



ARTICLE

# Research on Stock Price Prediction Method Based on the GAN-LSTM-Attention Model

Peng Li, Yanrui Wei and Lili Yin\*

College of Computer Science and Technology, Harbin University of Science and Technology, Harbin, 150080, China

\*Corresponding Author: Lili Yin. Email: yinlili234@126.com

Received: 27 July 2024 Accepted: 25 September 2024 Published: 03 January 2025

## ABSTRACT

Stock price prediction is a typical complex time series prediction problem characterized by dynamics, nonlinearity, and complexity. This paper introduces a generative adversarial network model that incorporates an attention mechanism (GAN-LSTM-Attention) to improve the accuracy of stock price prediction. Firstly, the generator of this model combines the Long and Short-Term Memory Network (LSTM), the Attention Mechanism and, the Fully-Connected Layer, focusing on generating the predicted stock price. The discriminator combines the Convolutional Neural Network (CNN) and the Fully-Connected Layer to discriminate between real stock prices and generated stock prices. Secondly, to evaluate the practical application ability and generalization ability of the GAN-LSTM-Attention model, four representative stocks in the United States of America (USA) stock market, namely, Standard & Poor's 500 Index stock, Apple Incorporated stock, Advanced Micro Devices Incorporated stock, and Google Incorporated stock were selected for prediction experiments, and the prediction performance was comprehensively evaluated by using the three evaluation metrics, namely, mean absolute error (MAE), root mean square error (RMSE), and coefficient of determination ( $R^2$ ). Finally, the specific effects of the attention mechanism, convolutional layer, and fully-connected layer on the prediction performance of the model are systematically analyzed through ablation study. The results of experiment show that the GAN-LSTM-Attention model exhibits excellent performance and robustness in stock price prediction.

## KEYWORDS

Stock price prediction; generative adversarial network; attention mechanism; time-series prediction

## 1 Introduction

With the rapid development of global financial markets and the increase in stock trading activities, stock price forecasting has become one of the key focuses of economic research [1]. Effective stock price forecasting can significantly improve investment returns and reduce financial risks, and this forecasting ability not only helps to alleviate market volatility, moreover, it bolsters the confidence of market participants and fosters the overall stability and healthy development of financial markets [2]. In addition, accurate stock price forecasting plays an important role in optimizing the allocation of resources and guiding the flow of capital, thus promoting the improvement of economic efficiency. Therefore, the mastery and application of advanced forecasting technology are crucial for maintaining



the long-term stability and growth of the economy [3]. In the computer field, stock price forecasting is a typical time series analysis problem, which is challenging and practical [4]. This problem involves the application of computational methods such as machine learning, data mining, and artificial intelligence to predict future stock price movements [5–7].

Early stock price forecasting methods relied heavily on time series analysis, particularly the AutoRegressive Integrated Moving Average (ARIMA) model and its variants [8,9]. These models are widely used to process and forecast linear time series data, but their effectiveness is limited when confronted with the nonlinear nature of financial markets. To deal with possible nonlinear relationships, researchers introduced Generalized AutoRegressive Conditional Heteroskedasticity (GARCH) models to better capture and predict the volatility of financial markets [10]. With the increase in computational power, machine learning techniques started to be applied in stock price prediction. In particular, techniques such as Decision Trees and Support Vector Machines (SVM) have gained attention for their ability to handle nonlinear data [11,12]. In addition, integrated learning such as Random Forest (RF) and Gradient Boosting Tree (GBT) have been widely used to predict stock prices due to their high accuracy and robustness [13–15].

Recently, deep learning models have become cutting-edge methods for stock market prediction due to their advantages in big data processing and feature learning. In particular, Recurrent Neural Networks (RNN) and their variants such as Long Short-Term Memory Networks (LSTM) are capable of efficiently dealing with temporal dependencies in time-series data [16,17]. In 2020, Lu et al. proposed a hybrid CNN-LSTM model [18]. The model first uses CNN to extract the temporal features of various historical data including closing price, and then in using LSTM to predict stock price data. This prediction method not only provides a new research idea for stock price prediction, but also essentially improves the accuracy of stock prediction. Chen et al. proposed a hybrid deep learning model based on Long and Short-Term Memory Networks (LSTM), Multilayer Perceptron, and Attention Mechanism [19]. Firstly, a multilayer perceptron was utilized to quickly transform the feature space and perform fast gradient descent. Subsequently, a Bidirectional Long Short-Term Memory Neural Network (BiLSTM) is utilized to extract temporal features of stock time series data. Finally, the attention mechanism is utilized to make the neural network pay more attention to the critical temporal information by assigning higher weights. The model demonstrates that the attention mechanism is particularly important in the field of stock price prediction, and the model leads to a dramatic improvement in prediction accuracy.

In addition, emerging self-supervised learning methods and Generative Adversarial Networks (GAN) are beginning to be explored for enhancing the model's generalization ability and prediction accuracy [20,21]. Generative Adversarial Networks is an innovative machine-learning framework proposed by Goodfellow et al. in 2014. The core of this framework is the estimation of generative models through an adversarial process involving the simultaneous training of two models: the Generator (G) and the Discriminator (D). The Generator's task is to capture the distribution of the training data so that it can generate new data points that are as close as possible to the real data, while the Discriminator's task is to evaluate the veracity of the samples and distinguish between the fake data generated by the Generator and the actual real data. During the training process, the generator and the discriminator present a dynamic adversarial relationship: the generator continuously learns how to better mimic the data distribution to deceive the discriminator. While the discriminator continuously improves its ability to recognize generated data from real data. This adversarial process leads to a continuous improvement in the capabilities of both models until an equilibrium is reached. Despite the success of the above methods in stock price prediction, they still face some important challenges in practical applications. Firstly, it is often difficult for a single prediction model to adequately capture the

complexity and variability of stock market data, which limits its prediction accuracy and generalization ability [22]. In addition, traditional deep learning models such as RNN and LSTM, while capable of handling time series dependencies, still fall short in capturing long-term dependencies and focusing on key information [23]. Meanwhile, although integrated learning methods improve the stability of forecasting, the responsiveness and flexibility to new information in dynamic market environments still need to be improved [24]. To address these issues and further improve the predictive performance of the model, a new hybrid GAN-LSTM-Attention model is used in this paper. The model combines the powerful data generation capability of generative adversarial networks and the time series analysis advantage of LSTM, while introducing the attention mechanism to improve the focus and accuracy of prediction. In this way, this paper expects that the model can not only effectively deal with the long-term dependence problem in time series data, but also more accurately identify and focus on the key information in the forecasting process, to achieve more accurate and robust stock price forecasts in the complex and changing financial market environment.

## 2 Methodology

The GAN-LSTM-Attention model forms a comprehensive model that performs well in stock price prediction by effectively integrating the time series processing ability of LSTM, the key feature focusing ability of the attention mechanism, the local feature extraction ability of CNN, and the adversarial training mechanism of GAN. The model is not only able to accurately capture the dynamic changes and long-term dependencies of the market, but also able to significantly improve the accuracy and stability of the prediction through an automated feature selection mechanism. In addition, the multilayer design and adversarial training of the model make it more robust and generalizable in dealing with market noise and nonlinear features, thus providing more reliable support for investment decision-making and risk management. These features enable the model proposed in this paper to show excellent performance in the complex and changing financial market environment, and has a wide range of application prospects.

