International Journal of Electrical and Computer Engineering (IJECE)
Vol. 16, No. 3, June 2026, pp. 1298~1306
ISSN: 2088-8708, DOI: 10.11591/ijece.v16i3.pp1298-1306

An enhancement of stock price forecasting based on hybrid BiLSTM-Transformer model

Pham Hoang Vuong[1,2], Lam Hung Phu[2], Le Nhat Duy[1], Pham The Bao[2], Tan Dat Trinh[2]
[1]Faculty of Information Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam
[2]Department of Computer Science, Information Science Faculty, Saigon University, Ho Chi Minh City, Vietnam

Article Info

Article history:

Received Nov 6, 2024
Revised Jan 24, 2026
Accepted Mar 16, 2026

Keywords:

BiLSTM
Deep learning
Hybrid model
Stock price forecasting
Transformer model

ABSTRACT

Stock price forecasting presents a challenging problem due to factors like nonlinearity, seasonality, and economic volatility in financial data. Deep learning approaches can handle nonlinearity and complexity of financial data, but they often face limitations in capturing both local and global dependencies. This study introduces a hybrid Transformer–bidirectional long short-term memory (BiLSTM) model to improve stock price forecasting. Our method combines the strength of BiLSTM with the global context understanding of the Transformer by embedding a 1D convolutional layer. The model can efficiently capture short-term and long-term dependencies in stock data. Experimental results on various datasets show that our hybrid model outperforms other well-known models.

Corresponding Author:

Pham The Bao
Information Science Faculty, Saigon University
273 An Duong Vuong Street, Cho Quan Ward, Ho Chi Minh City, Vietnam
Email: ptbao@sgu.edu.vn

1. INTRODUCTION

Stock price forecasting (SPF) has been a topic of financial research, essential for various trading and investment strategies. Financial markets, with their volatility and complexity, are affected by many factors such as political events or market trends [1], [2]. This makes predicting stock prices a difficult task. Predicting a stock's future price involves finding patterns within unpredictable, non-linear, and multi-dimensional time-series data. Many approaches of the SPF has developed over the years [1], [3]. Statistical approaches for the SPF are mathematical frameworks that aim to forecast future prices using historical data. Autoregressive integrated moving average (ARIMA) [4], [5] are well-known methods for the SPF. Subakkar *et al.* [4] applied ARIMA to forecast Tesla's stock data. The results obtained from the ARIMA demonstrate robust accuracy for short-term stock price predictions. Suripto [5] demonstrated that the ARIMA (1,1,1) model had an applicability rate of 98% for predicting stock prices. Although ARIMA can be sufficient for short-term forecasts, its ability to predict over longer terms is uncertain.

Machine learning (ML) models have increasingly been used for the SPF. ML algorithms such as neural network (ANN) [6], support vector regression (SVR) [7], [8], ensemble models (random forest (RF), XGBoost) [6], [9], provides significant advantages over traditional statistical models. Vijh *et al.* [6] explored ANN and RF for predicting stock market closing prices. Zhang *et al.* [7] introduced an ensemble model that combines features of both SVR and the ensemble adaptive neuro fuzzy inference system. Bazrkar and Hosseini [8] applied a method for the SPF using SVM and particle swarm optimization (PSO) algorithms. The results indicated that their method consistently outperformed other techniques, achieving a prediction accuracy consistently above 90%.

Recently, deep learning (DL), with its automatic feature learning, ability to process complex patterns, and better adaptation to large datasets, offers a significant advantage. The DL models such as convolutional neural network (CNN) [10], [11] and recurrent neural network (RNN) [12]–[15] have been successfully applied to stock price forecasting. Fathali *et al.* [16] applied CNN and RNN to predict the stock trends. Satria [17] investigated RNN, long short-term memory (LSTM), and gated recurrent unit (GRU) models for predicting banking stock prices. Mootha *et al.* [18] focused on predicting future values of a stock using Bidirectional LSTM (BiLSTM) based Seq2Seq modeling. The model significantly outperformed existing algorithms.

Recently, new hybrid DL approaches [19]–[21] have shown promise in forecasting stock data. Aldhyani and Alzahrani [22] introduced CNN-LSTM approach to forecast the closing price using two years' worth of data. Chen *et al.* [23] introduced a model which combining CNN-BiLSTM-ECA (efficient channel attention) for forecasting stock data. Experiments reveal that their model overcame existing methods. Wang [24] proposed BiLSTM-MTRAN-TCN model for stock price forecasting. It demonstrates better performance on each index stock compared to other existing methods, achieving the highest R2 score in 85.7% of the stock dataset.

