Enhancing Bitcoin price forecasting: a comparative analysis of advanced time series models with hyperparameter optimization

Amine Batsi[1,2], Mohamed Biniz[1,3], Rachid El Ayachi[1,2]
[1]Computer Science Department Laboratory TIAD, Faculty of Sciences and Technics, Sultan Moulay Slimane University, Beni Mellal, Morocco
[2]Faculty of Sciences and Technics, Sultan Moulay Slimane University, Beni Mellal, Morocco
[3]Faculty of Polydisciplinary, Sultan Moulay Slimane University, Beni Mellal, Morocco

Article Info

Article history:

Received May 3, 2025
Revised Dec 28, 2025
Accepted Mar 30, 2026

Keywords:

Bitcoin
Cryptocurrencies
Forecasting
Hyperparameter optimization
Time Series

ABSTRACT

This paper evaluates state-of-the-art time series forecasting to predict next-day Bitcoin prices via distinct architectures and methodologies in a real-time setting. We study six advanced models, KAN, TimesNet, NBEATS, NHITS, PatchTST and BiTCN, applied to a Jan 1, 2023, to Dec 1, 2024. We simulate real world applications via a rolling forecast strategy, in which we predict daily prices from the most recent data. The dataset consists of daily Bitcoin closing prices and data preprocessing and integrity checks for its constituent data. Additionally, rigorous accuracy and reliability were investigated using performance metrics such as the MAE, RMSE, MAPE, and R^2. NBEATS and NHITS were the top performers, achieving an R^2 score of 0.967, explaining complex patterns in volatile cryptocurrency data. The specific importance of model architecture and further hyperparameter optimization in achieving higher forecasting accuracy is highlighted in this study. The practical implications of these findings for the advancement of time series forecasting in financial markets are leveraged here, where timely and accurate forecasts are critical.

Corresponding Author:

Amine Batsi
Computer Science Department Laboratory TIAD, Faculty of Sciences and Technics
Sultan Moulay Slimane University
Beni Mellal, Morocco
Email: batsiamine@gmail.com

1. INTRODUCTION

Artificial intelligence (AI) and its subfields have rapidly evolved, which has helped different domains of finance. These prediction tasks are challenging, but one of the more challenging tasks has been predicting cryptocurrency prices because of the high volatility and complexity of this market. The ability of cryptocurrencies to decentralize identities and susceptibilities to market sentiments offers researchers and investors opportunities to look beyond traditional modelling framework forecasting techniques [1]–[3]. Bitcoin, the biggest of the lot, defines this volatility. For example, December 2024 was the first time that the Bitcoin price hit $100,000 because of the positive environment following the crypto-friendly USA administration. However, that surge was followed by a flash crash, as prices fell close to 7 percent before rebounding. The Bitcoin price is not uncommon in such fluctuations, for instance, regulatory news, macroeconomic events, or even market speculation [4], [5]. For example, the history of Bitcoin exchange-traded fund (ETF) approvals has shown that prices increase as regulatory crackdowns in major markets cause large price declines [6].

However, since these fluctuations are fast and unexpected, traditional forecasting methods, which usually assume linearity and may not capture the complex and nonlinear dynamics of cryptocurrency markets, do not work well under these conditions. As such, there is an increasing desire to advocate advanced time series analysis models that are capable of encompassing the specific properties of cryptocurrencies. Recently, several works have demonstrated the effectiveness of advanced models, such as neural basis expansion analysis for time series (N-BEATS), Kolmogorov-Arnold networks (KAN), neural hierarchical time series (N-HITS) and temporal 2D-variation modelling for general time series analysis (TimesNet), which are capable of capturing the complexity of patterns and dependencies contained in time series data [7]–[9].

This paper investigates the performance of these advanced models in this highly dynamic and unpredictable cryptocurrency market. We compare their performance in predicting price movements and market trends to provide practical insight into their efficacy. This work enhances the literature on machine learning applications for financing, and its findings provide actionable knowledge for investors interested in navigating crypto market dynamics. The remainder of this paper is organized as follows: the next section outlines the methodologies and model architectures employed, followed by the empirical results and analysis. Finally, the conclusion summarizes the key findings and discusses implications for future research and practical applications.

2. LITERATURE REVIEW

2.1. Prior works on cryptocurrency forecasting

Previous work on the cryptocurrency price prediction problem involved statistical models such as ARIMA and exponential smoothing, which laid the foundation for linear dependency factors [10]. However, long short-term memory (LSTM) and gated recurrent units (GRUs) have emerged as widely preferred models to feature long-term trends and temporal dependency [1], [2]. Hybrid models, which combine neural networks and traditional statistical approaches for prediction, have recently improved prediction accuracy by capitalizing on the strengths of both paradigms [11]. Financial time series forecasting is a brand-new era enabled by these hybrid approaches with advanced feature engineering and interpretability.

