Comparative Analysis of Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) Models for Bitcoin Price Prediction Using Closing Price, Volume, and Technical Indicators

Haikal Fikri [1], Noper Ardi [2]

Software Engineering Technology, Department of Informatics Engineering, Politeknik Negeri Batam

fikri.haikal1254@gmail.com[1]

Article history:

Received …

Revised …

Accepted …

Keywords:

Bitcoin price prediction[1],
LSTM[2],
GRU[3],
Technical indicators[4],
Deep learning[5]

ABSTRACT

Blockchain technology has turned Bitcoin into one of the most heavily traded digital assets worldwide, but its sharp price swings make investment decisions difficult and call for a model that can capture complex, non-linear behavior. This study compares two Recurrent Neural Network (RNN) variants, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), for Bitcoin price prediction using a combined input of closing price, trading volume, and three technical indicators, Moving Average (MA), Relative Strength Index (RSI), and Moving Average Convergence Divergence (MACD). The workflow consists of gathering daily Bitcoin price data, deriving the technical indicators, preprocessing the resulting dataset, training the LSTM and GRU networks, and scoring each model with Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). On the held-out test set, GRU obtained an RMSE of $12,194.34, an MAE of $10,125.90, a MAPE of 10.0629%, and 89.9371% accuracy, whereas LSTM obtained an RMSE of $26,401.57, an MAE of $23,476.99, a MAPE of 23.9027%, and 76.0973% accuracy. Both networks were able to track the broad price trend, yet GRU clearly outperformed LSTM under this multivariate setup, a gap attributed to its leaner gating structure and more efficient training. These findings position the resulting model as a possible building block for decision-support tools or automated-trading systems in cryptocurrency markets, adding to the growing body of deep-learning approaches for forecasting dynamic digital financial markets.

I. INTRODUCTION

Blockchain technology has advanced quickly enough to push Bitcoin into the position of one of the most heavily traded digital assets anywhere, and the currency's price swings are large enough to create real opportunity for investors while exposing them to equally real risk. Because of that risk, reliable price-prediction techniques have become an important ingredient of sound decision-making in cryptocurrency trading.

Evidence from the literature consistently points to machine learning and deep learning as more capable than classical statistical techniques when it comes to capturing the complex, non-linear, and dynamic behavior of cryptocurrency markets [6], [15]. Two Recurrent Neural Network (RNN) variants stand out as the most common choices for this kind of time-series problem: Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU). LSTM tackles the vanishing-gradient problem through a dedicated cell state coupled with three gates, while GRU strips this down to a leaner design with fewer parameters that still manages to retain long-term dependencies [6]. Reported comparisons between the two are mixed, with the better performer often depending heavily on the dataset and the particular set of features being used [15].

LSTM and GRU have already moved beyond theory into concrete cryptocurrency price-prediction work. Dutta et al. [6], for instance, built a GRU-based Bitcoin prediction model that surpassed several widely used benchmarks, while Lee [15] paired Attention-LSTM and Attention-GRU architectures with technical indicators on Bitcoin data and obtained better trend-prediction accuracy as a result. What these studies have in common, however, is a tendency to rely on a single input variable, usually closing price alone, which leaves considerable room to bring in richer combinations of technical and volume-based features and to weigh the two architectures against each other more thoroughly.

Closing price is not the only useful signal: technical indicators such as Moving Average, RSI, and MACD have also been shown to sharpen a model's ability to pick up cryptocurrency price trends [15]. Kang et al. [9] found that folding trading volume in alongside historical price data adds useful information about market pressure and lifts the performance of GRU-based cryptocurrency prediction, while Wang et al. [19] reported that pairing technical indicators with historical price data strengthens trendpattern recognition and went on to build a stacking-ensemble model combining LSTM and GRU that beat single-model baselines on prediction accuracy. Htay et al. [8], working from a different angle, proposed a multivariate deep-learning approach that merges historical data with social-media sentiment and observed performance gains during volatile periods. Despite this body of work, studies that directly pit LSTM against GRU while jointly combining price, technical-indicator, and trading-volume variables remain scarce, leaving a research gap that is especially relevant given how volatile the Bitcoin market tends to be.

