CrossMark

Financial Time Series Prediction Based on Deep Learning

Hongju Yan[1] · Hongbing Ouyang[1]

Abstract By combining wavelet analysis with Long Short-Term Memory (LSTM) neural network, this paper proposes a time series prediction model to capture the complex features such as non-linearity, non-stationary and sequence correlation of financial time series. The LSTM is then applied to the prediction of the daily closing price of the Shanghai Composite Index as well as the comparison of its prediction ability with machine learning models such as multi-layer perceptron, support vector machine and K-nearest neighbors. The empirical results show that the LSTM performs a better prediction effect, and it shows excellent effects on the static prediction and dynamic trend prediction of the financial time series, which indicates its applicability and effectiveness to the prediction of financial time series. At the same time, both wavelet decomposition and reconstruction of financial time series can improve the generalization ability of the LSTM prediction model and the prediction accuracy of long-term dynamic trend.

Keywords Deep learning · Long Short-Term Memory neural network · Wavelets · Financial time series prediction

1 Introduction

In order to construct a scientific and effective financial time series prediction model, the following characteristics must be considered. Firstly, it should be able to fully reflect complex features such as non-linearity, non-stationary and sequence correlation of financial time series data. Secondly, it can effectively reflect the non-linear dynamic interaction of the influencing factors of financial time series. The third is to have a powerful and robust feature learning ability that can extract important information that affects the volatility of

✉ Hongbing Ouyang
hellobotong@163.com

[1] School of Economics, Huazhong University of Science and Technology, 1037 Luoyu Road, Wuhan, Hubei, China

financial time series. Therefore, the prediction of financial time series has become very challenging, and possesses important academic and practical value.

Traditional time series prediction models such as ARMA and GARCH are used to describe the financial time series by assuming a specific model. However, the complex real-time series containing noises that cannot be reflected by the analytical equation with parameters. This is because the dynamic equation of financial time series is either too complicated or unknown [1]. Therefore, the traditional econometric model for time series prediction has some limitations.

However, machine learning, especially deep learning, exhibits great advantages in processing non-linear and non-stationary data, capturing the non-linear interactions among feature vectors, as well as the extraction of data features. And these algorithms have been widely and successfully used in many fields of artificial intelligence, such as image recognition [2], natural language processing [3], driver-less car and so on. This tremendous success has proven the statement. Hence, does the machine learning, especially the deep learning, has applicability in financial time series prediction? Can it improve the prediction accuracy of financial time series? These questions deserve our in-depth investigation. This is of great importance to the expansion of research methods for financial time series prediction. This is also very important to achieve the financial prediction based on massive data under the background of big data and to monitor the financial market risk as well as to explore the law of financial operation.

Machine learning enables the computer programs to learn automatically based on accumulated experience. In recent years, scholars have made a lot of meaningful explorations in financial time series prediction based on different machine learning models such as support vector machine (SVM) [4], BP neural network [5]. Ahmed et al. [6] compared the prediction capabilities of time series with different machine learning algorithms, such as multi-layer perceptron, K-nearest neighbors, classification and regression tree, support vector regression, Gaussian process. It was found that the multi-layer perceptron and the Gaussian process have better regression effect.

The SVM, BP neural network and other shallow machine learning algorithms have many limitations in learning complex, high-dimensional data, and there are many problems such as dimensionality and invalid feature representations [7]. The deep learning is to further deepen the number of hidden layers in the neural network. Meanwhile, it combines the underlying features to form a more abstract feature of high-level representation in order to discover the distributed characteristics of the input data.

As compared with shallow machine learning, the deep learning has the following two advantages: First, deep learning that used an unsupervised learning layer-by-layer feature extraction with more powerful features of the expression can learn more complex function representation. Second, deep learning is more likely to alleviate the overfitting problem while improving the accuracy of sample prediction [8], and thus have a stronger generalization ability. The great success of deep learning has been confirmed in the field of complex time series data such as speech recognition [9], video processing [10], etc.

In recent years, scholars have begun to explore the applicability of deep learning algorithms in financial time series prediction. Recurrent neural network (RNN) can achieve better predictive results in the prediction of stock prices [11]. The RNN is a neural network that processes time series data and incorporating the sequence dependency. In other words, the current state in the learning process will contain all the historical information of the previous time series. However, the problem of vanishing gradient or exploding gradient may exist in the standard RNN, and there is some difficulty in learning the long-term dependency of the sequence [12]. On the other hand, the Long Short-Term Memory

(LSTM) neural network that contains memory modules solves the problem of long-term dependency of sequences and achieved significant success in the field of sequential data applications such as sequential text translation [13].

In conclusion, the LSTM neural network can reflect the non-linearity of financial time series data and the complex interaction among data, which embodies the sequence-related features of time series and avoid many problems existing in financial time series prediction method based on the traditional econometric model and the shallow machine learning. However, there is little research on the applicability and effectiveness of LSTM neural network in financial time series prediction. Therefore, this paper proposes a combination of the LSTM neural network and wavelet analysis to construct financial time series prediction model. At the same time, the Shanghai Composite Index is taken as an example for empirical prediction and the prediction effect is compared with other machine learning models such as MLP, SVM, K-nearest neighbors in order to explore the effectiveness of LSTM neural network in the prediction of actual financial time series.

2 Model Building

2.1 Wavelet Analysis

Financial time series are susceptible to many factors, such as politics, economy and investor psychology, etc., which usually contain a lot of noise and show significant non-stationary characteristics. The deep neural network has a strong ability to deal with non-stationary data. However, it will undoubtedly reduce the generalization ability of the model if it pays more attention to the fitting of noise data in the training process, thereby reducing its prediction ability on the out-of-sample financial time series.

Wavelet analysis can perform multi-scale refinement analysis on the signal by computing functions such as stretching, translating, etc. The wavelet also exhibits considerable superiority in dealing with non-stationary time series. Wavelet transform can effectively eliminate the noise in the financial time series and fully preserve the characteristics of the original signal [14]. Therefore, this paper proposes the data preprocessing of the financial time series using wavelet transform. The noise components of high frequency in the time series are eliminated by "wavelet denoising" to reduce the influence of short-term noise disturbance on the neural network structure, as well as to improve the prediction ability of the model.

