A Novel Hybrid Transformer-Based Deep Learning Approach for Multi-Step Bitcoin Price Forecasting

Rza Hasanli
Department of Information Systems, Graduate School of Informatics, Gazi University, Ankara, Turkiye
rza.hasanli@gazi.edu.tr (corresponding author)

Mahir Dursun
Department of Electrical and Electronics Engineering, Faculty of Technology, Gazi University, Ankara, Turkiye | Construction of Engineering Systems and Structures Department, Faculty of Water Management and Engineering Communication Systems, Azerbaijan University of Architecture and Construction, Baku, Azerbaijan
mdursun@gazi.edu.tr

Received: 3 December 2025 | Revised: 28 December 2025 | Accepted: 6 January 2026

 DOI: https://doi.org/10.48084/etasr.16684

ABSTRACT

The unpredictable and highly dynamic nature of cryptocurrency markets has driven researchers to develop advanced forecasting techniques that can support decision-making in trading and risk management. This study proposes a hybrid deep learning framework that combines a Transformer with recurrent models for multi-step Bitcoin price forecasting. The model operates on log-differenced closing prices and is evaluated for 7-, 14-, and 21-day ahead prediction using recursive multi-step forecasting schemes. In addition to the proposed Transformer-based architectures, a comprehensive comparison was conducted against four benchmark recurrent models, namely Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), along with their bidirectional variants Bidirectional-LSTM (BiLSTM) and Bidirectional GRU (BiGRU). Model performance was assessed using standard regression metrics, including Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and Directional Accuracy (DA) to quantify the accuracy of predicted price movements. The experimental results show that the Transformer-based models outperform standalone recurrent architectures across all forecasting methods, with the Transformer-LSTM achieving the lowest error values and strong trend-tracking behavior. These findings highlight the effectiveness of hybrid attention-recurrent architectures for modeling the nonlinear and volatile dynamics of Bitcoin markets.

Keywords-Bitcoin; cryptocurrency forecasting; deep learning; transformer; LSTM; GRU; multi-step forecasting

I. INTRODUCTION

Since its introduction in 2009 [1], Bitcoin has grown from a decentralized payment system into a major financial asset. The cryptocurrency market is highly volatile, with rapid price swings that create both high return potential and substantial risk. As of October 1, 2025, the global cryptocurrency market capitalization exceeded 4 trillion USD [2], highlighting its rising importance and the need for advanced predictive tools for trading and risk management. Accurate price forecasting is essential for systematic trading, portfolio optimization, and risk control. Traditional econometric models, such as ARIMA and GARCH, often fail to capture the nonlinear, noisy, and sometimes chaotic behavior of cryptocurrency markets. In contrast, machine learning and deep learning models can learn complex patterns, model temporal dependencies, and better handle noisy financial time series [3].

Recurrent Neural Networks (RNNs), particularly LSTM and GRU architectures, are widely used in cryptocurrency forecasting for their ability to process sequential data and capture long-range dependencies [4]. However, they can struggle during regime shifts and periods of extreme volatility, often underestimating sharp jumps and reversals. To address these issues, authors in [5, 6] proposed hybrid deep learning frameworks that augment recurrent models with additional components to improve robustness and adaptability. The transformer architecture introduced in [7] replaced recurrence with self-attention and transformed sequence modeling by efficiently learning both short- and long-range dependencies. Its success in natural language processing has motivated

applications in time series and financial forecasting [8, 9]. Hybrid models that combine Transformer-based attention with other neural architectures have been explored, using attention to capture global dependencies while preserving flexibility in modeling complex financial dynamics [10, 11]. Directional forecasting has also become a central focus, since it aligns more directly with trading decisions and supports the design of profitable yet risk-aware strategies [12, 13].

Cryptocurrency price forecasting has become an active research area, driven by extreme volatility and the need for robust trading strategies. A wide range of machine learning methods has been applied. Authors in [14] used Random Forests with Alpha101 factors and OHLCV features, showing that engineered variables can effectively predict short-term movements. Authors in [15] compared classification and time series forecasting models across several cryptocurrencies and found that simple time-series models, such as Exponential Smoothing, can outperform basic classifiers. Authors in [16] reported that ensemble-based trading strategies outperform single models for Bitcoin, Ethereum, and Litecoin, while authors in [17] demonstrated that combining resampling and ensemble methods (Logistic Regression, Random Forest, Gradient Boosting) yields steady gains over Buy-and-Hold. Deep learning models generally outperform traditional ML methods due to their capacity to learn long-horizon links and complex time series behavior. Authors in [3] demonstrated that LSTM surpasses standard RNN and ARIMA for Bitcoin forecasting. Authors in [4] found that LSTM outperforms GRNN and deep feedforward networks in modeling cryptocurrency volatility. Authors in [18] reported that GRU achieves the lowest errors among LSTM, GRU, and BiLSTM on Bitcoin, Ethereum, and Litecoin, while authors in [19] concluded that RNN-based models (LSTM, GRU) outperform CNNs in terms of both precision and trading returns. Authors in [20] showed that univariate LSTM models accurately forecast 1-min Bitcoin and Ethereum prices during early COVID-19, with Ethereum being more stable and predictable than Bitcoin. Hybrid architectures that combine convolutional and recurrent layers show strong potential for cryptocurrency forecasting. Authors in [5] proposed a CNN-LSTM model for Bitcoin with macroeconomic factors and reported better performance than a standalone LSTM. Authors in [21] used a 1D-CNN with stacked GRU and obtained higher accuracy than CNN-LSTM. Authors in [22] introduced a multi-input CNN-LSTM that reduced overfitting and improved efficiency for Bitcoin, Ethereum, and Ripple during the COVID-19 period. Similarly, authors in [23] proposed a 1D-CNN-LSTM hybrid for daily Bitcoin prices that achieved lower RMSE than simpler baselines, though performance remained sensitive to epoch choice and major market shocks. Incorporating non-price information further improves results. Authors in [24] demonstrated that trading and social indicators increase classification accuracy. Authors in [25] proposed DL-GuesS, which combines sentiment analysis with LSTM and GRU, and outperforms purely price-based models.