### 2.1 Model Structure

In this paper, a generative adversarial network model incorporating an attention mechanism is used to radically improve the accuracy and efficiency of stock price prediction. The model incorporates advanced deep learning techniques, including attention mechanisms, Long Short-Term Memory Networks, Convolutional Neural Networks, and Generative Adversarial Networks. With Generative Adversarial Networks as the basic framework, Long Short-Term Memory Networks and Attention Mechanism, as well as Fully Connected Layers, are used as the generators of the model, and Convolutional Neural Networks and Fully Connected Networks are used as the discriminators of the model.

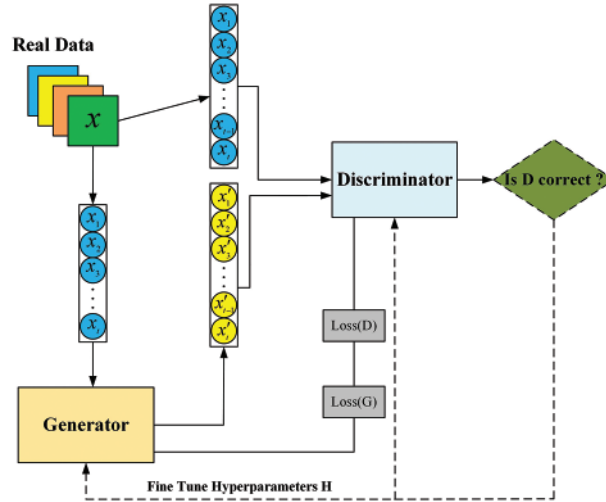
### 2.2 Stock Price Prediction Method Based on GAN-LSTM-Attention

This paper utilizes a generative adversarial network model (GAN-LSTM-Attention) with an incorporated attention mechanism, aiming to enhance the accuracy of stock price prediction. The model consists of two main parts: a generator and a discriminator. The preprocessed stock data is fed into the generator, which outputs the predicted close price of the stock. The discriminator's task is to distinguish whether the input data is generated by the generator or real stock data. The predicted data output by the generator and the real stock data of the day are jointly input to the discriminator for authenticity identification. In the training process, a combined loss function for  $\text{Loss}(D)$  and  $\text{Loss}(G)$

is as in Eq. (1).

$$L(D, G) = E_{x \sim p_r(x)}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))] \quad (1)$$

$D(x)$  is the probability that the discriminator assigns to  $x$  being real data, and  $G(z)$  is the fake data generated by the generator based on noise  $z$ .  $E_{x \sim p_r(x)}$  is the expected value obtained by sampling from the probability distribution of real data  $x$ .  $E_{z \sim p_z(z)}$  is the expected value obtained by sampling from the probability distribution of noise  $z$ . The training is divided into two phases, discriminator training: firstly, the generator  $G$  is fixed and the discriminator  $D$  is trained to improve its ability to discriminate between real and generated data. Generator training: then fix the discriminator  $D$  and train the generator  $G$ , so that its generated data gradually approximates the distribution of real data. These two steps alternate and are executed in multiple cycles until the data generated by the generator cannot be distinguished in the discriminator's identification, the specific steps of the training process of the GAN-LSTM-Attention model algorithm are given in Algorithm 1. The specific structure of the model and the data flow are shown in detail in Fig. 1.



**Figure 1:** The architecture of the GAN-LSTM-Attention

---

**Algorithm 1:** Stock price prediction algorithm based on GAN-LSTM-Attention method

---

Inputs: Stock data  $x = [x_1, x_2, x_3, \dots, x_i]^T$ , where  $x_i$  is the data vector

Outputs: The prediction stock data  $x' = [x'_1, x'_2, x'_3, \dots, x'_i]^T$ , where  $x'_i$  is the data vector

- 1:  $x$  is given to the LSTM layer
  - 2:  $h_i$  is output by the LSTM layer
  - 3:  $h_i$  is input to the attention layer,  $a_i$  is output
  - 4:  $h_i$  and  $a_i$  are dot-multiplied to output  $y_i$
  - 5:  $y_i$  is given to three fully connected layers
  - 6: Output  $x'_i$
  - 7:  $x_i$  and  $x'_i$  are given to three 1D convolutional layers and three fully connected layers
  - 8: Compute the combined loss function of the generator and discriminator  

$$L(D, G) = E_{x \sim p_r(x)}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$
  - 9: Fine-tune hyperparameters  $H$
- 

(Continued)

**Algorithm 1 (continued)**


---

```

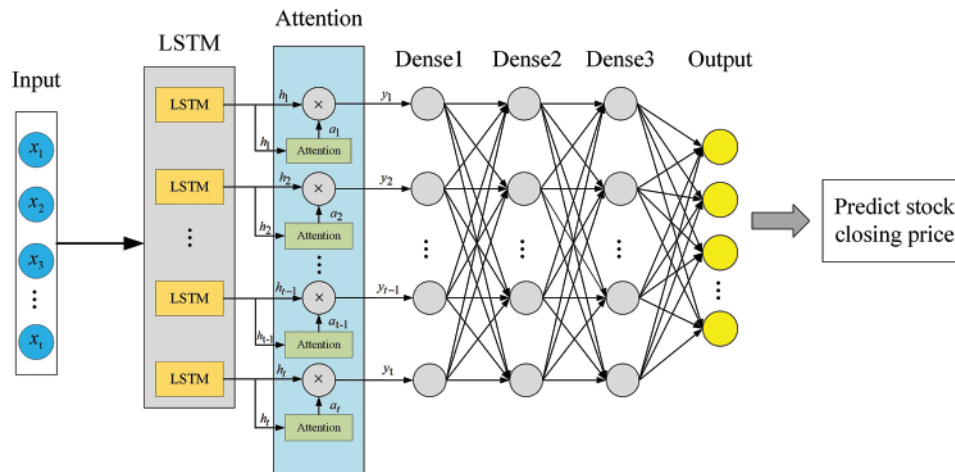
10: Update  $x'$ 
11: End for
12: Compute error rate
13: Return  $x'$ 

```

---

**2.2.1 Generator Model**

The composition of the generator mainly consists of the LSTM layer, the attention layer, and the fully connected layer. The LSTM layer is mainly used to capture long-term dependencies in time series data, which is particularly important for understanding complex patterns in stock prices. Through its unique gating mechanism, LSTM can effectively manage the information flow, retaining important historical information while avoiding the interference of irrelevant data in the past, improving the accuracy and stability of the prediction. The attention layer helps the model focus on the information that is most critical to the prediction results by assigning different weights to different parts of the input data. This mechanism allows the generator to highlight those factors that have the greatest impact on future stock prices when dealing with large amounts of historical data. Finally, the fully connected layer, as the output layer of the generator, integrates the information processed by the previous layers and outputs the final prediction result, namely the predicted closing price of the stock. This structural configuration not only optimizes the information flow through the network, but also enhances the model's ability to learn the key features in the time series data, thus improving the performance and reliability of stock price prediction. The specific structure of the generator is shown in Fig. 2.

