Data Science in Finance and Economics

https://www.aimspress.com/journal/dsfe

DSFE, xx(x): xxx–xxx
DOI:10.3934/DSFE.2025xxx
Received:
Revised:
Accepted:
Published:

Research article

Enhancing Cryptocurrency Price Prediction Using Hybrid Regression and RNN Models

Wiem Ben Romdhane[1]*, **and Heni Boubaker**[1,2]

[1] LaREMFiQ, IHEC , University of Sousse - Tunisia
[2] Departement of Quantitative Methods and Economics, IHEC, University of Sousse - Tunisia

* **Correspondence:** Email wiembr@gmail.com

Abstract: The volatile and complex nature of cryptocurrency markets demands advanced forecasting techniques to enhance predictive accuracy. This study explores the efficacy of hybrid models combining Recurrent Neural Networks (RNNs)—specifically Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU)—with statistical methods like Ordinary Least Squares (OLS) and simple averaging for forecasting Bitcoin (BTC), Ethereum (ETH), and Litecoin (LTC) prices. Using daily closing price data from 2018 to 2025, we evaluate model performance via Root Mean Squared Error (RMSE), Root Mean Absolute Error (RMAE), and Mean Absolute Percentage Error (MAPE). Empirical results reveal that the combined OLS model excels for BTC and LTC, achieving the lowest error metrics (e.g., MAPE of 2.693% for BTC), while LSTM outperforms for ETH (RMSE: 114.52). Hybrid approaches demonstrate robustness, balancing the temporal learning strengths of RNNs with the stability of statistical methods. Visual comparisons further validate the models' alignment with actual price trends. This research underscores the potential of hybrid models to mitigate volatility challenges, offering actionable insights for investors and advancing methodological innovation in cryptocurrency forecasting.

Keywords: Cryptocurrency Forecasting; OLS; LSTM; GRU; Combined models; Time series analysis

1. Introduction

The rapid rise of cryptocurrencies, such as Bitcoin, Ethereum, and Litecoin, has revolutionized the financial landscape, offering decentralized and secure alternatives to traditional currencies. However, the highly volatile and unpredictable nature of cryptocurrency markets poses significant challenges for investors, policymakers, and researchers. Accurate forecasting of cryptocurrency prices is crucial for making informed decisions, yet the complexity of these markets, driven by factors such as market sentiment, global economic trends, and technological advancements, makes this task particularly challenging.

In recent years, machine learning and deep learning techniques have emerged as powerful tools for financial time series forecasting. Among these, Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) networks and Gated Gated Recurrent Unit (GRU), have shown remarkable success in capturing temporal dependencies and patterns in sequential data. LSTMs networks are a specialized form of Recurrent Neural Networks tailored for handling time-dependent data, making them particularly effective for tasks like time series forecasting. GRUs is an advanced variant of Recurrent Neural Networks specifically engineered to overcome certain shortcomings of traditional RNNs. A key issue with standard RNNs is the ”vanishing gradient problem,” where the gradients responsible for updating the network’s weights during training diminish to such an extent that the model struggles to learn long-term patterns in the data. GRUs mitigate this problem by incorporating gating mechanisms that allow for more efficient information flow over time.

However, the inherent noise and non-linearity of cryptocurrency data often limit the performance of standalone models. To address this, researchers have increasingly turned to combined or hybrid models that integrate the strengths of multiple approaches, such as combining RNNs with statistical methods, optimization algorithms, or other machine learning techniques.

This research paper explores the potential of combined models that leverage RNNs for forecasting cryptocurrency prices. By combining RNNs with complementary methodologies, such as ordinary least square (OLS) and simple average techniques, the study aims to enhance prediction accuracy and robustness. Through empirical analysis and comparative evaluation, this research seeks to contribute to the growing body of knowledge on cryptocurrency forecasting, offering insights into the development of more reliable and adaptive predictive models.

2. Related works

Predicting cryptocurrency prices has become a topic of great interest in recent years, driven by the increasing significance of digital assets in global financial markets. Due to the highly volatile and nonlinear nature of cryptocurrency markets, conventional statistical models often struggle to effectively capture the intricate patterns and dependencies within the data. Consequently, researchers have increasingly adopted machine learning and deep learning approaches, particularly Recurrent Neural Networks (RNNs), to enhance forecasting accuracy. In this context, hybrid models that combine RNNs with other techniques have demonstrated significant potential in overcoming the limitations of standalone models (Oyewola, et al. 2022; Ferdiansyah, et al. 2023; García-Medina, and Aguayo-Moreno 2024).

In their paper, (Zhang, et al. 2021) proposed the weighted and attentive memory convolutional neural network (WAMC) to enhance cryptocurrency price prediction accuracy. The model integrates a GRU for attentive memory, a channel-wise weighting module to capture interdependencies among cryptocurrencies, and a CNN to extract local temporal features. Tested on ETH and BTC, the WAMC model achieved RMSE scores of 9.70 and 1.37, respectively, demonstrating its potential for accurate price forecasting. (Aghashahi, and Bamdad 2023) explore how integrating technical attributes, such as price-related and lagged features, into neural networks enhances prediction accuracy, ultimately influencing profitability. They evaluate model performance and examine how specific configurations contribute to reliable and effective predictions for profitable trading. (García-Medina, and Aguayo-

