Integraling NASDAQ Spillover Signals into Hybrid 1D-CNN-LSTM Frameworks for Multi-Horizon Bitcoin Forecasting

Hoa Tran Thai[1][0009-0006-2405-4697] , Thanh Manh Le[2], Huu-Trung Hoang[1], Nam Nguyen Ngoc[1], Hieu T. Truong-Nguyen[1], and Cuong H. Nguyen-Dinh[3][0000-0001-6872-4644]

[1] University of Economics, Hue University, Viet Nam
[2] University of Sciences, Hue University
[3] Phu Xuan University, Viet Nam
tranthaihoa@hueuni.edu.vn, lmthanh@husc.edu.vn, hoanghu-utrung@hueuni.edu.vn, nnnam@hce.edu.vn, tnthieu@hce.edu.vn, cuong-nguyen@pxu.edu.vn

Abstract. Bitcoin's extreme volatility and non-linear dynamics pose substantial challenges for traditional econometric models. This study incorporates cross-market spillover signals from the NASDAQ Composite Index into a hybrid deep learning framework. An empirical analysis confirms a time-varying dependency between Bitcoin and NASDAQ returns, supporting the inclusion of exogenous features. We systematically evaluate architectures combining 1D-CNN with four recurrent variants (Simple RNN, LSTM, GRU, Bi-LSTM) across 3-, 5-, and 7-day forecasting horizons. The results reveal a horizon-dependent trade-off: the 1D-CNN-GRU achieves the lowest error for short-term forecasts (MSE: 0.000424, R^2: 0.9945), outperforming LSTM by approximately 20%. Conversely, the 1D-CNN-LSTM provides higher stability in the 7-day horizon (MSE: 0.000428), reducing error by 11.9% relative to the GRU. These findings validate the efficacy of combining CNN-based feature extraction with LSTM's long-term sequence modeling.

Keywords: Bitcoin, NASDAQ, deep learning, 1D-CNN - LSTM, Spillover effects, Time-series forecasting.

1. Introduction

Bitcoin influences global markets, yet its price forecasting remains challenging due to volatility, non-stationarity, and nonlinear behavior [1]. Traditional econometric models, such as ARIMA or GARCH, are often inadequate for capturing these chaotic properties, and frequently struggle with volatility clustering and heavy-tailed distributions [2]. To address these limitations, deep learning approaches have achieved substantial progress. In particular, hybrid architectures have emerged as an effective solution by combining the strengths of individual model components. Specifically, integrating 1D-CNN for local feature extraction with LSTM networks for sequence modeling has demonstrated improved predictive performance. This study focused on the 1D-CNN – LSTM combination, leveraging the LSTM's ability to mitigate the vanishing gradient problem and retain long-term temporal dependencies [3], [4]. Beyond architectural design, recent research has expanded the scope of input features, recognizing that the cryptocurrency market is influenced by spillover effects from traditional financial markets [5]. The NASDAQ Composite Index, in particular, reflects investor risk appetite and has been shown to influence Bitcoin valuations [6]. While macroeconomic indicators have been incorporated into forecasting models [7], [8] a systematic evaluation of NASDAQ spillover signals within a unified 1D-CNN-RNN framework—particularly one comparing recurrent variants across multiple forecasting horizons—remains under-explored.

This study aimed to bridge these gaps through three primary contributions. First, we conducted an empirical analysis using dynamic rolling correlation to validate the time-varying relationship between Bitcoin and the NASDAQ index, thereby justifying the integration of stock market returns as an exogenous predictive feature. Second, we proposed and evaluated a hybrid 1D-CNN framework combined with several recurrent variants to identify the most suitable architecture for cryptocurrency forecasting. Finally, unlike previous studies that primarily focus on single-step predictions, we conducted a detailed comparative analysis of performance trade-offs across three forecasting horizons (3-day, 5-day, and 7-day). Our results revealed a critical distinction: while the computationally efficient 1D-CNN-GRU model excelled in short-term forecasting, the 1D-CNN-LSTM model demonstrated superior robustness and accuracy for medium- to long-term predictions.

The remainder of this paper is organized as follows. The remainder of this paper is organized as follows. Section 2 reviews the relevant literature.. Section 3 details the research methodology, including data preprocessing, the proposed model architectures, and evaluation metrics. Section 4 presents the experimental results and in-depth analysis. Finally, Section 5 concludes the paper and summarizes the key findings of the study.

2. Related Work

The challenge of forecasting Bitcoin prices has attracted extensive academic interest due to the cryptocurrency's inherent volatility and complex non-linear dynamics [9]. While traditional econometric models often fail to capture these chaotic patterns, deep learning architectures have demonstrated enhanced capability in modeling financial time series [2]. Sangheer et al. [10] developed a hybrid deep learning model based on feature selection in different frequency domains, incorporating indicators such as technology, economy, green finance and media attention, demonstrating that advanced deep learning approaches can significantly outperform benchmark models in both prediction accuracy and simulated trading returns. Research in this domain has progressed from standalone deep learning models to hybrid architectures, and more recently, to frameworks incorporating exogenous market signals. Bouteska et al. [11] conducted a comprehensive comparative analysis confirming that for cryptocurrency price forecasting, LSTM and GRU neural networks remain the most widely adopted due to their self-feedback mechanisms enabling effective retention and exploitation of sequential data patterns over extended periods.

