Hybrid Long Short-Term Memory prediction model improved by particle swarm optimization with sine and cosine factors

Wanyue Pei1,a*, Yanfei Yang2,b

1International college, Zhengzhou University, Zhengzhou, Henan, China
2College of Textile, Zhongyuan University of Technology, Zhengzhou, Henan, China

Abstract: The Long Short-Term Memory network of deep learning neural network is widely used to predict stock price in financial field. In order to optimize the accuracy of stock price prediction by LSTM network, this paper firstly uses principal component analysis method to extract various influencing indexes of stock. Then, use Circle mapping method to select the initial value more evenly, use sine and cosine factors to improve factors of particle swarm optimization algorithm, so as to find the optimal parameters of LSTM model more effectively. Finally, the optimization results of IPSO algorithm are substituted into the LSTM model for the regression prediction with the principal components. Through empirical analysis and comparative test, the results show that the improved particle swarm optimization algorithm proposed in this paper has better optimization effect, is not prone to local optimal problems, and the prediction model based on this method has higher prediction accuracy.

1.Introduction

Due to the high noise and nonlinear characteristics of financial data, the linear prediction method is not good for stock price prediction. Nonlinear prediction method is the application of machine learning method in financial data prediction field, including traditional machine learning method, BP neural network and deep learning network. Today, the application of machine learning to stock price prediction has yielded more accurate and efficient results.

Deep learning network is further developed on the basis of ANN to solve the problem of over-fitting and has stronger feature extraction ability. Deep learning networks are widely used in the field of image recognition and are good at processing sequence data. Long Short-Term Memory network is widely used in stock price prediction. Peng[1] used LSTM model to predict stock price, which improved the shortcoming of RNN forgetting historical status information. Song[2] used adaptive particle swarm optimization algorithm to optimize the LSTM model, proving that the optimization of APSO algorithm can improve the accuracy of prediction and has universality. Kim[3] proposed a CNN - LSTM model to predict stock price with the time feature and image feature. The experimental results prove that this model could improve the prediction accuracy effectively.

This paper further improves the LSTM model, builds an IPSO - LSTM model to make regression prediction of stock price. In terms of data processing, principal component analysis method is introduced to reduce the dimension of data, so as to maximize the retention of the original data and improve the learning speed. Additionally, the particle swarm optimization algorithm improved by a new method is used to optimize the LSTM model to improve the prediction accuracy.

2. Model building

This part will introduce the principle of a single model and show how the hybrid model is constructed.

2.1. Basic principle of model

2.1.1. Principal Component Analysis

Principal component analysis is an algorithm to reduce dimension, which can extract the principal component from some elements, and convert single index variables into comprehensive index variables. These principal components are linear combinations of the raw variables and are independent of each other. It is a widely used data simplification method that can extract the key information from the raw data. The specific calculation steps of principal component analysis are as follows.

\[ X_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}, i = 1, 2, \cdots, n; j = 1, 2, \cdots, p \]

\[ \bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_{ij}, s_j^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_{ij} - \bar{x})^2 \]

To solve the correlation coefficient matrix of the...
standardized matrix, the formula is:

\[ R = \begin{bmatrix} r_{ij} \end{bmatrix}_{p \times p} = \frac{X^T X}{n - 1}, \quad r_{ij} = \sum_{n=1}^{n-1} b_{ij}, \quad i, j = 1, 2, \ldots, p \]

Solving characteristic equation \( R - \lambda I = 0 \) to compute the eigenvalues of the correlation coefficient matrix \( \lambda_i \), feature vector \( b_i = [b_{i1}, b_{i2}, \ldots, b_{ip}]^T, \quad i = 1, 2, \ldots, p \).

The contribution rate is calculated according to the characteristic value, and the first few items with high contribution rate are selected as the main components. The cumulative contribution rate usually takes 85% or 90%. The extracted principal component is obtained by multiplying the standardized variable with the feature vector. The calculation formula is as follows:

\[ U_j = X_j b_j^*, \quad j = 1, 2, \ldots, m \] (2)

2.1.2. Long Short-Term Memory network

LSTM model is improved on the structure of Recursive Neural Network, and is widely used in long-term time dependent problem and time series prediction problem[4]. LSTM model is a special RNN model. Based on the RNN model, one cell state and three gate units are added to control the use of historical data. This improvement effectively solves the gradient explosion and gradient disappearance problem. Three additional door units are memory door, input door and output door. The formulas of gate structure control and cell update are as follows:

\[ i_t = \sigma \left[ W_i (h_{t-1}, x_t) + b_i \right] \] (3)

\[ f_t = \sigma \left[ W_f (h_{t-1}, x_t) + b_f \right] \] (4)

\[ \tilde{C}_t = \tanh \left[ W_c (h_{t-1}, x_t) + b_c \right] \] (5)

\[ C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \] (6)

\[ O_t = \sigma \left[ W_o (h_{t-1}, x_t) + b_o \right] \] (7)

\[ h_t = O_t \odot \tanh C_t \] (8)

where \( i_t \) is the degree to which cellular information is forgotten, \( \tilde{C}_t \) is the candidate value variable constructed by tanh layer, \( C_t \) is the current step cell state, \( O_t \) is the degree of output information and \( h_t \) is the output of the current step.