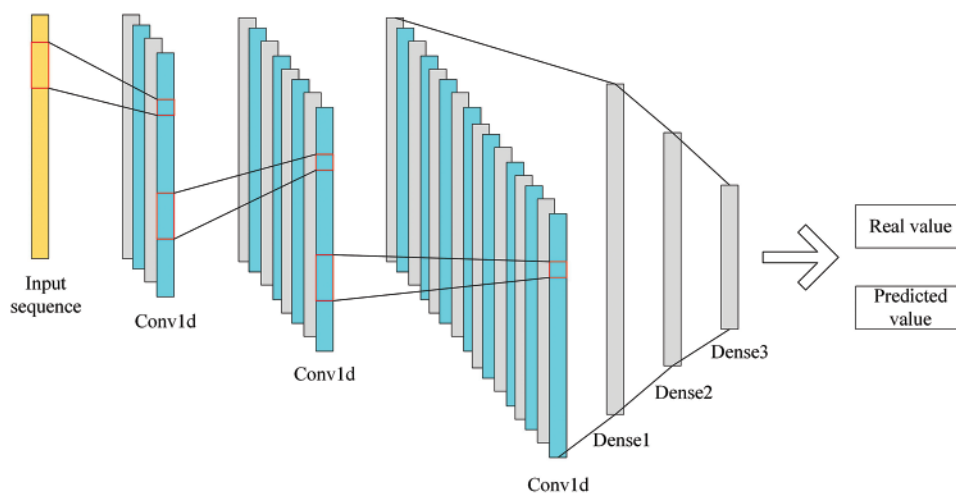


**Figure 2:** The architecture of the generator

**2.2.2 Discriminator Model**

The discriminator consists of three 1D convolutional layers and three fully connected layers, and the activation function is set to ReLU. 1D convolutional layers are specially designed to process time series data by applying one-dimensional convolutional operations to extract local features in the time series data, which is very effective for recognizing complex stock price patterns. 1D convolutional layers are followed by three fully connected layers, which are tasked to combine and map high-level features extracted by convolutional layers to the final output decision, to determine the authenticity

of the input data. The fully connected layers are the key part of the discriminator's output decision, as they transform the learned features into the final classification result, aiming to distinguish whether the input data of the convolutional neural network is real or not. The specific structure of the discriminator is shown in Fig. 3.



**Figure 3:** The architecture of the discriminator

### 3 Experiment

In this paper, 16 key technical indicators were selected as model input variables to ensure the comprehensiveness and representativeness of the input data. To optimize the quality of the inputs to the model, pre-processing operations were performed on the data. To comprehensively assess the performance of the adopted model, the performance of the model is evaluated through three assessment metrics, and the validity of the model is verified through extensive experiments. Further, the superiority of the model adopted in this paper is verified by comparing it with several other models.

#### 3.1 Data Selection and Preprocessing

To construct an effective medium and long-term market forecasting model, the S&P 500 Index data covering the period from 01 January 2012 to 31 December 2023 is selected for this study, and this United States stock dataset is obtained through the Yahoo Finance interface.

Before data analysis, data preprocessing is required to ensure that the model can effectively learn and predict market trends. Firstly, a comprehensive detection and processing of possible missing data problems was carried out. For the detected missing values, linear interpolation was used in this study to fill them in. The linear interpolation method utilizes the linear relationship between the known data points before and after to estimate the missing values, which is more commonly used in financial time series data processing because it can maintain the time continuity and trend of the data. If continuous missing values exceeding a certain threshold are encountered, to avoid adverse effects on model training, we choose to eliminate these data segments to ensure the high quality of model training data. Secondly, to eliminate the influence caused by the difference in the scale, this paper adopts the MinMaxScaler function in the Sklearn library to perform the data normalization process, which makes the price factor and the volume factor indicator in the same order of magnitude, and the process scales



all the numerical features between 0 and 1. The formula for data normalization is shown in [Eq. \(2\)](#).

$$x_{std} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2)$$

$x$  is the original data,  $x_{std}$  is the normalized value,  $x_{min}$  is the minimum value in the original data and  $x_{max}$  is the maximum value in the original data. After normalization, the dataset is divided into training and test data with a ratio of 8:2 to ensure that the model can learn from a sufficiently large amount of data and validate its predictive performance on an independent test set.

Since the original dataset is 2D (samples and features), which is not suitable for neural network training, this study employs a sliding window approach to uplift 2D stock data into 3D. To implement the sliding window technique, this study first determines the size of the window (30 days), a parameter that is set according to the needs of historical data analysis and predictive modeling. Each window contains 30 days of continuous trading data, which are progressively updated by sliding one day forward. Specifically, the first window contains data from Day 1 through Day 30, the second window contains data from Day 2 through Day 31, and so on until the entire data set is covered. During the sliding window process, each new window generates an independent sample that contains the complete set of features, such as opening price, closing price, highest price, and lowest price. In this way, the original 2D data (samples and features) are converted into 3D data (samples, time steps, and features), making it more suitable for time series analysis and neural network training. In addition, to minimize edge effects, especially at the beginning and end of the dataset, we deal with the edge data through an appropriate padding strategy. At the beginning of the dataset, if it is not enough to form a complete window, forward padding (filling the previous gaps with the first day's data) can be used to ensure that each window has the complete 30-day data. This strategy helps to maintain data completeness and consistency while avoiding potential bias due to insufficient data.

### 3.2 Feature Extraction

When constructing stock data sets, if only the daily opening price (open), the highest price (high), the lowest price (low), the closing price (close), and the trading volume (volume) of these market indicators as features to deep learning training is too superficial, reflecting the essence of stock price fluctuations. Stock price prediction should be based on a deep understanding of market behavior. Therefore, when constructing the stock dataset in this paper, this paper combined the market data with technical indicators, and finally got the dataset composed of 16 features as shown in [Table 1](#).

**Table 1:** Basic indicators for prediction

Indicators	Full name of index	Indicators	Full name of index
Open	Opening price	RSV	Raw stochastic value
High	Highest price	ATR	Average true range
Low	Lowest price	MA7	7-day moving average
Close	Closing price	MA14	14-day moving average
Adjust close	Adjusted closing price	MA21	21-day moving average
Volume	Trading volume	Upper_band	Upper bollinger band
Pct_change	Percentage change	Lower_band	Lower bollinger band
RSI	Relative strength index	MACD	Moving average convergence divergence

### 3.3 Evaluating Indicators

To comprehensively assess the effectiveness of the stock price prediction model, the following three evaluating indicators are used in this paper: mean absolute error (MAE), root mean square error (RMSE), and coefficient of determination ( $R^2$ ). The formulas for these indicators are shown in Eqs. (3)–(5), respectively:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - y'_i| \quad (3)$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - y'_i)^2} \quad (4)$$