In this research, we introduce a new model that applies Transformer and BiLSTM architectures. BiLSTM is well-known to capture bidirectional dependencies in stock data, processing information from both the past and future to understand the local temporal structure more effectively. By integrating BiLSTM with a Transformer, our model capitalizes on both short-term sequential processing and long-term contextual understanding. Through experimental analysis, we aim to understand these models' efficacy in processing the complex time sequences in stock price changes and their capability to forecast future prices with increased precision and reliability.

2. PROPOSED METHOD

2.1. LSTM and BiLSTM models for stock price forecasting

Recurrent neural network (RNN) is effective for sequence data but have difficulty learning long-range dependencies. LSTM models were developed to overcome these limitations by reducing the vanishing gradient problem [9]. Figure 1 shows the structure of a LSTM unit, which is used for processing information. A typical LSTM consists of four main components: a memory cell, an input gate, an output gate, and a forget gate.

The memory cell equally values past and future information, ensuring that both are stored effectively. The input and output gates facilitate the unit's ability to maintain information over long periods. Specifically, the input gate determines whether new information should be added into the memory cell. Conversely, the output gate decides which portions of the memory contribute to the output. Meanwhile, the forget gate is responsible for deciding which pieces of information are to be retained and which are to be discarded. The LSTM model benefits from aggregating contexts from both past and future to produce outputs, thereby providing more comprehensive information to adjacent elements. The forward and backward LSTMs can be combined into a single network, termed a BiLSTM [25]. This network is designed to exploit information in both directions [23], [26].

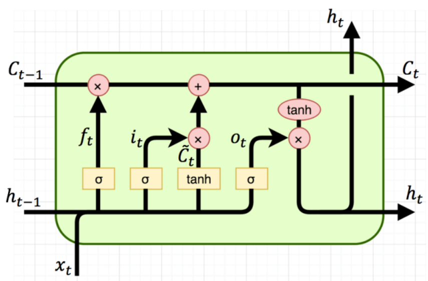

Figure 1. Illustration of a basic LSTM unit [9]

2.2. Overview of Transformer model

The Transformer model was developed for sequence-to-sequence tasks [27], [28]. It stands out from traditional RNN models by incorporating a self-attention mechanism. This feature enables the model to focus on distinct parts of the input while generating outputs. Transformers can learn relationships across various parts in the input, significantly enhancing their ability to manage long-range dependencies. The overall architecture of a Transformer is shown in Figure 2.

The encoder, described on left side of the Transformer architecture in Figure 2, essentially maps an input sequence into a series of continuous representations. These representations are passed on to the decoder. The decoder, described on right side of the architecture, takes in the encoder's output along with its own output from the previous timestep, to generate an output sequence. Transformers use multi-head self-attention to overcome CNN and LSTM limitations by evaluating input relationships and capturing long-term dependencies.

2.3. Hybrid model for stock price forecasting

The attention mechanism has been integrated with CNN–LSTM architectures [21], [23], [29], [30] to improve stock price prediction accuracy, but sequential processing in LSTMs restricts parallelization and becomes inefficient for long sequences. Transformers overcome these drawbacks by efficiently modeling complex patterns and long-term dependencies. To leverage the strengths of both architectures, our proposed hybrid model integrates multi-head attention, Conv1D, and Bidirectional LSTM layers, combining the local feature extraction with the temporal modeling and the global dependency modeling. Table 1 shows proposed hybrid model architecture for stock price forecasting using Keras framework.

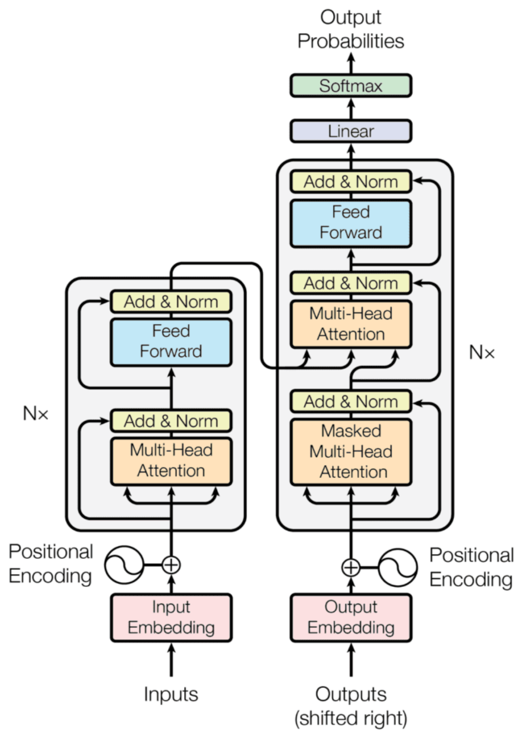

Figure 2. The Transformer architecture [31]

Table 1. Our model architecture for stock price forecasting using the Keras framework

