Multi-layer Feature Fusion method based on Convolutional Neural Network for Stock Trend Forecasting

Mengyang Liu[a], Mingyan Xu[a], Wen Jiang[a], Feng Zhao[b], Yan Jiang[c], Yong Zhang[d,*]

Mengyang Liu: 1227750364@qq.com; Mingyan Xu: xumingyan1015@163.com
Wen Jiang: 389706296@qq.com; Feng Zhao: zhaofeng1016@126.com
Yan Jiang: 675706420@qq.com
* Corresponding author: Yong Zhang. E-mail: 304043687@qq.com

[a.] School of Statistics, Shandong Technology and Business University, Yantai, China
[b.] School of Computer Science and Technology, Shandong Technology and Business University, Yantai, China
[c.] School of Law, Shandong Technology and Business University, Yantai, China
[d.] School of Information and Electronic, Shandong Technology and Business University, Yantai, China

Abstract. Stock trend prediction remains a crucial area of research in the financial domain, where the stock market is characterized by a plethora of indicators that describe various stock features. A key challenge lies in effectively capturing the interrelationships among these features and extracting them comprehensively from stock data. To tackle this, a novel multi-layer feature fusion method based on convolutional neural networks is proposed in this paper, which transforms the time series prediction problem into an image classification problem. Specifically, the representative indicators of stock features are first selected, and then the indicator data are generated into a two-dimensional matrix with different indicators and different times as two dimensions. Finally, the multi-layer feature fusion convolutional neural network model is constructed to classify the stock trends, in which the shallow convolutional features and deep convolutional features of this network are fused. Experiments on real data show that the proposed method can be more effective in fully capturing stock features to the extent of improving the prediction accuracy.

Keywords: Technical indicators, Feature fusion, Convolutional neural network, Stock trend forecastin

1 Introduction

Stock trend forecasting plays a pivotal role in the financial sector, as it enables investors to make informed decisions concerning their investments. However, the highly volatile nature of stock markets means that prediction outcomes can be influenced by various factors, encompassing political, economic, social, and other aspects [1,2]. Consequently, stock trend forecasting continues to pose a significantly challenging issue. When making stock trend forecasts, it is essential to take into account a multitude of factors and employ effective methods to enhance the accuracy and dependability of prediction.

BDEDM 2024, January 12-14, Ningbo, People's Republic of China

DOI 10.4108/eai.12-1-2024.2347222

There are many methods for stock trend forecasting studies, and these studies can be divided into two types from the point of view of data [3]. One is single-factor analysis, which analyses a solitary stock attribute from the perspective of one-dimensional time-series data, such as the closing price; the other is multi-factor analysis, that is, it considers several indicators of the stock during the forecasting process and integrates the features of these diverse data to predict future stock trends [4]. Among them, a significant proportion of financial forecasting research has concentrated on single-factor analysis [5,6]. For example, Karim et al. developed an innovative hybrid deep learning model, integrating bidirectional long short-term memory (Bi-LSTM) and gated recurrent unit (GRU) networks, to predict stock market opening prices using recent time-series price data [5]. The above research only considered individual price values, but data selected from only a single perspective cannot only adequately represent the characteristics of stock trends, but also reflect the stock trends very comprehensively.

Besides, there are many types of data describing stock features, and stock trend forecasting also relies on technical indicators [7]. Therefore, it is necessary to take these factors into account. Currently, a small number of studies have explored the integration of multiple technical indicators of stock characteristics with deep neural networks to develop sophisticated predictive models [8,9]. For example, Gao et al. designed a stock prediction system using long short-term memory (LSTM), in which the fifteen technical indicators were downscaled by principal component analysis (PCA) and then multiple series of data were fed into the input layer to predict stock market closing prices [9]. In addition to feeding technical indicators directly into the deep model as a multidimensional time series, the means of converting technical indicators into images also can be used to classify stock trends [11,12,13]. Leveraging the remarkable capabilities of Convolutional Neural Networks (CNN) in image processing, several researchers have commenced employing CNN models for predicting stock movements through an image-based perspective [10]. For example, Sim et al. proposed a stock price prediction model based on a convolutional neural network (CNN) for the stock market, where nine technical indicators were converted into time series graphs as input images for CNN [11]. Similarly, Sezer et al. introduced a new algorithmic trading model using a two-dimensional convolutional neural network, in which fifteen different technical indicators each with different parameter selections were transformed into 2-D images for prediction classification [12]. Veritably, it is more effective to analyse stock trends based on multiple factors, that is multiple technical indicators, in comparison to individual factors.