Moreno 2024) predict the volatility of major cryptocurrencies using Generalized Autoregressive Conditional Heteroskedasticity (GARCH) models, Multilayer Perceptron (MLP), Long Short-Term Memory (LSTM), and hybrid LSTM-GARCH models. They conclude that MLP models yield the best predictive performance while their accuracy is not statistically distinct from LSTM and LSTM-GARCH models based on the Diebold–Mariano test. (Seabe, et al. 2023) employ three deep learning techniques -LSTM, GRU and Bi-LSTM to predict the price of the top three cryptocurrencies BTC, ETH and LTC and they find that the Bi-LSTM model delivered the most accurate predictions across all three currencies. The study of (Xiang, 2024) explores LSTM, Neural Networks, and Deep Learning (CNN) for predicting cryptocurrency prices, utilizing technical indicators and sentiment analysis. The hybrid model combining these techniques achieves the highest predictive accuracy for cryptocurrencies like Bitcoin, Ethereum, and Binance Coin, as measured by RMSE. Challenges include adaptability to market volatility and data noise, with future improvements potentially incorporating external factors and interdisciplinary approaches. The research highlights the potential of hybrid deep learning models in improving cryptocurrency valuation and risk management strategies. The study of (Amberkhani, et al. 2025) explores multiple machine learning and deep learning models and fbProphet, to improve cryptocurrency price forecasting. XGBoost and the LSTM-GRU hybrid demonstrated strong performance, though challenges like overfitting remain. The research of (Hoa, et al. 2025) proposes a novel hybrid model that integrates one-dimensional convolutional neural networks (1D-CNN) with long short-term memory (LSTM) networks to predict Ethereum's closing prices. The experimental findings reveal that the hybrid model achieves superior performance compared to baseline models, highlighting its potential to enhance forecasting accuracy in the highly volatile cryptocurrency markets. (Hafidi, et al. 2025) introduce a novel methodology for cryptocurrency price prediction by automating hyperparameter selection and analyzing interconnections among cryptocurrencies. The approach integrates deep learning techniques, including RNNs and LSTMs, with the Genetic Algorithm (GA) to optimize model parameters. By automating the hyperparameter tuning process and accounting for the relationships between different cryptocurrencies, the method significantly improves forecasting accuracy. This innovative solution is designed to address the challenges of dynamic and interconnected cryptocurrency markets, providing practical benefits for investors and contributing to academic advancements in the field. (Kiranmai Balijepalli , and Thangaraj 2025) develop a dynamic forecasting model using ensemble methods (GBR, RFR, SVR, MLP) to predict top 15 cryptocurrencies' prices, optimizing parameters for accuracy. Statistical and econometric models are evaluated, with validation curves enhancing prediction precision. The ensemble model leverages machine learning to improve forecasting performance. The study of (Amirshahi, and Lahmiri 2025) extracts sentiment from cryptocurrency-related tweets using an ensemble of three pre-trained language models. Twenty model versions were evaluated on 27 top cryptocurrencies, incorporating historical sentiment and price data for predictions. The model's flexible input layer adapts to different cryptocurrencies, optimizing accuracy. Findings indicate longer input sequences yield the most accurate predictions on average.

3. Materials and method

3.1. Overview of LSTM and GRU Models

3.1.1. LSTM Model

Long Short-Term Memory (LSTM) is an advanced recurrent neural network (RNN) architecture that integrates gating mechanisms to address the limitations of traditional RNNs, such as vanishing gradient problems (Hochreiter, and Schmidhuber 1997). Specifically designed for sequential data, LSTM excels at capturing both short-term and long-term dependencies within time series, making it highly effective for tasks requiring temporal modeling [14]. Each LSTM unit consists of recurrent cells that maintain two key internal states over time: the cell state c_t, which stores long-term information, and the hidden state h_t, which captures short-term context. These cells are governed by nonlinear gating mechanisms—input, forget, and output gates—that regulate the flow of information. The input gate determines which new information to store, the forget gate decides what to discard from the cell state, and the output gate controls the information passed to the next time step. This dynamic gating structure enables LSTMs to adaptively learn and model complex temporal patterns, making them particularly suited for applications involving time series forecasting, natural language processing, and other sequential data tasks.

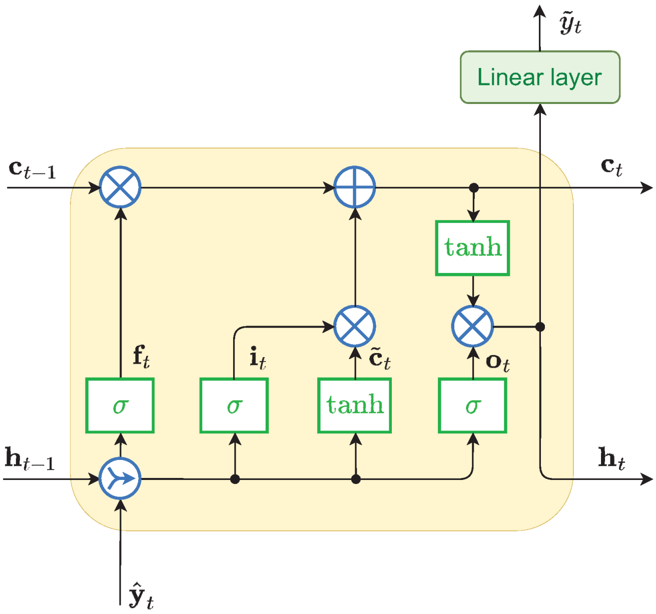

Figure 1. LSTM model - Source: (Dudek, 2023).

In Figure 1, the architecture of an LSTM block is illustrated, showcasing its key components: the input signal x_t, the output h_t, and the activation functions that govern its operations. The LSTM block is designed to manage information flow through a series of gating mechanisms, which are critical for its ability to learn and retain long-term dependencies. The input gate plays a pivotal role in determining which elements of the incoming data x_t should be stored in the cell state c_t, ensuring that only relevant information is preserved. Conversely, the output gate regulates the release of information from the cell state to the hidden state *ht*, controlling what is passed on to subsequent time steps. These gates, along with the forget gate (which decides what information to discard from the cell state), enable the LSTM to dynamically adapt to the temporal structure of the data.

The forward training process of an LSTM network can be mathematically described by the following set of equations, which define the operations within a single LSTM unit at time step *t*:

1. **Forget Gate:**

$$f_t = \sigma\left(W_f[h_{t-1}, x_t] + b_t\right) \tag{1}$$

2. **Input Gate:**

$$i_t = \sigma\left(W_i\dot{[}h_{t-1}, x_t] + b_i\right) \tag{2}$$

$$\tilde{c}_t = \tanh\left(W_c\dot{[}h_{t-1}, x_t] + b_c\right) \tag{3}$$

   Here, i_t controls the update of the cell state, and $\tilde{c}_t$ represents the candidate values for the new cell state.

3. **Cell State Update:**

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{4}$$

   The cell state c_t is updated by combining the information retained from the previous state and the new candidate values.

4. **Output Gate:**

$$o_t = \sigma\left(W_o\dot{[}h_{t-1}, x_t] + b_o\right) \tag{5}$$

$$h_t = o_t \odot \tanh(c_t) \tag{6}$$

   The output gate o_t determines the portion of the cell state c_t that will be output as the hidden state h_t.

These equations collectively enable the LSTM to process sequential data effectively, balancing the retention of long-term information with the incorporation of new inputs. The gating mechanisms ensure that the network can adaptively learn and model complex temporal patterns, making LSTMs a powerful tool for tasks such as time series forecasting, natural language processing, and more.

