

THE OPTIMIZED GATE RECURRENT UNIT BASED ON
IMPROVED EVOLUTIONARY ALGORITHM TO PREDICT
STOCK MARKET RETURNS *

CHAO LIU¹, FENGFENG GAO^{*1}, QI ZHAO² AND MENGWAN
ZHANG¹

Abstract. In order to accelerate the learning ability of neural network structure parameters and improve the prediction accuracy of deep learning algorithms, an evolutionary algorithm, based on a prior Gaussian mutation(PGM) operator, is proposed to optimize the structure parameters of a gated recurrent unit (GRU) neural network. In this algorithm, the sensitivity learning process of GRU model parameters into the Gaussian mutation operator, used the variance of the GRU model parameter training results as the Gaussian mutation variance to generate the optimal individual candidate set. Then, the optimal GRU neural network structure is constructed using the evolutionary algorithm of the prior Gaussian mutation operator. Moreover, the PGM-EA-GRU algorithm is applied to the prediction of stock market returns. Experiments show that the prediction model effectively overcomes the GRU neural network, quickly falling into a local optimum and slowly converging. Compared to the RF, SVR, RNN, LSTM, GRU, and EA-GRU benchmark models, the model significantly improves the searchability and prediction accuracy of the optimal network structure parameters. It also validates the effectiveness and the progressive nature of the PGM-EA-GRU model proposed in this paper with stock market return prediction.

Keywords: Gate Recurrent Unit (GRU), Evolutionary Algorithm (EA), Prior Gaussian Mutation (PGM), Parameter Optimization.

Mathematics Subject Classification. 91B84, 68W50

March 15, 2023.

* *Corresponding author: gao.flora@163.com.*

¹ School of Economics and Management, Beijing University of Technology, Beijing, China.

² Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China.

© EDP Sciences 2001

1. INTRODUCTION

As an essential part of the financial system, the prediction of stock market return has become a hot issue of governments, academia and stock analysts. Accurate prediction of stock market returns is of great importance to optimize asset allocation and risk management. Thus, is the stock market truly predictable? First, scholars analyzed the predictability of the stock market based on a theoretical perspective, and their research found that economic variables can be used to predict the stock market. Welch & Goyal [1] first proposed exploring the predictability of the stock market based on the empirical perspective and using the traditional econometric model to test the predictive effect of economic variables on stock market returns. They demonstrated that stock market returns are almost unpredictable. This result is basically consistent with the efficient market hypothesis: the arbitrage behavior of rational investors will continuously correct mispricing in the market, thereby obtaining excess profits in the capital market. In fact, the stock market does not conform to the strong efficient market hypothesis, and stock market returns are predictable [2].

As a typical nonlinear complex system, the stock market is vulnerable to market uncertainty, network media, investor sentiment and other factors. Changes in the stock market have the characteristics of randomness, nonlinearity and long memory, which increases the difficulty of capital market supervision. With the development of artificial intelligence and big data technology, the time series data of the stock market are high frequency, high-dimensional multisource data, and there are complex interactions and correlations between various influencing factors. This makes the traditional econometric model, machine learning and other methods show certain limitations while processing stock market time series data, and it also brings more challenges to stock market return forecasting [3, 4].

In recent years, deep learning algorithms have shown certain advantages in time series prediction research. A deep learning algorithm has the characteristics of a multilayer network structure, stronger learning ability for high-dimensional data features and good generalization ability [5]. In particular, a recurrent neural network has the characteristics of long-term memory, which can solve the problem of long-term dependent learning to a certain extent and is more suitable for the modelling of stock market time series, and good performance in terms of stock market prediction has been achieved using this network [6]. However, there are also many superparameters in a recurrent neural network, which makes the training process more difficult; in addition, the calculation cost is high, the convergence speed is slow, it is easy to fall into the local optimum, and it is difficult to accurately find the optimal neural network structure parameters [7, 8]. In view of the slow convergence rate of neural network structure parameter optimization, existing research has shown that evolutionary algorithms can effectively overcome the problem of neural network structure parameter optimization by iterating on the candidate set population of the objective function and improving the convergence speed and prediction performance of the algorithm; however, the robustness and global search ability of these algorithms are poor [9–11].

To solve the above issues, this paper proposes an EA algorithm, based on prior parameters controlling the Gaussian mutation operator to optimize the structural parameters of the GRU neural network and then explores an effective model for stock market time series prediction. The main contributions are as follows:

(1) To solve the problem of manual experience design and slow convergence speed of the GRU neural network structure. This paper combines the EA with a GRU neural network and proposes an EA algorithm to search for the optimal parameters of the GRU neural network structure. Based on the EA algorithm, the parameters of the time step, hidden units, and batch size of the GRU neural network structure are optimized. Focusing on optimizing the network structure parameters related to the topological structure and calculation cost of the neural network speeds up the searchability of optimal network structure parameters. In addition, it effectively improves the convergence speed and prediction accuracy of the GRU model.

(2) To solve the problems of premature convergence and poor local searchability of the EA. The EA, based on prior parameters controlling the Gaussian mutation operator is designed to optimize the structural parameters of the GRU neural network. In this strategy, the variance of GRU neural network structure parameter sensitivity learning is used as Gaussian mutation. It can realize the individual evolution in the best direction of the population, effectively overcome the optimal local problem, improve the diversity of the final solution and accelerate the convergence speed.

(3) In this paper, the optimal structural parameters of a neural network with good convergence are applied to predict stock market returns. The prediction results of the RF, SVR, RNN, LSTM, GRU, EA-GRU, and PGM-EA-GRU models for stock market returns are compared and analyzed. The effectiveness of the proposed PGM-EA-GRU model in the prediction of stock market time series is verified as well.

Following this introduction, the paper progresses as follows. Section 2 describes literature review. Section 3 details components of proposed PGM-EA-GRU. Section 4 describes experimental setup. Empirical results and analysis are drawn in Section 5, discussion and conclusion are drawn in Section 6.

2. LITERATURE REVIEW

There are two kinds of forecasting methods for stock market returns: statistical methods and intelligent forecasting methods. Statistical methods include ARMA, GARCH, and regression models. For example, Rounaghi & Zadeh [12] use ARMA to predict stock returns, which improves the prediction accuracy. Arellano & Rodriguez [13] forecast the exchange rate and stock market based on Markov-switching GARCH models and achieved good forecasting results. Naeem et al. [14] study the effect of investor sentiment on cryptocurrency prediction based on an OLS regression model. Such methods need to make assumptions about the parameters in the model in advance, which is easily constrained by the feature

dimension of financial market data. With the improvement of the financial market data dimension, the prediction accuracy gradually decreases, which has limitations in the application of nonlinear complex systems [15].

Intelligent methods for stock market return prediction can be divided into machine learning and deep learning methods. The more common machine learning algorithms include support vector machines and artificial neural networks. These methods need to provide a set of labeled training data in advance and then substitute the test data set to obtain the prediction result. For example, Zhao et al. [16] found that the LS-SVM algorithm improves the prediction accuracy and has good robustness in nonlinear prediction problems. Ince et al. [17] used an artificial neural network to forecast the exchange rate market, which improved the accuracy of exchange rate forecasting. Nah et al. [18] predicted the prices of different cryptocurrencies based on the SVM-PSO combined algorithm and, compared to the SVM method, the prediction accuracy was significantly improved. In comparison to traditional statistical methods, these methods can substantially improve the prediction accuracy of stock market returns [19]. However, traditional machine learning methods rely on data label knowledge, have decisive human intervention, are only applicable to the case of small training data sets, and cannot obtain good prediction results in high-dimensional financial data sets [20].