Layer (type)	Output shape
input_layer (InputLayer)	(None, 10, 1)
layer_normalization (LayerNormalization)	(None, 10, 1)
multi_head_attention (MultiHeadAttention)	(None, 10, 1)
dropout_1 (Dropout)	(None, 10, 1)
add (Add)	(None, 10, 1)
layer_normalization_1 (LayerNormalization)	(None, 10, 1)
conv1d (Conv1D)	(None, 10, 55)
bidirectional (Bidirectional)	(None, 10, 110)
dropout_2 (Dropout)	(None, 10, 110)
add_1 (Add)	(None, 10, 110)
global_average_pooling1d (GlobalAveragePooling1D)	(None, 10)
dense (Dense)	(None, 256)
dropout_3 (Dropout)	(None, 256)
dense_1 (Dense)	(None, 1)

In our model, multi-head attention and layer normalization are applied at the input stage to identify global dependencies in stock data sequences. The Conv1D layer then captures local temporal patterns and short-term fluctuations in price movements. Following this, a BiLSTM layer processes the sequential data in both forward and backward directions, enabling the model to learn from both past and future time steps. Dropout layers are added to prevent overfitting, while Add layers establish residual connections that stabilize training and preserve important learned features. The output from the BiLSTM is passed through a global average pooling layer to reduce dimensionality. Finally, two Dense layers perform the regression task. Overall, our hybrid configuration allows the model to effectively capture short-term dependencies through convolutional and recurrent layers and long-term relationships through self-attention mechanisms. The integration of Transformer with BiLSTM and CNN components leads to improved prediction performance and robustness in stock price forecasting.

3. RESULTS AND DISCUSSION

3.1. Data collection and pre-processing

We investigate performance of our method for predicting stock prices by using a dataset at Yahoo website (*https://finance.yahoo.com/*) [25]. The dataset includes ten leading companies — Alphabet Inc., Amazon, Apple, Baidu, Intel, Microsoft, Netflix, NVIDIA, Tencent and Tesla — chosen for their strong market impact and volatility. Each dataset spans from January 3, 2012, to December 24, 2020, with 2,261 samples. Table 2 presents the statistical analysis of closing prices of various stock indices. From Table 2, we realized that Amazon has the highest average price and volatility, while Intel is the most stable with the lowest average price and standard deviation. Tesla and NVIDIA also show relatively high standard deviations (31.55 and 32.78, respectively) compared to their means, indicating high volatility and significant price fluctuations. Companies like Baidu and Microsoft show medians close to their respective means, indicating a more symmetric distribution of daily prices. More established companies like Intel and Microsoft, while still showing growth, demonstrate greater stability. The diverse behaviors of these stocks, from high volatility to stability, make them suitable for modeling stock price predictions. This value is commonly utilized to determine whether a transaction resulted in a profit or loss. Moreover, it provides critical information for investors. Therefore, we use closing price as forecasting target.

In our experiments, the dataset was split into 80% for training and 20% for testing, while 20% of training data was further used for validation. Before implementing the prediction method, preprocessing is essential. To decrease disparities and inconsistencies in the data and enhance the accuracy of our forecasting system, data normalization is employed. We utilize the MinMaxScaler method to scale data values into the [0, 1] interval.

Table 2. Statistical description of closing price of various stock indices over the study period

Stock index	Daily				
	Mean	SD	Median	Min	Max
Apple	38.64	24.13	29.77	13.94	134.17
Amazon	1009.20	822.86	726.72	175.92	3531.44
NVIDIA	29.75	32.78	11.88	2.84	145.61
Tesla	23.05	31.55	15.81	1.51	231.66
Netflix	172.78	148.49	109.95	7.68	556.54
Tencent	29.04	19.01	22.69	3.99	80.83
Microsoft	77.95	52.06	55.66	26.37	231.64
Intel	37.43	11.63	34.86	19.36	68.47
Alphabet Inc.	40.89	18.51	37.58	13.92	91.39
Baidu	165.43	46.97	166.22	83.58	284.07

SD: standard deviation

3.2. Experiment setup

Choosing an appropriate time step presents significant challenges in time series forecasting. A too small timestep can cause large deviations and excessively high computation cost, overlooking impact of global market factors. Conversely, a large time step may introduce high volatility, reducing the accuracy of predictions. Based on experiments, we utilize a time step of 10. It incorporates a 60-head multi-head attention mechanism, a 55-unit Bi-LSTM layer, and one Transformer blocks. A dense layer with 256 rectified linear unit (ReLU) units and a dropout rate of 0.3 is applied before the final linear output layer. During the training process, learning rate was set to 0.0001. The network is trained using the backpropagation algorithm with the Adam optimizer, a batch size of 20, and 100 epochs. Loss function is calculated by MSE loss. Model accuracy was assessed using root mean square error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE) as performance metrics.