3.1.2. GRU Model

Gated Recurrent Units (GRUs) are a specialized variant of Recurrent Neural Networks (RNNs) introduced by (Cho, et al. 2014) as a streamlined and computationally efficient alternative to Long Short-Term Memory (LSTM) networks. Like LSTMs, GRUs are designed to process input sequences of variable lengths and maintain a hidden state that encapsulates historical information, enabling them to model temporal dependencies effectively. However, GRUs achieve this with a more simplified architecture, utilizing only two gating mechanisms: an update gate and a reset gate. The update gate controls how much of the previous hidden state should be retained and how much new information should be incorporated, while the reset gate determines which portions of the past information should be discarded. This reduction in gating mechanisms eliminates the need for a separate cell state, as used in LSTMs, resulting in a more compact and computationally efficient model.

The simplified design of GRUs offers several advantages, including faster training times and reduced complexity, making them particularly suitable for scenarios where computational resources are limited or real-time processing is required. Despite their architectural simplicity, GRUs have demonstrated performance comparable to LSTMs across a wide range of sequence modeling tasks, such as natural language processing, speech recognition, and time series forecasting (Yang, et al. 2020). This balance of efficiency and effectiveness has solidified GRUs as a popular choice for researchers and practitioners seeking robust yet lightweight solutions for sequential data analysis. The architecure of a GRU block is illustred in Figure 2.

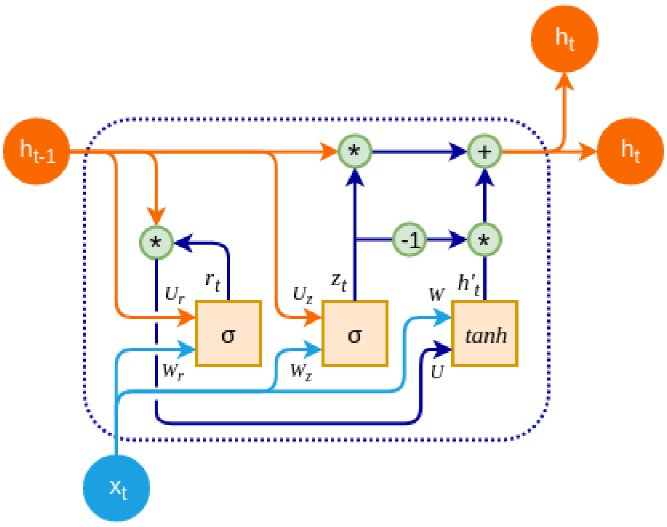

Figure 2. GRU model - Source: (Vasilev, 2019).

Below are the key equations that define the operations within a GRU unit at time step t:

1. **Reset Gate:**
$$r_t = \sigma\left(W_r\dot{[}h_{t-1}, x_t] + b_r\right) \tag{7}$$
The reset gate r_t determines how much of the previous hidden state h_{t-1} should be ignored when computing the candidate activation.
2. **Update Gate:**
$$z_t = \sigma\left(W_z\dot{[}h_{t-1}, x_t] + b_z\right) \tag{8}$$
The update gate z_t balances the influence of the previous hidden state h_{t-1} and the candidate hidden state $\tilde{h}_t$.
3. **Candidate Hidden State:**
$$\tilde{h}_t = \tanh\left(W_h[r_t \odot h_{t-1}, x_t] + b_h\right) \tag{9}$$
The candidate hidden state $\tilde{h}_t$ represents the proposed new state based on the current input and the filtered past information.
4. **Hidden State Update:**
$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{10}$$
The final hidden state h_t is a weighted combination of the previous hidden state and the candidate hidden state, controlled by the update gate.

3.2. Proposed Method

The proposed method combines the strengths of LSTM and GRU models by leveraging their individual outputs through a regression-based approach following theese steps:

Independent Training: LSTM and GRU models are trained separately on the same dataset to capture different aspects of temporal dependencies.
Output Generation: Both models generate forecasts for the same time steps, producing two sets of predictions, i.e. $f_t = (f_t^{\text{LSTM}}, f_t^{\text{GRU}})'$.
Combinaison Methods: Suppose y_t is a variable of interest and we have 2 predictors, $f_t^{\text{LSTM}}, f_t^{\text{GRU}}$. The goal of combinaison methods is to find a vector $w = (w_1, w_2)$ to form a new predictor $\widehat{y_t} = w_1 f_t^{\text{LSTM}} + w_2 f_t^{\text{GRU}}$ which optimal in terms of mean squared prediction: $\min_{w_i} E(y_t - \widehat{y_t})^2$.

- **Simple Average:** The outputs of LSTM and GRU are averaged to produce the final forecast, $\widehat{y_t} = \frac{1}{2} f_t^{\text{LSTM}} + \frac{1}{2} f_t^{\text{GRU}}$.
- **Bates-Granger Method:** (Bates, and Granger 1969) propose determining the weight for the i-th predictive model by optimizing the combination to minimize the forecast error variance and find that:

$$w_i^{BG} = \frac{\widehat{\sigma}^{-2}(i)}{\sum_{j=1} \widehat{\sigma}^{-2}(j)}$$

- **Regression-Based Combinaison:**A regression model (e.g., linear regression) is trained to combine the outputs of LSTM and GRU. The regression layer learns the optimal weights to assign to each model's output, minimizing the overall prediction error.

Final Prediction: The combined output from the regression layer serves as the final forecast, benefiting from the complementary strengths of LSTM and GRU.

This approach ensures that the hybrid model can adaptively leverage the long-term memory capabilities of LSTM and the computational efficiency of GRU, resulting in more accurate and robust forecasts. The comparison with simple average and Bates-Granger methods highlights the superiority of the regression-based approach in capturing complex dependencies and optimizing forecast accuracy.

