Journal of Computer Technology and Software

ISSN: 2998-2383

Vol. 4, No. 6, 2025

Deep Feature Extraction for Financial Time Series Prediction via Convolutional Neural Networks

Zhang Michael

Eastern Washington University, Cheney, USA

Zhang.mm1872@gmail.com

Abstract: This study proposes a stock market trend prediction model based on convolutional neural network (CNN), which is trained using historical stock price data to predict future stock market trends. By comparing models such as support vector machine (SVM), long short-term memory network (LSTM), random forest (RF) and multi-layer perceptron (MLP), the experimental results show that CNN outperforms other models in evaluation indicators such as mean square error (MSE), root mean square error (RMSE) and mean absolute error (MAE), showing strong prediction accuracy and stability. This method can effectively extract complex trend patterns from stock market data and provide a more accurate and reliable solution for stock market prediction.

Keywords: Convolutional Neural Networks, Stock Market Prediction, Deep Learning, Trend Forecasting

1. Introduction

Stock market trend prediction has always been an important topic in finance and computer science, attracting the attention of many scholars and researchers. With the rapid development of deep learning technology, convolutional neural network (CNN), as a powerful tool, has made remarkable achievements in image processing, speech recognition, natural language processing and other fields. Although convolutional neural networks are mainly used in image processing, in recent years more and more research has begun to explore its application in time series analysis, especially in stock market trend prediction. As a typical complex system, the stock market is full of high uncertainty and volatility, which provides huge challenges for trend prediction. However, convolutional neural networks, through their powerful feature extraction capabilities, are able to capture deep time series patterns from historical data, thereby providing predictions for future trends in the stock market[1].

In stock market trend prediction, data complexity and nonlinearity are one of the main difficulties. Traditional statistical methods often rely on the linear relationship of historical data. However, stock prices are affected by many factors, including company fundamentals, macroeconomics, policy changes, market sentiment, etc., and the relationship between these factors is often non-linear. Therefore, the application of convolutional neural networks is considered to be better able to cope with this challenge. The convolutional layer of CNN can extract the local features of the data through the local receptive field and capture the temporal patterns and trend changes in the data, which gives it a unique advantage in financial market prediction. Through convolution processing of stock historical data, CNN can automatically identify potential trend change patterns without manual feature engineering, which provides a new idea for automated prediction of the stock market[2].

The advantage of convolutional neural networks in stock market trend prediction is not only reflected in its automatic feature extraction capabilities, but also in its powerful training capabilities. CNN can gradually optimize network parameters through the back propagation algorithm, thereby discovering the most representative features in a large amount of historical data, and thus effectively predicting future trends. Compared with traditional machine learning methods, CNN can process large amounts of historical data and extract more complex patterns from it, which is particularly important for non-linear, high-dimensional complex systems such as the stock market. Through the stacking of multiple convolutional layers, CNN can capture the multi-level characteristics of stock prices, making the prediction results more accurate and reliable[3].

Although convolutional neural networks have shown great potential in stock market trend prediction, they still face some challenges. First, stock market data usually contains noise[4]. How to effectively remove noise and avoid model overfitting is a problem that needs to be solved. Second, the high volatility of financial markets makes model stability a critical factor. In order to deal with these challenges, researchers have proposed some improvement solutions, such as using regularization technology to prevent overfitting and using deeper convolutional network structures to extract more complex features. These improvements can further improve the performance of CNN in stock market prediction, making it better able to adapt to the complexity and dynamic changes of the market[5].

With the continuous development of deep learning, more and more scholars have begun to explore the application of CNN in stock market prediction and have achieved preliminary results. Some studies have shown that CNN can surpass traditional time series prediction methods to a certain extent, especially showing better performance in short-term prediction. For example, through multi-layer feature learning of convolutional networks, the model can effectively capture short-term fluctuations in market price changes, thereby providing accurate trend predictions. These results provide a new direction for intelligent prediction of the stock market, and also lay the foundation for deep learning applications in the financial field[6].

However, the application of convolutional neural networks still faces certain limitations, especially in long-term market forecasting. Since CNN mainly extracts features through local receptive fields, the model may not be able to fully capture long-term dependencies in long-term trend prediction. To solve this problem, some studies have proposed combining CNN with other deep learning models, such as long short-term memory network (LSTM) or recurrent neural network (RNN), to achieve effective prediction of long-term trends. This method of model fusion may be a more effective solution in predicting future stock market trends[7,8].