Similar to Fourier analysis, the basic principle of wavelet analysis is to use a set of wavelet functions to represent or approximate a signal or function. It is called as a basis wavelet function if $\psi(t) \in L^2(R)$ and the Fourier transform $\hat{\psi}(w)$ satisfies the admissible condition:

$$C_\psi = \int_R \frac{\left|\hat{\psi}(w)\right|^2}{|w|} dw < \infty \tag{1}$$

The wavelet $\psi(t)$ is scaled by scaling and time translation to obtain sub wavelet:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{|a|}}\psi\left(\frac{t-b}{a}\right) \tag{2}$$

where a is the scaling factor, which reflects the length of the wavelet periodic, and b is the translation factor, which reflects the displacement on the wavelet time. For any $f(t) \in L^2(R)$, its continuous wavelet transform is:

$$W_f(a,b) = \langle f(t), \psi_{a,b}(t)\rangle = \frac{1}{\sqrt{|a|}}\int_R f(t)\bar{\psi}\left(\frac{t-b}{a}\right)dt \tag{3}$$

where $W_f(a,b)$ is the wavelet transform coefficients, and $\bar{\psi}\left(\frac{t-b}{a}\right)$ is the complex conjugate function of $\psi\left(\frac{t-b}{a}\right)$.

For discrete financial time series data, scaling factor a and the translation factor b need to be discretized separately. Assuming that $a = a_0^j$, $b = ka_0^j b_0$, $k,j \in Z$, then the discrete wavelet function is:

$$\psi_{j,k}(t) = a_0^{-\frac{j}{2}}\psi\left(a_0^{-j}t - kb_0\right) \tag{4}$$

The corresponding discrete wavelet transform is:

$$WT_f(j,k) = \int f(t)\bar{\psi}_{j,k}(t)dt \tag{5}$$

As shown in Eqs. (4) and (5), the wavelet analysis obtains the low frequency or high frequency information of the financial time series data signal by increasing or decreasing the scaling factor a, thereby analyzing the outline of the sequence signal or details of the information. Specifically, when the scaling factor a is small, time view window range is very small. At this point, the detailed observation on the use of high-frequency wavelet on the financial time series signal was carried out. On the contrary, when the scaling factor a is large, time view window range is very large. At this time, the detailed observation on the use of low-frequency wavelet on the financial time series signal was carried out. Hence, the long-term overall trend was reflected.

The concept of multi-resolution analysis when constructing orthogonal wavelet and the Mallat algorithm are proposed to realize the wavelet decomposition and reconstruction. The multi-resolution analysis decomposes each layer of the input signal into a low-frequency signal and a high frequency signal and only the low-frequency part is decomposed (as shown in Fig. 1). The mathematical expression is: $C_0 = C_1 + D_l + D_{l-1}\cdots + - D_2 + D_1$, C_0 is the original financial time series signal whereas $C_1, C_2,..., C_l$ and $D_1, D_2, ..., D_l$ are the 1, 2,..., lth low frequency and high frequency signals of wavelet decomposition respectively.

In order to achieve the purpose of de-noising, Mallat wavelet reconstruction is carried out according to the low frequency coefficient of the Nth layer and the high frequency coefficient of $1 \sim N$ layers of the wavelet decomposition, and the high frequency part is set to zero [14]. The low frequency part of the wavelet decomposition of the financial time series data reflects the overall trend of the sequence, and the high frequency part reflects the short-term random disturbance of the sequence. Therefore, setting the high frequency part to zero can eliminate the noise and smooth signal and obtain an approximate signal of the original financial time series data. This can avoid the excessive learning of the neural

**Fig. 1** Wavelet decomposition of financial time series
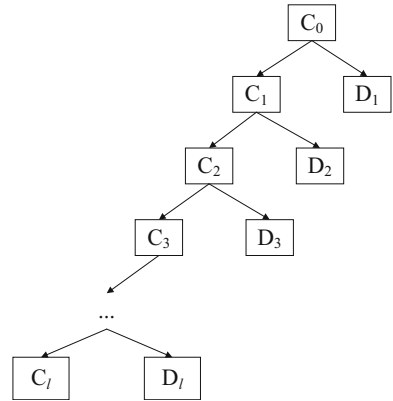
network structure caused by short-term random disturbance factors and improve the extrapolation ability of the model.

2.2 Deep Neural Network Structure

Extracting robust features to capture relevant information about data has a crucial impact on the ability to better simulate complex real-world data [15]. The deep learning based on deep neural network can set different layers of hierarchical feature representation to learn the more abstract and nonlinear features that exist in data according to a given task. For this deep neural network, which is superimposed by multi-level feature extraction, it is no doubt that it is easier to capture and simulate complex structures of the data. Therefore, this paper proposes the use of deep neural network to simulate complex financial time series so as to predict financial time series.

Artificial neural network (ANN) is a widely interconnected network of simple units (or nodes) with self-adaptability, and its organization can simulate the interaction of biological nervous systems to real-world objects. The architecture of ANN (shown in Fig. 2) contains the input layer, the hidden layer, and the output layer, and more hidden layers can fit more
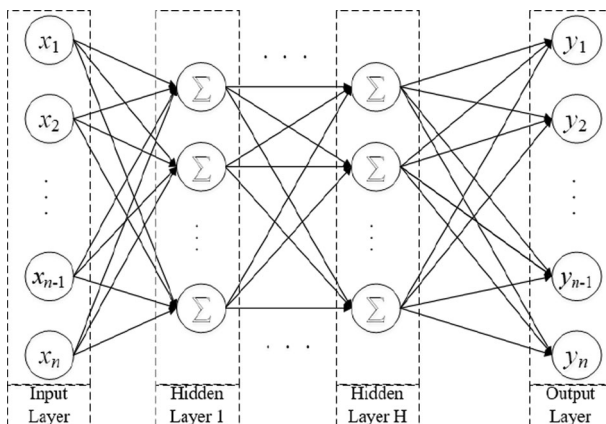
Fig. 2 Architecture of ANN

complex functions. However, the number of hidden layers of traditional neural networks is usually less, and the network training is usually more difficult when the hidden layer is more than two or three layers [16]. Therefore, the neural network of shallow structure has limited ability to express complex functions. Moreover, the commonly used BP training algorithms based on feed forward propagation and error back propagation have problems such as slow training speed and easy to fall into local optimal solution.