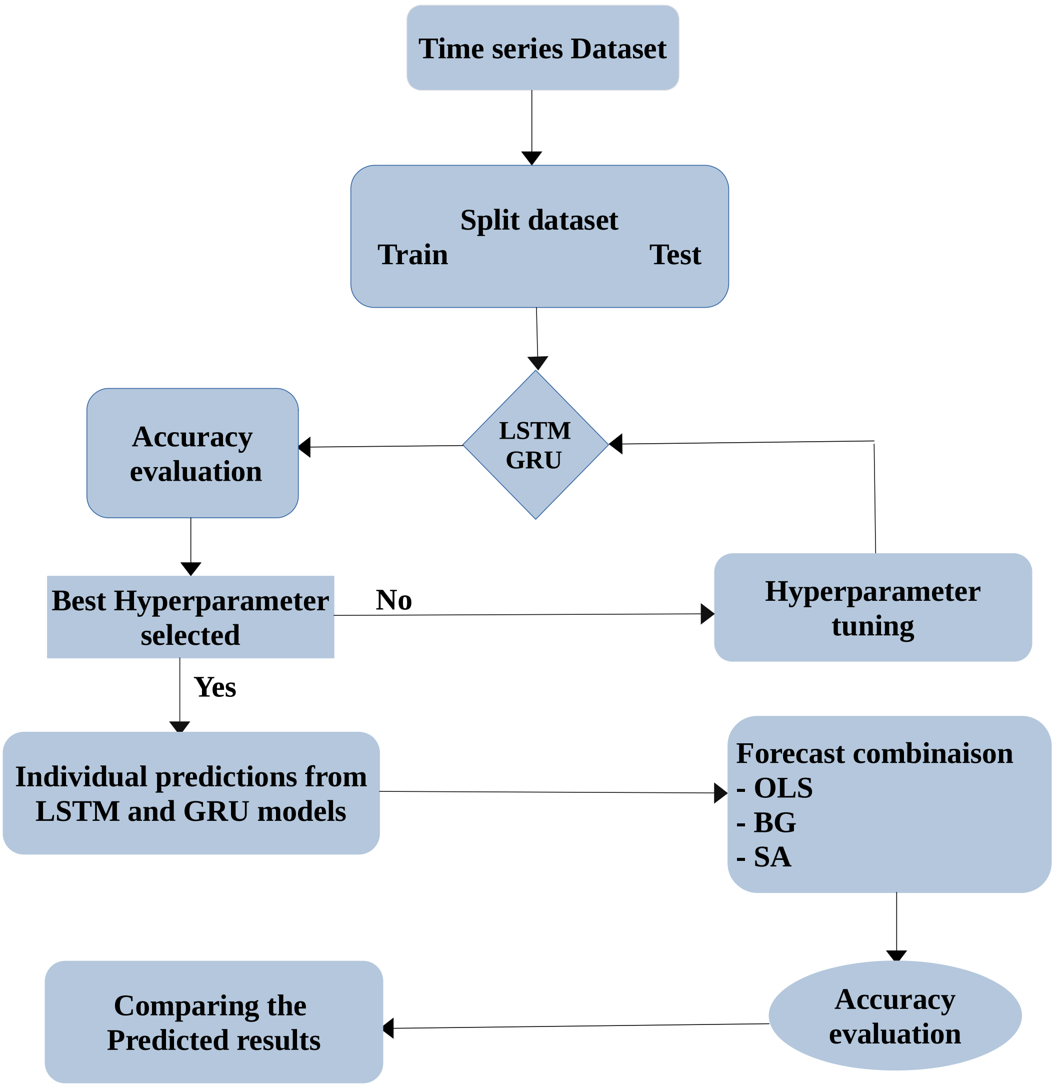

Figure 3. Overview of the forecasting combinaison system.

4. Empirical Results

4.1. Data and descriptive statistics

In this study, we use the three leading cryptocurrencies by market capitalization: Bitcoin (BTC), Ethereum (ETH), and Litecoin (LTC). The dataset comprises daily closing prices sourced from CoinMarketCap , covering the period from January 1, 2018, to March 22, 2025. Figure 4 illustrates the temporal evolution of these three time series, highlighting key trends and volatility patterns over the observed timeframe.

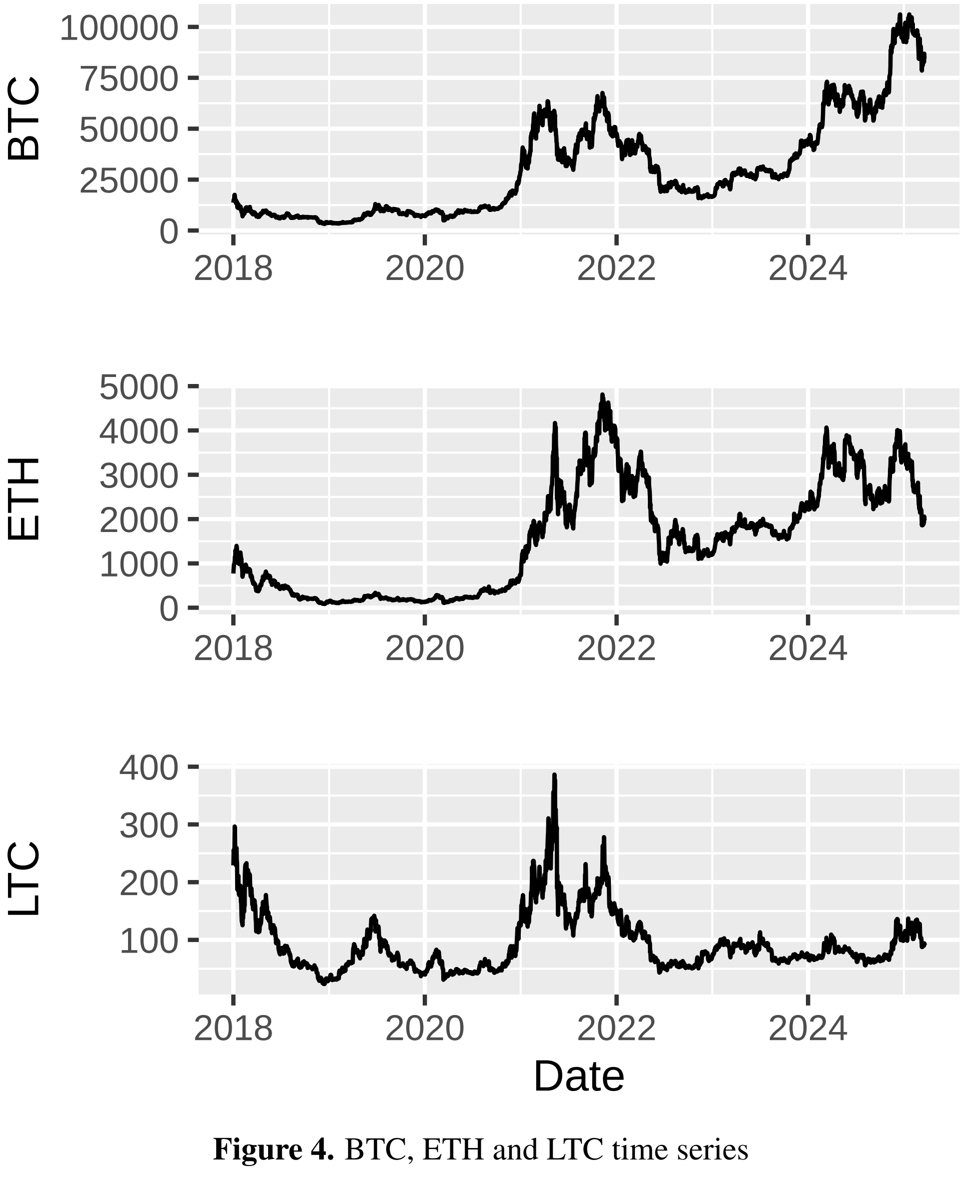

Figure 4. BTC, ETH and LTC time series

The descriptive statistics for the daily closing prices of Bitcoin (BTC), Ethereum (ETH), and Litecoin (LTC) are summarized in Table 1. Key metrics include measures of central tendency (mean, median), dispersion (standard deviation), and distributional properties (skewness, kurtosis), providing a comprehensive overview of the price dynamics across the study period.