In addition to modelling techniques, feature engineering and data augmentation are indispensable to time series forecasting. Additionally, adding technical indicators, such as moving averages, the relative strength index (RSI), and Bollinger bands, has increased the richness of the input features, as shown in [12]. Additionally, social media platforms, news articles, and trading forums sentiment analysis have been incorporated and applied to prediction frameworks, providing an understanding of market behaviour, which cannot be processed with numerical data [10]–[15].

The introduction of interpretable deep learning architectures such as N-BEATS and N-HITS represents a real advance in the field. In addition to their state-of-the-art predictive accuracies, these models go a step further to provide the ability for a deeper understanding of the underlying data patterns and hence are fit for all academic research and industrial applications [16]. The KAN has been adapted for time series forecasting, with some variants integrating to emphasize critical time steps and features [17], [18]. The latest entrant, TimesNet, exploits a multiscale structure to uncover temporal dependencies across varying time horizons, surpassing the boundaries of what is possible in forecasting financial time series [19].

2.2. Models overview

KAN is motivated by the Kolmogorov-Arnold representation theorem: any multivariate function can be represented as a sum and composition of univariate functions. KAN models can capture the complicated nonlinear dependencies efficiently and adapt to critical time steps. TimesNet designs a 2D temporal variation modeling mechanism to model time series into image-like features. It can capture both intra-period and inter-period variations in financial data with multiple scales of volatility, achieving excellent performance.

NBEATS is an interpretable architecture with backward and forward residual stacks. It can learn trend and seasonality components in time series separately, which makes NBEATS very powerful for both accuracy and interpretability. NHITS is an extension of NBEATS that learns long-term dependencies through hierarchical interpolation in the backward residual stack. It refines the final prediction recursively at multiple resolutions, which alleviates the error propagation during long-horizon forecasting.

PatchTST is a transformer-based architecture with a patching mechanism; the time series is split into non-overlapping segments, "patches," before feeding into self-attention. This makes the model more efficient and more robust for long sequences. BiTCN utilizes dilated convolutions in both forward and backward directions. Thus, it can model the dependency of multiple horizons efficiently. Due to its rigid nature, the model performs poorly on highly volatile financial data, e.g., Bitcoin.

3. RESEARCH METHOD
3.1. Data preparation
To analyse the stock market, we used historical Bitcoin price data between January 1, 2023, and December 1, 2024, retrieved from the yfinance library to enable effective financial data retrieval Figure 1. The dataset contained daily closing prices and was prepared for modelling via predictive methods after being split into training (70%) and testing (30%) subsets to be used to calibrate the models, with the latter being out-of-sample evaluation. Data preparation was accomplished by resetting the index to retrieve dates explicitly, renaming the columns per modelling standard (ds for the dates, y for the y target values), adding a unique identifier (unique_id) for compatibility with the forecasting frameworks, and performing data integrity checks to discover null values and missing dates to ensure dataset completeness and reliability.

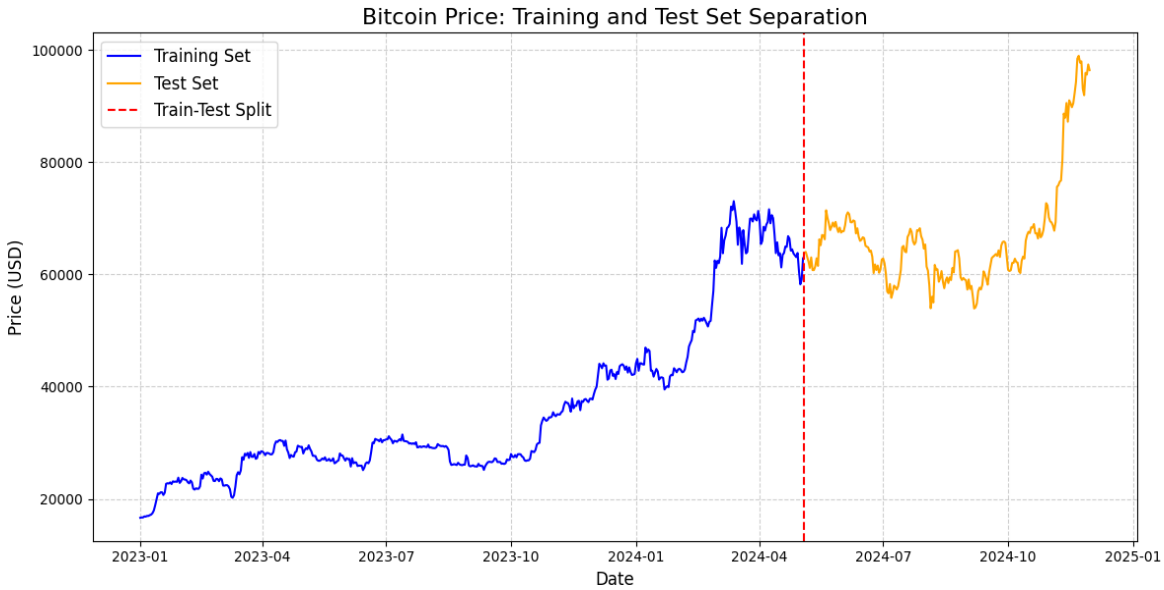

Figure 1. Training and test separation plot

3.2. Models and parameters
This study covers a broad spectrum of advanced methods in time series forecasting, including PatchTST [13], KAN [14], hierarchical models NHITS [15], and interpretable models NBEATS [16], for which the models used in our analysis represent a variety of approaches. To achieve a fair comparison, all the models were set to use the same defaults for the core hyperparameters, such as the horizon (horizon = 1), learning rate (learning rate = 1e-3), and finite number of steps during training (max_steps = 100). The unique characteristics of each model were adjusted to the following parameters: input_size and val_check_steps. By implementing this setup, as shown in Table 1, we were able to analyze the performance of each model in the Bitcoin price forecasting problem on a more comprehensive basis.