Looking past recurrent networks alone, the field has also branched into related directions: hybrid CNN-LSTM models built for interpretable forecasting [2], reinforcement-learning-based trading systems [3], [20], encoder–decoder LSTM/GRU architectures [5], Bayesian-LSTM forecasting frameworks [18], and multi-asset comparisons spanning Bitcoin, Litecoin, and Ethereum [10]. Taken together, these developments make a strong case for first running a systematic, head-to-head comparison of LSTM and GRU on a richly engineered multivariate feature set before moving on to more elaborate hybrid or multi-asset architectures.

On the practical side, a dependable Bitcoin price-prediction model could assist both retail and institutional traders in shaping their trading strategies, feed into the development of automated-trading systems, and offer a more objective alternative to manual technical analysis.

Taking these points together, this study sets out to compare how LSTM and GRU perform when closing price, trading volume, and technical indicators are used jointly as predictive variables, with the aim of offering a more accurate and comprehensive prediction model and adding a fresh data point to the LSTM-versus-GRU literature in a market as fluctuating as cryptocurrency. Two research questions guide the work: (1) how well do LSTM and GRU predict Bitcoin prices when closing price, volume, and technical indicators are combined as inputs, and (2) which of the two architectures comes out ahead under that combination of features. In line with these questions, the study aims to build and implement both LSTM and GRU models for Bitcoin price prediction from closing price, volume, and technical indicators, and then to compare their performance so that their relative strengths can be identified and a recommendation made for the model best suited to supporting cryptocurrency trading strategies. The scope is kept deliberately bounded: only daily closing price and volume data together with the MA, RSI, and MACD indicators are used; the comparison covers solely the LSTM and GRU architectures, leaving out hybrid models or other architectures such as Transformers; performance is judged using the standard RMSE, MAE, and MAPE metrics; and the focus throughout stays on price prediction itself rather than on investment strategy or risk-management design.

II. METHOD

A. Research Design

This research follows a quantitative, experimental research-and-development (R&D) approach built around developing and comparatively evaluating two deep-learning models, LSTM and GRU, for Bitcoin price prediction. The goal is not to ship a commercially deployed prediction system but to run a controlled comparative experiment in which both models are trained and evaluated under matching conditions, so that whatever performance gap shows up can be traced back to the intrinsic character of each recurrent cell type rather than to differences in setup. Fig. 1 lays out the overall research stages, starting from data collection and ending with result analysis and conclusion.

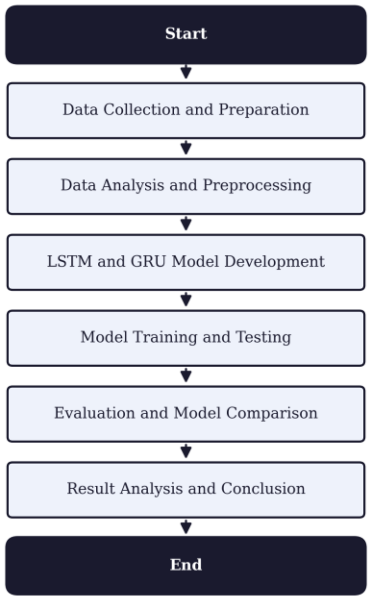

Fig. 1. Stages of the research process.

B. Materials and Tools

The core material for this study is historical daily Bitcoin (BTC-USD) price data pulled from Yahoo Finance via the yfinance library, covering date, closing price, and trading volume in United States dollars. Closing price alone, however, is not the whole picture, so three technical indicators derived from that price series are added to give the model a richer read on market behavior: Moving Average (MA) for gauging short- and long-term trend direction, Relative Strength Index (RSI) for flagging overbought or oversold conditions, and Moving Average Convergence Divergence (MACD) for assessing momentum strength and possible trend reversals.