3.3. Results analysis

In this section, the well-known forecasting models, and our hybrid model in stock price forecasting are evaluated. We explore the performance of these models on ten stock price indices. Table 3 compares the performance of the models on the stock datasets. Our model's accuracy varies across different stocks, reflecting each stock's unique volatility and market behavior.

Table 3. Performance of different approaches across ten stock indices

Measures	Model					
	ARIMA	LSTM	CNN+LSTM	CRNN+Attention	Transformer	Our model
Apple (AAPL)						
RMSE	0.064	0.031	0.033	0.024	0.021	0.022
MAE	0.044	0.024	0.023	0.017	0.014	0.014
MAPE	4.488	2.471	2.327	1.705	1.482	1.422
Amazon (AMZN)						
RMSE	0.037	0.024	0.037	0.024	0.017	0.026
MAE	0.026	0.018	0.027	0.017	0.012	0.018
MAPE	2.635	1.873	2.704	1.702	1.293	1.814
NVIDIA (NVDA)						
RMSE	0.045	0.033	0.052	0.068	0.019	0.021
MAE	0.032	0.022	0.036	0.044	0.013	0.014
MAPE	3.224	2.275	3.693	4.469	1.354	1.445
Tesla (TSLA)						
RMSE	0.234	0.023	0.032	0.021	0.018	0.018
MAE	0.131	0.013	0.023	0.014	0.010	0.010
MAPE	1.319	1.338	2.395	1.412	1.097	0.999
Netflix (NFLX)						
RMSE	0.034	0.041	0.038	0.037	0.035	0.035
MAE	0.025	0.029	0.028	0.027	0.024	0.024
MAPE	2.517	2.913	2.827	2.761	2.429	2.434
Tencent (TCEHY)						
RMSE	0.035	0.034	0.035	0.033	0.039	0.030
MAE	0.027	0.025	0.026	0.024	0.023	0.021
MAPE	2.774	2.503	2.634	2.484	2.396	2.189
Microsoft (MSFT)						
RMSE	0.054	0.038	0.033	0.032	0.031	0.031
MAE	0.037	0.026	0.023	0.022	0.021	0.020
MAPE	3.733	2.689	2.331	2.216	2.109	2.058
Intel (INTC)						
RMSE	0.068	0.062	0.068	0.070	0.059	0.059
MAE	0.046	0.039	0.044	0.042	0.037	0.036
MAPE	4.693	3.912	4.444	4.224	3.750	3.601
Alphabet Inc. (GOOG)						
RMSE	0.039	0.039	0.037	0.036	0.034	0.034
MAE	0.028	0.027	0.027	0.026	0.022	0.022
MAPE	2.834	2.778	2.721	2.609	2.279	2.268
Baidu (BIDU)						
RMSE	0.026	0.034	0.038	0.034	0.030	0.031
MAE	0.018	0.022	0.025	0.023	0.020	0.021
MAPE	1.815	2.253	2.542	2.359	2.048	2.068

As shown in Table 3, AAPL and AMZN achieve the highest accuracy with the lowest RMSE, MAE, and MAPE values. TSLA also performs well with low error rates, while NVDA, MSFT, and BIDU show moderate errors. In contrast, NFLX, TCEHY, GOOG, and especially INTC obtain higher errors across all metrics, indicating greater prediction difficulty. Figure 3 compares actual and predicted stock prices across multiple companies, revealing our model's forecasting performance. For AAPL and MSFT as shown in Figures 3(a) and 3(g), predictions closely follow actual trends, reflecting strong performance with stable stocks. AMZN and GOOG, as shown in Figures 3(b) and 3(i), also show accurate trend capture, with slight lag during rapid price increase. The model captures long-term patterns but struggles in highly volatile cases such as NVDA and TSLA as shown in Figures 3(c) and 3(d). NFLX, INTC and BIDU, as shown in Figures 3(e), 3(h) and 3(j), achieve moderate accuracy, while TCEHY as shown in Figure 3(f) predictions occasionally diverge yet still capture overall trends.

Furthermore, we evaluate the performance of our method in comparison to other well-known models in predicting market prices as shown in Table 3. Different models, including ARIMA [4], [9], LSTM [13], [32], CNN+ LSTM [19], [29], CRNN+Attention [23], [29], [30], Transformer [33], [34], and proposed hybrid model, have been evaluated on ten stock price datasets. First, we consider that classical ARIMA model generally obtains higher error rates across all datasets. The ARIMA results reflect its limitations in capturing nonlinear dynamics of stock data. Its effectiveness is limited, particularly in financial time series, which often present nonlinear and non-stationary characteristics [32], [35].

