Proceedings of the 8th International Conference on Economic Management and Green Development
DOI: 10.54254/2754-1169/104/2024ED0152

Application of Attention-Based LSTM Hybrid Models for Stock Price Prediction

Xinhao Sun[1,a,*]

[1]*Qianweichang College, Shanghai University, Shanghai 200444, China.*

[a]*sunxinhao@shu.edu.cn*

[*]*corresponding author*

Abstract: The stock market plays a pivotal role in the national economy, while the application of artificial intelligence (AI) in stock price prediction has gained traction. This paper evaluates the performance of five advanced deep learning (DL) models: Long Short-Term Memory (LSTM), Self-attention, Convolutional Neural Network-LSTM with attention (CNN-LSTM-attention), Gated Recurrent Unit-LSTM with attention (GRU-LSTM-attention), and CNN-Bidirectional LSTM-GRU with attention (CNN-BiLSTM-GRU-attention), utilizing a decade of data on Amazon's closing prices. Our results show that the CNN-BiLSTM-GRU-attention model exhibits superior performance, achieving a root mean square error (RMSE) of 1.054589 and a coefficient of determination (R^2) of 0.970123, indicative of its proficiency in handling intricate financial data. This paper's significance lies in its validation of the effectiveness of attention-based ensemble models in stock market prediction, as well as the introduction of the innovative application of the CNN-BiLSTM-GRU-attention model in financial forecasting, which holds potential for wide-ranging applicability.

Keywords: Stock price prediction, Deep learning, LSTM, Attention-based model, CNN-BiLSTM-GRU-attention model

1. Introduction

The stock market, serving as a fundamental component of every national economy, plays a critical role by facilitating the trading of shares and other financial instruments. Accurately predicting stock prices is crucial for investors seeking to maximize profits through strategic long and short decisions. In recent decades, with billions of dollars traded daily, there has been a marked increase in public and institutional interest in the stock market, highlighting its significance in global financial dynamics.

Artificial intelligence (AI), particularly deep learning (DL), has emerged as an essential tool in quantitative finance, adept at modeling the intricate, long-term dependencies inherent in stock market movements. DL, such as Convolutional Neural Networks (CNNs), are renowned for their prowess in feature extraction. Meanwhile, recurrent neural network model, such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), excel in sequence prediction, capturing temporal dynamics essential for time series analysis. Bidirectional LSTM (BiLSTM) further enhances the model's capability by processing data in both forward and backward direction. Additionally, attention mechanisms allow models to focus on specific parts of the input sequence, allowing for more precise and robust predictions. These advanced technologies collectively empower the extraction of subtle patterns in financial data, leading to more accurate forecasts of stock prices.

In this study, we explore five advanced DL algorithms: LSTM, Self-attention, CNN-LSTM with attention (CNN-LSTM-attention), GRU-LSTM with attention (GRU-LSTM-attention), and CNN-Bidirectional LSTM-GRU with attention (CNN-BiLSTM-GRU-attention). The CNN-LSTM-attention model combines CNNs' feature extraction capabilities with LSTMs' sequential prediction proficiencies, augmented by an attention mechanism. This integration fosters a synergistic effect, significantly improving the model's ability to uncover underlying patterns in time series data. Similarly, the GRU-LSTM-attention model merges the operational efficiencies and strengths of both GRU and LSTM models to enhance forecast accuracy. Furthermore, the hybrid CNN-BiLSTM-GRU-attention model incorporates BiLSTM, a technique with promising results across various applications, such as environmental pollutant forecasting and emotion recognition. This model demonstrates effectiveness in complex pattern recognition tasks.

To evaluate these models' performance, this paper utilizes the closing prices of Amazon stock spanning from April 25, 2014, to April 25, 2024, totaling 2,518 data entries. We employ root mean square error (RMSE) and the coefficient of determination (R^2) as evaluation metrics. Among the analyzed models, the hybrid CNN-BiLSTM-GRU-attention model outperforms others, achieving an RMSE of 1.054589 and an R^2 of 0.970123, albeit with the longest operation time. In contrast, the standalone LSTM model, exhibits the lowest performance with an RMSE of 2.103577 and an R^2 of 0.881128, despite the least computational time.

The principal contributions of this paper can be summarized in three main aspects:

1. Innovative application of attention-based integration models: This study pioneers the application of Attention-based LSTM integration models, including CNN-LSTM-attention, GRU-LSTM-attention, and CNN-BiLSTM-GRU-attention, in the domain of stock price prediction. By exploring their synergistic effects in capturing complex financial patterns, this research expands the scope of their applicability in quantitative finance.
2. Novel implementation of the CNN-BiLSTM-GRU-attention Model: The introduction of this model for stock price forecasting fills a significant gap in existing financial modeling research. Through a detailed comparative analysis, this paper illustrates the effectiveness of this model relative to established ones, highlighting its unique advantages and robustness in predicting financial time series.
3. Superior performance and potential applicability: Our model stands out not only for its accuracy but also for its versatility across various contexts. The CNN-BiLSTM-GRU-attention model has consistently demonstrated superior predictive precision and reliability compared to other models. This success suggests its potential applicability to diverse financial markets and instruments, with significant implications for both theoretical advancements and practical applications in financial analysis and forecasting.

The remainder of this paper is structured as follows: In Section 2., we provide a comprehensive literature review, synthesizing relevant prior studies and their findings in the realm of financial time series prediction. Section 3. elaborates on the methodology, delineating the DL employed. The experimental setup, encompassing data description and model configurations, is outlined in Section 4., alongside the presentation of results through both graphical and quantitative analyses. Finally, Section

DOI: 10.54254/2754-1169/104/2024ED0152

5. offers a conclusive summary of the paper's findings, discusses the implications of the results, and outlines avenues for future research.

