Electronic
Research Archive

http://www.aimspress.com/journal/era

ERA, 30(10): 3855–3871.
DOI: 10.3934/era.2022196
Received: 20 June 2022
Revised: 14 August 2022
Accepted: 18 August 2022
Published: 25 August 2022

Research article

A prediction model for stock market based on the integration of independent component analysis and Multi-LSTM

Hongzeng He[1,2,*] and Shufen Dai[1]

[1] School of Economics and Management, University of Science and Technology Beijing, Beijing 100083, China
[2] Returned Overseas Talent and Expert Service Center, MOHRSS, Beijing 100083, China

* **Correspondence:** Email: hehongzeng@126.com.

Abstract: In this paper, we investigate the statistical behaviors of the stock market complex network. A hybrid model is proposed to predict the variations of five stock prices in the securities plate sub-network. This model integrates independent component analysis (ICA) and multivariate long short-term memory (Multi-LSTM) neural network to analyze the trading noise and improve the prediction accuracy of stock prices in the sub-network. Firstly, we apply ICA to deconstruct the original dataset and remove the independent components that represent the trading noise. Secondly, the rest of the independent components are given to Multi-LSTM neural network. Finally, prediction results are reconstructed from the outputs of the Multi-LSTM neural network and the corresponding mixing matrix. The experiment results indicate that the hybrid model outperforms the benchmark approaches, especially in terms of the stock market complex network.

Keywords: independent component analysis; Multi-LSTM; stock market; complex network; prediction model

1. Introduction

It is believed that the stock market could be described by complex networks. The applications of complex networks have provided a new perspective for studying the in-depth mechanisms of the stock market [1]. In the fields of statistics and finance, stock market is regarded as a complex nonlinear dynamic system with numerous variables. The market is affected by economy, politics and society. Consequently, variations in the stock market are difficult to predict accurately. Due to the complexity of the stock market, some prediction models have been proposed to study and analyze the fluctuations of the market prices, mainly through historical data [2–7]. In recent studies, the prediction approaches of stock prices can mainly be divided into statistical models, econometric models, neural network models and so on.

Since the financial market data is dynamic, nonlinear, nonstationary, nonparametric and volatile, various techniques are combined with artificial prediction approaches to model the financial time series. For instance, the independent component analysis (ICA) method is applied to reduce the trading noise of the financial data [8,9]. Haifan Liu et al. [9] integrated the ICA method with the neural network model to investigate the statistical behaviors of Shanghai Composite Index fluctuations. Results proved that the proposed model outperforms the conventional single BP model. Kao et al. [10] combined nonlinear ICA with support vector regression (SVR) for stock price forecasting to improve the prediction accuracy of the traditional methods. Jianwei E et al. [11] proposed a variational hybrid method of mode decomposition (VMD), ICA and autoregressive integrated moving average (ARIMA) to analyze the influence factors of crude oil price and predict future prices. Jianwei E et al. [12] offered a novel combination technique based on ICA and gate recurrent unit neural network (GRUNN) to predict the gold price. The combination method achieved higher accuracy than the benchmark methods. Then Jianwei E et al. [8] presented a hybrid model based on extreme-point symmetric mode decomposition (ESMD), kernel independent component analysis (KICA) and least squares support vector regression (LSSVR) to predict the carbon prices from European Union Emissions Trade System (EU ETS).

Recently, neural network models have played a significant role in time series prediction fields. Especially the long short-term memory (LSTM) can memory the information for a longer period to overcome the vanishing gradient problem. It could be summarized that LSTM models are more effective in recognizing trends of the time series than the traditional neural network models [13]. Since the financial time series is nonstationary, nonlinear and with trading noise. Some hybrid prediction models are based on neural networks with historical data, such as ICA, wavelet transform (WT), autoencoder (SAE), complete ensemble empirical mode decomposition (CEEMD), ensemble empirical mode decomposition (EEMD), empirical mode decomposition (EMD), principal component analysis (PCA) and so on. These approaches are combined with LSTM to smooth the original data and further enhance the prediction accuracy. Wq A et al. [14] proposed a combination of SAE and LSTM on gas production and consumption data to improve the prediction performance and solve the gradient disappearance problem of LSTM. Zhang Y et al. [15] applied CEEMDAN and PCA to eliminate redundant information and improve the prediction response speed. Bao W et al. decomposed the stock price series by WT and fed the high-level denoising features into LSTM to forecast the next day’s closing price [16]. Akhil Sethia et al. [17] applied LSTM, GRU, ANN and SVM to predict S&P500 index close prices. The ICA is implemented as the data processing to reduce the input attribute dimension from 50 to 12. These 12 attributes are given to the prediction model to optimize the prediction strategy. And these models are among the most popular approaches because of the improvements in accuracy and efficiency.

In all the surveyed studies, ICA was usually combined with predicting methods, such as gate recurrent unit neural network (GRUNN), neural networks (NN), autoregressive integrated moving average (ARIMA), support vector regression (SVR), least squares support vector regression (LSSVR), single back propagation (BP) neural network or implemented to preprocess original attributes. Nevertheless, most of the previous studies focus on the prediction of a single time series without dataset reconstruction. To our knowledge, few studies employ methodologies that take into account the relevance between multiple stocks in a stock market complex network. To fill this gap, we proposed a prediction model for the stock market based on the integration of ICA and Multi-LSTM, including a dataset reconstruction process, to expand the research perspectives on the stock market network and enhance the prediction accuracy.

The innovations of this study are as follows: Firstly, several independent components are decomposed from the same securities plate sub-network dataset by ICA method. Trading noise is recognized and reduced through dimension reduction process. Secondly, remaining independent components are conducted as the input of the Multi-LSTM model. Final predicting results are reconstructed from predicting results of the independent components. Then, experiment results indicate that ICA is suitable for stock market complex network dataset, which could match the input type for Multi-LSTM with less calculation amount. And the proposed ICA-Multi-LSTM model produced better performance than the benchmark approaches.

The organization of this paper is as follows. Section 2 describes the dataset and outlines the proposed model for predicting the stock price in a stock market complex network. The empirical results and comparisons of performance are listed in Section 3. Section 4 concludes the paper. Finally, acknowledgments and references are also presented.

2. Materials and methods

2.1. Stock market dataset

In this study, we choose the securities plate sub-network of the stock market complex networks in reference [1]. The dataset includes five attributes: daily opening prices, daily closing prices, daily high prices, daily low prices and daily trading volumes of 5 securities companies. The invalid data and empty data are dropped. The dataset is from Jul. 26th, 2015 to Dec. 31st, 2020 (Data sources: Resset.cn) which contains 32,675 data points of 1307 trading days. The samples of the dataset are presented in Table 1.

Table 1. Trading data samples of 5 stocks on Sept. 7th, 2020.

Stock code	Company name	Opening prices	High prices	Low prices	Closing prices	Trading volumes
000166	SWHY	5.53	5.58	5.4	5.43	111,350,918
002736	GUOSEN	13.74	13.93	13.46	13.48	22,030,539
601198	DFZQ	11.61	11.7	11.27	11.27	34,864,661
600958	DXZQ	13.2	13.34	12.85	12.88	17,851,633
601211	GTJA	18.75	19.1	18.48	18.52	39,169,235

The network structure of the securities plate sub-network is shown in Figure 1. As shown in Figure 1, stocks are connected according to the correlation coefficient of the stock price series [1]. The correlation relationships of the stocks could be calculated by correlation analysis or clustering methods, such as Pearson correlations, detrended cross-correlation analysis (DCCA) [1], K-means clustering [18] and feature-based clustering [19]. The correlation results can express the historical correlation and trend correlation of stock prices. It is believed that multiple stock price data of the same cluster instead of a single self-data, as input data to train LSTM, helps to improve prediction accuracy [18–20]. The overview of the dataset time series is shown in Figure 2. The graph shows that there exists a relevant tendency between the time series of five stocks, which means high correlations of the stocks in the same complex network. In this study, we select the daily closing prices, daily high prices, daily low prices and daily trading volumes as input of the prediction model to predict the daily opening prices and verify the predicting results.

