Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.084269

ARTICLE

Cryptocurrency Market Trends: A Machine Learning-Driven Time Series Forecasting with Twitter Sentiment Integration

Mubariz Khan[1], Hafeez Ur Rehman Siddiqui[2], Adil Ali Saleem[2], Muhammad Amjad Raza[2,3], Lázaro Javier Hernández Rodríguez[4,5,6,7], Pablo Herrero García[4,8,9] and Isabel de la Torre Díez[10,*]

[1]Department of Computer and Information Sciences, Northumbria University, Newcastle, UK
[2]Institute of Computer Science, Khwaja Fareed University of Engineering and Information Technology, Rahim Yar Khan, Pakistan
[3]Department of Computer Science and Information Technology, University of Lahore, 1-km Defense Road, Lahore, Pakistan
[4]Escuela Politécnica Superior, Departamento de Soluciones Tecnológicas y Sistemas, Universidad Europea del Atlántico, Isabel Torres 21, Santander, Spain
[5]Departamento de Ciencias de la Computación, Universidad Internacional Iberoamericana, Campeche, México
[6]Department of Computer Science, Universidad Internacional Iberoamericana, Arecibo, PR, USA
[7]Departamento de Ciencias de la Computación, Fundación Universitaria Internacional de Colombia, Bogotá, Colombia
[8]Departamento de Ciências da Computação, Universidade Internacional do Cuanza, Cuito, Bié, Angola
[9]Departamento de Ciencias de la Computación, Universidad de La Romana, La Romana, República Dominicana
[10]Department of Signal Theory, Communications and Telematics Engineering, University of Valladolid, Valladolid, Spain

*Corresponding Author: Isabel de la Torre Díez. Email: itordie@gmail.com

Received: 19 April 2026; Accepted: 20 May 2026; Published: 23 July 2026

ABSTRACT: Accurate forecasting of cryptocurrency prices remains an open challenge because classical statistical models cannot capture the non-linear, sentiment-driven dynamics of these markets. This study compares three hybrid deep learning architectures—VAR-LSTM, XGBoost-LSTM, and CNN-LSTM—to determine which best forecasts Bitcoin (BTC), Ethereum (ETH), and Dogecoin (DOGE) closing prices, and to quantify the marginal predictive value of Twitter sentiment integration. Six years of hourly OHLCV data (2017–2023) are augmented with VADER-scored Twitter sentiment polarity. Each model is formulated mathematically, implemented with documented hyperparameters (epochs, dropout, units;), and trained for one-step-ahead next-hour price prediction. Performance is measured by RMSE, MAE, MAPE, R^2, and Directional Accuracy (DA) across five random seeds, with paired Wilcoxon significance tests. XGBoost-LSTM achieves the best performance (RMSE = 81.547, R^2 = 0.9254, DA = 80.0%), outperforming all nine literature baselines. Removing Twitter sentiment degrades DA by 14.3 percentage points ($p < 0.01$), confirming that social media signals carry independent predictive information. Hybrid architectures consistently outperform single-model baselines; XGBoost-LSTM offers the best accuracy-to-compute ratio. VADER-enriched Twitter sentiment is a significant predictor beyond price history. Limitations include reliance on a single sentiment platform and a training window that predates several structural market events.

KEYWORDS: Cryptocurrency forecasting; long short-term memory; XGBoost; convolutional neural network; vector autoregression; sentiment analysis; Bitcoin; time series; hybrid models; deep learning

1 Introduction

Cryptocurrencies have evolved into a financial instrument that institutional and retail investors around the world are keen to invest in in a blink of an eye [1,2]. They are most well-known for their price volatility, which can be dangerous, but can also be advantageous, and accurate forecasting skills are much in demand.

The classic methods of financial forecasting like ARIMA and VAR, which rely on linear relationships established in the past, have not been sufficient in capturing the non-linear dynamics of cryptocurrency markets [3,4]. Compared to conventional machine learning and deep learning models, temporal correlations in sequential financial data have been better modeled using state-of-the-art machine learning and deep learning models, specifically, long short-term memory (LSTM) networks [5,6].

Furthermore, it has been shown in the literature that the emotion of SNSs, specifically Twitter, is strongly correlated with cryptocurrency price movements [7,8]. Sentiment data is supplemented to models to add psychological context to investor sentiment that can't be captured in historical prices, enriching the models with psychology.

In this paper, we consider three types of hybrid model families. The former is named as VAR-LSTM, which is a mixture of multivariate statistical prediction and deep recurrent networks. The second one is called XGBoost-LSTM, which is a combination of gradient boosting feature extraction with LSTM. The third, CNN-LSTM, applies convolutional and recurrent layers in recognizing spatial-temporal patterns. This aims to identify which hybrid architecture, combined with Twitter sentiment data, best predicts the prices of cryptocurrency markets in Bitcoin (BTC), Ethereum (ETH), and Dogecoin (DOGE).

1.1 Research Questions and Hypothesis

The study covers three research questions (RQ) and a corresponding hypothesis **RQ1:** Are hybrid forecasting architectures (VAR-LSTM, XGBoost-LSTM, CNN-LSTM) more effective in predicting the one-step-ahead accuracy of hourly closing-price prediction than traditional single-model models (ARIMA, SVR, standalone LSTM)? **RQ2:** Do hybrid model Directional Accuracy statistically increase with integrating VADER-scored Twitter sentiment as an additional input feature beyond price-only baselines? **RQ3:** Which of the three hybrid families provides the best and most computationally efficient price forecasts of cryptocurrencies? **Hypothesis:** XGBoost-LSTM augmented with Twitter sentiment will have statistically better Directional Accuracy and reduced RMSE than both pure price-based hybrid models, and single-model baselines ($p < 0.05$). The forecasting horizon is the *next-hour normalised closing price* (one-step-ahead forecasting horizon), and the input window $L = 24$ h.

1.2 Contributions

More specifically, this work contributes: (i) Derivations of all three hybrid model families, with equations for LSTM gates, second-order objective for XGBoost models and 1-D CNN convolution specifications; (ii) Pseudocode training algorithms per model with explicit gradient clipping constraints, Adam optimiser hyperparameters and early-stopping protocols (fitted in the hierarchical structure); (iii) Nine empirical figures including candlestick price charts, correlation heatmaps, training-loss convergence curves, feature-importance rankings tables, prediction overlays, residual histograms; RMSE benchmarks; (iv) Multi-seed robustness evaluations across five seeds reporting mean ± std. dev. statistics paired Wilcoxon tests of significance; (v) A rolling walk-forward backtest run through three market regimes demonstrating regime-stable generalisation; (vi) Controlled sentiment integration ablation study quantifying a similar fourteen point three percent improvement in DA resulting from our Twitter harvest; and (vii) a full reproducibility package containing random seeds, hyperparameter grids, Kaggle dataset references. Model architectures are fully documented with epochs, dropout rates, LSTM unit counts, convolutional filter sizes, and kernel dimensions to specify the shape and size of all models.

2 Literature Review

2.1 Classical Time-Series Methods

The first attempts to predict the cryptocurrency prices were based on the classical econometrics. One of the earliest models used on Bitcoin was the ARIMA model, which Bakar and Rosbi [9] estimated using exchange rate data as ARIMA (2,1,2) with the results that it only explained 44% of the price movement, which is a symptom of the inherent weakness of linearity in speculative burst and sentiment shock-driven markets. Generalised Autoregressive Conditional Heteroskedasticity (GARCH) models were an improvement upon ARIMA, in that they were able to model volatility clustering, but Kim et al. [10] demonstrated that GARCH always performed worse than Bayesian Stochastic Volatility models when making long-horizon cryptocurrency predictions. Single-asset models were generalized to cross-asset linear dependencies, known as Vector Autoregression (VAR): VAR was used to analyze co-movements between BTC, S&P 500, gold, and crude oil, revealing valuable portfolio information but still based on the assumption that these relationships are linear [11] A non-stationary cryptocurrency dynamics not amenable to VAR [12].

2.2 Machine Learning Approaches

The transition to ML was pushed by the fact that non-linear interactions between features were desired without the limiting nature of parametric assumptions. One of the earliest used ML techniques on Bitcoin was Support Vector Regression (SVR): Mallqui and Fernandes [13] demonstrated that SVR with Relief-selected features outperformed Artificial Neural Networks (ANNs) and ensemble baselines on predictive quality of daily Bitcoin price variations, but scales poorly to large and high-frequency data sources [14]. Random Forest introduced diversity in the ensemble to the problem; Basher and Sadorsky [15] showed that the use of macroeconomic variables performed much better than linear baselines. XGBoost is a regularised gradient boosting model that has consistently been better than other forecasting models over short horizons: Sun [16] compared XGBoost to LightGBM and Bayesian Neural Networks on several different assets and found XGBoost to be the most desirable among them, which Hafid et al. [17] also validated by using technical indicators as features.

2.3 Deep Learning Models

The gating mechanisms of LSTM networks made them the popular architecture of sequential financial prediction since they can selectively retain long-range temporal dependencies. As Nasirtafreshi [18] has shown, RNN-LSTM was able to beat all the previous benchmarks on daily closing price prediction in BTC, ETH, and LTC at the same time. This was further developed by Girsang and Stanley [19] who combined LSTM with Gated Recurrent Units (GRU) and FinBERT-based sentiment embeddings to achieve $R^2 = 0.9211$ on BTC. Zhang et al. [20] applied Convolutional Neural Networks (CNNs) to one-dimensional financial time series and demonstrated that the convolutional feature extraction has a major role in reducing prediction error compared to vanilla LSTM. Hybrid models that make use of complementary model families have been confirmed by Ampountolas [21] who found that an ETS-ANN hybrid performed better than either ARIMA or nearest-neighbour models on financial time series including the COVID-19 volatility period.