RNNs and their variants, such as LSTM and GRU, have become the cornerstone of sequence modeling due to their ability to capture temporal dependencies [3], [4]. Specifically, the LSTM architecture addresses the vanishing gradient problem found in standard RNNs, making them effective for learning long-term dependencies [1]. The gating mechanisms—input, forget, and output—enable LSTM to effectively manage long-term dependencies, with studies demonstrating that this specialized architecture renders them especially effective for financial forecasting applications where the ability to identify and analyze long-term trends is essential [12]. Empirical studies by Wardak and Rasheed [13] and Gunarto et al. [14] have shown that LSTM models outperform basic RNNs in predicting major cryptocurrencies such as Bitcoin and Ethereum. However, a key limitation of standalone RNNs is their sensitivity to noise; they often struggle to extract high-level features from raw financial data, which can hinder predictive stability [4]. Recent comparative studies by Priadinata et al. [15] found that GRU consistently demonstrated superior performance in cryptocurrency price prediction, achieving notably lower RMSE and MAPE values compared to both LSTM and Bi-LSTM models, particularly for Bitcoin and Ethereum, suggesting that GRU's simpler architectural design and fewer training parameters enable enhanced ability to efficiently capture temporal dependencies in highly volatile financial data.

To mitigate the limitations of standalone models, recent studies have shifted toward hybrid architectures that integrate 1D-CNN with RNNs. Originally designed for computer vision, 1D-CNNs have been adapted to identify local patterns and short-term trends in sequential data [16]. Kiranyaz et al. [17] presented a comprehensive review establishing that 1D CNNs effectively capture local features and pattern recognition by leveraging convolutions and hierarchical feature extraction, making them particularly effective in financial data anomaly detection and trend analysis. Cavalli and Amoretti [8] showed that 1D-CNNs can outperform LSTMs in certain trend-prediction tasks by efficiently filtering input features. Zeng et al. [18] demonstrated that CNNs are particularly adept at capturing local patterns for modeling short-term dependencies, though they acknowledge that CNNs alone cannot learn long-term dependencies due to limited receptive fields, thereby justifying the integration with Transformer or RNN architectures to capture both short-term and long-term dependencies within time series.

Combining these architectures enables the 1D-CNN layer to act as an automated feature extractor—filtering noise and identifying local patterns—while the subsequent RNN layer models the temporal evolution of these features. Ahmed et al. [19] and

Cahyadi and Zahra [20] applied CNN–LSTM frameworks to Bitcoin forecasting and reported improved accuracy over baseline models. Ferdiansyah et al. [21] indecated that while LSTM struggles with highly fluctuating Bitcoin data, using a short recent window (six days) and adopting a CNN-LSTM hybrid significantly enhances predictive stability and improves evaluation performance compared to previous research. Furthermore, Moodi et al. [7] explored more complex configurations such as 1D-CNN-GRU-LSTM, highlighting the benefits of combining multiple recurrent structures.

Beyond architectural design, the scope of input data is critical. Increasing evidence suggests that the cryptocurrency market is not isolated but is influenced by macroeconomic conditions and traditional financial markets [22]. Of particular interest is the spillover effect originating from the technology sector. Cheon et al. [6] examined the relationship between the NASDAQ index and Bitcoin using machine learning models and confirmed a significant positive linkage with predictive value.

Despite advances in hybrid modeling and cross-market analysis, several gaps remain in the literature. First, few studies have systematically integrated NASDAQ spillover signals into a unified CNN-RNN framework for cryptocurrency forecasting. Second, there is limited comparative analysis of which recurrent variant (Simple RNN, LSTM, GRU, or Bi-LSTM) performs best within such hybrid structures. Vanderbilt et al. [23] conducted an applied study comparing Simple RNN, LSTM, and GRU models for cryptocurrency prediction, examining whether one model predicts cryptocurrency prices more accurately than others, yet their findings highlighted the need for more systematic comparison within hybrid architectures. Finally, most existing research focuses on single-step or short-term prediction, often neglecting the performance trade-offs inherent in multi-horizon forecasting (3-day, 5-day, and 7-day). Farooq et al. [24] addressed multi-horizon cryptocurrency forecasting through an advanced Deep Learning-Enhanced Temporal Fusion Transformer model, demonstrating reduced MAPE, MSE, and RMSE values while emphasizing that improving forecasting accuracy for multiple horizons requires addressing different normalization strategies and utilizing various market-related data. This study addresses these gaps by proposing a hybrid framework that leverages NASDAQ signals and rigorously evaluates multiple recurrent variants across several forecasting horizons.

3. Methodology

3.1. Data collection and feature engineering

To capture volatility and cross-market spillovers, we utilize historical daily data from Bitcoin (BTC) and NASDAQ (^IXIC). At each time step t, the input feature vector $x_t \in R^{10}$ comprises the OHLCV attributes (Open, High, Low, Close, Volume) for both markets, as defined in in Equation (1).

$$\mathbf{x_t} = \begin{bmatrix} BTC_{Open}, BTC_{High}, BTC_{Low}, BTC_{Close}, BTC_{Vol}, \\ NQ_{Open}, NQ_{High}, NQ_{Low}, NQ_{Close}, NQ_{Vol} \end{bmatrix}_t \quad (1)$$

where BTC denotes Bitcoin indicators and NQ denotes NASDAQ indicators. The primary target variable for the forecasting task is the Bitcoin closing price (BTC_{Close}).