2.1.3. Improved particle swarm optimization method

Particle swarm optimization (PSO) is a heuristic algorithm that is influenced by the foraging behavior of birds to find the optimal solution[5]. The position of each particle is constantly updated in the search for an optimal solution. Each particle shares the individual optimal solution with the group to get the global optimal solution. Each particle continuously moves according to its own optimal solution and the global optimal solution, looking for the optimal movement position. The speed update formula and position update formula are as follows:

\[ v_{t+1} = w v_t + c_1 r_1 (pbest_t - x_t) + c_2 r_2 (gbest - x_t) \] (9)

\[ x_{t+1} = x_t + v_{t+1} \] (10)

where \( \omega \) is inertia factor, \( c_1, c_2 \) is learning factor, \( r_1, r_2 \) are random numbers between 0 to 1, \( pbest_t \) is individual optimal solution and \( gbest \) is global optimal solution.

The basic particle swarm optimization algorithm has the inherent defect named local optimization, which is prone to premature convergence when solving nonlinear multi-modal functions. At present, there are many ways to improve the PSO algorithm.

In order to avoid local optimization, in the early stage of the optimization process, particles are required to have higher global search ability, while in the late stage, higher local search ability is required, so as to improve the optimal results. Based on this, the current improvement of PSO algorithm focuses on the improvement of inertia factor and learning factor. A function is used to define the factor to make it change dynamically to meet the need of optimization in different periods. The common improvement method is to make the inertia factor and learning factor meet a decreasing process with linear decreasing function or exponential function. The adaptive inertia weight adjusts the size of \( \omega \) with secant function[6]. The function formulas are as follows:

\[ c_1 = c_{max} - (c_{max} - c_{min}) \frac{k}{K} \] (11)

\[ c_2 = c_{min} + (c_{max} - c_{min}) \frac{k}{K} \] (12)

\[ \omega = \omega_{max} - (\omega_{max} - \omega_{min}) \left\{ \sec \left[ \frac{k}{K} \cdot \frac{\pi}{3} \right] - 1 \right\} \] (13)

where \( c_1, c_2 \) are two learning factors, \( c_{max}, c_{min} \) are the upper and lower bounds of the learning factor, \( \omega \) is inertia factor and \( \omega_{max}, \omega_{min} \) are the upper and lower bounds of the learning factor.

For previous methods, the effect of the improved learning factor is prone to be over-imposed with inertia factor. Although the convergence rate is increased, it is prone to reduce the population diversity and fall into the local optimal. Xu[7] proposed the optimization method of using sine and cosine factors instead of learning factors. The learning factor is no longer a simple monotonically increasing or decreasing trend, but forms an overall attenuation trend of oscillation, which better maintains the balance between global search and local search. The speed update formulas under this method are shown below:

\[ v_{t+1} = \omega v_t + r_1 \cdot \sin r_2 \cdot (pbest_t - x_t) \] (14)

\[ + r_1 \cdot \sin r_2 \cdot (gbest - x_t) \]

\[ v_{t+1} = \omega v_t + r_1 \cdot \cos r_2 \cdot (pbest_t - x_t) \] (15)

\[ + r_1 \cdot \cos r_2 \cdot (gbest - x_t) \]

\[ r_1 = 2 - 2 \cdot \frac{k}{K} \] (16)

Where \( r_2 \) is a random number between \([0, 2\pi]\) and \( p \) is a random number between 0 to 1. When \( p \geq 0.5 \), the updating rate of sine factor is adopted. When \( p < 0.5 \), the cosine factor is used to update the example speed, thus probabilistically switching sine cosine function as the
learning factor.

The first step of PSO algorithm is to select the initial value of particle swarm. In this paper, the method of Circle mapping is used to select the initial value, which could select the initial value more evenly and is conducive to generating a better search space and improving the algorithm convergence accuracy\([7]\). The expression of the chaotic sequence generated by the Circle mapping is:

\[
num_{i+1} = \text{mod} \left( num_i + 0.2 - \frac{0.5}{2\pi} \times \sin(2\pi \times num_i), 1 \right)
\]

(17)

The formulas for initializing particle swarm velocity and position are as follows:

\[
v_{ij} = v_{ij} + (v_{ij} - v_{ib}) \times num_{ij}
\]

(18)

\[
x_{ij} = x_{ij} + (x_{ij} - x_{ib}) \times num_{ij}
\]

(19)

2.2. PCA-IPOSO-LSTM modeling

The algorithm flow is shown in the figure 1. The detailed steps of the algorithm are as shown below.