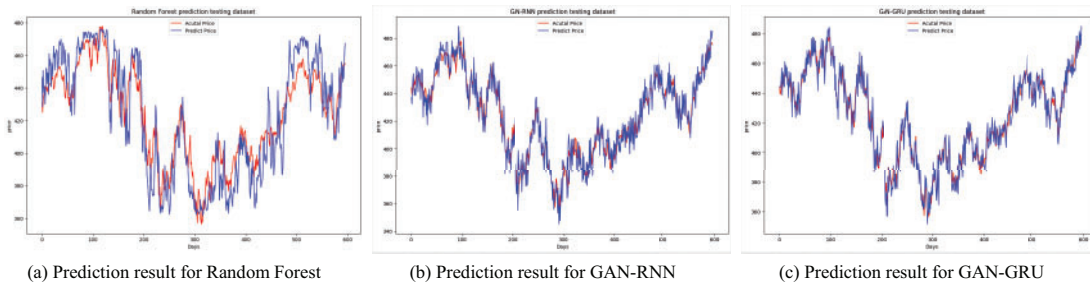
$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - y'_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2} \quad (5)$$

where  $y'_i$  represents the predicted value at moment the  $i$ ,  $y_i$  represents the actual value at moment  $i$ ,  $\bar{y}_i$  represents the actual mean value at moment  $i$ , and  $n$  is the total number of samples. Smaller values of MAE or RMSE mean better performance, while  $R^2$  takes values in the range of  $[0, 1]$ , and in general, the closer the value of  $R^2$  to 1, the better the model fits.

### 3.4 Experimental Results

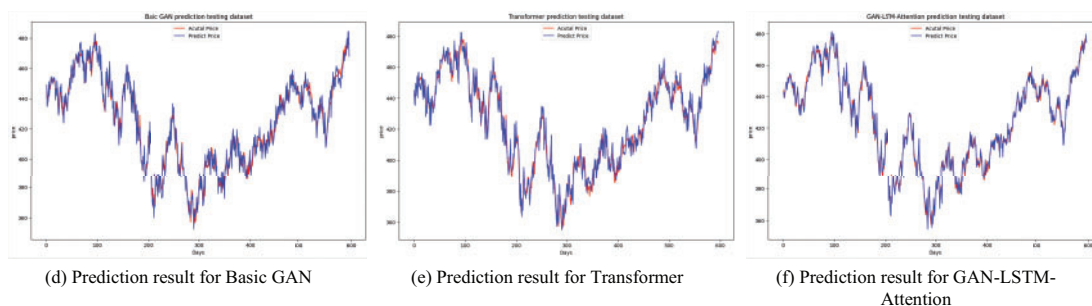
#### 3.4.1 S&P 500 Index Prediction Experiment

In the S&P 500 index prediction experiment, this paper compares six different models: Random Forest (RF), GAN-RNN (the generator consists of RNN network and fully connected layer, the discriminator consists of CNN network and fully connected layer), GAN-GRU (the generator consists of GRU network and fully connected layer, the discriminator consists of CNN network and fully connected layer), Basic GAN (the generator consists of LSTM network and fully connected layer, the discriminator consists of CNN network and fully connected layer), Transformer (A deep neural network model based on self-attention mechanism), and the GAN-LSTM-Attention model proposed in this paper. The test set prediction results are displayed in Fig. 4, where the horizontal axis represents the number of days and the vertical axis represents the stock price. In each figure, the red solid line represents the actual stock closing price and the blue solid line represents the predicted stock closing price of the corresponding model. The closer the two curves are, the better the model's prediction is. Meanwhile, three evaluation metrics were calculated according to Eq. (3): MAE, RMSE and,  $R^2$ . The results are displayed in Table 2, where the best results are shown in bold.



**Figure 4:** (Continued)





**Figure 4:** The predicted curves of different methods in the S&P 500 index

**Table 2:** Comparison of evaluation indicators of the four methods

Method	MAE	RMSE	$R^2$ (%)
Random forest	9.113	9.783	78.108%
GAN-RNN	6.080	6.998	94.233%
GAN-GRU	4.872	5.689	96.189%
Basic GAN	4.195	4.797	97.289%
Transformer	2.609	3.012	97.811%
GAN-LSTM-Attention	<b>2.098</b>	<b>2.403</b>	<b>98.320%</b>

Observations show that the Random Forest model performs the worst with an MAE of 9.113, an RMSE of 9.783, and an  $R^2$  of 78.108%. This is due to the fact that Random Forest is a decision tree-based integration method that is good at dealing with static features, but it has limited ability to deal with time series data. Random Forest cannot effectively capture sequential dependencies and long-term trends in time series, and therefore performs poorly in complex financial time series forecasting. Compared to deep learning models, Random Forest lacks specialized processing mechanisms for time-series features, such as recursive structures or attention mechanisms, which makes it difficult to compete with other models when dealing with complex financial market dynamics. The GAN-RNN model outperforms Random Forest with an MAE of 6.080, an RMSE of 6.998, and an  $R^2$  of 94.233%. Compared to Random Forest, GAN-RNN can better capture order dependence in time series by Recurrent Neural Networks (RNN). However, RNN is prone to gradient vanishing or explosion problems when dealing with long sequence data, which limits its ability to model long-term dependencies. Although the GAN-RNN model has introduced an adversarial learning mechanism through generative adversarial networks, which enhances the generalization ability of the model, the inherent defects of RNN still result in a less effective prediction than other improved GAN models. The GAN-GRU model further improves the prediction performance with an MAE of 4.872, an RMSE of 5.689, and an  $R^2$  of 96.189%. The GRU (Gated Recurrent Unit) is structurally improved compared to RNN and partially mitigates the gradient vanishing problem by introducing update gates and reset gates, allowing the model to capture long-time dependencies more efficiently, more consistently and exhibit faster convergence and better generalization during training. This makes it more effective than GAN-RNN in capturing long time trends and short-term fluctuations in the S&P 500 index.

The Basic GAN model outperforms GAN-GRU with an MAE of 4.195, RMSE of 4.797, and  $R^2$  of 97.289%. This is due to the fact that the Basic GAN model uses LSTM (Long Short-Term

