Improved Whale Optimization Algorithm with LSTM for Stock Index Prediction

Yu Sun[1], Sofianita Mutalib[2]*, Liwei Tian[3]

School of Management, Guangdong University of Science and Technology, Dongguan, Guangdong Province, China[1]
School of Computing Sciences-College of Computing-Informatics and Mathematics,
Universiti Teknologi MARA, Shah Alam, Selangor, Malaysia[1, 2]
School of Computing, Guangdong University of Science and Technology, Dongguan, Guangdong Province, China[3]

***Abstract*—After the COVID-19 pandemic, the global economy began to recover. However, stock market fluctuations continue to affect economic stability, making accurate predictions essential. This study proposes an Improved Whale Optimization Algorithm (IWOA) to optimize the parameters of the Long Short-Term Memory (LSTM) model, thereby enhancing stock index predictions. The IWOA improves upon the traditional Whale Optimization Algorithm (WOA) by integrating logistic chaotic mapping to increase population diversity and prevent premature convergence. Additionally, it incorporates a dynamic adjustment mechanism to balance global exploration and local exploitation, thus boosting optimization performance. Experiments conducted on five representative global stock indices demonstrate that the IWOA-LSTM model achieves higher accuracy and reliability compared to WOA-LSTM, LSTM, and RNN models. This highlights its value in predicting complex time-series data and supporting financial decision-making during economic recovery.**

Keywords—Long short-term memory network; chaotic mapping; dynamic adjustment mechanism; improved whale optimization algorithm; financial time series forecasting

I. Introduction

Stock market indices are published by stock exchanges or financial institutions and are important financial indicators that reflect market fluctuations. These indices directly affect investor sentiment and decision-making and serve as key references for investors. Given the significant impact of stock market movements on the global economy, predicting these movements has been a top priority for researchers and investors. Their goal is to develop effective investment strategies and reduce risk. Despite the passage of time, the complex patterns and return metrics that emerge in the stock market within the framework of unsupervised automated prediction remain difficult to predict accurately. This underscores the critical need for a progressive predictive approach that combines human expertise with advanced technological capabilities to improve the accuracy and reliability of stock and economic forecasts [1].

In recent decades, the adoption of machine learning techniques and metaheuristic algorithms in financial time series forecasting has attracted significant attention. Conventional neural networks, such as Recurrent Neural Networks (RNNs), have been shown to be effective in capturing complex fluctuations in the stock market. Chen et al. introduced a deep learning prediction model based on RNN, which integrates social media news content (sentiment and topic features) with technical indicators to strengthen the predictive accuracy of stock market volatility [2]. Haromainy et al. utilized a genetic algorithm-optimized RNN model to predict stock trends for stock prices, demonstrating its ability to capture nonlinear features in stock market data [3]. Additionally, Zhao et al. incorporated fuzzy logic with RNN to improve stock market volatility prediction [4].

Despite the advantages of RNN models, prediction accuracy remains limited due to the high nonlinearity and chaotic nature of the stock market. As deep learning continues to grow, LSTM network, known for its superior time series processing capabilities, has become a prominent research focus. Abdul Quadir et al. utilized the LSTM algorithm to analyze normalized time series data, addressing the vanishing gradient issue observed in simpler RNN and determining the relationship between historical and future values [5]. In the medical field, Academician Zhong Nanshan's team utilized an LSTM-based recurrent neural network to study and predict the peaks and sizes of COVID-19 [6]. In the domain of energy consumption forecasting, LSTM-based methods have demonstrated outstanding predictive performance [7]. Notably, in the research area of financial time series forecasting, LSTM has shown remarkable predictive performance. Li et al. further validated the potential of LSTM in capturing complex stock price patterns and improving prediction accuracy [8]. Singh et al. combined Convolutional Neural Networks (CNN) with LSTM to propose a hybrid model for Indian stock portfolio management, which outperformed traditional models [9].

However, LSTM networks also face challenges, such as susceptibility to overfitting, sensitivity to hyperparameter selection, and the risk of falling into local optima, which can limit prediction accuracy [10]. Hyperparameter selection in LSTM models typically relies on manual experience, which significantly impacts model performance. To address these issues, metaheuristic optimization algorithms have been extensively utilized for hyperparameter optimization [11–13]. For example, Zhang et al. applied Particle Swarm Optimization (PSO) to optimize LSTM hyperparameters for predicting short-term fluctuations in the highest prices of U.S. stocks, demonstrating the superiority of the optimized LSTM model [14]. However, the PSO method is prone to slow convergence and inefficiency in high-dimensional spaces, particularly in multimodal optimization problems or complex search spaces, leading to local optima and reduced operational efficiency [15]. In response to these challenges, several studies have explored

*Corresponding Author.

the integration of the Whale Optimization Algorithm (WOA) with deep learning models. Xin et al. combined the WOA with the LSTM model to predict the stock market, achieving significant improvements in forecasting accuracy [16]. Hasan enhanced the convergence mechanism of WOA and applied it to the optimization of time-series prediction models, yielding excellent results in temperature and humidity forecasting [17].

Inspired by the aforementioned studies, this paper proposes a novel approach that integrates an Improved Whale Optimization Algorithm (IWOA) with LSTM networks. IWOA enhances population diversity and reduces the risk of premature convergence by incorporating chaotic mapping. Additionally, it features an adaptive factor mechanism that dynamically balances exploration and exploitation during the optimization process, leading to improved efficiency and accuracy. By leveraging IWOA's robust optimization capabilities, the hyperparameters of the LSTM model are fine-tuned more effectively, enabling the model to better capture the complexity of nonlinear and chaotic patterns in financial time series.

II. Methods

A. LSTM

In the early stages of time series forecasting, recurrent neural networks (RNNs) gained widespread use due to their capability to process sequential data. Unlike traditional feedforward neural networks (e.g., backpropagation neural networks, BPNN), which propagate signals in a single direction, RNNs introduce weighted connections between hidden layer neurons. This architecture enables the output of hidden layer neurons at each time step to depend on information from the previous time step, allowing the network to effectively capture temporal dependencies. By incorporating both feedforward and internal feedback connections, RNNs exhibit dynamic temporal behavior that influences their internal states. However, in practice, the hidden state of an RNN at each time step is determined by both the hidden layer values from the previous time step and the input values at the current time step, which restricts its ability to retain long-term memory [18].

To overcome the limitations of traditional RNNs, Graves extended the Long Short-Term Memory (LSTM) neural network, which effectively addresses these challenges [19]. LSTM replaces the hidden layer nodes of standard RNNs with specialized memory units, allowing the network to better retain and manage temporal information, particularly for modeling long-term dependencies. The core component of LSTM is the cell state, which functions as a channel for transmitting information across the network. LSTM introduces input, forget, and output gates to enhance the functionality of global memory cells. These gates regulate the retention, updating, or discarding of information at each time step, enabling the network to efficiently learn long-term dependencies. Fig. 1 illustrates the architecture of the LSTM model.

The structure of LSTM is highly efficient in managing long-term relationships within time series data, especially when events are delayed. In LSTM, three gates regulate the cell state, each employing a Sigmoid activation function and pointwise multiplication. The Sigmoid output, ranging from 0 to 1, determines how much information passes through: a value of 0 indicates complete "blocking," while a value of 1 signifies full "pass-through." This gating mechanism enables LSTM to efficiently retain and propagate long-term dependency information in time series data. The workflow of the LSTM network is described by the following equations.

$$i_t = \sigma(W_i * [h_{t-1}, X_t] + b_i) \tag{1}$$

$$f_t = \sigma\left(W_f * [h_{t-1}, X_t] + b_f\right) \tag{2}$$

$$o_t = \sigma(W_o * [h_{t-1}, X_t] + b_o) \tag{3}$$

$$\hat{C}_t = tanh(W_c * [h_{t-1}, X_t] + b_c) \tag{4}$$

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t \tag{5}$$

$$h_t = o_t * \tanh(C_t) \tag{6}$$

In Eq. (1) to Eq. (6), W refers to the weight vectors, while b denotes the bias terms.

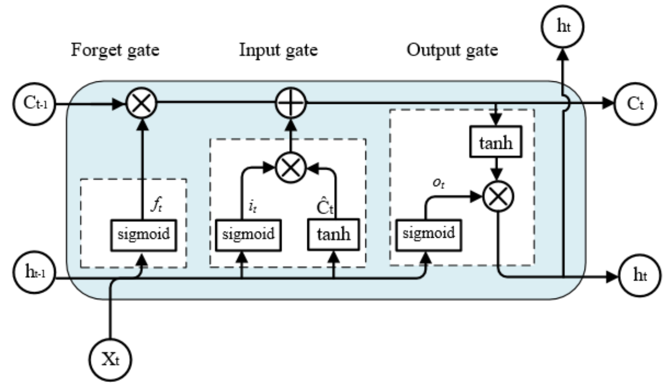

Fig. 1. Structure of LSTM.