Although the experimental results validated the robustness of the CNN model, the aforementioned studies exhibit several limitations: (1) The data selection lacks representativeness. Technical indicators are selected arbitrarily rather than based on their inherent characteristics, potentially leading to deviations in prediction outcomes. (2) The neglect of multi-layer feature information. The prediction method based on the CNN model mainly only exploits deep-level convolutional feature information for classifier construction, ignoring the feature information of shallow convolutional layers, which may lead to incomplete classification features.

To overcome the above issues, we propose a multi-layer feature fusion method based on the CNN model (MFFCNN), which can more effectively extract the features of stock data and enhance prediction accuracy. This approach transforms the time series prediction challenge into an image classification problem, given the diversity of stock index data and the nature of two-dimensional data representation. Specifically, firstly the more representative technical indicators are selected from different perspectives. Secondly, the indicator data are generated into images with different indicators and times as two dimensions. Finally, a multilayer fusion CNN model is constructed to perform triple classification prediction on the images. There are mainly two aspects of improvement: (1) More comprehensive indicator selection. Technical indicators are selected from multiple perspectives of the stock and the time series forecasting problem is transformed into an image classification problem, which can provide a more comprehensive understanding of and capture of the characteristics of stock trends. (2) Fusion of multi-layer feature information. By combining shallow and deep feature information extracted from the CNN model, the multilayer feature fusion method is used to enhance the feature information, resulting in obtaining more comprehensive stock features and enhancing classification accuracy.

2 Related works

In this section, the types of technical indicators and the theory related to convolutional neural network models are introduced.

2.1 Stock technical indicators

Technical index analysis stands as a pivotal tool in stock analysis and prediction, extensively utilized by professional forex traders in the international forex market. It is a quantitative analysis method to determine the stock trend based on certain mathematical and statistical methods, using some complex calculation formulas [10]. The stock market boasts over 1,000 technical indicators, broadly categorized into three main types based on functionality: swing indicators, trend indicators, and energy indicators.

Swing indicators. Swing indicators refer to a numerical value obtained through a certain calculation formula based on the four elements of stock trading volume, price, time, and space. The numerical value fluctuates around a certain space, offering practical operational guidance through its fluctuation patterns.

Trend indicators. Trend indicators refer to indicators that analyse the strength of stock price trends based on the relationship between stock prices and indicators, using the trend analysis theory as a guiding idea and combining the characteristics of averages.

Energy indicators. Energy indicators are mainly from the perspective of volume to examine the movement of stock prices, through the volume of price to guide the actual operation of a class of indicators.

2.2 Convolutional neural network (CNN)

A convolutional neural network (CNN) is a type of artificial neural network (ANN) that can learn from images and other high-dimensional data. It is very popular in computer vision, natural language processing, pattern recognition, and other fields [17]. A CNN has a fundamental architecture that consists of the input layer, convolutional layer, pooling layer, fully connected layer, and output layer [18]. Typically, several convolutional and pooling layers are alternated. The most famous CNN algorithm, LeNet-5, has the structure shown in Figure 1 in detail.