Feature enrichment is a central theme in many studies. Authors in [26] combined technical indicators and macroeconomic variables with SVM, XGBoost, and LSTM, and reported higher predictive power. Authors in [27] embedded MACD into LSTM and obtained profitable trading strategies with high win rates. Authors in [28] used technical indicators together with Twitter sentiment and achieved prediction accuracies close to 95 %. Research also focuses on directional prediction and trading performance. Authors in [29] used Deep Cross Networks for Bitcoin direction and obtained superior profitability and risk-adjusted returns. Authors in [30] compared machine learning with traditional technical analysis for 861 cryptocurrencies and found that machine learning performs the best on illiquid assets but may suffer from data-snooping bias. Authors in [31] combined autoencoders and LSTM for high-frequency direction forecasting and reported strong F1-scores on order book data. Authors in [32] enriched daily Bitcoin, Ethereum, and Ripple prediction with a high-dimensional set of technical indicators and PCA-based compression, reporting moderate directional accuracies (≈50–56%) but stronger risk-adjusted performance in long-only backtests, especially for XGBoost.

Overall, the literature points toward hybrid and deep sequence models. The present study proposes a hybrid Transformer–RNN framework for recursive multi-step Bitcoin forecasting over 1-, 2-, and 3-week horizons. This framework is compared with LSTM, GRU, BiLSTM, and BiGRU benchmarks, using log-differenced Bitcoin-USD closes in a sliding-window setup, while regression metrics and DA are evaluated.

II. METHODOLOGY

The proposed methodology includes a multi-horizon Bitcoin forecasting framework, where Bitcoin-USD closes are converted to standardized log returns using supervised samples via a sliding window. Furthermore, the study presents the hybrid Transformer and RNN benchmarks, and details the train–test split and evaluation setup across horizons.

A. Data Collection and Preprocessing

1) Cryptocurrency Price Data

The dataset consists of daily Bitcoin closing prices sourced from Yahoo Finance [33], covering the period from January 1, 2017, to November 1, 2025. This extended period captures multiple market regimes, including prolonged bull markets, sharp crashes, and sideway consolidation phases. Let P_t denote the closing price at trading day t. To stabilize variance and focus on relative changes, the logarithm of the closing price was defined as:

$$\ell_t = log(P_t) \quad (1)$$

and then form the first difference of the log-series:

$$r_t = \ell_t - \ell_{t-1} \quad (2)$$

which corresponds to the daily log return. Missing values resulting from the differencing step are discarded.

Figure 1 presents the daily log returns over the full sample period and illustrates typical cryptocurrency behavior with high volatility, volatility clustering, and sudden extreme movements. Episodes, such as the early 2020 COVID-19 crash and late bull-market corrections, show sharp negative spikes, whereas calmer periods display returns tightly concentrated around zero.

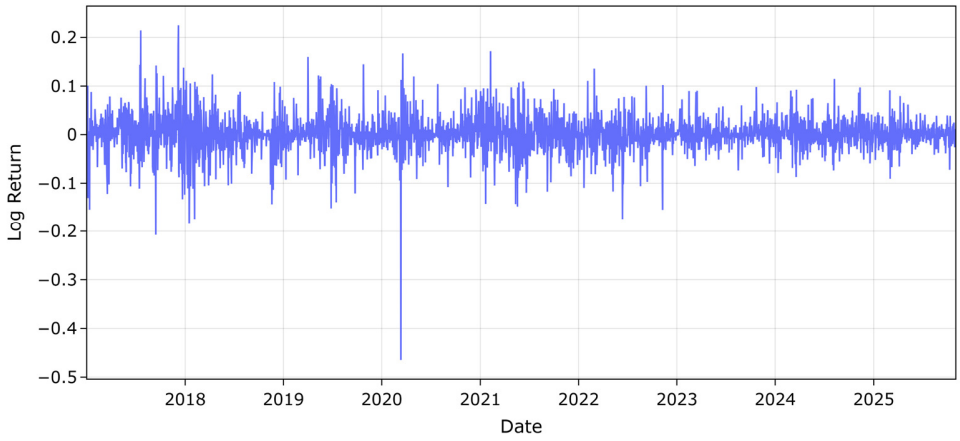

Fig. 1. Bitcoin-USD daily log returns for the considered period.

2) Data Normalization

The log-return series is standardized prior to model learning to ensure numerical consistency and improve training stability. The normalized log return is computed as:

$$\tilde{r}_t = \frac{r_t - \mu_{train}}{\sigma_{train}} \quad (3)$$

where μ_{train} and σ_{train} denote the mean and standard deviation, respectively, computed solely on the training set.

3) Sliding Window Representation

The standardized series $\{\tilde{r}_t\}$ are reformulated into supervised samples through a sliding window technique with a fixed window length T. For each time index t, the input-target pair was constructed as:

$$x_t = [\tilde{r}_t, \tilde{r}_{t+1}, \dots, \tilde{r}_{t+T-1}], y_t = \tilde{r}_{t+T}, \quad (4)$$

so that the model receives a sequence of the most recent T standardized log returns and predicts the next standardized log return. In the experiments, the window length was set to $T = 10$, which provides a balance between capturing short-term temporal dependencies and maintaining a compact representation suitable for deep sequence models.

B. Model Development

This section describes the deep neural network architectures used for Bitcoin price forecasting. The study considered both hybrid Transformer-RNN models and standalone recurrent networks. The hybrid models combine self-attention with recurrent sequence processing, and the purely recurrent models serve as direct benchmarks.

1) Transformer

The Transformer architecture [7] is a widely used framework for sequential modeling that captures temporal relationships without step-by-step recurrence. Through self-attention, each time step can attend to all others in parallel, which suits financial time series with both short- and long-range dependencies and abrupt movements in cryptocurrency returns. The main component of the Transformer encoder is the multi-head self-attention layer, formulated as:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \quad (5)$$

where Q, K, and V are the query, key, and value matrices, respectively, and d_k is the key dimension. This operation allows the encoder to capture relationships within the input sequence.

The present study uses only the encoder part of the Transformer. The encoder stacks input projection, positional encoding, multi-head self-attention, position-wise feedforward layers, dropout, and residual connections with layer normalization. This setup allows the model to process standardized log returns and capture temporal relationships across the input window.