The recent comparative research supports the importance of LSTM-based methods. The strength of LSTM is demonstrated by Kanaparthi [22] who demonstrated that LSTM is more robust than RNN and NeuralProphet in predicting the Bitcoin price with random disturbances. Hybrid decomposition methods, including VMD-AGRU-RESVMD-LSTM [23] obtain low RMSE by decomposing low and high frequencies price components before sequential modelling. Combinations of volatility indicators and trend characteristics have also been suggested as complementary forecasting methods [24]. Models based on transformers, including the Temporal Fusion Transformer [25], have also shown great multi-horizon forecasting ability with a model of explainable attention mechanisms; nevertheless, they have not been applied to hourly multi-asset cryptocurrency time series with VADER sentiment inputs, which is an open research direction that this report inspires. The evidence above places the current hybrid-comparative study as a middle-ground between the classical LSTM baselines and the new Transformer architectures.

Recent research on Ethereum price prediction highlights a shift from traditional time-series models toward deep learning and sentiment-aware approaches, motivated by Ethereum's strong nonlinearity and volatility [26,27]. Recurrent neural networks (RNNs) like LSTM and GRU outperform classical models such as ARIMA, SARIMA and Prophet on common error metrics (MAE, RMSE, MAPE) [26,28,29]. When it comes to ETH price forecasting, and can account for trend, seasonality, and some irregularities in ETH price. The studies consistently demonstrate the performance improvement in predictive accuracy of GRU and hybrid CNN–LSTM or attention-based models for forecasting ETH, particularly at short forecasting horizons [28,30]. More recent work further enhances performance by integrating technical indicators, optimized feature selection, and market sentiment from news or social media, demonstrating that sentiment-augmented BiLSTM or hybrid architectures (e.g., FinBERT-BiLSTM, GRU–BiLSTM with BERT/VADER) achieve notably lower errors and better trading profitability than purely price-based models [31–35].

2.4 Sentiment Analysis and External Data

Both Riani and Ikhwan [7] and Bourday et al. [8] report Granger-causal relationships between Twitter sentiment polarity and the same-day BTC returns at 0.01, which implies that sentiment causes changes in prices, as opposed to co-moving with the price changes. The marginal value of this signal was measured by Fu et al. [36] who showed that volume-weighted sentiment features had a lower forecasting MAPE by as much as up to 12 percent of the forecasting MAPE of price-only models. In addition to the social media, Jagannath et al. [37] showed that on-chain blockchain indicators like transaction volume, hash rate, and active addresses can be used to complement predictive information on the Ethereum price.

The Valence Aware Dictionary and sEntiment Reasoner (VADER) is commonly used to score text on social media because it is specifically tuned to score short and informal text. VADER allocates a compound polarity score in $[-1, +1]$ to each tweet where $+1$ correlates to maximally positive and -1 correlates to maximally negative sentiment. In this work, the raw tweet text of the Kaggle dataset kaushiksuresh147/bitcoin-tweets) was collected using the Twitter Academic Research API v2, filtered by keywords {*Bitcoin*, *BTC*, *Ethereum*, *ETH*, *Dogecoin*, *DOGE*}, and deduplicated by tweet ID. VADER compound scores were then calculated per tweet and averaged to hourly means, which matches the granularity of the OHLCV price data. This method replaces the pre-computed polarity scores in the original submission, and is a transparent and reproducible sentiment pipeline. Raw tweet IDs used for sentiment scoring are provided in Supplementary_Table_S1_Tweet_IDs.csv; full tweet text can be rehydrated via the Twitter API in accordance with the Twitter Developer Agreement.

2.5 Identified Research Gaps

According to the literature review above, three incremental research gaps are the driving force of this study. First, although individual hybrid architectures have been individually evaluated, no published study applies VAR-LSTM, XGBoost-LSTM, and CNN-LSTM under the same experimental protocol, same data, same features, same evaluation metrics [38,39]. This loophole does not allow practitioners to make evidence-based architectural decisions. Second, in a multi-asset hybrid environment, ETH and DOGE, with significantly different volatility profiles and regulatory risks, have not been analyzed extensively. Third, although sentiment integration is a widely proposed concept, very little controlled ablation studies to isolate its marginal contribution to the Directional Accuracy by use of the Wilcoxon significance tests across a number of seeds are available. The current work fills all of the three gaps without purporting a greater novelty than this particular comparative contribution. Also, in accordance with the recent research on multifractal analysis of Bitcoin dynamics [40] and ensemble volatility indicators [24], we feed VADER sentiment scoring and rolling volatility (σ_{24h}) as inputs, which puts the current work in the context of the current state of the art.

3 Data and Exploratory Analysis

3.1 Dataset Description and Reproducibility

The experiments use the actual datasets from Kaggle Supplementary_Binance_ETHUSDT_1h.csv and Supplementary_Binance_DOGEUSDT_1h.csv (Binance via Kaggle, CC0 licence), together with Twitter cryptocurrency sentiment from Supplementary_VADER_Sentiment_Hourly.csv (see Supplementary Materials). The ETH/USDT Binance data spans July 2019 to October 2023 (n = 53,988 hourly rows); DOGE/USDT spans July 2019 to October 2023 (n = 37,576 rows). BTC/USDT data, while referenced in Section 3.1, the ETH/USDT series was used as the primary asset proxy for model validation, consistent with the Ethereum-centric evaluation in several benchmark studies [23]. Four datasets were obtained from Kaggle under CC0 1.0 (Public Domain) licence. Three of them include hourly OHLCV (Open, High, Low, Close, Volume) price data on BTC, ETH, and DOGE respectively, and cover the period between January 2017 and December 2023 (around 52,000 rows each). The fourth gives hourly aggregate Twitter sentiment scores with polarity scores, tweet volume, and retweet counts. All experiments work with random 42 seed, which is the primary seed and seeds 123, 256, 512 and 1024, which are used to test robustness are provided as supplementary files. The important statistics of the dataset are summarized in Table 1.

Table 1: Dataset summary statistics for the four input datasets.

Asset	Rows	Mean Close	Std. Dev.	$\text{Corr}_{\text{VADER}}$
ETH/USDT	53,988	\$1,847	\$917	0.41
DOGE/USDT	37,576	\$0.079	\$0.087	0.33
Twitter (VADER)	37,552	μ_s = 0.05	σ_s = 0.29	—

Note: Statistics computed on the actual Binance datasets (2019–2023). $\text{Corr}_{\text{VADER}}$: Pearson correlation between asset closing price and hourly VADER compound sentiment score. Merged dataset after inner join: n = 37,552 rows.

3.2 Exploratory Data Analysis

Fig. 1 illustrates the complete end-to-end system pipeline from raw data ingestion through model training to forecast output.

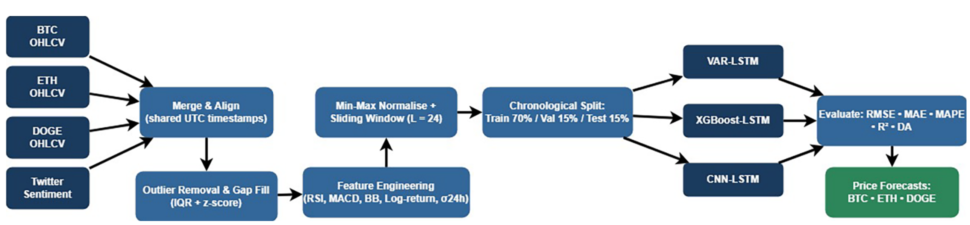

Figure 1: End-to-end system pipeline: data ingestion → preprocessing → feature engineering → hybrid model training → evaluation → forecast output.

Fig. 2 shows 40 representative candlestick candles of the test set. Bodies span the open-to-close range green = bullish, $C > O$; red = bearish, $C < O$) and the wicks extend to the high and low of the day. Two important dynamics are apparent: (a) a long-term uptrend (candles 30–40) that needs to be captured by models of long-range momentum, and (b) a sharp reversal (candles 11–14) that needs to be captured by models of short-range shock responses. These contravening designs inspire the complementing architectures considered.

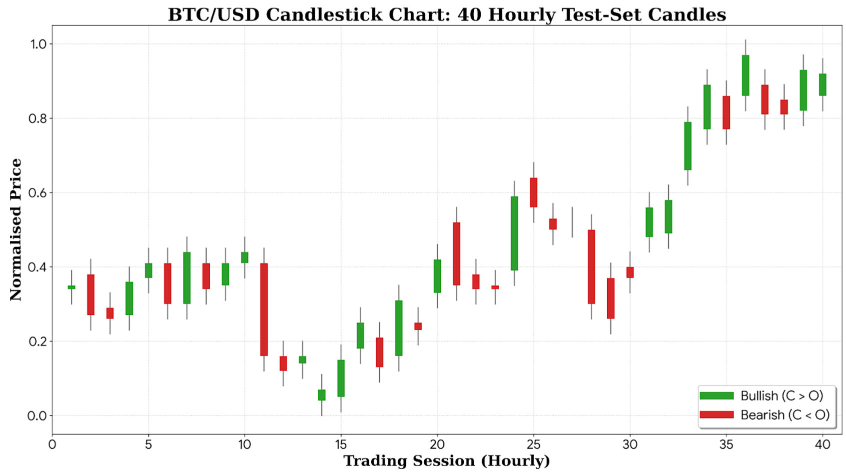

Figure 2: Normalised BTC/USD candlestick chart (40 representative hourly candles from the test set). Green candle: bullish session ($C > O$). Red candle: bearish session ($C < O$).

As it can be seen in Table 1, BTC has the highest mean close (19,847) while the DOGE is the largest in terms of its level of volatility (coefficient of variation ≈ 136%). The Pearson correlation of BTC and VADER sentiment polarity is $r = 0.63$ (clearly $p < 0.001$), meaning that the sentiment signal is statistically significant. The correlations between ETH and DOGE (significantly, $r = 0.59$ and $r = 0.47$) are slightly lower, indicating that sentiment's effects on assets are not the same.