- Multiple influencing indicators of stock price are selected as samples of regression prediction. The principle components extracted by PCA method are used as the input of the LSTM model, and the closing price of the next day is used as the output.
- Divide the data into two parts: one is the training set, and the other is the testing set. Use the training set to train the model, and use the test set to verify the predictive effect of the model.
- Circle mapping is used to select the particle swarm’s initial value evenly. Using sine and cosine factors instead of learning factors for particle swarm optimization. The root mean square error of the prediction result is taken as the fitness function, and the optimization objective is the minimum value of this function.
- After reaching the maximum number of iterations, the optimal parameters of the LSTM model are obtained.
- The optimized parameters are substituted into LSTM model for training and prediction. Model can be used to predict stock prices

![Figure 1. Flowchart of the entire algorithm](image)

3. Empirical analysis

3.1. Data processing

The data used in this paper is the stock data of Ping An Bank (000001.SZ) from January 1, 2015 to February 28, 2023, which comes from the Tushare library of Python. Based on the original data, 35 technical indicators of different types are extracted, include KDJ, OBV, PSY etc. Some data will be lost in the calculation, and the final number of samples used for training is 1966, of which 80% is used as the training set and 20% as the test set. The principal component analysis method is used to process the raw data to obtain the principal components of the stock technical indicators. After calculation, the results of principal component analysis are shown in Table 1. The first eight indexes with a cumulative contribution rate of 90% are selected as the main components.

<table>
<thead>
<tr>
<th>Number</th>
<th>Eigenvalue</th>
<th>Contribution rate (%)</th>
<th>Cumulative contribution rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12.0223</td>
<td>0.3435</td>
<td>0.3435</td>
</tr>
<tr>
<td>2</td>
<td>10.3459</td>
<td>0.2956</td>
<td>0.6391</td>
</tr>
<tr>
<td>3</td>
<td>2.7221</td>
<td>0.0778</td>
<td>0.7169</td>
</tr>
<tr>
<td>4</td>
<td>2.2245</td>
<td>0.0636</td>
<td>0.7804</td>
</tr>
<tr>
<td>5</td>
<td>1.7438</td>
<td>0.0498</td>
<td>0.8302</td>
</tr>
<tr>
<td>6</td>
<td>1.0794</td>
<td>0.0308</td>
<td>0.8611</td>
</tr>
<tr>
<td>7</td>
<td>0.8611</td>
<td>0.0246</td>
<td>0.8857</td>
</tr>
<tr>
<td>8</td>
<td>0.6699</td>
<td>0.0191</td>
<td>0.9048</td>
</tr>
<tr>
<td>9</td>
<td>0.6196</td>
<td>0.0177</td>
<td>0.9225</td>
</tr>
<tr>
<td>......</td>
<td>......</td>
<td>......</td>
<td>......</td>
</tr>
</tbody>
</table>
3.2. Prediction result analysis

In order to verify the effect of the PCA-IPSO-LSTM model, three control groups of comparative experiments are set up.

The first group of PCA-LSTM model directly uses the LSTM model, and the parameters are set according to experience. The second group of PCA-PSO-LSTM model is optimized by using unimproved particle swarm optimization method. The third group of PCA-APSO-LSTM improves the inertia factor and learning factor of PSO only by monotone decreasing function.

The optimization results of IPSO method are shown in Table 2. Figure 2 shows the change of the adaptation function in the optimization process of particle swarm optimization algorithm improved by different methods. By comparison, it can be found that the IPSO algorithm has a faster decline in its fitness function and can search for better values faster. The unimproved PSO algorithm is likely to fall into local optimization because its adaptation function remains unchanged in finite iterations.

<table>
<thead>
<tr>
<th>Table 2. The result of IPSO optimization method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimized parameter</td>
</tr>
<tr>
<td>Learning rate</td>
</tr>
<tr>
<td>Number of hidden layer nodes</td>
</tr>
<tr>
<td>Regularized parameter</td>
</tr>
</tbody>
</table>

Figure 2. Different adaptation function change curves

Compare the stock price prediction effect under different optimization models. Table 3 shows the error of different models when predicting the test set, among which the PCA-IPSO-LSTM model has the smallest error of all types. The results prove that the model is effective in improving the accuracy of LSTM model in predicting stock price.

<table>
<thead>
<tr>
<th>Table 3. Comparison of prediction errors of different model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
</tr>
<tr>
<td>PCA-LSTM</td>
</tr>
<tr>
<td>PCA-PSO-LSTM</td>
</tr>
<tr>
<td>PCA-APSO-LSTM</td>
</tr>
<tr>
<td>PCA-IPSO-LSTM</td>
</tr>
</tbody>
</table>

4. Conclusions

In this paper, a PCA-IPSO-LSTM hybrid network model is constructed for stock price prediction. Use the PCA method to reduce the dimension of raw data to reduce the computation and shorten the training time of the model. The improved PSO algorithm is used to find the optimal parameters of the LSTM model. As for the improvement of the PSO algorithm, the oscillatory property of sine and cosine factors is used in this paper to make the parameter factors of the PSO algorithm meet the oscillation attenuation, which improves the effect of finding the optimal parameters of the LSTM model and the prediction accuracy of the hybrid model. The empirical study set up three control groups for comparative verification. The results prove that the IPSO model could effectively improve the defect of local optimization, and the PCA-IPSO-LSTM model has higher prediction accuracy when predicting stock prices.

References