Effect of Technical Indicator Combinations on the Performance of a Hybrid CNN-LSTM Model for Bitcoin Price Direction Classification

Frederikus Felix Bintang Setyawan[1*], Lailil Muflikhah[2], Rizal Setya Perdana[3]
[1, 2, 3]Department of Informatics Engineering, Faculty of Computer Science, University of Brawijaya, Malang, Indonesia
[1]bintangfelix55@student.ub.ac.id, [2]lailil@ub.ac.id, [3]rizalespe@ub.ac.id

Abstract. Bitcoin price direction classification is challenging due to high volatility and complex temporal dynamics. This study investigates the contribution of four technical indicator categories-Momentum (MOM30), Relative Strength Index (RSI14, RSI30), Stochastic Oscillator (%K30, %D30, %K200, %D200), and Trend-Following (MACD)-within a hybrid CNN-LSTM model. Eight selected technical indicators are combined with OHLCV variables and evaluated through a systematic ablation study comprising six configurations. Minute-level BTC-USD data from January 2023 to January 2025 are resampled into 15-minute intervals, with MA10/MA60 used to generate the binary target. To reduce look-ahead bias, each input sequence contains only the preceding 32 observations, while the target corresponds to the subsequent timestep. Model hyperparameters are selected using Random Search and fixed across experiments. The Full Model achieves 97.74% accuracy and 99.78% AUC-ROC, while the OHLCV-only Baseline achieves 55.88% accuracy. Trend-Following shows the largest observed marginal contribution (+0.65 percentage points), followed by Momentum (+0.38) and RSI (+0.22), whereas removing the Stochastic Oscillator yields a slight accuracy increase (+0.17). These results provide empirical evidence on the relative contribution of the evaluated indicator categories within the adopted CNN-LSTM setting.

Keywords: Bitcoin · CNN-LSTM · Deep Learning · Technical Indicators · Ablation Study · Price Direction Classification.

1 Introduction

The rapid evolution of blockchain and cryptocurrency markets has transformed global finance, with Bitcoin emerging as a prominent digital asset since its introduction as a peer-to-peer electronic cash system in 2008 [9]. In Indonesia, cryptocurrency investors increased from approximately 18.61 million in October 2025 to 19.08 million in November 2025 (2.5%), while cumulative transaction value reached IDR 446.77 trillion in 2025 [11]. Bitcoin's high volatility and changing market dynamics make price direction classification challenging [6,7]. OHLCV data are commonly used in financial time-series prediction, with technical indicators derived as predictive features. Previous studies have applied CNN-LSTM to Bitcoin prediction, including Li and Dai [5], who reported better forecasting performance than individual neural networks, and Qiang and Shen [13], who combined 30 technical indicators with CNN-LSTM for high-frequency trend prediction. Hafid et al. [3], Omole and Enke [10], and Badar et al. [1] further examined feature selection, on-chain variables, and feature extraction and interpretation using CNN-LSTM.

© The Author(s) 2026
M. Murtadha et al. (eds.), *Proceedings of the 2026 11th International Conference on Sustainable Information Engineering and Technology (SIET 2026)*, Atlantis Highlights in Computer Sciences 26,
https://doi.org/10.2991/978-94-6239-828-3_30

Despite these studies, comparatively less attention has been given to the contribution of individual technical indicator categories within an identical architecture. It remains unclear whether different categories provide complementary information or introduce redundant features. This study therefore evaluates Momentum, RSI, Stochastic Oscillator, and Trend-Following through a controlled ablation study using identical CNN-LSTM architecture, preprocessing, hyperparameters, and training protocol. Using 15-minute BTC-USD data, six configurations are evaluated: a Full Model with OHLCV and eight selected indicators, four ablation models removing one category at a time, and an OHLCV-only Baseline. The study aims to evaluate CNN-LSTM classification performance and quantify the contribution of each indicator category, providing evidence that may inform feature engineering decisions.