B. WOA

The Whale Optimization Algorithm (WOA) is a novel swarm intelligence method inspired by the bubble net foraging strategy of humpback whales. This algorithm demonstrates superior performance compared to traditional optimization approaches. WOA simulates this behavior through two main search strategies: exploration and exploitation. During exploration, whales move randomly in search of prey, while in the exploitation phase, they navigate toward the prey in a spiral pattern to find the optimal solution. The algorithm uses a series of mathematical equations to simulate these behaviors, enabling it to effectively search for optimal solutions in complex and high-dimensional spaces [20]. In WOA, each candidate solution corresponds to a position within the search space, and the algorithm optimizes the objective function by mimicking the whale's hunting behavior. The algorithm operates through three main phases [21].

1) Encircling prey: When a whale detects the position of its prey, it adjusts its position based on the current best solution. During this process, the whale moves closer to the optimal solution by a certain proportion. The position of the whale is updated as described in Eq. (7).

$$\begin{cases} D = |C \cdot X^*(t) - X(t)| \\ X(t+1) = X^*(t) - A \cdot D \\ \mathrm{A} = 2a \cdot r - a \\ \mathrm{C} = 2 \cdot r \end{cases} \tag{7}$$

In Eq. (7), D represents the distance between the whale individual and its prey. t represents the iteration number; A and C are coefficient vectors; X and X^* and are the current whale position and the current best whale position; a decreases linearly from 2 to 0 during the iteration; r is a random number in the range [0, 1].

2) Bubble-net attacking: There are two mechanisms designed for Bubble-net attacking: shrinking encircling mechanism and spiral updating position.

a) Shrinking encircling mechanism: This mechanism is implemented using the parameters in Eq. (7). During the iterations, the behavior is achieved by linearly decreasing the value of a from 2 to 0, while A fluctuates within the range [−a, a]. When A is a random value between [−1, 1], the whale's position is updated to lie somewhere between its original position and the current optimal position.

b) Spiral updating position: Initially, the distance between the whale and its prey is calculated. Then, a spiral equation is derived to simulate the whale's spiral movement, the specific equation is as follows.

$$\begin{cases} D = |X^*(t) - X(t)| \\ X(t+1) = D \cdot e^{bl} \cdot cos(2\pi l) + X^*(t) \end{cases} \tag{8}$$

In Eq. (8), b is the spiral shape constant; l is a random value within the range [-1, 1].

When the whale approaches the prey, its behaviors of shrinking encircling and spiral position updating occur simultaneously. To simulate this bubble-net attack, it is assumed that the humpback whale has a 50% chance of performing either shrinking encircling or spiral position updating. X(t+1) is shown in Eq. (9).

$$X(t+1) = \begin{cases} X^*(t) - A \cdot D & p < 0.5 \\ D \cdot e^{bl} \cdot cos(2\pi l) + X^*(t) & P \geq 0.5 \end{cases} \tag{9}$$

In Eq. (9), p is a random number in the range [0, 1].

3) Search for prey: At this phase, the whale population performs global exploration. When |A| > 1, the whale population ceases to adjust its position according to the current optimal solution. Instead, the position is updated based on a randomly selected whale, with the goal of expanding the search range and seeking the optimal solution to maintain population diversity. Therefore, only a small modification to Eq. (7) is needed to obtain the mathematical model for this stage.

$$\begin{cases} D = |C \cdot X_{rand} - X(t)| \\ X(t+1) = X_{rand} - A \cdot D \end{cases} \tag{10}$$

In Eq. (10), X_{rand} represents the position of a randomly selected whale.

WOA achieves a balance between local and global exploration through its three primary operations, thereby enhancing global search capabilities and mitigating the premature convergence problem commonly encountered in traditional optimization algorithms. As a result, WOA demonstrates superior performance in solving complex optimization problems [22].

III. Proposed IWOA-LSTM Prediction Model

A. IWOA-LSTM Model

The performance of an LSTM network is highly influenced by the selection of appropriate hyperparameters. However, traditional hyperparameter tuning methods often rely on manual expertise, which may not ensure optimal results across diverse scenarios [23]. To address this issue, this research proposes an improved Whale Optimization Algorithm (IWOA) integrated with the LSTM model to automate the hyperparameter optimization process.

IWOA combines Whale Optimization with Logistic Chaos Mapping to improve population diversity during initialization and prevent premature convergence to local optima, providing a more effective starting point for the optimization process. Additionally, IWOA incorporates a dynamic adjustment mechanism based on fitness variations, enabling the automatic fine-tuning of the mutation factor. This mechanism enhances the algorithm's global search capability and increases optimization efficiency.

In the IWOA-LSTM model, IWOA optimizes key LSTM hyperparameters, including the number of hidden units and the learning rate. By balancing global exploration and local exploitation, IWOA accelerates convergence and fine-tunes LSTM parameters, ultimately improving prediction accuracy. This approach minimizes manual intervention, enhancing the adaptability and performance of the LSTM network.

B. Chaotic Map Initialization

In WOA, the initialization of the population is a crucial factor that influences the optimization performance. Traditional random initialization can result in an uneven distribution of initial solutions, potentially reducing the algorithm's efficiency. To address this challenge, chaotic mapping is employed to adjust the control parameters of WOA, thereby improving the balance between exploration and exploitation. This technique enhances the algorithm's global convergence rate by generating a more dispersed and diversified initial population, thus reducing the likelihood of premature convergence to local optima.

In this study, chaotic mapping is introduced during the initialization phase of WOA, specifically utilizing the logistic chaotic mapping proposed by Prasad [24] and Yousri [25]. This mapping exhibits both random and deterministic characteristics, which facilitates the adjustment of WOA's control parameters. The initialization equation for chaotic mapping is as follows.

$$x_{i,j} = x_{i,j} * (b_j - a_j) + a_j \tag{11}$$

In Eq. (11), $x_{i,j}$ represents the position of the i-th whale in the j-th dimension. a_j and b_j denote the boundaries of the j-th dimensional space. The initialized $x_{i,j}$ is generated using the

Logistic chaotic mapping, and its iteration equation is as follows.

$$x = r * x * (1 - x) \tag{12}$$

In Eq. (12), r is the control parameter. In this study, its value is set to 4. This initialization method ensures the diversity of the population distribution, providing a strong starting point for subsequent iterations.

C. Dynamic Adjustment Mechanism

In the optimization process of WOA, applying an appropriate mutation operation after each individual's position update significantly enhances the algorithm's global search capability [26]. The mutation factor plays a pivotal role in this process. By dynamically adjusting the mutation factor based on changes in fitness and iteration step size, the algorithm effectively balances global exploration and local exploitation, thereby improving both convergence efficiency and solution accuracy.

To enable the dynamic adjustment of the mutation factor, this study proposes a fitness-based adjustment mechanism. This mechanism takes into account the current range of fitness changes, the step size, and the global optimal fitness value. By modulating the mutation factor, the algorithm achieves enhanced diversity and convergence performance. The dynamic adjustment factor is mathematically defined in Eq. (13).

$$\mu_t = \mu_{t-1} * \left(1 + \beta * \frac{|f_t - f_{t-1}|}{f_{max}} + \gamma * \sqrt{|f_t - f_{t-1}|} - \alpha * \frac{t}{T}\right) \tag{13}$$

In Eq. (13), μ_t represents the adjustment factor for the current iteration, while μ_{t-1} is the adjustment factor for the previous iteration. f_t and f_{t-1} are the fitness values for the current and previous iterations, respectively. β, γ, and α are hyperparameters that control the update of the adjustment factor. These parameters are used to measure the linear and nonlinear weights of the fitness change, as well as the effect of the time step size. In this study, they are set to 0.1, 0.2, and 0.05, respectively. t is the current iteration number, and T is the maximum number of iterations. μ_t is constrained within the range (0, 1).