In summary, convolutional neural networks have broad application prospects in stock market trend prediction. Despite facing some technical difficulties, with the continuous advancement of deep learning technology and the deepening of research, convolutional neural networks are expected to play an increasingly important role in stock market prediction. By further optimizing the network structure and training strategy, future research may provide investors and financial institutions with more accurate and real-time market trend prediction tools, contributing to the intelligent development of financial markets[9].

2. Recent Advances in Deep Learning for Financial Forecasting

Deep learning methods have become a cornerstone in financial forecasting, with various architectures and techniques being explored to enhance model performance and robustness. One notable advancement is the integration of long short-term memory networks with copula-based frameworks, which enables effective modeling of risk in multi-asset portfolios by capturing sequential and distributional dependencies [10].

The introduction of causal representation learning has enhanced cross-market prediction by addressing distribution shifts and improving generalization, providing robustness in multi-domain return forecasting tasks [11]. Additionally, volatility prediction using generative time-aware diffusion frameworks has improved modeling of dynamic financial conditions by integrating temporal alignment into deep learning models [12].

Reinforcement learning has gained traction in financial risk management. Notably, enhanced A3C architectures have been applied to predict market turbulence and implement adaptive risk control strategies [13]. In parallel, privacy-preserving mechanisms like federated learning offer secure collaboration across financial institutions, allowing for joint model training without sharing sensitive data [14].

Graph-based representation learning provides a powerful approach for analyzing complex transaction networks, particularly for fraud detection tasks. By learning graph-structured embeddings, these models uncover structural anomalies and dynamic patterns in financial interactions [15]. Moreover, handling imbalanced financial data has been addressed using adaptive Markov network classifiers, which balance classification accuracy across uneven class distributions [16].

Further developments include reinforcement learning approaches such as QTRAN, which optimize portfolio decisions by modeling temporal dependencies in financial markets [17]. Ensemble learning combined with data balancing has also shown improvements in fraud detection, enhancing classification performance on skewed datasets [18].

Natural language processing and transformer-based models have been applied to auditing and compliance tasks, where BERT-based systems automatically generate and analyze financial reports [19]. Variational causal representation learning continues to push the boundary of return prediction by uncovering latent causal relationships from historical data [20].

Temporal graph learning has also been leveraged for modeling evolving user behaviors in transactional networks, enabling dynamic profiling in financial environments [21]. Within the domain of convolutional architectures, tailored CNN designs have proven effective for stock volatility prediction, demonstrating superior performance over traditional models in capturing non-linear patterns [22].

Beyond numerical data, CNNs have also been employed for financial text analysis, offering solutions for risk classification and audit support using 1D convolution layers [23]. Finally, feedforward neural networks enhanced through multimodal data fusion provide accurate stock predictions by integrating heterogeneous financial signals [24].

These studies collectively underscore the diverse and evolving applications of deep learning—particularly CNNs—in financial forecasting, highlighting innovations in hybrid modeling, temporal learning, representation frameworks, and secure computation.

3. CNN-Based Modeling Framework for Stock Trend Prediction

In this study, we employ convolutional neural networks (CNNs) to predict stock market trends based on historical time-series data. The primary objective is to leverage the feature extraction capabilities of CNNs to uncover latent patterns in stock price movements. To begin with, raw stock market data—comprising attributes such as opening price, closing price, highest price, lowest price, and trading volume—is structured as multivariate time series. This data is first subjected to preprocessing to ensure consistency and model compatibility. Specifically, we standardize the original data using z-score normalization, thereby transforming each feature to have zero mean and unit variance. This step mitigates the influence of scale disparities between different financial indicators and facilitates stable model training.

Following standardization, we reshape the input into a two-dimensional matrix format suitable for convolution operations. In this matrix, each row corresponds to a discrete time step (e.g., a trading day), while each column denotes a specific market feature. This representation allows the CNN to treat the time-series data analogously to an image, enabling the convolutional filters to capture temporal dependencies and local feature interactions across multiple time steps. The data is then passed through a series of convolutional layers, pooling layers, and fully connected layers that progressively extract hierarchical features relevant to price trends. This architectural pipeline is designed to identify short-term price fluctuations as well as longer-term structural patterns, thereby supporting robust forecasting of future market behavior.