2) Long Short-Term Memory

LSTM learns long-term dependencies by using a dedicated cell state that carries information along the sequence and helps mitigate the vanishing gradient problem. An LSTM cell has three gates that regulate information flow: the input gate, forget gate, and output gate, which act as adaptive filters deciding which past information is kept or discarded. The update equations are given as:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (6)$$

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (7)$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (8)$$

$$\tilde{C}_t = tanh(W_C[h_{t-1}, x_t] + b_C) \quad (9)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad (10)$$

$$h_t = o_t \odot tanh(C_t) \quad (11)$$

where x_t is the input at time t, C_t is the cell state, σ is the sigmoid activation function, and $\odot$ denotes element-wise multiplication.

3) Gated Recurrent Unit

The GRU is a streamlined version of the LSTM that offers similar predictive power with fewer parameters and lower computational cost. It also mitigates vanishing gradients using two gates that control information flow. The update gate z_t

determines how much of the previous hidden state is retained, and the reset gate r_t controls how the new input is blended with past information. The GRU equations are:

$$z_t = \sigma(W_z[h_{t-1}, x_t] + b_z) \quad (12)$$

$$r_t = \sigma(W_r[h_{t-1}, x_t] + b_r) \quad (13)$$

where σ is the sigmoid function, x_t is the input at time t, h_{t-1} is the previous hidden state, and W and b are learnable parameters. The hidden state is updated as:

$$\tilde{h}_t = tanh(W_h[r_t \odot h_{t-1}, x_t] + b_h) \quad (14)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \quad (15)$$

where $\odot$ denotes element-wise multiplication.

4) Bidirectional LSTM

BiLSTM extends LSTM by processing the sequence in both forward and backward directions, allowing each time step to exploit past and future information within the window and enriching temporal representations. This bidirectional setup is beneficial when patterns are not strictly one-directional. The hidden state at time t is defined by:

$$h_t = [\, \overrightarrow{h_t}, \overleftarrow{h_t} \,] \quad (16)$$

where $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ denote the forward and backward states, respectively, and their concatenation is passed to subsequent layers.

5) Bidirectional GRU

The BiGRU applies the same bidirectional idea as the BiLSTM to the GRU architecture. Forward and backward GRU layers run in parallel. So, each time step can use both past and future context within the sliding window, which can improve pattern recognition when dependencies are not strictly forward driven. The hidden state at time t is given by:

$$h_t = [\, \overrightarrow{h_t}, \overleftarrow{h_t} \,] \quad (17)$$

where $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ are the forward and backward GRU hidden states, respectively, and the gating mechanisms control how information from inputs and past states is combined.

6) Proposed Model

The forecasting approach used in this work relies on a hybrid deep learning design that integrates a transformer encoder with a recurrent layer for prediction. The model is intended to learn sequential temporal behavior from a fixed-size input window of standardized log returns and predict the next value in the sequence. Let the standardized log-return input window at time t be:

$$x_t = [r'_{t-T+1}, r'_{t-T+2}, \ldots, r'_t] \in \mathbb{R}^T \quad (18)$$

which is reshaped as a sequence matrix:

$$S_t \in \mathbb{R}^{T \times 1} \quad (19)$$

Each element in the sequence is projected into $d_{model} = 32$ dimensional embedding using a linear projection:

$$E_t = S_t W_e + 1 b_e^T, E_t \in \mathbb{R}^{T \times d_{model}} \quad (20)$$

where $W_e \in \mathbb{R}^{1 \times d_{model}}$, $b_e \in \mathbb{R}^{d_{model}}$, and 1 is a T-dimensional vector of ones. A fixed sinusoidal positional encoding matrix $P \in \mathbb{R}^{T \times d_{model}}$ is added to retain temporal ordering:

$$H_t^{(0)} = E_t + P \quad (21)$$

The transformed sequence is then passed through one Transformer encoder block. Each encoder block applies multi-head self-attention and a position-wise feedforward layer, both wrapped with residual connections and layer normalization. The encoder operations are expressed as:

$$Z_t^{att} = MHA\left(H_t^{(0)}\right) \quad (22)$$

$$H_t^{(1)} = LayerNorm\left(H_t^{(0)} + Dropout(Z_t^{att})\right) \quad (23)$$

$$Z_t^{ff} = FFN\left(H_t^{(1)}\right) \quad (24)$$

$$H_t^{(2)} = LayerNorm\left(H_t^{(1)} + Dropout\left(Z_t^{ff}\right)\right) \quad (25)$$

The resulting representation $H_t^{(2)} \in \mathbb{R}^{T \times d_{model}}$ captures dependencies across both short-term and long-term time spans across the input window. To summarize the encoded sequence into a fixed-length temporal representation, the Transformer output is passed to an LSTM with 64 units:

$$h_T = LSTM\left(H_t^{(2)}\right), h_T \in \mathbb{R}^{64} \quad (26)$$

The predicted standardized next log return is obtained from a fully connected output layer:

$$\hat{r}'_{t+1} = w_{out}^{\top} h_T + b_{out} \quad (27)$$

To isolate the effect of the recurrent unit in the hybrid design, a Transformer-GRU model was built that mirrors the Transformer-LSTM, but replaces the LSTM layer with a 64-unit GRU layer. All other components, including input projection, positional encoding, encoder setup, and output layer, remain the same. This variant enables a direct comparison between LSTM and GRU cells when combined with attention.

7) Benchmark Models

As baselines, four pure recurrent architectures without the Transformer encoder were included:

1. LSTM: a unidirectional LSTM [34] with 64 units followed by a dense output layer:

$$h_T = LSTM(X), \hat{y} = w_{out}^{\top} h_T + b_{out} \quad (28)$$

2. GRU: a unidirectional GRU [35] with 64 units followed by a dense output layer.
3. BiLSTM: a bidirectional LSTM [36] with 64 units in each direction. The forward and backward final states are concatenated and passed to the output layer.
4. BiGRU: a BiGRU [36] with 64 units in each direction.

All recurrent baselines receive the standardized log returns sequence $X \in \mathbb{R}^{T \times 1}$, without the attention-based encoder. By comparing these models to the hybrid architectures, the added

value of the transformer encoder was quantified for multi-horizon cryptocurrency forecasting.