	min	max	median	mean	std.dev	skewness	kurtosis
BTC	3236.762	106146.263	23175.891	30098.419	24624.704	1.027	0.353
ETH	84.308	4812.088	1561.749	1549.258	1234.662	0.454	-0.965
LTC	23.464	386.451	77.263	94.459	52.309	1.627	3.131

Table 1. Descriptive statistics

4.2. Pre-processing

Figure 3 outlines the pre-processing pipeline applied to the cryptocurrency dataset to ensure compatibility with deep learning architectures, including LSTM and GRU. Given the sensitivity of deep learning models to feature scales, we applied MinMax normalization to rescale the data into a $[0, 1]$ range, mitigating bias from disparate variable magnitudes. This approach aligns with established practices in time-series forecasting, where feature-wise scaling has been shown to enhance model convergence and performance (Ahsan, et al. 2021). To evaluate model generalizability, we partitioned the dataset using an 90:10 training-test split, preserving temporal continuity—a critical consideration for autoregressive forecasting tasks. The pre-processed data was then restructured into sequences to facilitate supervised learning with recurrent neural networks (RNNs).

4.3. Hyperparameter tuning

The performance of Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks in time series forecasting heavily depends on careful hyperparameter selection. Our study employs random search optimization to efficiently navigate the hyperparameter space and identify configurations that enhance predictive accuracy. This method systematically evaluates combinations of critical parameters including network depth (number of layers), model capacity (hidden units), training dynamics (learning rate and batch size), regularization (dropout rate), and temporal context (sequence length).

Hyperparameter	Range/Values
Number of layers	1 to 5 (integer)
Number of units	32 to 512 (Increments of 32)
Learning rate	0.0001, to 0.01 (Logarithmic scale)
Dropout rate	0.1 to 0.5 (Increments of 0.1)
Dense activation	['relu','sigmoid']

Table 2. Hyperparameter search space

Compared to exhaustive grid search, this approach offers superior computational efficiency by randomly sampling parameter combinations within predefined bounds while maintaining competitive model performance. Our implementation follows a structured workflow: first establishing parameter bounds (detailed in Table 2), then iteratively training and evaluating models using validation sets. We employ standard regression metrics - Mean Squared Error (MSE) and Mean Absolute Error (MAE) - to objectively compare model variants.

The optimization process culminates in selecting the top-performing configuration for final evaluation. This methodology provides an effective balance between thorough parameter exploration and practical computational requirements, making it particularly valuable for tuning complex recurrent architectures.

4.4. *Performance metrics*

To rigorously evaluate the performance of our proposed RNN algorithms, we employed three widely-adopted statistical metrics:

- Root Mean Squared Error (RMSE) - measuring the standard deviation of prediction errors:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \widehat{y_i})^2}{n}} \tag{11}$$

- Root Mean Absolute Error (RMAE) - assessing average error magnitude:

$$RMAE = \sqrt{\frac{\sum_{i=1}^{n} \left|y_i - \widehat{y_i}\right|}{n}} \tag{12}$$

- Mean Absolute Percentage Error (MAPE) - expressing accuracy as a percentage:

$$MAPE = \frac{100}{n} \sum_{i=1}^{n} \frac{|y_i - \widehat{y_i}|}{ai} \tag{13}$$

Where y_i represents the actual value (observed) , $\widehat{y_i}$ denotes the predicted value, and n stands for the total number of time steps.

These carefully selected metrics provide complementary perspectives on prediction accuracy. RMSE emphasizes larger errors through squaring, making it particularly sensitive to significant deviations. RMAE offers a more balanced view of typical error magnitudes, while MAPE provides an intuitive, scale-independent measure expressed as a percentage.

4.5. *Results and discussion*

The performance of various models in predicting the prices of Bitcoin (BTC), Ethereum (ETH), and Litecoin (LTC) is summarized in Table 3, with the optimal model identified based on the smallest error metrics.

A visual comparison between the actual prices and the predicted values generated by the LSTM and GRU models is presented in Figure 5 and Figure 6, respectively. Additionally, Figure 7 illustrates the forecasts produced by different hybrid forecasting approaches. The plots demonstrate that the predicted values closely follow the actual trends, albeit with minor deviations. These discrepancies are further quantified by the performance metrics in Table 3, providing a comprehensive assessment of each model's predictive accuracy.