The key parameters used for each model are summarized in Table 1. Both NBEATS and NHITS are good choices for forecasting hierarchical or interpretable tasks, whereas KAN is particularly strong with complex dependencies because of the Kolmogorov-Arnold framework. The attention-based PatchTST and transformer models TimesNet possess robust temporal feature extraction capabilities. This configuration offers a balanced and fair comparison, making the strengths and weaknesses of each approach appear clearly while addressing the challenges in financial time series forecasting.

Table 1. Model configurations and hyperparameters

Model	Horizon (h)	Input size	Stack types	Learning rate	Max steps	Validation check steps
KAN [14]	1	24	N/A	1e-3	100	50
TimesNet [17]	1	24	N/A	1e-3	100	50
NBEATS [16]	1	12	['identity']	1e-3	100	10
NHITS [15]	1	12	['identity']	1e-3	100	50
PatchTST [13]	1	12	N/A	1e-3	100	50
BiTCN [18]	1	12	N/A	1e-3	100	50

3.3. Training and forecasting procedure

A rolling forecast strategy, as shown in Figure 2, was applied to the training and forecasting process and adhered to realistic, sequential predictions that are characteristic of real-world applications via the NeuralForcast library implemented within Nixtla [19]. The process was simulated iteratively, simulating a day in the testing period, and included several steps. We had our training and test datasets copied to maintain data integrity in the initial step. Finally, the model was trained on the current training dataset. After fitting, the model made a prediction for the future. Following a Continuous Learning approach, the model was then reinforced with the newly predicted value that was appended to the training dataset. It corresponds to the real-world applications where models need to learn dynamically, from new information, in contrast to using a static data set. The rolling forecast strategy brings in the continuous update mechanism, which enables not only enhancing the performance of prediction but also having the adaptability to the possible changes in data patterns. This process was then iterated across the testing period so that the rolling forecast strategy served as a robust framework to assess the predictive performance of the models in dynamic and evolving scenarios.

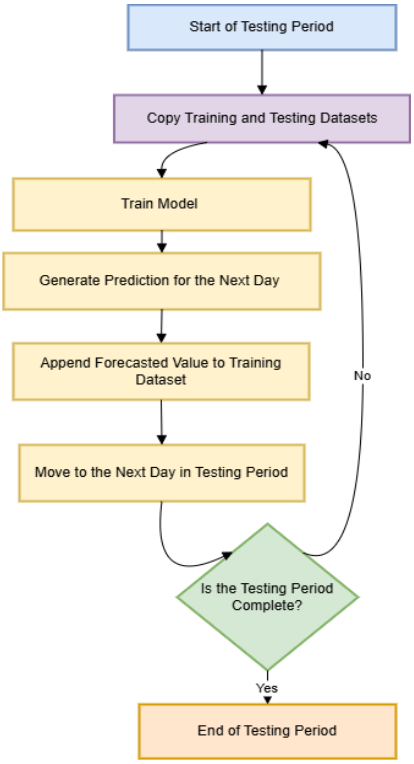

Figure 2. Workflow for model training and forecasting

3.4. Evaluation metrics

To rigorously evaluate the predictive performance of the models, a suite of standard time series evaluation metrics, including the mean absolute error (MAE), root mean squared error (RMSE), mean absolute percentage error (MAPE), and coefficient of determination (R^2 score), were used. Together, these metrics collectively form explicit representations of model accuracy and robustness.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{1}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{2}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3}$$

$$R2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{4}$$

3.5. Hyperparameter optimization

The best performing model, NBEATS, was therefore hyperparameter tuned on in the tree-structured parzen estimator (TPE) optimization algorithm implemented via the Optuna framework [20], [21] to ensure maximal model performance. The sequential model-based optimization (SMBO) technique in the TPE algorithm, iteratively refines hyperparameters to maximize predictive accuracy. To begin the process, a comprehensive search space is defined, as outlined in Table 2.