2. Literature review

The rise of AI has led to its widespread adoption across various sectors, including healthcare [1], finance [2], autonomous systems [3], and natural language processing [4]. DL, an offshoot of machine learning, has emerged as a focal point in this technological landscape [5]. Its deployment in the realm of stock market forecasting has been particularly prominent, primarily due to DL's intrinsic capabilities to decipher nonlinearity, navigate chaotic and noisy datasets, and process complex financial data, making it a powerful tool for predicting market trends and movements [6][7].

Among DL techniques, CNNs excel at detecting spatial patterns in data, beneficial for identifying market trends [8][9]. Recurrent Neural Networks (RNNs), on the other hand, process sequential information, crucial for modeling stock price movements [10][11]. GRUs advance RNNs by resolving the vanishing gradient problem through gating mechanisms, ensuring crucial long-term information is preserved in financial time series analysis [12].

LSTM models, a subclass of RNNs, have become a cornerstone in the prediction of stock market trends due to their exceptional ability to learn from sequence data over long periods without the risk of vanishing gradients [13]. These models are particularly suited for financial markets where long-term historical data is pivotal in capturing the underlying patterns that drive movements in stock prices [14]. A seminal work by [15] demonstrates that LSTM networks outperform traditional time series models and various machine learning techniques in predicting stock market directions. [16] proposes a deep LSTM with an embedding layer and a LSTM neural network with an autoencoder to predict the stock market. Bidirectional LSTM (BiLSTM) models enhance traditional LSTMs by processing data bidirectionally, thus boosting context awareness [17][18][19]. Hybrid models are often compared with standalone or other hybrid models to evaluate their effectiveness: GRU-LSTM hybrids surpass traditional LSTM models in accuracy and processing speed, crucial for real-time financial forecasting [20], since it can enhance learning efficiency and speed by addressing the vanishing gradient problem and preserving long-term dependency capabilities [21][22].

Attention mechanisms have revolutionized the field of DL, offering a means to enhance model interpretability and focus on relevant features in large datasets [23]. [24] establish the foundation for many modern attention-based architectures. Furthermore, [25] adapt attention mechanisms for financial applications, creating a hybrid attention network that leverages news patterns to predict stock price trends effectively. The attention-based LSTM model has been particularly fruitful for sequential data analysis [26][27]. For instance, [28] applies Self-attention to enhance trading signal generation, indicating the robustness of attention in complex decision-making scenarios. The efficacy of attention mechanisms extends beyond traditional models to more complex hybrid architectures in various fields. [11] find that attention-enhanced RNN models, such as attention-based LSTM, GRU, and GRU-LSTM, significantly outperform traditional machine learning approaches in business applications. More attention-based models have been proposed, such as CNN-LSTM-attention, GRU-LSTM-attention, CNN–BiLSTM-GRU-attention [29][30][31].

3. Methodology

In this section, first we explore two DL strategies and then propose three attention-based LSTM hybrid models tailored for stock market price prediction. Additionally, we will outline the metrics utilized for model evaluation.

DOI: 10.54254/2754-1169/104/2024ED0152

3.1. LSTM

LSTM has been widely used in a variety of challenges and has yielded impressive outcomes. LSTM memory cell consists of three parts: the forget gate, the input gate, and the output gate. The computational methodology of the LSTM is outlined sequentially in the Figure 1:

1. Inputs from the previous time step (h_{t-1}) and current input (x_t) are processed by the forget gate to generate its output (f_t) as:
$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f), \tag{1}$$
where f_t resides within the interval (0,1). The terms W_f and b_f represent the weight matrix and bias vector of the forget gate, respectively.
2. The input gate (i_t) and candidate cell state ($\tilde{C}_t$) are determined from the input of the last (h_{t-1}) and current (x_t) time step via:
$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \tag{2}$$
$$\tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_c), \tag{3}$$
with i_t also constrained to the interval (0,1). Here, W_i and W_C denote the weight matrices for the input gate and candidate cell state, and b_i and b_c are their respective biases.
3. The cell state at the current time step (C_t) is updated by:
$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t, \tag{4}$$
where $\odot$ denotes the Hadamard multiplication, maintaining C_t in the range (0,1).
4. The output gate's activation (o_t) is computed from the previous output (h_{t-1}) and current input (x_t), given by:
$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o), \tag{5}$$
with the value of o_t bound between 0 and 1, and where W_o and b_o signify the weight matrix and bias for the output gate.
5. Finally, the LSTM's output (h_t) is derived by coupling the output gate's activation with the current cell state:
$$h_t = o_t \odot \tanh(C_t). \tag{6}$$

3.2. Self-attention mechanism

The Self-attention mechanism operates based on three fundamental parameters: key (K), query (Q), and value (V). Initially, the embedding dimension is determined across the input data series, followed by the construction of weight parameter matrices (W). The values of K, Q, and V are derived by multiplying the input signal with the W matrix, as detailed in Equation 7.

$$\text{attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}V\right) \tag{7}$$

where Q and K are multiplied and subsequently normalized by d_k, the dimensionality of the key vectors. Post-normalization, the V values are employed to scale the result, which is then processed through a softmax function to compute the attention scores. This method is recognized as the scaled-dot product attention mechanism.

Multi-attention integrates multiple Self-attention processes through the deployment of several attention heads. Each attention head is equipped with distinct sets of parameters for Q, K, and V, allowing the model to concurrently attend to various segments of the sequence.