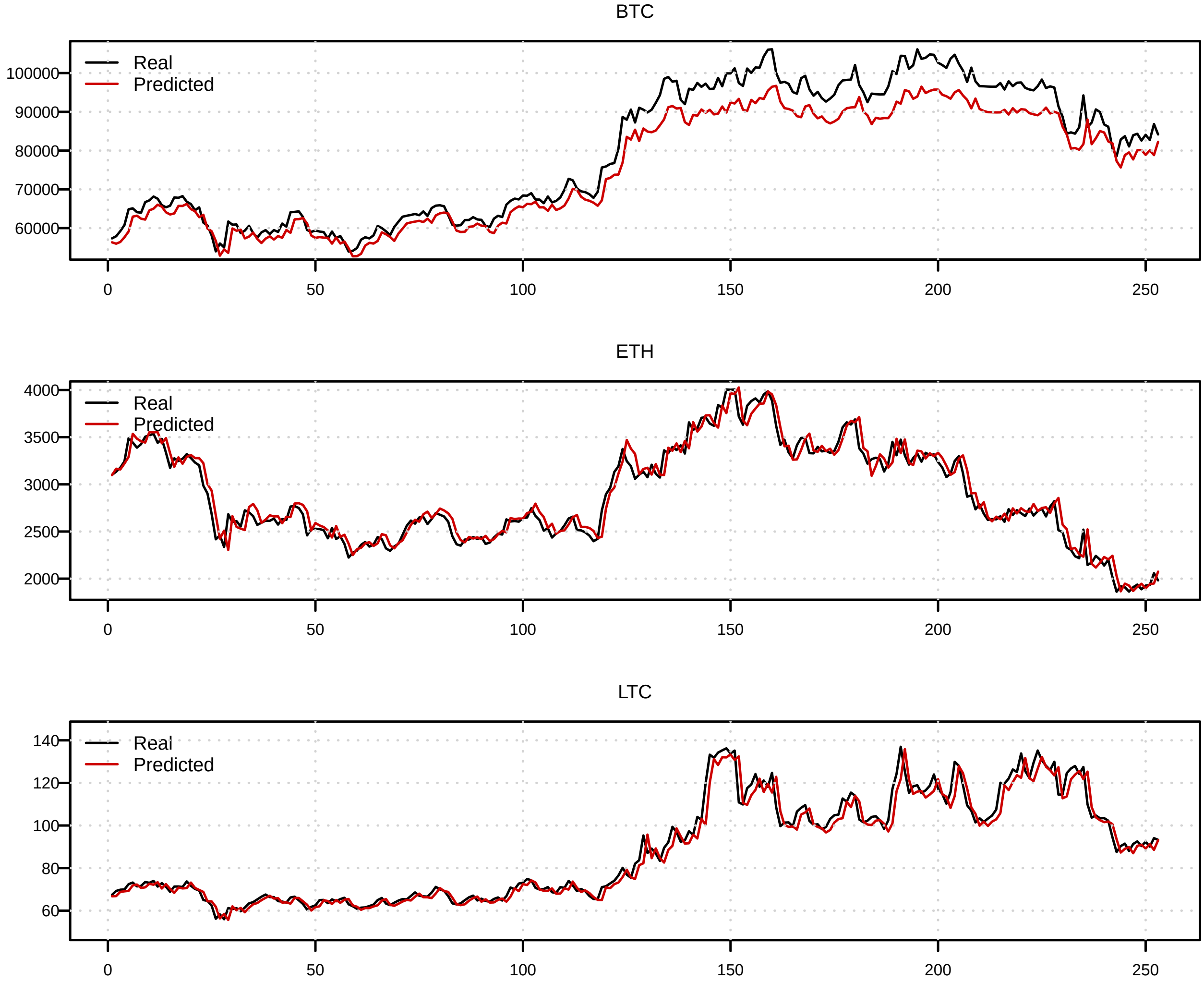

Figure 5. Actual and predicted values using LSTM model

4.5.1. Results for BTC

According to Table 3, the combined Ordinary Least Squares (OLS) model outperforms all others in predicting BTC prices, achieving the lowest RMSE, RMAE, and MAPE values. This superiority is visually confirmed in Figure 7, where the combined OLS predictions closely align with actual price movements, demonstrating high accuracy.

The results indicate that the combined OLS approach is more effective at capturing BTC trends than LSTM and GRU models, though the GRU remains competitive with only a marginal performance gap. Notably, the GRU model ranks second for BTC forecasting, exhibiting slightly elevated RMSE, RMAE, and MAPE values compared to OLS.

Overall, these findings suggest that hybrid statistical approaches (like combined OLS) surpass pure RNN-based methods (LSTM/GRU) in BTC price prediction, likely due to their robustness in handling linear trends and volatility.

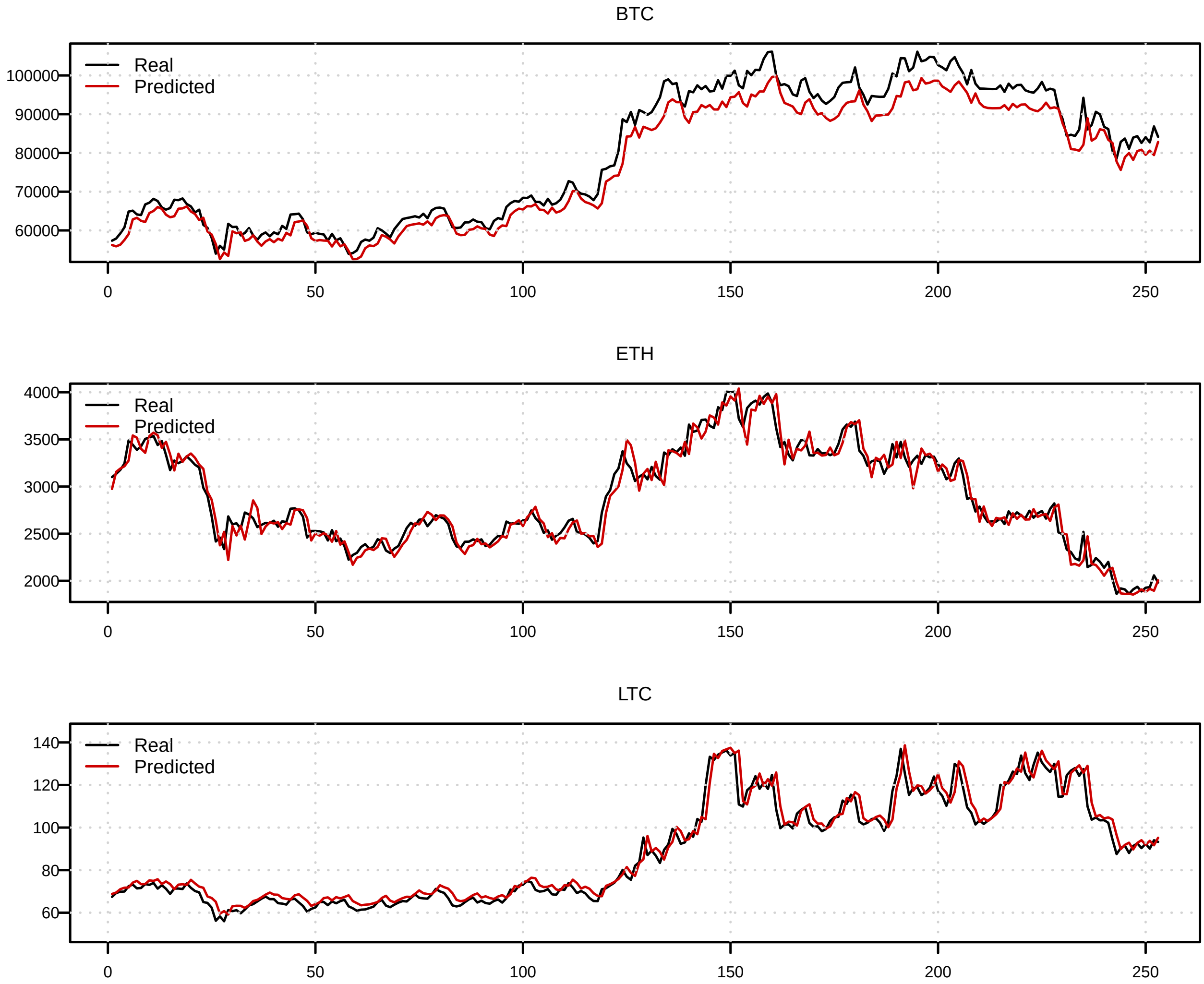

Figure 6. Actual and predicted values using GRU model

4.5.2. Results for ETH

The analysis of Table 3 and Figure 5 reveals that the LSTM model delivers the best performance in forecasting Ethereum (ETH) prices, achieving the lowest error metrics—an RMSE of 114.52 and an RMAE of 9.216. Figure 5 further supports this finding, demonstrating that the LSTM model's predictions exhibit the smallest deviation from actual ETH prices, highlighting its superior accuracy.