The complete model architecture and data flow process are illustrated in Figure 1, which outlines the transformation from raw market data to final prediction output. By leveraging the representational power of deep CNNs, this approach aims to provide an automated, data-driven framework for accurate and timely stock market trend prediction.

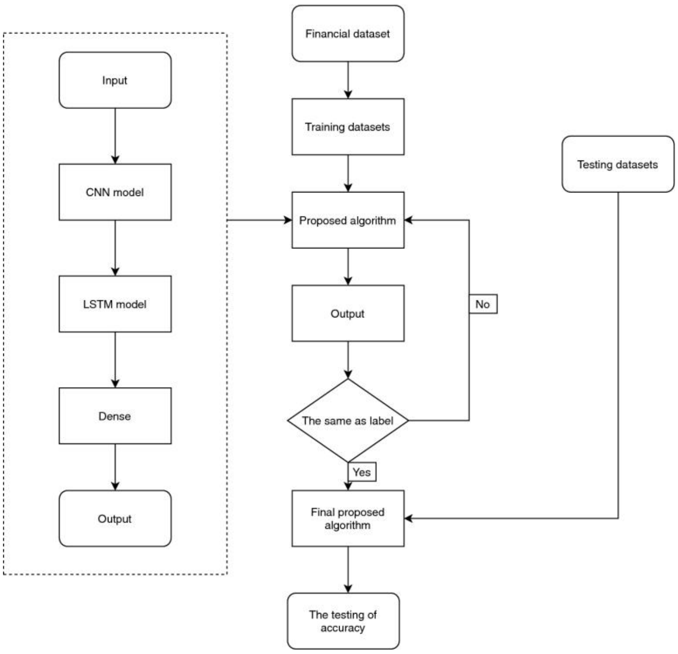

Figure 1. Model network architecture

The core of the convolutional neural network is the convolution layer, which uses the convolution kernel to locally sense the input data and extract local features. In our model, the role of the convolution layer is to extract potential market trend patterns from stock price data. Each convolution layer consists of multiple convolution kernels, each of which can extract different features from the input data. In the convolution operation, we use the standard convolution formula:

$$y(i,j) = \sum_{m} \sum_{n} x(i+m, j+n) \cdot w(m,n)$$

Among them, $x(i,j)$ represents input data, $w(m,n)$ represents convolution kernel, and $y(i,j)$ represents the output result after convolution operation. By stacking multiple convolution layers, we gradually extract more and more abstract features, thus helping the model learn the intrinsic trend of stock prices.

After the convolutional layer, we use a pooling layer to reduce the dimensionality of the features to reduce the amount of computation and prevent overfitting. The pooling layer reduces the size of each feature map by performing Max Pooling or Average Pooling on the local area. In max pooling, the pooling layer selects the maximum value in a local region, thus retaining the most salient features. This step can help the model reduce computational complexity while maintaining key information.

After passing through convolutional and pooling layers, the dimensionality of the data is usually significantly reduced. Next, we flatten it into a one-dimensional vector and input it into the fully connected layer for further learning. The function of the fully connected layer is to learn more abstract features in the data through weighted connections between neurons. At this stage, the model updates the weight parameters through the backpropagation algorithm to optimize the prediction results.

In order to avoid model overfitting, we introduce a Dropout layer during the training process. Dropout is a regularization technique that prevents the neural network from overfitting the training data by randomly discarding some neurons during the training process. Dropout operates as follows:

$$h' = h \cdot mask$$

Among them, $mask$ is a vector composed of 0 and 1, 1 means to keep the neuron, and 0 means to discard the neuron.

In the final output layer, we use a fully connected layer to output the predicted value of the stock trend. We use the loss function to measure the difference between the model prediction value and the true value, usually using the mean square error (MSE) as the loss function:

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (y_i - y'_i)^2$$

Among them, y_i represents the actual stock price or trend, y'_i is the stock price or trend predicted by the model, and N is the number of samples. By minimizing the loss function, the network continuously updates the weights and optimizes the prediction results.