After each position update, to boost the algorithm's global exploration capability, this study introduces a mutation operation based on the dynamic adjustment factor. After calculating the dynamic adjustment factor μ_t, a normal distribution random disturbance N(0,1) is applied to fine-tune the individual positions. This disturbance operation not only enhances population diversity but also effectively prevents individuals from getting trapped in local optima, especially in the later stages of optimization. The specific position update equation is presented in Eq. (14).

$$x_{i,j}(t+1) = x_{i,j}(t+1) + \mu_t * N(0,1) \tag{14}$$

This operation enhances the algorithm's ability to avoid local optima by introducing random disturbances, especially during the later stages of population convergence or optimization. The dynamic adjustment factor integrates fitness differences with a square root term, significantly increasing the variation intensity during the early stages of optimization, when fitness differences are more pronounced. This approach introduces greater randomness into the search process, fostering a more thorough exploration of the global solution space.

As iterations progress, the influence of the adjustment factor is gradually reduced through a time attenuation mechanism, enabling the algorithm to transition from global exploration to local exploitation. This shift allows the algorithm to focus on refining the solution, thereby improving optimization accuracy.

Furthermore, the recursive calculation of the dynamic adjustment factor relies on the value from the previous generation. This design ensures smoother variation intensity, preventing abrupt fluctuations during the optimization process and maintaining stability. With this adaptive adjustment mechanism, the algorithm can adjust its search strategy in real-time based on changes in fitness, enabling it to maintain robust global search capabilities while avoiding premature convergence. Ultimately, this improves both optimization efficiency and solution quality.

D. The Construction Steps of IWOA-LSTM

This study employs the IWOA to optimize the LSTM neural network for predicting financial time series. Fig. 2 illustrates the overall process, with the steps detailed below:

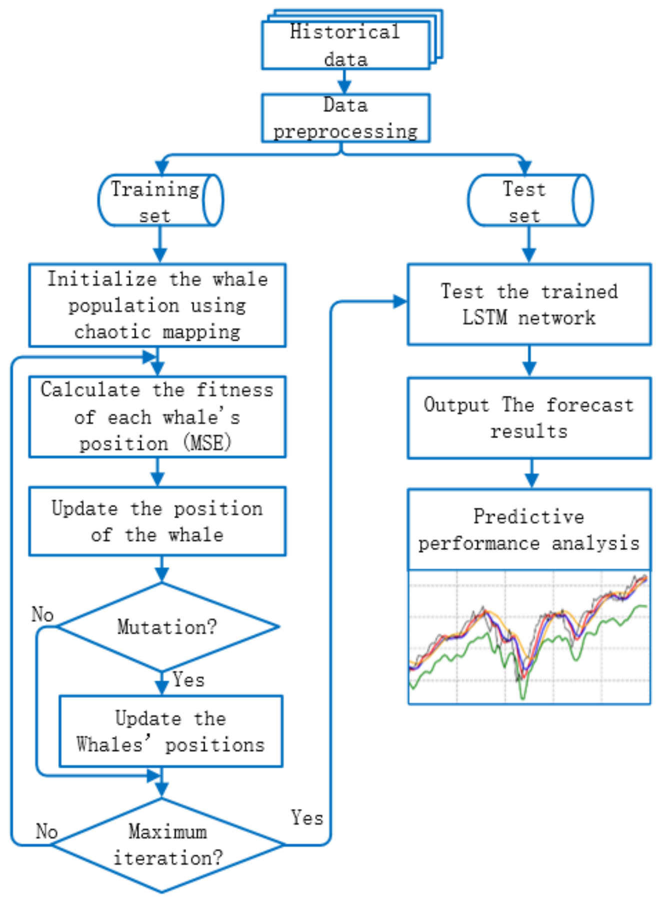

Fig. 2. Flowchart of IWOA-LSTM.

Step 1: Data preprocessing. Normalize the original time series data adopting Min-Max scaling to bring it into the [0, 1], then partition the data into training and test sets.

Step 2: Initialize IWOA and LSTM parameters. For IWOA, set the population size, search space boundaries (for LSTM's hidden units and learning rate), maximum iterations, and dynamic adjustment parameters (β, γ, α). Use the logistic map for chaotic initialization to enhance population diversity. For the LSTM model, set the initial parameters to ensure smooth optimization.

Step 3: Train and optimize the LSTM model using IWOA by minimizing the mean squared error (MSE) through iterative updates. Evaluate the fitness of each whale in the swarm, identifying the best fitness values for both individual whales and the global solution. Use IWOA's position update formula and dynamic adjustment factor to refine whale positions, balancing global exploration and local exploitation. Continue the process until the maximum iteration threshold is reached or the global fitness threshold is achieved.

Step 4: Apply the fine-tuned LSTM model to the test data to validate its prediction capabilities.

Step 5: Evaluate the prediction performance by applying the optimized LSTM model to forecast the financial time series data. The results are then measured against five performance indicators to evaluate the model's accuracy and effectiveness.

IV. Experiments

All programming works are implemented in the Python 3.9 environment. The computational experiments are performed on a system featuring a 12th Gen Intel(R) Core(TM) i5-1235U CPU, 16 GB of RAM, and operating on Windows 10.

A. Dataset

This study selected five globally representative stock indices as research subjects: the S&P 500 Index (Code: ^GSPC), Dow Jones Industrial Average (Code: ^DJI), FTSE 100 Index (Code: ^FTSE), Nasdaq Composite (Code: ^IXIC), and Shanghai Composite Index (Code: 000001.SS). These indices represent major economies in the Americas, Europe, and Asia, providing comprehensive insights into the global capital market.

Specifically, the S&P 500 Index and the Dow Jones Industrial Average represent the U.S. capital market comprehensively. The S&P 500 Index includes 500 companies with the largest market capitalizations and significant industry representation, providing a broad view of the market. In contrast, the Dow Jones Industrial Average Index focuses on 30 prominent companies, often referred to as blue-chip stocks, representing major sectors of the U.S. economy. The FTSE 100 Index reflects the performance of the largest British companies by market capitalization and serves as a key benchmark for the European market. The Nasdaq Composite Index, heavily weighted by technology stocks, highlights global trends in technological innovation and growth. Lastly, the Shanghai Composite Index is a vital indicator of mainland China's capital market, encompassing all listed stocks on the Shanghai Stock Exchange.

The experimental data is obtained from https://finance.yahoo.com/ and spans a period of 10 years, from October 23, 2014, to October 21, 2024. This timeframe encompasses significant global economic events, including the financial crisis triggered by the COVID-19 pandemic in 2020, subsequent recovery phases, and various national policy adjustments, which resulted in substantial market fluctuations. The dataset includes daily recorded basic information, providing a solid foundation for model training and testing. These data facilitate the assessment of the model's adaptability and robustness within a complex and volatile market environment. In addition, to verify the predictive ability of the model when external factors (e.g., macroeconomic indicators, social sentiment) are not introduced, the experimental design uses only historical price data for prediction. This design highlights the model's intrinsic performance in financial time series analysis, avoids interference from external variables, and objectively evaluates the performance of IWOA-LSTM in volatile market environments. The historical trends of the five stock indices are shown in Fig. 3. The detailed statistical analysis of the closing prices for each stock index is presented in Table I.

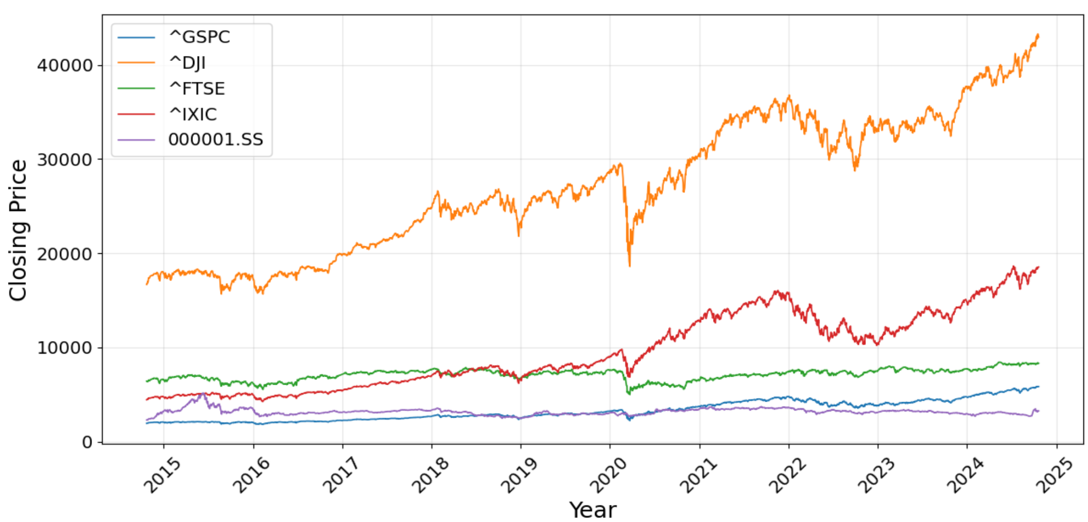

Fig. 3. Historical closing price charts for five stock indices.