Table 2. Hyperparameter search space for model optimization

Parameter	Search space
Scaler type	["minmax", "robust", "standard"]
Stack types	[["identity"]]
Activation function	["ReLU", "LeakyReLU"]
Number of learning rate decays	[0 to 3]
Input size	[6, 12, 24]
Random seed	[1 to 20]
Batch size	[32, 64, 128, 256]
Windows batch size	[128, 256, 512, 1024]
Learning rate (η)	Log-uniform distribution between 1e-5 and 1e-2
Max steps	100
Validation check steps	10
Early stop patience steps	2

It leverages the TPE algorithm to construct a probabilistic model $p(\theta|y)$ that represents the likelihood of hyperparameters θ given the metrics y (validation accuracy or loss). It uses a combination of two density functions: $p(\theta|y<y*)$ (modelling the distribution of hyperparameters that results in better-than average performance) and $p(\theta|y\geq y*)$ (modelling the distribution of hyperparameters that lead to worse-than average performance). The algorithm then selects the next set of hyperparameters to evaluate by maximizing the expected improvement (EI):

$$EI(\theta) = \int_{-\infty}^{y^*} (y^* - y)p(y|\theta)dy \quad (5)$$

y∗, a threshold performance value (here, best validation accuracy so far) Once this convergence or stopping criterion is achieved, the process is continued iteratively. From this optimization process, we chose our best performing model to guarantee that the final model provides the highest predictive accuracy on the validation set. First, it not only improves model performance but also helps avoid overcutting from the hyperparameter space systematically.

4. RESULTS AND DISCUSSION

We assessed the performance of six advanced time series forecasting models KAN, TimesNet, NBEATS, NHITS, PatchTST, and BiTCN using the metrics MAE, RMSE, MAPE, and R^2 and the training and testing duration for the initial iteration to measure computational complexity. With these metrics, we evaluate each model in terms of its accuracy and explanatory power relative to Bitcoin price data. Table 3 summarizes the results.

Table 3. Model performance metrics

Model	MAE	RMSE	MAPE	R^2	Training time	Testing time
KAN	1802.15	2440.90	2.64%	0.938	12.35 secs	0.15 secs
TimesNet	1367.81	1920.33	2.04%	0.961	24.05 secs	0.24 secs
NBEATS	1262.98	1776.72	1.89%	0.967	1.3 secs	0.14 secs
NHITS	1263.81	1777.03	1.89%	0.967	2.81 secs	0.18 secs
PatchTST	1354.68	1873.25	2.03%	0.963	4.6 secs	0.17 secs
BiTCN	35439.52	36014.39	53.01%	-12.443	2.98 secs	0.2 secs

Both the MAE and R^2 values of the NBEATS and NHITS models were recorded as 1263 and 0.967, respectively with the lowest computational time. This suggests that the models can account for nearly 96.7% of the variance in the Bitcoin price data, which validates the models suitability for forecasts. It is possible that their appropriate hierarchical and interpretable architectures make them suitable for learning long-term dependencies and complex patterns in financial time series.

TimesNet and PatchTST performed similarly, with MAE values of 1367.81 for TimesNet and 1354.68 for PatchTST and higher R^2 values. However, in terms of computational complexity, TimesNet got a much greater training time 24.05 seconds, and testing time 0.24 seconds, than PatchTST 4.6 seconds for training and 0.17 seconds for testing, which was more computationally efficient, but also achieved very similar predictive performance. Although their accuracy was notable, they were just slightly bested by NBEATS and NHITS, indicating that NBEATS architectural features may have an edge in modeling the customized minutiae of Bitcoin price movement. Alternatively, even though KAN achieved a good MAE of 1802.15 and an R^2 of 0.938, it appeared to undershoot the other models in predicting Bitcoin price dynamics. From a computational perspective, KAN showed a moderate training time of 12.35 seconds and a testing time of 0.15 seconds, which makes it between TimesNet and PatchTST but with less accuracy.

In contrast, the performance of the BiTCN model was dramatically poor, with an MAE of 35439.52 and a negative R^2 of -12.443. These results demonstrate that in the case of the Bitcoin price data, the BiTCN provides inaccurate predictions that are less reliable than those of a simple baseline model, indicating that the BiTCN is insufficient to predict very volatile and complex Bitcoin price data despite its relatively low training time 2.98 seconds, and testing time 0.2 seconds. We run hyperparameter tuning on the NBEATS model via the TPE optimization algorithm using Optuna. The optimized hyperparameters are provided in detail in Table 4.

Table 4. Optimized hyperparameters for NBEATS

Hyperparameter	Value
Scaler type	Robust
Stack types	['identity']
Activation function	LeakyReLU
Number of learning rate decays	2
Input size	6
Random seed	13
Batch size	64
Windows batch size	1024
Learning rate (η)	0.000214

These hyperparameters were used, and the model, NBEATS_HP, achieved an MAE of 1262.12 and R^2 of 0.967, as shown in Table 5. Systematic hyperparameter tuning results in a marginal performance improvement over the base NBEATS model, which is accurate, validating the role of the hyperparameters in refining the model performance.

Table 5. Performance metrics of NBEATS_HP

Model	MAE	RMSE	MAPE	R^2
NBEATS_HP	1262.12	1776.40	1.88%	0.967

Additionally, the visual analysis of the plots in Figure 3, which overlays the actual Bitcoin closing prices with the predictions from each of the six models, provides deeper insights into their relative performance beyond numerical metrics.