DOI: 10.54254/2754-1169/104/2024ED0152

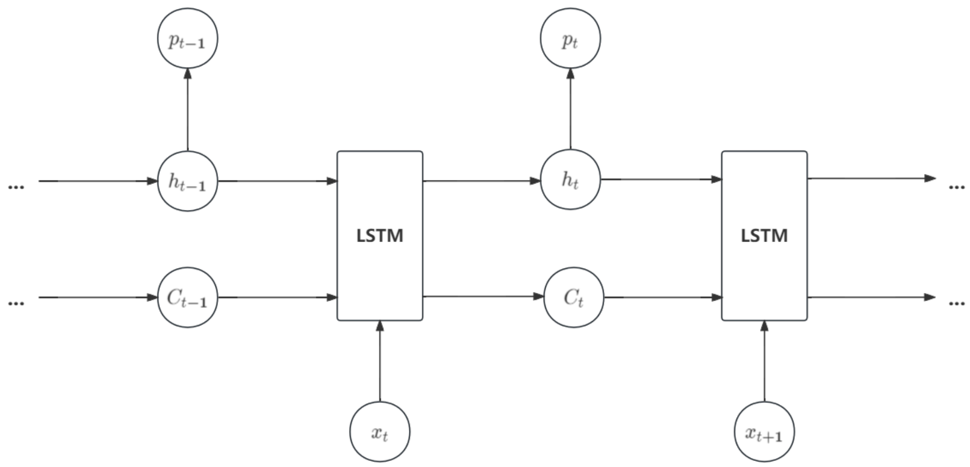

Figure 1: LSTM structure

3.3. CNN-LSTM-attention

Different from the standalone LSTM model, the CNN-LSTM-attention model begins by reformatting the input data into a three-dimensional array to facilitate convolutional operations. This preprocessing allows each data subsequence to be individually analyzed and transformed, which is crucial for capturing local and temporal features effectively. The subsequent steps detail the complete workflow of this model in the Figure 2:

1. Data Standardization: Normalize the distribution of closing price data by employing preprocessing techniques for scaling. Reshape the input data X to fit the model's input structure.
2. Network Initialization: Set initial weights and biases for this model architecture.
3. CNN Layer Operation: Process input data through convolutional and pooling layers to extract features and obtain initial output values.
4. LSTM Layer Operation: Compute output values from the CNN output through the LSTM layer. Apply 1D convolutions on data subsequences with `TimeDistributed` layers for local temporal feature extraction. Implement a max pooling layer to reduce the dimensionality of features. Incorporate a flatten layer to prepare the data for LSTM processing.
5. Incorporating Attention: Introduce a `MultiHeadAttention` layer after the initial LSTM processing to refine feature representation through Self-attention mechanisms. This layer allows the model to focus on important features by assigning higher weights to more relevant data points, enhancing the predictive accuracy of the network.
6. Output Layer Computation: Include dropout layers to mitigate overfitting. Finalize the architecture with a dense layer for output prediction. Input the LSTM layer output into a fully connected layer to yield the final output values.
7. Calculation Error: Compare output layer results with actual data to determine error. Store the trained the model for future forecasting.
8. Forecasting Input Acquisition: Gather input data for forecast prediction. Enter standardized data into the model to retrieve forecasted output.
9. Data Restoration: Apply inverse scaling to predictions and actual targets to revert to the original data scale.

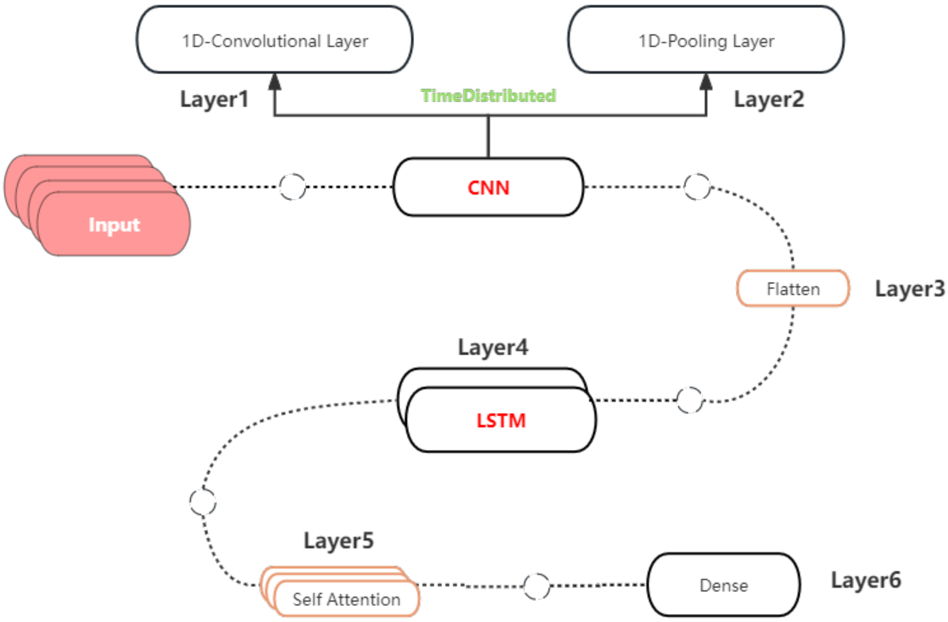

Figure 2: CNN-LSTM-attention framework

3.4. GRU-LSTM-attention

Distinct from a conventional single LSTM model, this sophisticated configuration utilizes a sequence of LSTM layers followed by GRU layers, designed to harness the strengths of both architectures effectively. The subsequent steps show the whole workflow of this model in the Figure 3:

1. Data Preprocessing and Transformation: Normalize the closing price data by employing preprocessing techniques. Generate supervised learning sequences with the `split_sequences` function.
2. Data Splitting: Partition the data into training, validation, and test sets.
3. Model Definition and Architecture: Construct a sequential model with LSTM and GRU layers, culminating in a dense output layer. A `MultiHeadAttention` layer is added between the LSTM and GRU layer.
4. Model Compilation: Compile the model using the optimizer and MSE loss, monitoring RMSE.
5. Inverse Scaling of Predictions: Apply inverse scaling to the predictions and the targets using the earlier fitted preprocessing function.