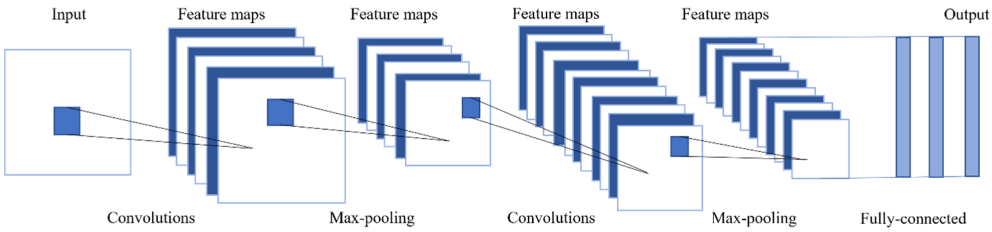

Figure 1. The LeNet-5 structure.

The CNN takes two-dimensional or three-dimensional structures as input models, utilizes convolutional layers to extract features from the input data using convolutional operations, in which the activation functions are required to enhance the nonlinear signal, and then apply pooling layer and fully connected layers to perform classification or regression tasks. The convolution operation refers to the inner product operation of the image and filter matrix, which is an element-by-element matrix multiplication operation. This operation is generally represented by $\otimes$, as shown in equation (1).

$$p_t = f(x_t \otimes \omega_t + b_t) \tag{1}$$

where p_t denotes the output value after convolution, f is the activation function, x_t indicates the input matrix, ω_t represents the filter matrix, and b_t stands the bias term. The pooling layer is often located behind the convolutional layer, which is used to decrease the output feature size of the convolutional layer, to reduce the complexity of the operation and avoid overfitting the model [14].

3 Multi-layer feature fusion of CNN model

In this section, we propose a new multi-layer fusion method based on the CNN model, which can fully consider multi-angle features and extract features of stock data more effectively to improve the accuracy of stock prediction. Specifically, we first introduce the sample construction of the stock forecasting model in Section 3.1, including data conversion and sample labelling. After that, the multilayer feature fusion method based on CNN is introduced in Section 3.2.

3.1 Sample construction of stock forecasting model

According to previous experience and the characteristics of the indicators, we selected twenty more representative indicators, being typical data in each of the three categories, from the indicator data introduced in Section 2.1, as shown in Table 1. Concretely, the Swing indicators are Stochastic oscillator (KD), Relative Strength Index (RSI), William Overbought/Oversold Index (W%R), Commodity Channel Index (CCI), and Chande Momentum Oscillator (CMO), Rate of change (ROC); Trend indicators include Exponential Moving Average (EMA), Simple Moving Average (SMA), Weighted Moving Average (WMA), Different of Moving Average

(DMA), Moving Average Convergence and Divergence (MACD), Directional Movement Index (DX), Stop and Reverse (SAR); Energy indicators are On Balance Volume (OBV), Accumulation/Distribution Line (AD, AODSC), Volume (VOL).

Table 1. The parameters of technical indicators.

Indicator Name	Parameter	Indicator Name	Parameter	Indicator Name	Parameter
KD	fastk=9, slowk=3, slowd=3	EMA	n=5 n=20	OBV	-
RSI	N=12	SMA	n=5 n=20	AD	-
W%R	n=14	WMA	n=5 n=20	AODSC	-
CCI	n=14	MACD	fast =12, slow=26, signal=9	VOL	-
CMO	n=14	DX	n=14	-	-
ROC	n=10	SAR	-	-	-

Considering that the input for convolutional neural networks typically takes the form of a two-dimensional image, time variation is incorporated as an additional dimension during sample generation. The sampling procedure involves creating 20x20 size samples via a sliding window operation, as shown in Figure 2. Concurrently, the pace of the sliding window is set to 1 for increasing the sample capacity. For sample labelling, we divide the stock samples into three categories: down, oscillation, and up, as shown in equation (2).

$$label = \begin{cases} -1, change < -0.1 \\ 0, -0.1 \le change \le 0.1 \\ 1, change > 0.1 \end{cases} \tag{2}$$

Regarding the change of stocks in the next week (i.e. five days), we mark samples of stocks above 0.1 as up with 1, below -0.1 as down with -1, and in-between as an oscillator with 0. At the same time, we employ One Hot Encoding for labelling, which is used, i.e., [1,0,0] for down, [0,1,0] for the oscillator, and [0,0,1] for up.