The actual data fit the KAN model fairly well but display pronounced deviations, especially when sharp price changes occur. The model's limitations in adapting to fast fluctuations are emphasized by these discrepancies, confirmed by its R2 value of 0.938. In calmer periods, KAN is adequate, but in Bitcoin prices, dynamic patterns, KAN cannot track.

The actual prices are closely followed by the results of the TimesNet model, which demonstrates significantly better predictive capability. While periods with high volatility show minor deviations, its prediction is consistent with the data. This strong fit demonstrates its strong performance metrics, with an R^2 value of 0.961, and shows that it can handle volatile time series data.

The NBEATS and NHITS models have almost perfect ordering of the actual data and are extremely accurate. The prediction lines fit the actual price trajectory closely and with very little error, even given the most volatile patterns. Our results show that these outstanding parameters of 0.967 R^2 and low MAE are the most reliable models from the group. The reason for the success of these models is that they have hierarchical, interpretable architectures that are suitable for capturing long-range dependencies and complex patterns in financial time series.

The PatchTST model can also make predictions very close to the actual data. However, the deviations are somewhat larger than those of NBEATS and NHITS, especially in volatile phases. This R^2 value of 0.963 is slightly lower than the first value since these larger discrepancies are reflected. However, this does not hinder PatchTST from continuing to be a robust approach to forecasting, as its transformer-based architecture is able to capture temporal dependencies.

The performance of the BiTCN model is in stark contrast. Large deviations from actual data can be seen in the plot of the predictions: predictions not only do not capture the trend but also do not capture the variability in Bitcoin prices. Correspondingly, these deficiencies produce a very low R2 value (-12.443), meaning that the model's predictions are not accurate relative to a naive baseline. This poor performance indicates that the BiTCN architecture is not well suited for financial time series forecasting, especially in markets such as Bitcoin, where the market is extremely volatile.

Finally, with hyperparameter tuning for TPE on the NBEATS_HP model, it is still a closer fit to the actual data than the base NBEATS model is. The hyperparameter optimization is proven to be effective since the predicted and actual lines mate nearly completely. The model is made even more precise with this refined configuration, and the R^2 value does indeed improve slightly to 0.967, and the MAE marginally decreases. We find that the success of NBEATS_HP highlights the benefit of a systematic optimization strategy in refining high-performing models.