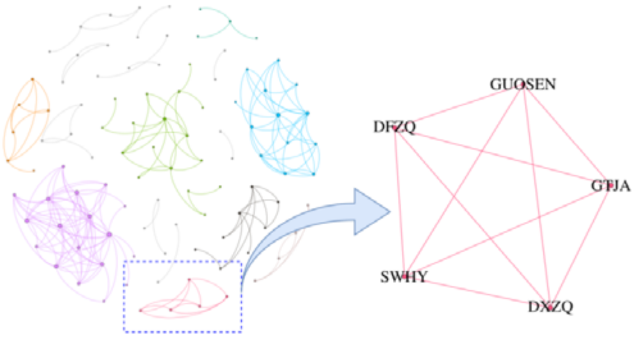

Figure 1. The network structure of the securities plate sub-network [1].

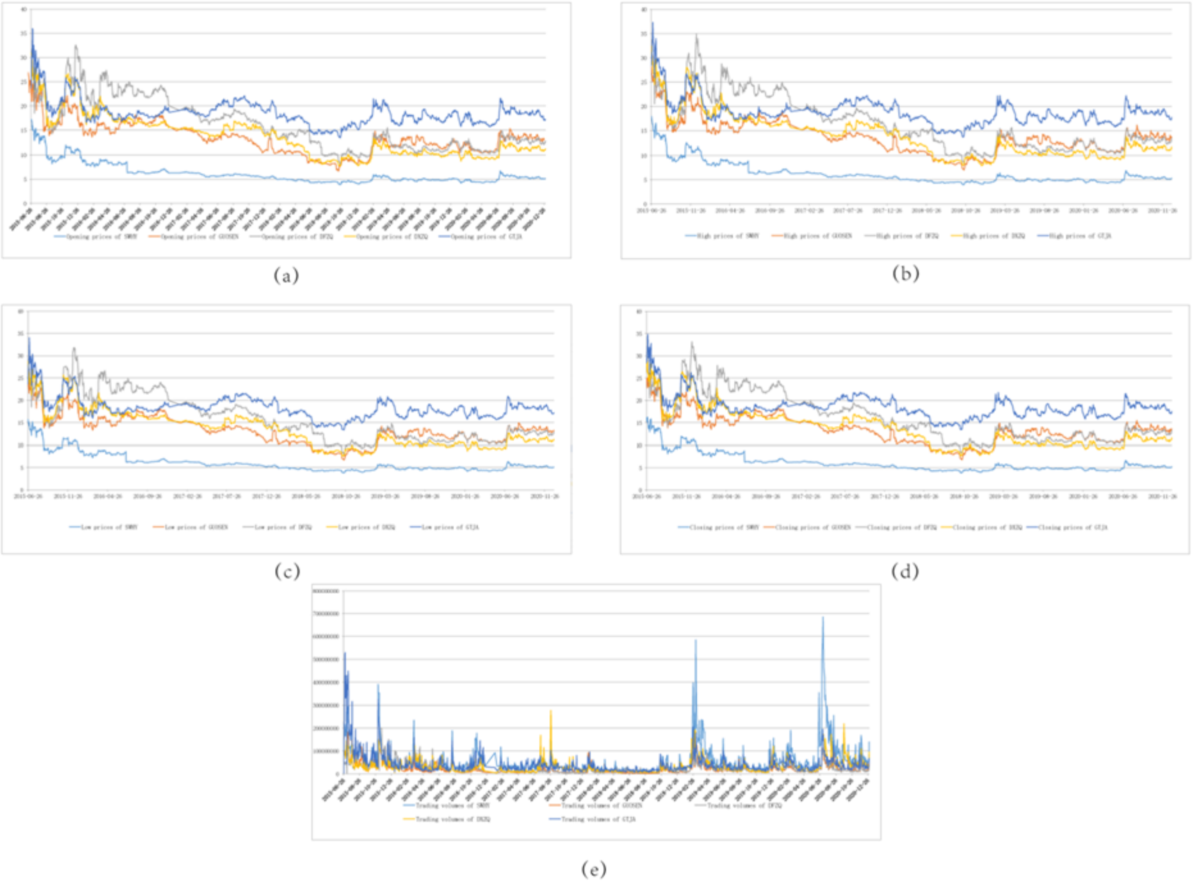

Figure 2. The opening prices (a), closing prices (b), high prices (c), low prices (d) and trading volumes (e) of 5 securities stocks.

In this study, we obtain a prediction model for the stock market based on the integration of ICA and Multi-LSTM. For this purpose, we introduce a brief theoretical description of the two methods.

2.2. Independent component analysis

The ICA method is a novel feature extraction technique, which is used to find the hidden influence factors from the mixed source data [10,17]. Since the trading data of the stock market is nonlinear, nonstationary and mixed with noise pollution, the ICA method can separate the independent components of mixed time series data. This technology could be employed to solve the blind source separation problem, time series denoising and so on. The ICA method is usually applied to the field of stock market data denoising. Several independent components (ICs) are separated from the original data, which could be analyzed, selected and reconstructed to the denoising data. Generally, the ICA algorithm consists of the following two steps.

Step 1: Supposing that the $m \times n$ trading data time series matrix $\boldsymbol{X} = [\boldsymbol{x}_1,\ \boldsymbol{x}_2, \dots, \boldsymbol{x}_i, \dots, \boldsymbol{x}_m]^T$, where $\boldsymbol{x}_i$ represents the $\boldsymbol{ith}$ trading data with the length of n and $i = 1,\dots,m$. Based on the ICA algorithm, the matrix $\boldsymbol{X}$ could be represented by

$$\boldsymbol{X} = \boldsymbol{AS} = \sum_{i=1}^{m} \boldsymbol{a}_i \boldsymbol{s}_i \tag{1}$$

where $\boldsymbol{A}_{m\times m} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \dots, \boldsymbol{a}_i, \dots \boldsymbol{a}_m]^T$ represents the mixing matrix which includes noise trading data and $\boldsymbol{S}_{m\times n} = [\boldsymbol{s}_1, \boldsymbol{s}_2, \dots, \boldsymbol{s}_i, \dots \boldsymbol{s}_m]^T$ represents the separated trading data matrix. The purpose of the ICA algorithm is to find a separating matrix $\boldsymbol{W}$ which could be represented by

$$\boldsymbol{Y} = [\boldsymbol{y}_i] = \boldsymbol{WX} \tag{2}$$

where $\boldsymbol{y}_i$ is the $\boldsymbol{ith}$ independent component. The independent components must be as independent with each other as possible. If we let the separating matrix $\boldsymbol{W} = \boldsymbol{A}^{-\boldsymbol{1}}$, and the real trading data matrix $\boldsymbol{S}$ could be reconstructed by

$$\boldsymbol{Y} = \boldsymbol{WX} = \boldsymbol{W} \cdot \boldsymbol{A} \cdot \boldsymbol{S} = \boldsymbol{A}^{-1}\boldsymbol{AS} = \boldsymbol{S} \tag{3}$$

All the $\boldsymbol{y}_i (i = 1,\dots,m)$ are considered independent if they are non-Gaussian distribution. The gaussianity of $\boldsymbol{y}_i$ could be given by

$$J(\boldsymbol{y}) = H(\boldsymbol{y}_{gauss}) - H(\boldsymbol{y}) \tag{4}$$

where $\boldsymbol{y}_{gauss}$ is a Gaussian random vector with the same covariance matrix as $\boldsymbol{y}$ and $H(\boldsymbol{y}) = -\int p(\boldsymbol{y}) logp(\boldsymbol{y}) d\boldsymbol{y}$. Some fast algorithms are applied in engineering on account of the huge amount of calculations [21].