C. Training and Testing Setup

The standardized log-return series $\{r'_t\}$ is divided chronologically into a training part and a test horizon of length $H \in \{7,14,21\}$. From the training portion, supervised samples are constructed using a sliding window of length T. For each time index t in the training set:

$$x_t = [r'_{t-T+1}, r'_{t-T+2}, \dots, r'_t] \in \mathbb{R}^T, y_t = r'_{t+1} \quad (29)$$

and the model $f(\cdot;\theta)$ is trained to approximate y_t from x_t. The parameters θ are estimated by minimizing the Mean Squared Error (MSE) using the Adam optimizer:

$$\min_\theta \mathcal{L}(\theta) = \frac{1}{N}\sum_{t\in\mathcal{T}_{\text{train}}}(y_t - f(x_t;\theta))^2, \quad (30)$$

During testing, recursive multi-step forecasting is employed. Let x^*_T denote the last observed input window before the test horizon. The first standardized forecast is:

$$\hat{r}'_{T+1} = f(x^*_T;\theta) \quad (31)$$

The prediction is then fed back into the input to form the next window:

$$x^*_{T+1} = [r'_{T-T+2}, \dots, r'_T, \hat{r}'_{T+1}] \quad (32)$$

and, in general, for $k = 1, \dots, H$:

$$\hat{r}'_{T+k} = f(x^*_{T+k-1};\theta) \quad (33)$$

The resulting sequence $\{\hat{r}'_{T+1}, \dots, \hat{r}'_{T+H}\}$ is inverse-transformed to obtain $\hat{r}_{T+k}$, accumulated to reconstruct predicted log prices $\hat{\ell}_{T+k}$, and finally converted to forecasted closing prices:

$$\hat{P}_{T+k} = exp(\hat{\ell}_{T+k}), k = 1, \dots, H \quad (34)$$

This framework is applied identically to all models, allowing a consistent comparison across architectures and horizons.

To illustrate the evaluation setup, Figure 2 presents the final portion of the dataset along with the testing segments corresponding to the 7-, 14-, and 21-day forecasting horizons. As observed, each horizon occupies the most recent observations, while the rest of the historical dataset is reserved solely for training the models.

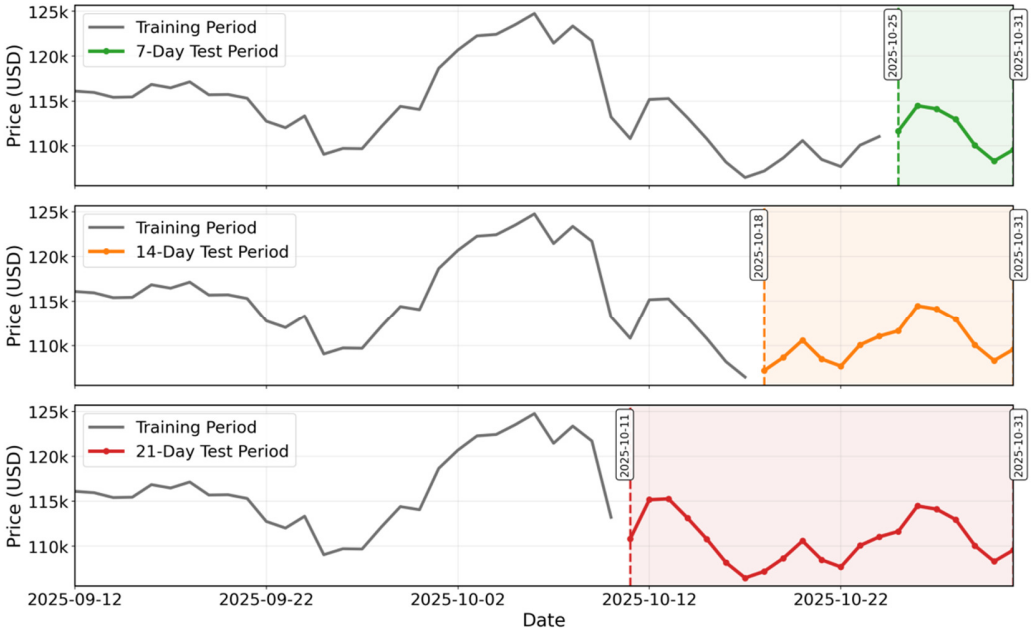

Fig. 2. Visualization of the testing setup over the final 50 observations.

D. Evaluation

Bitcoin price forecasting is formulated as a regression task, where the goal is to minimize the difference between the predicted closing price $\hat{P}_t$ and the true observed value P_t. Let N denote the number of test observations for a given forecasting horizon. The following evaluation metrics are used to assess predictive performance:

- MAE:

$$MAE = \frac{1}{N}\sum_{t=1}^{N} | P_t - \hat{P}_t | \quad (35)$$

- RMSE:

$$RMSE = \sqrt{\left(\frac{1}{N}\sum_{t=1}^{N}(P_t - \hat{P}_t)^2\right)} \quad (36)$$

- MAPE:

$$MAPE = \frac{100\%}{N}\sum_{t=1}^{N} | \frac{P_t - \hat{P}_t}{P_t} | \quad (37)$$

- DA:

$$DA = \frac{1}{N-1}\sum_{t=1}^{N} \mathbb{I}\left[sign(P_t - P_{t-1}) = sign\left(\hat{P}_t - \hat{P}_{t-1}\right)\right] \quad (38)$$

where $\mathbb{I}[\cdot]$ denotes the indicator function.

III. RESULTS AND DISCUSSION

A. 7-Day Forecasting Results

Table I reports the forecasting performance for all models on the 7-day horizon. Both hybrid Transformer-based models outperform the standalone recurrent networks. The Transformer-LSTM attains the lowest MAE and MAPE, while the Transformer-GRU delivers similar RMSE and DA, and remains competitive on all metrics. The high DA of both hybrids indicates that attention helps capture short-term movements in volatile prices, whereas conventional LSTM and GRU models struggle to predict direction changes.

TABLE I. MODEL PERFORMANCE FOR 7-DAY AHEAD FORECASTING

Model	MAE	RMSE	MAPE	DA
LSTM	2379.36	2944.65	2.11	0.33
GRU	2290.92	2564.10	2.06	0.17
BiLSTM	3818.07	4695.25	3.38	0.50
BiGRU	2724.37	3389.91	2.47	0.33
Transformer-LSTM	1599.52	1987.78	1.42	0.83
Transformer-GRU	1781.27	1986.54	1.60	0.83

Figure 3 further illustrates model behavior. The Transformer-based models track the actual price path more closely and preserve realistic trend direction, whereas the standalone recurrent models yield less stable forecasts.