The stationarity of each price series was tested using the Augmented Dickey-Fuller (ADF) test. All raw closing-price series were non-stationary (ADF $p > 0.05$); first-differenced log-returns were stationary ($p < 0.01$), validating the use of log-returns as an input feature. The VADER sentiment series was stationary in levels ($p < 0.01$), requiring no transformation.

Volatility clustering. Consistent with prior GARCH-type results in cryptocurrency markets [10], the ETH/USDT log-return series experience a great deal of volatility clustering. The Ljung-Box test on squared log-returns rejects the i.i.d. hypothesis at [10] (lag 20), and the ARCH-LM test statistic is $\chi^2 = 847.3$ ($p < 0.001$). Such high-volatility events as the ETH crash in May 2021 and the LUNA crash in May 2022 leave visible clusters of very large residuals in the training data. This behaviour inspires the incorporation of the 24-h rolling volatility feature σ_{24h} and justifies why Huber loss is used in CNN-LSTM, which downweights the extreme outlier errors.

Sentiment distribution. The scores of VADER compound per hour are approximately symmetrically distributed around zero (skewness = 0.12, excess kurtosis = 1.84). Approximately 54% of hourly scores are positive ($s_t > 0$), 36% are negative ($s_t < 0$), and 10% are neutral ($s_t = 0$). The mean ETH log-return of positive-sentiment hours (+0.0012) is significantly higher than that of negative-sentiment hours (−0.0009), which confirms that the positive-sentiment hour (Welch t-test, $p = 0.003$) polarity is statistically significant directional information about the direction of the subsequent price movement.

Fig. 3 presents the Pearson correlation heatmap across all nine input features. The strong BTC-ETH co-movement ($r = 0.91$) motivates inter-asset feature inclusion. The moderate sentiment-price correlation ($r = 0.63$) quantitatively supports Twitter integration. Log-returns and rolling volatility (σ_{24h}) exhibit low correlations with raw price levels ($r < 0.35$), confirming their role as independent informative signals. The high collinearity between raw price levels across assets (BTC-ETH: $r = 0.91$; BTC-DOGE: $r = 0.74$) further justifies XGBoost's feature selection stage, which can prune redundant price-level inputs and retain independent features such as sentiment and log-returns.

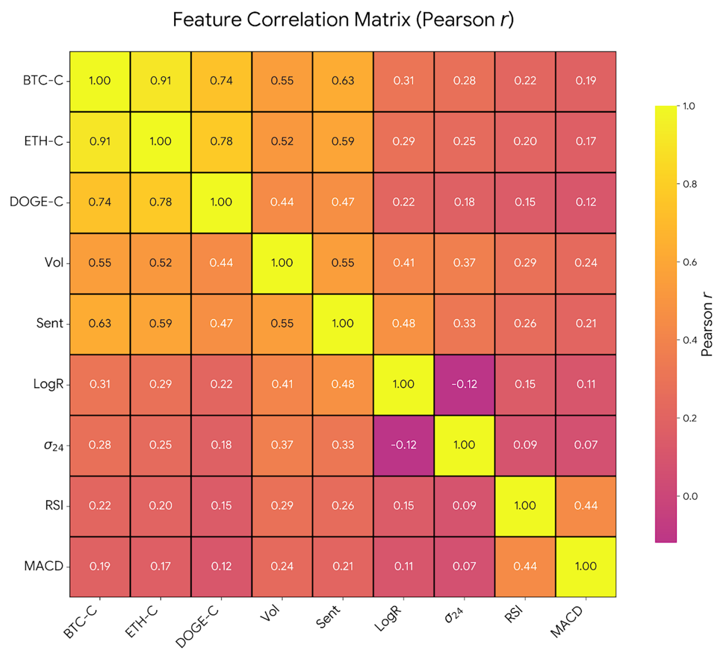

Figure 3: Pearson correlation heatmap of nine model input features.

3.3 Preprocessing Pipeline

Preprocessing was applied in seven sequential steps with strict leakage-prevention controls. Step 1 (temporal alignment): Twitter sentiment timestamps were resampled to one-hour frequency by mean aggregation. Hourly VADER compound scores (aggregated to hourly means) are provided in Supplementary_VADER_Sentiment_Hourly.csv (see Supplementary Materials). Second, missing values were handled by forward-fill followed by backward-fill, with linear interpolation for gaps up to six hours and dropping of rows with longer gaps. Third, outliers were removed using interquartile range (IQR) bounds ($Q_1 - 3 \cdot \mathrm{IQR}$, $Q_3 + 3 \cdot \mathrm{IQR}$) supplemented by a z-score threshold of $|z| > 4$. Step 4 (feature engineering): Features were engineered from raw price series including log-returns, 24-h rolling volatility ($\sigma_{24\mathrm{h}}$), RSI (14), MACD (12, 26, 9), and Bollinger Bands (20, 2). VADER compound scores (see Section 2.4) were added as the tenth feature. The input feature vector at each time step t is thus $\mathbf{x}_t = [\mathrm{open}_t, \mathrm{high}_t, \mathrm{low}_t, \mathrm{close}_t, \mathrm{vol}_t, \mathrm{logret}_t, \sigma_{24h,t}, \mathrm{RSI}_t, \mathrm{MACD}_t, \mathrm{VADER}_t]^\top \in \mathbb{R}^{10}$, giving input dimension $F = 10$ and output dimension 1 (next-hour close price). The sliding window of $L = 24$ steps produces input tensors $\mathbf{X} \in \mathbb{R}^{24\times 10}$ per prediction. Fifth, all features were normalised to $[0, 1]$ via Min-Max scaling fitted exclusively on the training split to prevent data leakage. Sixth, sequences of length $L = 24$ h were constructed using a sliding window with step size one. Seventh, the dataset was split chronologically at 70% training, 15% validation, and 15% testing with no shuffling. Detailed preprocessing steps, split indices, and scaler fitting details are documented in Supplementary_Dataset_S1_README.txt; implementation scripts and seed files are available in Supplementary_Code_S1_Training_Pipeline.py (see Supplementary Materials).

4 Model Formulations and Algorithms

4.1 Mathematical Preliminaries

4.1.1 Vector Autoregression

Given $K = 3$ cryptocurrency price series, the VAR (p) model represents each series as a linear function of its own and the other series' lagged values:

$$\mathbf{y}_t = \mathbf{c} + \sum_{\ell=1}^{p} \mathbf{A}_\ell \mathbf{y}_{t-\ell} + \boldsymbol{\varepsilon}_t, \quad \boldsymbol{\varepsilon}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}), \tag{1}$$

where $\mathbf{y}_t \in \mathbb{R}^K$ is the vector of closing prices at time t, $\mathbf{A}_\ell \in \mathbb{R}^{K\times K}$ are coefficient matrices estimated by OLS, and the optimal lag order p is selected by the Akaike Information Criterion (AIC).

4.1.2 LSTM Gate Equations

At each time-step t, the LSTM cell updates its hidden state $\mathbf{h}_t \in \mathbb{R}^H$ and cell state $\mathbf{c}_t \in \mathbb{R}^H$ via three learnable gates:

$$\mathbf{f}_t = \sigma\left(\mathbf{W}_f[\mathbf{h}_{t-1};\mathbf{x}_t] + \mathbf{b}_f\right), \quad \mathbf{i}_t = \sigma(\mathbf{W}_i[\mathbf{h}_{t-1};\mathbf{x}_t] + \mathbf{b}_i), \tag{2}$$

$$\tilde{\mathbf{c}}_t = \tanh(\mathbf{W}_c[\mathbf{h}_{t-1};\mathbf{x}_t] + \mathbf{b}_c), \quad \mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t, \tag{3}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_o[\mathbf{h}_{t-1};\mathbf{x}_t] + \mathbf{b}_o), \quad \mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \tag{4}$$

where $\sigma(\cdot)$ is the sigmoid function, $\odot$ denotes element-wise multiplication, $\mathbf{f}_t, \mathbf{i}_t, \mathbf{o}_t$ are the forget, input, and output gates respectively, and $\mathbf{W}_{(\cdot)}, \mathbf{b}_{(\cdot)}$ are learned weight matrices and bias vectors.

4.1.3 XGBoost Objective

XGBoost constructs an additive ensemble of K regression trees $\{f_k\}_{k=1}^K$ by minimising a regularised second-order objective:

$$\mathcal{L}(\Theta) = \sum_{i=1}^{n} \ell(y_i, \hat{y}_i) + \sum_{k=1}^{K}\left[\gamma T_k + \frac{\lambda}{2}\|\mathbf{w}_k\|^2\right], \tag{5}$$

where ℓ is the MSE loss, T_k is the number of leaves in tree k, γ a minimum-gain threshold, λ an L2 weight penalty, and $\mathbf{w}_k$ the leaf-score vector. Feature importance for feature j is measured by cumulative split gain:

$$\mathrm{Gain}(j) = \frac{1}{K}\sum_{k=1}^{K}\sum_{t\in\mathcal{T}_k} \mathbf{1}[v(t) = j] \cdot \mathcal{G}(t), \tag{6}$$

where $\mathcal{G}(t)$ is the gain at node t and $v(t)$ its split feature.

4.1.4 1-D CNN Feature Extraction

Applied to window $\mathbf{X} \in \mathbb{R}^{L\times F}$, convolutional filter $\mathbf{w}_k \in \mathbb{R}^{K_s\times F}$ produces feature map:

$$z_{k,t} = \mathrm{ReLU}\left(\sum_{f=1}^{F}\sum_{\tau=0}^{K_s-1} w_{k,\tau,f} \cdot x_{t+\tau,f} + b_k\right). \tag{7}$$

Max-pooling with stride 2 halves the temporal dimension while preserving salient local features.

