Int. j. inf. tecnol.
https://doi.org/10.1007/s41870-025-02472-6

ORIGINAL RESEARCH

Hybrid model of 1D-CNN and LSTM for forecasting Ethereum closing prices: a case study of temporal analysis

Tran Thai Hoa[1] · Thanh Manh Le[2] · Cuong H. Nguyen-Dinh[3]

Received: 10 November 2024 / Accepted: 10 February 2025

Abstract The volatility of cryptocurrency markets has sparked significant interest in developing predictive models capable of accurately forecasting price movements. Addressing the complexities posed by the non-linear and dynamic nature of cryptocurrency price data, this study introduces a hybrid model that combines one-dimensional convolutional neural networks (1D-CNN) and long short-term memory (LSTM) networks to forecast Ethereum's closing prices. The 1D-CNN component captures localized patterns within the time-series data, while the LSTM component effectively models long-term dependencies and sequential trends. A sliding window technique is applied to preprocess Ethereum trading data, enabling the model to manage temporal structures and enhance predictive accuracy. Experiments were performed to assess the hybrid model's performance under various configurations, benchmarking it against standalone 1D-CNN and LSTM models. Experimental results demonstrated that the hybrid model significantly outperforms these baseline models, promising its potential for improving forecasting accuracy in cryptocurrency markets.

Keywords Cryptocurrency · Ethereum · Forecasting · 1D-CNN · LSTM · Deep learning

✉ Cuong H. Nguyen-Dinh
cuong.nguyen@pxu.edu.vn

Tran Thai Hoa
tranthaihoa@hueuni.edu.vn

Thanh Manh Le
lmthanh@hueuni.edu.vn

[1] University of Econimics, Hue University, Hue, Viet Nam

[2] Hue University, Hue, Viet Nam

[3] Phu Xuan University, Hue, Viet Nam

1 Introduction

The rapid rise of cryptocurrencies, particularly Ethereum, has attracted widespread attention from investors, researchers, and financial analysts [1]. Unlike traditional financial markets, cryptocurrency markets exhibit unique characteristics, including high volatility, non-stationary trends, and intricate temporal dependencies, making them challenging to predict accurately [2, 3]. In this context, effective forecasting models play a pivotal role in aiding investors and decision-makers to navigate the uncertain landscape of cryptocurrency trading. Deep learning advancements drive novel predictive methods in volatile markets [1], as seen in healthcare [4], traffic [5], and e-commerce [6] applications. This study examines the potential of a hybrid model, combining 1D-CNN and LSTM networks, to forecast the closing prices of Ethereum.

The hybrid model leverages the strengths of both LSTM and 1D-CNN architectures to address the limitations of traditional statistical and machine learning methods in forecasting cryptocurrency prices. 1D-CNN is particularly advantageous in extracting meaningful patterns from local sequences, allowing the model to identify significant features within smaller time frames. When paired with LSTM network, which excels at capturing long-term dependencies in time-series data, the hybrid model gains a powerful capability to address temporal dependencies in cryptocurrency price data. This hybrid approach is anticipated to improve model robustness, enabling a more comprehensive analysis of cryptocurrency price fluctuations.

The contributions of this paper are threefold. First, it introduces a novel hybrid model architecture that combines the strengths of 1D-CNN and LSTM networks to enhance predictive accuracy. Second, the model is applied to temporal data processed through a sliding window technique, capturing sequential patterns and dependencies in cryptocurrency trading data. Third, experimental results demonstrate that the hybrid model significantly outperformed other baseline models, including standalone 1D-CNN and LSTM architectures, in most experiments conducted on Ethereum price data. This study provides a robust framework for improving forecasting precision in the cryptocurrency market, with potential applications extending to other volatile financial assets.

The rest of this paper is organized as follows: Sect. 2 reviews current state-of-the-art research on deep learning approaches for predicting cryptocurrency trading movements. Section 3 describes the architecture of the proposed hybrid model, and Sect. 4 presents the experimental results. Finally, Sect. 5 summarizes the key contributions and outlines potential directions for future work.

2 Related work

The dynamic and volatile nature of financial markets has driven the adoption of various computational techniques to enhance the accuracy of price prediction. Traditional statistical methods [1] have long been employed for modeling and forecasting; however, their limitations in capturing complex, nonlinear patterns have shifted attention towards machine learning and deep learning approaches. These advanced methodologies, leveraging their ability to model intricate dependencies and relationships in large datasets, have shown significant promise in financial market prediction [1, 7]. Within the scope of this section, we focus on highlighting the state-of-the-art applications of CNN models, LSTM networks, and hybrid models that combine these architectures. These approaches have been extensively explored in the financial domain in general and in cryptocurrency price forecasting in particular. For a comprehensive review of deep learning applications in this field, readers are referred to these valuable surveys [1, 7–9] that provide detailed insights and broader perspectives.

CNN networks have proven to be powerful tools for predicting financial market trends, including stock prices and cryptocurrency prices, due to their ability to capture spatial dependencies and patterns in sequential data. Cavalli and Amoretti [10] introduced a novel approach for predicting Bitcoin trends using 1D-CNN network. The authors developed a methodology for constructing datasets that integrate social media data, blockchain transaction history, and financial indicators, utilizing a cloud-based distributed architecture to collect and analyze extensive datasets. Their results demonstrated that the 1D-CNN model outperformed LSTM models in accuracy, attributed to its compact and computationally efficient architecture. Additionally, the study proposed a trading strategy based on the 1D-CNN predictions, showing profitability in bullish trends and minimized losses in bearish markets. Our proposed approach differs from this study in the architecture of the deep learning model and the use of a dataset that does not incorporate external sources.

Further demonstrating the utility of CNN in financial forecasting, Alonso-Monsalve et al. [11] evaluated the performance of CNN-based architectures for trend classification in high-frequency cryptocurrency exchange rates. The study highlighted the ability of CNNs to efficiently capture spatial patterns in technical indicators, making them a promising alternative to traditional multilayer perceptrons. Four architectures—CNN, hybrid CNN-LSTM, multilayer perceptron (MLP), and radial basis function (RBF) neural networks—were compared using 18 technical indicators derived from one-minute resolution data over a one-year period. Results indicated that CNNs performed well, particularly for Bitcoin, Ether, and Litecoin, demonstrating their effectiveness in cryptocurrency prediction. Zhang [12] addressed the challenges of forecasting cryptocurrency prices, which are influenced by non-stationary behavior and stochastic market effects. To improve prediction accuracy, the authors proposed the Weighted & Attentive Memory Channels model, leveraging deep learning to capture interdependencies among cryptocurrencies and extract temporal features. The model integrates three modules: an Attentive Memory module combining GRUs with self-attention, a Channel-wise Weighting module to recalibrate interdependencies among cryptocurrency prices, and a Convolution and Pooling module for local feature extraction. Experimental results demonstrated that the model achieved state-of-the-art performance, surpassing baseline models in accuracy, prediction error, and profitability. In contrast to these studies, our research focuses on a specific model within the CNN family, known as 1D-CNN, and a hybrid model combining 1D-CNN and LSTM.