However, when evaluating forecast precision based on MAPE (2.987), the combined simple average model emerges as the top performer, suggesting its advantage in percentage error minimization.

The LSTM model's high predictive accuracy is also evident in Figure 16, where its forecasted ETH prices closely track the actual values throughout the testing period. The minimal discrepancies, as reflected by the low RMSE and RMAE, reinforce its reliability.

That said, a comparative assessment indicates that while the LSTM model outperforms GRU, its accuracy margin over combined models is relatively narrow, implying that hybrid approaches remain competitive alternatives for ETH price forecasting.

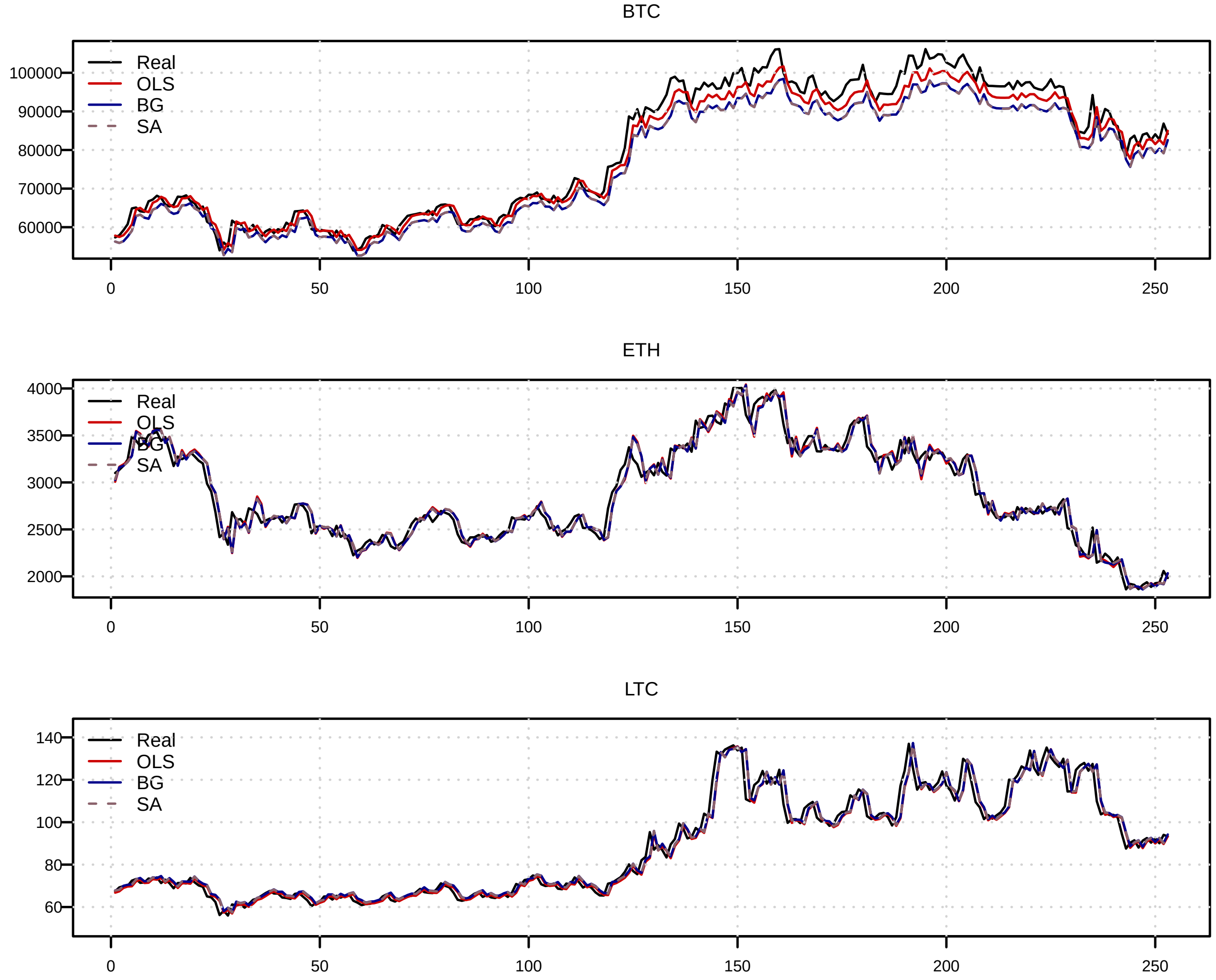

Figure 7. Actual and predicted values using different combinaison models

4.5.3. Results for LTC

Table 3 presents a comprehensive evaluation of model performance for Litecoin (LTC) price prediction, revealing distinct advantages among competing approaches. The combined Ordinary Least Squares (OLS) model emerges as particularly effective, achieving the lowest error metrics with an RMAE of 1.755 and MAPE of 3.214, indicating superior predictive accuracy among all tested models. However, the combined simple average model demonstrates the best performance in terms of RMSE (4.691), suggesting particular strength in minimizing large prediction errors. Visual analysis through **??** provides compelling graphical evidence of these quantitative findings. Figure 5 highlights the LSTM model's strong performance, with its predictions closely tracking actual LTC prices as evidenced by excellent error metrics (MAPE: 3.341; RMSE: 1.792). The combined OLS model's results, depicted in Figure 7, show marginally better performance than LSTM in percentage terms (MAPE: 3.214; RMAE: 1.755), though the visual representation suggests the practical differences between these top-performing models are relatively minor. This nuanced interpretation of both tabular and