2 Related Work

Early studies showed that CNN-LSTM combines CNN-based local feature extraction with LSTM-based temporal dependency modelling for Bitcoin prediction. Li and Dai [5] integrated historical Bitcoin prices with macroeconomic variables and investor attention indices, reporting better forecasting and direction prediction than individual neural networks; however, their framework used only three technical indicators (RSI, MFI, and OBV) and a relatively limited testing dataset. Livieris et al. [6] introduced a multi-input CNN-LSTM using multiple cryptocurrencies and an adaptive design to reduce overfitting, while Peng et al. [12] incorporated attention to model relationships across cryptocurrencies and temporal resolutions for high-frequency prediction. Feature engineering and selection were also investigated: Hafid et al. [3] combined technical indicators, chi-squared selection, and XGBoost, identifying eight indicators for 15-minute Bitcoin classification; Dubey and Enke [2] combined on-chain variables and Boruta selection with CNN-LSTM; and Omole and Enke [10] compared deep learning architectures under identical conditions and examined CNN-LSTM using Boruta-selected features. Badar et al. [1] further integrated a variational autoencoder and SHAP with CNN-LSTM for feature extraction and interpretation. Overall, these studies focus on architecture, additional information, and feature selection, while comparatively less attention has been given to isolating individual technical indicator categories within an identical CNN-LSTM architecture. This study therefore examines Momentum, RSI, Stochastic Oscillator, and Trend-Following through controlled ablation while keeping the architecture, preprocessing, hyperparameters, and training protocol consistent.

3 Theoretical Background

3.1 Technical Indicators

Moving Average Crossover (Target Variable). The binary target is generated using the Moving Average Crossover (MAC) strategy [3]. MA10 and MA60 are calculated from Close prices using historical observations available up to each timestep, with windows of 10 periods (2.5 hours) and 60 periods (15 hours), respectively. A Buy signal is assigned

when MA10 ≥ MA60 and Sell otherwise, as defined in Eq. (1). The target and input sequences use only historical information, preventing look-ahead bias.

$$Y(t_i) = \begin{cases} 1, & if\ MA_{s,t_i} \geq MA_{l,t_i} \\ 0, & if\ MA_{s,t_i} < MA_{l,t_i} \end{cases} \quad (1)$$

Relative Strength Index (RSI). RSI measures the magnitude of recent price changes on a 0-100 scale; values above 70 generally indicate overbought and below 30 oversold conditions. This study uses RSI14 and RSI30 to represent short- and medium-term momentum, respectively [3].

Momentum (MOM30). MOM30 measures the rate of price change: MOM(t) = Close(t) - Close($t - 30$). Positive values signal bullish inertia; negative values signal bearish inertia [3].

Stochastic Oscillator (%K, %D). The Stochastic Oscillator compares the current Close with the highest and lowest prices over a specified look-back period to identify overbought, oversold, and potential trend-reversal conditions. The study uses %K30, %D30, %K200, and %D200 to represent shorter- and longer-term dynamics [3].

Moving Average Convergence Divergence (MACD). MACD measures the relationship between the 12-period and 26-period EMAs, with a 9-period EMA of the MACD line as its signal line; their crossovers are commonly used to indicate potential changes in trend direction [3].

3.2 CNN-LSTM Hybrid Architecture

CNN-LSTM combines CNN-based local temporal feature extraction with LSTM-based sequential dependency modelling. Conv1D applies learnable filters to sequential data, while LSTM models dependencies across timesteps, allowing the hybrid architecture to represent both local and longer-range temporal patterns. CNN-LSTM has been applied in Bitcoin and cryptocurrency forecasting studies, supporting its use as the backbone architecture in this study [5,6].

4 Methodology

4.1 Dataset and Preprocessing

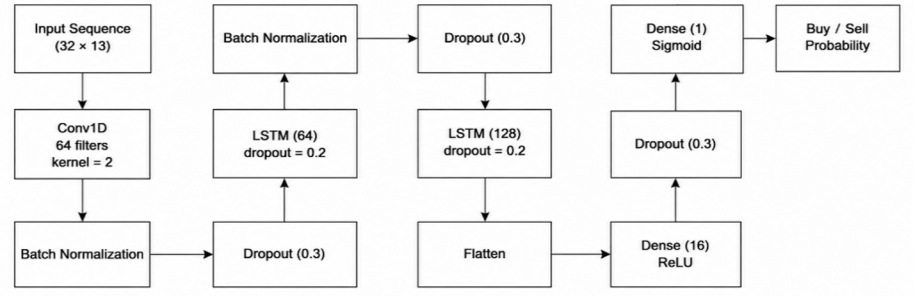
Bitcoin (BTC-USD) minute-level OHLCV data from 1 January 2023 to 31 January 2025 were obtained from the Kaggle dataset *btcusd_1-min_data.csv*, containing approximately 1,095,841 observations. Following Hafid et al. [3], the data were resampled into 15-minute intervals using standard OHLCV aggregation, producing 73,057 observations. Eight technical indicators (RSI14, RSI30, MOM30, MACD, %K30, %D30, %K200, and %D200) were calculated using TA-Lib. The binary target was generated using the Moving Average Crossover strategy in Section 3.1, with MA10 and MA60 calculated only from Close prices available up to each timestamp. The target therefore does not use future price information, and the Close value used for target generation is excluded from the corresponding input sequence. After removing missing values from