Besides the effectiveness of CNN models in capturing spatial patterns for cryptocurrency price prediction, LSTM networks play a crucial role in modeling temporal dependencies and sequential patterns inherent in financial time series data. For example, Kadhim et al. [13] introduced a multi-modal hybrid model based on deep learning to predict Bitcoin values by integrating sentiment analysis from social media with on-chain and market data. The analysis utilized tools such as Twitter-RoBERTa and VADER on datasets collected from 2014 to 2022, achieving notable accuracy with low error rates (MAPE of 4.37% and RMSE of 6.55%). By combining LSTM neural networks with sentiment and financial data, the study underscored the importance of social media sentiment in forecasting market trends, providing valuable guidance for economic prediction and decision-making. Similarly, B. A. Pai et al. [14] addressed the challenges of cryptocurrency volatility and complexity by proposing a deep learning-based model to forecast cryptocurrency prices as a time series. The authors utilized LSTM networks to overcome the limitations of traditional neural networks in handling long sequences of data, focusing on predicting the open prices of cryptocurrencies such as Bitcoin, Ethereum, Litecoin, and Bitcoin Cash. The approach demonstrated accurate predictions with minimal errors and showcased scalability and automation capabilities through deployment via APIs and web applications.

In another study, Kim et al. [15] proposed a novel framework for predicting Bitcoin prices. The framework utilized change point detection to segment time-series data, enabling effective normalization and incorporating on-chain data as input features for prediction. A self-attention-based multiple LSTM (SAM-LSTM) model was developed, combining multiple LSTM modules for different on-chain variable groups with an attention mechanism. Experimental results demonstrated the framework's effectiveness, achieving promising accuracy metrics, including MAE of 0.3462 and RMSE of 0.5035. In general, while these studies used only LSTM networks, our proposed approach employs a hybrid model combining 1D-CNN and LSTM.

Specifically, variations of artificial neural networks (ANN) were also applied to predict cryptocurrency prices in the study by S. Behera et al. [16]. The authors applied the geometric mean optimization (GMO) algorithm to adjust the hidden layer input weights and biases of six ANN variants, creating hybrid models for predicting the closing prices of four major cryptocurrencies. For comparison, traditional gradient descent was also used to train the same ANN variants, resulting in alternate models. The models were evaluated using MSE, RMSE, and MAPE metrics, with results showing that the GMO-based PSNN outperformed the others. Additionally, the performance of the GMO-based PSNN was compared to existing hybrid models, further demonstrating its effectiveness. In a similar effort, R.M Aziz et al. [17] tried to improve the performance of machine learning models by using the Light Gradient Boosting Machine (LGBM) approach. By comparing various machine learning models, including Random Forest and Multi-Layer Perceptron, LGBM demonstrated superior accuracy (98.60%), which was further improved to 99.03% through hyper-parameter tuning, proving its effectiveness for Ethereum datasets with limited attributes. The most significant differences between these studies and our proposed research lie in the use of traditional machine learning model structures and the hybrid architecture of deep learning models.

The research trend of using a single model for price prediction is not limited to traditional machine learning models, CNNs, or LSTMs but also includes the introduction of novel deep learning models. For example, Gajjar et al. [18] presented a novel deep-learning pipeline to predict NSE stock prices for companies like Adani Ports, Reliance, and Tata Steel. They introduced Liquid Time-Constant Networks (LTCs) as a core component of an enhanced computing paradigm, evaluating their performance against traditional architectures such as RNNs, GRUs, LSTMs, and BiLSTMs. The research highlighted the potential of LTCs as a superior alternative to LSTMs for stock market trend prediction, offering an effective balance between computational efficiency and prediction accuracy. While this study focused on the stock market, its potential application to the cryptocurrency market is promising.

Building on the strengths of individual deep learning models, the emergence of hybrid architectures, particularly the combination of CNN and LSTM, has gained significant attention for improving the accuracy and robustness of cryptocurrency price prediction. For example, Peng et al. [19] introduced an attention-based CNN-LSTM model (ACLMC) capable of multi-currency predictions by leveraging correlations across frequencies and currencies. Experiments demonstrated that this approach outperformed traditional baselines, achieving better financial metrics and reducing the number of transactions, thereby mitigating investment risks. Similarly, Zhong et al. [20] also explored the combination of relationwise graph attention network (ReGAT) with LSTM network. Experimental results with real-world data demonstrated that LSTM-ReGAT achieved superior predictive performance and profitability, offering valuable insights for investment decision-making in the cryptocurrency market.

In another study, García-Medina and Aguayo-Moreno [21] analyzed cryptocurrency volatility using GARCH models, MLP, LSTM, and hybrid LSTM-GARCH models, incorporating GARCH parameters as features for LSTM. Covering the period around the March 2020 pandemic declaration, results showed that deep learning models, particularly MLP, outperformed GARCH models in terms of heteroscedastic, absolute, and squared errors. While MLP achieved the best predictive results with lower computational cost, it performed similarly to LSTM and LSTM-GARCH under statistical tests. Our proposed approach differs from these hybrid models in terms of the use of 1D-CNN instead of graph attention networks.

Khattak et al. [22] explored the use of Fibonacci technical indicators (TI) and multi-class classification based on trend and price strength (trend-strength) to enhance the accuracy and profitability of AI models, particularly hybrid CNN-LSTM architectures. The research introduced a six-stage predictive system, including data collection, preprocessing, and evaluation, and demonstrated that incorporating Fibonacci TI improved model performance in 44% of configurations and profitability in 68%. Empirical results showed that hybrid CNNs, especially C-LSTM models, performed best for trend-strength predictions in 4-class and 6-class settings, achieving significant gains in return on investment (ROI) and showcasing their potential for accurate and profitable cryptocurrency trading. The key difference in model architecture between this study and our research lies in the incorporation of 1D-CNN as a core component in our proposed hybrid model.

In cryptocurrency price prediction, models are often trained not only on financial data but also on sentiment analysis of textual information, such as social media posts and news, to capture market sentiment and its influence on price movements. For example, Rateb et al. [23] proposed a method for predicting cryptocurrency prices and trends by integrating sentiment analysis with time series forecasting. Using over one million tweets about Bitcoin, Ethereum, and Binance Coin during the Russian–Ukrainian War, the authors compared three models: CNN-LSTM, SVM with GloVe and TF-IDF features, and Pysentimento for sentiment classification, with Pysentimento achieving the highest accuracy. Combining sentiment analysis results, Google Trends, and cryptocurrency market data, the SARIMA model was applied for price prediction, providing valuable insights into investor preferences and market behavior amid geopolitical uncertainties. Similarly, Huang et al. [24] focused on predicting cryptocurrency price fluctuations by analyzing sentiment from social media, specifically Chinese posts on Sina-Weibo, a popular Chinese platform. The researchers developed a crypto-specific sentiment dictionary and an LSTM-based recurrent neural network, incorporating historical price data to forecast future trends. Experimental results showed that the proposed approach outperformed traditional autoregressive models. In short, our method does not utilize textual information to train the model like these studies.

3 Hybrid model design for Ethereum price forecasting

This section first introduces the architecture of the proposed hybrid model, which combines 1D-CNN and LSTM for advanced forecasting of cryptocurrency price movements. Next, the sliding window technique, applied to generate window-based time series data for the hybrid model, is discussed. Then, each component of the hybrid model is elaborated in detail. Finally, the algorithm that utilize the proposed model is presented.