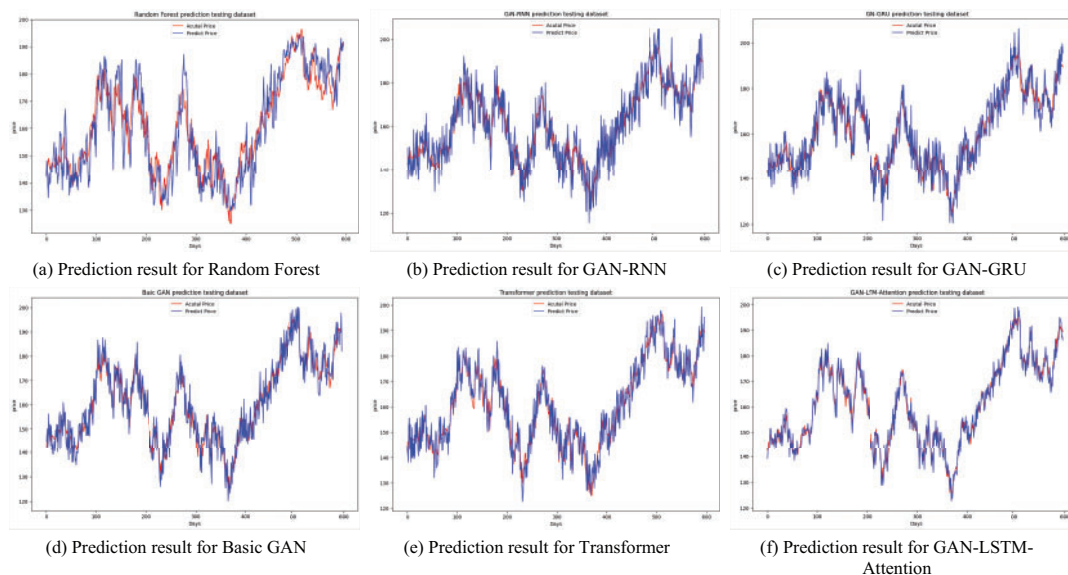
Memory Network) as a generator. LSTM significantly improves the ability to model long sequence data by introducing forgetting gates, input gates, and output gates, allowing the model to perform better when dealing with time-dependent financial data. Compared with GRU, the complex gating mechanism of LSTM is can control the information flow in a more detailed way, thus maintaining a stable gradient in extremely long sequences. As a result, the Basic GAN model can better capture the long-term dependencies and complex market characteristics of the S&P 500 index, making its predictive performance superior to that of the GAN-GRU. The Transformer model, through its unique self-attention mechanism, performs particularly well in terms of predictive performance, with an MAE of 2.609, an RMSE of 3.012, and an  $R^2$  of 97.811%. The Transformer model is can process sequence data in parallel without a recursive structure, which makes it more efficient in processing long-time sequence data. Meanwhile, the self-attention mechanism can dynamically assign attention weights to accurately capture important features and key moments in the time series. Compared with LSTM, the Transformer is more advantageous in processing long time series because its parallel computing capability significantly reduces the training time, and the self-attention mechanism can more flexibly adjust the model's focus to adapt to different market conditions. As a result, Transformer can more accurately capture the nuances of market changes when forecasting the S&P 500.

The GAN-LSTM-Attention model performs the best out of all the models, with an MAE of 2.098, an RMSE of 2.403, and an  $R^2$  of 98.320%. This is due to the fact that the model combines GAN, LSTM, and Attention Mechanisms to take full advantage of their respective strengths. The LSTM provides a powerful ability to model long-term dependencies, while the Attention Mechanisms allow the model to focus on the features in the input data that have the greatest impact on the prediction results. Compared to the Basic GAN model using only LSTM, the GAN-LSTM-Attention model further improves sensitivity to market changes through the attention mechanism, enabling the model to extract key information from a large amount of noisy data, thus improving prediction accuracy. Compared to the Transformer, although the Transformer has an advantage in parallel processing, GAN-LSTM-Attention enhances the generalization ability of the model through the adversarial learning mechanism, which enables it to maintain high-precision prediction results under various market conditions. In summary, the GAN-LSTM-Attention model demonstrates the highest prediction accuracy in predicting the S&P 500 index by combining the powerful time-series modeling capability of LSTM with the feature-focusing capability of the attention mechanism. The success of this model lies in its ability not only to accurately capture complex dependencies in long time series, but also to flexibly respond to the importance of different features in the market, thus providing the most accurate forecasting results.

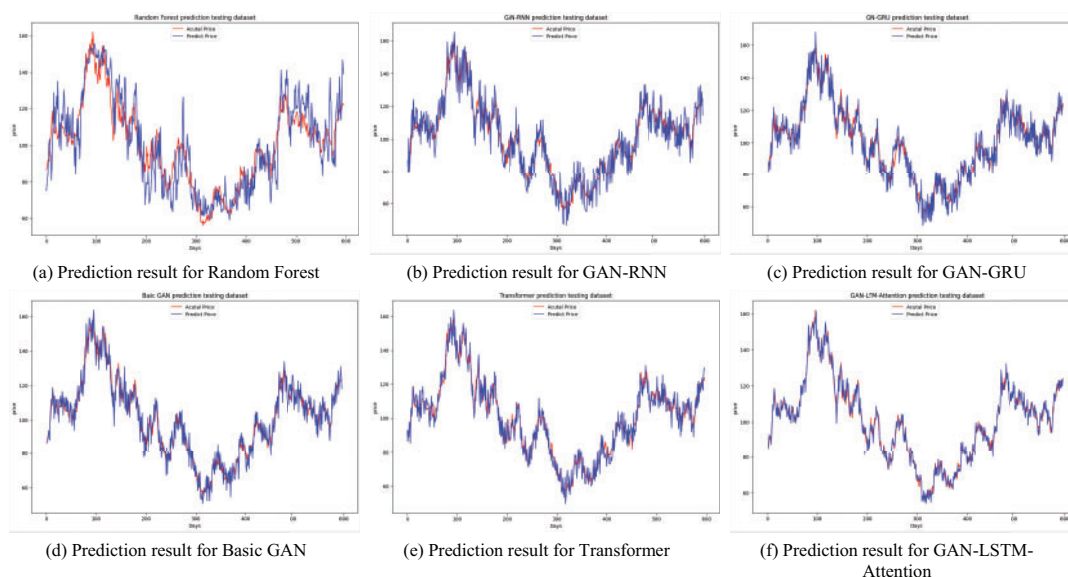
### 3.4.2 Individual Stock Prediction Experiment

After performing the prediction of S&P 500 index, although the excellent performance of the GAN-LSTM-Attention model in market index prediction has been observed, the results of a single index may occasionally be affected by atypical market behaviors, leading to the instability of the results. Therefore, to comprehensively assess the practical application and generalization ability of the model, this experiment further extends the application scope of the model by selecting the stocks of three technology giants, Apple Incorporated (stock code: AAPL), Advanced Micro Devices Incorporated (stock code: AMD), and Google Incorporated (stock code: GOOG) for in-depth analysis. These three companies have significant representation and influence in the global market, and their stock price fluctuations reflect the economic dynamics of the technology sector as well as the broader market. The experiment covers data from 01 January 2012 to 31 December 2023. After preprocessing, the data is divided into a training dataset and a testing dataset in chronological order, where the first 80% of