Figure 2. The process of generating the sample matrix.

3.2 Multi-layer feature fusion method based on CNN

In the case of conventional convolutional neural networks, only the deep-level convolutional features are taken into account, and the shallow-level convolutional feature information is indirectly neglected. Therefore, to fully consider the features in this data, a multi-layer feature fusion method based on CNN is proposed. Specifically, we construct a convolutional neural network classification model on which a fusion layer is added to effectively fuse the features of the lower convolutional layers with the features learned from the higher convolutional layers before feeding them into the classifier.

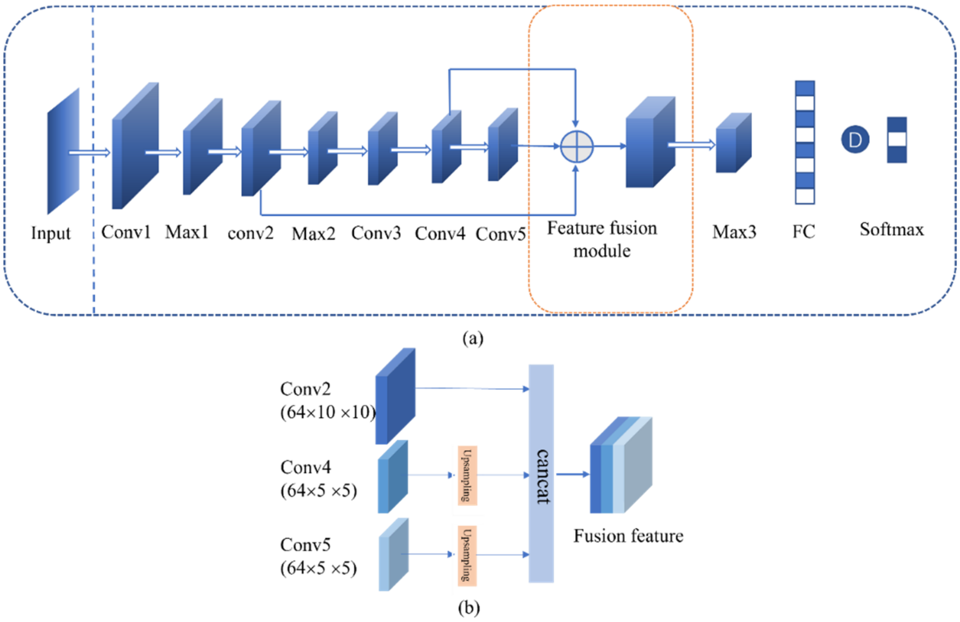

Figure 3. The network model of multi-layer feature fusion method based on CNN.

Considering that the size of the input matrix in this study is 20×20, the total number of convolutional layers is five [14]. The network model of this algorithm is shown in Figure 3 (a), which contains an input layer, five convolutional layers, three pooling layers, two up-sampling layers, one feature fusion module, two dropout layers, two fully connected layers, and one output layer. The feature fusion module is performed by a splicing fusion operation as shown in Figure 3 (b). The main steps of the algorithm: firstly, the sample matrix is fed into the input layer as the input image, and the convolution layer and the pooling layer are used for feature extraction. Then, the up-sampling operation is performed on the output of conv4 and conv5 respectively to obtain a feature map with the same size as the output of conv2, and the three parts of the features are spliced and fused by the fusion layer. In addition, the dropout layer is added to prevent overfitting. Finally, the fusion features are classified through the full connection layer and sent to the output layer.