Figure 1 illustrates the architecture of the proposed hybrid model combining 1D-CNN and LSTM for forecasting cryptocurrency price movements. The process begins with historical time-series data, where a sliding window technique is applied to generate window-based time series segments. These segments are fed into the input layer, which serves as the starting point for the convolutional network. The data passes through three convolutional layers, each followed by pooling operations, to extract hierarchical spatial features. The output of the final convolutional layer is passed to the LSTM layer, which captures sequential dependencies and temporal patterns in the data. Finally, the LSTM output is connected to a fully connected layer, which aggregates the learned features, and an output layer produces the final prediction. This design integrates the feature extraction capabilities of 1D-CNN with the temporal modeling strength of LSTM for enhanced prediction performance.

3.1 Sliding-Window technique for time series data processing

Given the transactional dataset of crypto-currency as $X = \left\{x_t | x_t \in R^n; t = 0, \pm 1, \pm 2, \cdots\right\}$. This time series dataset ($X \in R^{m \times n}$) requires the chronological order of every data

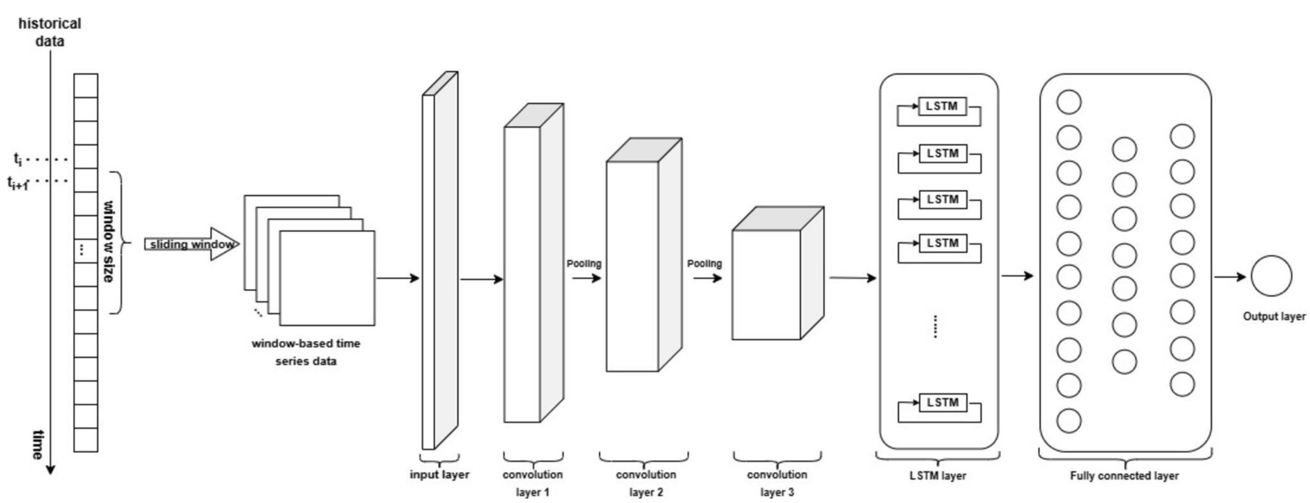

Fig. 1 The architecture of the proposed hybrid model

item, which also means that the data items are immutable. In which, $x_t \in R^n$ specifies the prices of a cryptocurrency at the time t. In order to make use of this transactional time series data set, the sliding window method is applied to pick up data windows throughout the original dataset.

The sliding window method is mathematically described in Eq. (1).

$$\varphi\left(x_t, w\right) : R^n \to R^{w\times n}, \forall x_t \in X \tag{1}$$

where $X \in R^{m\times n}$, w is the window size and $w \ll m$.

In other words, the function $\varphi\left(x_t, w\right)$ maps the original dataset from a tensor with rank 2 ($R^{m\times n}$) to a tensor with rank 3 ($R^{k\times w\times n}$). This result dataset is convenient for facilitating model training and feature engineering, reducing computation, and capturing short-term trends. Algorithm 1 presents the pseudocode of the operation of sliding window over transactional time series data.

Algorithm 1: Sliding window technique

```
function slidingWindow(data, window, forecast)
    data_size ← |data|
    X ← [ ]
    y ← [ ]
    for i from window to data_size – window – forecast:
        X ← X ∪ data[i − window − forecast :i − forecast, : ]
        y ← y ∪ data[i − forcast :i, : ]
    return X, y
end function
```

3.2 Architecture of the 1D-CNN layer

1D-CNN is a special type of convolutional neural network that focuses on temporal data or sequential structures like time series data or natural language. A typical 1D-CNN is composed of several important layers and components. The convolutional layers are responsible for extracting features. The activation function, which is used by layers, can capture complex patterns. The Pooling layers are applied to reduce the data dimension. In this research, 1D-CNN is connected to LSTM. Hence, the fully connected layers are moved to the end of the hybrid model and are discussed in the sub-Sect. 3.3.

Theoretically, a convolutional operation at the timestep t, which takes the input x and uses the kernel w of size k, yields the output y at the same timestep as in Eq. (2).

$$y(t) = (x * w)(t) = \sum_{i=0}^{k-1} x(t+i) \bullet w(i) \tag{2}$$

where * denotes the convolution operation, $x(t+i)$ is the $t+i$-th input element, and $w(i)$ is the i-th element of the kernel.

The activation function used in this hybrid model is the Rectified Linear Unit (ReLU) function. Its formula is defined in Eq. (3).

$$ReLU(z) = \max(0, z) \tag{3}$$

The max pooling, which is applied for the hybrid model, is defined in Eq. (4).

$$y(t) = \max(x(t), x(t+1), \cdots, x(t+p-1)) \tag{4}$$

During the training phase, 1D-CNN uses backpropagation to adjust the weights of the filters based on the continuous update of the errors. The loss function calculates the error based on the difference between the actual output and the predicted output. This error value is then propagated back through the network using gradient descent in order to update the weights. Equation (5) defines the gradient of convolution.

$$\frac{\partial L}{\partial w(i)} = \sum_t \frac{\partial L}{\partial y(t)} \bullet x(t+i) \tag{5}$$

where $y(t) = \sum_{i=0}^{k-1} x(t+i) \bullet w(i)$ and $\frac{\partial L}{\partial y(t)}$ is the gradient of the loss with respect to the output.

3.3 Architecture of the Long-Short-Term Memory layer

In this hybrid model, the LSTM layer, which contains a number of LSTM cells, is designed with the purpose of capturing information over long periods in sequences of crypto trading and overcoming the traditional vanishing gradient problem. Specifically, the structure of the LSTM cell is visualized in Fig. 2.