Given the scale differences between the variables—where Bitcoin prices are in the tens of thousands, NASDAQ index points are in the thousands, and trading volumes are in the millions—data normalization is essential to ensure the stability and convergence of the gradient descent optimization. We applied Min-Max Scaling to transform all 10 input features into a normalized range of $[0,1]$, defined as in Equation (2).

$$x_{Scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2)$$

To prevent data leakage, the scaler was fitted exclusively on the training dataset, and the computed parameters (x_{min}, x_{max}) were subsequently applied to transform the validation and testing sets.

3.2. Sliding window data structuring

To adapt the multivariate time series for supervised learning, we employ a sliding window technique. Let L and H denote the look-back window size and the forecasting horizon, respectively. The model maps a sequence of historical observations $\mathbf{X_t} \in R^{L \times F}$ (where $F = 10$ features) to a future price trajectory $\mathbf{Y_t} \in R^{H}$. We evaluate three temporal configurations to capture short, medium, and long-term dependencies: $(L,H) \in \{(10,3),(15,5),(20,7)\}$. The procedure for generating these input-output pairs is formalized in Algorithm 1.

Algorithm 1. Sliding window data construction

Input:
$\mathscr{D}$: Normalized multivariate time series of shape (N,F).
L: Lookback window size.
H: Forecast horizon.
idx: Index of target variable (Bitcoin Close Price).
Output:
$\mathscr{X}$: Input tensor of shape (M,L,F)
$\mathscr{Y}$: Target matrix of shape (M,H)

Procedure:
1: $\mathscr{X} \leftarrow \phi$, $\mathscr{Y} \leftarrow \phi$
2: $M \leftarrow N - L - H + 1$ (Calculate total samples)
3: For $t \leftarrow 0$ to $M - 1$ do:
4: $\mathbf{x_t} \leftarrow \mathscr{D}[t : t + L, :]$ (Extract input window)

5: $\mathbf{y_t} \leftarrow \mathscr{D}[t + L{:}t + L + H, idx]$ (Extract target horizon)
6: Append $\mathbf{x_t}$ to $\mathscr{X}$
7: Append $\mathbf{y_t}$ to $\mathscr{Y}$
8: end for
9: return Tensor($\mathscr{X}$), Matrix($\mathscr{Y}$)

3.3. Proposed Hybrid 1D-CNN - LSTM architecture

This study proposed a hybrid forecasting framework that integrates a 1D-CNN with a LSTM network to enhance the prediction of Bitcoin closing price dynamics. This two-stage architecture was designed to first extract salient features from the input time series and then model the temporal relationships between them. A general diagram of this architecture was shown in Fig. 1.

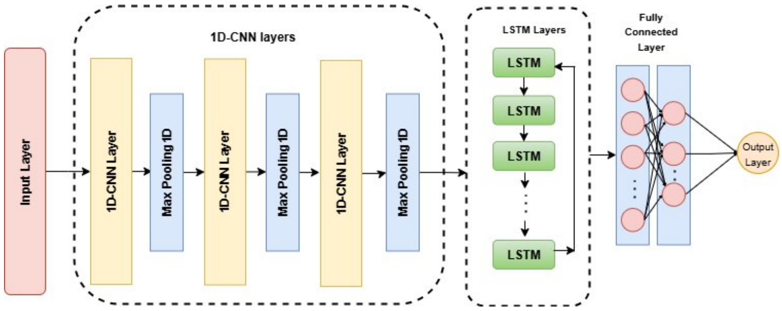

Fig. 1. The structure of the proposed hybrid 1D-CNN-LSTM model

The first stage of the proposed architecture consists of the 1D-CNN layers. At this stage, a set of learnable filters (kernels) slide across the input window to capture localized patterns in the time series. Formally, the convolution operation for the filter k is defined as shown in Equation (5):

$$Z_t^{(k)} = \sum_{i=0}^{m-1} X_{t+i}.W_i^{(k)} + b^{(k)} \tag{5}$$

where:

$Z_t^{(k)}$ denotes the activation at time t in the layer k.

xt+i represents the input at position t+i

$W_i^{(k)} \in \mathbb{R}^D$ are the learnable weights of the position i in the kernel.

$b^{(k)}$is the bias term.

m is the kernel size

The output of the convolution is then passed through a non-linear activation function, typically the Rectified Linear Unit (ReLU), formally expressed as defined in Equation (6):

$$h_t^{(k)} = ReLU\left(Z_t^{(k)}\right) = \max\left(0, Z_t^{(k)}\right) \tag{6}$$

To further refine the representation, a MaxPooling operation is subsequently applied, which downsamples the feature maps by selecting the maximum value within each pooling window, as shown in (7):

$$P_j^{(k)} = max\left\{h_s^{(k)}, h_{s_1+1}^{(k)}, \dots, h_{s+p-1}^{(k)}\right\} \tag{7}$$

where p denotes the pooling size and s is the stride.
This operation reduces the dimensionality of the feature maps while retaining the most salient information. Consequently, the 1D-CNN stage produces condensed high-level representations, enhancing computational efficiency and robustness against minor fluctuations in the input data.

The output of the 1D-CNN stage is a sequence of condensed, high-level feature representations, which are subsequently provided as input to the second stage comprising LSTM layers. LSTM networks represent an advanced RNN architecture specifically designed to mitigate the vanishing gradient problem inherent in conventional RNNs. By employing memory cells with gated mechanisms, LSTMs are capable of effectively capturing and preserving long-term temporal dependencies within sequential data [25].

