Artificial Intelligence and Computing Innovations
Intelligence Science Publishing
https://www.ispublishing.com
ISSN: 3070-4944
Vo.4, No.2, 2024

Modeling Financial Market Dynamics with Temporal and Relational Learning: An LSTM-GNN Approach

Yihan Zheng

Columbia University, New York, USA

Gary.zheng22@gmail.com

Abstract: This study proposes a stock market prediction method based on long short-term memory network (LSTM) and graph neural network (GNN) to simultaneously capture the time series characteristics of individual stocks and the market network relationship. Traditional stock prediction methods mainly rely on time series models such as LSTM and GRU, but these methods often ignore the complex interactive relationships between stocks. To solve this problem, this study constructs a graph structure of the stock market and uses GNN for relationship modeling, so that the prediction model can comprehensively consider the historical trading data of individual stocks and the overall market dynamics. The experiment is verified based on the trading data of the S&P 500 index in 2018, and MLP, 1D-CNN, GRU and Transformer are used as comparison models. The experimental results show that LSTM+GNN achieves the best performance in terms of mean square error (MSE), root mean square error (RMSE) and coefficient of determination (R^2), verifying the effectiveness of integrating time series and market structure information. In addition, ablation experiments and optimizer sensitivity experiments further illustrate the contribution of GNN structure and AdamW optimizer in improving prediction accuracy. Future research can combine multimodal data and reinforcement learning to further optimize the model and explore more efficient financial market prediction methods to enhance the decision-making ability of intelligent trading systems.

Keywords: Stock prediction, Time series analysis, Graph neural networks, Deep learning.

1. Introduction

As an important part of the global economy, the stock market's price fluctuations are affected by many factors, such as the macroeconomy, market sentiment, and corporate financial conditions[1]. Due to market uncertainty, how to accurately predict stock prices has become an important research topic in the fields of finance and computer science[2]. Traditional stock analysis methods mainly rely on fundamental analysis, technical analysis, and quantitative trading models. However, these methods often fail to fully utilize massive market data, especially when faced with complex nonlinear relationships, the limitations of traditional statistical methods gradually emerge. With the development of artificial intelligence and deep learning technologies, data-driven methods have gradually become an important tool in the field of stock prediction, which can automatically extract features from historical data and establish more complex prediction models[3].

In the financial market, stock prices often show a high degree of time dependence, that is, past price information has an important impact on future prices. This time series characteristic makes long short-term memory networks (LSTMs) widely used in stock prediction tasks[4]. LSTM can effectively capture long-term dependencies and avoid the problem of gradient vanishing in traditional recurrent neural networks (RNNs), making it an ideal choice for processing financial time series data. However, relying solely on time series data for prediction may ignore another important characteristic of the stock market - network correlation. The price fluctuations between stocks often have complex mutual influence relationships. For example, stocks within an industry may have a linkage effect, and the interaction between different market sectors may also affect the price of individual stocks. Therefore, relying solely on LSTM to model the time series of a single stock may not be enough to fully reflect market dynamics[5].

In order to solve this problem, the application of graph neural networks (GNNs) in the financial field has gradually attracted attention in recent years. GNNs can effectively model the relationship network in the stock market, such as building a graph structure between stocks based on industry classification, shareholding relationships between companies, and similarities in market fluctuations, and use graph neural networks for information dissemination and feature learning. This method can further integrate the correlation information between stocks while considering time series factors, thereby improving the accuracy of predictions. Especially under extreme market conditions, such as financial crises or periods of abnormal market fluctuations, the network relationship between stocks may be more important than a single time series feature. Therefore, combining GNN with LSTM provides a new idea for stock prediction, that is, on the basis of time series modeling, the network structure information of the stock market is introduced to improve the robustness and generalization ability of predictions[6].