indicator warm-up periods, 72,854 observations remained, comprising 13 features: five OHLCV variables and eight technical indicators. The data were sequentially split without shuffling into training (70%), validation (10%), and testing (20%) sets before sequence generation. RobustScaler was fitted only on the training set and then applied to validation and testing sets. A sliding window of $\tau = 32$ timesteps, equivalent to eight hours, was used to construct each input sequence. This length was selected through Random Search (Section 4.3). As shown in Fig. 2, each sequence contains the preceding 32 observations and predicts the Moving Average Crossover label at the following timestep. The resulting sets contained 51,107, 7,274, and 14,580 samples, respectively, with input shape (samples, 32, 13). The class distribution was 52.7% Buy and 47.3% Sell; therefore, no additional class balancing was applied.

4.2 Model Architecture

The CNN-LSTM model combines CNN-based local temporal feature extraction with LSTM-based temporal dependency modelling [5,6,10]. The input shape is (32, 13). As shown in Fig. 1, Conv1D uses 64 filters, kernel size 2, same padding, and ReLU, followed by batch normalisation and dropout (0.3). Two stacked LSTM layers with 64 and 128 units use ReLU, internal dropout (0.2), and return sequences; batch normalisation and dropout (0.3) follow the first LSTM. The second LSTM output is flattened and passed to a Dense layer with 16 ReLU neurons and dropout (0.3), followed by a single sigmoid output neuron. The model uses Adam [4] with a learning rate of 0.0001 and binary cross-entropy loss. Training uses batch size 64 for up to 30 epochs. EarlyStopping monitors validation accuracy with patience = 5 and restores the weights with the highest validation accuracy.

Figure 1. Proposed CNN-LSTM Architecture



4.3 Hyperparameter Optimisation

Hyperparameter optimisation was performed using the Random Search algorithm over 20 trials on the Full Model configuration. The search space included the number of Conv1D filters ∈ {8, 16, 32, 64}, kernel size ∈ {2, 3, 5}, first LSTM units ∈ {32, 64, 128}, second LSTM units ∈ {64, 128, 256}, Dense units ∈ {8, 16, 32, 64}, external

dropout rate ∈ {0.3, 0.4, 0.5, 0.6}, learning rate ∈ {0.0001, 0.0005, 0.001}, batch size ∈ {16, 32, 64}, and sequence length ∈ {16, 32, 64}. Trial 14 achieved the best validation performance, and the corresponding hyperparameter configuration was subsequently fixed for all six ablation experiments. Consequently, all experimental configurations employed identical model architecture and training settings, ensuring that any observed performance differences could be attributed to differences in the input feature sets rather than changes in the model architecture or training configuration.

4.4 Ablation Study Design

The ablation study systematically removes one technical indicator category at a time, yielding six experimental configurations: (1) a Baseline using only OHLCV features (5 features); (2) a Full Model incorporating all 13 features; (3) a configuration without RSI, where RSI14 and RSI30 are excluded (11 features); (4) a configuration without Momentum, where MOM30 is removed (12 features); (5) a configuration without Oscillator, where %K30, %D30, %K200, and %D200 are excluded (9 features); and (6) a configuration without Trend-Following, where MACD is removed (12 features). The marginal contribution of indicator category X is quantified by Eq. (2):

$$Contribution(X) = Accuracy(Full\ Model) - Accuracy(Without\ X) \quad (2)$$

A positive value denotes a net performance gain from including category X; a negative value denotes degradation attributable to its inclusion.

4.5 Evaluation Metrics

Model performance is assessed using Accuracy, Precision, Recall, F1-score, and AUC-ROC, supplemented by confusion matrix analysis. These metrics are consistent with established standards for binary classification in financial signal prediction [3,5].

5 Results and Discussion

5.1 Verification of Target Label Construction

Table 1 presents representative consecutive observations illustrating the Moving Average Crossover (MAC) target construction, including OHLC values, MA10, MA60, RSI14, and the resulting labels. These observations are provided to illustrate the labelling procedure and do not constitute a separate evaluation dataset.

Table 1. Representative consecutive observations illustrating the Moving Average Crossover target generation procedure.