Deep neural network is to further deepen the hidden layers of the network, and to introduce more effective algorithm to overcome problems existed in the BP optimization algorithm. In the deep neural network, the unsupervised learning algorithm is used to learn the data characteristics layer by layer start from the input layer data. Each layer is extracted as a feature of the next layer of data input, that is, bottom-up unsupervised feature learning extraction. Thus, most neural networks are linked in a chained structure, and each layer is a function of the upper layer output. In this structure, the output of the hidden layer 1 is:

$$h^{(1)} = g^{(1)}(W^{(1)}x + b^{(1)}) \tag{6}$$

where the weight matrix $W^{(1)} = \begin{bmatrix} W_{11}^{(1)} & W_{12}^{(1)} & \cdots & W_{1n}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} & \cdots & W_{2n}^{(1)} \\ \cdots & \cdots & \cdots & \cdots \\ W_{k1}^{(1)} & W_{k2}^{(1)} & \cdots & W_{kn}^{(1)} \end{bmatrix}$, $W_{ij}^{(1)}$ is the weight between

the ith unit of the input layer and the jth unit of the hidden layer 1. The weight matrix can reflect the non-linear interaction between the different influencing factors (i.e., the input feature vector) of the financial time series where the input feature vector $x = (x_1, x_2, \ldots, x_n)^T$, the output vector $h^{(1)} = \left(h_1^{(1)}, h_2^{(1)}, \ldots, h_k^{(1)}\right)^T$, the bias $b^{(1)} = \left(b_1^{(1)}, b_2^{(1)}, \ldots, b_n^{(1)}\right)^T$, n is the number of input units, k is the number of hidden units of the hidden layer 1. $g^{(1)}$ is an activation function that implements a non-linear transformation of the weighted data. In a multi-level superimposed deep neural network, the non-linear activation function enables a more expressive model to learn more abstract feature representation [15].

Based on the feature of the hidden layer 1 learning, the output of the hidden layer 2 is:

$$h^{(2)} = g^{(2)}\left(W^{(2)}h^{(1)} + b^{(2)}\right) \tag{7}$$

As a result, the final output layer is:

$$\hat{y}(x) = g^{(y)}\left(W^{(y)}h^{(H)} + b^{(y)}\right) \tag{8}$$

In the case of a given deep neural network structure and the activation function, the learning goal is based on the top-down supervisor learning fine-tuning to find the optimal parameters $\hat{W} = \left(\hat{W}^{(1)}, \hat{W}^{(2)}, \ldots, \hat{W}^{(y)}\right)$, $\hat{b} = \left(\hat{b}^{(1)}, \hat{b}^{(2)}, \ldots, \hat{b}^{(y)}\right)$, inducing the minimum empirical risk, which is:

$$\min \frac{1}{N} \sum_{i=1}^{N} L(y_i, \hat{y}_i(x_i)) \tag{9}$$

with the loss function (i.e., the error measurement) takes the form of an absolute error:

$$L(y_i, \hat{y}_i(x_i)) = |y_i - \hat{y}_i(x_i)| \tag{10}$$

When the sample size is small, the empirical risk is the least likely to lead to the "over-fitting" problem of the deep neural network. At this point, it is necessary to further introduce the regularization term or penalty term that represents the complexity of the network structure to prevent overfitting. That is:

$$\min \frac{1}{N} \sum_{i=1}^{N} L(y_i, \hat{y}_i(x_i)) + \lambda J(\theta) \tag{11}$$

in which $J(\theta)$ reflects the complexity of the network structure, the more complex the network structure θ, the greater the $J(\theta)$. The $\lambda \geq 0$ is used to weigh the empirical risk and the complexity of the network structure. The structured model with low risk has better ability to predict training set and test set. The most common parameter penalty is known as the weight attenuation of the L^2 parameter norm penalty and the strategy is to add a regular $J(f) = \frac{1}{2}\|W\|_2^2 = \frac{1}{2}W^T W$ to the objective function [17].

Therefore, the training process of deep neural network consists of two steps: the unsupervised learning feature extraction from down to top and the fine-tuning of super-vised learning from top to down. Based on the data of the input feature vector and the output vector, the appropriate optimization algorithm is used to determine the depth of the neural network structure, as well as the super parameters in the deep neural network such as the learning rate of the optimization, penalty parameters, etc. Finally, the prediction of out-of-sample financial time series can be achieved based on the trained deep neural network model.

2.3 LSTM Neural Network Structure

Sequence relevance is a very important feature of financial time series and is indispensable in building predictive models. However, the output of the deep neural network shown in Fig. 2 is only relevant to the current input and has nothing to do with past or future input. Thus, historical information or sequence dependency features of financial time series cannot be captured. The Recurrent neural network (RNN; as shown in Fig. 3) is a neural network that can process time series data, and its current state contains the previous information for the entire time series.