TABLE I. DESCRIPTIVE STATISTICS OF STOCK CLOSING PRICES

Name	Count	Max	Min	Range	Mean	Std	Kurtosis	Skewness	Normality Test Statistic	Normality Test P-Value
^GSPC	2515	5864.67	1829.08	4035.59	3280.57	1035.14	-0.88	0.49	403.10	0.00
^DJI	2515	43275.91	15660.18	27615.73	27194.68	7082.78	-1.12	0.13	1129.42	0.00
^FTSE	2524	8445.80	4993.90	3451.90	7109.40	601.09	-0.02	-0.49	92.50	0.00
^IXIC	2515	18647.45	4266.84	14380.61	9573.88	3959.34	-1.11	0.41	1101.31	0.00
000001.SS	2427	5166.35	2290.44	2875.91	3183.23	349.92	5.52	1.45	782.03	0.00

From Fig. 3 and Table I, it is evident that the closing prices of these five stock indices (^GSPC, ^DJI, ^FTSE, ^IXIC, and 000001.SS) exhibit obvious nonlinear characteristics and high volatility, with the price trends showing irregularity and substantial noise. For example, the fluctuation range of ^DJI index is as high as 27615.73, which is significantly higher than that of the other indices, indicating extreme volatility. Although some trend changes are observed in the data, the overall price fluctuation demonstrates strong uncertainty, and the fluctuation pattern cannot be simply summarized as a linear relationship. The price fluctuations of these indices are influenced by a variety of complex factors, displaying both randomness and nonlinear characteristics. Traditional linear models struggle to effectively capture these intricate dynamics.

Conventional forecasting approaches face significant limitations in addressing the nonlinear and complex nature of stock market data, particularly in highly volatile time series. These methods, based on linear assumptions, are inadequate for capturing long-term dependencies and nonlinear trends. To overcome these challenges, this study employs the LSTM model, which excels at handling complex time series data due to its unique architecture and memory mechanism. By capturing long-term dependencies and nonlinear patterns, LSTM outperforms traditional methods, offering improved prediction accuracy and stability in the face of noise and complexity in stock market data.

B. Data Preprocessing

Data preprocessing is a critical component of data analysis, as it significantly influences model performance and prediction accuracy. In this study, the raw stock index data are cleaned to eliminate null and duplicate values, ensuring the integrity and reliability of the dataset. Subsequently, the Min-Max normalization technique is applied to scale the daily closing prices to the range [0, 1]. This normalization step mitigates the impact of varying data dimensions on model training. The equation used for normalization is in Eq. (15).

$$x_i = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{15}$$

In Eq. (15), x_i denotes the normalized data, while x_{max} and x_{min} represent the maximum and minimum value in the financial time series, respectively.

After the prediction is completed, the prediction results need to be transformed using inverse normalization. The inverse normalization equation is in Eq. (16).

$$x = (x_{max} - x_{min})\, x_i + x_{min} \tag{16}$$

We prepare the data and normalize it to ensure uniform scaling and provide a stable input for the mode training.

C. Evaluation Metrics

Measuring prediction accuracy is a multifaceted task, and no single evaluation metric applies universally across different application scenarios. To comprehensively assess the model's prediction performance, this study employs five widely used evaluation metrics: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Coefficient of Determination (R^2), and Explained Variance Score (EVS).

These evaluation metrics can be classified into error measures and fitting measures. Error measures, including RMSE, MAE, and MAPE, primarily assess the differences between predicted and actual values. Fitting measures, including R^2 and EVS, evaluate the model's fit. Specifically, the closer the R^2 value is to 1, the better the model's fit, while the closer the EVS value is to 1, the stronger the model's ability to explain the data. These five metrics together offer a comprehensive and accurate reflection of the model's prediction accuracy and fitting ability. The five evaluation metrics are expressed by the following equations:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{17}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{18}$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{19}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \tag{20}$$

$$EVS = 1 - \frac{Var(y_i - \hat{y}_i)}{Var(y)} \tag{21}$$

Among them, y_i stands for the true value, $\hat{y}_i$ represents the predicted value, $\bar{y}_i$ denotes the mean of the financial time series, and Var () refers to the variance.

To construct the input data for the LSTM model, this study employs sliding window method for time series and evaluates the impact of different time step settings on model's effectiveness. Specifically, the sliding window method uses fixed time steps to extract sequential features from stock data.

Four different time step settings are tested: 10, 20, 30, and 60. The MSE and training time of the model are calculated for each setting.

The closing price data are divided into training and test sets based on an 80:20 ratio, using various time step configurations (time_steps = 10, 20, 30, and 60). The LSTM model is trained on the training set and generates predictions for the test set. For each time step configuration, the training time (in seconds) and the MSE on the test set are recorded. Table II presents an overview of the experimental results.

TABLE II. IMPACT OF LSTM TIME STEP ON PREDICTION PERFORMANCE

Time Steps	MSE	Training Time (s)
10	0.000463	9.974107
20	0.000577	12.020250
30	0.000701	13.437092
60	0.000464	20.137358

The experimental results reveal significant variations in both the MSE and the training duration of the model across different time step configurations. Specifically, when the time step is 10 days, the MSE of the model is 0.000463, and the training time is approximately 9.97 seconds. When the time step is increased to 60 days, the MSE slightly increases to 0.000464, while the training time rises significantly to about 20.14 seconds. Although the MSE at 60 days is similar to that at 10 days, the longer training time notably affects computational efficiency. Therefore, considering the need to balance prediction accuracy with computational efficiency, a 10-day time step is selected as the optimal configuration.

Next, the IWOA algorithm is applied to optimize the number of LSTM units and the learning rate. By leveraging IWOA, the model can dynamically fine-tune these hyperparameters, enhancing both prediction accuracy and computational efficiency. To evaluate the effectiveness of IWOA optimization, this study compares it with other popular models, including WOA-LSTM, LSTM, and RNN. Table III shows the models compared in this study and their parameters.

TABLE III. MODELS COMPARED IN THIS RESEARCH

Model	Description of model parameters
IWOA-LSTM	time_steps=10，epochs=50，batch_size=32，population_size=5，units∈ [10, 100]，learning_rate∈ [0.0001, 0.01]
WOA-LSTM	time_steps=10，epochs=50，batch_size=32，population_size=5，units∈ [10, 100]，learning_rate∈ [0.0001, 0.01]
LSTM	time_steps=10，units=50, learning_rate=0.001，epochs=50，Dropout=0.2， batch_size=32
RNN	time_steps=10，units=50, learning_rate=0.001，epochs=50，Dropout=0.2， batch_size=32

D. S&P500 Forecasting

This study first selects the S&P 500 index (^GSPC) as the target for prediction. The dataset contains 2,515 records, spanning from October 23, 2014, to October 21, 2024, with daily closing prices. For model training, a retrospective period of 10 days is used, meaning that the closing prices from the past 10 days are employed in the prediction process. To thoroughly assess the prediction performance of each model, this study adopts five different evaluation indicators. Each model is evaluated based on these five indicators, with the best performance metrics marked in bold. The prediction comparison results for the various models on the S&P 500 index are presented in Table IV.

TABLE IV. COMPARISON RESULTS OF PREDICTIONS BETWEEN VARIOUS MODELS OF ^GSPC

Model	RMSE	MAE	MAPE	R2	EVS
IWOA-LSTM	**48.1826**	**37.8081**	**0.9377%**	**0.9935**	**0.9936**
WOA-LSTM	55.6435	44.2542	0.9638%	0.9913	0.9916
LSTM	70.4543	55.8142	1.2216%	0.9860	0.9863
RNN	93.7222	76.6777	1.5764%	0.9752	0.9869

As observed from the results in Table IV, the effectiveness of the IWOA-LSTM model is significantly superior to that of the other models across all evaluation metrics. Specifically, the IWOA-LSTM model outperforms WOA-LSTM in RMSE, MAE, MAPE, R^2, and EVS by 13.41%, 14.57%, 0.03%, 0.22%, and 0.20%, respectively. Furthermore, IWOA-LSTM shows improvements over the LSTM model by 31.61%, 32.26%, 0.28%, 0.76%, and 0.74%, and is 48.59%, 50.69%, 0.64%, 1.88%, and 0.68% better than the RNN model in the same metrics.

In addition to evaluating prediction accuracy, this study also investigates the computational efficiency of the IWOA-LSTM model by comparing its runtime performance with that of the standard WOA during hyperparameter optimization and model training. Fig. 4 presents the comparison of their runtime performance.