Every experiment ran on a personal computer fitted with a 12th-generation Intel Core i5-12500 processor (3.00 GHz), 16 GB of DDR4 RAM, an integrated graphics processor, a 477 GB SSD, and Windows 10 Pro 64-bit. On the software side, Python 3.10 or later served as the main programming language; TensorFlow and Keras handled the construction and training of the LSTM and GRU networks; Pandas and NumPy took care of data handling and numerical computation; the ta (Technical Analysis) library computed the technical indicators; Scikit-learn managed data splitting, normalization, and evaluation-metric computation; Matplotlib and Seaborn produced the visualizations; and Streamlit powered an interactive web interface for presenting the prediction results. Because every tool in this stack is open-source, the setup should be straightforward to reproduce on other computing environments.

C. Mathematical Model of LSTM

The LSTM architecture is designed to capture long-term dependencies in the Bitcoin price time series through a cell-state pathway together with three controlling gates: a forget gate (f_t), an input gate (i_t), and an output gate (o_t). The governing equations are as follows [7], [15], [17]:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (1)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (3)$$

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \quad (4)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t \times \tanh(C_t) \quad (6)$$

Here σ stands for the sigmoid activation function, W and b denote weight matrices and bias vectors, and the input vector at time step t gathers the feature combination this study relies on, x_t = [Pt, MAt, RSIt, MACDt], where Pt is the closing price.

D. Mathematical Model of GRU

Following Dutta et al. [6], GRU simplifies the recurrent cell by replacing the three LSTM gates and separate cell state with two gates, an update gate (z_t) and a reset gate (r_t), as formulated below:

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \quad (7)$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \quad (8)$$

$$\tilde{h}_t = \tanh(W_h \cdot [r_t \times h_{t-1}, x_t]) \quad (9)$$

$$h_t = (1 - z_t) \times h_{t-1} + z_t \times \tilde{h}_t \quad (10)$$

Just as in the LSTM case, the input vector here combines closing price, volume, and technical indicators, xt = [Pt, MAt, RSIt, MACDt]. Folding the cell state into the hidden state and getting by with only two gates leaves GRU with noticeably fewer trainable parameters than LSTM, which should translate into a more computationally efficient training process.

E. Dataset and Feature Engineering

1) Data collection: An automated download retrieved daily BTC-USD data covering January 1, 2015 through May 5, 2026, for a total of 4,140 initial rows across six columns: Date, Open, High, Low, Close, and Volume. This 11-year window was chosen on purpose so that it would sweep across Bitcoin's major market regimes, the pre-boom years (2015–2016), the first extreme price surge (2017), the prolonged correction (2018–2019), the largest bull market on record (2020–2021), the bear market that followed (2022), and the recovery after the fourth halving (2024), giving the models a representative mix of historical price-movement patterns to learn from.

2) Technical-indicator computation: The ta library was used to compute three Simple Moving Averages, MA7, MA21, and MA50, covering short-, medium-, and long-term trends respectively, along with a 14-day RSI for momentum and overbought/oversold conditions, and a MACD calculation (fast period 12, slow period 26, signal period 9) that yields three components: MACD, MACD_Signal, and MACD_Hist. The date column also supplied two temporal features, Day_of_Week (0–6) and Month (1–12), included to account for any seasonal effects that might be present.

3) Feature selection: Rows carrying missing values from the rolling-window calculations, the 50-day moving average being the main culprit, were dropped, leaving 4,094 valid rows. Eleven input features were then selected from the remaining columns for use by both models, listed in Table I.

TABLE I. INPUT FEATURE SET USED BY THE LSTM AND GRU MODELS

No.	Feature	Description
1	Close	Daily Bitcoin closing price (USD); also the prediction target
2	Volume	Daily Bitcoin trading volume
3	MA7	7-day simple moving average
4	MA21	21-day simple moving average
5	MA50	50-day simple moving average
6	RSI	14-day Relative Strength Index (range 0–100)
7	MACD	Difference between the 12-day and 26-day EMA
8	MACD_Signal	9-day EMA of the MACD line
9	MACD_Hist	Difference between MACD and the signal line
10	Day_of_Week	Day of the week (0 = Monday, 6 = Sunday)
11	Month	Calendar month (1–12)