For RNN, the output of the hidden layer 1 at time t is as:
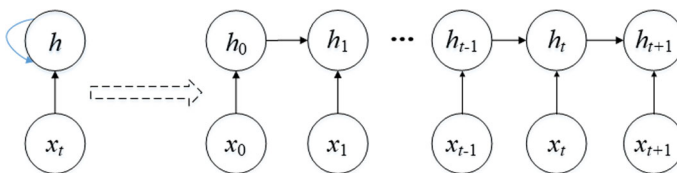
Fig. 3 Architecture of RNN

$$h_t^{(1)} = g^{(1)}\left(W^{(1)}x_t + \mathbf{U}^{(1)}h_{t-1}^{(1)} + b^{(1)}\right) \tag{12}$$

where $U^{(1)} = \begin{bmatrix} U_{11}^{(1)} & U_{12}^{(1)} & \cdots & U_{1k}^{(1)} \\ U_{21}^{(1)} & U_{22}^{(1)} & \cdots & U_{2k}^{(1)} \\ \cdots & \cdots & \cdots & \cdots \\ U_{k1}^{(1)} & U_{k2}^{(1)} & \cdots & U_{kk}^{(1)} \end{bmatrix}$, $U_{ij}^{(1)}$ is the value of the ith hidden unit of the

hidden layer 1 at time t and the jth hidden unit at time $t - 1$, x_t is the value of the input vector x at time t.

In other words, in the RNN, the hidden layer 1 can be expressed as:

$$h_t = f(x_t, x_{t-1}, \ldots, x_1) \tag{13}$$

While the hidden layer 1 that does not take time dependency (Eq. 1) is:

$$h_t = f(x_t) \tag{14}$$

The RNN contains the historical information of the input time series data, which can reflect the sequence-related characteristics of the financial time series. However, due to the existence problem of vanishing gradient or exploding gradient in the Recurrent neural network, the revelation of historical information of financial time series is very limited. Long Short-Term Memory (LSTM) neural network is a kind of special RNN which can deal with the long-term dependency of time series data. The LSTM neural network structure (shown in Fig. 4) contains a series of reconnected subnets (i.e., memory modules), each memory module contains one or more self-connected cells, as well as three threshold unit systems such as the input gate, output gate and forget gate for the control information flow.

In LSTM neural network, the execution steps can be summarized as follows.

First, the information removed from the cell is determined by the forget gate f_t,

$$f_t = \sigma(b_f + W_f x_t + U_f h_{t-1}) \tag{15}$$

where σ is the sigmoid activation function, the information flow weight is set to a value between 0 and 1, 0 means that the information is completely deleted, and 1 means that all information is retained. x_t is the current input vector, h_t is the current hidden layer vector, and b_f, W_f, U_f are the bias, the input weight and the weight of forget gate.
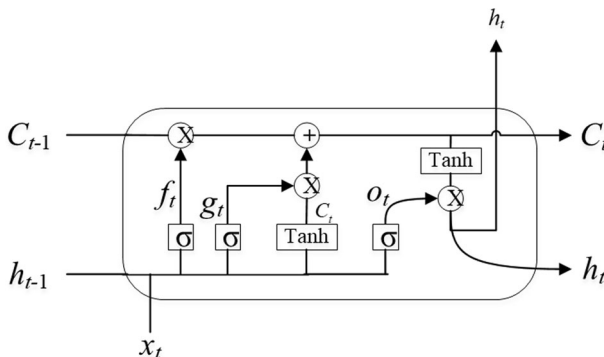

Fig. 4 Architecture of LSTM

Second, the information status of the cells is updated. Let g_t be an external input gate between 0 and 1 controlled by the sigmoid activation function:

$$g_t = \sigma\left(b_g + W_g x_t + U_g h_{t-1}\right) \tag{16}$$

The updated cell status C_t based on C_{t-1} is:

$$C_t = f_t * C_{t-1} + g_t * \tanh(b_c + W_c x_t + U_c h_{t-1}) \tag{17}$$

Finally, the output information controlled by output gate O_t is:

$$h_t = o_t * \tanh(C_t) \tag{18}$$

with which the output gate (function controlled by sigmoid activation) is:

$$o_t = \sigma(b_o + W_o x_t + U_o h_{t-1}) \tag{19}$$

It can be seen that the LSTM neural network not only contains the external recurrent between the hidden layer elements involved in the RNN, but also contains self-circulation within the cell. The information input, update and output in the threshold system unit are controlled so that the LSTM memory module cells can store and evaluate long-term historical information [18]. Therefore, the LSTM neural network model is more feasible for the revelation of historical information of financial time series and the consideration of sequence dependency. Theoretically, the LSTM is feasible for establishing financial time series.

3 Empirical Prediction of Financial Time Series

In order to explore the applicability and validity of LSTM to the prediction of the actual financial time series, the financial forecasting model based on wavelet analysis and LSTM neural network is applied to the prediction of the daily closing price of Shanghai Composite Index. The data samples were selected from January 4, 2012 to June 31, 2017 where the data is taken from Wind information.

Table 1 Index selection

Feature vector	Index	Feature vector	Index
x_1	Closing price of yesterday	x_8	5-day Moving Average (MA (5))
x_2	Opening price of yesterday	x_9	20-day Moving Average (MA (20))
x_3	Highest price of yesterday	x_{10}	12-day Exponential Moving Average (EXPMA (12))
x_4	Lowest price of yesterday	x_{11}	Moving Average Convergence/Divergence(MACD)
x_5	Volume of yesterday	x_{12}	Relative Strength Index (RSI (9))
x_6	5-day Average Trading Volume (ATV (5))	x_{13}	Directional Movement Index (DMI)
x_7	20-day Average Trading Volume (ATV (20))	x_{14}	Volume Ratio

3.1 Index Selection and Data Description

Based on the availability of the existing research and data, the following indexes (Table 1) are selected as the feature input vector to construct LSTM neural network model of the stock index prediction.

In deep learning, the sample data is usually divided into training set, validation set and test set, the data division is listed in Table 2. The training set is used to estimate the model parameters (such as the weight matrix W), the validation set is used to adjust the neural network structure (such as the number of hidden layers, the number of the hidden units), the test set is to evaluate the generalization ability of the trained model.

During the training of deep neural networks, the sample data need to carry out standardized treatment, which plays an important role in the best effect of the deep learning algorithm. For the time series $x_1, x_2, \ldots, x_t$, its standardization is:

$$y_i = \frac{x_i - \bar{x}}{s}, \quad i = 1, 2, \ldots, t \tag{20}$$

where $\bar{x} = \frac{\sum x_i}{t}, s = \sqrt{\frac{1}{t-1}\left(\sum x_i - \bar{x}\right)^2}$.