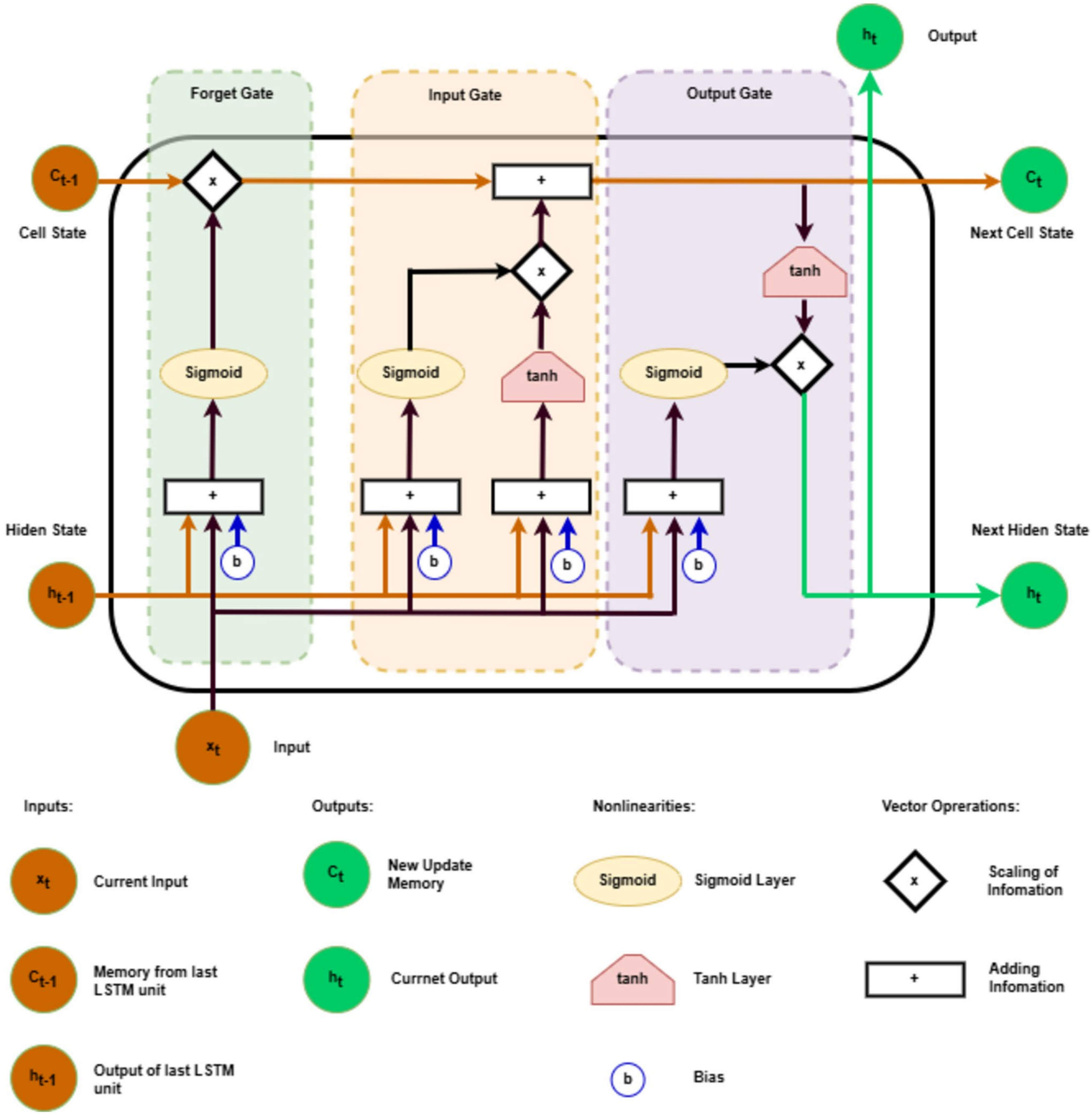

Fig. 2 The structure of a LSTM cell

The core components of a LSTM cell include the forget gate, input gate and output gate. The forget gate decides which information from the previous cell state should be discarded, while the input gate decides which new information should be added to the cell state. The output gate determines the next hidden state, which also serves as output for that time step. Mathematically, a LSTM cell can be described as a tuple $\mathcal{L}_c = \langle f_t^c, i_t^c, o_t^c, c, \widetilde{c}, h_t \rangle$ where each component is explained in Table 1.

3.4 Forecasting short-term Ethereum price

Algorithm 2 outlines a systematic approach to forecast future Ethereum price movements using a hybrid model that

Table 1 Mathematical description of the components of LSTM cell unit

Math symbols	Meaning
x_t	The input vector to the LSTM unit
W_d	The input-weight matrix where d can be $f, i, o,$ or c
U_d	The matrix of the recurrent connections where d can be f, i, o or c
b_d	The bias vector where d can be $f, i, o,$ or c
σ	The sigmoid function
τ	The hyperbolic tangent function
$f_t^c = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right)$	The forget gate of the memory cell c at the time t
$i_t^c = \sigma\left(W_i x_t + U_i h_{t-1} + b_i\right)$	The input gate of the memory cell c at the time t
$o_t^c = \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right)$	The output gate of the memory cell c at the time t
$\tilde{c}_t = \tau\left(W_c x_t + U_c h_{t-1} + b_c\right)$	The cell input activation vector at the time step t
$c_t = f_t^c \odot c_{t-1} + i_t^c \odot \tilde{c}_t$	The cell state vector at the time step t
$h_t = o_t^c \odot \tau\left(c_t\right)$	The output vector of the LSTM unit (or the hidden state vector) at the time step t
$\odot$	This operator denotes the element-wise product

combines 1D-CNN and LSTM. This algorithm is designed to process historical time-series data, transform it into window-based segments, and utilize the hybrid model to predict Ethereum prices over a specified forecast horizon. It leverages key preprocessing techniques such as sliding windows and integrates a robust model initialization and prediction pipeline.

Algorithm 2: Predict next n-day Ethereum prices

```
function PREDICT_NEXT_PRICES(file, window_size, forecast_size)
    timeseries_data ← read_data(file)
    tensors_data ← slidingWindow(timeseries_data, window_size, forecast_size)
    model ← initialize_hybrid_model()
    results ← model.predict(tensors_data.X)
    return results
end function
```

To be more specific, Algorithm 2 begins by reading the historical Ethereum price data. The data is then transformed into a series of tensor inputs using the *slidingWindow* function, which segments the data based on specified *window_size* and *forecast_size* parameters. Next, the hybrid model is initialized, which sets up the combined 1D-CNN and LSTM architecture. The prepared tensor data is fed into the model's *predict* method to generate the predicted price values for the next *n* days. Finally, the predicted results are returned, providing a comprehensive forecast for the Ethereum prices. This algorithm efficiently combines preprocessing, model initialization, and prediction in a streamlined manner.

4 Experiment

In this study, we conduct experiments to forecast Ethereum closing prices using a hybrid model that integrates 1D-CNN and LSTM networks. Ethereum trading data, including daily closing prices, was collected from Yahoo Finance,[1] covering the period from September 2014 to October 2024. To capture temporal dependencies effectively, we applied a sliding window technique on the dataset with three different configurations: (i) a window size of 5 and a step size of 1, (ii) a window size of 10 and a step size of 3, and (iii) a window size of 15 and a step size of 5. These varied window and step sizes were chosen to evaluate the impact of different temporal contexts on model performance. These sliding window time series datasets were split into training, validation, and test sets with proportions of 70%, 15%, and 15%, respectively.