Step 2: Since we have got the independent components from the original data, independent components could be evaluated and analyzed. We could remove the independent components that represent the trading noise and reconstruct the real trading data matrix by

$$\boldsymbol{X}^R = \sum_{1\le i\le m}^{i\in S} \boldsymbol{a}_i \boldsymbol{y}_i, \quad 1 \le i \le m \tag{5}$$

where $\boldsymbol{X}^R = [\boldsymbol{x}_{\boldsymbol{1}}^{\boldsymbol{R}}, \boldsymbol{x}_{\boldsymbol{2}}^{\boldsymbol{R}}, \dots, \boldsymbol{x}_{\boldsymbol{m}}^{\boldsymbol{R}}]^T$ represents the reconstructed real trading data matrix and the mixing matrix $\boldsymbol{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \dots, \boldsymbol{a}_m] = \boldsymbol{W}^{-\boldsymbol{1}}$ and $\boldsymbol{y}_i$ is the $\boldsymbol{ith}$ independent component with less trading noise.

2.3. Multi-LSTM neural network

The long short-term memory (LSTM) neural network is an improved version of the recurrent neural network (RNN). The method is designed to solve the long-term dependency problem and avoid the vanishing gradient problem or the radiant exploding problem of the traditional RNN [14]. LSTM network can predict time series from multivariate input data with higher accuracy. The structure of the Multi-LSTM network is comprised of several LSTM cells. There are input gate (***i***), output gate (***o***), forget gate (***f***) and cell state (***C***), as shown in Figure 3.

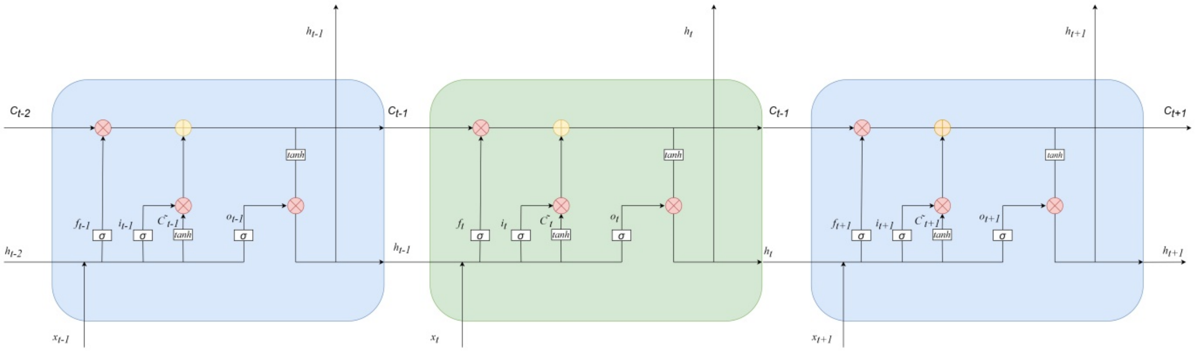

Figure 3. Structures of the Multi-LSTM neural network.

As is presented in Figure 3, $\boldsymbol{x}_{t-1}$, $\boldsymbol{x}_t$, $\boldsymbol{x}_{t+1}$ are the input data at time *t-1, t, t+1*. $\boldsymbol{h}_{t-1}$, $\boldsymbol{h}_t$, $\boldsymbol{h}_{t+1}$ are the output data at time *t-1, t, t+1*. There are input gate ***i***, output gate ***o***, forget gate ***f*** and cell state ***C*** in each cell. The *σ* and *tanh* are the activation functions. ***W*** is the weight coefficient and ***b*** is the deviation matrix. The cell state $\boldsymbol{C}_t$ at time *t* could be adjusted by the forget gate $\boldsymbol{f}_t$ and input gate $\boldsymbol{i}_t$. The forget gate $\boldsymbol{f}_t$ could remember or forget the cell state $\boldsymbol{C}_{t-1}$. The input gate $\boldsymbol{i}_t$ could allow or forbid updating of the cell state. The cell state $\boldsymbol{C}_t$ could be transmitted to the next cell. The structure of the LSTM establishes advantages to predict the long dependent and nonlinear time series.

$$\begin{aligned}
\boldsymbol{f}_t &= \sigma\left(\boldsymbol{W}_f \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_f\right) \\
\boldsymbol{i}_t &= \sigma(\boldsymbol{W}_i \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_i) \\
\widetilde{\boldsymbol{C}}_t &= tanh\,(\boldsymbol{W}_C \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_C) \\
\boldsymbol{C}_t &= \boldsymbol{f}_t \boldsymbol{C}_{t-1} + \boldsymbol{i}_t \widetilde{\boldsymbol{C}}_t \\
\boldsymbol{o}_t &= \sigma(\boldsymbol{W}_o \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_o) \\
\boldsymbol{h}_t &= \boldsymbol{o}_t tanh\,(\boldsymbol{C}_t)
\end{aligned} \tag{6}$$

2.4. ICA-Multi-LSTM prediction model

In this subsection, we describe the details of the proposed prediction model. The daily opening price could be predicted by multivariate factors by this model, such as daily closing prices, daily high prices, daily low prices and daily trading volumes. The hybrid model consists of the following four steps:

Step 1: data preprocessing

We get the historical attributes including daily closing prices, daily high prices, daily low prices and daily trading volumes as the input data to predict daily opening prices. The daily opening price matrix $\boldsymbol{A}_{Oppr}$, daily high price matrix $\boldsymbol{A}_{Hipr}$, daily low price matrix $\boldsymbol{A}_{Lopr}$, daily closing price matrix $\boldsymbol{A}_{Clpr}$ and daily trading volume matrix $\boldsymbol{A}_{Trdvol}$ are extracted from the historical trading data by certain data pre-processing techniques such as data purifying and data smoothing. Each row of the matrix represents the stocks in the securities plate sub-network and each column represents the time series data of each stock. The brief example is as follows:

$$\boldsymbol{A}_{Oppr} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \dots, \boldsymbol{a}_i, \dots, \boldsymbol{a}_m]^T \tag{7}$$

As shown in Eq (7), $\boldsymbol{A}_{Oppr}$ is the $m \times n$ daily opening price matrix, where m is the number of stocks in the securities plate sub-network and $\boldsymbol{a}_i$ is the daily opening price vector of the stock i, where $i = 1,\dots,m$. The length of $\boldsymbol{a}_i$ is the total trading days n.