3.2 Parameter description

The selection of the activation function is a crucial part in the training of the neural network which can learn the non-linear factors in the data. This paper employs the commonly used LeakReLu activation function which possesses a faster convergence rate during training. It is in the form of:

$$f(x) = \begin{cases} \alpha x & x < 0 \\ x & x \geq 0 \end{cases} \tag{21}$$

The RMSProp algorithm is proven to be an efficient and practical deep neural network training algorithm which is one of the best algorithms used by deep learning practitioners [17]. Therefore, the optimization algorithm of the LSTM training is based on RMSProp algorithm. In order to prevent overfitting of the model training, the D'ropout method is further applied during the training process based on the introduction of penalties which can randomly remove some hidden units in the hidden layer.

3.3 Empirical Results

In order to improve the generalization ability of the predicting model, this paper first uses the sym4 wavelet basis to decompose the daily closing price of the Shanghai Composite

Table 2 Description of sample data division

Data	The number of days of trading	Time
Training set	1077	2012/01/04–2016/06/13
Validation set	135	2016/06/14–2016/12/28
Test set	135	2016/12/29–2017/06/30

Index (SHI) into 4 layers and to reconstruct the financial time series. The closing price of the SHI is shown in Fig. 5, whereas the closing price of the SHI after reconstruction is shown in Fig. 6. By comparing both Figs. 5 and 6, it is found that the financial time series after wavelet reconstruction can effectively smoothen the raw data and retain its approximation signals. Therefore, it is feasible to establish the prediction model based on the wavelet reconstruction data.


Fig. 5 Closing price of SHI
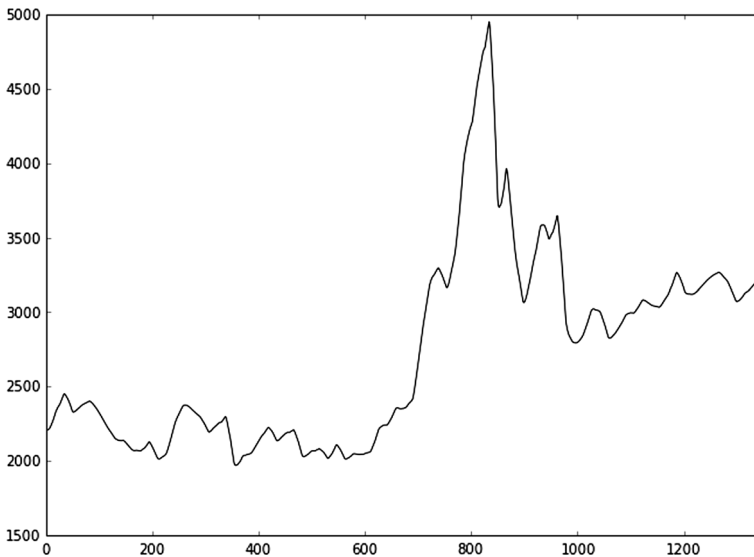
Fig. 6 Closing price of SHI after wavelet reconstruction