In addition, with the rapid growth of financial market data, how to efficiently process multi-source heterogeneous data is also a key issue in current stock prediction research. Modern stock trading involves not only structured data such as historical prices and trading volumes, but also unstructured data such as news texts, social media sentiment, and investor behavior. The rise of deep learning provides the possibility of integrating these multimodal data, allowing researchers to simultaneously use multidimensional factors such as time series information, network relationships, and market sentiment in prediction tasks to build a more intelligent prediction system. This interdisciplinary research direction not only promotes the development of financial intelligence, but also provides a scientific basis for financial risk management and investment decisions, which helps to improve market efficiency and reduce investor uncertainty[7].

Overall, the research on stock market prediction not only has important theoretical value, but also has a wide range of impacts in practical applications. From the perspective of academic research, exploring stock prediction methods based on LSTM and GNN can enrich the cross-research field of time series analysis and graph neural networks, and provide new solutions for complex pattern modeling of time series data. From the perspective of financial applications, accurate stock prediction can provide scientific investment references for investment institutions, financial technology companies, and individual investors, optimize investment portfolios, improve returns, and reduce market risks. Therefore, stock prediction is not only a frontier research topic at the intersection of computer science and finance, but will also play a key role in the future development of intelligent finance.

2. Related Work

Stock market prediction has always been an important research direction in the field of financial technology and machine learning. Scholars have proposed a variety of methods to improve the accuracy and stability of predictions[8,9]. Early studies were mainly based on traditional statistical methods, such as ARIMA (autoregressive integrated moving average model) and GARCH (generalized autoregressive conditional heteroskedasticity model), which model market behavior by analyzing the historical trends and volatility of time series. However, due to the complexity and nonlinear characteristics of the stock market, these linear models have limitations in dealing with long-term dependencies and complex market dynamics. Therefore, in recent years, machine learning and deep learning methods have gradually become mainstream, and researchers have begun to use more complex neural network architectures to capture nonlinear changes in stock prices[10].

In the application of deep learning models, recurrent neural networks (RNNs) and their variants, long short-term memory networks (LSTMs), are widely used in time series prediction tasks. LSTM can effectively overcome the gradient vanishing problem of traditional RNNs and capture long-term dependencies through its memory mechanism, so it has been widely studied in the field of stock prediction. For example, some studies use LSTM to process historical trading data of individual stocks, combined with technical indicators such as moving averages and relative strength indexes, to improve prediction accuracy[11,12]. In addition, methods such as bidirectional LSTM, attention mechanism and Transformer have also been introduced to further enhance the modeling ability of market information. However, these methods mainly focus on time series data, while ignoring the structural information in the stock market, that is, the relationship and mutual influence between stocks[13].

In recent years, graph neural network (GNN) has provided a new idea for stock prediction, and researchers have begun to pay attention to the network relationship between stocks in the market, such as the equity structure, industry connection, news co-occurrence relationship between companies, etc[14]. GNN-based research usually constructs the graph structure of the stock market, regards stocks as nodes, and associations as edges, and learns the interactive information between stocks through methods such as graph convolutional neural network (GCN) and graph attention network (GAT). For example, some studies combine GNN and LSTM, using GNN to capture the dependency between stocks, and using LSTM to model the time series changes of individual stocks, thereby achieving good results in prediction accuracy and stability. This type of research shows that multimodal learning methods that combine time series and network information are an important direction for future stock prediction research[15].

3. Method

In this study, we combined the long short-term memory network (LSTM) and the graph neural network (GNN) to build a stock prediction model to simultaneously model the time series characteristics of individual stocks and the structural information of the market. The change of stock prices is not only affected by its own historical data, but also by the relationship between other stocks in the market. Therefore, we use LSTM to extract the time-dependent characteristics of individual stocks, and use GNN to model the network relationship of the stock market to improve the accuracy and stability of the prediction. The model architecture is shown in Figure 1.