Step 2: ICA and dimension reduction

In this step, we begin with the independent component analysis to separate the ICs from $\boldsymbol{A}_{Oppr}$, $\boldsymbol{A}_{Hipr}$, $\boldsymbol{A}_{Lopr}$, $\boldsymbol{A}_{Clpr}$, $\boldsymbol{A}_{Trdvol}$ and calculate the contribution rate of each IC, followed by a dimension reduction process to get rid of the ICs with low contribution rates that represent the trading noise. Finally, the IC matrices with less trading noise are considered as the input of the Multi-LSTM model. The IC matrices are represented as daily opening price IC matrix $\boldsymbol{C}_{Oppr}$, daily high price IC matrix $\boldsymbol{C}_{Hipr}$, daily low price IC matrix $\boldsymbol{C}_{Lopr}$, daily closing price IC matrix $\mathbf{C}_{Clpr}$, daily trading volume IC matrix $\boldsymbol{C}_{Trdvol}$, separating matrix $\boldsymbol{W}$ and mixing matrix $\boldsymbol{A}$. A brief example of the IC matrix $\boldsymbol{C}_{Oppr}$ is as follows:

$$\boldsymbol{C}_{Oppr} = [\boldsymbol{c}_1, \boldsymbol{c}_2, \dots, \boldsymbol{c}_i, \dots, \boldsymbol{c}_l]^T \tag{8}$$

where $\boldsymbol{C}_{Oppr}$ is the $l \times n$ IC matrix of the daily opening price, where l is the number of independent components where the noise is eliminated. And $\boldsymbol{c}_i$ is the ***ith*** IC vector, where $i = 1,\dots,l$. The length of $\boldsymbol{c}_i$ is the total trading days n.

Step 3: training and predicting independent components with Multi-LSTM

The dataset $\boldsymbol{C}_{Hipr}$, $\boldsymbol{C}_{Lopr}$, $\boldsymbol{C}_{Clpr}$, $\boldsymbol{C}_{Trdvol}$, $\boldsymbol{C}_{Oppr}$ are divided into the training set, validation set and test set. And $\boldsymbol{C}_{Hipr}$, $\boldsymbol{C}_{Lopr}$, $\boldsymbol{C}_{Clpr}$, $\boldsymbol{C}_{Trdvol}$ are given to Multi-LSTM to train and predict the $\boldsymbol{C}_{Oppr}$. So the training and predicting results $\boldsymbol{C}_{Oppr_{pdt}}$ could be obtained from the output of Multi-LSTM prediction results:

$$\boldsymbol{C}_{Oppr_{pdt}} = \left[\boldsymbol{c}_{1\_pdt}\ , \boldsymbol{c}_{2\_pdt}\ , \dots, \boldsymbol{c}_{i\_pdt}\ , \dots, \boldsymbol{c}_{l\_pdt}\ \right]^T \tag{9}$$

where $\boldsymbol{C}_{i\_pdt}$ represents the output results of the ***ith*** IC vector. The $\boldsymbol{C}_{i\_pdt}$ spliced by training results validation results and predicting results respectively.

Step 4: final predictions reconstruction

The final prediction results $\boldsymbol{A}_{Oppr\_pdt}$ could be reconstructed by $\boldsymbol{C}_{Oppr\_pdt}$ and mixing matrix $\boldsymbol{A}$ by

$$\boldsymbol{A}_{Oppr\_pdt} = \boldsymbol{A} \cdot \boldsymbol{C}_{Oppr\_pdt} \tag{10}$$

where $\boldsymbol{A}_{Oppr\_pdt} = [\boldsymbol{a}_{Oppr\_pdt\_1}\ , \boldsymbol{a}_{Oppr\_pdt\_2}\ , \dots, \boldsymbol{a}_{Oppr\_pdt\_i}\ , \dots, \boldsymbol{a}_{Oppr\_pdt\_m}\ ]^T$ and $\boldsymbol{a}_{Oppr\_pdt\_i}$ are the final prediction results of the daily opening price of stock i. Hence, the framework of the proposed model is reflected in Figure 4.

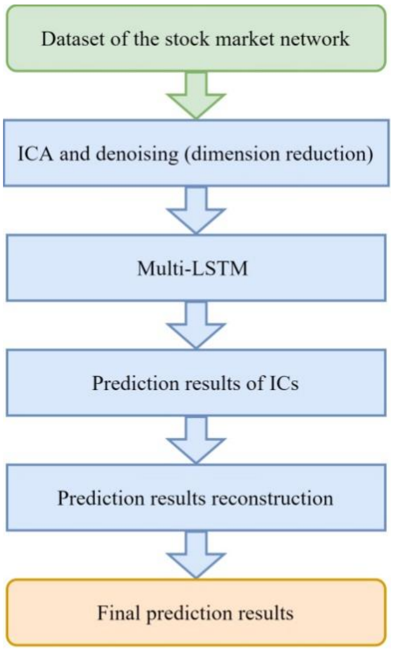
Dataset of the stock market network
ICA and denoising (dimension reduction)
Multi-LSTM
Prediction results of ICs
Prediction results reconstruction
Final prediction results

Figure 4. The framework of the proposed model.

3. Results

3.1. Independent component analysis and denoising effect

The first stage of the proposed model is independent component separation, in which the original trading data are separated into independent components via the ICA method. Further, the trading noise could be eliminated via the methods mentioned in Section 2.

In this subsection, we get five independent component matrices of the original data $\boldsymbol{A}_{Oppr}$, $\boldsymbol{A}_{Hipr}$, $\boldsymbol{A}_{Lopr}$, $\boldsymbol{A}_{Clpr}$, $\boldsymbol{A}_{Trdvol}$ and calculate the contribution rate of each IC based on eigenvalues of the IC matrices by the method in [22,23]. The eigenvalues and contribution rates of the IC matrices are listed in Tables 2 and 3.

Table 2. Eigenvalues of the IC matrices.

IC matrix	Eigenvalue of IC 1	Eigenvalue of IC 2	Eigenvalue of IC 3	Eigenvalue of IC 4	Eigenvalue of IC 5
Opening prices (Oppr)	59.011864893367957	5.0628123317108947	1.470329972507548	0.975816442916122	0.314082455856973
High prices (Hipr)	63.907984112057726	5.563327136031251	1.558448860320623	0.947959409670055	0.342468031744766
Low prices (Lopr)	54.885599499233635	4.656578315170588	1.410614358896386	0.936401503338652	0.288942244294947
Closing prices (Clpr)	59.046651761582176	5.124892353324127	1.48402084266901	0.931158457564463	0.319906218596268
Trading volumes (Trdvol)	6766662391794044	1092710704368497	436895019922287.9	262959252833420.4	113538274772889.1

As is illustrated in Tables 2 and 3, the cumulative contribution rates of the first three independent components are 98.07% in the daily opening price IC matrix, 98.22% in the daily high price IC matrix, 98.03% in the daily low price IC matrix, 98.13% in the daily closing price IC matrix, 95.66% in daily trading volume IC matrix. It is now understood that the contribution rate is used to select the independent components representing the real trading information and trading noise [8,9]. In this study, we find that the IC 4 and IC 5 with the lowest contribution rates denote the noise trading data. We guarantee the first three independent components that contain the real trading information are IC 1, IC 2 and IC 3. These ICs are used as the input of the Multi-LSTM. The independent components with less trading noise are in Figure 5. As shown in Figure 5, the fluctuation patterns of the independent components may contain the trading information of periodicity, seasonality or random events. These independent components are selected as the input for the Multi-LSTM model.

Table 3. Contribution rates of the IC matrices.