As the basis of deep learning, DNNs are widely used in financial market prediction. This method is more suitable for training sets with data labels, and the more labeled data sets there are, the higher the prediction effectiveness [21]. However, DNN has more redundant information, and its feature representation ability is not strong, which also makes the convergence speed of DNN slower [7, 22]. For this reason, scholars have proposed an improved multiobjective evolutionary algorithm to optimize DNN model parameters, and it has been shown to be an effective method in solving network structure complexity and prediction accuracy problems [23]. Later, Lachiheb & Gouider [24] also proposed a layer-by-layer DNN to predict stock prices using the error rate and regularization parameters as the objective function. In contrast to existing studies, it has higher prediction accuracy and stronger learning and training capabilities. However, when the error is the smallest, the complexity of the network structure also has a greater impact on the training time and training difficulty. Therefore, Jia et al. [25] proposed a multiobjective evolutionary algorithm MOEA/D to optimize the connection of the DNN network structure and verified that the method can improve the convergence speed and enhance the learnability of the network structure. Huang et al. [26] also proposed a multiobjective evolutionary algorithm MOE-CL to optimize DNN and used the DNN classification error rate and network structure sparsity as the objective function. The results also proved that the MOE-CL-optimized DNN method can considerably improve the prediction accuracy and convergence speed, but these methods are not suitable for dealing with changes in time series data and cannot be modeled.

As a classic deep learning algorithm, DBN has the ability to capture high-level network structure data features and strong learning capabilities [27]. Compared to traditional BP, GA-BP, and ARMA, DBN has significantly improved prediction

accuracy and effectively shortened training time; the effectiveness of the algorithm is verified in the gold price prediction results [28]. Considering the continuity of time series data in the financial market, Shen et al. [29] proposed an improved DBN to forecast the exchange rate, which aims to improve the processing capacity of continuous data. Compared to the ARMA and the FFNN, it has better performance in prediction error and accuracy. Lean et al. [30] proposed the DBN-SVM combination algorithm and applied it to the study of credit risk classification. The results show that the accuracy of credit risk classification has been significantly improved, which also verifies the effectiveness of the algorithm in dealing with the credit risk classification problem with unbalanced data. The method first trains the labeled training data set and then tests it on the test data set. Although the above methods are superior to traditional neural networks and mathematical statistics methods in terms of financial market prediction and classification accuracy, they are not suitable for situations in which the labels of the training data set are not available. Even if the training data set has label information, the complexity of its network structure and its feature extraction learning ability will also affect the accuracy of the prediction results [31].

Compared to the above methods, the long short-term memory network (LSTM), as a special recurrent neural network (RNN), has long-term memory characteristics, can solve long-term dependent learning problems and is more suitable for stock market time series modeling. Kim & Won [32] used the LSTM-GARCH combined model and took the KOSPI 200 index as an example to verify that the combined method is better than LSTM. Altan et al. [33], based on the LSTM-EWT-CS combination algorithm to forecast the price fluctuation trend of digital currency, proved the effectiveness and superiority of the combination algorithm in stock price forecasting. However, the LSTM neural networks structural parameters depend on the practical design and are often constrained by parameter setting. Therefore, the accuracy of the prediction results will be affected [8]. Compared with LSTM, the gated recurrent unit has the same prediction performance as LSTM. However, its network structure is simple, the network structure parameters are few, and the calculation cost is lower. Hence, the GRU has been widely used. For example, Awoke [34] compared the forecasting results of LSTM and GRU models and found that the GRU model has a better forecasting effect than the LSTM model. The GRU model is more suitable for forecasting time series data with major price fluctuations. In the first mock exam, Niu et al. [35] used the ICEEMDAN-R-AttGRU combination model to predict energy prices. After comparing it to a single model, the prediction accuracy was greatly improved. The GRU model and its combination model have high accuracy in predicting stock market returns. However, its network structure parameters depend on artificial experience design, leading to high calculation cost, slow convergence speed, easily falling into local optimum, and affects the accuracy of prediction results.

Aiming at the problems of practical design and slow convergence speed of neural network structure parameters, evolutionary algorithms (EAs) effectively overcome the optimization problem of neural network structure parameters. The prediction performance and robust performance have been greatly improved, and thus,

EAs have been widely considered by the academic community. EA evolves neural network structure. First, the network structure is initialized, crossed, mutated, and selected. The accuracy of verification evolves as applicability and can search neural network structure parameters more efficiently. For example, Loshchilov & Hutter [36] optimized the structural parameters of a convolutional neural network (CMA-ES-CNN) based on a covariance matrix adaptation evolution strategy. They verified the effectiveness of its search for the optimal structural parameters of a convolutional neural network. Chung & Shin [37] used a genetic algorithm to optimize the structure time window and the number of hidden layer nodes of the LSTM neural network (GA-LSTM) and used the combination model to predict the Korean stock price index. The combination model effectively improved the models predictability. Valdez & Rojas-Domnguez [38] proposed a linear evolutionary algorithm based on the direction of principal variance (LEA-MVD) to optimize the structural parameters of a DBN neural network. Compared to the traditional DBN algorithm, the effectiveness of the LEA-MVD algorithm for optimizing the structural parameters of the DBN was verified. Li et al. [10] combined the EA algorithm with an LSTM neural network structure, which uses an attention mechanism by transferring shared weight parameters. It applies the combined model to predict time series data. Compared to other benchmark models, the prediction performance of the combined model has been dramatically improved. Deng et al. [11] proposed an improved quantum-inspired differential evolution algorithm to optimize the structural parameters of the DBN neural network (MSIQDE-DBN). It was applied to practical engineering fault classification research. The results show that the model has better fault classification accuracy, which also verifies the algorithms effectiveness in dealing with fault classification problems.

According to what has been mentioned above, there are several problems in the current algorithm. For example, empirical selection of GRU parameters, slow convergence speed and poor global search ability. To solve the above issues, this paper uses the improved EA algorithm to optimize the structural parameters of the GRU neural network and proposes an EA algorithm based on a prior parameter control Gaussian mutation operator, while optimizing the structural parameters of the GRU neural network (PGM-EA-GRU). An effective model for stock market time series prediction is built to improve the prediction performance of stock market returns.

3. METHODOLOGY

To improve the searchability, convergence speed and prediction performance of neural network structural parameters, this paper proposes an EA-GRU combination model based on a prior parameter-controlled Gaussian mutation operator. The research method flow of this paper is shown in Figure 1. This paper will briefly introduce the components of the model in the next section.