The parameters such as the convolutional kernel and sliding step size for each layer corresponding to the multi-layer feature fusion CNN are shown in Table 2. The most common kernel sizes in the literature are 3×3, 5×5, and 7×7, but considering the size of the input samples in this study, we adopted the filter size of 3×3 chosen for the CNN filter. In addition to this, one of the disadvantages of larger kernel sizes is the loss of more information. Notably, the activation function of each layer is ReLU, and the loss function used for the classification layer is cross Entropy Loss shown in equation (3).

$$Loss = -N^{-1}\sum_{i=0}^{N-1}\sum_{k=0}^{K-1} y_{i,k} \ln p_{i,k} \tag{3}$$

Where N stands the total number of samples, K indicates the number of classification labels, $y_{i,k}$ represents the value of the ith sample taken on the kth classification label, and $p_{i,k}$ denotes the probability that the ith sample is predicted to be the kth classification label. By fitting this loss function, the inter-class distance is also increased to some extent.

Table 2. Network description of multi-layer feature fusion CNN.

Layer	Type	Kernel size	Stride	Padding
Input	input			
Conv1	convolution	(3,3)	1	same
Max1	pooling	(2,2)	1	
Conv2	convolution	(3,3)	1	same
Max2	pooling			
Conv3	convolution	(2,2)	1	same
Conv4	convolution	(2,2)	1	same
Conv5	convolution	(2,2)	1	same
UpSampling1	up Sampling	(2,2)	1	
UpSampling2	up Sampling	(2,2)	1	
Concat	concat			
Max3	pooling	(2,2)	1	
Dropout	dropout			
FC	Fully connection			

During the training process, the loss function is taken as the objective and the optimizer is exploited to perform the optimal decomposition. The total number of epochs, the maximum learning rate, and the optimizer function are 50, 0.01, and Adam, respectively. These parameters are opted on the basis of common parameters in correlative studies and experimental debugging [15,16].

4 Experiment and analysis

To verify the effectiveness of our proposed method, we experimentally analyse the model from three aspects. Specifically, we first elaborate on the experimental data and perform preprocessing in Section 4.1. The effect of the selected indicators on the experimental results is shown in Section 4.2. In Section 4.3, the impact of different convolutional feature fusion layers on the results is analysed. A comparison with different prediction methods is presented in Section 4.4. The experimental results emphasize that our proposed method can effectively extract the features of stock data as well as improve the accuracy of stock trend prediction.

4.1 Experimental data and preprocessing

The data in the experiment is obtained through the Tushare financial data interface, which contains 47 stocks from the SSE 50 constituents from January 12, 2018, to January 9, 2023. There are 20 indicator data for each stock, which is the indicator data calculated according to the indicator formula mentioned in section 3.1. The data need to be normalized concerning the different metrics. Considering that the values of the image matrix are generally positive, Min-Max normalization is chosen. The 20 indicators for each stock were individually normalized and their computation is shown in equation (4).

$$y_i = \frac{x_i - \min_{1 \le j \le m} x_j}{\max_{1 \le j \le m} x_j - \min_{1 \le j \le m} x_j} \tag{4}$$

In addition, we chose to perform elimination processing for cases where missing values exist in the data. For performance evaluation, we use classification accuracy (ACC), loss function (LOSS), and Fβ-score (Fβ) as the metrics. For the up and down scenarios, more attention is paid to how many of the samples labelled as increasing returns are correctly predicted, so the parameter β is set to 1/2. The larger the value of ACC and Fβ, the better the performance of prediction.

4.2 The impact of indicator selection on experimental results

In order to highlight the validity of selecting indicator data from multiple perspectives on the experimental results, we designed three comparison schemes regarding indicator selection. Specifically, based on the data in section 3.1, we select data from just one of the three primary categories of indicators for each scheme. Considering the small number of indicator data selected from one category, we adopt a strategy of selecting different parameters for that indicator, making it consistent with the number of indicator data in this study.