Timestamp	Open	High	Low	Close	MA10	MA60	RSI14	Label
2023-01-03 04:15:00	16709	16710	16704	16709	16721.6	16717.16	46.17	1
2023-01-03 04:30:00	16707	16709	16694	16697	16719.1	16716.55	41.62	1

2023-01-03 04:45:00	16698	16698	16665	16673	16714.2	16715.68	34.33	0
2023-01-03 05:00:00	16668	16680	16655	16680	16709.4	16715.05	37.757	0

As shown in Table 1, the labels are determined by comparing MA10 and MA60 at each timestamp. Both moving averages are calculated using only Close prices available up to the corresponding timestamp. Therefore, future price information is not used in target construction. RSI14 is included as a representative predictor and is not used to determine the target label.

5.2 Hyperparameter Optimisation Results

The 20 Random Search trials produced validation accuracies ranging from 53.68% to 98.02%. Trial 14 achieved the highest validation accuracy, using 64 Conv1D filters, kernel size 2, LSTM units of 64 and 128, 16 Dense units, dropout 0.3, learning rate 0.0001, batch size 64, and sequence length 32. This configuration was subsequently fixed for the Full Model and all ablation models to maintain consistent experimental conditions.

Figure 2. Training history of the proposed CNN-LSTM model using the selected hyperparameter configuration. The dashed vertical line indicates the epoch with the highest validation accuracy.

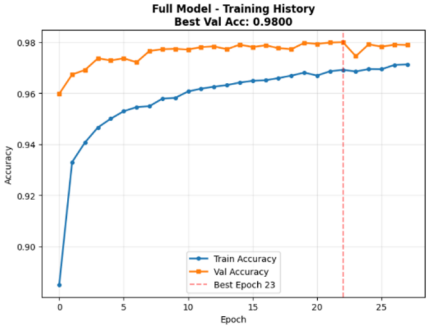


Figure 2 shows the training and validation accuracy across epochs. Validation accuracy reached approximately 98.00% at epoch 23 and remained relatively stable thereafter. The relatively small difference between training and validation accuracy indicates similar performance on the two sets under the adopted training setting; however, this observation alone does not establish the absence of overfitting.

5.3 Ablation Study Results

Table 2 presents the test performance of the six experimental configurations. Test sample sizes range from 14,539 to 14,612 because the indicator groups have different warm-up requirements. The same architecture, hyperparameters, preprocessing procedure, and training protocol were otherwise maintained across configurations.

Table 2. Ablation study evaluation results (test set, 14,539-14,612 samples).

Configuration	Feat.	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Without Oscillator	9	97.90%	98.57%	97.60%	98.09%	99.75%
Full Model (proposed)	13	97.74%	98.95%	96.91%	97.92%	99.78%
Without RSI	11	97.52%	98.06%	97.41%	97.74%	99.78%
Without Momentum	12	97.36%	97.12%	98.11%	97.61%	99.71%
Without Trend-Following	12	97.08%	97.36%	97.34%	97.35%	99.61%
Baseline (OHLCV only)	5	55.88%	56.70%	84.05%	67.72%	56.74%

The OHLCV-only Baseline achieved 55.88% accuracy, 56.70% precision, 84.05% recall, 67.72% F1-score, and 56.74% AUC-ROC. The Full Model achieved 97.74% accuracy, 98.95% precision, 96.91% recall, 97.92% F1-score, and 99.78% AUC-ROC. This corresponds to an absolute accuracy difference of 41.86 percentage points relative to the Baseline. Under the adopted experimental setting, the indicator-inclusive Full Model therefore achieved higher classification performance than the OHLCV-only Baseline. For the Full Model, 14,210 of 14,539 test samples were correctly classified, comprising 6,452 true negatives and 7,758 true positives, with 82 false positives and 247 false negatives. Among the indicator-inclusive configurations, the Without Oscillator model achieved the highest observed accuracy of 97.90%, slightly above the Full Model.

Figure 3. Confusion matrix of the proposed CNN-LSTM Full Model for Bitcoin price direction classification on the test set.

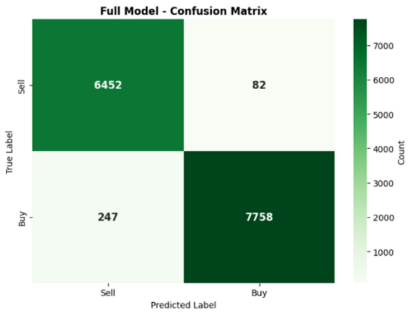


Figure 3 shows that most Buy and Sell observations in the Full Model were correctly classified. No statistical significance test was conducted. Therefore, differences among the

indicator-inclusive configurations are interpreted descriptively and are not treated as statistically significant effects.