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

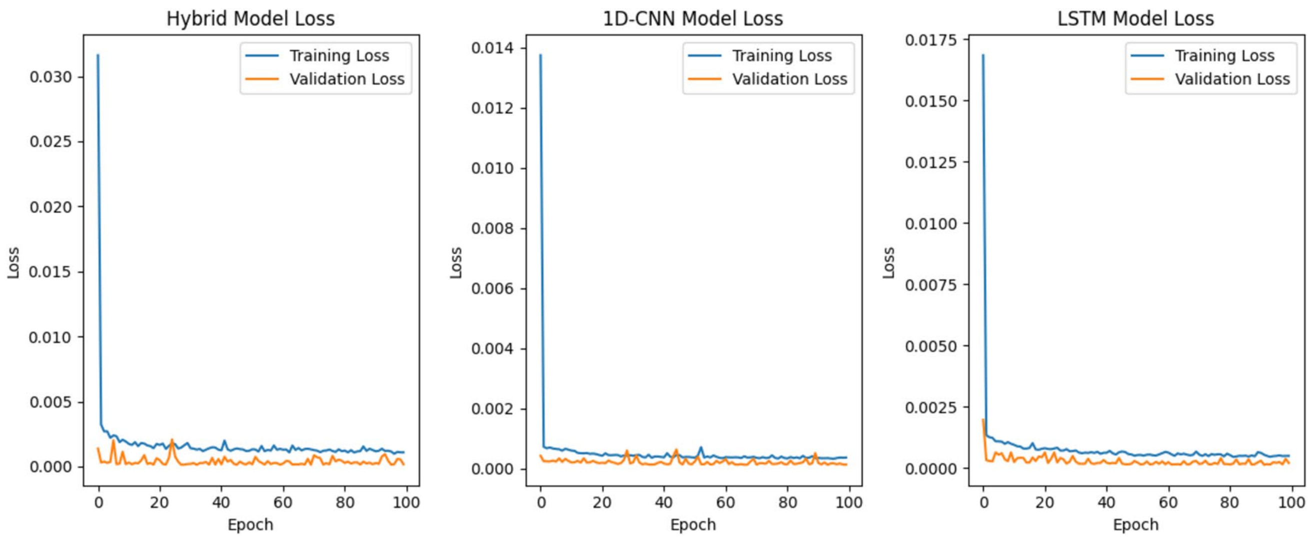

Fig. 3 Loss Curves for Window Size 5 and Step Size 1

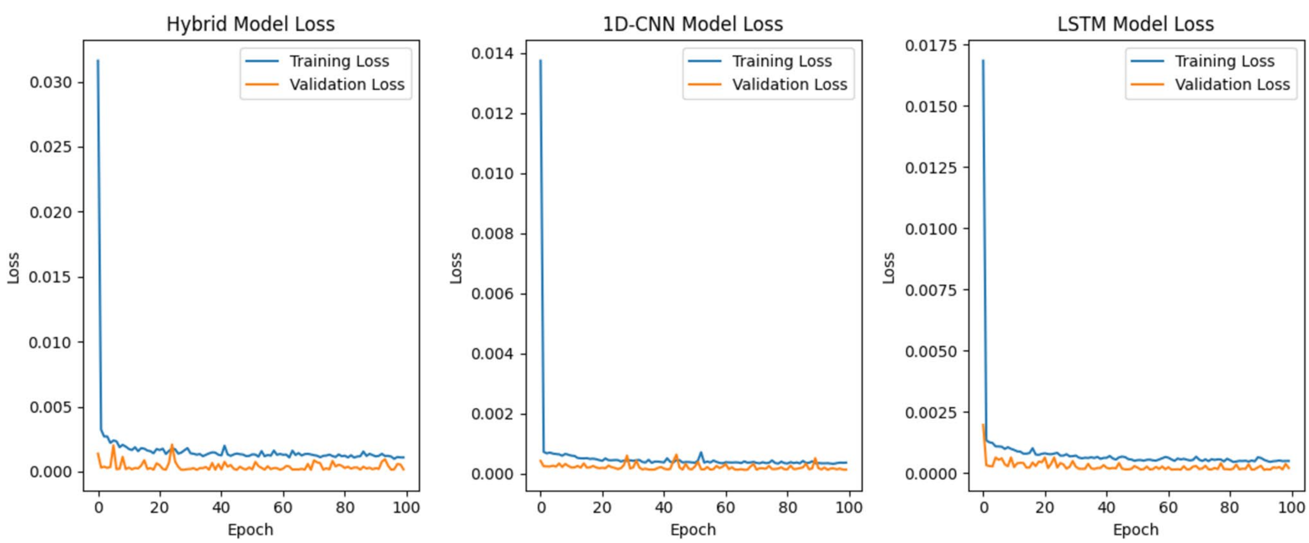

Fig. 4 Loss Curves for Window Size 10 and Step Size 3

For model development and training, we utilized the TensorFlow[2] framework, leveraging Google Colab[3] for efficient computational resources and GPU support. The hybrid 1D-CNN-LSTM model and baseline models of standalone 1D-CNN and LSTM were trained under these settings to assess performance improvements introduced by the hybrid approach in capturing Ethereum price trends. The training results of three different sliding window configurations are illustrated in Figs. 3, 4, and 5.

The three images illustrate the training and validation loss curves for the hybrid 1D-CNN-LSTM model, the standalone 1D-CNN model, and the standalone LSTM model, each trained with different sliding window configurations. In each set of plots, the hybrid model, 1D-CNN model, and LSTM model show a significant decrease in training loss during the initial epochs, followed by a stabilization phase as the models approach convergence. The validation loss remains relatively low throughout, indicating that each model generalizes well without substantial overfitting.

[2] https://www.tensorflow.org/

[3] https://colab.google/

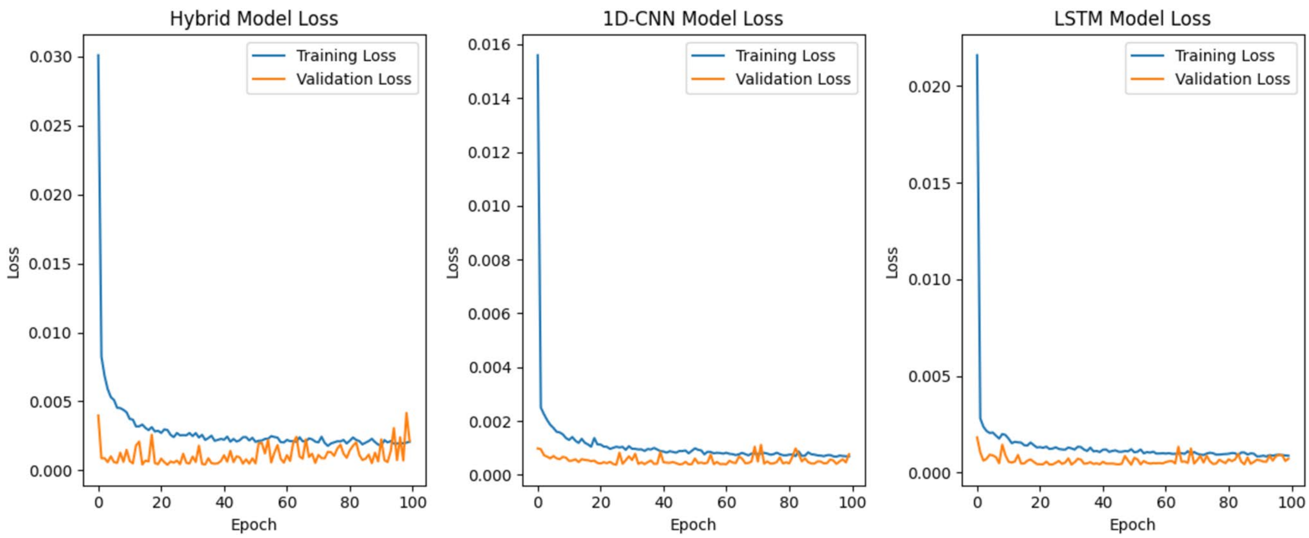

Fig. 5 Loss Curves for Window Size 15 and Step Size 5

In order to evaluate the trained models, the *MSE*, *MAE*, *RMSE*, and R^2 metrics, with formulas listed in Eqs. (6), (7), (8), and (9), were applied. The predicted values of the trained models over three test sets are visualized in Fig. 6,Fig. 7, and Fig. 8, while the evaluation results are presented in Table 2.