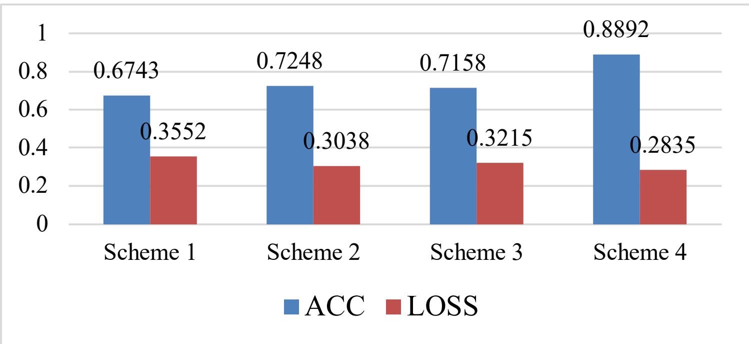

Figure 4. The experimental results using data from different indicators.

The experimental results are displayed in Figure 4, which emphasizes the impact of utilizing different indicator selection schemes on the results. In Figure 4, Scheme 1, Scheme 2, and Scheme 3 denote the selection of only swing, trend, and energy indicators, respectively, while Scheme 4 represents the inclusion of indicators from all three categories, which is the experimental data for this study.

In Figure 4, the comparison shows that Scheme 4 has the highest accuracy as well as relatively small losses. This shows that when considering indicator data from multiple perspectives, the data characteristics can be more fully represented, resulting in better experimental results.

4.3 The impact of multilayer convolutional feature fusion on results

To determine the optimal number of fusion layers for enhanced experimental outcomes, we conducted a comparative analysis of the prediction results derived from fusing different convolutional layers. Our network structure contains five convolutional layers, for which we established four distinct fusion schemes. Specifically, the first one merges the output features of conv1, conv2, and conv3, representing shallow-layer features; the second one combines the output features of conv3, conv4, and conv5, indicative of deep-layer features; the other two are comprehensive fusion methods that integrate both shallow and deep features, fusing the output features of Conv1 and Conv2 with Conv4 and Conv5, respectively.

Table 3. The predictive performance of different fusion schemes.

	Fusion 1	Fusion 2	Fusion 3	Fusion 4
ACC	0.7832	0.8436	0.7650	**0.8892**
F-β	0.6012	0.6264	0.5667	**0.6864**

The experimental results are displayed in Table 3, where the above Fusion schemes are shown from left to right in the table, with the bold font representing the best performance. Noting that Fusion 4 has the highest accuracy rate, we chose the approach of fusing conv2 with the output features of conv4 and conv5. From Table 3, we can obtain that (1) Fusion 1 has a lower accuracy rate compared to Fusion 2, which indicates that the feature information represented by the shallow features to achieve stock classification is shallow; whereas the deeper convolution layer outputs a convolution layer that embodies an increasing amount of image information, which expresses the overall information of the deeper convolution. (2) The accuracy of Fusion 3 and Fusion 4 is higher than that of Fusion 1 and 2, which indicates that it is not enough to consider only shallow features or deep feature fusion. By combining the shallow and deep features, this feature map contains both the detailed information of the shallow features and the overall information of the deep convolutional layers. This strengthens the model's ability to recognize matrix images, and can effectively multi-stock feature maps for more accurate classification.

4.4 The predictive performance of multi-layer feature fusion CNN model

To verify the effect of the MFFCNN on the performance of stock trend prediction, we compare the multilayer feature fusion CNN model with the traditional CNN model and the cross-layer connected CNN model. All three models are based on the network structure using five convolutional layers. The multilayer feature fusion CNN model passes the output features of three convolutional layers (conv2, conv4, conv5) to the fusion layer for multilayer feature fusion as mentioned in the previous experiment. The traditional CNN model is the original network structure without adding the fusion layer. The cross-layer connection CNN model makes cross-layer connections in the way of connecting the first pooled layer with the last fully connected layer. The stock trend prediction results of the three models on this dataset are shown in Figure 5.