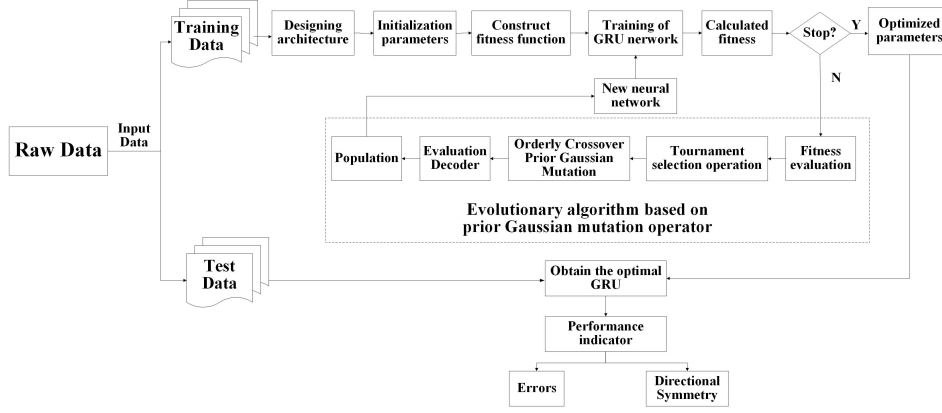


FIGURE 1. The research framework of the proposed method

3.1. GATE RECURRENT UNIT

The gated recurrent unit (GRU) is a variant of LSTM that can effectively solve the gradient problem in long-term dependent learning and backpropagation of time series data. The GRU neural network structure is composed of a reset gate and update gate, which replace the output gate, input gate, and forgetting gate in LSTM. The update gate determines which information to discard and which new information to add, and the reset gate determines the degree of discarding the previous information. The specific steps of GRU are as follows:

The formula of the t time update gate in the GRU neural network structure can be expressed as follows:

$$z_t = \sigma(W^z x_t + U^z h_{t-1} + b_z) \quad (1)$$

The calculation formula of the reset gate of time t is as follows:

$$r_t = \sigma(W^r x_t + U^r h_{t-1} + b_r) \quad (2)$$

The calculation formula of cellular memory of time t is as follows:

$$\tilde{h}_t = \tanh(t^t \times U h_t + W x_t + b) \quad (3)$$

The output of t time can be expressed as:

$$h_t = (1 - z_t) \times \tilde{h}_t + z_t \times h_{t-1} \quad (4)$$

Where z_t , r_t represent the update gate and reset gate, h_t represent the candidate hidden layer. W^z , U^z , W^r denote the weight matrices for the corresponding connected input vector. U^r , W , U represent the weight matrices of the previous time step, and b_z , b_r , b are bias, σ is a sigmoid function.

3.2. IMPROVED EVOLUTIONARY ALGORITHM

Based on the Gaussian mutation operator, this paper proposes an EA that uses prior parameters to control the Gaussian mutation operator to optimize the structural parameters of the GRU neural network. Similar to the traditional EA, the improved EA mainly includes crossover, mutation and selection.

Step 1: **Crossover.** The sequence crossover method was used to copy the gene fragment cut out from parent 1 into the self-carrying fragment and check the position with the same value in parent 2. Then, the position with the same value in parent 2 was excluded, and the remaining values in parent 2 were filled into the offspring to obtain the gene fragment of the offspring. The crossover process is shown in Figure 2.

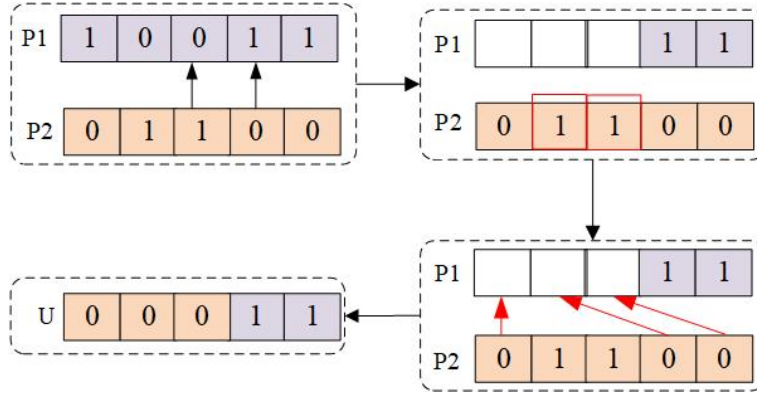


FIGURE 2. Crossover process

Step 2: **Mutation.** Gaussian mutation is an optimization algorithm that uses a random number selected according to the Gaussian distribution of variance to act on the original location and then generate a new location. Because most mutation operators are distributed around the original location and only a few mutation operators are far away from the current location, Gaussian mutation operators can not only effectively narrow the search scope and avoid falling into local optimum but also enhance the diversity of the population and speed up the convergence of the algorithm. The probability density function of the Gaussian distribution can be expressed as:

$$F(X) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(X-\mu)^2}{2\sigma^2}} \quad -\infty < X < \infty \quad (5)$$

In Formula (5), σ is the variance of the Gaussian distribution, and μ refers to the desired value.

In the standard Gaussian probability density, $\mu = 0$, $\sigma = 1$. To consider the characteristics of the current population, two positions in the population are

arbitrarily selected, and the Gaussian distribution interacts with the difference between any two individuals. Then, the Gaussian variation expression can be expressed as:

$$M = G(\zeta) \cdot (X_1 - X - 2) \quad (6)$$

In Formula (6), $G(\zeta)$ refers to the Gaussian distribution of the probability density formation in Formula (5), and $\zeta \in [0, 1]$, X_1 and X_2 represent the positional information for two randomly selected individuals in the population.

Because prior Gaussian mutation takes the variance generated by the sensitivity learning of neural network structure parameters as the variance of Gaussian mutation, its preset prior Gaussian mutation operator can guide individuals to mutation in the direction of global optimization, which not only improves the probability of the algorithm finding the optimal solution but also enables the optimal individuals to be evenly distributed around the prior Gaussian variance so that the solution set maintains good diversity. Then, the prediction performance of the model can be effectively improved. On this basis, the prior Gaussian distribution probability density can be expressed as

$$F'(X) = \frac{1}{\sqrt{2\pi\sigma'}} e^{-\frac{(X-\mu')^2}{2\sigma'^2}}, \quad -\infty < X < \infty \quad (7)$$

In Formula (7), σ' is the variance of the prior Gaussian distribution, and μ' refers to the desired value.

In the prior Gaussian probability density $\mu = 0$, σ' represents the variance of the time step, batch size, and hidden layer cell size of parameter sensitivity learning. Similarly, the prior Gaussian variant expression can be expressed as:

$$M' = G'(\zeta') \cdot (X_1 - X - 2) \quad (8)$$

In Formula (8), G' refers to the Gaussian distribution of the probability density formation in Formula (7), and the other symbolic meanings are the same as above.

Step 3: Selection. The tournament selection method is used to calculate the population according to the fitness function $P_t = 2N$, including each individual fitness value. Next, $tour\ size = N$ is randomly selected from $P_t = 2N$. Then, individuals with the best fitness values are selected to enter the breeding population from $P_t = 2N$. This process is repeated $2N$ times for the next round of breeding population Q_{t+1} .

The tournament selection process is shown in Figure 3:

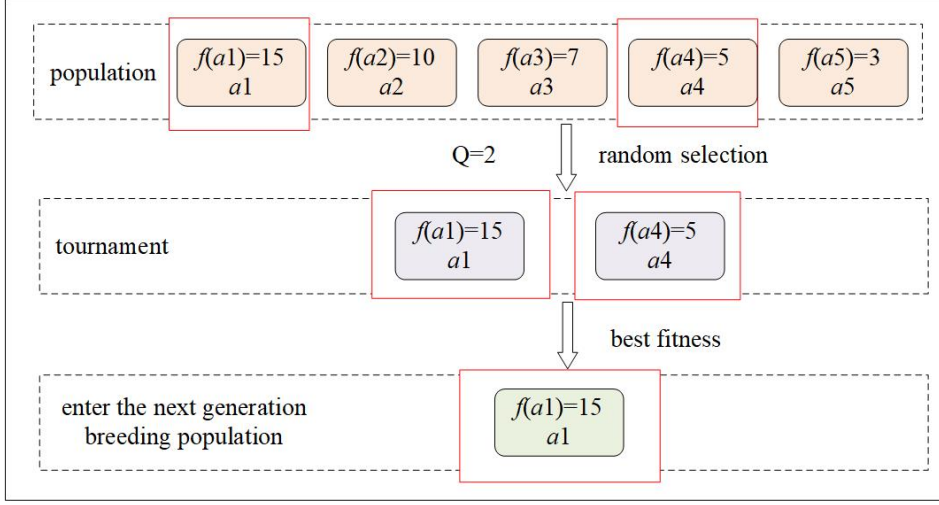


FIGURE 3. Tournament selection process

3.3. PGM-EA-GRU MODEL

Compared to the LSTM neural network, the GRU is more straightforward in structure, better in prediction performance, and lower in calculation cost, making it widely used in time series prediction. However, the parameter setting of the GRU neural network structure often depends on the practical design, which has some problems. For instance, slow convergence speed, ease of falling into a local optimum, and low efficiency of parameter search affect the prediction accuracy of the model. Evolutionary algorithms (EAs) are mathematical algorithms that limit the natural evolution process. EA algorithm have been widely used in solving neural network structure searches. Therefore, in order to solve the problem of searching optimal structural parameters of GRU neural network, using the improved EA algorithm to optimize the structural parameters of GRU neural network. Propose an EA algorithm based on prior Gaussian mutation operator to optimize the structural parameters of GRU neural network. Due to prior Gaussian mutation is the neural network structure parameter sensitivity study variance as Gaussian mutation of variance. So, produced by the preset prior Gaussian mutation operator to guide individual toward the global optimal direction variation. It can increase the probability of algorithm find the optimal solution, and make the best individual is uniformly distributed around the transcendental Gaussian variance. Actually, it can maintain good diversity and improve the prediction performance of the model effectively.

This study is divided into the following two parts: First, the initial GRU neural network is constructed. The original data set of stock market return prediction is divided into a training set and a test set. The training set is recorded as $X(t) = (x_t^1, x_t^2, \dots, x_t^n)$. The structural parameters of the GRU neural network

and EA parameters are initialized. Meanwhile, X_t is inputted into the GRU neural network for training, and the prediction error is obtained on the test set. Then, the EA of the Gaussian mutation operator and prior Gaussian mutation operator to evolve the GRU neural network structural parameters time step, hidden units, batch size, and record the two evolutionary algorithms as EA and PGM-EA. The fitness value of each individual is evaluated according to the fitness function. If the condition is satisfied, the loop is terminated, and the optimal network structure parameters are output. Otherwise, new populations will be generated by reselection, crossover, mutation, and other evolutionary methods. Then, the optimal structural parameters of the neural network will be obtained. In this paper, chromosomes are encoded in binary form, and the mean square error (MSE) of the prediction model is taken as the fitness function.

The framework of the PGM-EA-GRU algorithm proposed in this paper is shown in Algorithm 1. In PGM-EA-GRU, an initial solution set P_0 is randomly generated, and each individual of P_0 is trained on the training set and evaluated on the verification set D_{train} . First, the fitness function f is built, and a subset S is randomly selected from P_0 at each evolutionary step. According to the fitness value, the best individual B and the worst individual W are selected from S . When W is eliminated, B will remain as the parent of the next generation. Then, a mutant offspring C is generated, and C is trained and evaluated with respect to its fitness f_C . If the maximum number of iterations is reached, the training process is terminated, and the optimal parameters are output. Otherwise, iterative optimization is continued, the optimal solution is output, and the optimal solution is retrained and tested to obtain the final prediction effect.

Algorithm 1 The Framework of PGM-EA-GRU

Input: num epochs $\#T$, population size $\#P$, chromosome length $\#B$, sample size $\#S$, training set $\#D_{train}$, validation set $\#D_{val}$

Output: A set of optimal neural network structure parameters: batch size, hidden units, time step

- 1: $P_0 \leftarrow$ Population initialization ($\#P, \#B$)
 - 2: $S_G \leftarrow$ sample ($P_{G-1}, \#S$)
 - 3: **while** Does not meet the stop condition of the algorithm **do**
 - 4: $B, W \leftarrow$ tournament selection (S^G)
 - 5: $C, \omega^B \leftarrow$ crossover and prior Gaussian mutation (B)
 - 6: $\omega^C \leftarrow$ train (C, ω^B, D_{train})
 - 7: $f_C \leftarrow$ evaluate (C, ω^C, D_{val})
 - 8: Obtain the optimized parameters batch size, hidden units, time step
 - 9: Retrain the optimized parameters batch size, hidden units, time step
-

4. EXPERIMENTAL SETUP

4.1. PARAMETER SETTINGS

In this section, the GRU neural network is first used for training. Then, the GRU neural network is optimized based on an a priori Gaussian mutation evolutionary algorithm, the fitness function is used to evaluate the population, and the optimal neural network structure parameters are solved through selection, crossover, mutation and other evolutionary methods. Finally, the SVR, RF, RNN, GRU, LSTM, EA-GRU and PGM-EA-GRU models are selected for comparative research to verify the prediction performance of PGM-EA-GRU for stock market time series data. The GRU model parameters are obtained using a grid search. The parameters of the GRU model are set as batch size=[64,128,256,512,1024], hidden units=[16,32,64,128,256], time step=[3,6,12,18,24], and the number of iterations for each parameter is 100 [10]. The learning process of the grid search is shown in Figure 5. The batch size is 64, the number of hidden units are 256, and the time step is 6. At the same time, other models also perform a grid search to find the optimal model settings. In addition, PGM-EA-GRU model parameters are obtained through an evolutionary algorithm search to obtain the optimal neural network structure parameters, as shown in Table 1. The time step is 7, the number of hidden units is 63, and the batch size is 10. The EA-GRU model parameters are also obtained through an evolutionary algorithm search.

4.2. DATA SELECTION AND PROCESSING

This study selects the Shanghai Composite Index as the research object, taking 3rd January 2008 to 29th October 2020 as the sample period, 90% of the observations as the training data set, and 10% of the samples as the test set. Considering that the market trading data can genuinely reflect the trading situation of the day, the technical analysis index can reflect the changing trend of the Shanghai Composite Index. Therefore, referring to Gong et al. [20], this paper selects the PSY psychological line, RSI relative strength index, OBV energy tide, DMA average difference, DMI trend index, boll brin line, CCI homeopathy index, CR energy index, MACD moving average, KDJ random index, TRIX triple index smooth average index, trading volume, opening price, closing price, highest price, and lowest prices as the input variables of the model. The logarithmic return of the daily closing price is used as the output variable of the model (the daily return of the stock market $R_t = [\ln(P_t) - \ln(P_{t-1})] \times 100$, where P_t is the closing price of the Shanghai stock index on day t). All the data are from the Wind database. In addition, the data of each index selected in the model have different orders of magnitude and dimensions. Although the shallow neural network model can deal with nonlinear data, its learning ability for time series data is lacking, which will reduce the prediction accuracy. Therefore, to eliminate these effects, it is necessary to standardize the indexes and adopt the deviation standardization method to standardize the indexes.

4.3. PERFORMANCE INDICATOR

To compare and analyze the prediction effects of different models, this section selects the five evaluation indicators: Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Directional Symmetry (DS). These indicators evaluate the prediction effect of the model in the stock market through MSE, RMSE, MAE, MAPE, and DS, which can be expressed as Equations:

$$MSE = \frac{1}{T} \sum_{i=1}^T (y_i - y'_i)^2 \quad (9)$$

$$RMSE = \frac{1}{T} \sqrt{\left(\frac{\sum_{i=1}^T (y_i - y'_i)^2}{T} \right)} \quad (10)$$

$$MAE = \frac{1}{T} \sum_{i=1}^T |y_i - y'_i| \quad (11)$$

$$MAPE = \frac{1}{T} \sum_{i=1}^T \left| \frac{y_i - y'_i}{y_i} \right| \times 100 \quad (12)$$

$$DS = \frac{1}{T} \sum_{i=1}^T \mu_i, \quad \text{where } \mu_i = \begin{cases} 1, & \text{if } (y_{i+1} - y_i)(y'_{i+1}) > 0 \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

where $Y = (y_1, y_2, \dots, y_T)^T$ is the true value vector, $Y' = (y'_1, y'_2, \dots, y'_T)^T$ is the predicted value vector, and T is the number of samples.