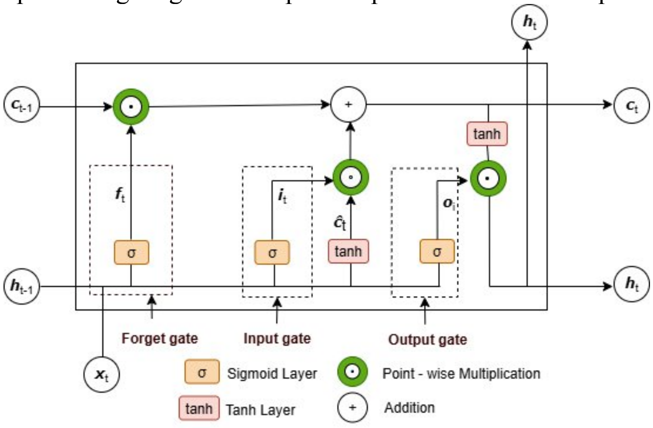

Fig.2. The Architecture of LSTM cell

As depicted in Fig.2, the architecture of an LSTM cell incorporates three primary gates—namely, the input gate, forget gate, and output gate—working in conjunction with the cell state (C_t). The cell state serves as a persistent memory pathway, enabling selective retention and discarding of information across time steps [19]. The mathematical formulation at each time step t is as follows:
The forget gate computation is given in Equation (8):

$$F_t = \sigma\ (W_f X_t + U_f H_{t-1} + b_f) \tag{8}$$

The input gate and candidate cell state update are defined in Equation (9), and (10):

$$I_t = \sigma\ \left(W_i X_t + U_i H_{t-1} + b_i\right) \tag{9}$$

$$\tilde{C}_t = tanh\ (W_c X_t + U_c H_{t-1} + b_c) \tag{10}$$

The cell state update is described in Equation (11):

$$C_t = \ F_t\ \odot\ C_{t-1} + I_t \odot \tilde{C}_t \tag{11}$$

The output gate and hidden state are finally computed as shown in Equation (12), and (13):

$$O_t = \sigma\ (W_0 X_t +\ U_0 H_{t-1} +\ b_0) \tag{12}$$

$$H_t = O_t\ \odot \tanh\ (C_t\ ) \tag{13}$$

where:

X_t: input vector at time step t

H_{t-1}: hidden state from the previous time step

C_{t-1}: cell state from the previous time step
F_t, I_t, O_t: activations of forget, input, and output gates
$\tilde{C}_t$: candidate cell state
W_0, U_0, b_0: trainable weight matrices and biases
σ: sigmoid activation function
tanh: hyperbolic tangent activation
$\odot$: element-wise multiplication

Following the recurrent stage, the output is passed to a Dense (fully connected) layer, which performs the final mapping of the learned temporal features to the required output dimension. This mapping facilitates the generation of multi-step forecasts of Bitcoin's closing price, specifically for horizons of 3, 5, or 7 days, aligning with the predictive objectives of the study.

3.4. Evaluation Metrics

To quantitatively assess the performance and predictive accuracy of our proposed hybrid models, we employed a set of four widely recognized statistical metrics with formulas listed in Equations (14 - (17). These metrics provided a comprehensive view of the model's errors and its goodness of fit.

$$MSE(y, \hat{y}) = \frac{1}{N}\sum_{i=0}^{N-1}\left(y_i - \hat{y}_i\right)^2 \tag{14}$$

$$MAE(y, \hat{y}) = \frac{1}{N}\sum_{i=0}^{N-1}\left|y_i - \hat{y}_i\right| \tag{15}$$

$$RMSE(y, \hat{y}) = \sqrt{MSE(y, \hat{y})} \tag{16}$$

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=0}^{N-1}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=0}^{N-1}\left(y_i - \bar{y}_i\right)^2} \tag{17}$$

where:

N is the total samples;
y and y^are true and predicted values, respectively; and
$\bar{y} = \frac{1}{N}\sum_{i=0}^{N-1} y_i$.

Together, these four metrics provide a robust framework for comparing the forecasting performance of the different models.

4. Experiment

4.1. Data Collection and Preprocessing

We collected historical data for Bitcoin (BTC-USD) and the NASDAQ Composite Index (^IXIC) from Yahoo Finance[1] via the yfinance Python library. The dataset cov-

[1] https://finance.yahoo.com

ers the period from January 1, 2015, to July 14, 2025. Each series includes the attributes Open, High, Low, Close, and Volume. The Close price was selected as the primary variable for prediction.

Since the cryptocurrency market trades continuously while NASDAQ operates only on weekdays, missing NASDAQ values on non-trading days were filled using the forward-fill method, which replaces each missing value with the most recent available observation. Rows with remaining null values were then removed.

Time series were transformed into supervised learning input-output pairs using a sliding window approach. Three forecasting configurations were prepared: 10 input days to predict 3 output days (10–3), 15 input days to predict 5 (15–5), and 20 input days to predict 7 (20–7).