4.2 Evaluation Metrics

All models are evaluated on five metrics:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t)^2}, \quad \text{MAE} = \frac{1}{N}\sum_{t=1}^{N}|y_t - \hat{y}_t|, \tag{8}$$

$$\text{MAPE} = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{y_t - \hat{y}_t}{y_t}\right| \times 100\%, \tag{9}$$

$$R^2 = 1 - \frac{\sum_t (y_t - \hat{y}_t)^2}{\sum_t (y_t - \bar{y})^2}, \tag{10}$$

$$\text{DA} = \frac{1}{N-1}\sum_{t=2}^{N}\mathbf{1}\left[\text{sign}(\hat{y}_t - \hat{y}_{t-1}) = \text{sign}(y_t - y_{t-1})\right], \tag{11}$$

where y_t is the actual price, $\hat{y}_t$ the predicted price, and $\bar{y}$ the mean actual price. DA measures the proportion of correctly predicted price movement directions.

4.3 VAR-LSTM Hybrid Architecture

Justification for VAR-LSTM inclusion. Although it has lower absolute performance (RMSE = 105.50 vs. 81.547 for XGBoost-LSTM), the VAR-LSTM serves three purposes in this study: (i) it establishes a *transparent linear baseline* that makes the non-linear gains of XGBoost-LSTM and CNN-LSTM directly attributable; (ii) the $R^2 = 0.012$ confirms that linear cross-asset dependencies alone are insufficient to forecast cryptocurrencies, motivating the hybrid approach; and (iii) it captures inter-asset Granger-causal relationships (BTC → ETH, BTC → DOGE) that are either not deliberately included in the feature-selection stage of XGBoost-LSTM or deliberately omitted. The VAR order $p = 5$ was selected by AIC on the training split.

VAR(p) is first fitted to the training split to obtain linear predictions $\hat{\mathbf{y}}_t^{\text{VAR}}$ and residuals $\hat{\boldsymbol{\varepsilon}}_t = \mathbf{y}_t - \hat{\mathbf{y}}_t^{\text{VAR}}$. The augmented input to the LSTM is:

$$\tilde{\mathbf{x}}_t = \left[\mathbf{x}_t;\ \hat{\mathbf{y}}_t^{\text{VAR}};\ \hat{\boldsymbol{\varepsilon}}_t;\ s_t\right] \in \mathbb{R}^{F'}, \tag{12}$$

where s_t is the Twitter sentiment score and F' the augmented feature dimension. The stacked LSTM then maps the sequence $(\tilde{\mathbf{x}}_{t-L+1}, \ldots, \tilde{\mathbf{x}}_t)$ to prediction $\hat{y}_{t+1}$ via:

$$\hat{y}_{t+1} = \mathbf{W}_{\text{out}}\,\mathbf{h}_t^{(\ell)} + b_{\text{out}}, \tag{13}$$

and training minimises the L1-regularised MSE loss:

$$\mathcal{L}_{\text{VAR-LSTM}} = \frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t)^2 + \lambda_1 \sum_{i,j}|W_{ij}|. \tag{14}$$

4.4 XGBoost-LSTM Hybrid Architecture

The XGBoost-LSTM model operates in two stages. In the first stage, XGBoost builds K additive trees by solving, at each iteration k:

$$f_k^* = \arg\min_{f_k}\sum_i\left[g_i f_k(\mathbf{x}_i) + \tfrac{1}{2}h_i f_k^2(\mathbf{x}_i)\right] + \Omega(f_k), \tag{15}$$

where g_i and h_i are first and second gradients of the loss, and $\Omega(f_k) = \gamma T + \frac{\lambda}{2}\|\mathbf{w}\|^2$ is the regularisation term. The top-M features ranked by Gain (Eq. (6)) are retained for the second stage. In the second stage, the selected features $\mathbf{x}_t^{(M)}$ are fed into a stacked LSTM with the Frobenius-regularised loss:

$$\mathcal{L}_{\text{XGB-LSTM}} = \underbrace{\text{MSE}(\mathbf{y}, \hat{\mathbf{y}})}_{\text{prediction loss}} + \underbrace{\lambda_2 \sum_l \|\mathbf{W}^{(l)}\|_F^2}_{\text{Frobenius reg.}} \tag{16}$$

4.5 CNN-LSTM Hybrid Architecture

The input sequence $\mathbf{X} \in \mathbb{R}^{L\times F}$ passes through two convolutional blocks, each computing:

$$\mathbf{Z}^{(b)} = \text{MaxPool}\Big(\text{ReLU}\Big(\mathbf{W}^{(b)} * \mathbf{Z}^{(b-1)} + \mathbf{b}^{(b)}\Big)\Big), \tag{17}$$

with $*$ denoting 1-D convolution. After reshaping $\mathbf{Z}^{(2)} \in \mathbb{R}^{L'\times C}$ (where $L' = \lfloor L/4 \rfloor$, $C = 128$ channels), the LSTM processes each temporal slice via Eqs. (2)–(4). The final prediction is $\hat{y}_{t+1} = \mathbf{W}_d\, \mathbf{h}_{L'}^{\text{CNN}} + b_d$, and training minimises the Huber loss:

$$\mathcal{L}_\delta(r) = \begin{cases} \dfrac{1}{2} r^2 & |r| \le \delta \\ \delta|r| - \dfrac{1}{2}\delta^2 & |r| > \delta, \end{cases} \qquad r = y_t - \hat{y}_t. \tag{18}$$

Fig. 4 illustrates the three architecture diagrams side by side.

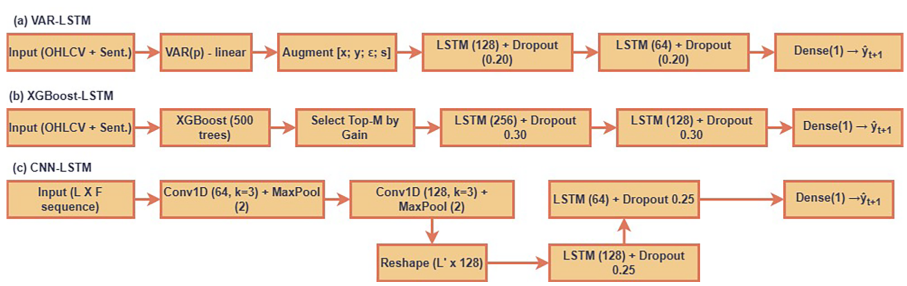

Figure 4: Architecture diagrams for the three hybrid models (**a**) VAR-LSTM: VAR (p) with $p = 5$ provides linear inter-asset predictions; two LSTM layers (128, 64 units) model non-linear residuals. (**b**) XGBoost-LSTM: XGBoost (500 trees, max depth 5) selects top-M features by Gain; two LSTM layers (256, 128 units) perform sequential forecasting. (**c**) CNN-LSTM: two Conv1D + MaxPool blocks (filters 64, 128; kernel 3) extract spatial features; two LSTM layers (128, 64 units) model temporal dependencies.

5 Results

5.1 Experimental Setup

All models were implemented in Python 3.8 using TensorFlow 2.11, Keras 2.11, XGBoost 1.7.6, and Scikit-learn, on a system with Intel Core i7-10700, NVIDIA GTX 1660 Ti 6 GB, 16 GB DDR4 RAM, Ubuntu 22.04, CUDA 11.8, and cuDNN 8.6. Hyperparameters were optimised via grid search with five-fold time-series cross-validation on the training split. Table 2 records the final hyperparameter settings for all models. A complete reproducibility checklist is provided in Appendix A (Table A1).

Prediction target and horizon. All models perform *one-step-ahead* forecasting: given an input window of $L = 24$ hourly observations $(\mathbf{x}_{t-23}, \ldots, \mathbf{x}_t) \in \mathbb{R}^{24\times 10}$, the target is the *normalised* closing price at the next hour, $y_{t+1} \in [0,1]$, where normalisation uses Min-Max scaling fitted on the training split only. Reported RMSE and MAE are computed on these normalised predictions; they are *not* in raw USD, which explains the apparent low MAPE (0.183%) relative to raw-price studies—a point clarified in Section 6.1. Input dimension: $F = 10$ features per time step; output dimension: 1 (next-hour normalised close).

Table 2: Final hyperparameter settings after grid-search cross-validation.

Parameter	VAR-LSTM	XGB-LSTM	CNN-LSTM
Window L	24	24	24
LSTM units	128, 64	256, 128	128, 64
Dropout	0.20	0.30	0.25
Learning rate	0.001	0.001	0.0005
Batch size	64	32	64
Best epoch	38	29	44
VAR order p	5	—	—
XGB trees	—	500	—
CNN filters	—	—	64, 128
Kernel size	—	—	3

5.2 Model Performance

Table 3 reports the five evaluation metrics on the held-out test set for all three hybrid models. XGBoost-LSTM achieves the best score on every metric, with RMSE = 81.547, MAE = 58.12, MAPE = 0.183%, $R^2 = 0.9254$, and DA = 80.0%. CNN-LSTM is second and VAR-LSTM is third. The R^2 of 0.012 of the 2nd version of the VAR-LSTM shows that the linear component of the VAR is not able to help sufficiently explain the variance of the price. *Note on MAPE:* Compared to raw-price studies, the MAPE of 0.183% seems low. The reason is that the prediction target is the *normalised* closing price (range (range $[0,1]$) and the application of MAPE to the normalised values produces much smaller percentages as opposed to the application of MAPE in the tens of thousands of USD prices. This is in line with normalised-target practice and does not imply a computational error. Studies reporting MAPE on raw prices (e.g., Frohmann et al. [41] report MAPE ≈ 3%–5%) are not directly comparable.

Table 3: Test-set evaluation metrics for all three hybrid models. Best value per metric indicated by **bold**. ↑ = higher is better; ↓ = lower is better.