IC matrix	Contribution rate of IC 1	Contribution rate of IC 2	Contribution rate of IC 3	Contribution rate of IC 4	Contribution rate of IC 5
Opening prices (Oppr)	88.29%	7.58%	2.20%	1.46%	0.47%
High prices (Hipr)	88.37%	7.69%	2.15%	1.31%	0.47%
Low prices (Lopr)	88.27%	7.49%	2.27%	1.51%	0.46%
Closing prices (Clpr)	88.25%	7.66%	2.22%	1.39%	0.48%
Trading volumes (Trdvol)	78.02%	12.60%	5.04%	3.03%	1.31%

3.2. Prediction results of the independent components

The second stage of the proposed model is the prediction module. The independent components with less trading noise are given to the Multi-LSTM model to predict the daily opening price in this stage. To investigate the effectiveness of the proposed model, the dataset structure and the Multi-LSTM network architecture are also illustrated in this subsection.

Dataset structure All the data sets are constructed from the independent components with less trading noise. In this study, there are 1–1052 trading days in the training set, 1053–1300 trading days in the validation set and 1301–1307 trading days in the test set. This study adopts the rolling prediction approach. In other words, we use the previous 60 days to predict the following one day iteratively. For example, the model uses the 1–60 days of daily high price, daily low price, daily closing price, and daily trading volume IC matrices as the input attributes to train and predict the 61st day of the daily opening price, 2–61 days to train and predict the 62nd day and so on. As a result, we could get training predictions of 61–1052 days, validation predictions of 1053–1300 days and test predictions of 1031–1037 days respectively.

Multi-LSTM neural network architecture The architecture of the Multi-LSTM neural network in this article is illustrated in Figure 6. As is shown in Figure 6, the network contains five layers, including the input layer, two hidden layers, the dropout layer and the output layer. We input the IC matrices $\boldsymbol{C}_{Hipr}$, $\boldsymbol{C}_{Lopr}$, $\boldsymbol{C}_{Clpr}$, $\boldsymbol{C}_{Trdvol}$ into the input layer to train and predict $\boldsymbol{C}_{Oppr}$. There are two hidden layers in the model according to the computing speed and prediction accuracy. It is practically proved that the model could achieve high precision and good effect if the numbers of the two hidden layers are very close. And the numbers of neural nodes in the two hidden layers are 64 and 10. The dropout layer is set to optimize the neural network and the deactivation rate is 0.25. The optimization algorithm is Adam and the dynamic learning rate is 0.01. To train the Multi-LSTM model, the parameters converge to 30 epochs with Adam Optimizer. The batch size is 1 according to the data size and the architecture of the Multi-LSTM.

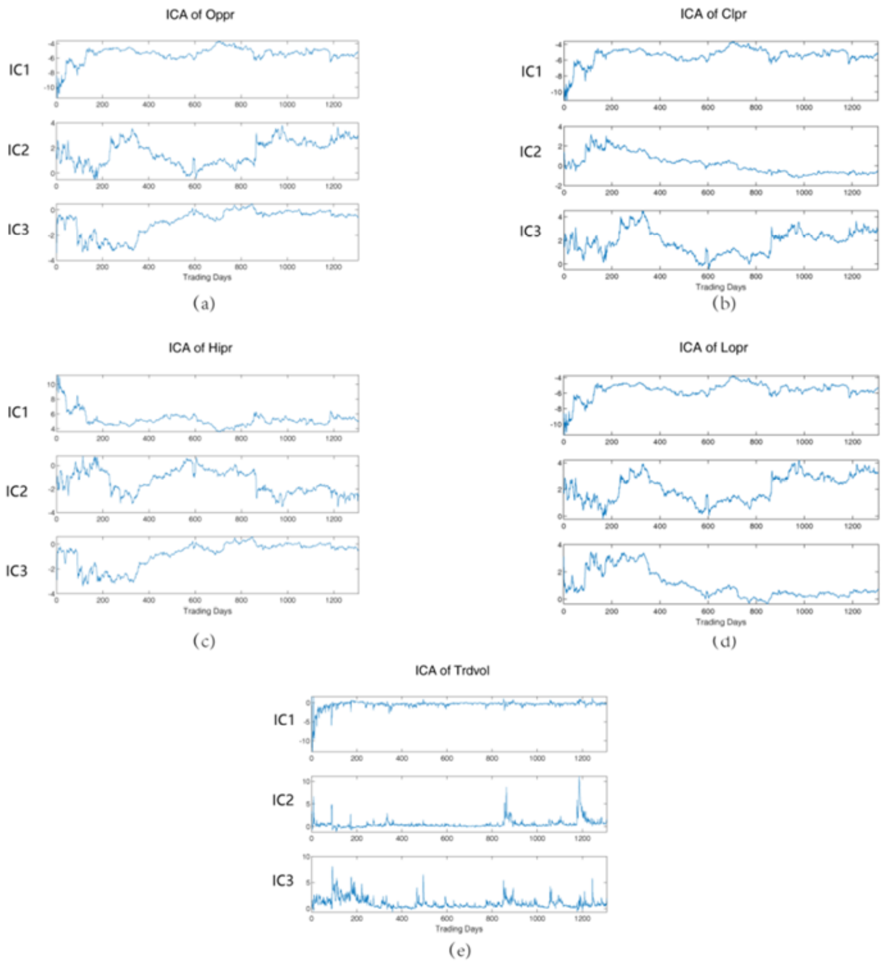

Figure 5. Independent components with less trading noise: opening prices (a), closing prices (b), high prices (c), low prices (d), and trading volumes (e) of the five securities stocks.

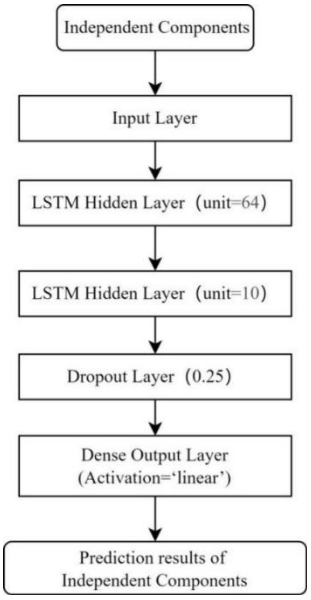

Figure 6. The architecture of the Multi-LSTM neural network.

Figure 7 shows the predicting results of the opening price independent components with less trading noise. The predicting results of the three stock daily opening price independent components contain the train predictions and future predictions. From Figure 7, we consider that the predicting results have the same trends as the actual daily opening price.

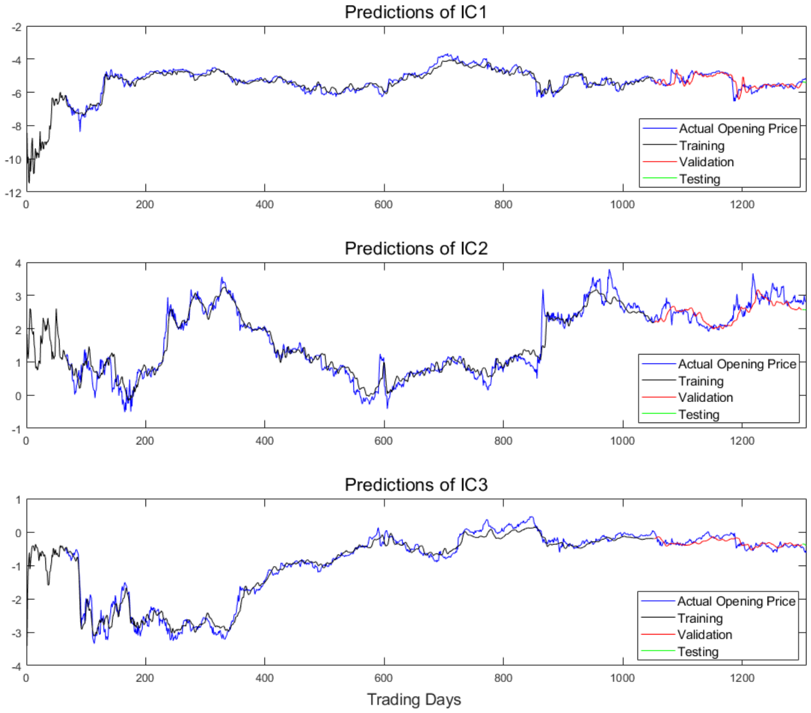

Figure 7. Prediction results of the opening price IC 1, IC 2 and IC 3.