F. Data Preprocessing

1) Normalization: Min-Max normalization was applied to all 11 input features so that every value would land within the range [0, 1], following

$$X' = (X - X_{min}) / (X_{max} - X_{min}) \quad (11)$$

Two MinMaxScaler instances were kept separate for this purpose: one covering the 11 input features and a dedicated one for the Close target, so that scaled predictions could be inverse-transformed back to their original USD scale afterward. Both scalers were fit using only the training portion of the data and then applied to the dataset as a whole, a precaution that keeps information from the validation and test sets out of the normalization parameters and avoids data leakage.

2) Time windowing: Because LSTM and GRU require three-dimensional input (samples, timesteps, features), the normalized data were converted into sliding windows using a lookback of 60 days; that is, each input sample consists of the preceding 60 days of all 11 features, with the target being the normalized closing price on day 61. A 60-day lookback was chosen because two calendar months provide a financially meaningful historical context for learning short- to medium-term trend patterns without imposing excessive computational cost.

3) Dataset split: The resulting sequences were split chronologically, without shuffling, into training, validation, and test subsets with a 70:15:15 ratio, as summarized in Table II. Splitting the data from oldest to newest ensures that each model is trained on historical data and evaluated on more recent, unseen data, mirroring a realistic forecasting scenario.

TABLE II. DATASET SPLIT

Subset	Proportion	Samples	Date range
Training	70%	2,821	2015-04-20 to 2023-01-08
Validation	15%	604	2023-01-09 to 2024-09-03
Test	15%	606	2024-09-04 to 2026-05-02

Because the technical indicators used in this study are strictly backward-looking, that is, the value on day T depends only on data from preceding days, computing them prior to splitting does not introduce temporal data leakage. For example, the 50-day moving average on a given date is calculated solely from the 50 days that chronologically precede it, and the same applies to RSI and all MACD components.

G. Model Architecture and Hyperparameters

To ensure that any performance difference reflects the intrinsic characteristics of each recurrent cell rather than configuration differences, both models were built with an identical, symmetric, two-layer stacked architecture using the TensorFlow/Keras framework. The first recurrent layer (GRU or LSTM) uses 64 units with return_sequences=True so that the full sequential output is passed to the second recurrent layer, which uses 32 units with return_sequences=False, since it is followed directly by a dense output stage that requires two-dimensional input. Each recurrent layer is followed by a dropout layer with a rate of 0.2 for regularization. A dense layer with 16 units and ReLU activation follows the recurrent stack, and a final dense layer with a single linear unit produces the price prediction. Each model is compiled with the Adam optimizer (learning rate = 0.001), the Mean Squared Error (MSE) loss function, and MAE as the monitored metric. The complete hyperparameter configuration is summarized in Table III.

TABLE III. HYPERPARAMETER CONFIGURATION FOR GRU AND LSTM

Parameter	GRU	LSTM
Units (Layer 1)	64	64
Units (Layer 2)	32	32
Dropout rate	0.2	0.2
Dense layer	16 units, ReLU	16 units, ReLU
Output layer	1 unit, linear	1 unit, linear
Optimizer	Adam	Adam
Initial learning rate	0.001	0.001
Loss function	MSE	MSE
Monitored metric	MAE	MAE
Batch size	32	32
Max. epochs	200	200
Lookback window	60 days	60 days
Input features	11	11
Shuffle	False	False

Training was further controlled with three identical callback mechanisms for both models. EarlyStopping monitored the validation loss with a patience of 20 epochs and a minimum delta of 0.0001, restoring the best weights once training stopped. ModelCheckpoint automatically saved the best-performing model whenever the validation loss improved. ReduceLROnPlateau reduced the learning rate by a factor of 0.5 when validation loss stagnated, allowing for finer convergence in later epochs. Both models were trained for up to 200 epochs with a batch size of 32 and shuffle=False to preserve chronological order, although EarlyStopping halted training earlier once no further improvement in validation loss was observed.