4.2. Empirical Validation of Cross-Market Linkages

To justify integrating NASDAQ signals into the Bitcoin forecasting model, we conducted a three-stage empirical analysis on logarithmic returns to validate the spillover hypothesis. First, we assessed the linear dependency via Fig. 3, which illustrated the

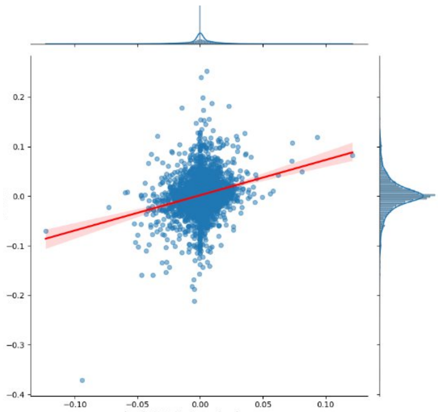

Fig. 3. Scatter plot of daily log returns: Bitcoin vs. NASDAQ

scatter plot of daily log returns for Bitcoin and NASDAQ. The most critical feature was the red linear regression trendline, which exhibited a distinct positive slope. This upward trajectory empirically confirmed a systemic spillover effect: as NASDAQ returns increased, Bitcoin returns generally trended upward. The regression line served as a proxy for market beta, indicating that Bitcoin was not an isolated asset but possessed significant sensitivity to investor sentiment in the traditional technology sector. Second, the correlation between the two markets was then explicitly examined. Fig. 4 displayed a heatmap of the Pearson correlation coefficient between the returns, revealing positive linear relationship with a Pearson coefficient of 0.23. To understand the time-varying nature of this relationship, we computed a 90-day rolling correlation, as shown in Fig. 5. The plot confirmed that the correlation was dynamic, not static, with periods where the linkage strengthened significantly, particularly during times of market stress or major economic events. This dynamic correlation highlighted the potential for NASDAQ signals to serve as a valuable predictive feature.

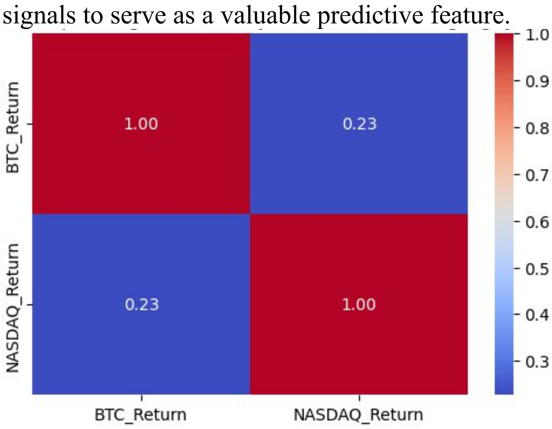

Fig. 4. Correlation heatmap of returns.

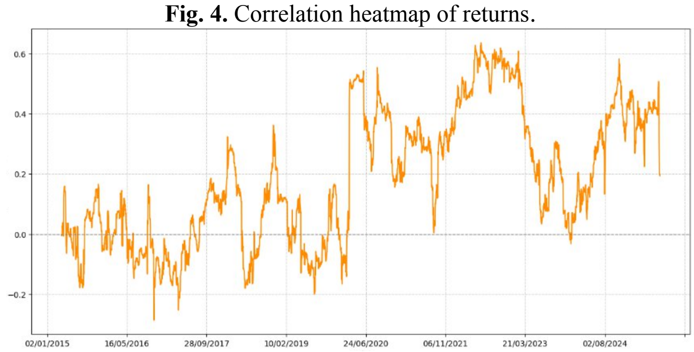

Fig. 5. 90-day rolling correlation between BTC and NASDAQ returns

Collectively, these findings provided robust empirical evidence that NASDAQ returns contained valuable predictive signals, validating their inclusion as exogenous features in our proposed hybrid framework. This empirical foundation underpinned our multivariate experimental design, where we aimed to leverage these systemic spillover effects to enhance the model's predictive accuracy and robustness across varying forecasting horizons.

4.3. Forecasting performance

We evaluated the performance of four hybrid models (1D-CNN - RNN, 1D-CNN - LSTM, 1D-CNN - GRU, and 1D-CNN - Bi-LSTM) on three forecasting horizons: 3-day, 5-day, and 7-day. The comprehensive results were summarized in Table 1, and the training/validation loss curves and prediction plots were shown in Fig. 6 - Fig. 11.

Table 1. Performance evaluation of hybrid models across different configurations

Config	Model	MSE	RMSE	MAE	R^2
(10,3)	1D-CNN - RNN	0.000438	0.01485	0.020936	0.994274
	1D-CNN - LSTM	0.000535	0.01665	0.023135	0.993008
	1D-CNN - GRU	0.000424	0.014772	0.020585	0.994464

	1D-CNN - Bi-LSTM	0.000441	0.015275	0.021011	0.994233
(15,5)	1D-CNN - RNN	0.000593	0.017815	0.024347	0.992193
	1D-CNN - LSTM	0.00045	0.014968	0.021217	0.994071
	1D-CNN - GRU	0.000566	0.017331	0.023785	0.992549
	1D-CNN - Bi-LSTM	0.00059	0.018355	0.024291	0.992229
(20,7)	1D-CNN - RNN	0.000489	0.016092	0.022116	0.993519
	1D-CNN - LSTM	0.000428	0.015094	0.020696	0.994324
	1D-CNN - GRU	0.000486	0.015533	0.022053	0.993556
	1D-CNN - Bi-LSTM	0.000625	0.018327	0.025004	0.991716

For the 3-day forecasting horizon, the results indicated that the 1D-CNN - GRU model achieved the lowest error metrics (MSE, RMSE, MAE) and the highest R2 score. Fig. 6 showed stable convergence for all models during training. The prediction plot in Fig. 7 confirmed that the 1D-CNN - GRU model's predictions tracked the true values more closely than the other architectures, particularly in capturing the peaks and troughs.