3.3. Final prediction results

In this stage, we reconstruct the final predicting results $\boldsymbol{A}_{Oppr\_pdt}$ by the mixing matrix $\boldsymbol{A}$ and $\boldsymbol{C}_{Oppr\_pdt}$.

$$\boldsymbol{A}_{Oppr\_pdt} = \boldsymbol{A} \cdot \boldsymbol{C}_{Oppr\_pdt} \tag{11}$$

The $\boldsymbol{C}_{Oppr\_pdt}$ is consisted of 1–60 days of original data, 61–1052 days of training prediction, 1053–1300 days of validation prediction and 1301–1307 days of testing prediction to fit the length of the original data and mixing matrices $\boldsymbol{A}$ in Eq (11). It could be noted that there exists subtle error because the mixing matrix $\boldsymbol{A}$ is an estimated value by the fast ICA algorithm [21].

3.4. Performance criteria

The prediction performance could be evaluated by the following performance measures: Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Mean Square Error (MSE). The corresponding definitions of N trading days are given as follows:

- Mean Absolute Error (MAE):

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N} |y_i' - y_i| \tag{12}$$

- Root Mean Square Error (RMSE):

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (y_i' - y_i)^2} \tag{13}$$

- Mean Square Error (MSE):

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N} (y_i' - y_i)^2 \tag{14}$$

where $\boldsymbol{y}_i'$ is the final prediction result in the trading day *i*. The $\boldsymbol{y}_{\mathrm{i}}$ is the original data in the trading day *i*, and *i* = *1,…, N*. The *N* is the number of trading days.

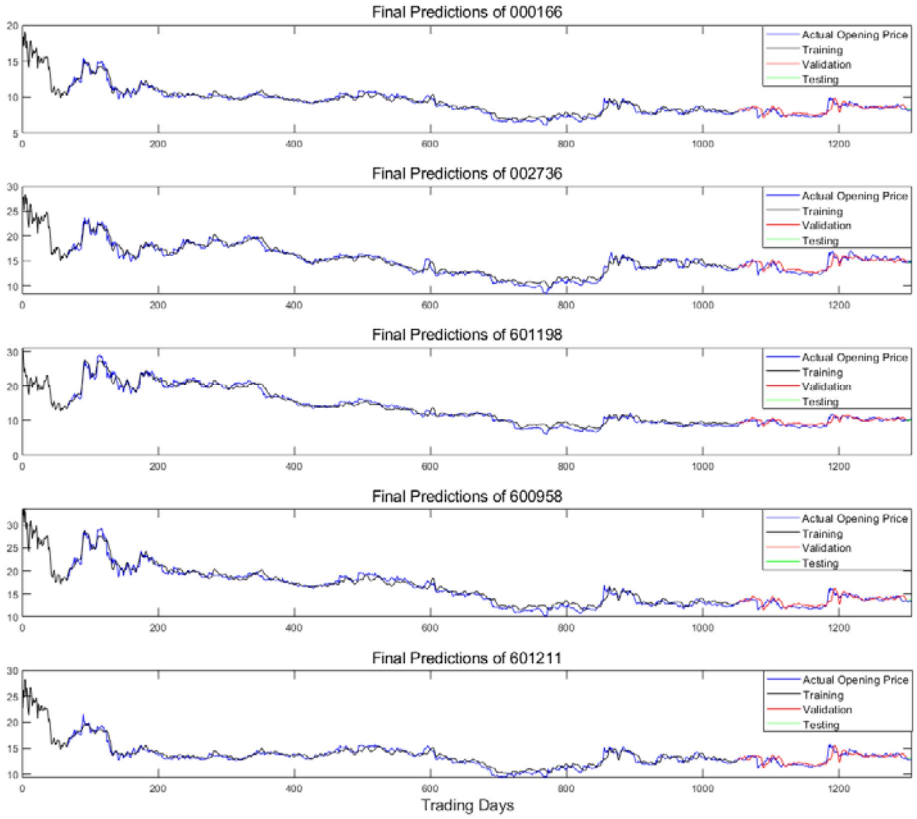

Figure 8. Final prediction results reconstructed by $\boldsymbol{A}$ and $\boldsymbol{C}_{Oppr\_pdt}$.

The performance criteria are in three parts: training performance criteria from 61 to 1052 trading days, validation performance criteria from 1053 to 1300 trading days and test performance criteria from 1301 to 1307 trading days. A lower value of the performance criteria means lower prediction error as well as higher accuracy of the prediction model. In Table 4, we can see the fact that the prediction accuracy of the ICA-Multi-LSTM model outperforms the traditional methods in most cases with lower performance criteria, which benefits from the ICA nonlinear and nonstationary data denoising preprocess strategy.

Table 4. Performance criteria on single stock.

Models	Stocks	MAE (Training)	RMSE (Training)	MSE (Training)	MAE (Validation)	RMSE (Validation)	MSE (Validation)	MAE (Testing)	RMSE (Testing)	MSE (Testing)
ICA-Multi-LSTM	000166	0.2895	0.382	0.1459	0.3183	0.4428	0.1961	0.1968	0.2046	0.0419
	002736	**0.4933**	**0.6615**	**0.4376**	**0.5792**	0.7911	0.6258	**0.155**	**0.2083**	**0.0434**
	601198	**0.6515**	**0.8503**	**0.7229**	**0.5192**	**0.6576**	**0.4325**	**0.2105**	**0.2661**	**0.0708**
	600958	**0.5691**	**0.7497**	**0.5621**	**0.5603**	**0.7597**	**0.5771**	**0.3206**	**0.3398**	**0.1155**
	601211	**0.4011**	**0.531**	**0.282**	**0.5603**	**0.6882**	**0.4736**	**0.3584**	**0.3676**	**0.1351**
Multi-LSTM	000166	**0.1921**	**0.3363**	**0.1131**	**0.2004**	**0.3134**	**0.0982**	**0.0339**	**0.051**	**0.0026**
	002736	0.5433	0.7804	0.6091	0.6068	**0.7424**	**0.5512**	0.2574	0.2993	0.0896
	601198	0.7184	1.249	1.5599	0.5426	0.7644	0.5844	0.7385	0.7567	0.5726
	600958	0.6561	0.9703	0.9415	0.7719	0.9881	0.9763	0.5324	0.5622	0.3161
	601211	0.4813	0.7005	0.4908	0.7193	0.9749	0.9504	0.5655	0.5949	0.3539

However, since the time series are complicated and changeable. The specific predicting results depend on the structural properties of the time sequence in some cases. In order to compare the overall performance of the proposed model on the securities plate stock market complex network, we introduce M-MAE, M-RMSE and M-MSE which could evaluate the average predicting error. These criteria could indicate the aggregate performance on the stock network. The corresponding definitions are given as follows:

- M-MAE：

$$\text{M-MAE}=\frac{1}{M\times N}\sum_{j=1}^{M}\sum_{i=1}^{N}\left|y'_{ji}-y_{ji}\right| \tag{15}$$

- M-RMSE：

$$\text{M-RMSE}=\sqrt{\frac{1}{M\times N}\sum_{j=1}^{M}\sum_{i=1}^{N}\left(y'_{ji}-y_{ji}\right)^2} \tag{16}$$

- M-MSE：

$$\text{M-MSE}=\frac{1}{M\times N}\sum_{j=1}^{M}\sum_{i=1}^{N}\left(y'_{ji}-y_{ji}\right)^2 \tag{17}$$

where $\boldsymbol{y}'_{ji}$ is the final prediction result of stock j in the trading day i, $\boldsymbol{y}_{ji}$ is the original data of stock j in the trading day i, and $i = 1,\ldots, N; j = 1,\ldots, M$. The N is the number of trading days and the M is the total number of stocks in the securities plate sub-network.