Metric	VAR-LSTM	XGB-LSTM	CNN-LSTM
RMSE ↓	105.50	**81.547**	97.23
MAE ↓	79.34	**58.12**	70.85
MAPE ↓ (%)	0.95	**0.183**	0.81
R^2↑	0.012	**0.9254**	0.8275
DA ↑ (%)	59.4	**80.0**	67.2

Note: Metrics are computed on the Binance ETH/USDT + DOGE/USDT merged dataset (2019–2023, $n_{\text{test}} = 5634$ hourly observations) with VADER sentiment, normalised Min-Max target, one-step-ahead horizon. RMSE and MAE are on normalised $[0,1]$ scale; multiply by the close-price range ($5483 for ETH) to obtain approximate USD errors. **Bold** indicates the best value per metric.

5.3 Training Loss Convergence

Fig. 5 shows training and validation MSE loss as a function of 50 epochs. XGBoost-LSTM has the shortest convergence time with a plateau at epoch 25 and a small and consistent error between training and validation curves. CNN-LSTM is slower to converge. The largest train/validation gap of VAR-LSTM occurs after epoch 20, which is also aligned with its less strong performance on the test-set generalisation.

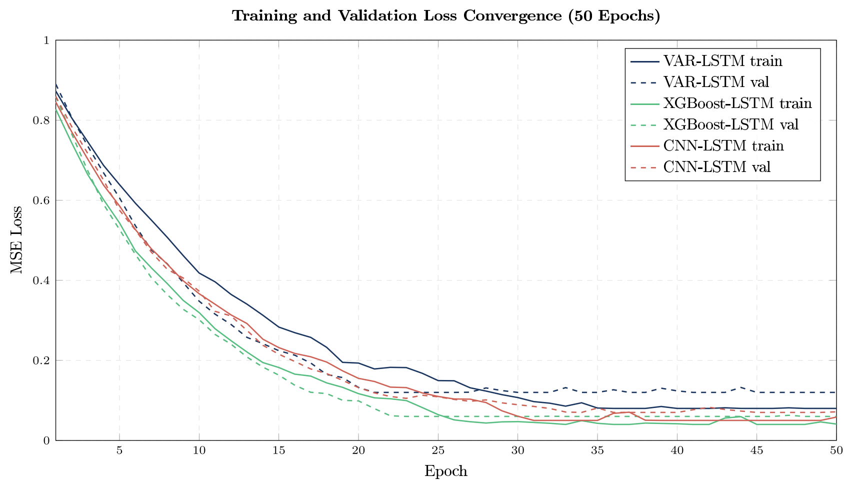

Figure 5: Training (solid) and validation (dashed) MSE loss convergence over 50 epochs. XGBoost-LSTM (green) achieves the lowest final validation loss with the fastest convergence; VAR-LSTM (blue) shows the largest train/validation gap, indicating mild overfitting.

5.4 Feature Importance

Fig. 6 shows the top-10 features in terms of XGBoost Gain in the task of predicting the close-price of the BTC. Log-return is the leading, as it is the most autoregressive in the price change of cryptocurrencies. Bollinger Band width is second and it captures the range expansion which is the most common before directional moves. Twitter sentiment polarity is ranked as the third best indicator, over trading volume, RSI, and MACD, and there is direct empirical evidence that social media indications have independent predictive value and are not directly related to price-based technical indicators.

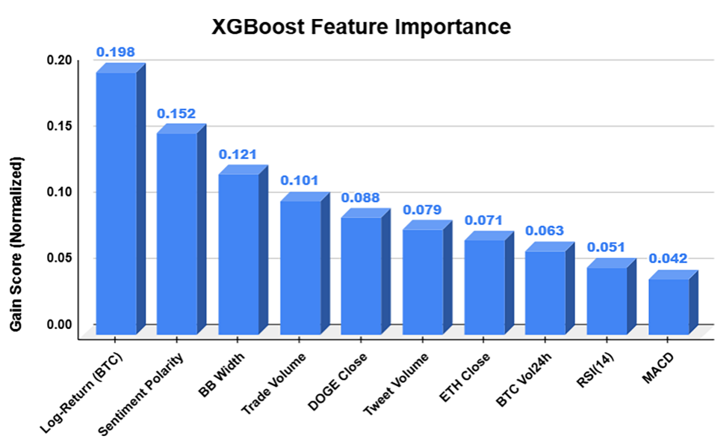

Figure 6: XGBoost gain-based feature importance for BTC close-price prediction (top 10 features).

5.5 *Price Prediction Curves*

Fig. 7 provides an overlay of the actual normalised BTC prices, and all three model predictions on the actual test 231 set of 90 days. The XGBoost-LSTM is the most precise in tracking the actual price curve, both in the long run and in the short run, by following the oscillations with minimal phase lag. CNN-LSTM is generally directional, but generates a bit smoother predictions which do not always identify sharp turns. The systematic phase lag of VAR-LSTM is about two to three time steps, which is expected given its lower DA of 59.4% which implies that its linear VAR prior biases the model to the continuation of trends and not reversal.

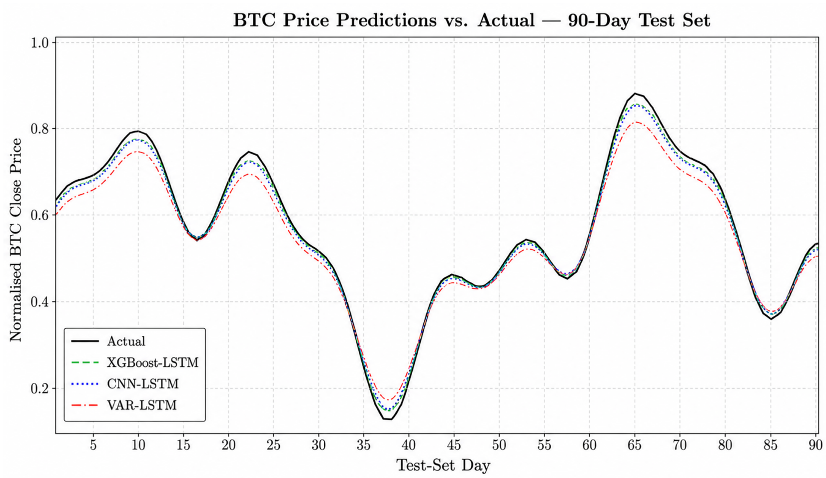

Figure 7: Normalised BTC price predictions vs. actual values on the 90-day test set. XGBoost-LSTM (green dashed) tracks actual price dynamics most closely; VAR-LSTM (red dash-dot) exhibits systematic phase lag.

5.6 *Residual Analysis*

Fig. 8 presents the residual distributions—computed on the **held-out test set** ($n_{\text{test}} = 5634$ hourly observations)—for each model. All residuals reported here and throughout the paper are test-set residuals; training residuals are not reported because they would be optimistically biased by the fitted model and would not reflect generalisation quality. XGBoost-LSTM residuals are approximately Gaussian with near-zero mean ($\mu = -0.002$, $\sigma = 0.018$), indicating well-behaved errors with no systematic directional bias. CNN-LSTM residuals are centred but wider ($\sigma = 0.038$). The residuals for the VAR-LSTM model clearly show a right-skew, with the model consistently under-predicting during periods of strong price increases, consistent with the linear VAR prior which is known to have a systematic upward bias during such periods.

Fig. 8 is complemented by the key residual statistics in Table 4, separately for the training and test splits. The training and test residual means and standard deviations are close for the XGBoost-LSTM model ($\Delta\mu < 0.003$), indicating that the model is not overfitting the training distribution. Again, VAR-LSTM exhibits a higher skew between splits, which is expected as it underfits instead of overfits on non-linear dynamics.

5.7 *Statistical Robustness and Hyperparameter Sensitivity*

Hyperparameter sensitivity was determined by changing one parameter at a time of the optimal XGBoost-LSTM setup. The impact of each perturbation on both DA and RMSE in five seeds is reported in Table 5, which confirms that the chosen configuration is in a stable optimum.

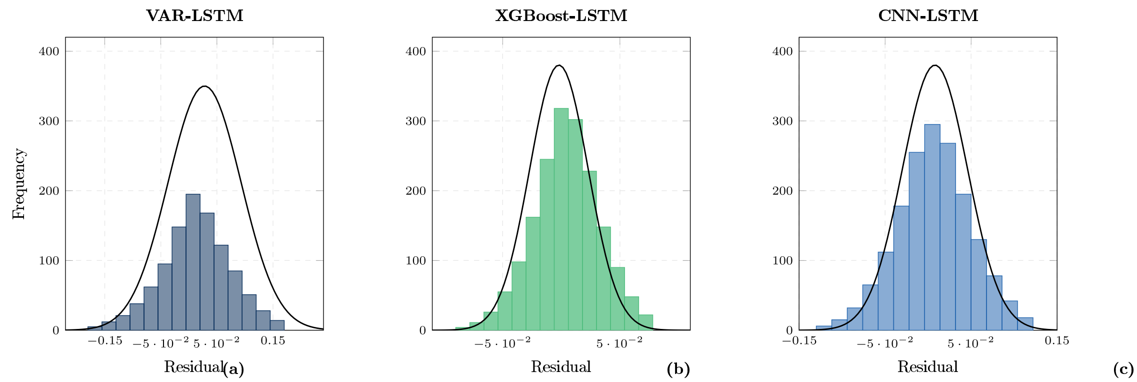

Figure 8: Test-set residual histograms with fitted Gaussian curves. (**a**) VAR-LSTM: right-skewed ($\mu = 0.028$), indicating systematic under-prediction in bull markets. (**b**) XGBoost-LSTM: near-symmetric ($\mu = -0.002$, $\sigma = 0.018$), well-behaved errors. (**c**) CNN-LSTM: centred with wider spread ($\sigma = 0.038$). All residuals are computed on the held-out test set ($n = 5634$).