5. EMPIRICAL RESULTS AND ANALYSIS

5.1. ANALYSIS OF PARAMETER SENSITIVITY

To verify the effectiveness of the proposed model in the prediction of stock market time series data, this paper presents a comparison and analysis of the prediction error effects of LSTM, RNN and GRU models first. The LSTM, RNN, GRU models are selected with experience. As shown in Figure 4, the prediction errors of the above three models display a rapid downward trend. Compared to the LSTM and RNN models, the prediction error of the GRU model is smaller. Moreover, the GRU model has a faster convergence rate, which indicates that the GRU model has an obvious advantage in the prediction of the stock market return rate. Therefore, this paper uses the GRU model to analyze the sensitivity of the structural parameters of the neural network.

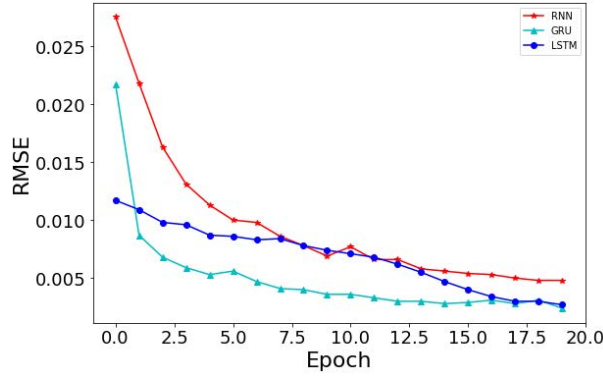


FIGURE 4. Relationship between Epoch and RMSE

To better analyze the impact of the changes of parameters in the GRU model on the prediction results of stock market return, this part sets the parameters in the GRU model as batch size=[64,128,256,512,1024], hidden units=[16,32,64,128,256], time step=[3,6,12,18,24], and the number of iterations of each parameter is 100. Through repeated iterative training, the RMSE corresponding to different parameters is obtained. The blue vertical line in Figure 5 is the standard deviation. The longer the blue vertical line is, the larger the standard deviation is. The circle is the mean value. When one of the parameters changes, the other parameters remain unchanged, which can effectively reflect the batch size, hidden units, and time steps on RMSE.

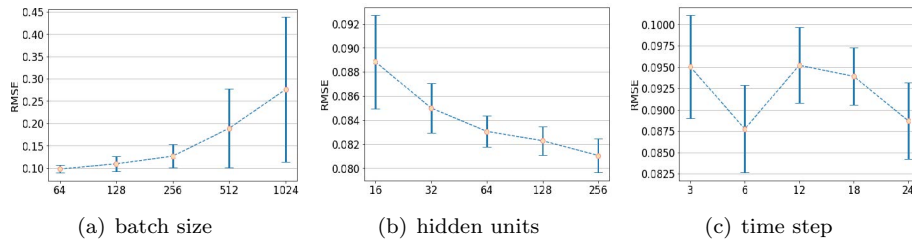


FIGURE 5. Parameter sensitivity results

5.2. ANALYSIS OF THE TRAINING PROCESS

The iterative process of the EA-GRU and PGM-EA-GRU model populations is shown in Figure 6. In each iteration, two individuals are randomly selected from 20 individuals, and the individuals with the best fitness are selected to enter the breeding population. This process is repeated 20 times, and then all the optimal individuals in this round are obtained, trained as the parent population in the next iteration, etc. Finally, this paper obtains the best individual in each round

of training. Among them, the vertical line in Figure 4 represents the standard deviation of other individuals except for the best individual in the iteration round, the circle is the mean value, and the blue five-pointed star represents the individual with the best fitness in the iteration round. RMSE is the smallest. It can be seen from Figure 4 that there are substantial differences in the fitness of all individuals in each iteration. With continuous population iteration, the fitness curve of the best individual gradually tends to be stable, indicating that the fitness difference of each individual in the population in the iteration gradually decreases. It also means that the number of the best individuals in the population in the iteration is increasing.

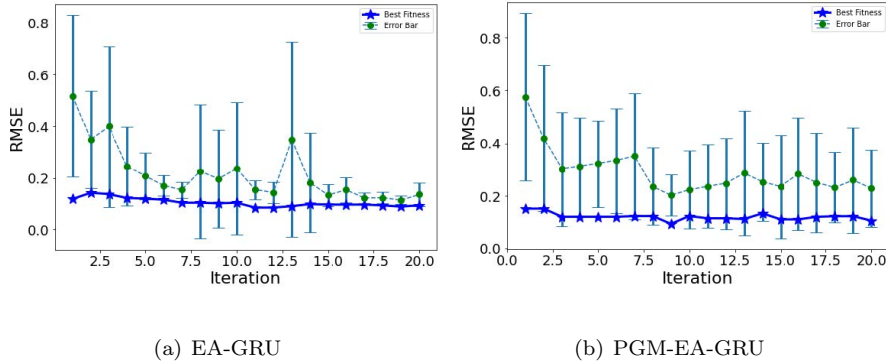


FIGURE 6. Training process of different combination models

The results of Figure 6(a) show that when the EA-GRU iteration number is 11, the root means the smallest square error of the best individual fitness value. However, the standard deviation of other individuals is not the smallest, indicating that the individual stability of the population is relatively poor in this iteration round. Figure 6(b) shows that PGM-EA-GRU can obtain the best individual fitness. The standard deviation of other individuals is the smallest when the number of iterations is 9, which indicates that the best population can be obtained in this iteration round. The individual of this population is relatively stable, which also verifies our proposed evolutionary algorithm, based on the prior parameter control Gaussian mutation operator. It can promote individuals to evolve toward the optimal global direction, effectively overcome the optimal local problem, and accelerate the convergence speed of the algorithm.

Through the above genetic algorithms and evolutionary algorithms for GRU neural network structure training and learning, when EA-GRU and PGM-EA-GRU iterations are 12 and 9, respectively, the optimal parameters of the GRU neural network structure can be obtained, as shown in Table 1. Currently, the prediction model has higher prediction accuracy and network training time. The optimal structure parameters of EA-GRU are as follows: time steps are 3, hidden

units are 55, and batch size is 22. The optimal structure parameters of PGM-EA-GRU are as follows: time steps are 7, hidden units are 63, and batch size is 10.

TABLE 1. Optimal hyperparameter settings

Algorithm	Optimal parameters	Value
EA-GRU	time steps	3
	hidden units	55
	batch size	22
PGM-EA-GRU	time steps	7
	hidden units	63
	batch size	10

5.3. FORECAST RESULTS AND ANALYSIS OF STOCK MARKET RETURNS

When the EA is used to determine the optimal parameters of the GRU neural network structure, the stock market return can be predicted based on the optimal parameters of the GRU neural network structure obtained by the EA algorithm. Figure 7 is the resulting chart of stock market return prediction based on the Gaussian mutation operator and prior parameter control Gaussian mutation operator evolution GRU neural network structure. Figure 7 shows that the EA-GRU and PGM-EA-GRU models have a slight deviation from the actual stock market return, and they have a high degree of coincidence.