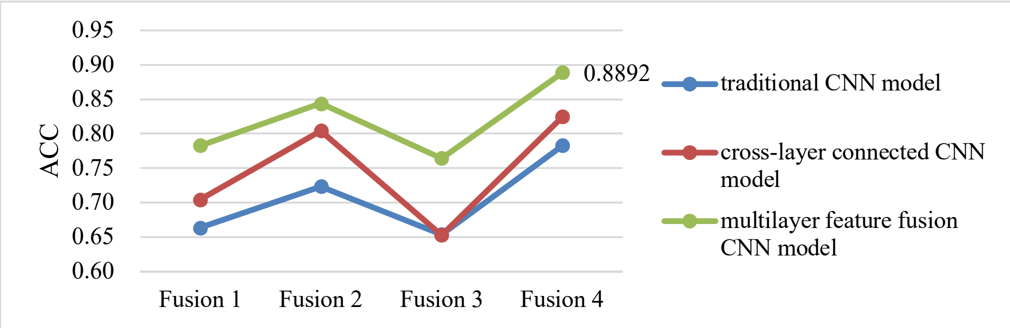

Figure 5. The result by different models based on CNN

As can be seen in Figure 5, the accuracy of the multilayer feature fusion CNN model is higher than the other two models. Compared to the cross-layer connected CNN that just utilizes the features extracted from the two pooling layers to predict the classification, the accuracy is improved. Once again, the previous experimental results were validated, which showed that better classification performance can be achieved when fully considering shallow and deep features.

To further emphasize the predictive performance of MFFCNN, we compare the predictive performance of different models, including logistic regression (LR), back propagation neural network (BP), long short-term memory networks (LSTM), and convolutional neural network. Considering that some of these models have one-dimensional inputs, two-dimensional matrix data need to be converted to one-dimensional before modelling.

Table 4. The results by using different prediction models.

	LR	BP	LSTM	CNN	MFFCNN
ACC	0.5632	0.6948	0.7852	0.7836	**0.8892**
F-β	0.4987	0.6036	0.6297	0.6523	**0.6864**

The prediction results of different models are shown in Table 4, from which it can be seen that the accuracy of multilayer feature fusion CNN is higher compared to other prediction models. We can conclude that: (1) Compared to other time series models, the methods based on the CNN model have some advantages in stock price prediction. It shows that the model can also be applied to process time series data and proves the effectiveness of transforming the time series prediction problem into an image classification problem. (2) The MFFCNN model also has outstanding performance compared to the CNN model, which reveals that our proposed multi-layer fusion approach further improves the performance of stock trend prediction.

5 Conclusion and Future Prospects

In this paper, a multilayer feature fusion convolutional neural network prediction model based on multi-angle stock indicators is proposed to improve the accuracy of stock trend prediction. Specifically, representative data is first selected from various perspectives of stock technical indicators and transformed into matrix images. Then the sample images are labelled with categories and the sample capacity is increased by sliding window. Finally, a multilayer feature fusion CNN model is applied to classify the trend categories. In order to verify the effectiveness of the multilayer feature fusion CNN model, we conduct three experiments aimed at demonstrating its robustness.

The experimental results demonstrate that: (1) selecting data from all angles of stock technical indicators can more comprehensively characterize stock trends and enable the subsequent capture of more stock features. (2) The multilayer fusion method can effectively utilize the output features of each convolutional layer and fully capture the informative features of the stock trend by combining the shallow features with the deep features, thus improving the recognition accuracy of the model. (3) By transforming the time series prediction problem into an image classification problem, the multilayer feature fusion CNN model can also be applied to processing time series data. Through the analysis of these experimental results and algorithms, our method has been proven to be very effective in stock trend prediction.

The outlook for future research is as follows: (1) This article has certain limitations that only use daily frequency stock price data. Therefore, in future research, it can be considered to add financial data or more high-frequency data to expand information sources and enlarge the sample size, so as to utilize deep learning more effectively. (2) This study analyses stock trends from an image perspective, but time series models also have certain advantages in dealing with stock prices. Therefore, future research can consider adding the time series model on this basis to synthesize the advantages of both and improve the model's robustness.