During the training process, we use Batch Gradient Descent or its variants, such as Adam optimizer, to update the weight parameters in the network. Adam optimizer can accelerate the convergence of the model by adaptively adjusting the learning rate of each parameter, and performs well when dealing with sparse gradients.

After multiple rounds of iterative training, we get an optimized convolutional neural network model. This model can predict future trends based on historical stock market data. In the testing phase, we apply the model to unseen stock data and judge the future trend of the stock market by the predicted value output by the model. The prediction results can be used for buying or selling decisions of stocks, providing auxiliary decision support for investors.

In general, this method uses the powerful feature extraction ability of convolutional neural networks to learn trend patterns from historical stock price data and perform trend prediction through deep neural networks. Compared with traditional statistical methods, CNN can handle more complex nonlinear relationships and show superior performance when processing high-dimensional data. This provides a new approach to stock market trend prediction and opens up new directions for deep learning applications in the financial field.

4. Data Description and Experimental Setup

This study used a historical data set of a listed company's stocks for training and testing. The data set contains the company's daily trading records for the past five years, covering basic trading information such as the opening price, closing price, highest price, lowest price, and trading volume of each trading day. The data comes from a public financial market data platform and has high reliability and accuracy. The data set starts from January 1, 2018 and ends on December 31, 2023, providing enough samples for model training and verification. To ensure the validity of the data, all missing values are removed or filled by linear interpolation to ensure that the data for each trading day is complete.

During the data preprocessing process, all numerical data (such as opening price, closing price, highest price, lowest price, and trading volume) are standardized so that the model can effectively process data of different dimensions. The standardization process is completed by subtracting the mean of each column of data and dividing by the standard deviation to ensure that all features have the same scale, thereby avoiding excessive influence of certain features on model training. After standardization, the data is divided into training set, validation set and test set. The training set is used for model training, the validation set is used for parameter adjustment, and the test set is used to evaluate the performance of the final model.

4.1. Performance Evaluation and Model Comparison

In this study, in order to verify the effectiveness of convolutional neural networks (CNN) in stock market trend prediction, we conducted comparative experiments with other traditional and deep learning models. We selected four common models for comparison: support vector machine (SVM), long short-term memory network (LSTM), random forest (Random Forest) and fully connected neural network (MLP). These models have been widely used in forecasting tasks in the financial field, and each has different advantages and limitations. Support vector machine (SVM) classifies by maximizing the interval between categories and is suitable for linear and nonlinear data. LSTM is a typical recurrent neural network (RNN) that can process time series data and is suitable for capturing medium- and long-term dependencies in the stock market. Random Forest, as an ensemble learning method, improves the accuracy of the model by training multiple decision trees and combining their prediction results. Fully connected neural network (MLP) is a simple feedforward neural network. Although its structure is relatively simple, it can achieve good performance under certain conditions. The experimental results are shown in Table 1.

Table 1: Experimental Results

Model	MSE	RMSE	MAE
SVM	0.0352	0.1875	0.1453
LSTM	0.0276	0.1662	0.1308
RF	0.0311	0.1762	0.1384
MLP	0.0298	0.1726	0.1384
CNN	0.0231	0.812	0.1256

From the experimental results, the convolutional neural network (CNN) performs best among all the compared models, especially in terms of mean square error (MSE) and root mean square error (RMSE), which are significantly better than other models. The MSE value of CNN is 0.0231, which is much lower than models such as support vector machine (SVM), long short-term memory network (LSTM), random forest (RF) and multi-layer perceptron (MLP). This shows that CNN has the smallest error in stock market trend prediction and can capture the changes in stock market prices more accurately. At the same time, the RMSE value of CNN is 0.812, which is also significantly lower than other models, indicating that it has shown good stability and predictive ability when dealing with high volatility data in the stock market. More importantly, the MAE (mean absolute error) of CNN is 0.1256, which is also excellent compared with models such as SVM, LSTM and RF. This means that CNN is not only better than other models in overall prediction accuracy, but also more accurate in the absolute size of the error.