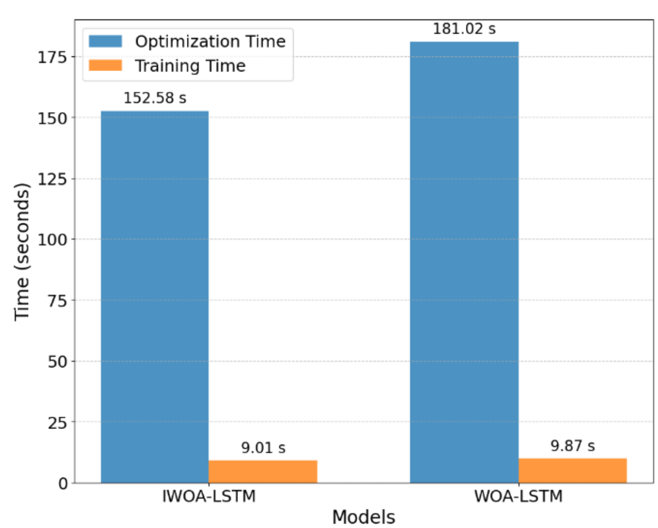

Fig. 4. Runtime comparison.

Fig. 4 compares the optimization and training times of the IWOA-LSTM and WOA-LSTM models, highlighting the performance differences. The optimization time for IWOA-LSTM is 152.58 seconds, while WOA-LSTM takes 181.02 seconds, indicating that IWOA-LSTM is more efficient in parameter optimization. This improvement is attributed to the use of logistic chaotic mapping and a dynamic adjustment factor mechanism, which enhance search precision and reduce unnecessary iterations. For training time, IWOA-LSTM takes 9.01 seconds, while WOA-LSTM takes 9.87 seconds. Although the difference is small, IWOA-LSTM shows a slight advantage, proving that improved optimization efficiency does not slow down the training process. The results show that IWOA-LSTM improves operational efficiency while maintaining predictive performance and is able to converge on complex datasets relatively quickly.

Overall, the results of these experiments on the ^GSPC stock index show that the IWOA-LSTM model performs well across all evaluation metrics. In addition to its excellent performance on these metrics, the IWOA-LSTM model also shows an advantage in hyperparameter optimization time compared to the original WOA, reflecting an improvement in the model's computational efficiency during the optimization process. This highlights the effectiveness of optimizing the LSTM hyperparameters, which not only leads to more accurate predictions but also improves computational efficiency.

Next, to provide a more intuitive view of the fitting performance of each model, the final fitting comparison results are presented in Fig. 5 and Fig. 6.

Fig. 5 demonstrates the fitting results of the ^GSPC index prediction based on different models. The curves in the figure illustrate the prediction results of the IWOA-LSTM, WOA-LSTM, LSTM, and RNN models, respectively, and are compared with the actual price curves. It can be observed that the prediction curves of the IWOA-LSTM model are closest to the actual prices, especially in most time intervals, where the prediction curves of the IWOA-LSTM almost coincide with the actual price curves, showing very high prediction accuracy. In contrast, although the WOA-LSTM model also shows relatively accurate predictions, the discrepancy between the predicted and actual values becomes more pronounced during periods of high volatility. Nonetheless, despite slight deviations in more volatile markets, the overall performance remains robust, suggesting that the model is able to effectively capture the general trend. The LSTM and RNN models show relatively lower prediction accuracies, especially in regions of high price volatility, where the predicted curves of both models deviate more from the actual prices. This suggests that the LSTM and RNN models are not as effective as the IWOA-LSTM and WOA-LSTM models in capturing complex patterns.

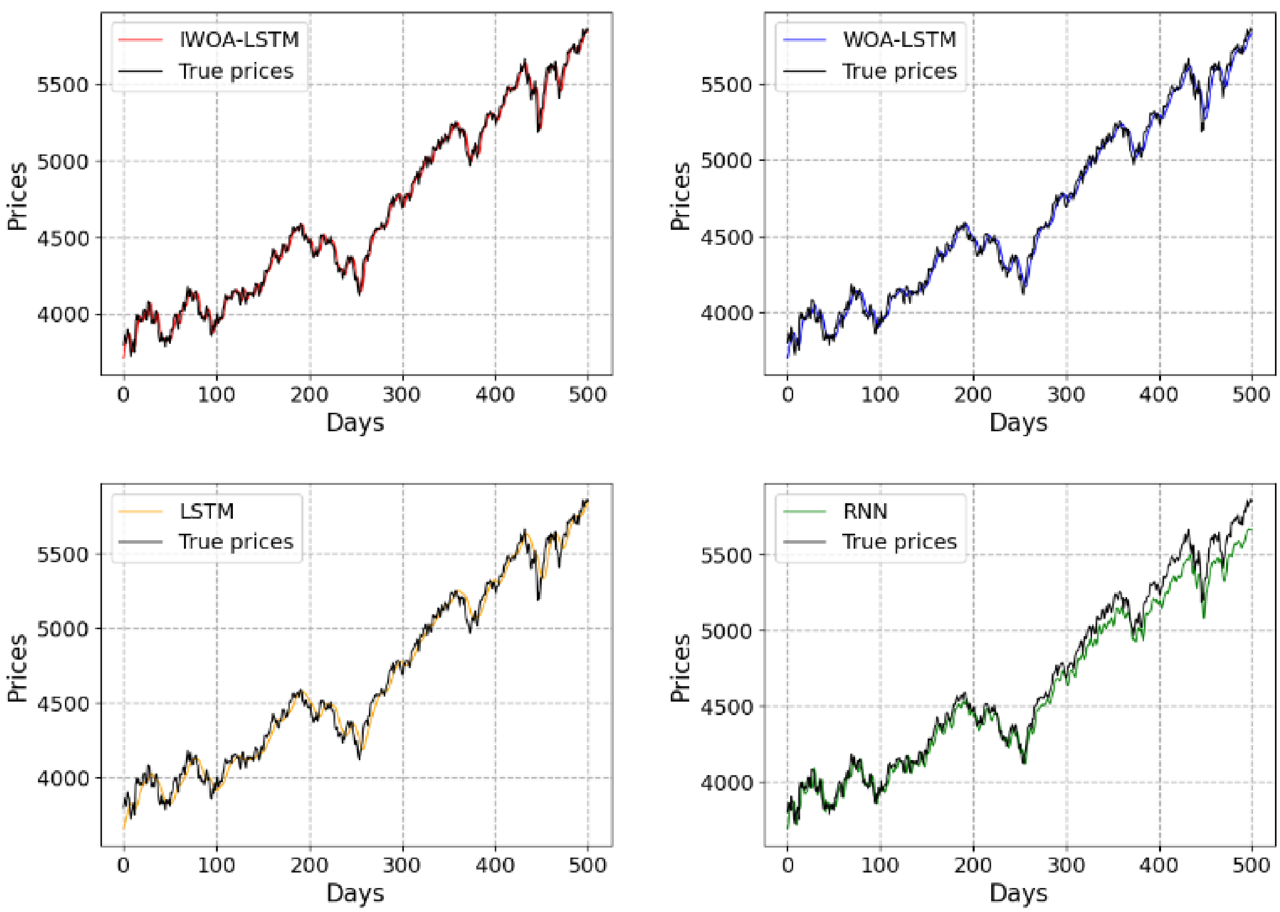

Fig. 5. Individual forecasts of four models on the ^GSPC index.