H. Evaluation Metrics

Model performance was assessed using four quantitative metrics computed after inverse-transforming the predictions back to their original USD scale: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and accuracy, derived as the complement of MAPE.

$$RMSE = \sqrt{ (1/n) \cdot \Sigma (y_i - \hat{y}_i)^2 } \quad (12)$$

$$MAE = (1/n) \cdot \Sigma \, |y_i - \hat{y}_i| \quad (13)$$

$$MAPE = (100\%/n) \cdot \Sigma \, |(y_i - \hat{y}_i) / y_i| \quad (14)$$

$$Accuracy = \max(0, 100\% - MAPE) \quad (15)$$

where n is the number of test samples, y_i is the actual Bitcoin price, and $\hat{y}_i$ is the model's predicted price. RMSE penalizes large errors more heavily through its squared term, MAE treats all errors proportionally and is more robust to outliers, and MAPE expresses error as a scale-independent percentage, making it convenient for comparing models regardless of the absolute price level. As a practical demonstration, the trained models were also used to generate a 30-day-ahead Bitcoin price forecast through a recursive forecasting procedure, in which each newly predicted value is fed back as the most recent Close feature for the subsequent prediction step.

III. RESULTS AND DISCUSSION

A. Dataset Overview

The historical BTC-USD dataset retrieved from Yahoo Finance is summarized in Table IV. The initial download yielded 4,140 daily rows spanning more than eleven years, with no missing values detected in any of the six original columns, confirming that the source data required no imputation before further processing.

TABLE IV. DATASET OVERVIEW

Item	Description
Data source	Yahoo Finance (yfinance library)

Item	Description
Ticker symbol	BTC-USD
Start date	January 1, 2015
End date	May 5, 2026
Initial rows	4,140
Initial columns	Date, Open, High, Low, Close, Volume
Frequency	Daily

Table V presents the descriptive statistics of the closing price and trading volume over the full sample period. The wide gap between the minimum ($178.10) and maximum ($124,752.53) closing price, together with a standard deviation ($32,385.24) that is larger than the median, confirms the strongly right-skewed and highly volatile nature of Bitcoin's price history, which provided the motivation in Section I for adopting deep-learning models capable of representing non-linear, non-stationary behavior.

TABLE V. DESCRIPTIVE STATISTICS OF CLOSING PRICE AND VOLUME

Statistic	Close (USD)	Volume (B)
Count	4,140	4,140
Mean	28,970.60	22.77
Std. dev.	32,385.24	22.99
Min	178.10	0.0079
25%	3,865.57	3.04
50% (median)	11,968.44	18.42
75%	46,454.19	34.73
Max	124,752.53	350.97

B. Data Preprocessing Results

After computing the three moving averages, RSI, and the MACD components, the rows affected by the rolling-window calculations, most notably the 50-day window required by MA50, were removed, reducing the dataset from 4,140 to 4,094 valid rows. A correlation analysis among the engineered features showed that MA7, MA21, and MA50 are highly correlated with Close, which is expected because all three are derived directly from the closing-price series, whereas RSI and the MACD components exhibit comparatively low correlation with price, confirming that they contribute relatively independent information about market momentum and trend strength rather than duplicating the price signal itself.

C. Model Training

Both the GRU and LSTM models were trained for up to 200 epochs with a batch size of 32, using identical training and validation splits and an identical callback configuration. In both cases, EarlyStopping halted training before the full 200-epoch budget was exhausted once the validation loss ceased to improve by more than the specified minimum delta over 20 consecutive epochs, and ModelCheckpoint together with restore_best_weights ensured that the weights ultimately retained for evaluation corresponded to the lowest validation loss observed during training rather than to the final epoch. This identical training protocol means that any differences in the resulting predictive performance can be attributed primarily to the architectural distinction between the GRU and LSTM cells rather than to differences in the optimization procedure.

D. Evaluation Results

Table VI summarizes the test-set evaluation metrics for both models after the predicted closing prices were inverse-transformed back to their original USD scale, and Fig. 2 presents the same comparison graphically.