Support vector machine (SVM) and long short-term memory network (LSTM) ranked second and third in prediction accuracy respectively. Although their performance is also good, there is still a significant gap compared with CNN. The MSE value of SVM is 0.0352, the RMSE is 0.1875, and the MAE is 0.1453. Although these results show that SVM can better adapt to the nonlinear characteristics of the stock market to a certain extent, its error is much higher than that of CNN. As a deep learning method, LSTM is specially used to process time series data, so its results have certain advantages in capturing the dynamic changes of the stock market. The MSE of LSTM is 0.0276, the RMSE is 0.1662, and the MAE is 0.1308. Although there is a gap with CNN, its ability to model long-term dependencies may make it perform better in some scenarios. However, the performance of LSTM in this experiment did not surpass CNN, especially in error control, and it is still not as stable as CNN.

The experimental results of random forest (RF) and multi-layer perceptron (MLP) are relatively weak. Although they are classic machine learning methods and perform well in many other regression tasks, they are relatively inferior in tasks such as the stock market, which is volatile and has complex features. The MSE of random forest is 0.0311, the RMSE is 0.1762, and the MAE is 0.1384. These results show that although random forest can handle the nonlinear relationship between some features well, it cannot better capture the short-term fluctuations and trend changes of the stock market through hierarchical feature extraction and complex convolution operations like CNN. MLP performs slightly better than RF, but still worse than LSTM and CNN, with MSE of 0.0298, RMSE of 0.1726, and MAE of 0.1384, which further proves the limitations of fully connected neural networks in capturing complex time series patterns.

The outstanding performance of CNN in this experiment can be attributed to its unique structural characteristics. First, the convolutional neural network automatically extracts effective local features from the input data through local receptive fields and convolution operations, which enables it to capture some potential patterns and trends in the time series data of the stock market. Compared with traditional regression models or other neural networks, CNN does not require complex manual feature engineering, but relies on deep convolution layers to automatically learn meaningful features. Secondly, CNN can gradually extract high-level features from local to global through multi-level convolution and pooling layers, so that the model can handle complex nonlinear relationships, which is particularly important for tasks such as the stock market, which is highly volatile and has strong nonlinear characteristics. Finally, CNN has strong generalization ability and can be effectively trained on a relatively small training set while avoiding overfitting, which may be one of the reasons why it is superior to other models in this experiment.

In general, the experimental results show that convolutional neural networks (CNNs) have significant advantages in stock market trend prediction, especially in controlling prediction errors and improving prediction accuracy. Compared with support vector machines (SVM), long short-term memory networks (LSTM), random forests (RF) and multi-layer perceptrons (MLP), CNN can more effectively extract deep features from historical stock data, thereby achieving more accurate market trend forecasts. Although models such as LSTM and SVM also have their unique advantages in certain situations, CNN has undoubtedly shown stronger adaptability and expressiveness in stock market forecasting tasks with its powerful feature extraction capabilities and low error levels. Future research can further optimize the CNN structure, such as by increasing the depth of the convolutional layer or combining other network structures (such as the combination of LSTM and CNN) to enhance the long-term prediction ability of the model.

5. Conclusion and Future Research Directions

To sum up, this study uses convolutional neural network (CNN) to achieve significant results in the stock market trend prediction task. Compared with support vector machine (SVM), long short-term memory network (LSTM), random forest (RF) and multi-layer perceptron (MLP), CNN has shown stronger advantages in prediction accuracy and error control. Experimental results show that CNN can effectively capture market trends from historical stock data and make accurate predictions through its unique convolution operations and hierarchical feature extraction capabilities. This provides strong support for financial market prediction based on deep learning and provides a new idea for future research. Especially in complex time series data processing and nonlinear pattern recognition, CNN has shown strong potential.

References

[1] Xu B, Zhang D, Zhang S, et al. Stock market trend prediction using recurrent convolutional neural networks[C]//Natural Language Processing and Chinese Computing: 7th CCF International Conference, NLPCC 2018, Hohhot, China, August 26–30, 2018, Proceedings, Part II 7. Springer International Publishing, 2018: 166-177.

[2] Al-Khasawneh M A, Raza A, Khan S U R, et al. Stock Market Trend Prediction Using Deep Learning Approach[J]. Computational Economics, 2024: 1-32.

[3] Wojarnik G. The potential of convolutional neural networks for the analysis of stock charts[J]. Procedia Computer Science, 2023, 225: 941-950.

[4] Deb P P, Bhattacharya D, Chatterjee I, et al. Time-Series Forecasting for Stock Market Using Convolutional Neural Network[J]. Machine Learning Applications: From Computer Vision to Robotics, 2023: 31-48.