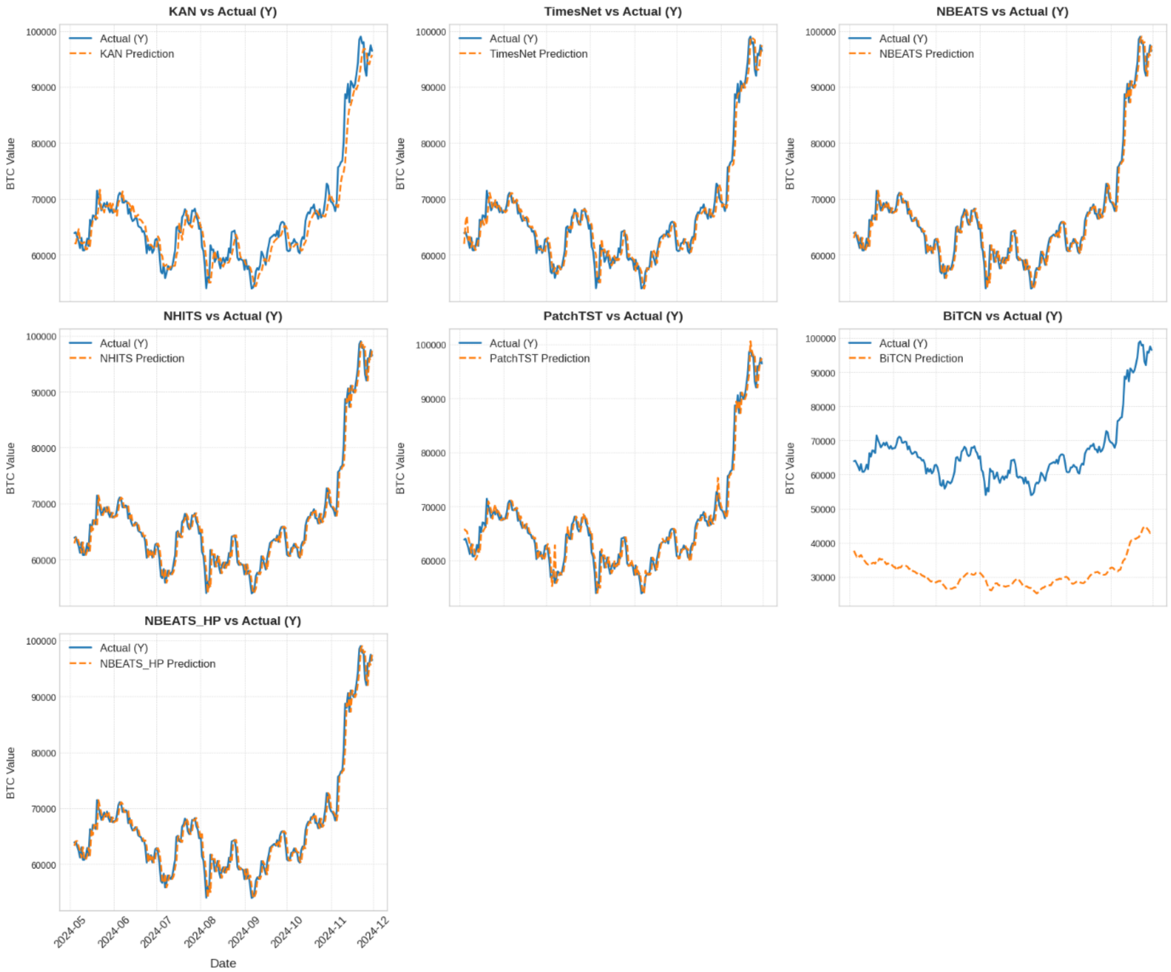

Figure 3. Comparison of actual Bitcoin prices vs predicted values across different models

As such, these visual insights reinforce the quantitative evaluation, the accuracy of NBEATS and NHITS, the role of hyperparameter tuning for NBEATS_HP, and the difficulties facing models such as the BiTCN in dealing with volatile financial data. The analysis leads to key observations that guide which factors are important in model performance. Compared with other methods, models with hierarchical and interpretable structures (such as NBEATS and NHITS) consistently demonstrate superior performance, enabling effective long-term dependency and complex pattern modelling in financial data, while maintaining low computational cost, making them highly suitable for forecasting tasks. Thus, these models are highly suitable for forecasting tasks. Moreover, the marginal improvement attained by the NBEATS_HP model further suggests that hyperparameter optimization is important even for reliable architectures and can lead to improvements in techniques such as TPE optimization. In contrast, the failure of BiTCN to perform well apart from essentially random contributions to out-of-sample returns highlights the difficulty of forecasting the most volatile data of all, Bitcoin prices, and the need to choose strong models and configurations that can contend with the intricacies of financial data.

One of the findings of this study underlines the importance of model selection and configuration in financial time series forecasting. All high-performing models provided strong prediction effects; however, this depended on the architectural design and optimization techniques employed in each case. Hierarchical and interpretable models such as NBEATS and NHITS performed strongly, suggesting that those approaches are good for financial forecasting where accuracy and explanation are both important.

Recent research on cryptocurrency forecasting supports and extends our findings. This study [22] utilized a CNN-LSTM model with Variational Autoencoder (VAE) feature extraction and SHAP interpretability, achieving extremely high accuracy ($R2\approx0.99$) on Bitcoin data. These results support our finding that more complex architectures, such as TimesNet, NBEATS, and NHITS are preferable. Similarly, CryptoMamba [23] is a state-space model that captures regime shifts and long-range dependencies. They found that this increased robustness to changes in volatility greatly improved their forecasts, which supports our use of a rolling forecast to simulate real-world use.

Finally, other works emphasize the importance of going beyond univariate models. This paper [24] found that adding exogenous macroeconomic variables, including the VIX, gold, and oil prices, improved Bitcoin prediction accuracy, suggesting a direction for improving our univariate model. This investigation [25] found that SARIMA, LSTM, and Prophet performed differently across pre-, during-, and post-COVID periods, with LSTMs consistently outperforming traditional statistical methods but requiring careful tuning in volatile regimes. Taken together, these results support our conclusion that well-tuned architectures are better suited for next-day Bitcoin prediction, but suggest a number of directions for future improvement, including interpretability, incorporation of global factors, and adaptive modeling strategies for regime shifts.

However, there is a limitation that should be mentioned. The models focus only on univariate bitcoin time series data; on the other hand, integrating other variables such as trading volume, social sentiment, geopolitical indicators, or macroeconomic indicators could enhance accuracy.

The results of the analysis indicate that interpretable and hierarchical architectures (NBEATS and NHITS) consistently deliver strong performance with low errors and high explainability. This suggests that they may prove useful in trading strategies and risk management scenarios where reliable short-term predictions are needed. Practically, financial analysts, portfolio managers, and individual investors may find value in using interpretable and hierarchical architectures for next-day Bitcoin price prediction to improve their ability to time the market and mitigate volatility risk.

From a methodological perspective, this paper demonstrates that a rolling forecast strategy is used, which is more reflective of a real-world trading environment than static train-test splits. Furthermore, the benefits of hyperparameter optimisation highlight the importance of careful consideration of empirical setup choices. Overall, the results of this paper suggest promising directions forward for future research, including extending the methodology to include exogenous variables, such as trading volume, macroeconomic or sentiment information; extending the interpretable technique to improve model explainability; and/or considering ensemble or hybrid approaches to further increase model robustness in cryptocurrency markets.