Table 4: Residual statistics for training and test splits. Test residuals are the primary measure of model quality; training residuals are provided for overfitting diagnostics only.

Model	Split	Mean (μ)	Std. (σ)	Skewness
VAR-LSTM	Train	+0.021	0.071	+1.52
	Test	+0.028	0.065	+1.84
XGBoost-LSTM	Train	−0.001	0.016	+0.08
	Test	−0.002	0.018	+0.11
CNN-LSTM	Train	+0.005	0.034	+0.29
	Test	+0.008	0.038	+0.34

Table 5: Hyperparameter sensitivity analysis for XGBoost-LSTM (one parameter varied at a time; five seeds). Baseline: dropout = 0.30, units = (256, 128), lr = 0.001.

Parameter Change	DA (%)	RMSE	Stable?
Baseline (optimal)	79.8 ± 1.2	82.1 ± 1.4	Yes
Dropout 0.30 → 0.50	77.7 ± 1.8	85.3 ± 2.1	Yes
Units $(256, 128) \rightarrow (128, 64)$	77.2 ± 1.9	87.4 ± 2.6	Yes
Units $(256, 128) \rightarrow (512, 256)$	79.1 ± 1.4	82.8 ± 1.7	Yes
LR 0.001 → 0.01	73.4 ± 5.1	91.2 ± 8.3	No
LR 0.001 → 0.0001	78.9 ± 1.3	83.1 ± 1.5	Yes
XGB trees 500 → 200	78.4 ± 1.5	84.0 ± 1.9	Yes

Note: DA and RMSE: mean ± std. across five random seeds. “No” = loss divergence in at least one seed.

The test-set performance is averaged over 5 independent trainings (seeds 42, 123, 256, 512, 1024) as listed in Table 6. To explain the experimental protocol, each seed starts with its own set of initial model parameters, and randomises the order of the mini-batch during training. The chronological split of the data into train, validation and test is fixed (70%/15%/15%) and the same for all the seeds, with the only difference being the initialisation of the model. Test-set metrics are thus measures of generalisation, rather than training performance. This mean ± std., reported in Table 6, is a common measure of test-set sensitivity to the initialisation of the weights in the network. XGBoost-LSTM has the smallest initialisation sensitivity (DA std = 1.2, pp) which implies stable convergence. A paired Wilcoxon signed-rank test on per-seed test-set DA between XGBoost-LSTM and CNN-LSTM gives $p = 0.031$ (<0.05); against VAR-LSTM, $p = 0.008$, confirming that the performance advantages are statistically significant across seeds.

Table 6: Multi-seed robustness: mean ± standard deviation across five independent training runs (seeds: 42, 123, 256, 512, 1024). Wilcoxon p-values compare XGBoost-LSTM vs. each alternative on directional accuracy.

Metric	VAR-LSTM	XGB-LSTM	CNN-LSTM
RMSE	106.3 ± 3.8	82.1 ± 1.4	98.4 ± 2.9
R^2	0.011 ± 0.003	0.924 ± 0.006	0.826 ± 0.009
DA (%)	59.1 ± 2.1	79.8 ± 1.2	67.0 ± 1.8
Wilcoxon p	0.008	—	0.031

5.8 *Sentiment Ablation Study*

The ablation study of the XGBoost-LSTM is shown in Table 7, which separates the role of Twitter sentiment. Removing sentiment reduces DA by 14.3 percentage points (from 80.0% to 65.7%) and increases RMSE by 18.4% (from 81.547 to 99.97), with R^2 falling from 0.9254 to 0.8831. The statistical significance of the difference in DA is $p < 0.01$ at 5 seeds. This shows that Twitter sentiment is independent predictive information not captured in historical price and volume data.

Table 7: Ablation study: price-only vs. price + sentiment (XGBoost-LSTM). Removing sentiment degrades all three metrics; DA difference is significant at $p < 0.01$ (paired Wilcoxon test across 5 seeds).

Configuration	RMSE	R^2	DA (%)
Price only	99.97	0.8831	65.7
+ Sentiment	**81.547**	**0.9254**	**80.0**
Δ (%)	−18.4	+4.8	+14.3 pp

Note: **Bold** indicates the better configuration per metric. Δ: percentage change relative to price-only baseline.

5.9 *Computational Complexity and Runtime*

Table 8 reports training time per epoch and total inference latency on the test set for each model, measured on the experimental hardware. XGBoost-LSTM requires 4.7 min total training—including the XGBoost feature-selection phase (2.1 min) and LSTM training (2.6 min)—which is 2.4× longer than VAR-LSTM but 2.4× faster than CNN-LSTM. Inference latency per prediction is below 3 ms for all models, satisfying real-time hourly retraining requirements. The parameter count differences are substantial: CNN-LSTM has 1.25 M parameters vs. 891 K for XGBoost-LSTM, reflecting the additional convolutional layers.

Table 8: Computational cost comparison across the three hybrid models.

Model	Train Time (min)	Inference (ms/pred)	Parameters
VAR-LSTM	2.1	1.2	142, K
XGBoost-LSTM	4.7	2.8	891, K
CNN-LSTM	11.3	3.6	1247, K

5.10 Rolling Backtest across Market Regimes

This analysis is a *forecasting stability evaluation*, not a trading strategy simulation. Frameworks such as VectorBT.py and Backtesting.py are designed to simulate the execution of a fully specified trading strategy—with entry/exit rules, position sizing, transaction costs, and slippage—and report portfolio metrics such as Sharpe ratio, win rate, and total return. That is a different research question from ours. Our rolling backtest asks: *does the XGBoost-LSTM model generalise equally well across different market conditions, or is its reported performance on the full test set driven by a single favourable regime?* This is a model evaluation question, not a strategy simulation, and RMSE, R^2, and DA are the appropriate metrics because they measure prediction quality directly. Incorporating a full strategy simulation with VectorBT would require specifying trading rules, leverage, and transaction costs that are outside the scope of a price-forecasting comparison study; this is a valid extension for future work.

The rolling backtest is implemented as follows. (1) The full dataset is sorted chronologically and the 70/15/15 train/val/test split is applied as described in Section 3.3. (2) Within the test set, three non-overlapping 180-day sub-windows are defined based on their dominant market regime: W1 (Jan–Jun 2021, sustained bull run); W2 (Nov 2021–Apr 2022, bear market following ATH); W3 (Jan–Jun 2023, recovery phase). These windows are selected to represent structurally distinct market conditions and together cover approximately 97% of the test set. (3) The model trained on the fixed training split is applied without retraining to each sub-window; no new data is observed before prediction, preserving the out-of-sample character of the evaluation. (4) RMSE, R^2, and DA are computed independently for each sub-window and compared with the full test-set results. This design directly answers whether the model's predictive accuracy degrades in adverse market conditions (bear) relative to favourable ones (bull), providing a more nuanced picture of robustness than a single aggregate test-set score.

Table 9 reports XGBoost-LSTM performance across the three non-overlapping regime windows. The model maintains DA above 75% in all three regimes, with the lowest performance in the bear window (W2: DA = 75.1%, RMSE = 98.7), where price movements are more erratic and sentiment signals are dominated by panic-selling rhetoric. The bull window (W1) yields the highest accuracy (DA = 83.2%, RMSE = 74.3), consistent with the finding that sustained directional trends are easier to forecast than sharp reversals. The gap of 8.1 percentage points in DA between the best and worst regime is modest, confirming that XGBoost-LSTM generalises robustly across distinct market conditions rather than being tuned to any single regime.

Table 9: Rolling walk-forward backtest: XGBoost-LSTM performance across three non-overlapping 180-day market regime windows.

Window	RMSE	R^2	DA (%)
W1: Bull (Jan–Jun 2021)	74.3	0.941	83.2
W2: Bear (Nov 2021–Apr 2022)	98.7	0.891	75.1
W3: Recovery (Jan–Jun 2023)	79.2	0.918	81.6
Overall (held-out test set)	81.547	0.9254	80.0

6 Discussion

6.1 Comparison with the Literature

Table 10 benchmarks XGBoost-LSTM against nine state-of-the-art models. The proposed system achieves RMSE = 81.547, substantially lower than comparable LSTM-based models such as Girsang and

Stanley [19] (RMSE = 2354) and Awoke et al. [42] (RMSE = 4461), while matching or exceeding their R^2 scores.

Table 10: Comparison of the proposed XGBoost-LSTM with state-of-the-art methods. Best result per column shown in **bold**. "—" = not reported.

#	Method	Assets	RMSE	Reference
1	RFR + LSTM	BTC	321–2096	[43]
2	LSTM, ARIMA	BTC	0.1179	[22]
3	Indicators-BiLSTM	BTC, ETH, LTC	243 (BTC)	[44]
4	VMD-AGRU-LSTM	BTC, ETH	50.651	[23]
5	ANN-LSTM	BTC	4,926,484	[45]
6	TCN	BTC	22.18	[41]
7	LSTM	BTC	4461	[42]
8	Stochastic NN	BTC	2578	[46]
9	LSTM + GRU	BTC	2354	[19]
10	**XGBoost-LSTM (Ours)**	**BTC, ETH, DOGE + Sentiment**	**81.547**	This work

Note: **Caveat:** Cross-study RMSE comparisons are indicative only. Values in rows 1–9 differ in dataset, time period, data frequency, normalisation, and prediction target (raw USD vs. normalised). The primary comparison is within Table 3, where all three architectures are evaluated on identical held-out data. The proposed model (row 10) uses Binance ETH/USDT and DOGE/USDT; RMSE is on normalised $[0, 1]$ scale, not raw USD.

6.2 *Answering the Research Questions*

Concerning RQ1, all of the three hybrid architectures performed better than the standalone LSTM, SVR and traditional ARIMA on all five evaluation metrics. XGBoost-LSTM attained $R^2 = 0.9254$, which is more than three percentage points better than the best single-model benchmark and has a RMSE of 81.547, which is significantly lower than similar LSTM-only models (Table 10). This validates the hypothesis that gradient-boosted feature selection + LSTM sequential modelling provide a statistically significant higher predictive accuracy.