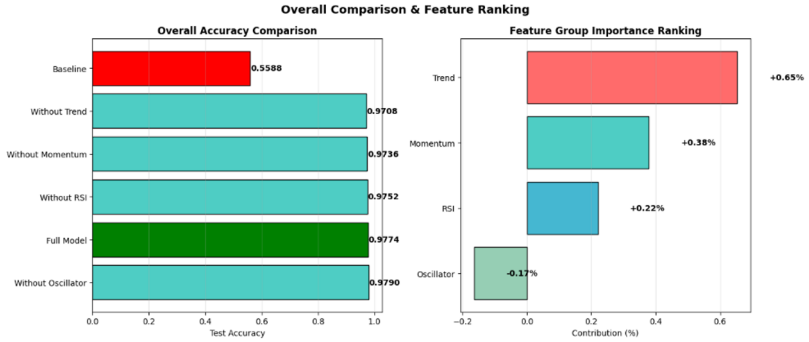
5.4 Contribution Analysis

Table 3 reports each category's marginal contribution as the accuracy difference between the Full Model and its corresponding ablation.

Table 3. Marginal contribution of each technical indicator category.

Category	Indicator(s)	Ablation Acc.	Δ Acc. (pp)
Trend-Following	MACD	97.08%	+0.65
Momentum	MOM30	97.36%	+0.38
RSI	RSI14, RSI30	97.52%	+0.22
Oscillator	%K30, %D30, %K200, %D200	97.90%	-0.17

Figure 4. Overall classification accuracy of the evaluated configurations and feature group contribution ranking obtained from the ablation study.



The observed accuracy differences were +0.65 percentage points for Trend-Following (MACD), +0.38 for Momentum, +0.22 for RSI, and −0.17 for Oscillator. Removing Trend-Following produced the largest observed decrease in accuracy, whereas removing Oscillator produced a slight increase. For Trend-Following, removing MACD increased false positives from 82 to 211, while false negatives decreased from 247 to 213. Removing Momentum increased false positives to 233, corresponding to the second-largest observed accuracy reduction. Removing RSI resulted in a 0.22-percentage-point decrease in accuracy. In contrast, removing the Oscillator group increased accuracy by 0.17 percentage points. One possible explanation is that these features may contain information overlapping with other indicators; however, feature redundancy was not directly evaluated in this study. The small difference should therefore be interpreted cautiously, particularly because no statistical significance test was performed. Overall,

these values represent the measured marginal contributions under the defined ablation procedure, rather than universal measures of technical indicator importance.

5.5 Practical Implications

The Without Oscillator configuration achieved the highest observed accuracy (97.90%) while using fewer input features than the Full Model. Within this experiment, this indicates that removing the Oscillator features did not reduce classification accuracy and was associated with a slight improvement. The other three indicator categories showed positive marginal accuracy differences when included according to the ablation definition. However, these findings are limited to one cryptocurrency (Bitcoin), one 15-minute prediction interval, one chronological train-validation-test split, and one optimised CNN-LSTM configuration. In addition, the target and technical indicators were derived from historical price data according to the adopted methodology. The results should therefore be interpreted as empirical observations for the selected dataset, target definition, and experimental setting, rather than evidence that similar performance or indicator contributions would necessarily occur under other datasets or market conditions. The absence of statistical significance testing also limits conclusions regarding the robustness of the relatively small differences between configurations. Future research could evaluate the same feature configurations across additional cryptocurrencies, market regimes, and prediction intervals; compare CNN-LSTM with CNN-only and LSTM-only architectures; incorporate other information sources such as on-chain data and news sentiment; and conduct repeated experiments with statistical significance testing. Trading-oriented backtesting using risk-adjusted measures such as Sharpe Ratio and Maximum Drawdown could also be used to examine the practical implications of the classification results.

6 Conclusion

This study systematically evaluated the contribution of four technical indicator groups within a CNN-LSTM framework for Bitcoin price direction classification using 15-minute BTC-USD data. Under the adopted experimental setting, the Full Model achieved 97.74% accuracy, 98.95% precision, and 99.78% AUC-ROC, representing an absolute accuracy improvement of 41.86 percentage points over the OHLCV-only Baseline. These results indicate that the selected technical indicators were associated with higher classification performance than OHLCV features alone for the defined target.