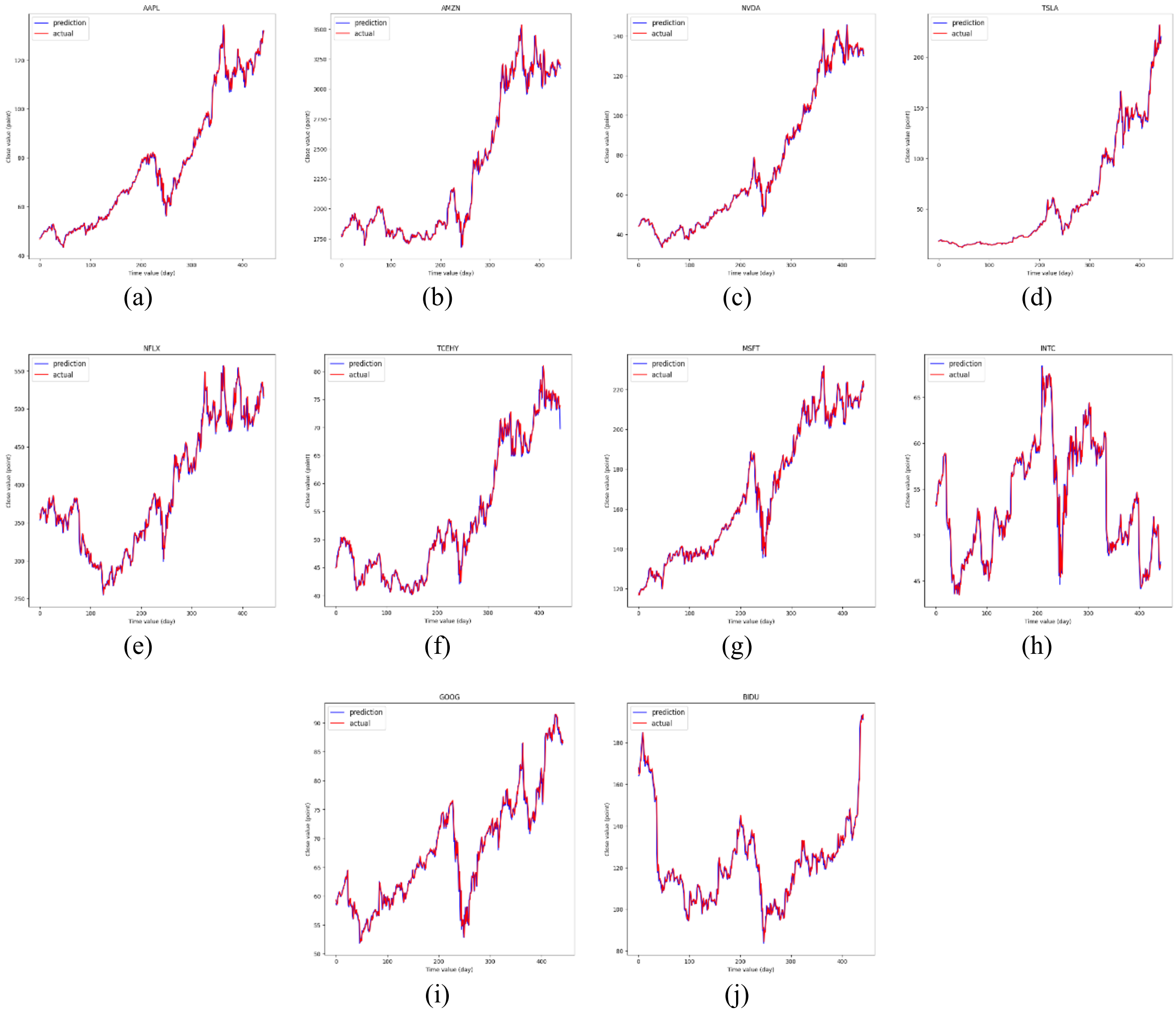

Figure 3. Results of stock price forecasting of proposed approach on ten stock data (a) AAPL, (b) AMZN, (c) NVDA, (d) TSLA, (e) NFLX, (f) TCEHY, (g) MSFT, (h) INTC, (i) GOOG, and (j) BIDU

The results in Table 3 indicate that deep learning models outperform the ARIMA on several datasets. Specifically, LSTM model demonstrates better performance compared to ARIMA, indicating their capability to model stock data with long-term dependencies. While combining convolutional layers with recurrent layers, CNN+LSTM model can capture temporal and sequential dependencies in time Serie data [19], [20]. They can outperform LSTM models alone. The CRNN+Attention has led to enhanced performance of the prediction model. Additionally, Transformer models often outperform other approaches, demonstrating their ability to capture complex dependencies using multi-head attention. Our proposed hybrid model consistently yields the best performance across most datasets. The significant performance of our hybrid model indicates a promising area for future research.