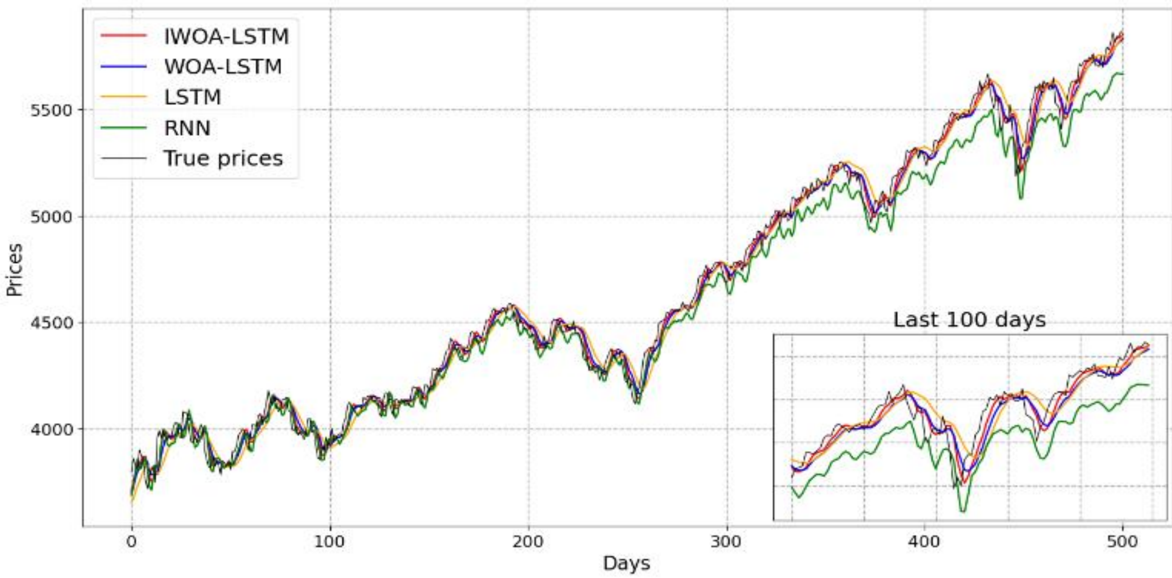

Fig. 6. Comparison of forecasts from four models on the ^GSPC index.

The illustration in Fig. 6 shows the forecast results for the last 100 days. During this period, the IWOA-LSTM prediction curve continues to outperform the other models, especially in short-term price fluctuations, where IWOA-LSTM more accurately captures the trend of stock prices. In contrast, the fitting effect of the RNN model is poor, particularly during periods of significant stock price fluctuations, where its prediction error is quite noticeable.

Overall, IWOA-LSTM performs particularly well in the ^GSPC index prediction task. It is better at capturing the complex dynamic patterns in the time series and providing high-precision prediction results, which demonstrates its advantages in financial time series forecasting.

To further validate the predictive ability of the IWOA-LSTM model, this study compares it with other improved WOA algorithms proposed by Shao [27] and Guan [28]. Apart from the improved algorithms discussed in the paper, all other parameter settings are consistent with those used in this study. Additionally, this study incorporates PSO-LSTM [29] and GA-LSTM [30], two advanced swarm intelligence optimization models. A comparative analysis is performed to evaluate the prediction performance of different optimization methods on the ^GSPC index. The comparison of prediction performance results is presented in Table V.

TABLE V. COMPARISON OF PREDICTION PERFORMANCE OF IWOA-LSTM MODEL AND OTHER IMPROVED LSTM MODEL

Model	Optimization Details	RMSE	MAE	R2
IWOA-LSTM (Proposed in this study)	Logistc chaotic mapping; dynamic adjustment factor mechanism	**48.1826**	**37.8081**	**0.9935**
IWOA-LSTM [27]	Tent chaotic mapping; adaptive weight	57.7395	47.2308	0.9906
WOA-BiLSTM [28]	Whale optimization algorithm optimized Bidirectional LSTM	65.0222	53.2598	0.9881
PSO-LSTM [29]	Particle swarm optimized LSTM	49.1197	38.8116	0.9931
GA-LSTM [30]	Genetic algorithm optimized LSTM	52.7665	42.0509	0.9921

From Table V, the IWOA-LSTM model achieves better prediction results compared to other models. With the integration of the logistic chaotic map and dynamic adjustment mechanism, IWOA-LSTM demonstrates improved prediction accuracy and stability. Although the other models have been enhanced, their accuracy and effectiveness in predicting financial time series do not fully match those of IWOA-LSTM. This suggests that IWOA-LSTM offers certain advantages in forecasting financial time series.

E. The Robustness and Reliability Verification

In order to conduct a deeper validation of the IWOA-LSTM model's robustness and reliability, this study performs experiments on other four famous stock indices: ^DJI, ^FTSE, ^IXIC, and 000001.SS. The corresponding experimental results are presented in Table VI to Table IX and Fig. 7 to Fig. 10.

TABLE VI. COMPARISON RESULTS OF PREDICTIONS BETWEEN VARIOUS MODELS OF ^DJI

Model	RMSE	MAE	MAPE	R2	EVS
IWOA-LSTM	**344.4983**	**277.7575**	**0.7700%**	**0.9870**	**0.9888**
WOA-LSTM	436.9144	357.2306	0.9766%	0.9791	0.9829
LSTM	483.1204	379.3579	1.0572%	0.9744	0.9749
RNN	632.4826	513.0577	1.3752%	0.9562	0.9732

TABLE VII. COMPARISON RESULTS OF PREDICTIONS BETWEEN VARIOUS MODELS OF ^FTSE

Model	RMSE	MAE	MAPE	R2	EVS
IWOA-LSTM	**73.6150**	**54.8546**	**0.7112%**	**0.9502**	**0.9502**
WOA-LSTM	73.9682	55.0725	0.7130%	0.9497	0.9500
LSTM	183.0727	157.7665	1.9980%	0.6917	0.8732
RNN	77.3710	60.2541	0.7754%	0.9449	0.9526

TABLE VIII. COMPARISON RESULTS OF PREDICTIONS BETWEEN VARIOUS MODELS OF ^IXIC

Model	RMSE	MAE	MAPE	R2	EVS
IWOA-LSTM	**208.4710**	**167.3047**	**1.1916%**	**0.9921**	**0.9923**
WOA-LSTM	259.5208	211.9485	1.4697%	0.9878	0.9899
LSTM	324.4165	264.5356	1.8428%	0.9810	0.9836
RNN	327.8000	260.9647	1.7722%	0.9806	0.4920

TABLE IX. COMPARISON RESULTS OF PREDICTIONS BETWEEN VARIOUS MODELS OF 000001.SS

Model	RMSE	MAE	MAPE	R2	EVS
IWOA-LSTM	**38.1677**	**25.8819**	**0.8397%**	**0.9371**	**0.9372**
WOA-LSTM	42.3903	28.5810	0.9294%	0.9224	0.9229
LSTM	56.3180	37.6384	1.2245%	0.8631	0.8631
RNN	43.9960	30.4456	0.9893%	0.9164	0.9175

As shown by the comparison results in Table VI to Table IX and Fig. 7 to Fig. 10, the IWOA-LSTM model demonstrates superior prediction accuracy and a better fit than other models in predicting four different stock indices (^DJI, ^FTSE, ^IXIC, and 000001.SS). The IWOA-LSTM model achieves lower errors and improved accuracy across multiple evaluation metrics (RMSE, MAE, MAPE, R^2, EVS). In particular, its RMSE and MAE values indicate better accuracy compared to WOA-LSTM.

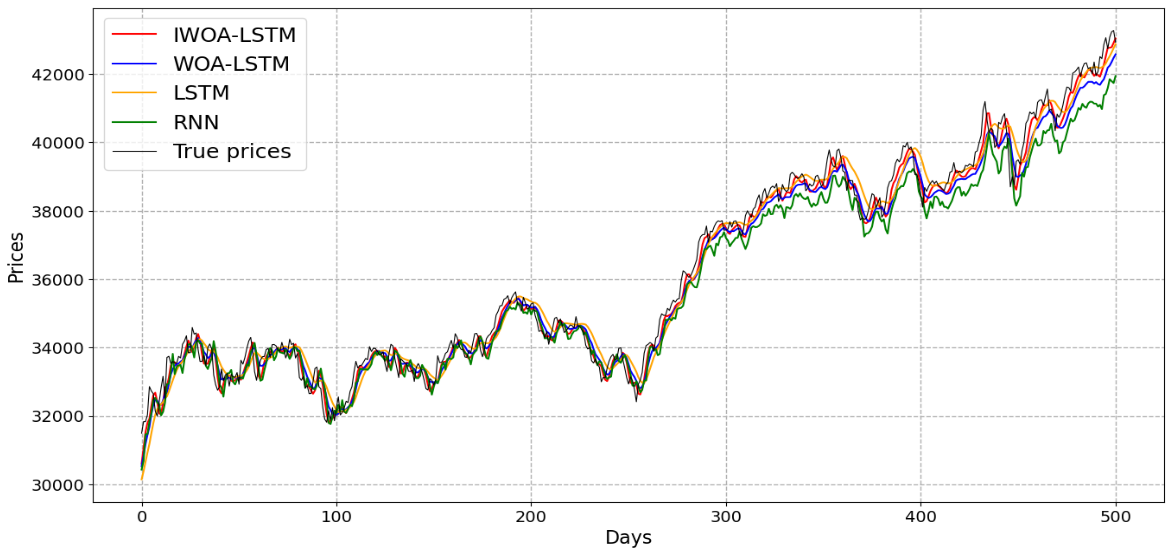

Fig. 7. Comparison of forecasts from four models on the ^DJI index.