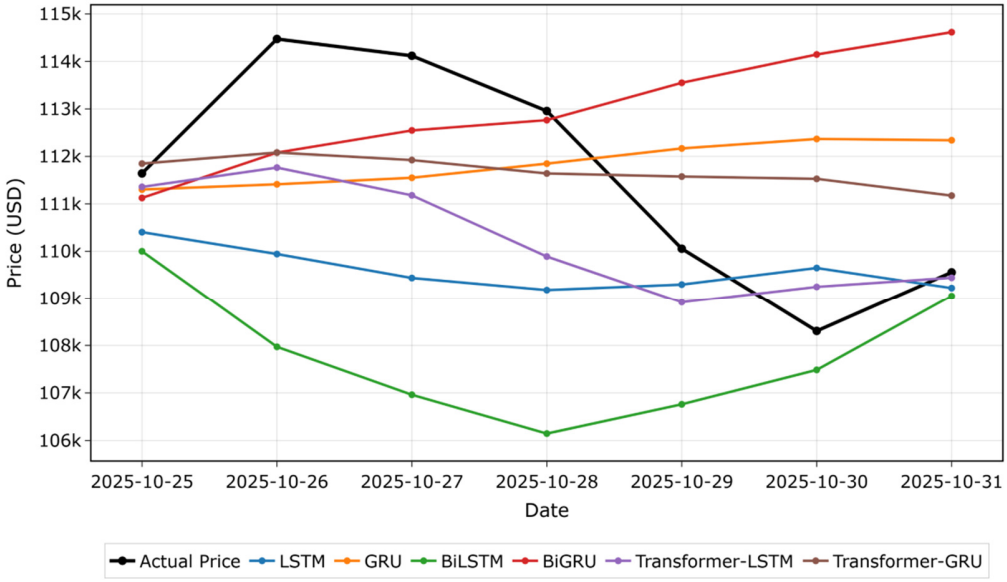

Fig. 3. Comparison of actual Bitcoin prices and model forecasts over the 7-day test horizon.

B. 14-Day Forecasting Results

Table II summarizes model performance on the 14-day horizon. As in the 7-day case, the Transformer-based architectures outperform the standalone recurrent models on all accuracy measures. The Transformer-LSTM attains the lowest errors, showing the strongest medium-term predictive accuracy, while the Transformer-GRU remains competitive with balanced performance across metrics. The DA values show that most models correctly capture price movement direction in more than half of the forecasts. The highest DA value occurs for both Transformer-LSTM and LSTM, but LSTM exhibits much larger numerical errors.

Figure 4 presents model behavior over the 14-day prediction window, where the actual price shows moderate short-term fluctuations followed by a downward correction. Deviations among model forecasts widen relative to the 7-day horizon, highlighting the greater difficulty of medium-term prediction in a volatile market. As displayed in Table II, the Transformer-based models track the true price more closely than the standalone recurrent architectures, with the Transformer-LSTM and Transformer-GRU both maintaining realistic trend dynamics as the horizon increases.

TABLE II. MODEL PERFORMANCE FOR 14-DAY AHEAD FORECASTING

Model	MAE	RMSE	MAPE	DA
LSTM	4799.16	5393.94	4.36	0.69
GRU	4307.66	5176.18	3.91	0.62
BiLSTM	3061.41	3997.16	2.79	0.62
BiGRU	4597.84	7325.59	4.18	0.46
Transformer-LSTM	2196.95	2740.62	1.98	0.69
Transformer-GRU	2758.09	3280.21	2.50	0.62

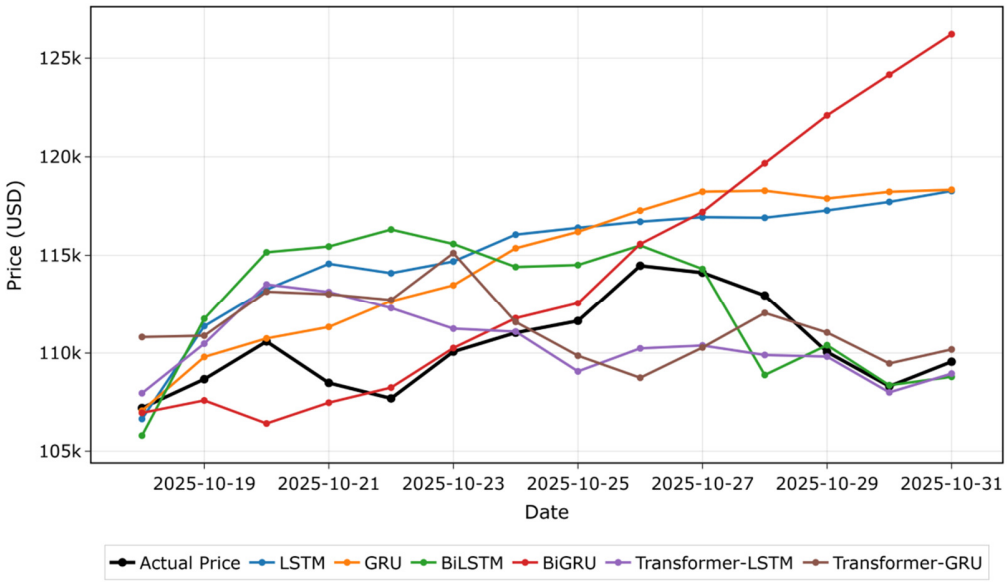

Fig. 4. Comparison of actual Bitcoin prices and model forecasts over the 14-day test horizon.