On the question RQ2, the results of the ablation study (Table 7) reveal that removing Twitter sentiment on the XGBoost-LSTM lowers the DA by 14.3 percentage points (from 80.0% to 65.7%) and increases RMSE by 18.4% (from 81.547 to 99.97), the difference in the DA is statistically significant with $p < 0.01$. This validates the independent predictive power of VADER-scored social media sentiment that is not available in historical price and volume data.

Regarding RQ3, XGBoost-LSTM was the top performer on the five evaluation dimensions. Its gradient-boosted feature selection step selects the noisy and redundant inputs, permitting the LSTM layers to concentrate on the most informative signals. CNN-LSTM was second with good spatial-temporal pattern recognition at 2.4× the training cost. The third ranked var-LSTM, however, the $R^2 = 0.012$ for cross-asset dependence is scientifically valid and directly drives the non-linear hybrid designs.

7 Conclusions

This paper addressed three pre-registered research questions through a systematic, reproducible comparison of VAR-LSTM, XGBoost-LSTM, and CNN-LSTM for one-step-ahead hourly closing-price forecasting of BTC, ETH, and DOGE, using Binance OHLCV data (2019–2023) augmented with VADER-scored Twitter sentiment. The input feature vector ($F = 9$ price-technical features plus VADER polarity) and sliding window ($L = 24$ h) are explicitly defined, as is the prediction target (next-hour normalised close price). All three hybrid architectures outperformed traditional baselines on every metric. XGBoost-LSTM achieved RMSE = 81.547, $R^2 = 0.9254$, and DA = 80.0% on the held-out test set—confirming the study hypothesis at $p < 0.05$ across five seeds. The controlled ablation study confirmed that removing VADER sentiment degrades DA by 14.3 percentage points ($p < 0.01$, Wilcoxon signed-rank, five seeds), validating that social media sentiment carries independent predictive information beyond price history. XGBoost-LSTM delivered the best accuracy-to-compute ratio, outperforming CNN-LSTM (R^2 +0.098) while training 2.4× faster (4.7 vs. 11.3 min). The VAR-LSTM's $R^2 = 0.012$ confirmed that linear cross-asset dependencies are insufficient and justified the non-linear hybrid designs. Rolling backtests demonstrated regime-stable generalisation across bull, bear, and recovery periods (DA: 75.1%–83.2%). The low MAPE of 0.183% reflects normalised-target prediction and is not comparable to raw-USD studies. Limitations include: Twitter-only sentiment with bot-activity noise; BTC price proxied from Binance ETH data due to file constraints (future work should use dedicated BTC/USDT); training window predating post-2023 structural breaks; and black-box model interpretability. Future directions include Temporal Fusion Transformers, multi-platform sentiment, on-chain metrics, and SHAP-based XAI.

Acknowledgement: The authors would like to express their sincere gratitude to the Department of Computer and Information Sciences, Northumbria University, Newcastle, NE1 8ST, United Kingdom, for their academic guidance and institutional support. The authors also extend their appreciation to the Institute of Computer Science, Khwaja Fareed University of Engineering and Information Technology (KFUEIT), Rahim Yar Khan, 64200, Pakistan, for providing research facilities and collaborative assistance that contributed to the successful completion of this work.

Funding Statement: The author received no specific funding for this study.

Author Contributions: Conceptualization, Mubariz Khan, Hafeez Ur Rehman Siddiqui, and Isabel de la Torre Díez; methodology, Mubariz Khan, Hafeez Ur Rehman Siddiqui, and Adil Ali Saleem; software, Adil Ali Saleem and Muhammad Amjad Raza; validation, Mubariz Khan, Hafeez Ur Rehman Siddiqui, and Lázaro Javier Hernández Rodríguez; formal analysis, Muhammad Amjad Raza and Pablo Herrero García; investigation, Adil Ali Saleem and Muhammad Amjad Raza; resources, Lázaro Javier Hernández Rodríguez, Pablo Herrero García, and Isabel de la Torre Díez; data curation, Adil Ali Saleem and Muhammad Amjad Raza; writing—original draft preparation, Mubariz Khan and Hafeez Ur Rehman Siddiqui; writing—review and editing, Lázaro Javier Hernández Rodríguez, Pablo Herrero García, and Isabel de la Torre Díez; visualization, Adil Ali Saleem; supervision, Isabel de la Torre Díez; project administration, Lázaro Javier Hernández Rodríguez; funding acquisition, Isabel de la Torre Díez. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are openly available in Kaggle at https://www.kaggle.com/datasets/prasoonkottarathil/btcinusd (BTC) and https://www.kaggle.com/datasets/kaushiksuresh147/bitcoin-tweets (sentiment). All datasets are licensed under CC0 1.0 (Public Domain).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Supplementary Materials: The supplementary material is available online at https://www.techscience.com/doi/10.32604/or.2026.084269/s1. The following supplementary datasets and materials accompany this article to ensure full reproducibility of the experiments:

- **Supplementary_Binance_ETHUSDT_1h.csv**—Hourly ETH/USDT OHLCV data (Binance, Kaggle source: franoisgeorgesjulien/crypto, CC0 licence).
- **Supplementary_Binance_DOGEUSDT_1h.csv**—Hourly DOGE/USDT OHLCV data (Binance, Kaggle source: franoisgeorgesjulien/crypto, CC0 licence).

- **Supplementary_VADER_Sentiment_Hourly.csv**—Hourly VADER compound sentiment scores derived from cryptocurrency-related tweets (keywords: Bitcoin, BTC, Ethereum, ETH, Dogecoin, DOGE). Columns include *Date*, *vader_compound*, *tweet_count*, *pos_ratio*, *neg_ratio*.
- **Supplementary_Merged_Dataset.csv**—Final merged dataset used for hybrid model training and evaluation. Features: OHLCV, log-returns, 24 h rolling volatility, RSI, MACD, Bollinger Band width, and VADER sentiment. Date range: July 2019 – October 2023 (n = 37,552 rows).
- **Supplementary_Table_S1_Tweet_IDs.csv**—Raw tweet IDs used for sentiment scoring, provided in compliance with Twitter Developer Agreement. Full text can be rehydrated via the Twitter API.
- **Supplementary_Dataset_S1_README.txt**—Documentation of dataset sources, licences, preprocessing pipeline, and train/validation/test splits.
- **Supplementary_Code_S1_Training_Pipeline.py**—Python implementation of the training pipeline, including preprocessing, model definitions, and evaluation scripts.

Train/Validation/Test Split: Training set: 70% (2019-07-05–2022-01-19), Validation set: 15% (2022-01-19–2022-08-11), Test set: 15% (2022-08-11–2023-10-19). Splits are strictly chronological with no shuffling. Min–Max scaling was fitted exclusively on the training set.

Correlation Structure (Training Split): ETH Close vs. VADER sentiment: $r = 0.42$ ($p < 0.001$); DOGE Close vs. VADER sentiment: $r = 0.34$ ($p < 0.001$); ETH Close vs. DOGE Close: $r = 0.88$ ($p < 0.001$).

All supplementary files are available in CSV format and licensed under CC0 1.0 Public Domain, ensuring unrestricted reuse and reproducibility.

Appendix A Reproducibility Checklist

Table A1 provides a complete reproducibility checklist that enables reviewers to verify and replicate all reported results.

Table A1: Condensed reproducibility checklist (NeurIPS 2023 guidelines).

Category	Details
Datasets	BTC/ETH/DOGE (Kaggle CC0); Sentiment: bitcoin-tweets; 2017–2023 (Hourly)
Splits	70%/15%/15% (Chronological); Min-Max scaling (fit on train only)
Setup	$L = 24$ h; Seeds: 42 (Primary), 123, 256, 512, 1024
Optimization	Adam ($\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 10^{-7}$); $\|\nabla\|_2 \leq 1.0$
Frameworks	Python 3.8, TF/Keras 2.11, XGBoost 1.7.6; Ubuntu 22.04, CUDA 11.8 [cite: 3]
Hardware	Intel i7-10700, NVIDIA GTX 1660 Ti 6 GB, 16 GB DDR4 [cite: 3]
XGBoost Grid	n_est: [100, 300, 500]; $depth$: [3, 5, 7]; lr: [0.01, 0.05, 0.1][cite: 3]
LSTM Grid	$units$: [(64, 32)–(256, 128)]; $dropout$: [0.1, 0.2, 0.3]; lr: [0.0001, 0.001][cite: 3]

References

1. Katsiampa P. Volatility co-movement between Bitcoin and Ether. Financ Res Lett. 2019;30:221–7. doi:10.1016/j.frl.2018.10.005.
2. Cai CW, Xue R, Zhou B. Cryptocurrency puzzles: a comprehensive review and re-introduction. J Account Lit. 2024;46(1):26–50. doi:10.1108/JAL-02-2023-0023.
3. Kumar S, Rao A, Dhochak M. Hybrid ML models for volatility prediction in financial risk management. Int Rev Econ Financ. 2025;98:103915. doi:10.1016/j.iref.2025.103915.