TABLE VI. TEST-SET EVALUATION RESULTS

Metric	GRU	LSTM
RMSE (USD)	12,194.34	26,401.57
MAE (USD)	10,125.90	23,476.99
MAPE (%)	10.0629	23.9027
Accuracy (%)	89.9371	76.0973

Fig. 2. Comparison of RMSE, MAE, MAPE, and accuracy between the GRU and LSTM models on the test set.

Across all four metrics, GRU consistently outperformed LSTM on unseen test data: its RMSE and MAE were roughly 54% and 57% lower than those of LSTM, respectively, and its prediction accuracy (89.94%) was almost 14 percentage points higher than that of LSTM (76.10%). The strong performance achieved by both models is attributable to the comprehensive 11-feature multivariate input, which combines closing price, volume, three moving averages, RSI, three MACD-derived components, and two temporal features, providing substantially richer information than a univariate approach based on closing price alone.

E. Prediction Visualization and 30-Day Forecast

Beyond the aggregate error metrics, both models were further examined through visualizations comparing actual and predicted Bitcoin prices across the training, validation, and test subsets. On the training subset, the predicted and actual curves for both models lay very close together, indicating that the historical patterns were learned effectively, while on the validation and test subsets both models continued to follow the overall price trend, with larger deviations appearing during periods of sudden, high-volatility price movement, a well-known intrinsic challenge for time-series prediction on highly fluctuating cryptocurrency data. A scatter-plot analysis of actual versus predicted test-set values, together with an examination of the residual distribution in both USD and percentage terms, indicated that prediction errors were distributed approximately symmetrically around zero for both models, suggesting that neither model exhibited a strong systematic tendency to over- or under-predict price levels, although the spread of residuals was visibly narrower for GRU than for LSTM, consistent with its lower RMSE and MAE.

As a demonstration of practical applicability, both trained models were also used to generate a 30-day-ahead Bitcoin price forecast through a recursive forecasting procedure, in which each newly generated prediction is fed back into the input window to generate the subsequent day's forecast. Both forecasts were presented together with the preceding 90 days of historical data as context and a ±5% uncertainty band surrounding the projected trajectory. Because small errors accumulate across each recursive prediction step, the uncertainty associated with this type of forecast grows with the length of the prediction horizon; consequently, the 30-day projection should be interpreted as an illustrative demonstration of the models' practical applicability rather than as an investment recommendation.

F. Discussion

With respect to the first research question, the results in Table VI and Fig. 2 confirm that both GRU and LSTM are able to predict Bitcoin prices with satisfactory performance when trained on the proposed 11-feature multivariate input, achieving test-set accuracies of 89.94% and 76.10%, respectively. With respect to the second research question, the direct comparison shows that GRU consistently produced lower RMSE, MAE, and MAPE values than LSTM, indicating that GRU is the superior model under the configuration examined in this study.

This advantage can be explained by GRU's comparatively simple architecture. Unlike LSTM, which relies on three gates and a separate cell-state pathway, GRU uses only two gates, an update gate that controls how much past information is retained and a reset gate that determines how much past information is forgotten, without maintaining a distinct cell state. The resulting reduction in the number of trainable parameters allows GRU to optimize more efficiently and reduces its risk of overfitting on a moderately sized dataset such as the one used in this study, which is consistent with the architectural efficiency reported for GRU in [6] and the long-term dependency modelling strengths of LSTM discussed in [17] and [19]. These findings are broadly consistent with prior comparative studies suggesting that GRU and LSTM performance is highly sensitive to dataset characteristics, normalization strategy, and feature composition [9], [8], and they extend that literature by demonstrating a clear GRU advantage under a richly engineered multivariate technical-indicator feature set, an area noted in [10] and [18] as still relatively underexplored for direct LSTM-versus-GRU comparison.