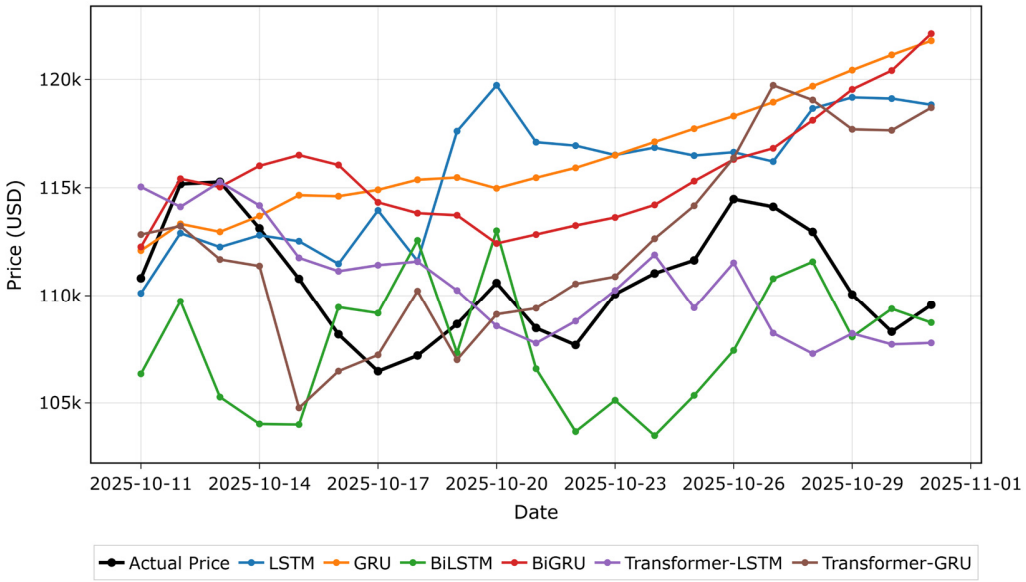

Fig. 5. Comparison of actual Bitcoin prices and model forecasts over the 21-day test horizon.

C. 21-Day Forecasting Results

Table III shows the 21-day forecasting results. Errors increase for all models at this longer horizon, but the Transformer-based models still perform the best. The Transformer-LSTM achieves the lowest MAE, RMSE, and MAPE, while the Transformer-GRU remains competitive and matches the highest DA, indicating stable trend tracking even for longer forecasts.

Figure 5 shows how model forecasts diverge over the 21-day horizon. The actual price path combines short-term fluctuations with a clear downward trend, and the gap between model trajectories widens compared with shorter horizons, reflecting the added difficulty of long-term forecasting. Among all models, the Transformer-LSTM and Transformer-GRU remain the closest to the true path, with the Transformer-LSTM providing smooth adjustments.

TABLE III. MODEL PERFORMANCE FOR 21-DAY AHEAD FORECASTING

Model	MAE	RMSE	MAPE	DA
LSTM	5499.71	6379.59	5.01	0.55
GRU	6131.99	6927.70	5.59	0.45
BiLSTM	4248.09	5036.29	3.81	0.60
BiGRU	4951.47	6007.00	4.52	0.40
Transformer-LSTM	2235.58	2842.58	2.02	0.65
Transformer-GRU	3448.04	4352.64	3.12	0.65

Most Bitcoin forecasting studies in the literature focus on one-step-ahead prediction (i.e., using information up to day *t* to predict the price at *t+1*). Although this setting often yields relatively low regression errors, many one-step studies do not report directional performance (DA/MDA) that is critical for trading-oriented decisions, and benchmark against simple naïve persistence baselines (e.g., tomorrow's price ≈ today's price), which can be surprisingly strong at very short horizons. In addition, reported one-step accuracy can become overly optimistic if the evaluation pipeline implicitly uses future information, which is not realistic in deployment. In contrast, the present study evaluates a recursive multi-step protocol over 7/14/21 days, where each future step is generated by feeding the previous predictions back into the input, thereby preventing access to unknown future prices and better reflecting multi-day planning.

Moreover, the present study reports DA in addition to regression metrics, enabling a more trading-relevant interpretation. As summarized in Table IV, the proposed model (Transformer-LSTM) achieves MAPE values of 1.42%, 1.98%, 2.02%, and DA of 0.83, 0.69, 0.65 for 7,14, and 21 days, respectively. In the closest comparable multi-horizon studies, longer-horizon errors are generally higher (e.g., 7-day MAPE ≈ 2.88% in a high-dimensional ML benchmark, and multi-horizon TFT variants report MAPE around 7‑8%, depending on covariates and probabilistic evaluation settings). Even when some one-step systems report competitive next-day regression metrics, their results are not directly comparable to multi-step horizons, and the directional performance is typically lower than reported by the present study. Overall, the comparison indicates that the proposed Transformer-based hybrid maintains stronger accuracy and direction consistency under the more challenging multi-step forecasting protocol.

TABLE IV. COMPARATIVE EVALUATION OF THE PROPOSED METHOD WITH OTHER RELATED RESEARCH WORKS

Study	Forecast type	Horizon(s)	Reported metrics	Best reported result(s)
[37]	Mostly one-step (next-day) multi-model system	1 day	RMSE, MAE, MAPE, sMAPE, DA, R^2	MAPE 2.79%, DA 47.18%, RMSE 2505.84, MAE 1760.93, R^2 0.99
[38]	ML with high-dimensional features	7 days (also 30/90)	MAE, RMSE, MAPE, DA	7-day MAPE ≈ 2.88%, DA up to 62%
[39]	BART versus ARIMA/ARFIMA	5–30 days	RMSE (%)	RMSE% around 4 (14d) and 6 (21d) (BART)
[40]	Transformer (TFT), multi-horizon (probabilistic)	Multi-horizon (output length ≈ 16)	MAE, RMSE, MAPE, quantile loss	MAPE reported around 7–8%, depending on the setup
[41]	Multi-stage multivariate DL	1 to 7 days	MAE, SMAPE, MDA	sMAPE ~ 2-5%, MDA ~ 50–55%
Proposed Transformer-LSTM	Recursive multi-step	7 / 14 / 21 days	MAE, RMSE, MAPE, DA	MAPE: 1.42 / 1.98 / 2.02 (%); DA: 0.83 / 0.69 / 0.65

IV. CONCLUSION AND FUTURE WORK

This study evaluated several deep learning models for recursive multi-step Bitcoin price forecasting over 7-, 14-, and 21-day horizons. Using standardized log returns and a sliding-window setup, each model was trained and tested on multi-step horizons.