Furthermore, this paper adopts the financial time series of the wavelet reconstrcuted data and raw data to explore the prediction effect of LSTM and the applicability of the wavelet resconstruction of financial time series in empirical analysis. The prediction effects are shown in Figs. 7, 8 and 9. It can be found that the LSTM neural networks of the wavelet reconstructed data and the raw data show excellent prediction effect on the training
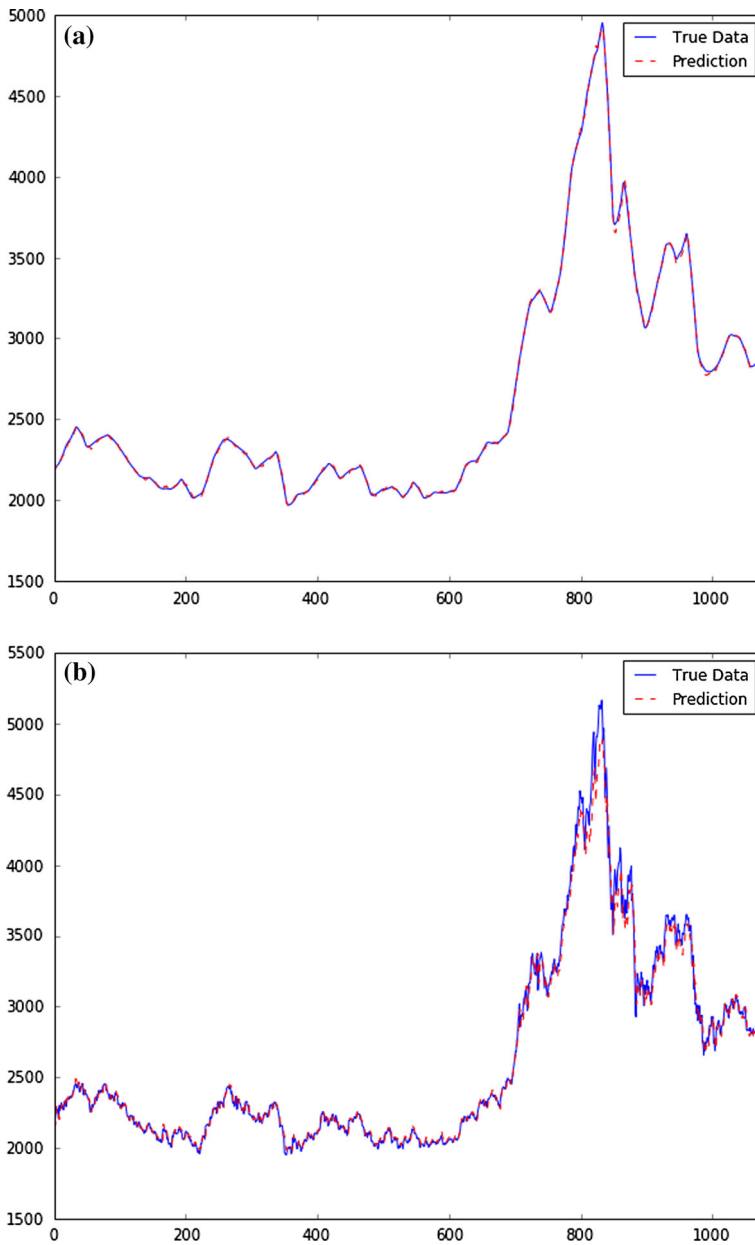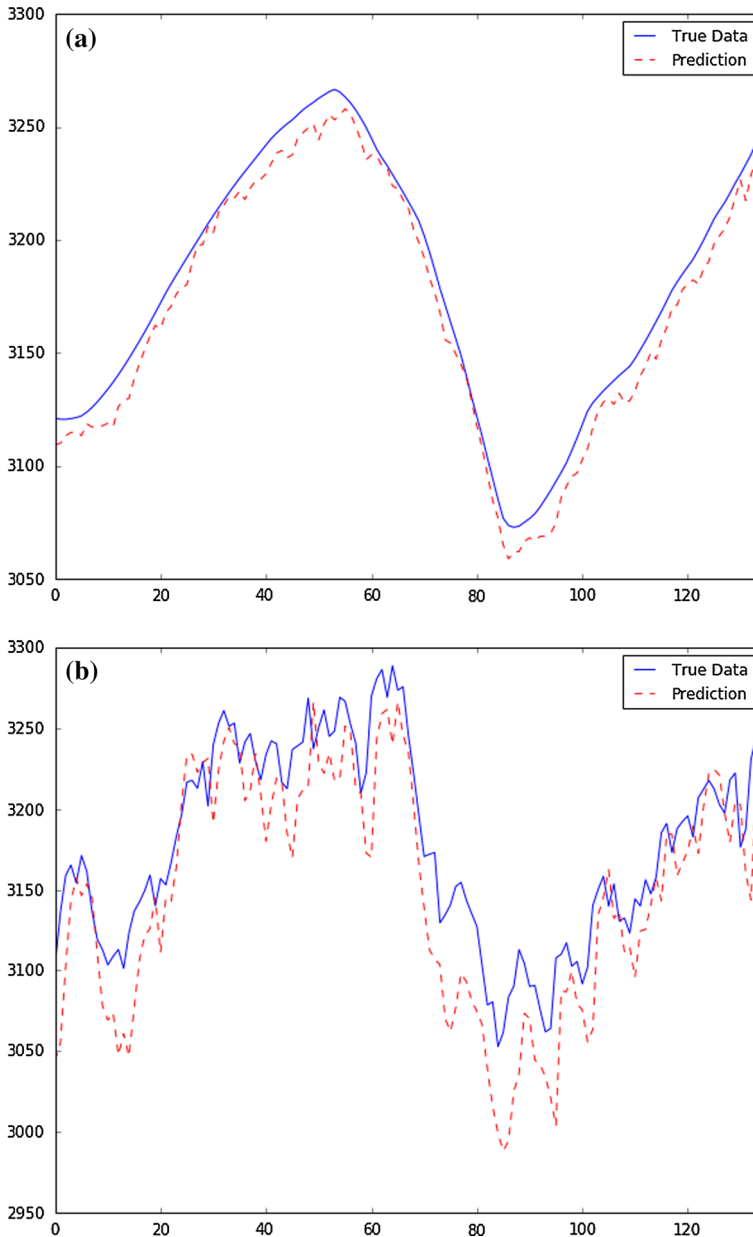

Fig. 7 **a** Prediction effect of training set (reconstruction). **b** Prediction effect of training set (raw)
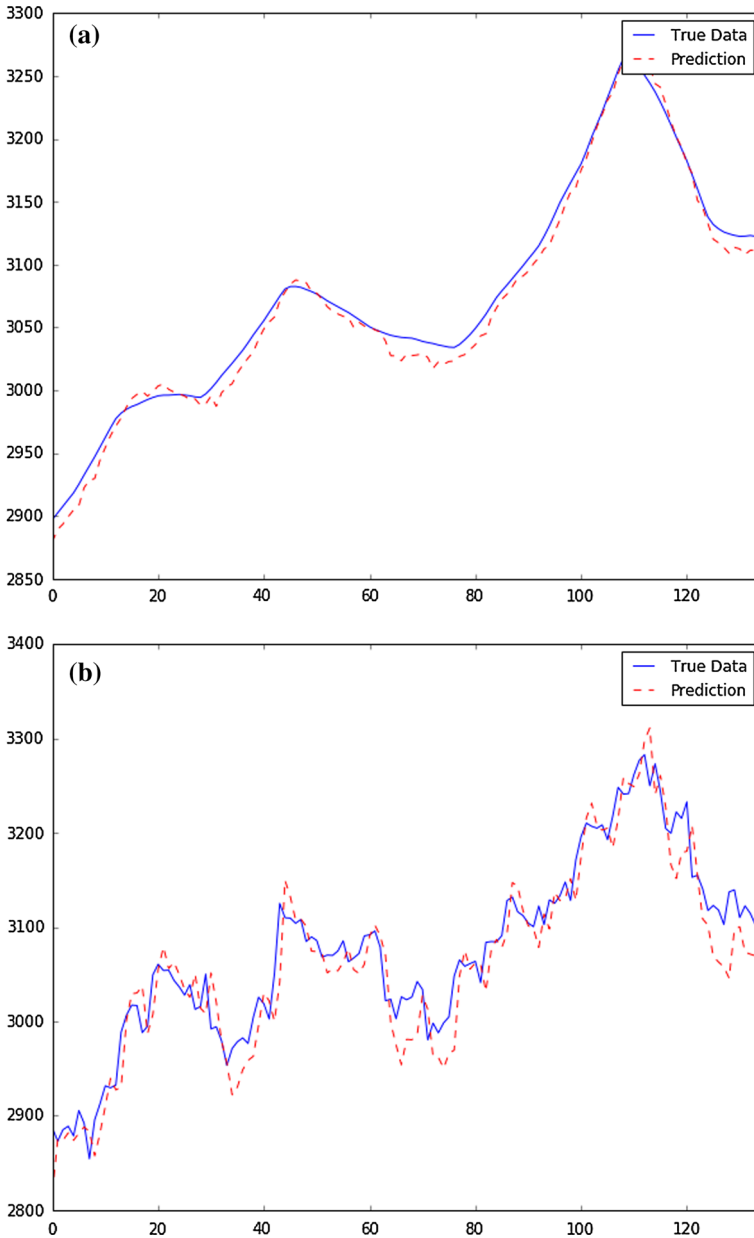
Fig. 8 **a** Prediction effect of validation set (reconstruction). **b** Prediction effect of validation set (raw)