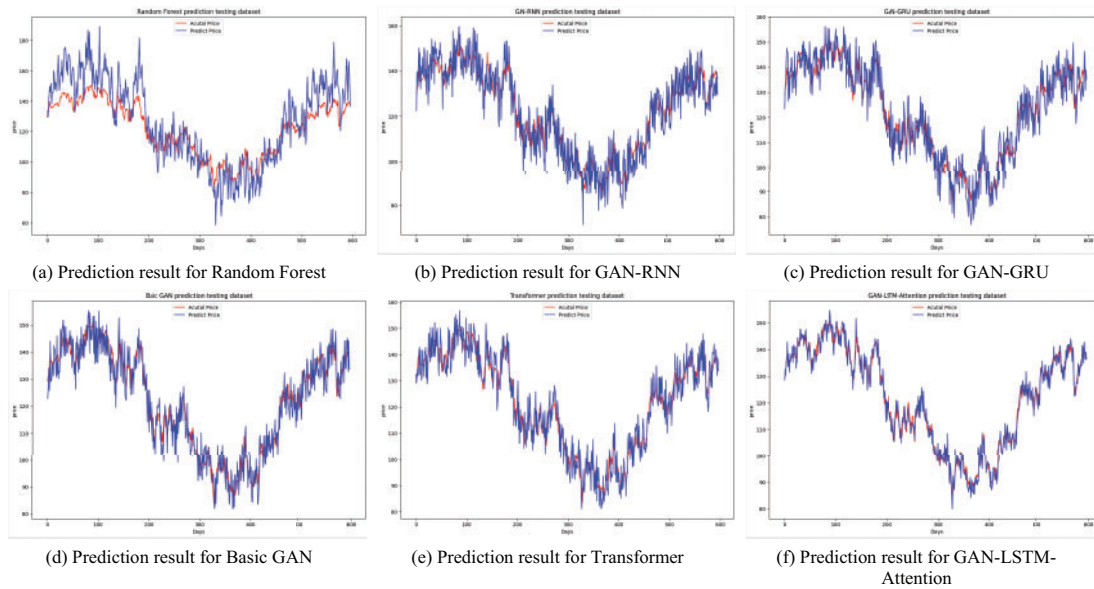
the data is used as the training dataset to train the model, and the second 20% of the data is used as the testing dataset to evaluate the model's prediction effect. There are a total of 3018 trading days, of which 80% of the training dataset is 2414 trading days data and 20% of the testing dataset is 604 trading days data. This time period contains multiple economic cycles, including rapid market growth, short-term adjustments, and global impact events, which makes the data set highly challenging and representative. In addition, Figs. 5–7 show the prediction result plots of the six models used in this paper for the test set of Apple Incorporated (stock code: AAPL), Advanced Micro Devices Incorporated (stock code: AMD), and Google Incorporated (stock code: GOOG) stocks, respectively.



**Figure 5:** The predicted curves of different methods in the AAPL



**Figure 6:** The predicted curves of different methods in the AMD



**Figure 7:** The predicted curves of different methods in the GOOG

Tables 3–5 summarize the three main evaluation metrics of each model on each stock: mean absolute error (MAE), root mean square error (RMSE), and coefficient of determination ( $R^2$ ), where the best results are shown in bold.

**Table 3:** Comparison of evaluation indicators of the four methods for AAPL

Method	MAE	RMSE	$R^2$ (%)
Random forest	9.596	10.365	72.607%
GAN-RNN	6.072	6.987	81.529%
GAN-GRU	4.847	5.609	88.099%
Basic GAN	3.934	4.558	92.138%
Transformer	3.321	3.547	95.176%
GAN-LSTM-Attention	<b>1.964</b>	<b>2.266</b>	<b>98.057%</b>

**Table 4:** Comparison of evaluation indicators of the four methods for AMD

Method	MAE	RMSE	$R^2$ (%)
Random forest	8.921	9.630	75.560%
GAN-RNN	6.128	7.035	89.688%
GAN-GRU	5.038	5.752	93.108%
Basic GAN	3.980	4.625	95.544%
Transformer	2.719	3.276	96.978%
GAN-LSTM-Attention	<b>1.992</b>	<b>2.310</b>	<b>98.889%</b>

**Table 5:** Comparison of evaluation indicators of the four methods for GOOG

Method	MAE	RMSE	R <sup>2</sup> (%)
Random forest	9.068	9.787	74.651%
GAN-RNN	6.062	7.033	84.424%
GAN-GRU	4.901	5.659	89.917%
Basic GAN	4.082	4.678	93.109%
Transformer	3.002	3.469	96.195%
GAN-LSTM-Attention	<b>2.048</b>	<b>2.339</b>	<b>98.277%</b>

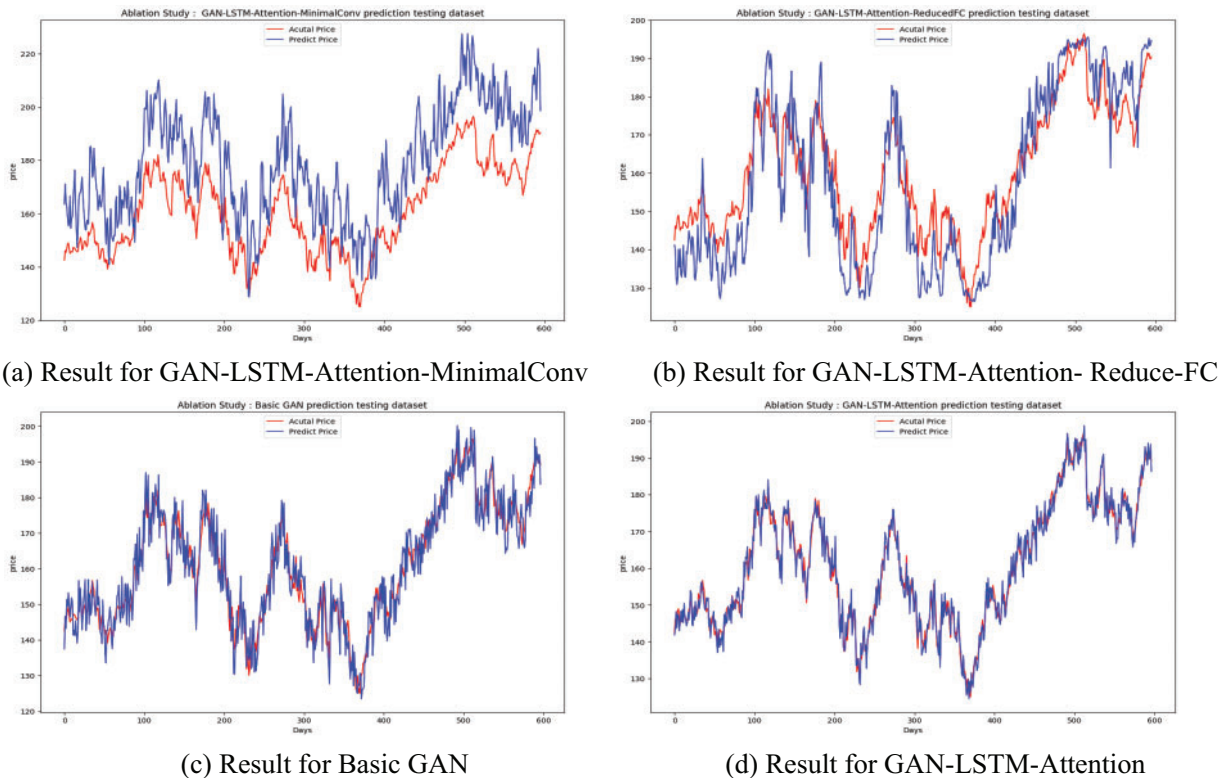
As shown in [Tables 3–5](#), the performance of the GAN-LSTM-Attention model is significantly better than the other models in all the indicators under different individual stock samples, which not only verifies its excellent performance in practical applications, but also proves its strong generalization ability. After several experiments, it is noted that the results of the GAN-LSTM-Attention model are always lower than those of the other five comparison models in terms of MAE and RMSE metrics for individual stock prediction, indicating that the model has obvious advantages in prediction accuracy and error control. Meanwhile, in terms of R<sup>2</sup> metrics, the results of the GAN-LSTM-Attention model are all better than those of the other five comparison models, which further proves that the model has a higher degree of fit in terms of its ability to capture the relationship between explanatory variables and prediction targets. These results clearly demonstrate that the GAN-LSTM-Attention model is not only able to perform well in stock index prediction, but also consistently provides highly accurate predictions at the individual stock level. This excellent and robust prediction performance makes the GAN-LSTM-Attention model an important reference value in practical applications. For investors, the model's ability to provide accurate market trend predictions in advance helps to make more informed investment decisions, which effectively reduces investment risks and improves investment returns. In addition, these experimental results further quantitatively verify the rationality and effectiveness of the GAN-LSTM-Attention model adopted in this paper in dealing with the complex financial market environment. The wide applicability of the model under different market conditions fully demonstrates its potential and value in practice.