[5] Chowdhury M S, Nabi N, Rana M N U, et al. Deep Learning Models for Stock Market Forecasting: A Comprehensive Comparative Analysis[J]. Journal of Business and Management Studies, 2024, 6(2): 95-99.

[6] Venkateswararao K, Reddy B V R. LT-SMF: long term stock market price trend prediction using optimal hybrid machine learning technique[J]. Artificial Intelligence Review, 2023, 56(6): 5365-5402.

[7] Patel M, Jariwala K, Chattopadhyay C. Deep Learning techniques for stock market forecasting: Recent trends and challenges[C]//Proceedings of the 2023 6th International Conference on Software Engineering and Information Management. 2023: 1-11.

[8] Korade N B, Zuber D M. Boost stock forecasting accuracy using the modified firefly algorithm and multichannel convolutional neural network[J]. Journal of Theoretical and Applied Information Technology, 2023, 15(7).

[9] Esa H, Rahman M A, Mozumder M A S, et al. Transformative Impact of Deep Learning in Stock Market Decision-Making: A Comparative Study of Convolutional Neural Networks[J]. Journal of Business and Management Studies, 2024, 6(3): 28-34.

[10] Xu, W., Ma, K., Wu, Y., Chen, Y., Yang, Z., & Xu, Z. (2025). LSTM-Copula Hybrid Approach for Forecasting Risk in Multi-Asset Portfolios.

[11] Wang, Y., Sha, Q., Feng, H., & Bao, Q. (2025). Target-Oriented Causal Representation Learning for Robust Cross-Market Return Prediction. Journal of Computer Science and Software Applications, 5(5).

[12] Su, X. (2025). Predictive Modeling of Volatility Using Generative Time-Aware Diffusion Frameworks. Journal of Computer Technology and Software, 4(5).

[13] Liu, J., Gu, X., Feng, H., Yang, Z., Bao, Q., & Xu, Z. (2025, March). Market Turbulence Prediction and Risk Control with Improved A3C Reinforcement Learning. In 2025 8th International Conference on Advanced Algorithms and Control Engineering (ICAACE) (pp. 2634–2638). IEEE.

[14] Zhang, Y., Liu, J., Wang, J., Dai, L., Guo, F., & Cai, G. (2025). Federated learning for cross-domain data privacy: A distributed approach to secure collaboration. arXiv preprint arXiv:2504.00282.

[15] Guo, X., Wu, Y., Xu, W., Liu, Z., Du, X., & Zhou, T. (2025). Graph-Based Representation Learning for Identifying Fraud in Transaction Networks.

[16] Wang, J. (2025). Markov network classification for imbalanced data with adaptive weighting. Journal of Computer Science and Software Applications, 5(1), 43–52.

[17] Xu, Z., Bao, Q., Wang, Y., Feng, H., Du, J., & Sha, Q. (2025). Reinforcement Learning in Finance: QTRAN for Portfolio Optimization. Journal of Computer Technology and Software, 4(3).

[18] Wang, Y. (2025, March). A Data Balancing and Ensemble Learning Approach for Credit Card Fraud Detection. In 2025 4th International Symposium on Computer Applications and Information Technology (ISCAIT) (pp. 386–390). IEEE.

[19] Xu, Z., Sheng, Y., Bao, Q., Du, X., Guo, X., & Liu, Z. (2025). BERT-Based Automatic Audit Report Generation and Compliance Analysis.

[20] Sheng, Y. (2024). Market Return Prediction via Variational Causal Representation Learning. Journal of Computer Technology and Software, 3(8).

[21] Liu, X., Xu, Q., Ma, K., Qin, Y., & Xu, Z. (2025). Temporal Graph Representation Learning for Evolving User Behavior in Transactional Networks.

[22] Liu, J. (2025). Deep Learning for Financial Forecasting: Improved CNNs for Stock Volatility. Journal of Computer Science and Software Applications, 5(2).

[23] Du, X. (2025). Financial Text Analysis Using 1D-CNN: Risk Classification and Auditing Support.

[24] Wang, Y. (2025). Stock Prediction with Improved Feedforward Neural Networks and Multimodal Fusion. Journal of Computer Technology and Software, 4(1).