3.5. CNN-BiLSTM-GRU-attention

The architecture of this CNN-BiLSTM-GRU-attention model is depicted in the Figure 4, illustrating the sequential and functional integration of these diverse but complementary neural network layers:

1. Convolutional Layer: Utilizes one-dimensional (1D) convolution with filters to extract shallow features from the input data.
2. Max-Pooling Layer: Applies max-pooling to reduce the spatial dimensions of the feature maps.
3. Flatten Layer: Converts the multi-dimensional array of features into a one-dimensional array.
4. Bidirectional LSTM Layer: Employs a bidirectional LSTM that processes data in both forward and reverse directions to capture better sequential dependencies.

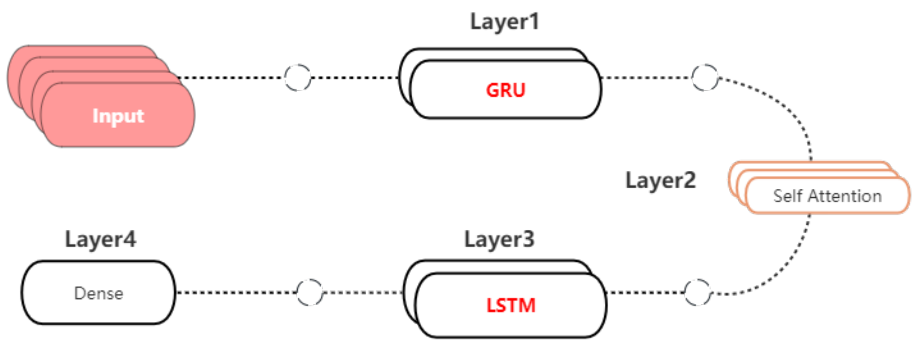

Figure 3: GRU-LSTM-attention framework

5. GRU Layer: Includes a GRU layer which simplifies the gating mechanism but effectively captures temporal dependencies.
6. Attention Layer: Integrate a `MultiHeadAttention` layer, which allows the model to focus on the most informative parts of the input sequence, improving the specificity of the model by attending to crucial temporal dynamics.
7. Dense Layer: A dense layer that transforms the aggregated features into the final prediction output.

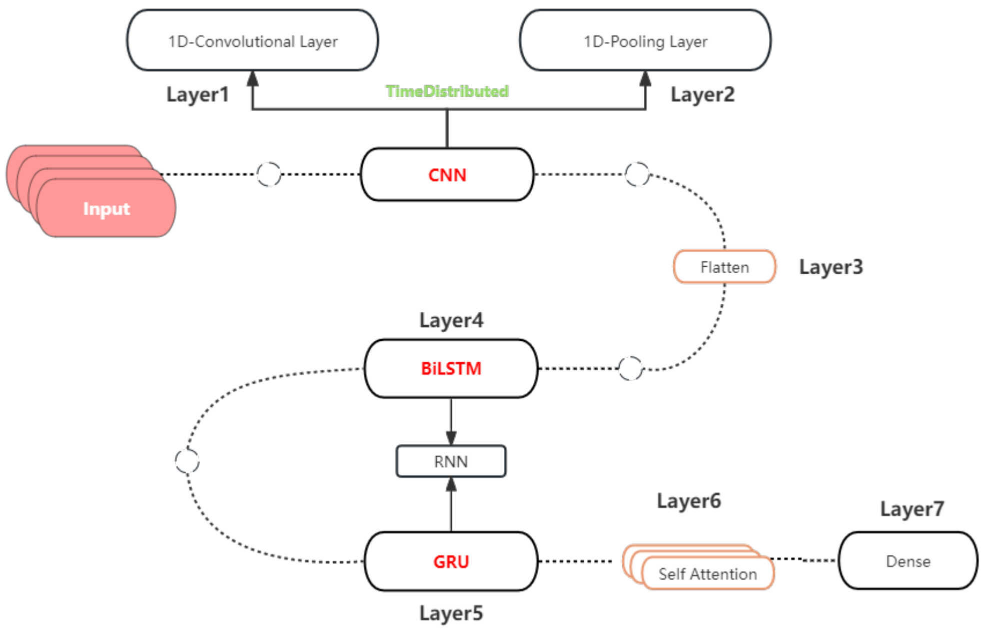

Figure 4: CNN-BiLSTM-GRU-attention Framework

BiLSTM is achieved by combining a forward-processing LSTM layer with a backward-processing counterpart, given by:

$$A_i = f_1(\omega_1 x_i + \omega_2 A_{i-1}), \tag{8}$$

$$B_i = f_2(\omega_3 x_i + \omega_4 B_{i+1}), \tag{9}$$

$$h_i = f_3(\omega_5 A_i + \omega_6 B_i), \tag{10}$$

where A_i represents the output of the forward-processing LSTM layer at time step i, B_i denotes the output of the backward-processing LSTM layer at time step i, and Y_i signifies the final output of the BiLSTM at time step i. The functions f_1, f_2, and f_3 are the activation functions for the forward, backward, and output layers, respectively. The weights $\omega_k, k = 1, ...6$ correspond to the connections within the BiLSTM network that modulate the input and recurrent connections across the layers.

3.6. Evaluation metrics

To evaluate the performance of the models, two commonly used statistical metrics are employed: root mean square error (RMSE), and R-squared (R^2).

RMSE is a standard metric used to measure the accuracy of predicted values. It quantifies the average magnitude of errors by calculating the square root of the average squared differences between the forecasted and observed values. This metric is particularly useful in contexts where it is necessary to penalize larger errors more severely. The RMSE formula is expressed as:

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

where $\hat{y}_i$ represents the forecasted value, y_i denotes the actual observed value, and n is the number of observations. RMSE emphasizes the significance of large prediction errors, making it particularly suitable for the stock prediction with large data volume. While an RMSE value of value represents an ideal error-free model, which is rarely achieved in practice. Lower values indicate better performance. If RMSE $\rightarrow 0$, it indicates that the model is perfectly fitting the observations.