### 3.5 Ablation Study

To validate the effectiveness of the GAN-LSTM-Attention model proposed in this paper for stock price prediction, as well as the validity and necessity of each functional module in the model, three model variants were designed in this study. Each variant is used to evaluate the contribution of these components to the overall model performance by removing specific modules.

Firstly, this paper introduces the GAN-LSTM-Attention-MinimalConv variant, a model that reduces the three one-dimensional convolutional layers in the discriminator to a single one-dimensional convolutional layer, while the remaining components remain unchanged. This variation aims to explore the necessity of multiple convolutional layers in the discriminator and their role in feature extraction and the distinction between true and false data. Next, this paper proposes a GAN-LSTM-Attention-Reduce-FC variant of the model that reduces the three fully connected layers to a single layer in both the generator and the discriminator. This variant helps to evaluate the impact of the fully connected layers on the predictive power and learning complexity of the model. Finally, this paper designs the Basic GAN variant, which removes the attentional mechanism from the generator, to

test the impact of this mechanism for enhancing the capture of critical information in time series data. Through these ablation experiments, this paper expects to gain insights into the specific contributions of individual modules to the performance of the GAN-LSTM-Attention model, as well as their role and efficiency in a complex stock market forecasting environment. Fig. 8 comprehensively shows the stock price prediction results of different variants of the GAN-LSTM-Attention model on the same dataset. Table 6 then details the performance metrics of these variants.



**Figure 8:** The predicted curves of different GAN-LSTM-Attention variants in the AAPL

**Table 6:** Comparison of evaluation indicators of the different variants for AAPL

Method	MAE	RMSE	R <sup>2</sup> (%)
GAN-LSTM-Attention-MinimalConv	10.047	11.138	68.992%
GAN-LSTM-Attention-Reduce-FC	9.169	9.723	76.283%
Basic GAN	3.934	4.558	92.138%
GAN-LSTM-Attention	<b>1.964</b>	<b>2.266</b>	<b>98.057%</b>

The results show that GAN-LSTM-Attention-MinimalConv has the worst performance, which highlights the fact that reducing the number of convolutional layers significantly reduces the discriminator's ability to distinguish between real and generated data. The reduction in the number of convolutional layers, which are responsible for extracting key features from the input data, results in the discriminator not being able to capture sufficient feature detail, especially when dealing with



complex stock market data. Due to the reduced discriminative power, the generator may be able to deceive the discriminator without much effort, which in turn reduces the quality of the generated data because the generator is not sufficiently challenged during the training process. Meanwhile, the performance of GAN-LSTM-Attention-Reduce-FC is not satisfactory. This suggests that simplifying the fully connected layer can weaken the model's ability to synthesize and utilize the learned features, while reducing the fully connected layer in the generator and discriminator can lead to an impaired model balance, which affects the overall learning effectiveness and generation quality. In contrast, Basic GAN exhibits strong competitiveness, but its performance still fails to outperform the full GAN-LSTM-Attention model. This suggests that in stock price prediction, the attention mechanism helps the model to focus on the data points that have critical information value for future trend prediction. Without the support of the attention mechanism, the LSTM layer, although capable of capturing time series dependencies, is not effective enough to filter and enhance the response to specific important data, thus affecting the overall prediction performance. Consequently, the performance results of the above variants confirm the positive effects of components of the proposed model, respectively, for stock price prediction.

#### 4 Conclusion and Future Research

The GAN-LSTM-Attention model used in this paper is based on Generative Adversarial Networks and combines Long Short-Term Memory Networks, Attention Mechanisms, Convolutional Neural Networks, and Fully Connected Layers to predict stock price. The LSTM is introduced to efficiently deal with long-term dependencies in the time-series data, while the attention mechanism further strengthens the model's ability to recognize the critical time-series information and extraction capabilities, the fully connected layer acts as the output layer, effectively integrating and outputting the previously processed information to ensure the accuracy and reliability of the prediction results, which makes them ideal for generators. On the other hand, CNN is selected as the core component of the discriminator as it works together with the fully connected layer to achieve accurate discrimination of the quality of the generated data by extracting local features in the sequence data and learning complex feature combinations through multilayer convolutional layers. The effectiveness of the model is demonstrated through experimental validation, and its excellent performance is verified through different evaluation indicators. A comprehensive comparative analysis shows that the model outperforms other models in various aspects. In addition, the model helps traders to mitigate financial risks and improve their decision-making ability. Given the complexity and rapid changes in the stock market, future research can be extended in several aspects, such as exploring the effects of different time steps on the prediction performance, and incorporating public opinion influences and unexpected events in financial news and stock commentary analysis into the prediction model to further enhance the prediction performance.

**Acknowledgement:** None.

**Funding Statement:** This research was funded by the project supported by the Natural Science Foundation of Heilongjiang Provincial (Grant Number LH2023F033) and the Science and Technology Innovation Talent Project of Harbin (Grant Number 2022CXRC CG006).

**Author Contributions:** The authors confirm contribution to the paper as follows: study conception and design: Peng Li, Yanrui Wei; data collection: Lili Yin; analysis and interpretation of results: Peng Li,

Yanrui Wei, Lili Yin; draft manuscript preparation: Peng Li, Yanrui Wei. All authors reviewed the results and approved the final version of the manuscript.