The results show that the Transformer-based hybrids consistently outperform standalone recurrent networks, especially as the forecast horizon lengthens. Among all models, the Transformer-Long Short-Term Memory (LSTM) achieved the best error metrics and directional accuracy, with the Transformer-Gated Recurrent Unit (GRU) performing closely behind. This demonstrates the benefit of combining self-attention with recurrent memory, allowing the models to capture both global dependencies and local temporal patterns. Conventional Recurrent Neural Network (RNN)-based models remained reasonable but degraded more quickly at longer horizons, particularly in volatile or reversal periods, indicating that attention-enhanced architectures offer a more robust framework for medium- to long-term cryptocurrency forecasting. However, Bitcoin markets are occasionally dominated by abrupt exogenous shocks (e.g., fraud/hacks or regulatory announcements) that cannot be reliably predicted using price history alone. Thus, the findings should be interpreted as learning recurring temporal structure and providing multi-step forecasts under typical conditions, while performance may degrade during shock-driven regime shifts.

Future research should integrate additional information, such as sentiment or on-chain indicators, consider alternative forecasting schemes beyond recursive prediction, and test more advanced architectures. Extending the analysis to other cryptocurrencies and linking forecasts to trading performance would further clarify practical usefulness and robustness. In addition, evaluating generalization over longer out-of-sample windows that include event-driven regimes (and incorporating exogenous signals to model such shocks) is an important direction for future work.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," *SSRN Electronic Journal*, 2008, https://doi.org/10.2139/ssrn.3440802.

[2] "Global Cryptocurrency Market Cap Charts," *CoinGecko*, Nov. 2025. https://www.coingecko.com/en/charts.

[3] S. McNally, J. Roche, and S. Caton, "Predicting the Price of Bitcoin Using Machine Learning," in *2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing*, Cambridge, UK, Mar. 2018, pp. 339–343, https://doi.org/10.1109/PDP2018.2018.00060.

[4] S. Lahmiri and S. Bekiros, "Cryptocurrency Forecasting with Deep Learning Chaotic Neural Networks," *Chaos, Solitons & Fractals*, vol. 118, pp. 35–40, Jan. 2019, https://doi.org/10.1016/j.chaos.2018.11.014.

[5] Y. Li and W. Dai, "Bitcoin Price Forecasting Method Based on CNN-LSTM Hybrid Neural Network Model," *The Journal of Engineering*, vol. 2020, no. 13, pp. 344–347, Jul. 2020, https://doi.org/10.1049/joe.2019.1203.

[6] A. Politis, K. Doka, and N. Koziris, "Ether Price Prediction Using Advanced Deep Learning Models," in *2021 IEEE International Conference on Blockchain and Cryptocurrency*, Sydney, Australia, May 2021, pp. 1–3, https://doi.org/10.1109/ICBC51069.2021.9461061.

[7] A. Vaswani *et al.*, "Attention Is All You Need," in *31st Conference on Neural Information Processing Systems*, Long Beach, CA, USA, 2017, https://doi.org/10.48550/ARXIV.1706.03762.

[8] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition Transformers with Auto-correlation for Long-Term Series Forecasting," in *Proceedings of the 35th International Conference on Neural Information Processing Systems*, Virtual (Online), 2021, pp. 22419–22430, https://doi.org/10.48550/ARXIV.2106.13008.

[9] H. Zhou *et al.*, "Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 12, pp. 11106–11115, May 2021, https://doi.org/10.1609/aaai.v35i12.17325.

[10] M. A. Izadi and E. Hajizadeh, "Time Series Prediction for Cryptocurrency Markets with Transformer and Parallel Convolutional Neural Networks," *Applied Soft Computing*, vol. 177, Jun. 2025, Art. no. 113229, https://doi.org/10.1016/j.asoc.2025.113229.

[11] M. A. L. Khaniki and M. Manthouri, "Enhancing Price Prediction in Cryptocurrency Using Transformer Neural Network and Technical Indicators," arXiv, Mar. 06, 2024, https://doi.org/10.48550/arXiv.2403.03606.

[12] M. Dixon, D. Klabjan, and J. H. Bang, "Classification-Based Financial Markets Prediction Using Deep Neural Networks," *Algorithmic Finance*, vol. 6, no. 3–4, pp. 67–77, Dec. 2017, https://doi.org/10.3233/AF-170176.

[13] T. Fischer and C. Krauss, "Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions," *European Journal of Operational Research*, vol. 270, no. 2, pp. 654–669, Oct. 2018, https://doi.org/10.1016/j.ejor.2017.11.054.

[14] J. Sun, Y. Zhou, and J. Lin, "Using Machine Learning for Cryptocurrency Trading," in *2019 IEEE International Conference on Industrial Cyber Physical Systems*, Taipei, Taiwan, May 2019, pp. 647–652, https://doi.org/10.1109/ICPHYS.2019.8780358.

[15] G. Attanasio, L. Cagliero, P. Garza, and E. Baralis, "Quantitative Cryptocurrency Trading: Exploring the Use of Machine Learning Techniques," in *Proceedings of the 5th Workshop on Data Science for Macro-modeling with Financial and Economic Datasets*, Amsterdam, Netherlands, Jun. 2019, pp. 1–6, https://doi.org/10.1145/3336499.3338003.

[16] H. Sebastião and P. Godinho, "Forecasting and Trading Cryptocurrencies with Machine Learning Under Changing Market Conditions," *Financial Innovation*, vol. 7, no. 1, Jan. 2021, Art. no. 3, https://doi.org/10.1186/s40854-020-00217-x.

[17] T. A. Borges and R. F. Neves, "Ensemble of Machine Learning Algorithms for Cryptocurrency Investment with Different Data Resampling Methods," *Applied Soft Computing*, vol. 90, May 2020, Art. no. 106187, https://doi.org/10.1016/j.asoc.2020.106187.

[18] 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, Oct. 2021, https://doi.org/10.3390/ai2040030.

[19] S. Goutte, H.-V. Le, F. Liu, and H.-J. Von Mettenheim, "Deep Learning and Technical Analysis in Cryptocurrency Market," *Finance Research Letters*, vol. 54, Jun. 2023, Art. no. 103809, https://doi.org/10.1016/j.frl.2023.103809.

[20] J. Bouslimi, S. Boubaker, and K. Tissaoui, "Forecasting of Cryptocurrency Price and Financial Stability: Fresh Insights Based on Big Data Analytics and Deep Learning Artificial Intelligence Techniques," *Engineering, Technology & Applied Science Research*, vol. 14, no. 3, pp. 14162–14169, Jun. 2024, https://doi.org/10.48084/etasr.7096.