$$MSE(y,\hat{y}) = \frac{1}{N}\sum_{i=0}^{N-1}\left(y_i - \hat{y}_i\right)^2 \tag{6}$$

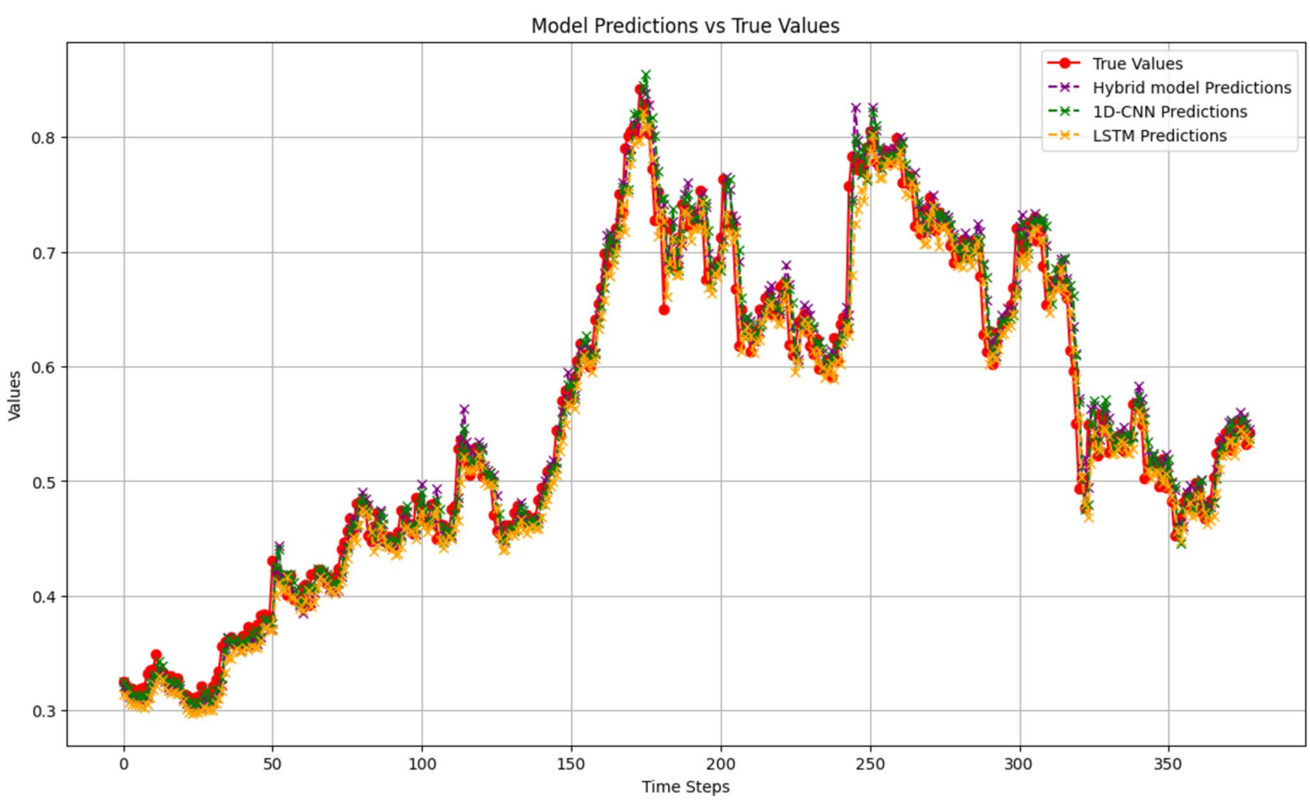

Fig. 6 Model Predictions vs. true values for 5-day window size and 1-day step size configuration

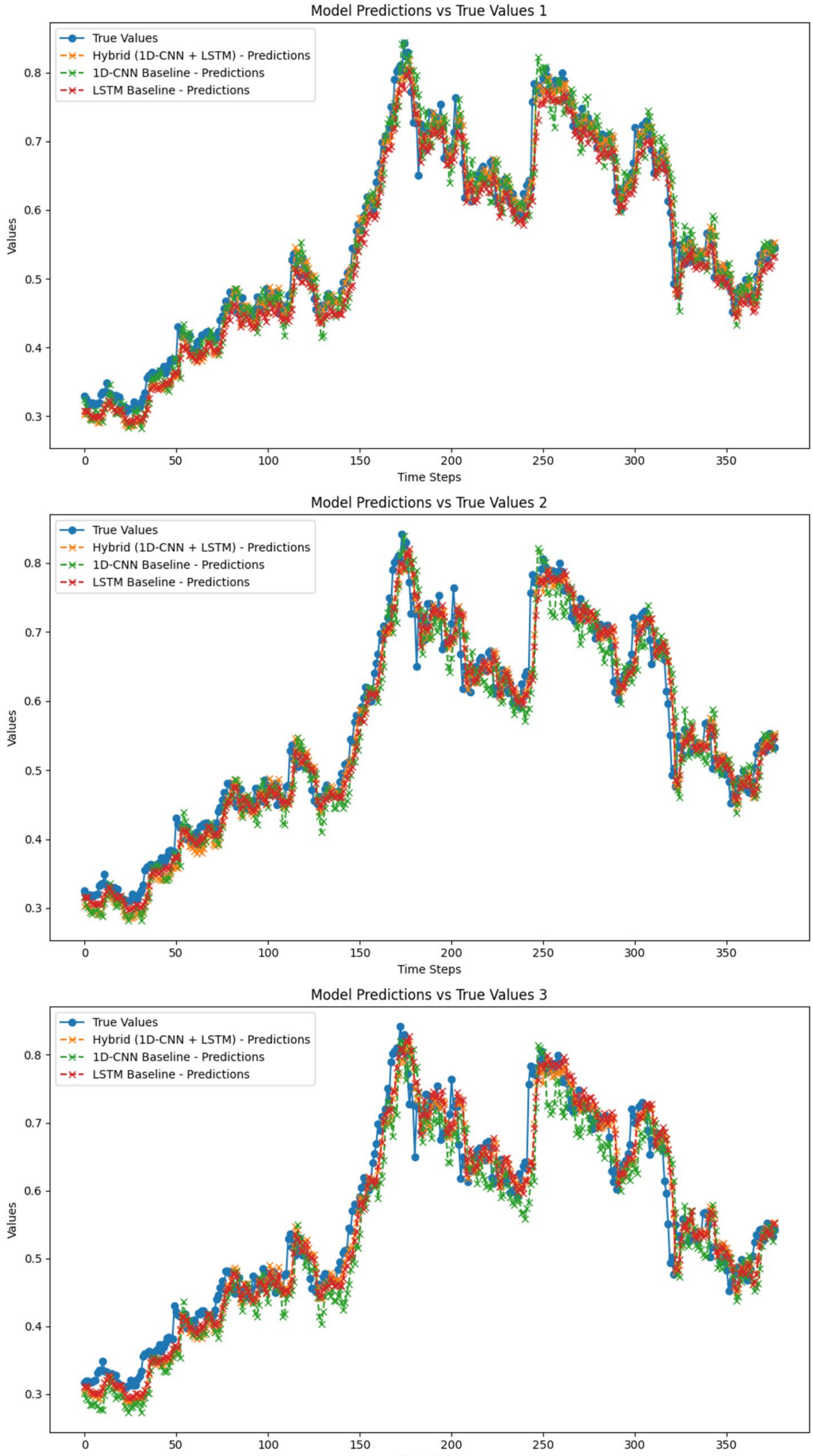

Fig. 7 Model Predictions vs. true values for 10-day window size and 3-day step size configuration