5. CONCLUSION

The results of this study emphasize that model and configuration selections for time series forecasting should be overridden, especially when highly volatile financial data such as Bitcoin prices are considered. The performance of these models (NBEATS and NHITS) was better (more hierarchical, interpretable), better capturing long-term dependencies as well as complex patterns with high accuracy and reliability. Furthermore, the results also show the benefit of hyperparameter optimization, with the marginal improvements observed over the optimized NBEATS_HP model, demonstrating further potential to refine already strong architectures. On the other hand, the performance of the BiTCN model is poor, featuring volatility and highlighting the pain that volatility brings to forecasting models and revealing that the architecture for financial data is suitable. Future research should focus on ensemble modeling approaches, where different architecture combinations can result in superior forecasting accuracy. Such advances could be used to build stronger, simpler, and more accountable tools for the navigation of the peculiarity of financial markets.

FUNDING INFORMATION

The authors state no funding is involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Amine Batsi	✓	✓	✓	✓	✓	✓		✓	✓		✓			✓
Mohamed Biniz	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	
Rachid El Ayachi	✓			✓			✓			✓	✓		✓	✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors state no conflict of interest.

DATA AVAILABILITY

The findings of this study are supported by the data that were obtained from publicly available sources using the yfinance Python library. The historical Bitcoin price data is available in the dataset through Yahoo Finance (https://finance.yahoo.com/) or directly using the yfinance package.

REFERENCES

[1] J. Katina, I. Katin, and V. Komarova, "Cryptocurrency price forecasting: a comparative analysis of autoregressive and recurrent neural network models," *Entrepreneurship and Sustainability Issues*, vol. 11, no. 4, pp. 425–436, Jun. 2024, doi: 10.9770/jesi.2024.11.4(26).

[2] T. Mahmud, T. Akter, R. Chakma, T. Barua, M. S. Hossain, and K. Andersson, "Predictive modeling of cryptocurrency prices using machine learning algorithms," in *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, IEEE, Jun. 2024, pp. 1–6. doi: 10.1109/ICCCNT61001.2024.10723951.

[3] A. Sharma, K. Gulia, N. Jindal, I. N. Tripathi, and G. Mishra, "Time-series forecasting and anomaly detection in Ethereum: a deep learning approach," *Journal of Multi Disciplinary Engineering Technologies Volume 18, Issue 01*, vol. 18, no. 1, pp. 77–84, 2024.

[4] T. Luc Minh, R. Senkerik, and T. K. Dang, "Predicting Bitcoin's price: a critical review of forecasting models and methods," in *Future Data and Security Engineering. Big Data, Security and Privacy, Smart City and Industry 4.0 Applications*, T. K. Dang, J. Küng, and T. M. Chung, Eds., Springer, Singapore, 2024, pp. 36–50. doi: 10.1007/978-981-96-0434-0_3.

[5] G. Vonitsanos, A. Kanavos, F. Grivokostopoulou, and S. Sioutas, "Optimized price prediction of cryptocurrencies using deep learning on high-volume time series data," in *2024 15th International Conference on Information, Intelligence, Systems & Applications (IISA)*, IEEE, Jul. 2024, pp. 1–8. doi: 10.1109/IISA62523.2024.10786656.

[6] A. Farooq, M. Irfan Uddin, M. Adnan, A. A. Alarood, E. Alsolami, and S. Habibullah, "Interpretable multi-horizon time series forecasting of cryptocurrencies by leverage temporal fusion transformer," *Heliyon*, vol. 10, no. 22, p. e40142, Nov. 2024, doi: 10.1016/j.heliyon.2024.e40142.

[7] D. Bhowmick, D. Barik, and D. Ghosh, "Forecasting Bitcoin prices using LSTM: a comparative evaluation with future prediction," in *Innovations in Data Analytics*, A. Bhattacharya, P. Dutta, S., Dutta, and D. Samanta, Eds., Springer, Singapore, 2024, pp. 425–449. doi: 10.1007/978-981-97-4928-7_33.

[8] A. D. Hassan and M. M. Ibrahim, "Neural foundations of crypto predictions: exploring deep and transformer-based models," 2024. doi: 10.2139/ssrn.5051299.

[9] M. F. Bin Hossain, L. Z. Lamia, M. M. Rahman, and M. M. Khan, "FinBERT-BiLSTM: a deep learning model for predicting volatile cryptocurrency market prices using market sentiment dynamics," *Applied Intelligence*, vol. 56, no. 3, p. 89, Feb. 2026, doi: 10.1007/s10489-026-07086-4.

[10] J. N. Pinky and R. Akula, "Enhancing cryptocurrency market forecasting: advanced machine learning techniques and industrial engineering contributions," in *Advances in Artificial Intelligence Applications in Industrial and Systems Engineering*, W. Karwowski, V. Duffy, and G. Salvendy, Eds., Wiley, 2025, ch. 9, pp. 197–262. doi: 10.1002/9781394257096.ch09.

[11] A. Yusuf Adhami, M. Parvej, N. A. Khan, and M. Khalid, "Enhancing cryptocurrency prediction: a fusion of machine learning and neutrosophic programming," *Neutrosophic Sets and Systems*, vol. 91, no. 1, 2025.