[21] 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, Oct. 2022, Art. no. 149, https://doi.org/10.3390/data7110149.

[22] I. E. Livieris, N. Kiriakidou, S. Stavroyiannis, and P. Pintelas, "An Advanced CNN-LSTM Model for Cryptocurrency Forecasting," *Electronics*, vol. 10, no. 3, Jan. 2021, Art. no. 287, https://doi.org/10.3390/electronics10030287.

[23] 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, Oct. 2023, https://doi.org/10.48084/etasr.6223.

[24] M. Ortu, N. Uras, C. Conversano, S. Bartolucci, and G. Destefanis, "On Technical Trading and Social Media Indicators for Cryptocurrency Price Classification Through Deep Learning," *Expert Systems with Applications*, vol. 198, Jul. 2022, Art. no. 116804, https://doi.org/10.1016/j.eswa.2022.116804.

[25] R. Parekh *et al.*, "*DL-GuesS*: Deep Learning and Sentiment Analysis-Based Cryptocurrency Price Prediction," *IEEE Access*, vol. 10, pp. 35398–35409, 2022, https://doi.org/10.1109/ACCESS.2022.3163305.

[26] N. Uras and M. Ortu, "Investigation of Blockchain Cryptocurrencies' Price Movements Through Deep Learning: A Comparative Analysis," in *2021 IEEE International Conference on Software Analysis, Evolution and Reengineering*, Honolulu, HI, USA, Mar. 2021, pp. 715–722, https://doi.org/10.1109/SANER50967.2021.00091.

[27] M. Rahmani Cherati, A. Haeri, and S. F. Ghannadpour, "Cryptocurrency Direction Forecasting Using Deep Learning Algorithms," *Journal of Statistical Computation and Simulation*, vol. 91, no. 12, pp. 2475–2489, Aug. 2021, https://doi.org/10.1080/00949655.2021.1899179.

[28] M. Khalid Salman and A. Abdu Ibrahim, "Price Prediction of Different Cryptocurrencies Using Technical Trade Indicators and Machine Learning," *IOP Conference Series: Materials Science and Engineering*, vol. 928, no. 3, Nov. 2020, Art. no. 032007, https://doi.org/10.1088/1757-899X/928/3/032007.

[29] P. K. Nagula and C. Alexakis, "A New Hybrid Machine Learning Model for Predicting the Bitcoin (BTC-USD) Price," *Journal of Behavioral and Experimental Finance*, vol. 36, Dec. 2022, Art. no. 100741, https://doi.org/10.1016/j.jbef.2022.100741.

[30] D.-G. Anghel, "A Reality Check on Trading Rule Performance in the Cryptocurrency Market: Machine Learning vs. Technical Analysis," *Finance Research Letters*, vol. 39, Mar. 2021, Art. no. 101655, https://doi.org/10.1016/j.frl.2020.101655.

[31] F. Fang *et al.*, "Ascertaining Price Formation in Cryptocurrency Markets with Machine Learning," *The European Journal of Finance*, vol. 30, no. 1, pp. 78–100, Jan. 2024, https://doi.org/10.1080/1351847X.2021.1908390.

[32] R. Hasanli and M. Dursun, "Integrating High-Dimensional Technical Indicators into Machine Learning Models for Predicting Cryptocurrency Price Movements and Trading Performance: Evidence from Bitcoin, Ethereum, and Ripple," *FinTech*, vol. 4, no. 4, Dec. 2025, Art. no. 77, https://doi.org/10.3390/fintech4040077.

[33] "BTC-USD," *Yahoo Finance*, Nov. 2025. https://finance.yahoo.com/quote/BTC-USD/.

[34] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, https://doi.org/10.1162/neco.1997.9.8.1735.

[35] K. Cho, B. Van Merrienboer, D. Bahdanau, and Y. Bengio, "On the Properties of Neural Machine Translation: Encoder–Decoder Approaches," in *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, Doha, Qatar, 2014, pp. 103–111, https://doi.org/10.3115/v1/W14-4012.

[36] M. Schuster and K. K. Paliwal, "Bidirectional Recurrent Neural Networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997, https://doi.org/10.1109/78.650093.

[37] C. Varalakshmi, M. Ranka, S. Christina, M. M. Sucharitha, and M. S. A. Basha, "Deploying a Multi-Model Forecasting System for Bitcoin Prices: Bridging Statistical Forecasting and Deep Learning Innovations,"

in *2025 3rd International Conference on Data Science and Information System*, Hassan, India, May 2025, pp. 1–8, https://doi.org/10.1109/ICDSIS65355.2025.11070535.

[38] M. Mudassir, S. Bennbaia, D. Unal, and M. Hammoudeh, "Time-Series Forecasting of Bitcoin Prices Using High-Dimensional Features: A Machine Learning Approach," *Neural Computing and Applications*, vol. 37, no. 28, pp. 22979–22993, Jul. 2020, https://doi.org/10.1007/s00521-020-05129-6.

[39] V. Derbentsev, N. Datsenko, O. Stepanenko, and V. Bezkorovainyi, "Forecasting Cryptocurrency Prices Time Series Using Machine Learning Approach," *SHS Web of Conferences*, vol. 65, 2019, Art. no. 02001, https://doi.org/10.1051/shsconf/20196502001.

[40] A. J. Amadeo, J. G. Siento, T. A. Eikwine, Diana, and I. H. Parmonangan, "Temporal Fusion Transformer for Multi Horizon Bitcoin Price Forecasting," in *2023 IEEE 9th Information Technology International Seminar*, Batu Malang, Indonesia, Oct. 2023, pp. 1–7, https://doi.org/10.1109/ITIS59651.2023.10420330.

[41] N. Sizykh, S. Dandamaev, and D. Sizykh, "Application of the Method of Multivariate Multi-stage Forecasting Based on the LSTM Deep Learning Model for Bitcoin Price Time Series," in *2023 16th International Conference Management of Large-Scale System Development*, Moscow, Russian Federation, Sept. 2023, pp. 1–5, https://doi.org/10.1109/MLSD58227.2023.10304008.