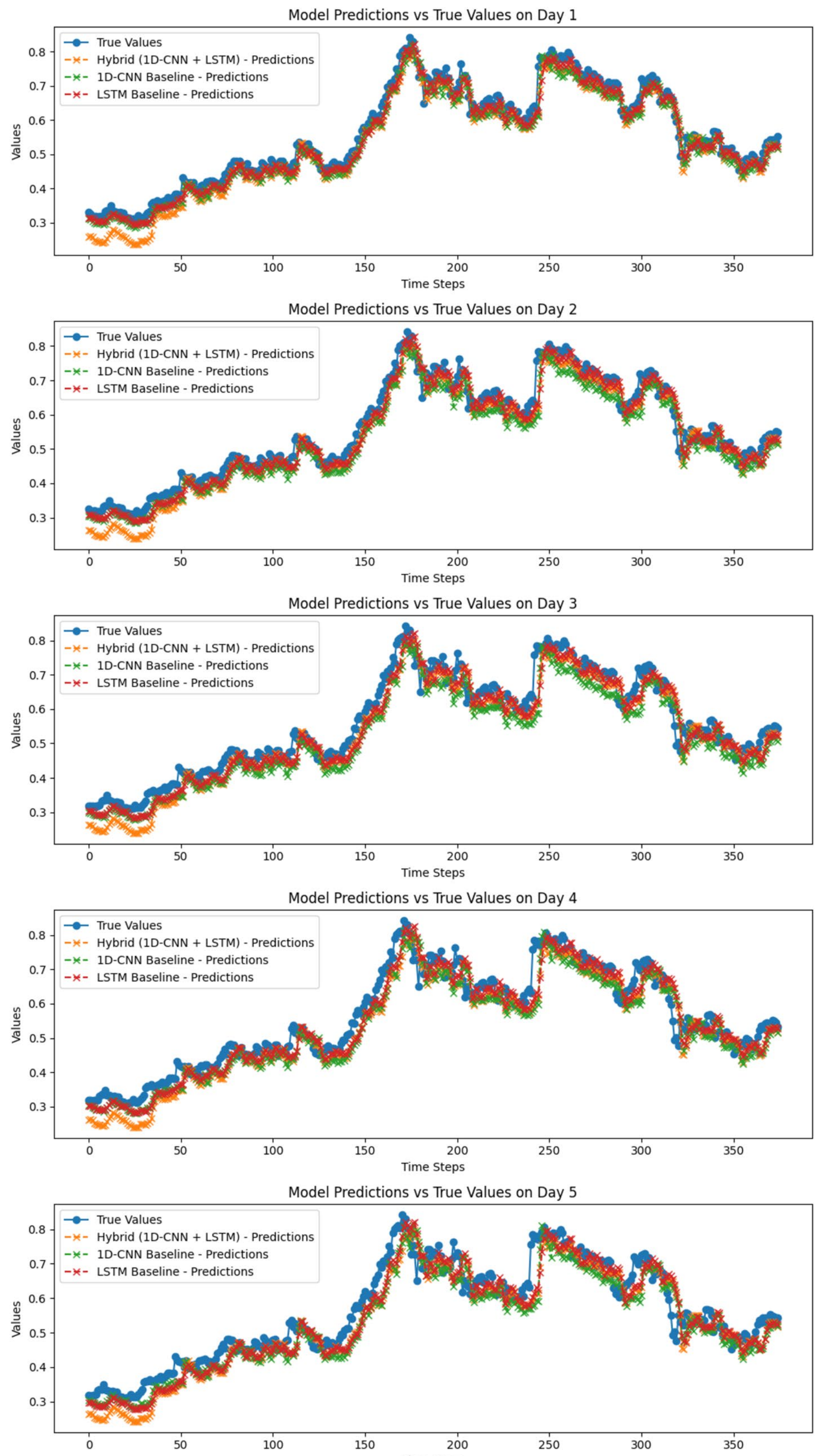

Fig. 8 Model Predictions vs. true values for 15-day window size and 5-day step size configuration

Table 2 Evaluation of model performances

Sliding window configurations	Model	MSE	MAE	RMSE	R^2
Configuration 1: Window size (5 days) Step size (1 day)	Hybrid	0.000474	0.015617	0.021764	0.975942
	1D-CNN	0.000493	0.015234	0.022201	0.974966
	LSTM	0.000519	0.017079	0.022780	0.973643
Configuration 2: Window size (10 days) Step size (3 days)	Hybrid	0.000919	0.022381	0.030307	0.953374
	1D-CNN	0.001510	0.029470	0.038856	0.923370
	LSTM	0.000953	0.023013	0.030865	0.951683
Configuration 3: Window size (15 days) Step size (5 days)	Hybrid	0.002202	0.037596	0.046921	0.887864
	1D-CNN	0.002081	0.036826	0.045618	0.893961
	LSTM	0.001442	0.028632	0.037974	0.926428

$$MAE(y,\hat{y}) = \frac{1}{N}\sum_{i=0}^{N-1}\left|y_i - \hat{y}_i\right| \tag{7}$$

$$RMSE(y,\hat{y}) = \sqrt{MSE(y,\hat{y})} \tag{8}$$

$$R^2(y,\hat{y}) = 1 - \frac{\sum_{i=0}^{N-1}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=0}^{N-1}\left(y_i - \bar{y}_i\right)^2} \tag{9}$$

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

Table 2 shows the experimental results in terms of Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R^2 scores. These scores allow us to compare the performance of the proposed hybrid model, standalone 1D-CNN, and standalone LSTM models across different sliding window configurations. For the first configuration, the hybrid model achieved the lowest MSE (0.000474) and RMSE (0.021764), with a R^2 score of 0.9759, indicating a strong predictive performance and alignment with the true values. The standalone 1D-CNN model performed similarly, with a slightly higher MSE (0.000493) and a slightly lower MAE (0.015234) but with a lower R^2 score of 0.9749. The LSTM model, while still accurate, had a higher MSE (0.000519) and the highest MAE (0.017079) among the three, reflecting a marginally lower predictive capability for this configuration.

For the second configuration, the hybrid model continued to outperform the standalone models, achieving an MSE of 0.000919 and an RMSE of 0.030307, alongside an R^2 score of 0.9534. These metrics showed that the hybrid model better captured the temporal structure with this configuration. Finally, for the third configuration, the hybrid model achieved the relatively lower scores of MSE, MAE, RMSE, and R^2 in comparison with those of the runner-up model (1D-CNN). However, all of the three models increased error metrics. These results indicated that the hybrid approach maintained an advantage across varying temporal contexts in the test sets.

In summary, the hybrid model consistently achieved the best performance in most of the configurations, reflected in its lower error metrics and higher R^2 scores. This suggested that combining the strengths of 1D-CNN in feature extraction and LSTM in handling sequential dependencies results in a robust model for time series forecasting tasks.