Acknowledgments. This work was supported in part by the National Natural Science Foundation of China (62176140, 61972235, 61976124, 61873117, 82001775, 61976125) and the Central Guidance on Local Science and Technology Development Fund of Shandong Province (YDZX2022093).

References

[1] Liu H, Qi L, and Sun M. (2022) Short-Term Stock Price Prediction Based on CAE-LSTM Method. *Wireless Communications and Mobile Computing*, *2022*.

[2] Lu W, Li J, Li Y, Sun A and Wang J. (2020) A CNN-LSTM-based model to forecast stock prices. *Complexity*, *2020*, 1-10.

[3] Ozbayoglu A M, Gudelek M U, and Sezer O B. (2020) Deep learning for financial applications: a survey. *Papers*, *2020*.

[4] Wang C, Chen Y, Zhang S, and Zhang Q. (2022) Stock market index prediction using deep transformer model. *Expert Systems with Application*.

[5] Lv P, Wu Q, Xu J and Shu Y. (2022) Stock Index Prediction Based on Time Series Decomposition and Hybrid Model. *Entropy (Basel)*.24(2):146.

[6] Karim M E, Foysal M, and Das S. (2022). Stock price prediction using Bi-LSTM and GRU-based hybrid deep learning approach. *Proceedings of Third Doctoral Symposium on Computational Intelligence.* 701-711.

[7] Zhang Z. (2022) Research on Stock Price Prediction Based on PCA-LSTM Model. *Academic Journal of Business amp; Management*, *4*(3):42-47.

[8] Agrawal M, Shukla P K, Nair R, Nayyar A and Masud, M. (2022) Stock Prediction Based on Technical Indicators Using Deep Learning Model. *Computers, Materials & Continua*, *70*(1): 287-304.

[9] Gao T and Chai Y. (2018) Improving stock closing price prediction using recurrent neural network and technical indicators. *Neural computation*, *30*(10):2833-2854.
[10] Gudelek M U, Boluk S A and Ozbayoglu A M. (2017). A deep learning based stock trading model with 2-D CNN trend detection. In *2017 IEEE symposium series on computational intelligence (SSCI)*. 2017: 1-8.
[11] Sim H S, Kim H I and Ahn J J. (2019). Is deep learning for image recognition applicable to stock market prediction? *Complexity*, *2019*.
[12] Sezer O B and Ozbayoglu A M. (2018) Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach. *Applied Soft Computing*, *70*:525-538.
[13] Ciresan D C, Meier U, Masci J, Gambardella L M and Schmidhuber J. (2011) Flexible, high performance convolutional neural networks for image classification. In *Twenty-second international joint conference on artificial intelligence*. 1237–1242.
[14] Khodaee P, Esfahanipour A and Taheri H M. (2022) Forecasting turning points in stock price by applying a novel hybrid CNN-LSTM-ResNet model fed by 2D segmented images. *Engineering Applications of Artificial Intelligence*, 116:105464.
[15] Sinha S, Mishra S, Mishra V and Ahmed T. (2022) Sector influence aware stock trend prediction using 3D convolutional neural network. *Journal of King Saud University-Computer and Information Sciences*, *34*(4):1511-1522.
[16] Ma Y, Han R and Wang W. (2021) Portfolio optimization with return prediction using deep learning and machine learning. *Expert Systems with Applications*, *165*, 113973.
[17] Zhang J, Ye L and Lai Y. (2023) Stock Price Prediction Using CNN-BiLSTM-Attention Model. *Mathematics*, *11*(9):1985.
[18] Chen N. (2023) Visual recognition and prediction analysis of China's real estate index and stock trend based on CNN-LSTM algorithm optimized by neural networks. *Plos one*, *18*(2):e0282159.