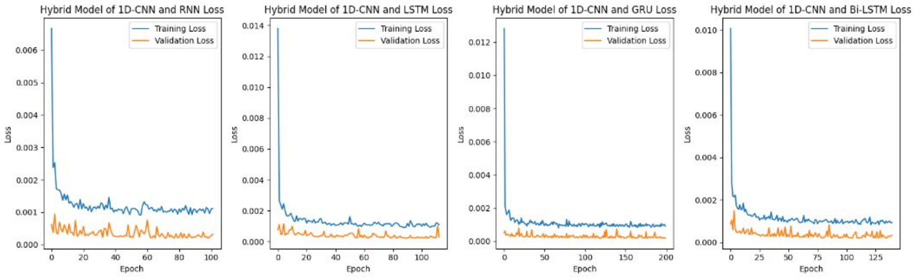

Fig. 6. Training & validation loss for 3-day models

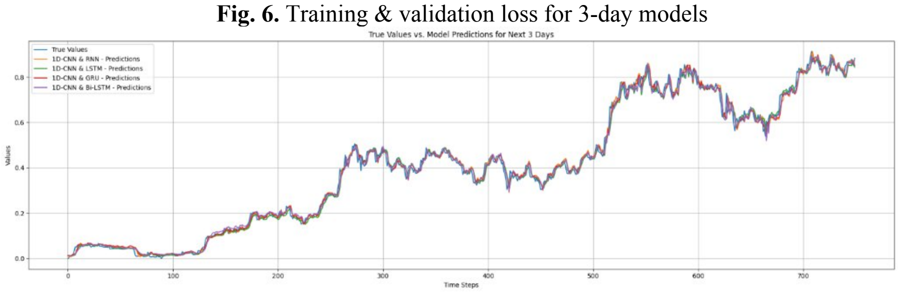

Fig. 7. Prediction results for the 3-day horizon

For longer forecasting horizons (5 and 7 days), the 1D-CNN–LSTM model consistently outperformed the other hybrid model s, as shown in Table 1. This result highlighted the LSTM's superior ability to model long-term dependencies, whereas the Bi-LSTM suffered from slight overfitting, reflected in its higher error metrics. The loss curves (Fig. 8 and Fig. 10) remained stable, indicating that the models did not exhibit severe overfitting despite the increased task complexity. The prediction plots (Fig. 9 and Fig. 11) illustrated that while forecasting further into the future became inherently more difficult, the 1D-CNN - LSTM model provided the most accurate trend predictions compared to the other models. This suggests that the memory-cell mechanism of LSTM was more effective at capturing longer-term dependencies in the financial time series data.