5 Conclusion

This study introduces a hybrid 1D-CNN-LSTM model designed to enhance the accuracy of cryptocurrency price forecasting by capturing both local and temporal dependencies in Ethereum price data. Experimental results reveal that the hybrid model consistently outperforms standalone 1D-CNN and LSTM models across various configurations, as indicated by lower MSE, MAE, and RMSE values. These findings highlight the efficacy of combining convolutional layers for feature extraction with LSTM layers for sequence modeling, creating a robust approach to time series forecasting in highly volatile cryptocurrency markets. The sliding window technique further contributed to capturing short-term trends within the temporal data, improving model robustness. Future research plans aim to optimize the hybrid model with other deep learning architectures or extend this framework to other financial assets, enhancing predictive capabilities in dynamic and complex markets.

Funding The authors did not receive support from any organization for the submitted work.

Data availability All of the data used in this research are downloaded directly from Yahoo Finance via yfinace library of Python language.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

References

1. Khedr AM, Arif I, Pravija Raj PV, El-Bannany M, Alhashmi SM, Sreedharan M (2021) Cryptocurrency price prediction using traditional statistical and machine-learning techniques: a survey. Intell Syst Account Finance Manag 28(1):3–34. https://doi.org/10.1002/isaf.1488
2. Sabry F, Labda W, Erbad A, Malluhi Q (2020) Cryptocurrencies and artificial intelligence: challenges and opportunities. IEEE Access 8:175840–175858
3. Pintelas E, Livieris IE, Stavroyiannis S, Kotsilieris T, Pintelas P (2020) Investigating the problem of cryptocurrency price prediction: a deep learning approach. In: Maglogiannis LI, Pimenidis E (eds) Artificial Intelligence Applications and Innovations, vol. 584. IFIP Advances in Information and Communication Technology. Springer International Publishing, Cham, pp. 99–110. https://doi.org/10.1007/978-3-030-49186-4_9
4. Sunil Kumar KN, Kashyap PP, Bhyratae DA, Bhyratae SA, Kalaivani A (2024) CNN-BO-LSTM: an ensemble framework for prognosis of liver cancer. Int J Inf Technol. https://doi.org/10.1007/s41870-024-02190-5
5. Vikraman BP, Mahadevan V, Jabbar RF (2024) Revolutionizing emergency vehicle response: a smart traffic management approach with adaptive CNN and hybrid deep learning. Int J Inf Technol. https://doi.org/10.1007/s41870-024-02088-2
6. Mamta K, Sangwan S (2024) AaPiDL: an ensemble deep learning-based predictive framework for analyzing customer behaviour and enhancing sales in e-commerce systems. Int J Inf Technol 16(5):3019–3025. https://doi.org/10.1007/s41870-024-01796-z
7. Bouteska A, Abedin MZ, Hajek P, Yuan K (2024) Cryptocurrency price forecasting—a comparative analysis of ensemble learning and deep learning methods. Int Rev Financ Anal 92:103055. https://doi.org/10.1016/j.irfa.2023.103055
8. Patel NP et al (2022) Fusion in cryptocurrency price prediction: a decade survey on recent advancements, architecture, and potential future directions. IEEE Access 10:34511–34538
9. Rathee N, Singh A, Sharda T, Goel N, Aggarwal M, Dudeja S (2023) Analysis and price prediction of cryptocurrencies for historical and live data using ensemble-based neural networks. Knowl Inf Syst 65(10):4055–4084. https://doi.org/10.1007/s10115-023-01871-0
10. Cavalli S, Amoretti M (2021) CNN-based multivariate data analysis for bitcoin trend prediction. Appl Soft Comput 101:107065. https://doi.org/10.1016/j.asoc.2020.107065
11. Alonso-Monsalve S, Suárez-Cetrulo AL, Cervantes A, Quintana D (2020) Convolution on neural networks for high-frequency trend prediction of cryptocurrency exchange rates using technical indicators. Expert Syst Appl 149:113250
12. Zhang Z, Dai H-N, Zhou J, Mondal SK, García MM, Wang H (2021) Forecasting cryptocurrency price using convolutional neural networks with weighted and attentive memory channels. Expert Syst Appl 183:115378
13. Kadhim AH, Al Farttoosi DS, Shakir HS, Almagtome AH (2024) An empirical cryptocurrency price forecasting model. Int J Inf Technol 16(6):3855–3860. https://doi.org/10.1007/s41870-024-01944-5
14. Aditya Pai B, Devareddy L, Hegde S, Ramya BS (2022) A time series cryptocurrency price prediction using LSTM. In: Shetty NR, Patnaik LM, Nagaraj HC, Hamsavath PN, Nalini N (eds) Emerging research in computing, information, communication and applications, vol. 790. Lecture notes in electrical engineering. Springer Singapore, Singapore, pp. 653–662. https://doi.org/10.1007/978-981-16-1342-5_50
15. Kim G, Shin D-H, Choi JG, Lim S (2022) A deep learning-based cryptocurrency price prediction model that uses on-chain data. IEEE Access 10:56232–56248. https://doi.org/10.1109/ACCESS.2022.3177888
16. Behera S, Kumar AVSP, Nayak SC (2024) Analyzing the performance of geometric mean optimization-based artificial neural networks for cryptocurrency forecasting. Int J Inf Technol. https://doi.org/10.1007/s41870-024-01953-4
17. Aziz RM, Baluch MF, Patel S, Ganie AH (2022) LGBM: a machine learning approach for Ethereum fraud detection. Int J Inf Technol 14(7):3321–3331. https://doi.org/10.1007/s41870-022-00864-6
18. Gajjar P, Saxena A, Acharya K, Shah P, Bhatt C, Nguyen TT (2024) Liquidt: stock market analysis using liquid time-constant neural networks. Int J Inf Technol 16(2):909–920. https://doi.org/10.1007/s41870-023-01506-1
19. Peng P, Chen Y, Lin W, Wang JZ (2024) Attention-based CNN–LSTM for high-frequency multiple cryptocurrency trend prediction. Expert Syst Appl 237:121520
20. Zhong C, Du W, Xu W, Huang Q, Zhao Y, Wang M (2023) LSTM-ReGAT: a network-centric approach for cryptocurrency price trend prediction. Decis Support Syst 169:113955
21. García-Medina A, Aguayo-Moreno E (2024) LSTM–GARCH hybrid model for the prediction of volatility in cryptocurrency portfolios. Comput Econ 63(4):1511–1542. https://doi.org/10.1007/s10614-023-10373-8
22. Khattak BHA, Shafi I, Rashid CH, Safran M, Alfarhood S, Ashraf I (2024) Profitability trend prediction in crypto financial markets using Fibonacci technical indicator and hybrid CNN model. J Big Data 11(1):58. https://doi.org/10.1186/s40537-024-00908-7
23. Rateb MN, Alansary S, Elzouka MK, Galal M (2024) Predicting cryptocurrency prices during periods of conflict: a comparative sentiment analysis using SVM, CNN-LSTM, and pysentimento. Oper Res Forum 5(3):74. https://doi.org/10.1007/s43069-024-00352-6
24. Huang X, et al. (2021) LSTM based sentiment analysis for cryptocurrency prediction. In: Jensen CS, Lim E-P, Yang D-N, Lee W-C, Tseng VS, Kalogeraki V, Huang J-W, Shen C-Y (eds) Database Systems for Advanced Applications, vol. 12683. Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 617–621. https://doi.org/10.1007/978-3-030-73200-4_47