The final comparative results are listed in Table 5. From Table 5, the following facts could be yielded. From the perspective of complex networks, the ICA-Multi-LSTM method is superior to the Multi-LSTM method, which shows the advantage of the integration of ICA and Multi-LSTM. The primary reason is that trading noise is reduced via the ICA dimension reduction process which verifies the improvement of the prediction accuracy.

To better capture the performance of ICA-Multi-LSTM on the stock market network, we comprehensively introduce the other eight methods and evaluate their predictive criteria in comparison experiments. The former five models are used to check the effectiveness of ICA and other variants. The last three models are used as the performance benchmark. In the CEEMD-Multi -LSTM, EEMD-Multi-LSTM, EMD-Multi-LSTM and WT-Multi-LSTM, every single time series of daily closing prices, daily high prices, daily low prices and daily trading volumes are decomposed by CEEMD [24], EEMD [25], EMD [26] and WT [16] to reduce the trading noise. And the denoised time series are reconstructed from the first 5 intrinsic mode functions (IMFs) according to the cumulative contribution rates to predict the actual daily opening prices [15]. In the WT-Multi-LSTM, the wavelet basis function is Haar and the number of retained principal components for denoising is 3. The CEEMD-Multi-LSTM, EEMD-Multi-LSTM, EMD-Multi-LSTM, WT-Multi-LSTM and Multi-LSTM are with the same LSTM model parameters as ICA-Multi-LSTM. But the CEEMD, EEMD, EMD and WT methods are usually suitable for single signal decomposition. So each denoised time series above are reconstructed from IMFs or wavelet components (WCs) before fed into Multi-LSTM. This is different from ICA-Multi-LSTM in which the final predicting results are reconstructed from predicting results of the independent components. And ICA-Multi-LSTM is suitable for stock market complex network dataset, which could handle the multiple time series with less calculation amount. Prophet is a recent forecasting model introduced by Facebook [27]. This model is proposed to predict the daily time series, including seasonal patterns. Single-LSTM is a single input model with similar parameters as ICA-Multi-LSTM. It uses the single input of historical opening prices to predict the future. 1D-CNN is a traditional deep learning framework that is based on 1D convolutional neural networks [28]. Experimental results are illustrated in Table 5. As shown in Table 5, the final average performance criteria are smaller for ICA-Multi-LSTM.

Table 5. Performance criteria on the stock networks.

Models	MAE	RMSE	MSE	MAE	RMSE	MSE	MAE	RMSE	MSE
	(Training)	(Training)	(Training)	(validation)	(validation)	(validation)	(Testing)	(Testing)	(Testing)
ICA-Multi-LSTM	0.4809	0.6349	0.4301	0.50746	0.66788	0.46102	0.24826	0.27728	0.08134
CEEMD-Multi-LSTM	0.52587	0.81018	0.76336	0.53626	0.72538	0.57096	0.29457	0.33532	0.15117
EEMD-Multi-LSTM	0.45969	0.70368	0.57596	0.47199	0.64125	0.45900	0.35949	0.38981	0.21623
EMD-Multi-LSTM	0.42559	0.64651	0.52103	0.56594	0.73289	0.56469	0.29039	0.33035	0.12832
WT-Multi-LSTM	0.57126	0.84307	0.86252	0.52998	0.71477	0.55637	0.38385	0.41014	0.24513
Multi-LSTM	0.51824	0.8073	0.74288	0.5682	0.75664	0.6321	0.42554	0.45282	0.26696
Prophet	0.92433	1.30890	1.92091	1.01698	1.09959	1.37334	0.75941	0.87861	0.82226
Single-LSTM	1.88343	2.58377	9.32841	1.17448	1.20207	1.90984	1.10417	1.30369	1.88658
1D-CNN	3.04857	3.99645	19.28480	2.39191	2.92371	15.52316	2.44776	3.24388	13.64232

Ulteriorly, we apply the Diebold-Mariano Test on the M-MAE to interpret the statistical significance and determine whether there is a striking difference among the comparisons of the final predicting results [29]. The p-value of the Diebold-Mariano Test is 0.001, which indicates that there is a striking difference between the final predicting results and the effectiveness of the ICA denoising process could be verified. Meanwhile, we adopt the Wilcoxon Signed-rank Test to evaluate the robustness of the proposed ICA-Multi-LSTM model. The p-value of the Wilcoxon Signed-rank Test is less than 0.001, which indicates that the proposed model with proper dimension reduction denoising process provides better predicting results. On the basis of the above analysis, the proposed model shows better performance not only from the perspective of single stock but also in terms of the whole stock network through comprehensive comparisons. The prediction accuracy could be improved by the ICA noise reduction procedure.

4. Conclusions

In this paper, a hybrid model is proposed that integrates the ICA and Multi-LSTM neural network to improve the prediction performance on the stock market complex networks. Previous researches have established certain prediction models from the perspective of single time series in economics, such as ESMD-KICA-LSSVR [8], ICA-BP [30], Bayesian-LSTM [31], ICA-SVR [32] and so on, but few studies have taken into account the relevance of the stock market complex network and ICA reconstruction, as well as the nonlinear, nonstationary and noise pollution of the stock market time series. In order to quantify the effectiveness of ICA on the prediction model and prediction performance on the stock market networks, we analyze the ICA results and denoising effect on the original data. And we make an econometric analysis about prediction results and performance comparison with the benchmark approaches. Firstly, the ICA is employed to the original data and identify the independent components. Five independent components are separated from the stock network. Meanwhile, the trading noise is reduced through ICA contribution degree analysis and dimension reduction process. Secondly, the Multi-LSTM prediction model is applied to each IC with less trading noise rather than the original time series. And the final prediction results are reconstructed from predicting results of the independent components. Finally, comparable experiments confirm that the proposed model outperforms the LSTM model without ICA from the perspective of the single stock in most cases. Additionally, since the predicting results depend on the structural properties of the time sequence in some cases, we deploy the securities plate stock market complex network as the target dataset to compare the overall performance of the proposed model. Experimental results show that the proposed model outperforms the benchmark models in terms of the stock market complex network prediction. This model is suitable for stock market complex network dataset, which could handle the multiple time series with less calculation amount. But it still needs to be optimized in practice. We believe that such studies are relevant for a better understanding of the future temporal evolution tendency of the stock market. And such methods could be used to build more effective prediction models in further study.