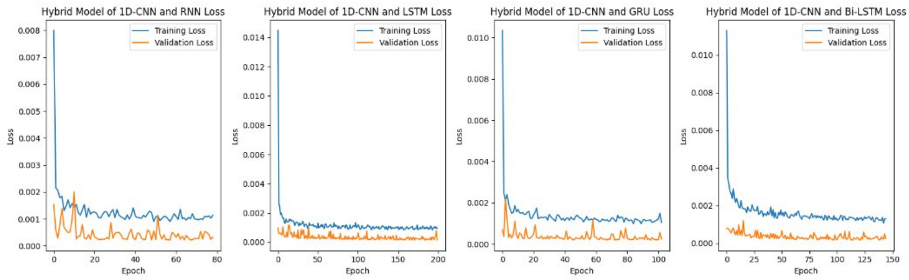

Fig. 8. Training & validation loss for 5-day models

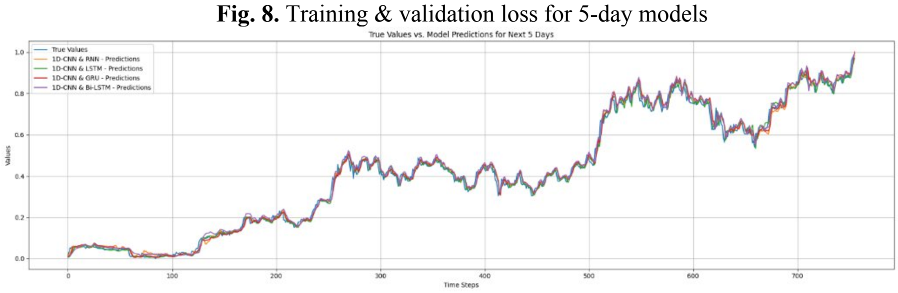

Fig. 9. Prediction results for the 5-day horizon

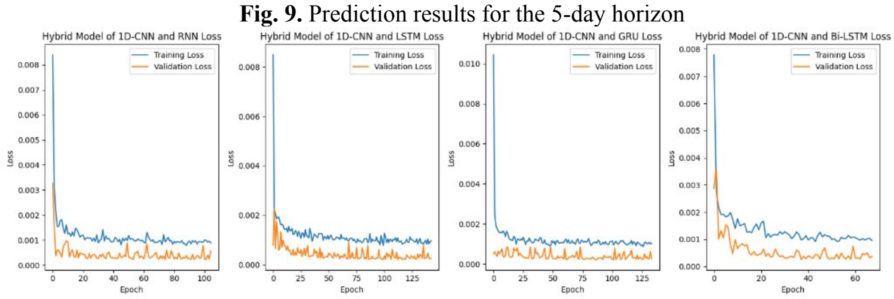

Fig. 10. Training & validation loss for 7-day models

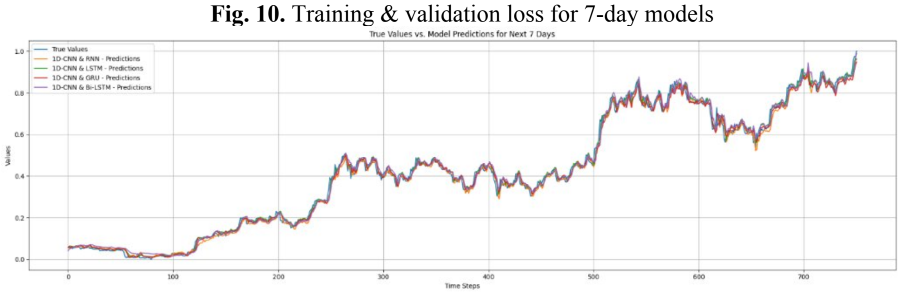

Fig. 11. Prediction results for the 7-day horizon

The experimental findings demonstrated three important insights. First, correlation analysis validated the hypothesis that the cryptocurrency market was linked to traditional financial markets, with NASDAQ signals providing measurable predictive value. Second, the hybrid 1D-CNN - LSTM framework successfully leveraged local feature extraction and long-range sequence modeling, resulting in superior accuracy for medium- and long-term horizons. Third, although GRU performed slightly better in the 3-day setting, its performance degraded as the forecast horizon extended, whereas

LSTM maintained consistency. This suggested that GRU may be preferable for very short horizons [26], but LSTM offered a more robust solution across different time scales.
Overall, the results highlighted the effectiveness of the 1D-CNN - LSTM hybrid model in multi-step Bitcoin forecasting and confirmed the role of cross-market information in enhancing prediction accuracy.

5. Conclusion

This study proposed a hybrid 1D-CNN–LSTM framework for Bitcoin price forecasting, explicitly integrating NASDAQ signals to capture cross-market spillover effects. Correlation analysis confirmed a dynamic and significant relationship between Bitcoin and NASDAQ returns, supporting the inclusion of exogenous financial indicators. Experimental results across multiple forecasting horizons demonstrated that the 1D-CNN – LSTM model consistently achieved the most robust performance for medium- and long-term predictions while remaining competitive in short-term horizons.

Despite these promising results, several avenues remain for future research to address the limitations of the current study. First, while statistical metrics (MSE, RMSE) validate the model's accuracy, future work will incorporate economic evaluation metrics, such as a trading simulation to calculate Return on Investment (ROI), to assess the practical profitability of the forecasts. Second, to rigorously validate the superiority of the proposed hybrid architecture, we aim to conduct statistical significance tests (e.g., the Diebold-Mariano test) and expand our comparative analysis to include standard baselines (e.g., ARIMA, pure LSTM) and modern Transformer-based models. Furthermore, we intend to broaden the input feature space by incorporating macroeconomic indicators and market sentiment data to capture a wider range of price drivers. Finally, to mitigate the black-box nature of deep learning, explainability techniques such as SHAP (SHapley Additive exPlanations) values will be integrated to visualize specifically how and when NASDAQ spillover signals influence Bitcoin price dynamics,. In summary, the proposed approach offers practical value for cryptocurrency forecasting, and addressing these future directions will further bridge the gap between theoretical modeling and real-world financial application.

References

1. I. Nasirtafreshi, “Forecasting cryptocurrency prices using Recurrent Neural Network and Long Short-term Memory,” *Data & Knowledge Engineering*, vol. 139, p. 102009, May 2022, doi: 10.1016/j.datak.2022.102009.
2. W. Chen, W. Hussain, F. Cauteruccio, and X. Zhang, “Deep Learning for Financial Time Series Prediction: A State-of-the-Art Review of Standalone and Hybrid Models,” Dec. 2023, doi: 10.32604/cmes.2023.031388.