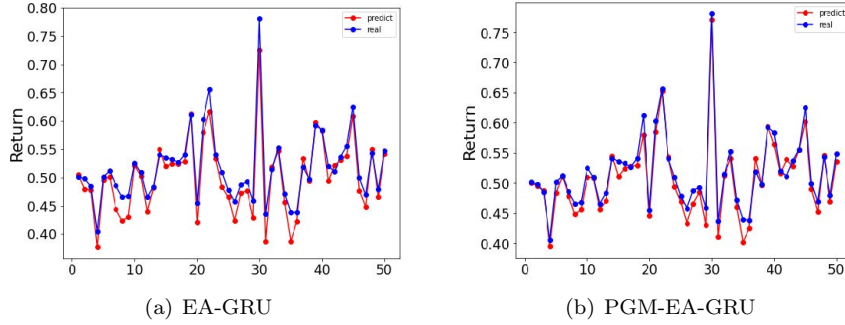


FIGURE 7. Stock market return forecast results

To evaluate the prediction performance of the model, not only should the accuracy of the prediction results be considered but also the relationship between the number of epochs and the training error of the model should be compared. Figure 8 gives the change chart of the number of epochs and the training error results of the two models. It can be seen from Figure 8 that the error value of

the EA-GRU model fluctuates violently in the process of training and testing. For the PGM-EA-GRU model, with the increase in the number of epochs, the training and testing errors continue to shrink, the error value fluctuation is slight. Therefore, the PGM-EA-GRU model has higher prediction accuracy and less network training time than the EA-GRU model.

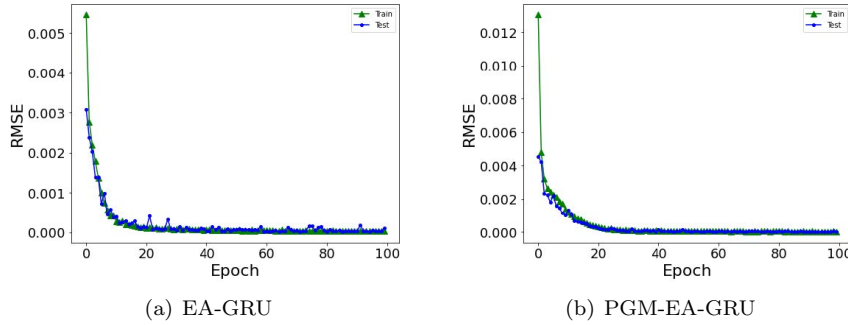


FIGURE 8. Prediction error results

In order to test the forecasting effect of PGM-EA-GRU, this study compares the forecasting effect of the RF, SVR, RNN, LSTM, and GRU models on stock market returns. Figure 9(a) shows that there is a significant deviation between the predicted value and the absolute value of the stock market return rate of the RF model. The possible reason is that the random forest model is a kind of external neural network with slow convergence speed, easily falls into a local optimum, and has a poor learning ability for high-dimensional, massive and nonlinear time series data, which leads to the poor fitting effect of the model. Figure 9(b) gives the prediction effect of the SVR model because the prediction effect of the SVR model is easily affected by the range of parameters. The error between the predicted value and the actual value of the SVR model is large, and the fitting accuracy is not high. Moreover, the prediction effect of the SVR model is consistent with that of the RF model. This result also shows that the traditional shallow machine learning algorithm has some limitations in predicting stock market returns. It is difficult to effectively describe the complex nonlinear characteristics of stock market time series data due to its poor feature learning ability for high-dimensional complex nonlinear time series data. Therefore, it has not achieved good prediction performance in stock market return prediction.

It can be seen from Figure 9(c) and Figure 9(e) that the RNN, LSTM and GRU models have a better fitting effect on the predicted value and the real value of stock market returns, especially when the stock market return volatility is small, and the predicted value and the actual value have a higher degree of coincidence. When the stock market return volatility is more severe, the error between the predicted value and the actual value of the model will also increase. This result reflects the law of stock price fluctuation. Unlike the traditional machine learning algorithms

of RF and SVR, the RNN model, LSTM and GRU model have multilayer hidden layers, so they have a higher nonlinear fitting ability and prediction accuracy for massive high-dimensional multisource heterogeneous data.

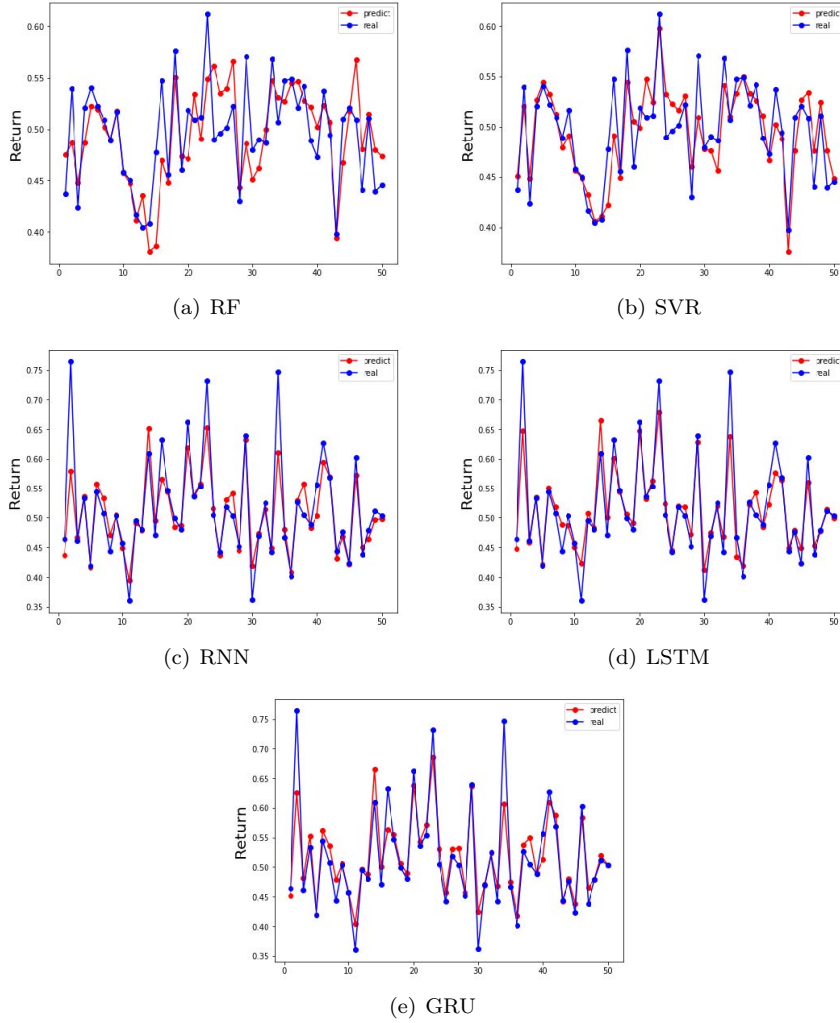


FIGURE 9. The prediction results of different variables on stock market returns

To better explain the prediction accuracy of the PGM-EA-GRU model, this paper introduces MSE, RMSE, MAE, MAPE, DS, and ST.D to evaluate the prediction effect of each model. Table 2 shows that compared with the RF, SVR, RNN, LSTM, GRU, EA-GRU models, the error values of the MSE, RMSE, MAE,

and MAPE of the PGM-EA-GRU combination model are the smallest, large standard deviation and the PGM-EA-GRU model has a higher prediction hit rate. With DS as high as 0.9421, the combination model not only has a higher prediction accuracy in stock market return prediction but also has a higher prediction accuracy. Moreover, it also shows good prediction performance in predicting the rising and falling trends of stock market returns.