[12] M. S. Islam *et al.*, "Machine learning-based cryptocurrency prediction: enhancing market forecasting with advanced predictive models," *Journal of Ecohumanism*, vol. 4, no. 2, Mar. 2025, doi: 10.62754/joe.v4i2.6663.

[13] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: long-term forecasting with transformers," *arXiv preprint arXiv:2211.14730*, 2023.

[14] K. Xu, L. Chen, and S. Wang, "Kolmogorov-Arnold networks for time series: bridging predictive power and interpretability," *arXiv preprint arXiv:2406.02496*, 2024.
[15] C. Challu, K. G. Olivares, B. N. Oreshkin, F. Garza Ramirez, M. Mergenthaler Canseco, and A. Dubrawski, "NHITS: neural hierarchical interpolation for time series forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Jun. 2023, pp. 6989–6997. doi: 10.1609/aaai.v37i6.25854.
[16] B. N. Oreshkin, D. Carpov, N. Chapados, and Y. Bengio, "N-Beats: neural basis expansion analysis for interpretable time series forecasting," in *8th International Conference on Learning Representations, ICLR 2020*, 2020.
[17] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "Timesnet: temporal 2D-variation modeling for general time series analysis," in *11th International Conference on Learning Representations*, 2023.
[18] O. Sprangers, S. Schelter, and M. de Rijke, "Parameter-efficient deep probabilistic forecasting," *International Journal of Forecasting*, vol. 39, no. 1, pp. 332–345, Jan. 2023, doi: 10.1016/j.ijforecast.2021.11.011.
[19] K. G. Olivares, C. Challú, F. Garza, M. M. Canseco, and A. Dubrawski, "NeuralForecast: User friendly state-of-the-art neural forecasting models," 2022, *PyCon, Salt Lake City, Utah, US*.
[20] S. Watanabe, "Tree-structured Parzen estimator: understanding its algorithm components and their roles for better empirical performance," *arXiv preprint arCiv:2304.11127*, 2025.
[21] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: a next-generation hyperparameter optimization framework," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, New York, NY, USA: ACM, Jul. 2019, pp. 2623–2631. doi: 10.1145/3292500.3330701.
[22] W. Badar, S. Ramzan, A. Raza, N. L. Fitriyani, M. Syafrudin, and S. W. Lee, "Enhanced interpretable forecasting of cryptocurrency prices using autoencoder features and a hybrid CNN-LSTM model," *Mathematics*, vol. 13, no. 12, p. 1908, Jun. 2025, doi: 10.3390/math13121908.
[23] M. S. Sepehri, A. Mehradfar, M. Soltanolkotabi, and S. Avestimehr, "CryptoMamba: leveraging state space models for accurate Bitcoin price prediction," in *2025 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, IEEE, Jun. 2025, pp. 1–3. doi: 10.1109/ICBC64466.2025.11114565.
[24] N. Köse, Y. E. Gür, and E. Ünal, "Deep learning and machine learning insights into the global economic drivers of the bitcoin price," *Journal of Forecasting*, vol. 44, no. 5, pp. 1666–1698, Aug. 2025, doi: 10.1002/for.3258.
[25] S. K. Panigrahi, "Predicting cryptocurrency prices comparing different machine learning techniques: a performance analysis for pre and post COVID-19 pandemic periods," *International Journal of Economics and Financial Issues*, vol. 15, no. 3, pp. 124–138, 2025, doi: 10.32479/ijefi.17794.

BIOGRAPHIES OF AUTHORS

Amine Batsi is a software engineer with a Master's degree in Systems Engineering from Sultane Moulay Slimane University. He is currently pursuing a Ph.D. at the Sultane Moulay Slimane University, focusing on AI and data science applications in finance. His research interests include the analysis of time series for the prediction and management of cryptocurrencies, aiming to predict cryptocurrency trends and explore the factors influencing their financial movements. He can be contacted at email: batsiamine@gmail.com.

Mohamed Biniz obtained his master's degree in business intelligence in 2014 and completed his Ph.D. in computer science in 2018 at the Faculty of Science and Technology, University Sultan Moulay Sliman Beni-Mellal. Currently, he serves as a professor of computer science at the Polydisciplinary Faculty of the University Sultan Moulay Slimane Beni Mellal in Morocco. His research focuses on the field of semantic web engineering and deep learning, with a specific interest in studying the evolution of ontology, big data, natural language processing, machine learning, and dynamic programming. He can be contacted at email: mohamedbiniz@gmail.com.

Rachid El Ayachi is a professor of higher education at the Faculty of Sciences and Techniques of Beni Mellal (Computer Science Department) since 2013, he obtained a Master's degree in Computer Science, Telecom and Multimedia (ITM) in 2006 at the Faculty of Sciences of Rabat (Mohammed V University) and Ph.D. degree in computer science at the Faculty of Sciences and Techniques of Beni Mellal (Sultan Moulay Slimane University) in 2012. Currently, he is a member of the information processing and decision support laboratory (TIAD). His research focuses on image processing, pattern recognition, machine learning, and natural language processing (NLP). He can be contacted at email: rachid.elayachi@usms.ma.