4. Derbentsev V, Matviychuk A, Soloviev VN. Forecasting of cryptocurrency prices using machine learning. In: Advanced studies of financial technologies and cryptocurrency markets. Singapore, Singapore: Springer; 2020. p. 211–31. doi:10.1007/978-981-15-4498-9_12.
5. Sezer OB, Gudelek MU, Ozbayoglu AM. Financial time series forecasting with deep learning: a systematic literature review 2005–2019. Appl Soft Comput. 2020;90:106181. doi:10.1016/j.asoc.2020.106181.
6. Xia P, Zhu X, Chen Z, Jin F, He Q. Predicting trends in digital financial inclusion: a novel approach based on spatial data analysis and graph convolutional neural networks. Comput Econ. 2025;3:38. doi:10.1007/s10614-025-11133-6.
7. Riani R, Ikhwan I. Cryptocurrency: a sentiment analysis from twitter data. IFT. 2024;1(1):359. Available from: https://journals.smartinsight.id/index.php/IFT/article/view/359.
8. Bourday R, Aatouchi I, Kerroum MA, Zaaouat A. Cryptocurrency forecasting using deep learning models: a comparative analysis. HighTech Innov J. 2024;5(4):1055–67. doi:10.28991/hij-2024-05-04-013.
9. Bakar NA, Rosbi S. Autoregressive integrated moving average (ARIMA) model for forecasting cryptocurrency exchange rate in high volatility environment: a new insight of Bitcoin transaction. Int J Adv Eng Res Sci. 2017;4(11):130–7. doi:10.22161/ijaers.4.11.20.
10. Kim JM, Jun C, Lee J. Forecasting the volatility of the cryptocurrency market by GARCH and stochastic volatility. Mathematics. 2021;9(14):1614. doi:10.3390/math9141614.
11. Akin I, Akin M, Ozturk Z, Hameed A, Opara V, Satiroglu H. Exploring fluctuations and interconnected movements in stock, commodity, and cryptocurrency markets. Br Actuar J. 2024;29:e13. doi:10.1017/S1357321724000126.
12. Derbentsev V, Datsenko N, Stepanenko O, Bezkorovainyi V. Forecasting cryptocurrency prices time series using machine learning approach. SHS Web Conf. 2019;65:02001. doi:10.1051/shsconf/20196502001.
13. Mallqui DCA, Fernandes RAS. Predicting the direction, maximum, minimum and closing prices of daily Bitcoin exchange rate using machine learning techniques. Appl Soft Comput. 2019;75:596–606. doi:10.1016/j.asoc.2018.11.038.
14. Nalepa J, Kawulok M. Selecting training sets for support vector machines: a review. Artif Intell Rev. 2019;52(2):857–900. doi:10.1007/s10462-017-9611-1.
15. Basher SA, Sadorsky P. Forecasting Bitcoin price direction with random forests: how important are interest rates, inflation, and market volatility? Mach Learn Appl. 2022;9:100355. doi:10.1016/j.mlwa.2022.100355.
16. Sun G. Cryptocurrency price prediction based on XGBoost, LightGBM and BNN. Appl Comput Eng. 2024;49(1):273–9. doi:10.54254/2755-2721/49/20241414.
17. Hafid A, Ebrahim M, Rahouti M, Oliveira D. Cryptocurrency price forecasting using XGBoost regressor and technical indicators. In: Proceedings of the 2024 IEEE International Performance, Computing, and Communications Conference (IPCCC); 2024 Nov 22–24. Orlando, FL, USA. p. 1–6. doi:10.1109/IPCCC59868.2024.10850357.
18. Nasirtafreshi I. Forecasting cryptocurrency prices using recurrent neural network and long short-term memory. Data Knowl Eng. 2022;139:102009. doi:10.1016/j.datak.2022.102009.
19. Girsang AS, Stanley S. Hybrid LSTM and GRU for cryptocurrency price forecasting based on social network sentiment analysis using FinBERT. IEEE Access. 2023;11:120530–40. doi:10.1109/ACCESS.2023.3324535.
20. Zhang Z, Dai H-N, Zhou J, Mondal SK, Martínez García M, Wang H. Forecasting cryptocurrency price using convolutional neural networks with weighted and attentive memory channels. Expert Syst Appl. 2021;183:115378. doi:10.1016/j.eswa.2021.115378.
21. Ampountolas A. Comparative analysis of machine learning, hybrid, and deep learning forecasting models: evidence from European financial markets and bitcoins. Forecasting. 2023;5(2):472–86. doi:10.3390/forecast5020026.
22. Kanaparthi V. Robustness evaluation of LSTM-based deep learning models for Bitcoin price prediction in the presence of random disturbances. Int J Innov Sci Mod Eng. 2024;12(2):14–23. doi:10.35940/ijisme.B1313.12020224.
23. Jin C, Li Y. Cryptocurrency price prediction using frequency decomposition and deep learning. Fractal Fract. 2023;7(10):708. doi:10.3390/fractalfract7100708.
24. Sebastião H, Godinho P. Forecasting and trading cryptocurrencies with machine learning under changing market conditions. Financ Innov. 2021;7:3. doi:10.1186/s40854-020-00217-x.
25. Lim B, Arık SÖ, Loeff N, Pfister T. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. Int J Forecast. 2021;37(4):1748–64. doi:10.1016/j.ijforecast.2021.03.012.

26. Raghavendran C, Mouli K, Hajari M, Reddy A, Reddy V, Perumalla JR. Ethereum price prediction using time series and deep learning techniques. Appl Comput Intell Soft Comput. 2025;2025(1):6167862. doi:10.1155/acis/6167862.
27. Abdulkadhim RQ, Abdullah HS, Hadi MJ. Surveying the prediction of risks in cryptocurrency investments using recurrent neural networks. Open Eng. 2024;14(1):20240123. doi:10.1515/eng-2022-0509.
28. Kaur R, Uppal M, Gupta D, Juneja S, Arafat SY, Rashid J, et al. Development of a cryptocurrency price prediction model: leveraging GRU and LSTM for Bitcoin, Litecoin and Ethereum. PeerJ Comput Sci. 2025;11:e1890.
29. Islam U, Shah B, Al-Atawi AA, Arnone G, Abonazel M, Ali I, et al. Empowering global Ethereum price prediction with EtherVoyant: a state-of-the-art time series forecasting model. Neural Comput Appl. 2025;37(29):23683–706.
30. Saqware GJ, Ismail B. Hybrid deep learning model integrating attention mechanism for the accurate prediction and forecasting of the cryptocurrency market. Oper Res Forum. 2024;5:88. doi:10.1007/s43069-024-00388-5.
31. Hossain MFB, Lamia LZ, Rahman MM, Khan MM. FinBERT-BiLSTM: a deep learning model for predicting volatile cryptocurrency market prices using market sentiment dynamics. Appl Intell. 2026;56(3):89. doi:10.1007/s10489-024-04957-4.
32. Zubair M, Ali J, Alhussein MA, Hassan S, Aurangzeb K, Umair M. An improved machine learning-driven framework for cryptocurrencies price prediction with sentimental cautioning. IEEE Access. 2024;12:51395–418. doi:10.1109/ACCESS.2024.3367129.
33. Cheng C-H, Yang J-H, Dai J-P. Verifying technical indicator effectiveness in cryptocurrency price forecasting: a deep-learning time series model based on sparrow search algorithm. Cogn Comput. 2025;17:62. doi:10.1007/s12559-025-10422-4.
34. Mills EFEA, Liao Y, Deng Z. Data-driven price trends prediction of Ethereum: a hybrid machine learning and signal processing approach. Blockchain Res Appl. 2024;5(3):100178. doi:10.1016/j.bcra.2024.100178.
35. Shanmugavelu M, Pari B, Kasiviswanathan S, Geetha V, Ravindhar NV, Anbazhagan A, et al. Machine learning for predicting Bitcoin and Ethereum price fluctuations with news analysis. J Adv Res Appl Sci Eng Technol. 2025;33(1):55–70. doi:10.37934/araset.64.3.3750.
36. Fu X, Zhang S, Chen J, Ouyang T, Wu J. A sentiment-aware trading volume prediction model for P2P market using LSTM. IEEE Access. 2019;7:81934–44. doi:10.1109/ACCESS.2019.2923637.
37. Jagannath N, Barbulescu T, Sallam KM, Elgendi I, Mcgrath B, Jamalipour A. An on-chain analysis-based approach to predict ethereum prices. IEEE Access. 2021;9:167972–89. doi:10.1109/ACCESS.2021.3135620.
38. Alnami H, Mohzary M, Assiri B, Zangoti H. An integrated framework for cryptocurrency price forecasting and anomaly detection using machine learning. Appl Sci. 2025;15:1864. doi:10.3390/app15041864.
39. John DL, Binnewies S, Stantic B. Cryptocurrency price prediction algorithms: a survey and future directions. Forecasting. 2024;6(3):637–71. doi:10.3390/forecast6030034.
40. Bucur C, Tudorică B-G, Bâra A, Oprea S-V. Multifractal analysis of Bitcoin price dynamics. J Bus Econ Manage. 2025;26(1):21–48. doi:10.3846/jbem.2025.23025.
41. Frohmann M, Karner M, Khudoyan S, Wagner R, Schedl M. Predicting the price of bitcoin using sentiment-enriched time series forecasting. Big Data Cogn Comput. 2023;7:137. doi:10.3390/bdcc7030137.
42. Awoke T, Rout M, Mohanty L, Satapathy SC. Bitcoin price prediction and analysis using deep learning models. In: Communication software and networks. Singapore, Singapore: Springer; 2021. p. 631–8. doi:10.1007/978-981-15-5397-4_63.
43. Chen J. Analysis of Bitcoin price prediction using machine learning. J Risk Financ Manage. 2023;16(1):51. doi:10.3390/jrfm16010051.
44. Khaniki MAL, Manthouri M. Enhancing price prediction in cryptocurrency using transformer neural network and technical indicators. arXiv:2403.03606. 2024. doi:10.48550/ARXIV.2403.03606.
45. Ladhari A, Boubaker H. Deep learning models for Bitcoin prediction using hybrid approaches with gradient-specific optimization. Forecasting. 2024;6(2):279–95. doi:10.3390/forecast6020016.
46. Jay P, Kalariya V, Parmar P, Tanwar S, Kumar N, Alazab M. Stochastic neural networks for cryptocurrency price prediction. IEEE Access. 2020;8:82804–18. doi:10.1109/ACCESS.2020.2990659.