Furthermore, R^2 statistic, also known as the coefficient of determination, is a measure used to ascertain the proportion of variance for a dependent variable that's predicted from the independent variables. It serves as an indicator of the fit quality of the model. The R^2 is calculated by:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, \tag{12}$$

where $\hat{y}_i$ represents the predicted value, y_i is the actual observed value, and $\bar{y}$ denotes the mean of the observed values. A high R^2 value indicates that the model's predictions closely correspond to the actual observations. However, it is essential to use R^2 in conjunction with other metrics, such as RMSE, for a more nuanced evaluation of the model's predictive performance, as R^2 alone does not convey the magnitude of prediction errors.

4. Experiments Analysis

4.1. Data description

This study utilizes a dataset comprising the daily closing prices of Amazon.com, Inc. (NASDAQ: AMZN) from April 25, 2014, to April 25, 2024, sourced from Yahoo Finance [1]. The dataset contains exactly 2,518 entries, corresponding to the total number of trading days within the ten-year interval, excluding weekends and public holidays when the stock market is closed. Each data point in the

[1] https://uk.finance.yahoo.com/quote/AMZN?.tsrc=fin-srch

dataset represents the closing price of Amazon's stock, recorded at the market's close, typically at 4:00 PM EST. The prices are denoted in U.S. dollars and reflect the market's valuation of Amazon shares at the end of each trading day.

4.2. Model setup

Five DL models: LSTM, Self-attention, CNN-LSTM-attention, GRU-LSTM-attention, CNN-BiLSTM-GRU-attention are developed for forecasting financial time series data, specifically aiming at predicting closing stock prices. The modeling initiative begins by converting sequence data into a form suitable for supervised learning. The preprocessing stage involves the normalization of feature data using the `PowerTransformer` technique, which stabilizes variance and more accurately approximates a normal distribution, which generally enhances the efficacy of numerous machine learning algorithms. Following the prediction phase, both the forecasted and actual values undergo reverse transformation to revert to their original scales, thereby enhancing their interpretability.

In this study, `GridSearchCV` is employed to fine-tune the hyperparameters of four DL models tailored for forecasting financial time series data. This approach systematically explores combinations of model layers, units, epochs, batch sizes, and optimizers to pinpoint the optimal settings. Specifically, we set the following parameters: layers [2, 4, 6], units [40, 50], batch sizes [32, 64, 128], epochs [200, 400, 500], and optimizers [RMSprop, Adam].

In particularly, the LSTM model is designed with a single LSTM layer and utilizes a `sigmoid` activation function to capture the data's nonlinear dynamics. In the CNN-LSTM-attention model, the convolutional layer equipped with 32 filters and a kernel size of 2 employs the `relu` activation function are used. In the LSTM layer, `relu` activation is used, diverging from the activation function used in the single-layer LSTM model, to effectively handle the enriched input from the CNN layers. In the GRU-LSTM-attention model, two LSTM layers are configured with a `sigmoid` activation function, and GRU layers are also integrated which share similarities with LSTMs. In the hybrid CNN-BiLSTM-GRU-attention model, a convolutional layer with 32 filters and a kernel size of 2 utilizes the `relu` activation function. In contrast to the CNN-LSTM-attention model, it is followed by a BiLSTM layer with a `sigmoid` activation function. Subsequently, a GRU layer with a `sigmoid` activation function is employed.

4.3. Data analysis

Figure 5 shows the prediction generated by the LSTM model. We note a general alignment between projected trends and actual stock closing prices, with a tendency for forecast values to exceed actual values. In particular, the precision of predictions at specific points, especially extremes, is suboptimal, with forecasts consistently overestimating. Additionally, the model consistently overestimates following peak values, anticipating continued upward trends despite observed declines, indicating systematic overprediction at turning points. Figure 6 shows that the overall predictive performance of the Self-attention model is commendable. While its predictions consistently forecast estimates slightly below the actual values, the model exhibits enhanced accuracy particularly during pivotal trend transitions, effectively minimizing significant errors in trend reversal.

Figure 7 shows that the CNN-LSTM-attention model generally predicts values that are higher than the actual values. However, its prediction performance closely matches the actual values, with small errors observed between the blue and red lines. Figure 8 indicates that the GRU-LSTM-attention model performs reasonably well in predicting the values, with the predicted values being slightly lower than the actual values. Figure 9 illustrates the exceptional performance of the CNN-BiLSTM-

DOI: 10.54254/2754-1169/104/2024ED0152

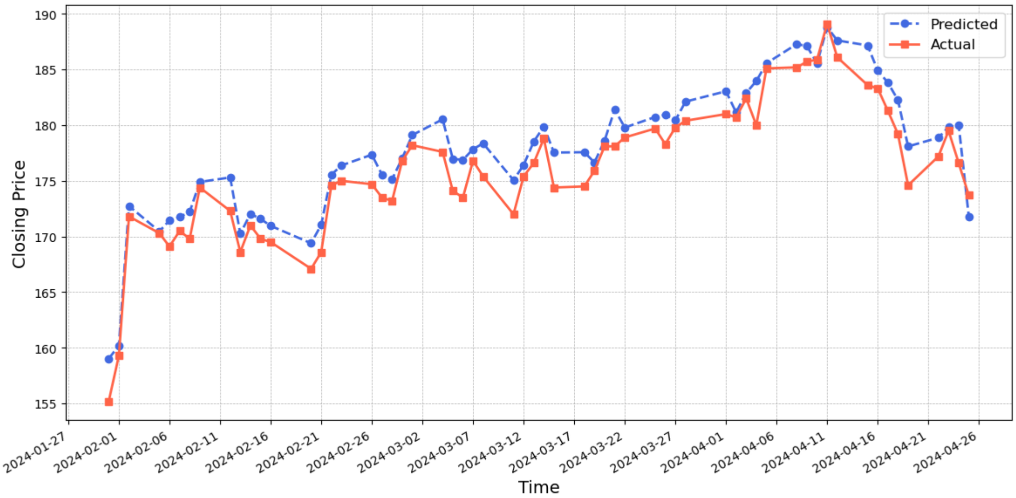

Figure 5: The actual stock prices and prediction generated by LSTM model. The solid red line represents the true value, and the dashed blue line represents the predicted value.