3.4. Research limitations

The most significant drawback of our work is the high computational time. The ARIMA model is the fastest method and does not require GPU computing. Deep learning approaches use GPU computing, resulting in increased computation cost [36]. The incorporation of the BiLSTM layer adds to the complexity and computational expense of the model, making real-time forecasting challenging. Additionally, finding optimal parameters for each stock dataset proved difficult, as variations in market behavior demand distinct configurations. Furthermore, relying only on historical data for predicting future values makes it challenging to address sudden changes, which vary different stocks during periods of high volatility or unexpected market events. Therefore, integrating external data sources like news sentiment or other market indicators, could enhance adaptability to real-world volatility.

4. CONCLUSION

In this study, we introduced a hybrid model for stock price forecasting that integrates BiLSTM into the feed-forward layer of Transformer model, we addressed limitations in capturing local and long-range temporal dependencies in stock prices. The model takes advantage of BiLSTM's strength in sequential data processing and the Transformer's self-attention mechanism, which allows for global context understanding. Our experiments show that the model consistently overcame baseline approaches, delivering better accuracy in forecasting stock prices across most datasets. Future work will aim to optimize the model for real-time use and enhance its predictive accuracy by integrating multivariate forecasting methods and incorporating additional data sources, including market sentiment and economic indicators.

FUNDING INFORMATION

Authors state no funding 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
Pham Hoang Vuong	✓	✓	✓	✓	✓	✓		✓	✓	✓				
Lam Hung Phu			✓	✓				✓			✓			
Le Nhat Duy				✓				✓						
Pham The Bao	✓									✓		✓		
Tan Dat Trinh	✓	✓		✓						✓		✓		

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

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] P. H. Vuong, L. H. Phu, T. H. Van Nguyen, L. N. Duy, P. T. Bao, and T. D. Trinh, "A bibliometric literature review of stock price forecasting: From statistical model to deep learning approach," *Science Progress*, vol. 107, no. 1, p. 00368504241236557, 2024, doi: 10.1177/00368504241236557.

[2] I. K. Nti, A. F. Adekoya, and B. A. Weyori, "A systematic review of fundamental and technical analysis of stock market predictions," *Artificial Intelligence Review*, vol. 53, no. 4, pp. 3007–3057, 2020, doi: 10.1007/s10462-019-09754-z.

[3] M. M. Kumbure, C. Lohrmann, P. Luukka, and J. Porras, "Machine learning techniques and data for stock market forecasting: A literature review," *Expert Systems with Applications*, vol. 197, p. 116659, 2022, doi: 10.1016/j.eswa.2022.116659.

[4] A. Subakkar, S. Graceline Jasmine, L. Jani Anbarasi, J. Ganesh, and C. M. Yuktha Sri, "An analysis on Tesla's stock price