Taken together, this study provides empirical evidence that GRU- and LSTM-based deep-learning models, when combined with a multivariate feature set spanning price, volume, technical indicators, and temporal features, can serve as reliable tools for Bitcoin price prediction. The resulting model has clear potential for integration into investor decision-support tools or as a component within an automated-trading system that reacts to market movements, as also envisioned for reinforcement-learning-based trading frameworks in [3] and [20]. Nevertheless, several limitations should be acknowledged. First, the models do not account for qualitative external factors such as social-media sentiment, regulatory decisions, or geopolitical events, all of which have been shown to move Bitcoin's price significantly within short periods. Second, the recursive forecasting procedure used for the 30-day-ahead projection is susceptible to the accumulation of error over longer prediction horizons. Third, this study did not perform systematic hyperparameter tuning, such as Grid Search or Bayesian Optimization, leaving room for further performance gains. Fourth, the very high correlation observed among the price-derived features (Open, High, Low, Close, and the three moving averages) suggests potential feature redundancy; feature-importance techniques such as SHAP values could be employed in future work to identify the most influential predictors and guide a more optimal feature-selection process.

IV. Conclusion

This study implemented and compared two deep-learning architectures, LSTM and GRU, for Bitcoin price prediction using a multivariate feature set comprising closing price, trading volume, three moving averages, RSI, three MACD-derived components, and two temporal features. Both models successfully followed the overall Bitcoin price trend across the training, validation, and test subsets, confirming that the multivariate feature set improves predictive performance compared with a univariate, closing-price-only approach. On the test set, GRU achieved an RMSE of $12,194.34, an MAE of $10,125.90, and a MAPE of 10.0629%, while LSTM achieved an RMSE of $26,401.57, an MAE of $23,476.99, and a MAPE of 23.9027%, demonstrating that GRU produced a substantially lower prediction error than LSTM under the configuration examined in this study. This advantage is attributed to GRU's simpler architecture and smaller parameter count, which allow for more efficient training and help maintain prediction stability under Bitcoin's highly fluctuating price conditions. Overall, the results indicate that deep-learning models, particularly GRU and LSTM, hold strong potential for application in cryptocurrency price-prediction systems and can be further developed to support investment decision-making or automated-trading applications.

Several directions are recommended for future research. First, incorporating additional external variables, such as social-media sentiment, economic news, on-chain blockchain data, or macroeconomic indicators, may help the model capture external influences on Bitcoin price movement more comprehensively. Second, systematic hyperparameter tuning through methods such as Grid Search, Random Search, or Bayesian Optimization could yield a more optimal model configuration and further improve predictive performance. Third, future work could compare LSTM and GRU against other deep-learning architectures, such as Bidirectional LSTM, Transformer-based models, CNN-LSTM, or other hybrid architectures, to obtain a broader understanding of time-series prediction performance on cryptocurrency data. Fourth, to improve generalization, subsequent studies could use a longer data period or extend the analysis to other cryptocurrency assets beyond Bitcoin, such as Litecoin or Ethereum, so that the models can be evaluated under more diverse market conditions. Fifth, because the recursive forecasting approach used in this study is prone to error accumulation over longer horizons, future studies may consider direct multi-step or sequence-to-sequence forecasting methods to produce more stable long-horizon predictions. Finally, the resulting models could be developed into a web- or mobile-based prediction application integrated with real-time market data, enabling direct use by investors and researchers for cryptocurrency market analysis.

ACKNOWLEDGMENT

The author would like to express sincere gratitude to the supervisor, Noper Ardi, S.Pd., M.Eng., for his guidance, direction, and patient feedback throughout the design and writing of this research, as well as to the examiners and colleagues who contributed through data discussion and scientific feedback. The author also thanks the Department of Informatics Engineering, Politeknik Negeri Batam, for the support provided during this study.

REFERENCES

[1] H. Alnami, M. Mohzary, B. Assiri, and H. Zangoti, "An integrated framework for cryptocurrency price forecasting and anomaly detection using machine learning," Applied Sciences, vol. 15, no. 4, p. 1864, 2025.

[2] W. Badar, S. Ramzan, A. Raza, N. L. Fitriyani, and M. Syafrudin, "Enhanced interpretable forecasting of cryptocurrency prices using hybrid CNN-LSTM model," Mathematics, vol. 13, no. 12, p. 1908, 2025.