GRU-attention model. The predicted values closely match the actual values, demonstrating a remarkable level of accuracy. Despite discrepancies in the turning points and the rate of decline during certain time periods, the model's effectiveness in capturing the crucial dynamics of the dataset is affirmed.

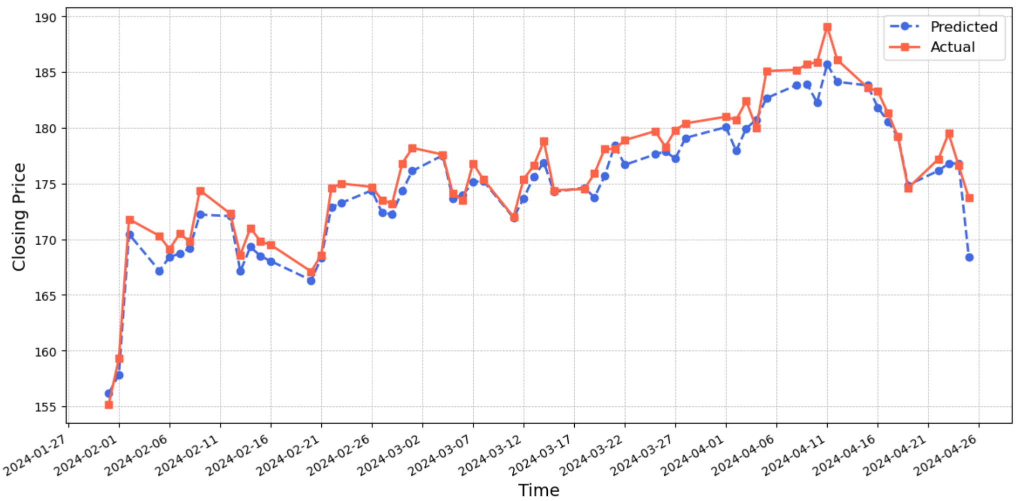

Figure 6: The actual stock prices and prediction generated by Self-attention model. The solid red line represents the true value, and the dashed blue line represents the predicted value.

In conclusion, among the five models evaluated, each demonstrates unique strengths and weaknesses. The the LSTM model serves as a foundational baseline, capturing overall trends but often overestimating actual values. The Self-attention model, conversely, tends to underestimate values, excelling in trend direction accuracy but falling short at peak predictions. The CNN-LSTM-attention model enhances precision, particularly within certain periods, though it shares the LSTM's challenge in turning point prediction. The GRU-LSTM-attention offers a compromise with better initial accuracy than LSTM and CNN-LSTM-attention models. The hybrid CNN-BiLSTM-GRU-attention model

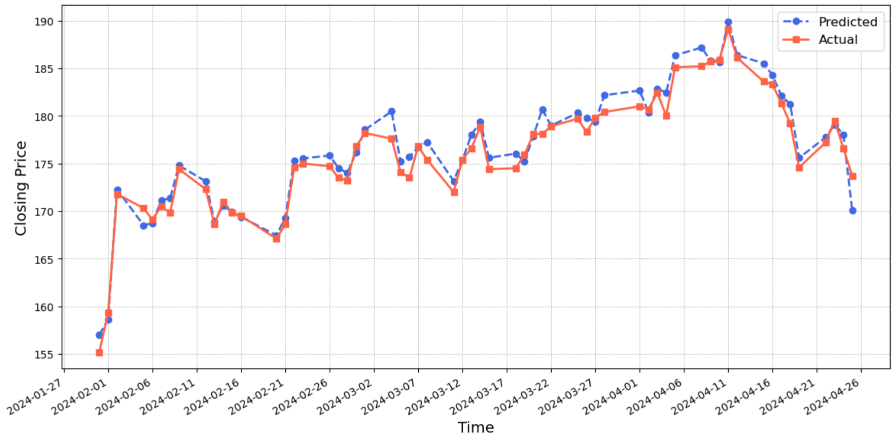

Figure 7: The actual stock prices and prediction generated by CNN-LSTM-attention model. The solid red line represents the true value, and the dashed blue line represents the predicted value.

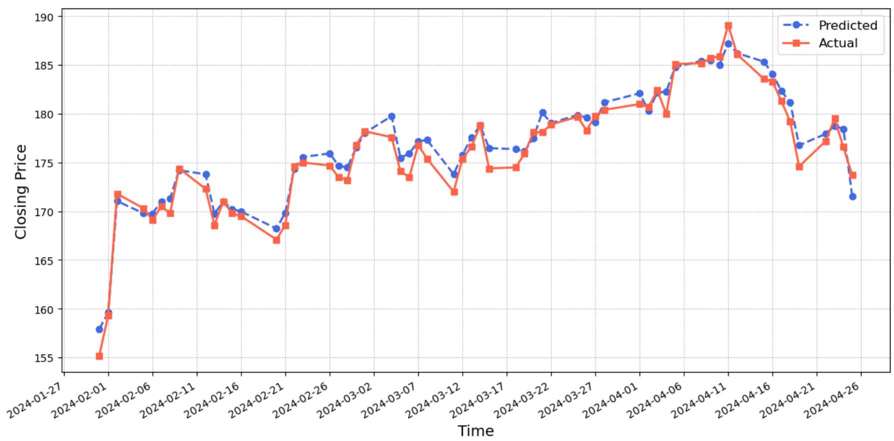

Figure 8: The actual stock prices and prediction generated by GRU-LSTM-attention model. The solid red line represents the true value, and the dashed blue line represents the predicted value.