set according to the empirical results. For the validation set, the prediction effect based on wavelet reconstruction is slightly better than that based on the raw data. For the test set, the prediction effect based on wavelet reconstruction data is obviously better than that based on the original data. This aspect shows that the LSTM neural network has a good ability to predict the actual financial time series. On the other hand, the wavelet decomposition and

Fig. 9 **a** Prediction effect of test set (reconstruction). **b** Prediction effect of test set (raw)

reconstruction of the financial time series data can effectively improve the generalization ability of the LSTM neural network and improve the prediction ability of the financial extrinsic data.

Figures 7, 8 and 9 above display the point-to-point static prediction effect, which cannot fully reflect the prediction ability of LSTM neural network. Thus, further exploration of the

Fig. 10 **a** Prediction effect of training set (the next 10 days). **b** Prediction effect of validation set (the next 10 days). **c** Prediction effect of test set (the next 10 days)
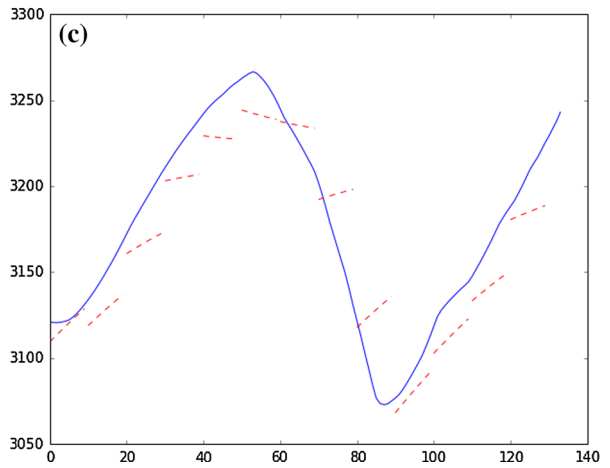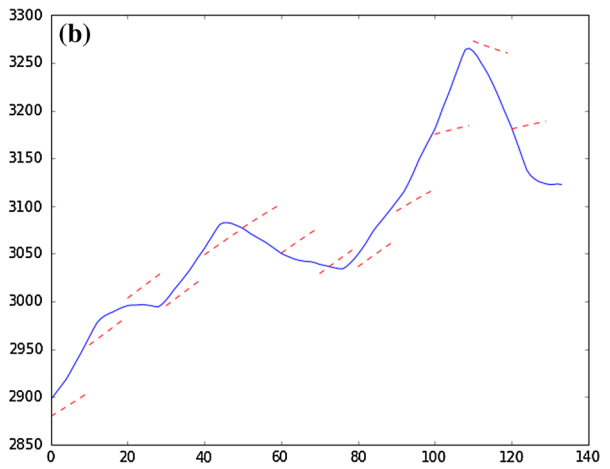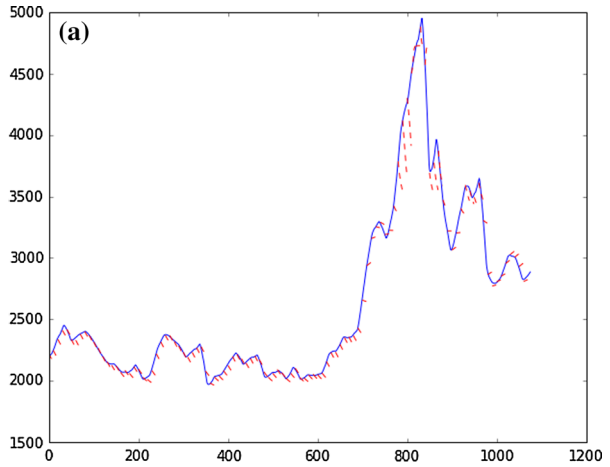
Table 3 Comparison of prediction effect

	Training set		Validation set		Test set	
	Theil	MAPE	Theil	MAE	Theil	MAE
LSTM	0.0022	0.0033	0.0015	0.0027	0.0016	0.0030
MLP	0.0020	0.0029	0.0023	0.0038	0.0024	0.0040
SVM	0.0057	0.0108	0.0048	0.0088	0.0058	0.0113
KNN	0.0050	0.0067	0.0177	0.0293	0.0265	0.0465

prediction effect on the long-term dynamic trend of financial time series is carried out. The LSTM neural network based on wavelet reconstruction data can better predict the long-term dynamic trend of financial time series data. Therefore, this paper only analyzes the dynamic prediction effect based on wavelet reconstruction data.

Figure 10 shows the dynamic prediction effect of the SHI for the next 10 days (shown by the dotted line in the figure). In general, the LSTM neural network can effectively predict the dynamic trend of the SHI for the next 10 days of the training set, validation set and test set. Therefore, LSTM neural network has a good prediction effect on long-term dynamic trend. This further confirms the applicability and validity of LSTM neural network to the actual financial time series data prediction.