TABLE 2. Prediction error of different models

Model	MSE	RMSE	MAE	MAPE	DS	ST.D
RF	0.0022	0.0473	0.023	6.3529	0.8682	0.0514
SVR	0.0026	0.0513	0.0351	7.4998	0.7621	0.0526
RNN	0.0018	0.0429	0.0257	5.5562	0.8932	0.0535
LSTM	0.002	0.0448	0.0252	5.4588	0.919	0.0652
GRU	0.0017	0.0411	0.0242	5.2763	0.9029	0.0695
EA-GRU	0.0009	0.0307	0.0189	4.4437	0.9163	0.0818
PGM-EA-GRU	0.0005	0.0216	0.0154	3.5472	0.9421	0.0857

In general, the first mock exam performance of the EA-GRU and PGM-EA-GRU combination models is better than that of a single model. Combining the EA algorithm and GRU neural network structure can greatly improve the prediction effect of the GRU model on stock market returns. This also means that compared with the single machine learning and deep learning model, the combined model has stronger feature extraction learnability, stronger hyperparameter searchability, and faster hyperparameter search speed for the optimal network structure for high-dimensional complex nonlinear stock market time-series data. This further verifies the effectiveness of the PGM-EA-GRU model in dealing with high-dimensional and complex nonlinear stock market data and the effectiveness of large-scale, nonlinear and complex dynamic systems. The PGM-EA-GRU model based on the prior parameter control Gaussian mutation operator is an effective method to predict stock market returns. This can effectively overcome the defects of traditional machine learning and deep learning, and improve the convergence speed and prediction accuracy of the model.

5.4. DIEBOLD MARIANO TEST

It can be seen from the above stock market yield prediction results that compared with other models, the PGM-EA-GRU model proposed in this study has the best prediction performance under five different evaluation indicators: MSE, RMSE, MAE, SMAPE and DS. To further compare whether different models have differences in the prediction of stock market returns, DM statistics are used to test the prediction ability of different prediction models based on the loss function (MSE). The DM test results are shown in Table 3.

It can be seen from Table 3 that when the PGM-EA-GRU prediction model proposed in this study is used as the tested model, the original hypothesis is

significantly rejected for the DM test results using other models. In other words, the models have the same prediction ability, both pass the 1% or 5% significance level test, and the coefficients of the DM test results of other models are less than 0. This shows that the PGM-EA-GRU model proposed in this study has significantly better prediction performance than other models at the 1% or 5% significance level, which also means that the PGM-EA-GRU prediction model has better prediction ability for stock market returns and is an effective stock market return prediction method. Therefore, the prediction performance comparison of the models in this study is statistically significant.

TABLE 3. DM test of model prediction performance

Model	PGM-EA-GRU	EA-GRU	GRU	LSTM	RNN	RF
EA-GRU	-2.804 (0.005)					
GRU	-4.873 (0.000)	-5.942 (0.000)				
LSTM	-3.696 (0.000)	-4.862 (0.000)	-1.941 0.053			
RNN	-1.738 (0.083)	-2.97 (0.003)	-2.268 (0.024)	-3.377 (0.001)		
RF	-2.472 (0.014)	-3.516 (0.000)	-1.201 (0.231)	-1.934 (0.054)	-1.825 (0.069)	
SVR	-2.961 (0.003)	-4.025 (0.000)	-0.433 (0.666)	-1.752 (0.080)	-2.572 (0.010)	-1.884 (0.060)

6. DISCUSSION AND CONCLUSION

6.1. DISCUSSION

To solve the problems of robustness and poor global searching ability of existing evolutionary algorithms in the process of searching optimal neural network structural parameters, a stock market yield prediction method based on a prior Gaussian mutation operator optimized LSTM neural network is proposed in this paper. The algorithm uses the variance obtained from GRU neural network structure parameter training as the Gaussian variance to generate the optimal candidate set of individuals to obtain the optimal parameters of the PGM-EA-GRU neural network structure and apply them to research on stock market yield prediction.

From the experimental research results in this paper, compared with other models, the PGM-EA-GRU prediction model has better prediction ability for stock market returns and is an effective stock market return prediction method. The DM statistical test results further verify that the prediction effect of this model is significantly better than that of other models, indicating that this model has robustness and applicability. The Gaussian mutation mechanism is introduced in

this paper. Compared with other models, this algorithm can effectively narrow the search scope, avoid falling into a local optimum, enhance the diversity of the population, accelerate the convergence speed of the algorithm, and improve the accuracy of the final stock market yield prediction results. Although the model in this paper shows good prediction performance in stock market forecasting, as a typical nonlinear complex system, the stock market is vulnerable to macroeconomic, external environment, investor sentiment and many other factors, which furthers the risk of stock price volatility and shows a significant stock market risk spillover effect, which in turn will affect these factors and have a new impact on the stock market. This also brings more challenges to the prediction of stock market yield. Therefore, there are still some shortcomings to be further explored. They are mainly reflected in the following aspects:

(1) In this study, only the historical data of the stock market is considered; that is, only technical analysis indicators are used to predict the stock market yield. In fact, the factors that affect the stock market, such as macroeconomic, external environment, investor sentiment and other factors, are complex and changeable. Therefore, this may also reduce the accuracy of stock market yield prediction.

(2) This research focuses on the Gaussian mutation operator and uses the evolutionary algorithm based on a priori Gaussian mutation to optimize the structural parameters of the GRU neural network. Compared with the benchmark model, the PGM-EA-GRU model has better prediction performance, but it often faces more complex problems in practice, while PGM-EA-GRU has difficulty modelling different tasks separately; that is, it cannot be used for parallel training.

6.2. CONCLUSION

In this paper, an evolutionary algorithm based on an a priori Gaussian mutation operator to search the optimal parameters of the neural network structure is proposed. The algorithm is used to select the best individual according to the ranking of population individuals in each iteration for the next iteration. This process is repeated to obtain the best fitness, thus the optimal neural network structure parameters are used to predict the stock market return. Compared with RF, SVR, LSTM, GRU, and EA-GRU, the PGM-EA-GRU algorithm has a significantly better prediction effect than the other algorithms, can effectively overcome the problems of slow convergence speed and easily falling into a local optimum, accelerate the search ability of optimal neural network structure parameters, and improve the search efficiency and prediction accuracy of optimal neural network structure parameters. In addition, the statistical test results show that the PGM-EA-GRU algorithm proposed in this study has greater advantages and stability than other algorithms.

Based on the above research, we can further expand our research in the following areas in the future:

(1) In follow-up research, it is necessary to deeply explore other factors that affect the stock market, form a multisource mixed and multidimensional stock

market yield prediction index system, and build a more effective in-depth learning model based on the characteristics of multisource mixed data to improve the accuracy and effectiveness of the stock market prediction model.

(2) In future research, we could attempt to introduce the idea of parallel computing into the model to improve its convergence speed.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China [No. 61773029, No.62073007].