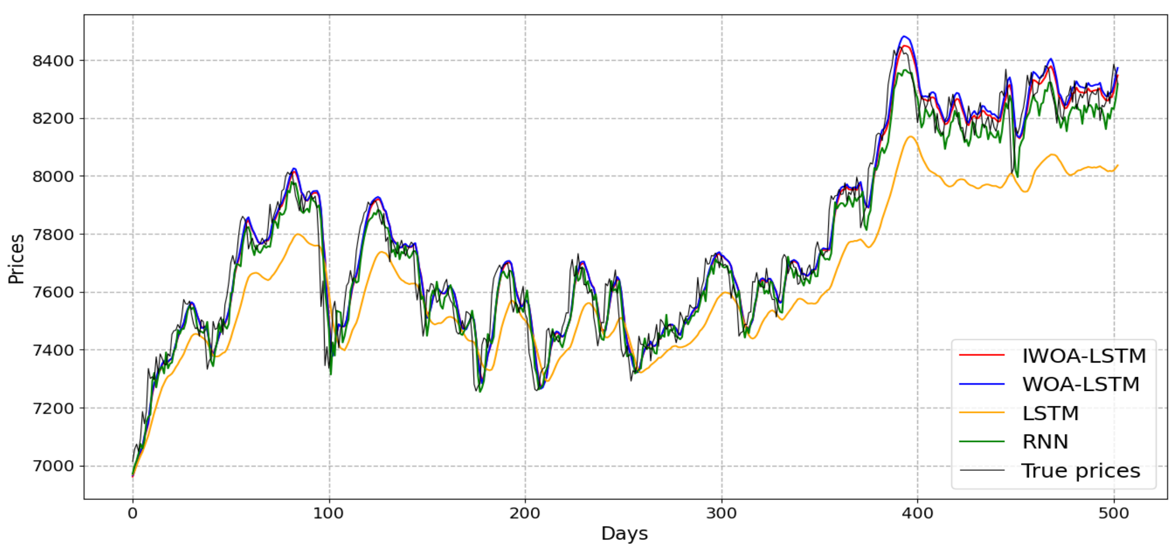

Fig. 8. Comparison of forecasts from four models on the ^FTSE index.

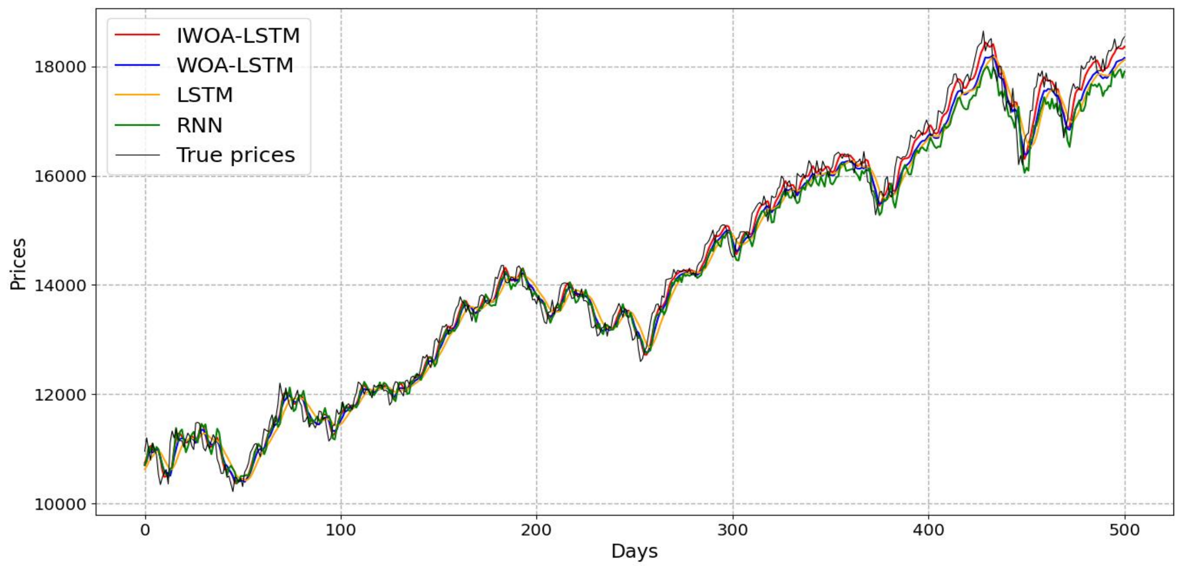

Fig. 9. Comparison of forecasts from four models on the ^IXIC index.

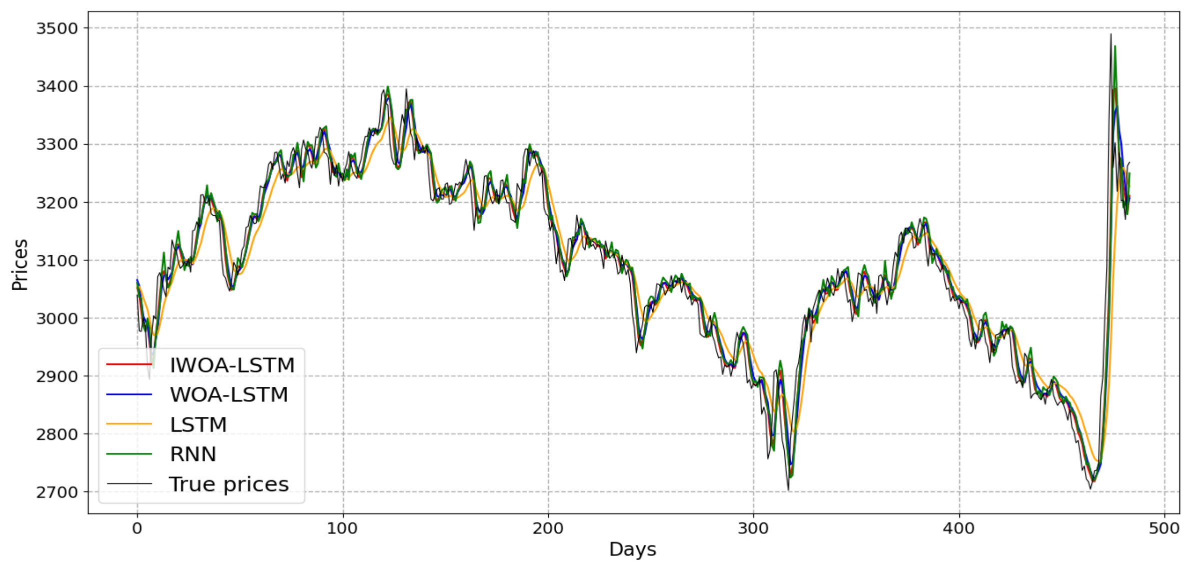

Fig. 10. Comparison of forecasts from four models on the 000001.SS index.

When compared with the LSTM and RNN models, the IWOA-LSTM model shows clear advantages, particularly in capturing the volatility of financial markets and the complexity of time series data. It demonstrates strong adaptability in the ^DJI, ^FTSE, and ^IXIC and also performs well in predicting the Shanghai Composite Index (000001.SS), with RMSE, MAE, and MAPE values lower than those of the other comparative models. Overall, the IWOA-LSTM model improves prediction accuracy and stability through optimized hyperparameter settings, demonstrating strong potential for application in financial time series forecasting, particularly in stock index prediction.

V. Conclusion

A novel model integrating the Improved Whale Optimization Algorithm (IWOA) with Long Short-Term Memory (LSTM) is proposed to enhance the accuracy of stock market index forecasting. This model employs chaotic map initialization and a dynamic adjustment mechanism to optimize the LSTM network's parameters, thereby improving prediction which confirms performance. Chaotic assignment improves the global search capability of the algorithm and enables a thorough exploration of the solution space. Additionally, the dynamic adjustment factor increases WOA efficiency, optimizes LSTM hyperparameters, and improves prediction accuracy and stability.

The study focuses on five representative stock market indexes (^GSPC, ^DJI, ^FTSE, ^IXIC, and 000001.SS) and evaluates the model's performance adopting five essential evaluation metrics: RMSE, MAE, MAPE, R^2, and EVS. These metrics comprehensively evaluate the model's prediction ability from the perspectives of error magnitude, absolute and relative error, goodness of fit, and variance explained. Experimental results show that the IWOA-LSTM model outperforms WOA-LSTM, LSTM and RNN in all five metrics. This highlights its superior accuracy, robustness and stability.