3. S. M. Al-Selwi *et al.*, "RNN-LSTM: From applications to modeling techniques and beyond—Systematic review," *Journal of King Saud University - Computer and Information Sciences*, vol. 36, no. 5, p. 102068, June 2024, doi: 10.1016/j.jksuci.2024.102068.
4. S. Das, A. Tariq, T. Santos, S. S. Kantareddy, and I. Banerjee, "Recurrent Neural Networks (RNNs): Architectures, Training Tricks, and Introduction to Influential Research," in *Machine Learning for Brain Disorders*, Humana, New York, NY, 2023, pp. 117–138. doi: 10.1007/978-1-0716-3195-9_4.
5. C. Zhong, W. Du, W. Xu, Q. Huang, Y. Zhao, and M. Wang, "LSTM-ReGAT: A network-centric approach for cryptocurrency price trend prediction," *Decision Support Systems*, vol. 169, p. 113955, June 2023, doi: 10.1016/j.dss.2023.113955.
6. M. Cheon, H. Choi, H. Joo, and O. Lee, "Disclosing the Relationship Between Nasdaq and Bitcoin via Artificial Intelligence Approach: Machine Learning and Deep Learning," 2023, pp. 1053–1062. doi: 10.52458/978-81-955020-5-9-100.
7. F. Moodi, A. Jahangard Rafsanjani, S. Zarifzadeh, and M. A. Zare Chahooki, "Advanced Stock Price Forecasting Using a 1D-CNN-GRU-LSTM Model," *Journal of AI and Data Mining*, vol. 12, no. 3, pp. 393–408, July 2024, doi: 10.22044/jadm.2024.14831.2581.
8. S. Cavalli and M. Amoretti, "CNN-based multivariate data analysis for bitcoin trend prediction," *Applied Soft Computing*, vol. 101, p. 107065, 2021.
9. J. Zhang, K. Cai, and J. Wen, "A survey of deep learning applications in cryptocurrency," *iScience*, vol. 27, no. 1, p. 108509, Jan. 2024, doi: 10.1016/j.isci.2023.108509.
10. A. Sagheer, A. Raza, M. R. Rashid, and F. Kiran, "A Hybrid Deep Learning Model for Accurate Bitcoin Price Forecasting," *Lahore Garrison University Research Journal of Computer Science and Information Technology*, vol. 9, no. 1, pp. 42–51, Aug. 2025, doi: 10.54692/lgurjcsit.2025.91663.
11. A. Bouteska, M. Z. Abedin, P. Hajek, and K. Yuan, "Cryptocurrency price forecasting – A comparative analysis of ensemble learning and deep learning methods," *International Review of Financial Analysis*, vol. 92, p. 103055, Mar. 2024, doi: 10.1016/j.irfa.2023.103055.
12. S. Giantsidi and C. Tarantola, "Deep learning for financial forecasting: A review of recent trends," *International Review of Economics & Finance*, vol. 104, p. 104719, Dec. 2025, doi: 10.1016/j.iref.2025.104719.
13. A. B. Wardak and J. Rasheed, "Bitcoin Cryptocurrency Price Prediction Using Long Short-Term Memory Recurrent Neural Network," *EJOSAT*, no. 38, Art. no. 38, Aug. 2022, doi: 10.31590/ejosat.1079622.
14. D. M. Gunarto, S. Sa'adah, and D. Q. Utama, "Predicting cryptocurrency price using rnn and lstm method," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 12, no. 1, pp. 1–8, 2023.
15. I. P. B. Priadinata, I. G. I. Sudipa, N. P. S. Meinarni, I. M. L. Radhitya, and I. K. D. G. Supartha, "Comparative Analysis of LSTM, GRU, and Bi-LSTM Deep Learning Models for Time Series Cryptocurrency Price Forecasting," *Sinkron : jurnal dan penelitian teknik informatika*, vol. 9, no. 3, pp. 1024–1035, July 2025, doi: 10.33395/sinkron.v9i3.14795.

16. I. E. Livieris, “A novel forecasting strategy for improving the performance of deep learning models,” *Expert Systems with Applications*, vol. 230, p. 120632, Nov. 2023, doi: 10.1016/j.eswa.2023.120632.
17. S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, “1D convolutional neural networks and applications: A survey,” *Mechanical Systems and Signal Processing*, vol. 151, p. 107398, Apr. 2021, doi: 10.1016/j.ymssp.2020.107398.
18. Z. Zeng, R. Kaur, S. Siddagangappa, S. Rahimi, T. Balch, and M. Veloso, “Financial Time Series Forecasting using CNN and Transformer,” Apr. 11, 2023, *arXiv*: arXiv:2304.04912. doi: 10.48550/arXiv.2304.04912.
19. 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, Art. no. 5, Oct. 2023, doi: 10.48084/etasr.6223.
20. J. Cahyadi and A. Zahra, “Bitcoin Price Prediction Model Development Using Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM),” *Journal La Multiapp*, vol. 5, no. 2, Art. no. 2, Mar. 2024, doi: 10.37899/journallamultiapp.v5i2.1070.
21. Ferdiansyah, R. Zahilah, S. Hajar, and D. Stiawan, “CNN-LSTM Hybrid Model for Improving Bitcoin Price Prediction Results,” *Applied Mathematics and Computational Intelligence (AMCI)*, vol. 12, no. 4, Art. no. 4, Nov. 2023, doi: 10.58915/amci.v12i4.349.
22. T. Sukomardojo, M. Pamikatsih, Y. Arpianto, A. Nuraini, and E. Fatmawati, “Cryptocurrency and Macro-Economic Stability: Impacts and Regulations,” *International Journal of Science and Society*, vol. 5, no. 4, pp. 734–745, Aug. 2023, doi: 10.54783/ijsoc.v5i4.842.
23. Dane Vanderbilt, Kun Xie, and Wenying Sun, “An Applied Study of Rnn Models for Predicting Cryptocurrency Prices,” *IIS*, 2020, doi: 10.48009/2_iis_2020_135-143.
24. A. Farooq, M. I. Uddin, M. Adnan, A. A. Alarood, E. Alsolami, and S. Habibullah, “Interpretable multi-horizon time series forecasting of cryptocurrencies by leverage temporal fusion transformer”, Accessed: Dec. 03, 2025. [Online]. Available: https://www.cell.com/heliyon/abstract/S2405-8440(24)16173-7
25. G. Van Houdt, C. Mosquera, and G. Nápoles, “A review on the long short-term memory model,” *Artificial Intelligence Review*, vol. 53, no. 8, Art. no. 8, May 2020, doi: 10.1007/s10462-020-09838-1.
26. P. Deka, M. Cordeiro-Costas, R. Pérez-Orozco, M. Chabiński, and A. Szlęk, “Novel NSGA-II optimized LSTM and GRU models for short-term forecasting of residential heating load,” *Energy and Buildings*, vol. 344, p. 115999, Oct. 2025, doi: 10.1016/j.enbuild.2025.115999.