The ablation results showed that removing Trend-Following (MACD) produced the largest observed accuracy reduction (+0.65 percentage points), followed by Momentum (+0.38) and RSI (+0.22). In contrast, removing the Stochastic Oscillator increased accuracy by 0.17 percentage points, with the Without Oscillator configuration achieving the highest observed accuracy of 97.90% while using fewer features. This suggests that the Oscillator group provided limited additional benefit within the evaluated feature set; however, the observed differences are descriptive because no statistical significance test was conducted.

Overall, the findings provide empirical evidence on the relative contribution of the evaluated indicator groups within the adopted CNN-LSTM and experimental setting. Future research should assess these configurations across additional cryptocurrencies, market regimes, and prediction intervals; compare alternative architectures; incorporate information such as on-chain data and news sentiment; and use repeated experiments with statistical significance testing. Trading-oriented backtesting using measures such as Sharpe Ratio and Maximum Drawdown could further examine their practical applicability.

Acknowledgments. The author thanks Prof. Dr. Ir. Lailil Muflikhah, S.Kom., M.Sc., IPM, ASEAN Eng. and Rizal Setya Perdana, S.Kom., M.Kom., Ph.D. for their supervision and guidance throughout this research. The source code and preprocessing pipeline used in this study are publicly available at https://github.com/peder24/bitcoin-price-direction-cnn-lstm to facilitate the reproducibility of the reported results.

Disclosure of Interests. The author has no competing interests to declare that are relevant to the content of this article.

References

1. Badar, W., Ramzan, S., Raza, A., Fitriyani, N.L., Syafrudin, M., Lee, S.W.: Enhanced Interpretable Forecasting of Cryptocurrency Prices Using Autoencoder Features and a Hybrid CNN-LSTM Model. Mathematics 13(12), 1908 (2025). https://doi.org/10.3390/math13121908
2. Dubey, R., Enke, D.: Bitcoin Price Direction Prediction Using On-Chain Data and Feature Selection. Mach. Learn. Appl. 20, 100674 (2025). https://doi.org/10.1016/j.mlwa.2025.100674
3. Hafid, A., Rahouti, M., Kong, L., Ebrahim, M., Serhani, M.A.: Predicting Bitcoin Market Trends with Enhanced Technical Indicator Integration and Classification Models. CoRR abs/2410.06935 (2024). https://doi.org/10.48550/arXiv.2410.06935
4. Kingma, D.P., Ba, J.: Adam: A Method for Stochastic Optimization. In: 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA (2015). https://arxiv.org/abs/1412.6980
5. Li, Y., Dai, W.: Bitcoin Price Forecasting Method Based on CNN-LSTM Hybrid Neural Network Model. J. Eng. 2020(13), 344-347 (2020). https://doi.org/10.1049/joe.2019.1203
6. Livieris, I.E., Kiriakidou, N., Stavroyiannis, S., Pintelas, P.: An Advanced CNN-LSTM Model for Cryptocurrency Forecasting. Electronics 10(3), 287 (2021). https://doi.org/10.3390/electronics10030287
7. Marthinsen, J.E., Gordon, S.R.: The Price and Cost of Bitcoin. Q. Rev. Econ. Finance 85, 280-288 (2022). https://doi.org/10.1016/j.qref.2022.04.003
8. Modi, P.D., Arshi, K., Kunz, P.J., Zoubir, A.M.: A Data-Driven Deep Learning Approach for Bitcoin Price Forecasting. CoRR abs/2311.06280 (2023). https://doi.org/10.48550/arXiv.2311.06280
9. Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system (2008). https://bitcoin.org/bitcoin.pdf. Accessed 1 Dec 2025
10. Omole, O., Enke, D.: Deep Learning for Bitcoin Price Direction Prediction: Models and Trading Strategies Empirically Compared. Financ. Innov. 10(1), 117 (2024). https://doi.org/10.1186/s40854-024-00643-1
11. Otoritas Jasa Keuangan: Rapat Dewan Komisioner Bulanan OJK November 2025 (2025). https://www.ojk.go.id. Accessed 1 Dec 2025
12. Peng, P., Chen, Y., Lin, W., Wang, J.Z.: Attention-Based CNN-LSTM for High-Frequency Multiple Cryptocurrency Trend Prediction. Expert Syst. Appl. 237(B), 121520 (2024). https://doi.org/10.1016/j.eswa.2023.121520
13. Qiang, Z., Shen, J.: Bitcoin High-Frequency Trend Prediction with Convolutional and Recurrent Neural Networks. Stanford CS230 Course Project Report, Stanford University (2021).

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