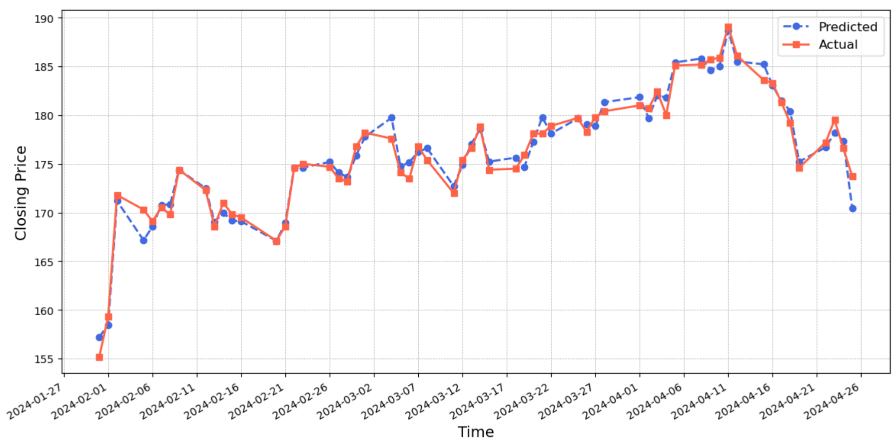

Figure 9: The actual stock prices and prediction generated by CNN-BiLSTM-GRU-attention model. The solid red line represents the true value, and the dashed blue line represents the predicted value.

DOI: 10.54254/2754-1169/104/2024ED0152

demonstrates the best overall performance, adeptly aligning its predictions with actual stock prices across multiple periods.

4.3.1. Evaluation metrics

Table 1: Evaluation Metrics and CPU Times for Models

Model	RMSE	R^2	CPU Times (s)
LSTM	2.103577	0.881128	**7.31**
Self-attention	1.754763	0.917282	16.5
CNN-LSTM-attention	1.263602	0.957107	35.8
GRU-LSTM-attention	1.243342	0.958472	36.4
CNN-BiLSTM-GRU-attention	**1.054589**	**0.970123**	59.2

Table 1 shows the evaluation metrics (RMSE & R^2) and CPU Times for the models. It provides an assessment of the efficacy of five DL models, alongside their computational efficiency as measured by CPU times. Among the models assessed, the CNN-BiLSTM-GRU-attention model outperforms others with the lowest RMSE value of 1.054589 and the highest R^2 value of 0.970123, indicating superior predictive accuracy and goodness-of-fit. However, this model requires the longest CPU time of 59.2 seconds. In contrast, the LSTM model demonstrates the shortest CPU time of 7.31 seconds but with highest RMSE and lowest R^2 values. Notably, the LSTM and attention-based models, CNN-LSTM-attention and GRU-LSTM-attention, outperform the standalone LSTM and Self-attention models. Overall, the CNN-BiLSTM-GRU-attention model emerges as the top performer in terms of predictive accuracy and goodness-of-fit, albeit at the expense of computational efficiency.

5. Conclusion

This paper delves into the challenging endeavor of forecasting stock market trends through the utilization of various DL models. A comparative analysis was undertaken, leveraging five distinct models-LSTM, Self-attention, CNN-LSTM-attention, GRU-LSTM-attention, and CNN-BiLSTM-GRU-attention to assess their performance in predicting Amazon stock prices using historical data spanning the past decade.

The evaluation, based on RMSE and R^2, demonstrates that the CNN-BiLSTM-GRU-attention model outperforms other with the highest accuracy with an R^2 value of 0.970123 and the lowest RMSE of 1.054589. While this model exhibits superior predictive capabilities, it also requires the longest processing time. Overall, there's a trade-off between accuracy and computational resources. Models, such as CNN-LSTM-attention, GRU-LSTM-attention, and CNN-BiLSTM-GRU-attention, which offer superior predictive capabilities, demand longer CPU times, indicating higher computational complexity. Conversely, models with faster execution times, such as LSTM, compromise on accuracy to some extent. Hence, the selection of a model may depend on the specific requirements of the forecasting task, considering acceptable thresholds for accuracy and available computational resources. In scenarios where there are no constraints on these conditions, the hybrid CNN-BiLSTM-GRU-attention model emerges as the preferred choice for stock price prediction.

There are several directions for future work. One direction is to consider a wider spectrum of stocks or indices, such as the S&P 500, which could enhance the robustness of predictions. Extending the analysis over a longer timeframe, possibly spanning ten years, could capture a more diverse range of market conditions and trends. Moreover, exploring other hybrid models, may uncover configurations that further improve accuracy while also optimizing computational time. The goal would be to develop a model that efficiently balances speed and precision. Lastly, investigating the adaptability of the CNN-BiLSTM-GRU-attention model in other domains, such as commodity pricing or cryptocurrency markets, could provide valuable insights into its effectiveness across different financial instruments. Integrating these models with real-time data streams and trading platforms has the potential to develop automated trading systems capable of promptly responding to market changes, thus enhancing the practical relevance of the research findings.

References

[1] Ahmad Chaddad, Jihao Peng, Jian Xu, and Ahmed Bouridane. Survey of explainable ai techniques in healthcare. *Sensors*, 23(2):634, 2023.

[2] Yogesh K Dwivedi, Laurie Hughes, Elvira Ismagilova, Gert Aarts, Crispin Coombs, Tom Crick, Yanqing Duan, Rohita Dwivedi, John Edwards, Aled Eirug, et al. Artificial intelligence (ai): Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy. *International Journal of Information Management*, 57:101994, 2021.

[3] Yifang Ma, Zhenyu Wang, Hong Yang, and Lin Yang. Artificial intelligence applications in the development of autonomous vehicles: A survey. *IEEE/CAA Journal of Automatica Sinica*, 7(2):315–329, 2020.

