitcoin**
DL	**Deep Learning**
DT	**Decision Tree**
GBM	**Gradient Boosting Machine**
LightGBM	**Light Gradient Boosting Machine**
LSTM	**Long Short-Term Memory**
MAE	**Mean Absolute Error**
MAPE	**Mean Absolute Percentage Error**
MACD	**Moving Average Convergence Divergence**
ML	**Machine Learning**
RF	**Random Forest**
RMSE	**Root Mean Squared Error**
RSI	**Relative Strength Index**
R^2	**Coefficient of Determination (R-squared)**
SMA	**Simple Moving Average**
SVR	**Support Vector Regression**
TF-IDF	**Term Frequency–Inverse Document Frequency**
USD	**United States Dollar**
UTC	**Coordinated Universal Time**
XGBoost	**Extreme Gradient Boosting**

References

1. N. Tripathy, S. Hota, D. Mishra, P. Satapathy, and S. K. Nayak, “Empirical forecasting analysis of bitcoin prices: A comparison of machine learning, deep learning, and ensemble learning models,” International journal of electrical and computer engineering systems, vol. 15, no. 1, pp. 21–29, 2024.
2. O. M. Ahmed, L. M. Haji, A. M. Ahmed, and N. M. Salih, “Bitcoin price prediction using the hybrid convolutional recurrent model architecture,” Engineering, Technology & Applied Science Research, vol. 13, no. 5, pp. 11735–11738, 2023.
3. Y. Zhang, R. Garg, L. L. Golden, P. L. Brockett, and A. Sharma, “Segmenting Bitcoin transactions for price movement prediction,” Journal of Risk and Financial Management, vol. 17, no. 3, p. 128, 2024.
4. O. M. Mustafa, O. M. Ahmed, and V. A. Saeed, “Comparative analysis of decision tree algorithms using gini and entropy criteria on the forest covertypes dataset,” in The International Conference on Innovations in Computing Research, Springer, 2024, pp. 185–193.
5. O. A. Al-Zakhali, S. Zeebaree, and S. Askar, “Comparative analysis of XGBoost performance for text classification with CPU parallel and non-parallel processing,” The Indonesian Journal of Computer Science, vol. 13, no. 2, 2024.

6. V. U. Pugliese, R. D. Costa, and C. M. Hirata, “Comparative evaluation of the supervised machine learning classification methods and the concept drift detection methods in the financial business problems,” in International Conference on Enterprise Information Systems, Springer, 2020, pp. 268–292.
7. X. Li and W. Wu, “A blockchain transaction graph based machine learning method for bitcoin price prediction,” arXiv preprint arXiv:2008.09667, 2020.
8. W. Wei, Q. Zhang, and L. Liu, “Bitcoin transaction forecasting with deep network representation learning,” IEEE Trans Emerg Top Comput, vol. 9, no. 3, pp. 1359–1371, 2020.
9. G. Palaiokrassas, S. Bouraga, and L. Tassiulas, “Machine learning on blockchain data: A systematic mapping study,” Available at SSRN 4530479, 2023.
10. G. Cohen and A. Aiche, “Predicting the Bitcoin’s price using AI,” Front Artif Intell, vol. 8, p. 1519805, 2025.
11. A. Hafid, M. Rahouti, L. Kong, M. Ebrahim, and M. A. Serhani, “Predicting Bitcoin market trends with enhanced technical indicator integration and classification models,” arXiv preprint arXiv:2410.06935, 2024.
12. S. Erfanian, Y. Zhou, A. Razzaq, A. Abbas, A. A. Safeer, and T. Li, “Predicting bitcoin (BTC) price in the context of economic theories: A machine learning approach,” Entropy, vol. 24, no. 10, p. 1487, 2022.
13. R. Alsini et al., “Forecasting cryptocurrency’s buy signal with a bagged tree learning approach to enhance purchase decisions,” Front Big Data, vol. 7, p. 1369895, 2024.
14. S. Bistarelli, F. Santini, and L. M. Tutino, “A Short Survey on Bitcoin Price Prediction,” in CEUR WORKSHOP PROCEEDINGS, CEUR-WS, 2024.
15. S. Ranjan, P. Kayal, and M. Saraf, “Bitcoin price prediction: A machine learning sample dimension approach,” Comput Econ, vol. 61, no. 4, pp. 1617–1636, 2023.
16. R. Mousa, M. Afrookhteh, H. Khaloo, A. A. Bengari, and G. Heidary, “Forecasting of Bitcoin Prices Using Hashrate Features: Wavelet and Deep Stacking Approach,” arXiv preprint arXiv:2501.13136, 2025.
17. Y. Elmougy and L. Liu, “Demystifying fraudulent transactions and illicit nodes in the bitcoin network for financial forensics,” in Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining, 2023, pp. 3979–3990.
18. Z. Z. Lua, C. K. Seow, C. B. Chan, Y. Cai, and Q. Cao, “Automated bitcoin trading dApp using price prediction from a deep learning model,” 2025.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of Dasinya Journal and/or the editor(s). Dasinya Journal and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.