Table 3. Forecasting quality metrics for different methods

	Train			Test		
	RMSE	RMAE	MAPE	RMSE	RMAE	MAPE
BTC						
LSTM	1390.264	31.308	4.563	5358.839	66.428	5.114
GRU	1308.704	28.497	3.073	4311.471	59.87	4.282
Comb_OLS	**1033.683**	**24.165**	**2.396**	**2987.408**	**47.695**	**2.693**
Comb_BG	1340.918	29.712	3.662	4786.434	63.029	4.672
Comb_SA	1343.37	29.799	3.709	4818.105	63.227	4.697
ETH						
LSTM	74.317	7.253	8.811	**114.527**	**9.216**	2.99
GRU	62.238	6.376	4.426	125.011	9.693	3.264
Comb_OLS	**59.769**	**6.114**	**4.288**	120.632	9.469	3.115
Comb_BG	62.258	6.478	5.764	117.045	9.313	3.018
Comb_SA	63.24	6.556	6.15	115.978	9.262	**2.987**
LTC						
LSTM	6.586	1.948	3.781	4.845	1.792	3.341
GRU	6.118	2.02	4.749	4.977	1.856	3.81
Comb_OLS	**5.801**	**1.845**	**3.424**	4.728	**1.755**	**3.214**
Comb_BG	5.974	1.914	3.981	4.698	1.758	3.261
Comb_SA	5.99	1.91	3.941	**4.691**	**1.755**	3.244

graphical data confirms that while statistical hybrids like combined OLS show theoretical advantages, advanced neural network approaches (LSTM/GRU) remain highly competitive for LTC forecasting.

5. Conclusions

This study systematically evaluates hybrid models integrating RNNs (LSTM and GRU) with statistical techniques for cryptocurrency price prediction. Key findings highlight the superiority of the combined OLS model for BTC and LTC, attributed to its ability to harmonize linear trends with deep learning's nonlinear pattern recognition. For ETH, LSTM's temporal modeling prowess yields the most accurate forecasts, though hybrid methods remain competitive. The results emphasize three critical insights:

Hybrid Superiority: Combining RNNs with statistical methods (e.g., OLS) enhances robustness, particularly for highly volatile assets like BTC.

Asset-Specific Performance: Model efficacy varies by cryptocurrency, suggesting tailored approaches—LSTM for ETH's complex dynamics, OLS hybrids for BTC/LTC's trends.

Practical Implications: The minimal discrepancies between predicted and actual values (evidenced by low error metrics and visualizations) validate these models' utility for real-world decision-making.

Future research could explore integrating exogenous variables (e.g., sentiment data) or ensemble techniques to further refine accuracy. By bridging statistical and deep learning paradigms, this work contributes a scalable framework for cryptocurrency forecasting, offering both academic and practical value in navigating decentralized financial markets.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

References

Aghashahi M and Bamdad S (2023) Analysis of different artificial neural networks for Bitcoin price prediction. *International Journal of Management Science and Engineering Management* 18(2): 126-133.

Ahsan M. M., Mahmud M. P., Saha P. K., Gupta K. D., and Siddique Z. (2021) Effect of data scaling methods on machine learning algorithms and model performance, *Technologies* 9(3).

Amberkhani A, Bolisetty H, Narasimhaiah R, Jilani G, Baheri B, Muhajab H, Gharinour K, and Shubbar S. (2025) Revolutionizing Cryptocurrency Price Prediction: Advanced Insights from Machine Learning, Deep Learning and Hybrid Models. *In Future of Information and Communication Conference.* Cham: Springer Nature Switzerland, 274-286.

Amirshahi B, and Lahmiri S (2025) Investigating the effectiveness of Twitter sentiment in cryptocurrency close price prediction by using deep learning. *Expert Systems* 42(1).

Bates J. M, and Granger C. W (1969) The combination of forecasts, *Journal of the operational research society* 20(4): 451-468.

Dudek G (2023) Combining Forecasts of Time Series with Complex Seasonality Using LSTM-Based Meta-Learning, *Engineering Proceedings* 39(1).

Ferdiansyah F, Othman S. H, Radzi R. Z. M, Stiawan, D, and Sutikno, T (2023) Hybrid gated recurrent unit bidirectional-long short-term memory model to improve cryptocurrency prediction accuracy. *IAES International Journal of Artificial Intelligence* 12(1).

García-Medina A and Aguayo-Moreno E (2024) LSTM–GARCH hybrid model for the prediction of volatility in cryptocurrency portfolios. *Computational Economics* 63(4): 1511-1542.

Hafidi N, Khoudi Z, Nachaoui M, and Lyaqini S. (2025) Cryptocurrency Price Prediction with Genetic Algorithm-based Hyperparameter Optimization. *Statistics, Optimization & Information Computing*.

Hoa T. T, Le T. M, and Nguyen-Dinh, C. H. (2025) Hybrid model of 1D-CNN and LSTM for forecasting Ethereum closing prices: a case study of temporal analysis. *International Journal of Information Technology*: 1-13. https://doi.org/10.1007/s41870-025-02472-6

Hochreiter S, and Schmidhuber J (1997) Long short-term memory. *Neural computation* 9(8): 1735-1780.

Kiranmai Balijepalli N. S. S, and Thangaraj V (2025) Prediction of cryptocurrency's price using ensemble machine learning algorithms. *European Journal of Management and Business Economics* https://doi.org/10.1108/EJMBE-08-2023-0244

Cho K, Merrienboer B. V , Gulcehre C , Bahdanau D , Bougares F, Schwenk H, and Bengio Y (2014) Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation, https://doi.org/10.48550/arXiv.1406.1078

Oyewola D. O, Dada E. G, and Ndunagu J. N (2022) A novel hybrid walk-forward ensemble optimization for time series cryptocurrency prediction. *Heliyon* 8(11).

Seabe P. L, Moutsinga C. R. B. and Pindza E (2023) Forecasting cryptocurrency prices using LSTM, GRU, and bi-directional LSTM: a deep learning approach, *Fractal and Fractional* 7(2).

Vasilev I (2019) Advanced Deep Learning with Python: Design and implement advanced next-generation AI solutions using TensorFlow and PyTorch, Packt Publishing

Xiang Q (2024) Cryptocurrency assets valuation prediction based on LSTM, neural network, and deep learning hybrid model, *Applied and Computational Engineering* 49: 265-272.

Yang, Shudong and Yu, Xueying and Zhou, Ying (2020) Lstm and gru neural network performance comparison study: Taking yelp review dataset as an example, *2020 International workshop on electronic communication and artificial intelligence (IWECAI)*:98-101.

Zhang Z, Dai H. N, Zhou J, Mondal S. K. García M. M., and Wang H (2021) Forecasting cryptocurrency price using convolutional neural networks with weighted and attentive memory channels. *Expert Systems with Applications* 183: 115378.

AIMS Press