In order to compare the prediction ability of LSTM with machine learning such as multi-layer perceptron (MLP), support vector machine (SVM), K-nearest neighbors (KNN), this paper uses machine learning to predict the Shanghai Composite Index. The prediction accuracy of the model is measured by using Theil unequal coefficient and mean absolute percentage error (MAPE). The prediction results are shown in Table 3.

where $Theil = \dfrac{\sqrt{1/T \sum_{t=1}^{T} (Y_t^{pre} - Y_t^{true})^2}}{\left(\sqrt{1/T \sum_{t=1}^{T} (Y_t^{pre})^2} + \sqrt{1/T \sum_{t=1}^{T} (Y_t^{true})^2} \right)}$, $MAPE = 1/T \sum_{t=1}^{T} \dfrac{|Y_t^{pre} - Y_t^{true}|}{Y_t^{true}}$. Also,

$Theil = 0$ represents that the predicted value and actual value are fully fitted, indicating the strongest prediction ability, $Theil = 1$ indicates the worst prediction ability.

Based on the predicted effect in Table 3, it can be seen that the LSTM neutral network has the best prediction ability on the test set, and can effectively balance the prediction effect of the training set, the validation set and the test set. MLP shows a relatively good prediction effect as compared with machine learning algorithms such as SVM, K-nearest neighbors, etc. SVM can effectively balance the prediction accuracy of the model in the training set, the validation set and the test set. However, the prediction effect of SVM is still far different from that of LSTM and MLP. The K-nearest neighbors show better prediction effect in the training set, but they have lower prediction accuracy in the validation set and test set. Therefore, there are some limitation for the use of shallow machine learning algorithm such as MLP, SVM and K-nearest neighbors to predict the financial time series data. This might be because the overfitting problem that exists in the shallow machine learning. The extraction ability is weak for the feature learning of the input feature vector, and the time-dependent feature of the financial time series data cannot be determined.

In a nutshell, LSTM overcomes the shortcomings of shallow machine learning algorithms such as the time series dependencies that cannot be described, weak data feature learning ability as well as prone to overfitting. Therefore, comparing to other models, LSTM shows a better prediction effect for the captures of the non-linear dynamic

characteristics of financial time series which is more comprehensive for its non-stationary characteristics and time-dependent characteristics.

4 Conclusions

This paper explores the application of both the theoretical basis and practical prediction of the LSTM neural network on the financial time series. It was proposed to decompose and reconstruct the financial time series data with wavelet analysis so as to eliminate the noise and to improve the ability of the model to predict the out-of-sample financial time series (i.e., generalization ability). Based on the empirical results of the Shanghai Composite Index, it is found that the LSTM neutral network shows a better prediction effect in which the prediction results are superior to the shallow machine learning models such as MLP, SVM, K-nearest neighbors, etc. This proves that the LSTM neutral network has a strong ability to predict the actual financial time series which is of great significance to establish the risk early warning mechanism for the securities market and provides the decision reference for the investors.

References

1. Taylor, G. W. (2009). *Composable, distributed-state models for high-dimensional time series*. Toronto: University of Toronto.
2. Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition*. arXiv:1409.1556.
3. Sarikaya, R., Hinton, G. E., & Deoras, A. (2014). Application of Deep Belief Networks for natural language understanding. *ACM Transactions on Audio Speech and Language Processing, 22*(4), 778–784.
4. Trafalis, T. B., & Ince, H. (2000). Support vector machine for regression and applications to financial forecasting. Neural Networks. In *Proceedings of the IEEE-INNS-ENNS international joint conference on IJCNN 2000* (pp. 348–353). IEEE.
5. Guresen, E., Kayakutlu, G., & Daim, T. U. (2011). Using artificial neural network models in stock market index prediction. *Expert Systems with Applications, 38,* 10389–10397.
6. Ahmed, N. K., Atiya, A. F., Gayar, N. E., et al. (2010). An empirical comparison of machine learning models for time series forecasting. *Econometric Reviews, 2010,* 594–621.
7. Bengio, Y., & LeCun, Y. (2007). Scaling learning algorithms towards AI. *Large-scale kernel machines, 34,* 1–41.
8. Heaton, J. B, Polson, N. G., & Witte, J. H. (2016). Deep learning in finance. arXiv preprint arXiv:1602.06561.
9. Hinton, G., Deng, L., Yu, D., et al. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine, 29,* 82–97.
10. Taylor, G. W., Fergus, R., Lecun, Y., & Bregler, C. (2010). *Convolutional learning of spatio-temporal features. European conference on computer vision* (Vol. 6316, pp. 140–153). Springer.
11. Hsieh, T. J., Hsiao, H. F., & Yeh, W. C. (2011). Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial bee colony algorithm. *Applied soft computing, 11,* 2510–2525.
12. Hochreiter, S., Bengio, Y., Frasconi, P., et al. (2001). Gradient flow in recurrent nets: The difficulty of learning long-term dependencies. In: J. F. Kolen, S. C. Kremer (Eds.), *A field guide to dynamical recurrent neural networks* (Vol. 28, pp. 237–244). IEEE Press.
13. Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems* (pp. 3104–3112).
14. Pang, Z. Y., & Liu, L. (2013). Can agricultural product price fluctuation be stabilized by future market: Empirical study based on discrete wavelet transform and GARCH model. *Journal of Finance Research, 11,* 126–139.

15. Längkvist, M., Karlsson, L., & Loutfi, A. (2014). A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern Recognition Letters, 42,* 11–24.
16. Tesauro, G. (1992). Practical issues in temporal difference learning. *Machine Learning, 8,* 257–277.
17. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. Cambridge: MIT Press.
18. Graves, A. (2012). *Supervised sequence labelling with recurrent neural networks* (pp. 5–13). Berlin: Springer.
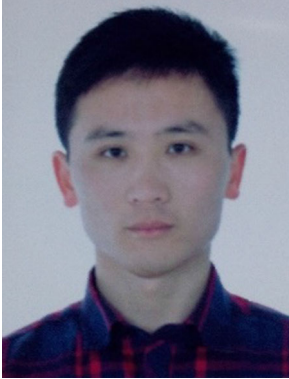
Hongju Yan is doctoral student of Huazhong University of Science and Technology. His main research direction is deep learning and financial time series analysis.

Hongbing Ouyang was born in 1968 and got the Ph.D. from the City University of Hong Kong in 2004. The main research direction is deep learning and financial economics.