[3] J. H. Chun and S. J. Lee, "Cryptocurrency futures portfolio trading system using reinforcement learning," Applied Sciences, vol. 15, no. 17, p. 9400, 2025.

[4] S. Corbet, B. Lucey, A. Urquhart, and L. Yarovaya, "Cryptocurrencies as a financial asset: A systematic analysis," International Review of Financial Analysis, vol. 62, pp. 182–199, 2021.

[5] J. D. Das, R. K. Thulasiram, and C. Henry, "Encoder–decoder based LSTM and GRU architectures for stocks and cryptocurrency prediction," Journal of Risk and Financial Management, vol. 17, no. 5, p. 200, 2024.

[6] A. Dutta, S. Kumar, and M. Basu, "A gated recurrent unit approach to bitcoin price prediction," Journal of Risk and Financial Management, vol. 13, no. 2, p. 23, 2020.

[7] M. J. Hamayel and A. Y. Owda, "A novel cryptocurrency price prediction model using GRU, LSTM and bi-LSTM machine learning algorithms," AI, vol. 2, no. 4, pp. 477–496, 2021.

[8] H. S. Htay, M. Ghahremani, and S. Shiaeles, "Enhancing bitcoin price prediction with deep learning: Integrating social media sentiment and historical data," Applied Sciences, vol. 15, no. 3, p. 1554, 2025.

[9] C. Y. Kang, C. P. Lee, and K. M. Lim, "Cryptocurrency price prediction with convolutional neural network and stacked gated recurrent unit," Data, vol. 7, no. 11, p. 149, 2022.

[10] R. Kaur, M. Uppal, D. Gupta, S. Juneja, and S. Y. Arafat, "Development of a cryptocurrency price prediction model: Leveraging GRU and LSTM for Bitcoin, Litecoin and Ethereum," PeerJ Computer Science, 2025.

[11] A. M. Khedr, I. Arif, M. El-Bannany, S. M. Alhashmi, and M. Sreedharan, "Cryptocurrency price prediction using traditional statistical and machine learning techniques: A survey," Intelligent Systems in Accounting, Finance and Management, vol. 28, no. 1, pp. 3–34, 2021.

[12] M. Kong and J. So, "Empirical analysis of automated stock trading using deep reinforcement learning," Applied Sciences, vol. 13, no. 1, p. 633, 2023.

[13] O. Kuznetsov, O. Kostenko, K. Klymenko, and Z. Hbur, "Machine learning analytics for blockchain-based financial markets: A confidence-threshold framework for cryptocurrency price direction prediction," Applied Sciences, vol. 15, no. 20, p. 11145, 2025.

[14] A. Ladhari and H. Boubaker, "Deep learning models for Bitcoin prediction using hybrid approaches," Forecasting, vol. 6, no. 2, p. 16, 2024.

[15] M.-C. Lee, "Bitcoin trend prediction with attention-based deep learning models and technical indicators," Systems, vol. 12, no. 11, p. 498, 2024.

[16] I. Nasirtafreshi, "Forecasting cryptocurrency prices using recurrent neural network and long short-term memory," Digital Signal Processing, 2022.

[17] P. L. Seabe, C. R. B. Moutsinga, and E. Pindza, "Forecasting cryptocurrency prices using LSTM, GRU, and bi-directional LSTM: A deep learning approach," Fractal and Fractional, vol. 7, no. 2, p. 203, 2023.

[18] S. Singh and P. Sharma, "Analysis and forecasting of cryptocurrency markets using Bayesian and LSTM-based deep learning models," Informatics, vol. 12, no. 3, p. 87, 2025.

[19] Y. Wang, S. Liu, Z. Wang, and Y. Li, "A stacking ensemble deep learning model for bitcoin price prediction using Twitter comments on bitcoin," Mathematics, vol. 10, no. 8, p. 1307, 2022.

[20] H. Yang and A. Malik, "Reinforcement learning pair trading: A dynamic scaling approach," Journal of Risk and Financial Management, vol. 17, no. 12, p. 555, 2024.