Conflict of interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

1. H. He, S. Dai, Effectiveness of price limit on stock market network: A time-migrated DCCA approach, *Complexity*, **2021** (2021). https://doi.org/10.1155/2021/3265843
2. S. Kumar Chandar, Hybrid models for intraday stock price forecasting based on artificial neural networks and metaheuristic algorithms, *Pattern Recognit. Lett.*, **147** (2021), 124–133. https://doi.org/10.1016/j.patrec.2021.03.030

3. A. Bose, C. Hsu, S. S. Roy, K. C. Lee, B. Mohammadi-ivatloo, S. Abimannan, Forecasting stock price by hybrid model of cascading Multivariate Adaptive Regression Splines and Deep Neural Network, *Comput. Electr. Eng.*, **95** (2021), 107405. https://doi.org/10.1016/j.compeleceng.2021.107405
4. A. Thakkar, K. Chaudhari, A comprehensive survey on deep neural networks for stock market: The need, challenges, and future directions, *Expert Syst. Appl.*, **177** (2021), 114800. https://doi.org/10.1016/j.eswa.2021.114800
5. H. Na, S. Kim, Predicting stock prices based on informed traders' activities using deep neural networks, *Econ. Lett.*, **204** (2021), 109917. https://doi.org/10.1016/j.econlet.2021.109917
6. S. Wang, Z. Li, J. Zhu, Z. Lin, M. Zhong, Stock selection strategy of A-share market based on rotation effect and random forest, *AIMS Math.*, **5** (2020), 4563–4580. https://doi.org/10.3934/math.2020293
7. Z. Dai, H. Zhou, X. Dong, Forecasting stock market volatility: the role of gold and exchange rate, *AIMS Math.*, **5** (2020), 5094–5105. https://doi.org/10.3934/math.2020327
8. J. E, J. Ye, L. He, H. Jin, A denoising carbon price forecasting method based on the integration of kernel independent component analysis and least squares support vector regression, *Neurocomputing*, **434** (2021), 67–79. https://doi.org/10.1016/j.neucom.2020.12.086
9. C. Lu, Integrating independent component analysis-based denoising scheme with neural network for stock price prediction, *Expert Syst. Appl.*, **37** (2010), 7056–7064. https://doi.org/10.1016/j.eswa.2010.03.012
10. L. Kao, C. Chiu, C. Lu, J. Yang, Integration of nonlinear independent component analysis and support vector regression for stock price forecasting, *Neurocomputing*, **99** (2013), 534–542. https://doi.org/10.1016/j.neucom.2012.06.037
11. J. E, Y. Bao, J. Ye, Crude oil price analysis and forecasting based on variational mode decomposition and independent component analysis, *Physica A*, **484** (2017), 412–427. https://doi.org/10.1016/j.physa.2017.04.160
12. J. E, J. Ye, H. Jin, A novel hybrid model on the prediction of time series and its application for the gold price analysis and forecasting, *Physica A*, **527** (2019), 121454. https://doi.org/10.1016/j.physa.2019.121454
13. C. Fang, F. Marle, Dealing with project complexity by matrix-based propagation modelling for project risk analysis, *J. Eng. Des.*, **24** (2013), 239–256. https://doi.org/10.1080/09544828.2012.720014
14. W. Qiao, W. Liu, E. Liu, A combination model based on wavelet transform for predicting the difference between monthly natural gas production and consumption of U.S., *Energy*, **235** (2021), 121216. https://doi.org/10.1016/j.energy.2021.121216
15. Y. Zhang, B. Yan. M. Aasma, A novel deep learning framework: Prediction and analysis of financial time series using CEEMD and LSTM, *Expert Syst. Appl.*, **159** (2020), 113609. https://doi.org/10.1016/j.eswa.2020.113609
16. W. Bao, J. Yue, Y. Rao, A deep learning framework for financial time series using stacked autoencoders and long-short term memory, *PLOS ONE*, **12** (2017), e0180944. https://doi.org/10.1371/journal.pone.0180944
17. P. Comon, Independent component analysis, A new concept, *Signal Process.*, **36** (1994), 287–314. https://doi.org/10.1016/0165-1684(94)90029-9

18. Y. Chen, J. Wu, Z. Wu, China's commercial bank stock price prediction using a novel K-means-LSTM hybrid approach, *Expert Syst. Appl.*, **202** (2022), 117370. https://doi.org/10.1016/j.eswa.2022.117370
19. K. Bandara, C. Bergmeir, S. Smyl, Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach, *Expert Syst. Appl.*, **140** (2020), 112896. https://doi.org/10.1016/j.eswa.2019.112896
20. H. G. Seedig, R. Grothmann, T. A. Runkler, Forecasting of clustered time series with recurrent neural networks and a fuzzy clustering scheme, in *2009 International Joint Conference on Neural Networks*, IEEE, (2009), 2846–2853. https://doi.org/10.1109/IJCNN.2009.5178775
21. A. Hyvärinen, Topographic independent component analysis, *Neural Comput.*, **13** (2001), 1527–1558. https://doi.org/10.1162/089976601750264992
22. W. Dai, J. Wu, C. Lu, Combining nonlinear independent component analysis and neural network for the prediction of Asian stock market indexes, *Expert Syst. Appl.*, **39** (2012), 4444–4452. https://doi.org/10.1016/j.eswa.2011.09.145
23. Y. Ouyang, Evaluation of river water quality monitoring stations by principal component analysis, *Water. Res.*, **39** (2005), 2621–2635. https://doi.org/10.1016/j.watres.2005.04.024
24. F. Zhou, Z. Huang, C. Zhang, J. Yan, Carbon price forecasting based on CEEMDAN and LSTM, *Appl. Energy*, **311** (2022), 118601. https://doi.org/10.1016/j.apenergy.2022.118601
25. Y. Wu, Q. Wu, J. Zhu, Improved EEMD-based crude oil price forecasting using LSTM networks, *Physica A*, **516** (2019), 114–124. https://doi.org/10.1016/j.physa.2018.09.120
26. M. A. Colominas, G. Schlotthauer, M. E. Torres, Improved complete ensemble EMD: A suitable tool for biomedical signal processing, *Biomed. Signal Process. Control*, **14** (2014), 19–29. https://doi.org/10.1016/j.bspc.2014.06.009
27. D. Borges, M. C. V. Nascimento, COVID-19 ICU demand forecasting: A two-stage Prophet-LSTM approach, *Appl. Soft Comput.*, **125** (2022), 109181. https://doi.org/10.1016/j.asoc.2022.109181
28. S. Mehrkanoon, Deep shared representation learning for weather elements forecasting, *Knowledge Based Syst.*, **179** (2019), 120–128. https://doi.org/10.1016/j.knosys.2019.05.009
29. F. X. Diebold, R S. Mariano, Comparing predictive accuracy, *J. Bus. Econ. Stat.*, **13** (1995), 134–144. https://doi.org/10.2307/1392185
30. H. Liu, J. Wang, K. Vajravelu, Integrating independent component analysis and principal component analysis with neural network to predict Chinese stock market, *Math. Probl. Eng.*, **2011** (2011), 1–15. https://doi.org/10.1155/2011/382659
31. B. Huang, Q. Ding, G. Sun, H. Li, Stock Prediction based on Bayesian-LSTM, in *ICMLC 2018: Proceedings of the 2018 10th International Conference on Machine Learning and Computing*, (2018), 128–133. https://doi.org/10.1145/3195106.3195170
32. C. Lu, T. Lee, C. Chiu, Financial time series forecasting using independent component analysis and support vector regression, *Decis. Support Syst.*, **47** (2009), 115–125. https://doi.org/10.1016/j.dss.2009.02.001