forecasting using ARIMA model," in *Cognitive Science and Technology*, 2023, pp. 83–89, doi: 10.1007/978-981-19-2358-6_8.
[5] Suripto, "Decision-making model to predict auto-rejection: An implementation of ARIMA for accurate forecasting of stock price volatility during the Covid-19," *Decision Science Letters*, vol. 12, no. 1, pp. 107–116, 2023, doi: 10.5267/j.dsl.2022.10.002.
[6] M. Vijh, D. Chandola, V. A. Tikkiwal, and A. Kumar, "Stock closing price prediction using machine learning techniques," *Procedia Computer Science*, vol. 167, pp. 599–606, 2020, doi: 10.1016/j.procs.2020.03.326.
[7] J. Zhang, L. Li, and W. Chen, "Predicting stock price using two-stage machine learning techniques," *Computational Economics*, vol. 57, no. 4, pp. 1237–1261, 2021, doi: 10.1007/s10614-020-10013-5.
[8] M. J. Bazrkar and S. Hosseini, "Predict stock prices using supervised learning algorithms and particle swarm optimization algorithm," *Computational Economics*, vol. 62, no. 1, pp. 165–186, 2023, doi: 10.1007/s10614-022-10273-3.
[9] P. H. Vuong, T. T. Dat, T. K. Mai, P. H. Uyen, and P. T. Bao, "Stock-price forecasting based on XGBoost and LSTM," *Computer Systems Science and Engineering*, vol. 40, no. 1, pp. 237–246, 2022, doi: 10.32604/CSSE.2022.017685.
[10] H. Rezaei, H. Faaljou, and G. Mansourfar, "Stock price prediction using deep learning and frequency decomposition," *Expert Systems with Applications*, vol. 169, p. 114332, 2021, doi: 10.1016/j.eswa.2020.114332.
[11] H. Chung and K. shik Shin, "Genetic algorithm-optimized multi-channel convolutional neural network for stock market prediction," *Neural Computing and Applications*, vol. 32, no. 12, pp. 7897–7914, 2020, doi: 10.1007/s00521-019-04236-3.
[12] S. Hansun and J. C. Young, "Predicting LQ45 financial sector indices using RNN-LSTM," *Journal of Big Data*, vol. 8, no. 1, 2021, doi: 10.1186/s40537-021-00495-x.
[13] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "The performance of LSTM and BiLSTM in forecasting time series," in *Proceedings - 2019 IEEE International Conference on Big Data, Big Data 2019*, Dec. 2019, pp. 3285–3292, doi: 10.1109/BigData47090.2019.9005997.
[14] X. Zhao, Y. Liu, and Q. Zhao, "Generalized loss-based CNN-BiLSTM for stock market prediction," *International Journal of Financial Studies*, vol. 12, no. 3, p. 61, 2024, doi: 10.3390/ijfs12030061.
[15] K. Srijiranon, Y. Lertratanakham, and T. Tanantong, "A hybrid framework using PCA, EMD and LSTM methods for stock market price prediction with sentiment analysis," *Applied Sciences (Switzerland)*, vol. 12, no. 21, p. 10823, 2022, doi: 10.3390/app122110823.
[16] Z. Fathali, Z. Kodia, and L. Ben Said, "Stock market prediction of NIFTY 50 index applying machine learning techniques," *Applied Artificial Intelligence*, vol. 36, no. 1, p. 2111134, 2022, doi: 10.1080/08839514.2022.2111134.
[17] D. Satria, "Predicting banking stock prices using RNN, LSTM, and GRU approach," *Applied Computer Science*, vol. 19, no. 1, pp. 82–94, 2023, doi: 10.35784/acs-2023-06.
[18] S. Mootha, S. Sridhar, R. Seetharaman, and S. Chitrakala, "Stock price prediction using Bi-directional LSTM based sequence to sequence modeling and multitask learning," in *2020 11th IEEE Annual Ubiquitous Computing, Electronics and Mobile Communication Conference, UEMCON 2020*, 2020, pp. 0078–0086, doi: 10.1109/UEMCON51285.2020.9298066.
[19] A. Kanwal, M. F. Lau, S. P. H. Ng, K. Y. Sim, and S. Chandrasekaran, "BiCuDNNLSTM-1dCNN — A hybrid deep learning-based predictive model for stock price prediction," *Expert Systems with Applications*, vol. 202, p. 117123, 2022, doi: 10.1016/j.eswa.2022.117123.
[20] J. Shah, D. Vaidya, and M. Shah, "A comprehensive review on multiple hybrid deep learning approaches for stock prediction," *Intelligent Systems with Applications*, vol. 16, p. 200111, 2022, doi: 10.1016/j.iswa.2022.200111.
[21] Q. Chen, W. Zhang, and Y. Lou, "Forecasting stock prices using a hybrid deep learning model integrating attention mechanism, multi-layer perceptron, and bidirectional long-short term memory neural network," *IEEE Access*, vol. 8, pp. 117365–117376, 2020, doi: 10.1109/ACCESS.2020.3004284.
[22] T. H. H. Aldhyani and A. Alzahrani, "Framework for predicting and modeling stock market prices based on deep learning algorithms," *Electronics (Switzerland)*, vol. 11, no. 19, p. 3149, 2022, doi: 10.3390/electronics11193149.
[23] Y. Chen *et al.*, "Stock price forecast based on CNN-BiLSTM-ECA model," *Scientific Programming*, vol. 2021, p. 2446543, 2021, doi: 10.1155/2021/2446543.
[24] S. Wang, "A stock price prediction method based on BiLSTM and improved Transformer," *IEEE Access*, vol. 11, pp. 104211–104223, 2023, doi: 10.1109/ACCESS.2023.3296308.
[25] H. V Pham, H. P. Lam, L. N. Duy, T. B. Pham, and T. D. Trinh, "An improved convolutional recurrent neural network for stock price forecasting," *IAES International Journal of Artificial Intelligence*, vol. 13, no. 3, pp. 3381–3394, 2024, doi: 10.11591/ijai.v13.i3.pp3381-3394.
[26] T. T. Dat *et al.*, "An improved CRNN for Vietnamese identity card information recognition," *Computer Systems Science and Engineering*, vol. 40, no. 2, pp. 539–555, 2021, doi: 10.32604/CSSE.2022.019064.
[27] T. Lin, Y. Wang, X. Liu, and X. Qiu, "A survey of Transformers," *AI Open*, vol. 3, pp. 111–132, 2022, doi: 10.1016/j.aiopen.2022.10.001.
[28] N. Patwardhan, S. Marrone, and C. Sansone, "Transformers in the real world: A survey on NLP applications," *Information (Switzerland)*, vol. 14, no. 4, 2023, doi: 10.3390/info14040242.
[29] W. Lu, J. Li, J. Wang, and L. Qin, "A CNN-BiLSTM-AM method for stock price prediction," *Neural Computing and Applications*, vol. 33, no. 10, pp. 4741–4753, May 2021, doi: 10.1007/s00521-020-05532-z.
[30] J. Zhang, L. Ye, and Y. Lai, "Stock price prediction using CNN-BiLSTM-attention model," *Mathematics*, vol. 11, no. 9, p. 1985, 2023.
[31] A. Vaswani *et al.*, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 2017-December, pp. 5999–6009, 2017, doi: 10.1201/9781003561460-19.
[32] D. Kobiela, D. Krefta, W. Król, and P. Weichbroth, "ARIMA vs LSTM on NASDAQ stock exchange data," *Procedia Computer Science*, vol. 207, pp. 3836–3845, 2022, doi: 10.1016/j.procs.2022.09.445.
[33] C. Wang, Y. Chen, S. Zhang, and Q. Zhang, "Stock market index prediction using deep Transformer model," *Expert Systems with Applications*, vol. 208, p. 118128, 2022, doi: 10.1016/j.eswa.2022.118128.
[34] C. Li and G. Qian, "Stock price prediction using a frequency decomposition based GRU Transformer neural network," *Applied Sciences (Switzerland)*, vol. 13, no. 1, p. 222, 2023, doi: 10.3390/app13010222.
[35] F. H. Mustapa and M. T. Ismail, "Modelling and forecasting S&P 500 stock prices using hybrid Arima-Garch model," *Journal of Physics: Conference Series*, vol. 1366, no. 1, p. 12130, 2019, doi: 10.1088/1742-6596/1366/1/012130.
[36] T. Singh, R. Kalra, S. Mishra, Satakshi, and M. Kumar, "An efficient real-time stock prediction exploiting incremental learning and deep learning," *Evolving Systems*, vol. 14, no. 6, pp. 919–937, 2023, doi: 10.1007/s12530-022-09481-x.