Although this study presents an innovative approach to stock index prediction and offers valuable contributions, the IWOA-LSTM model has certain limitations. Its performance depends heavily on the range of hyperparameter values, which need to be set based on empirical experience to ensure optimal results. Additionally, although the IWOA-LSTM model reduces runtime compared to the traditional WOA algorithm, the computational cost remains relatively high. Furthermore, the validation is conducted on only five representative stock indices, limiting the sample size.

Future research will focus on expanding the dataset to include additional stock indices and broader financial market data to assess the model's generalizability and robustness. The model will also be applied to other time series tasks, such as energy forecasting and medical trend analysis, to evaluate its cross-industry adaptability and performance. Additionally, advanced optimization techniques, including the integration of metaheuristic algorithms and hybrid optimization methods, will be explored to enhance feature selection and hyperparameter tuning. Modal decomposition methods will be studied to decompose and reconstruct time series data, reducing noise and improving the model’s ability to detect market fluctuations, thereby enhancing prediction accuracy and stability. Further research will also explore the correlations between the stock market and other financial markets to analyze the potential impact of external market fluctuations. Moreover, external features, such as macroeconomic indicators and social sentiment, will be incorporated to assess their influence on predictions and improve the model’s adaptability in complex market environments.

ACKNOWLEDGMENT

The research is funded by Guangdong Province Key Discipline Research Capacity Enhancement Project (No. 2022ZDJS146); Dongguan Science and Technology of Social Development Program (No. 20221800902782).

REFERENCES

[1] M. A. Rahim, M. Mushafiq, S. D. Khan, R. Ullah, S. Khan, and M. Ishaque, “Technical analysis-based unsupervised intraday trading DJIA index stocks: is it profitable in long term?, ” Applied Intelligence, vol. 55, no. 2, 2025, pp. 1–12.

[2] W. Chen, C. K. Yeo, C. T. Lau, and B. S. Lee, “Leveraging social media news to predict stock index movement using RNN-boost,” Data & Knowledge Engineering, vol. 118, 2018, pp. 14–24.

[3] M. M. Al Haromainy, D. A. Prasetya, and A. P. Sari, “Improving performance of RNN-based models with genetic algorithm optimization for time series data,” TIERS Information Technology Journal, vol. 4, no. 1, 2023, pp. 16–24.

[4] X. Zhang, N. Gu, J. Chang, and H. Ye, “Predicting stock price movement using a DBN-RNN,” Applied Artificial Intelligence, vol. 35, no. 12, 2021, pp. 876–892.

[5] A. Q. Md, S. Kapoor, C. J. AV, A. K. Sivaraman, K. F. Tee, H. Sabireen, and N. Janakiraman, “Novel optimization approach for stock price forecasting using multi-layered sequential LSTM,” Applied Soft Computing, vol. 134, 2023, p. 109830.

[6] Z. Yang, Z. Zeng, K. Wang, S. S. Wong, W. Liang, M. Zanin, and J. He, “Modified SEIR and AI prediction of the epidemics trend of COVID-19 in China under public health interventions,” Journal of Thoracic Disease, vol. 12, no. 3, 2020, pp. 165–174.

[7] J. Q. Wang, Y. Du, and J. Wang, “LSTM based long-term energy consumption prediction with periodicity, ” Energy, vol. 197, 2020, p. 117197.

[8] Z. Li, H. Yu, J. Xu, J. Liu, and Y. Mo, “Stock market analysis and prediction using LSTM: A case study on technology stocks,” Innovations in Applied Engineering and Technology, 2023, pp. 1–6.

[9] P. Singh, M. Jha, M. Sharaf, M. A. El-Meligy, and T. R. Gadekallu, “Harnessing a hybrid CNN-LSTM model for portfolio performance: A case study on stock selection and optimization,” IEEE Access, vol. 11, 2023, pp. 104000–104015.

[10] Y. Yu, X. Si, C. Hu, and J. Zhang, “A review of recurrent neural networks: LSTM cells and network architectures,” Neural Computation, vol. 31, no. 7, 2019, pp. 1235–1270.

[11] A. G. Nikolaev and S. H. Jacobson, “Simulated annealing,” in Handbook of Metaheuristics, 3rd ed., G. T. Rado and H. Suhl, Eds. New York: Academic, 2010, pp. 1–39.

[12] Y. Ji, A. W. C. Liew, and L. Yang, “A novel improved particle swarm optimization with long-short term memory hybrid model for stock indices forecast,” IEEE Access, vol. 9, 2021, pp. 23660–23671.

[13] X. Chen, L. Cheng, C. Liu, Q. Liu, J. Liu, Y. Mao, and J. Murphy, “A WOA-based optimization approach for task scheduling in cloud computing systems,” IEEE Systems Journal, vol. 14, no. 3, 2020, pp. 3117–3128.

[14] Y. Zhang and S. Yang, “Prediction on the highest price of the stock based on PSO-LSTM neural network,” in 2019 3rd International Conference on Electronic Information Technology and Computer Engineering (EITCE), Oct. 2019, pp. 1565–1569.

[15] M. Clerc and J. Kennedy, “The particle swarm-explosion, stability, and convergence in a multidimensional complex space,” IEEE Transactions on Evolutionary Computation, vol. 6, no. 1, 2002, pp. 58–73.

[16] H. Xin and H. Yu, “WOA-LSTM CSI 500 forecast model based on Baidu Index,” in International Conference on Business Intelligence and Information Technology, Singapore, Dec. 2023, Springer Nature Singapore, pp. 139–147.

[17] M. W. Hasan, “Building an IoT temperature and humidity forecasting model based on long short-term memory (LSTM) with improved whale optimization algorithm,” Memories-Materials, Devices, Circuits and Systems, vol. 6, 2023, p. 100086.

[18] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” IEEE Transactions on Neural Networks, vol. 5, no. 2, 1994, pp. 157–166.

[19] A. Graves, “Supervised sequence labelling,” in Springer Berlin Heidelberg, 2012, pp. 5–13.

[20] S. Mirjalili and A. Lewis, “The whale optimization algorithm,” Advances in Engineering Software, vol. 95, 2016, pp. 51–67.

[21] F. S. Gharehchopogh and H. Gholizadeh, “A comprehensive survey: Whale Optimization Algorithm and its applications,” Swarm and Evolutionary Computation, vol. 48, 2019, pp. 1–24.

[22] M. H. Nadimi-Shahraki, H. Zamani, Z. A. Varzaneh, and S. Mirjalili, “A systematic review of the whale optimization algorithm: theoretical foundation, improvements, and hybridizations,” Archives of Computational Methods in Engineering, vol. 30, no. 7, 2023, pp. 4113–4159.

[23] Z. Che, C. Peng, and C. Yue, “Optimizing LSTM with multi-strategy improved WOA for robust prediction of high-speed machine tests data,” Chaos, Solitons & Fractals, vol. 178, 2024, p. 114394.

[24] D. Prasad, A. Mukherjee, and V. Mukherjee, “Temperature dependent optimal power flow using chaotic whale optimization algorithm,” Expert Systems, vol. 38, no. 4, 2021, p. e12685.

[25] D. Yousri, D. Allam, and M. B. Eteiba, “Chaotic whale optimizer variants for parameters estimation of the chaotic behavior in Permanent Magnet Synchronous Motor,” Applied Soft Computing, vol. 74, 2019, pp. 479–503.

[26] E. Mezura-Montes and C. A. C. Coello, “Constraint-handling in nature-inspired numerical optimization: past, present and future,” Swarm and Evolutionary Computation, vol. 1, no. 4, 2011, pp. 173–194.

[27] C. Shao and J. Ning, “Construction and application of carbon emission prediction model for China’s textile and garment industry based on

improved WOA-LSTM," J. Beijing Inst. Fash. Technol. (Nat. Sci. Ed.), vol. 43, 2023, pp. 73–81.

[28] X. Gua, X. Gong, X. Yang, et al., "基于 WOA-BiLSTM 模型的短期光伏出力预测 [Short-term photovoltaic output forecasting based on WOA-BiLSTM model]," Electric Power and Energy, vol. 44, no. 6, 2023, pp. 613-616+653.

[29] W. Jia, Y. Zhang, Z. Wei, Z. Zheng, and P. Xie, "Daily reference evapotranspiration prediction for irrigation scheduling decisions based on the hybrid PSO-LSTM model," Plos One, vol. 18, no. 4, 2023, p. e0281478.

[30] X. Liang, "Stock Market Prediction with RNN-LSTM and GA-LSTM," SHS Web of Conferences, vol. 196, 2024, p. 02006, EDP Sciences.