REFERENCES

- [1] Ivo Welch and Amit Goyal. A comprehensive look at the empirical performance of equity premium prediction. *The Review of Financial Studies*, 21(4):1455–1508, 2008. 1
- [2] Suryoday Basak, Saibal Kar, Snehanshu Saha, Luckyson Khaidem, and Sudeepa Roy Dey. Predicting the direction of stock market prices using tree-based classifiers. *The North American Journal of Economics and Finance*, 47:552–567, 2019. 1
- [3] He Huang and Alec N Kercheval. A generalized birth–death stochastic model for high-frequency order book dynamics. *Quantitative Finance*, 12(4):547–557, 2012. 1
- [4] Martin Långkvist, Lars Karlsson, and Amy Loutfi. A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern recognition letters*, 42:11–24, 2014. 1
- [5] Maryam M Najafabadi, Flavio Villanustre, Taghi M Khoshgoftaar, Naeem Seliya, Randall Wald, and Edin Muharemagic. Deep learning applications and challenges in big data analytics. *Journal of big data*, 2(1):1–21, 2015. 1
- [6] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015. 1
- [7] Misha Denil, Babak Shakibi, Laurent Dinh, Marc’Aurelio Ranzato, and Nando De Freitas. Predicting parameters in deep learning. *Advances in neural information processing systems*, 26, 2013. 1, 2
- [8] Anita Yadav, CK Jha, and Aditi Sharan. Optimizing lstm for time series prediction in indian stock market. *Procedia Computer Science*, 167:2091–2100, 2020. 1, 2
- [9] Edoardo Conti, Vashisht Madhavan, Felipe Petroski Such, Joel Lehman, Kenneth Stanley, and Jeff Clune. Improving exploration in evolution strategies for deep reinforcement learning via a population of novelty-seeking agents. *Advances in neural information processing systems*, 31, 2018. 1
- [10] Youru Li, Zhenfeng Zhu, Deqiang Kong, Hua Han, and Yao Zhao. Ea-lstm: Evolutionary attention-based lstm for time series prediction. *Knowledge-Based Systems*, 181:104785, 2019. 1, 2, 4.1
- [11] Wu Deng, Hailong Liu, Junjie Xu, Huimin Zhao, and Yingjie Song. An improved quantum-inspired differential evolution algorithm for deep belief network. *IEEE Transactions on Instrumentation and Measurement*, 69(10):7319–7327, 2020. 1, 2
- [12] Mohammad Mahdi Rounaghi and Farzaneh Nassir Zadeh. Investigation of market efficiency and financial stability between s&p 500 and london stock exchange: monthly and yearly forecasting of time series stock returns using arma model. *Physica A: Statistical Mechanics and its Applications*, 456:10–21, 2016. 2
- [13] Miguel Ataurima Arellano and Gabriel Rodriguez. Empirical modeling of high-income and emerging stock and forex market return volatility using markov-switching garch models. *The North American Journal of Economics and Finance*, 52:101163, 2020. 2

- [14] Muhammad Abubakr Naeem, Imen Mbarki, and Syed Jawad Hussain Shahzad. Predictive role of online investor sentiment for cryptocurrency market: Evidence from happiness and fears. *International Review of Economics & Finance*, 73:496–514, 2021. 2
- [15] Tomasz Korol. Early warning models against bankruptcy risk for central european and latin american enterprises. *Economic Modelling*, 31:22–30, 2013. 2
- [16] Jiang Zhao, Zhiji Yang, and Yitian Xu. Nonparallel least square support vector machine for classification. *Applied Intelligence*, 45:1119–1128, 2016. 2
- [17] Huseyin Ince, Ali Fehim Cebeci, and Salih Zeki Imamoglu. An artificial neural network-based approach to the monetary model of exchange rate. *Computational economics*, 53:817–831, 2019. 2
- [18] Nor Azizah Hitam, Amelia Ritahani Ismail, and Faisal Saeed. An optimized support vector machine (svm) based on particle swarm optimization (psa) for cryptocurrency forecasting. *Procedia computer science*, 163:427–433, 2019. 2
- [19] Lu Han, Liyan Han, and Hongwei Zhao. Orthogonal support vector machine for credit scoring. *Engineering Applications of Artificial Intelligence*, 26(2):848–862, 2013. 2
- [20] Xiao-Li Gong, Xi-Hua Liu, Xiong Xiong, and Xin-Tian Zhuang. Forecasting stock volatility process using improved least square support vector machine approach. *Soft Computing*, 23:11867–11881, 2019. 2, 4.2
- [21] Matthew Dixon, Diego Klabjan, and Jin Hoon Bang. Implementing deep neural networks for financial market prediction on the intel xeon phi. In *Proceedings of the 8th Workshop on High Performance Computational Finance*, pages 1–6, 2015. 2
- [22] Chao Liu, Yixin Fan, and Xiangyu Zhu. Fintech index prediction based on rf-ga-dnn algorithm. *Wireless Communications and Mobile Computing*, 2021:1–9, 2021. 2
- [23] Maoguo Gong, Jia Liu, Hao Li, Qing Cai, and Linzhi Su. A multiobjective sparse feature learning model for deep neural networks. *IEEE transactions on neural networks and learning systems*, 26(12):3263–3277, 2015. 2
- [24] Oussama Lachiheb and Mohamed Salah Gouider. A hierarchical deep neural network design for stock returns prediction. *Procedia Computer Science*, 126:264–272, 2018. 2
- [25] Jia Liu, Maoguo Gong, Qiguang Miao, Xiaogang Wang, and Hao Li. Structure learning for deep neural networks based on multiobjective optimization. *IEEE transactions on neural networks and learning systems*, 29(6):2450–2463, 2017. 2
- [26] Junhao Huang, Weize Sun, and Lei Huang. Deep neural networks compression learning based on multiobjective evolutionary algorithms. *Neurocomputing*, 378:260–269, 2020. 2
- [27] Prasanna Tamilselvan and Pingfeng Wang. Failure diagnosis using deep belief learning based health state classification. *Reliability Engineering & System Safety*, 115:124–135, 2013. 2
- [28] Pinyi Zhang and Bicong Ci. Deep belief network for gold price forecasting. *Resources Policy*, 69:101806, 2020. 2
- [29] Furao Shen, Jing Chao, and Jinxi Zhao. Forecasting exchange rate using deep belief networks and conjugate gradient method. *Neurocomputing*, 167:243–253, 2015. 2
- [30] Lean Yu, Rongtian Zhou, Ling Tang, and Rongda Chen. A dbn-based resampling svm ensemble learning paradigm for credit classification with imbalanced data. *Applied Soft Computing*, 69:192–202, 2018. 2
- [31] Takashi Kuremoto, Shinsuke Kimura, Kunikazu Kobayashi, and Masanao Obayashi. Time series forecasting using a deep belief network with restricted boltzmann machines. *Neurocomputing*, 137:47–56, 2014. 2
- [32] Ha Young Kim and Chang Hyun Won. Forecasting the volatility of stock price index: A hybrid model integrating lstm with multiple garch-type models. *Expert Systems with Applications*, 103:25–37, 2018. 2
- [33] Aytaç Altan, Seçkin Karasu, and Stelios Bekiros. Digital currency forecasting with chaotic meta-heuristic bio-inspired signal processing techniques. *Chaos, Solitons & Fractals*, 126:325–336, 2019. 2
- [34] Temesgen Awoke, Minakhi Rout, Lipika Mohanty, and Suresh Chandra Satapathy. Bitcoin price prediction and analysis using deep learning models. In *Communication Software and Networks: Proceedings of INDIA 2019*, pages 631–640. Springer, 2020. 2

- [35] Hongli Niu, Kunliang Xu, and Cheng Liu. A decomposition-ensemble model with regrouping method and attention-based gated recurrent unit network for energy price prediction. *Energy*, 231:120941, 2021. 2
- [36] Ilya Loshchilov and Frank Hutter. Cma-es for hyperparameter optimization of deep neural networks. *arXiv preprint arXiv:1604.07269*, 2016. 2
- [37] Hyejung Chung and Kyung-shik Shin. Genetic algorithm-optimized long short-term memory network for stock market prediction. *Sustainability*, 10(10):3765, 2018. 2
- [38] S Ivvan Valdez and Alfonso Rojas-Domínguez. An evolutionary algorithm of linear complexity: Application to training of deep neural networks. *arXiv preprint arXiv:1907.05951*, 2019. 2