BIOGRAPHIES OF AUTHORS

Pham Hoang Vuong received his B.Sc. degree in mathematics and computer science from the University of Education – Vietnam National University, Ho Chi Minh City, in 1992. He earned his Engineering degree from Hanoi University of Science and Technology in 2000 and his M.Sc. degree from Ho Chi Minh City University of Science in 2001. He has been a Lecturer in the Department of Computer Science at Saigon University, Vietnam, since 2008. His research interests include stock price forecasting and computer vision using deep learning. He can be contacted at email: phvuong@sgu.edu.vn.

Lam Hung Phu received his bachelor's degree in information technology from Saigon University, Ho Chi Minh City, Vietnam, in 2023. He is a recent graduate, and his research interests include predictive modeling, computer vision, and data analysis. He can be contacted at email: pth.a3.30.hungphu@gmail.com.

Le Nhat Duy received his Ph.D. degree in computer science from Moscow State Pedagogical University, Russia, in 2013. Since 2013, he has been a Lecturer in the Department of Computer Science at the Industrial University of Ho Chi Minh City, Vietnam. His research interests include computational intelligence and cryptography. He can be contacted at email: lenhatduy@iuh.edu.vn.

Pham The Bao received his B.Sc. degree in Algebra from the University of Natural Science – Vietnam National University, Ho Chi Minh City, in 1995. He earned his M.Sc. degree in the mathematical foundations of computer science in 2000 and his Ph.D. degree in computer science in 2009, both from the same university. From 1995 to 2018, he served as a Lecturer and Professor in the Department of Computer Science, Faculty of Mathematics and Computer Science, at the University of Natural Science. Since 2019, he has been the Dean and a Professor in the Department of Computer Science at Saigon University, Vietnam. He has published over 50 papers in international journals and conferences. His research interests include image processing, pattern recognition, and intelligent computing. He can be contacted at email: ptbao@sgu.edu.vn.

Tan Dat Trinh received his B.Sc. degree in mathematics and computer science from the University of Natural Science – Vietnam National University, Ho Chi Minh City, in 2010. He earned his Master of Engineering (M.Eng.) and Ph.D. degrees in electronics and computer engineering from Chonnam National University, Korea, in 2013 and 2017, respectively. Since 2019, he has been a lecturer in the Department of Computer Science at Saigon University, Vietnam. His research interests include speaker recognition, speech signal processing, computer vision, and pattern recognition. He can be contacted at email: trinhtandat@sgu.edu.vn.