[4] Xinyi Huang, Di Zou, Gary Cheng, Xieling Chen, and Haoran Xie. Trends, research issues and applications of artificial intelligence in language education. *Educational Technology & Society*, 26(1):112–131, 2023.

[5] Ivano Lauriola, Alberto Lavelli, and Fabio Aiolli. An introduction to deep learning in natural language processing: Models, techniques, and tools. *Neurocomputing*, 470:443–456, 2022.

[6] Yingjun Chen and Yongtao Hao. A feature weighted support vector machine and k-nearest neighbor algorithm for stock market indices prediction. *Expert Systems with Applications*, 80:340–355, 2017.

[7] Wen Long, Zhichen Lu, and Lingxiao Cui. Deep learning-based feature engineering for stock price movement prediction. *Knowledge-Based Systems*, 164:163–173, 2019.

[8] Hongduo Cao, Tiantian Lin, Ying Li, Hanyu Zhang, et al. Stock price pattern prediction based on complex network and machine learning. *Complexity*, 2019, 2019.

[9] Omer Berat Sezer, Mehmet Ugur Gudelek, and Ahmet Murat Ozbayoglu. Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied soft computing*, 90:106181, 2020.

[10] Liheng Zhang, Charu Aggarwal, and Guo-Jun Qi. Stock price prediction via discovering multi-frequency trading patterns. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 2141–2149, 2017.

[11] Abdus Saboor, Arif Hussain, Bless Lord Y Agbley, Jian Ping Li, Rajesh Kumar, et al. Stock market index prediction using machine learning and deep learning techniques. *Intelligent Automation & Soft Computing*, 37(2), 2023.

[12] Chenyang Qi, Jiaying Ren, and Jin Su. Gru neural network based on ceemdan–wavelet for stock price prediction. *Applied Sciences*, 13(12):7104, 2023.

[13] Anita Yadav, CK Jha, and Aditi Sharan. Optimizing lstm for time series prediction in indian stock market. *Procedia Computer Science*, 167:2091–2100, 2020.

DOI: 10.54254/2754-1169/104/2024ED0152

[14] Ha Young Kim and Chang Hyun Won. Forecasting the volatility of stock price index: A hybrid model integrating lstm with multiple garch-type models. *Expert Systems with Applications*, 103:25–37, 2018.

[15] Thomas Fischer and Christopher Krauss. Deep learning with long short-term memory networks for financial market predictions. *European journal of operational research*, 270(2):654–669, 2018.

[16] Xiongwen Pang, Yanqiang Zhou, Pan Wang, Weiwei Lin, and Victor Chang. An innovative neural network approach for stock market prediction. *The Journal of Supercomputing*, 76:2098–2118, 2020.

[17] Wenjie Lu, Jiazheng Li, Jingyang Wang, and Lele Qin. A cnn-bilstm-am method for stock price prediction. *Neural Computing and Applications*, 33(10):4741–4753, 2021.

[18] Jaiwin Shah, Rishabh Jain, Vedant Jolly, and Anand Godbole. Stock market prediction using bi-directional lstm. In *2021 International Conference on Communication information and Computing Technology (ICCICT)*, pages 1–5. IEEE, 2021.

[19] Jilin Zhang, Lishi Ye, and Yongzeng Lai. Stock price prediction using cnn-bilstm-attention model. *Mathematics*, 11(9):1985, 2023.

[20] Nrusingha Tripathy, Surabi Parida, and Subrat Kumar Nayak. Forecasting stock market indices using gated recurrent unit (gru) based ensemble models: Lstm-gru. *International Journal of Computer and Communication Technology*, 9(1), 2023.

[21] Krzysztof Zarzycki and Maciej Ławryńczuk. Advanced predictive control for gru and lstm networks. *Information Sciences*, 616:229–254, 2022.

[22] I Sibel Kervanci, M Fatih Akay, and Eren Özceylan. Bitcoin price prediction using lstm, gru and hybrid lstm-gru with bayesian optimization, random search, and grid search for the next days. *Journal of Industrial and Management Optimization*, 20(2):570–588, 2024.

[23] Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter Stewart. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. *Advances in neural information processing systems*, 29, 2016.

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[25] Ziniu Hu, Weiqing Liu, Jiang Bian, Xuanzhe Liu, and Tie-Yan Liu. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pages 261–269, 2018.

[26] Xiao Teng, Xiang Zhang, and Zhigang Luo. Multi-scale local cues and hierarchical attention-based lstm for stock price trend prediction. *Neurocomputing*, 505:92–100, 2022.

[27] Jujie Wang, Quan Cui, Xin Sun, and Maolin He. Asian stock markets closing index forecast based on secondary decomposition, multi-factor analysis and attention-based lstm model. *Engineering Applications of Artificial Intelligence*, 113:104908, 2022.

[28] Dongkyu Kwak, Sungyoon Choi, and Woojin Chang. Self-attention based deep direct recurrent reinforcement learning with hybrid loss for trading signal generation. *Information Sciences*, 623:592–606, 2023.

[29] Jiaqi Qu, Zheng Qian, and Yan Pei. Day-ahead hourly photovoltaic power forecasting using attention-based cnn-lstm neural network embedded with multiple relevant and target variables prediction pattern. *Energy*, 232:120996, 2021.

[30] Hafiz Shahbaz Munir, Shengbing Ren, Mubashar Mustafa, Chaudry Naeem Siddique, and Shazib Qayyum. Attention based gru-lstm for software defect prediction. *Plos one*, 16(3):e0247444, 2021.

[31] Muharrem Çelebi, Sıtkı Öztürk, and Kaplan Kaplan. An emotion recognition method based on ewt-3d–cnn–bilstm-gru-at model. *Computers in Biology and Medicine*, 169:107954, 2024.