**Availability of Data and Materials:** The data used in this paper can be requested from the corresponding author.

**Ethics Approval:** Not applicable.

**Conflicts of Interest:** The authors declare no conflicts of interest to report regarding the present study.

## References

- [1] A. Thakkar and K. Chaudhari, "Fusion in stock market prediction: A decade survey on the necessity, recent developments, and potential future directions," *Inf. Fusion*, vol. 65, pp. 95–107, Jan. 2021. doi: [10.1016/j.inffus.2020.08.019](https://doi.org/10.1016/j.inffus.2020.08.019).
- [2] C. Wang, "Stock return prediction with multiple measures using neural network models," *Financ. Innov.*, vol. 10, no. 1, Jun. 2024, Art. no. 72. doi: [10.1186/s40854-023-00608-w](https://doi.org/10.1186/s40854-023-00608-w).
- [3] M. Agrawal, P. K. Shukla, R. Nair, A. Nayyar, and M. Masud, "Stock prediction based on technical indicators using deep learning model," *Comput. Mater. Contin.*, vol. 70, no. 1, pp. 287–304, 2022. doi: [10.32604/cmc.2022.014637](https://doi.org/10.32604/cmc.2022.014637).
- [4] D. M. Durairaj and B. H. K. Mohan, "A convolutional neural network based approach to financial time series prediction," *Neural Comput. Appl.*, vol. 34, no. 16, pp. 13319–13337, Aug. 2022. doi: [10.1007/s00521-022-07143-2](https://doi.org/10.1007/s00521-022-07143-2).
- [5] M. M. Kumbure, C. Lohrmann, P. Luukka, and J. Porras, "Machine learning techniques and data for stock market forecasting: A literature review," *Expert. Syst. Appl.*, vol. 197, Jul. 2022, Art. no. 116659. doi: [10.1016/j.eswa.2022.116659](https://doi.org/10.1016/j.eswa.2022.116659).
- [6] D. Das, A. S. Sadiq, N. B. Ahmad, and J. Lloret, "Stock market prediction with big data through hybridization of data mining and optimized neural network techniques," *J. Multiple-Valued Logic Soft Comput.*, vol. 29, no. 1–2, pp. 157–181, 2017.
- [7] T. B. Çelik, O. Ican, and E. Bulut, "Extending machine learning prediction capabilities by explainable AI in financial time series prediction," *Appl. Soft Comput.*, vol. 132, Jan. 2023, Art. no. 109876.
- [8] A. A. Adebisi, A. O. Adewumi, and C. K. Ayo, "Comparison of ARIMA and artificial neural networks models for stock price prediction," *J. Appl. Math.*, vol. 2014, 2014, Art. no. 614342. doi: [10.1155/2014/614342](https://doi.org/10.1155/2014/614342).
- [9] A. S. Ahmar, P. K. Singh, N. V. Thanh, N. V. Tinh, and V. M. Hieu, "Prediction of BRIC stock price using ARIMA, SutteARIMA, and holt-winters," *Comput. Mat. Continua*, vol. 70, no. 1, pp. 523–534, 2022. doi: [10.32604/cmc.2022.017068](https://doi.org/10.32604/cmc.2022.017068).
- [10] H. Herwartz, "Stock return prediction under GARCH—An empirical assessment," *Int. J. Forecast.*, vol. 33, no. 3, pp. 569–580, Jul.–Sep. 2017. doi: [10.1016/j.ijforecast.2017.01.002](https://doi.org/10.1016/j.ijforecast.2017.01.002).
- [11] A. Al Nasser, A. Tucker, and S. de Cesare, "Quantifying StockTwits semantic terms' trading behavior in financial markets: An effective application of decision tree algorithms," *Expert. Syst. Appl.*, vol. 42, no. 23, pp. 9192–9210, Dec. 2015. doi: [10.1016/j.eswa.2015.08.008](https://doi.org/10.1016/j.eswa.2015.08.008).
- [12] H. J. Kang, X. Y. Zong, J. Y. Wang, and H. N. Chen, "Binary gravity search algorithm and support vector machine for forecasting and trading stock indices," *Int. Rev. Econ. Financ.*, vol. 84, pp. 507–526, Mar. 2023. doi: [10.1016/j.iref.2022.11.009](https://doi.org/10.1016/j.iref.2022.11.009).
- [13] M. Q. Jiang, J. P. Liu, L. Zhang, and C. Y. Liu, "An improved Stacking framework for stock index prediction by leveraging tree-based ensemble models and deep learning algorithms," *Phys. A: Stat. Mech. Appl.*, vol. 541, Mar. 2020, Art. no. 122272. doi: [10.1016/j.physa.2019.122272](https://doi.org/10.1016/j.physa.2019.122272).

- [14] C. Krauss, X. A. Do, and N. Huck, "Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500," *Eur. J. Oper. Res.*, vol. 259, no. 2, pp. 689–702, Jun. 2017. doi: [10.1016/j.ejor.2016.10.031](https://doi.org/10.1016/j.ejor.2016.10.031).
- [15] L. L. Yin, B. L. Li, P. Li, and R. B. Zhang, "Research on stock trend prediction method based on optimized random forest," *CAAI Trans. Intell. Technol.*, vol. 8, no. 1, pp. 274–284, Mar. 2023. doi: [10.1049/cit2.12067](https://doi.org/10.1049/cit2.12067).
- [16] J. H. Zhao, D. L. Zeng, S. Liang, H. L. Kang, and Q. M. Liu, "Prediction model for stock price trend based on recurrent neural network," *J. Am. Intell. Humaniz. Comput.*, vol. 12, no. 1, pp. 745–753, Jan. 2021. doi: [10.1007/s12652-020-02057-0](https://doi.org/10.1007/s12652-020-02057-0).
- [17] S. Usmani and J. A. Shamsi, "LSTM based stock prediction using weighted and categorized financial news," *PLoS One*, vol. 18, no. 3, Mar. 2023, Art. no. e0282234. doi: [10.1371/journal.pone.0282234](https://doi.org/10.1371/journal.pone.0282234).
- [18] W. J. Lu, J. Z. Li, Y. F. Li, A. J. Sun, and J. Y. Wang, "A CNN-LSTM-based model to forecast stock prices," *Complexity*, vol. 2020, Nov. 2020, Art. no. 6622927. doi: [10.1155/2020/6622927](https://doi.org/10.1155/2020/6622927).
- [19] Q. Chen, W. Y. 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](https://doi.org/10.1109/ACCESS.2020.3004284).
- [20] S. R. Polamuri, K. Srinivas, and A. K. Mohan, "Multi-model generative adversarial network hybrid prediction algorithm (MMGAN-HPA) for stock market prices prediction," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 9, pp. 7433–7444, Oct. 2022.
- [21] J. L. Wu, X. R. Tang, and C. H. Hsu, "A prediction model of stock market trading actions using generative adversarial network and piecewise linear representation approaches," *Soft Comput.*, vol. 27, no. 12, pp. 8209–8222, Jun. 2023. doi: [10.1007/s00500-022-07716-2](https://doi.org/10.1007/s00500-022-07716-2).
- [22] Q. F. Liu, Z. Y. Tao, Y. M. Tse, and C. J. Wang, "Stock market prediction with deep learning: The case of China," *Financ. Res. Lett.*, vol. 46, May 2022, Art. no. 102209. doi: [10.1016/j.frl.2021.102209](https://doi.org/10.1016/j.frl.2021.102209).
- [23] S. Hansun and J. C. Young, "Predicting LQ45 financial sector indices using RNN-LSTM," *J. Big Data*, vol. 8, no. 1, Jul. 2021, Art. no. 104. doi: [10.1186/s40537-021-00495-x](https://doi.org/10.1186/s40537-021-00495-x).
- [24] R. Chiong, Z. W. Fan, Z. Y. Hu, and S. Dhakal, "A novel ensemble learning approach for stock market prediction based on sentiment analysis and the sliding window method," *IEEE Trans. Comput. Soc. Syst.*, vol. 10, no. 5, pp. 2613–2623, Oct. 2023. doi: [10.1109/TCSS.2022.3182375](https://doi.org/10.1109/TCSS.2022.3182375).