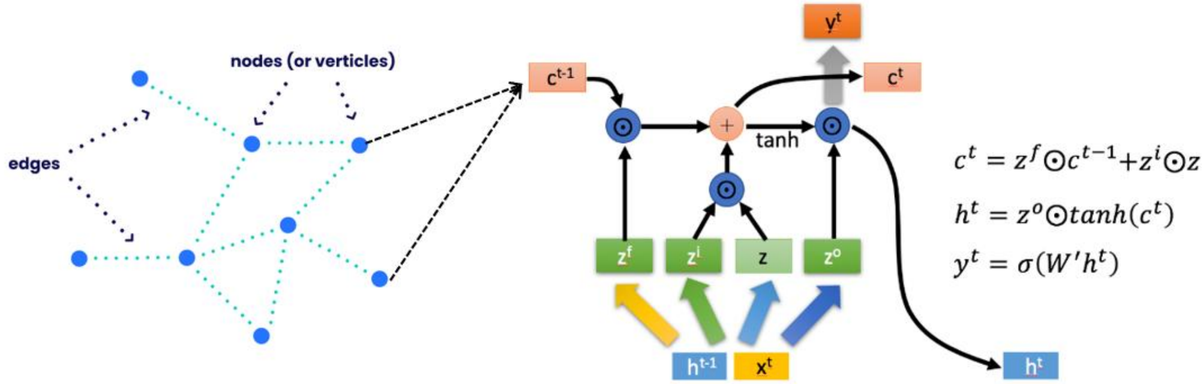

Figure 1. Overall model architecture

Assume that the stock market contains N stocks, and the historical transaction data of each stock constitutes a time series. We use matrix X to represent the features of all stocks at T time steps, where the F dimension represents the features of each stock (such as opening price, closing price, trading volume, etc.). For a single stock i, its time series is represented as x_i. We first use LSTM to model the time series of each stock, and the hidden state of LSTM is updated as follows:

$$h_t = f(W_h h_{t-1} + W_x x_t + b) \tag{1}$$

Among them, h_t is the hidden state of the tth time step, W_h and W_x are weight matrices, b is the bias term, and f is the nonlinear transformation of the LSTM unit. After LSTM encoding, we get the time series feature representation of each stock, where d is the dimension of the LSTM hidden layer.

To model the relationship between stocks, we construct a stock market graph A, where V represents the stock set and E represents the association between stocks. We use the correlation matrix A as the adjacency matrix of the graph, where A_i reflects the strength of the association between stock i and stock j, which can be constructed based on industry classification, price correlation, or market co-occurrence relationship. Based on this graph structure, we use a graph convolutional neural network (GCN) for information propagation, and its update formula is as follows:

$$H^{(l+1)} = \sigma(D'^{\frac{-1}{2}} A' D'^{\frac{-1}{2}} H^{(l)} W^{(l)}) \tag{2}$$

Among them, $A' = A + I$ is the adjacency matrix with self-loops, D' is the degree matrix of A', $H^{(l)}$ is the feature representation of the l-th layer, $W^{(l)}$ is the trainable weight matrix, and σ is the nonlinear activation function. After multiple layers of GCN update, we get the global representation H_G of the stock, which combines the mutual influence relationship between stocks.

Finally, we merge the time series feature h_i output by LSTM and the market relationship feature h_G extracted by GNN, and input them into the fully connected layer to predict the stock price:

$$y_i = W_o(h_i \| h_G) + b_o \tag{3}$$

Among them, $\|$ represents the vector concatenation operation, W_o is the weight matrix of the fully connected layer, and b_o is the bias term. The final output y_i represents the future price forecast of stock *i*. The model uses the mean square error (MSE) loss function during training:

$$L = \frac{1}{N}\sum_{i=1}^{N}(y_i - y'_i)^2 \quad (4)$$

Among them, y'_i is the real price and y_i is the predicted price. Through gradient descent optimization, the model can gradually adjust the parameters to minimize the prediction error, thereby improving the accuracy of stock market prediction.

4. Experiment

4.1 Dataset Introduction

The S&P 500 Index (Standard & Poor's 500 Index) serves as a key benchmark for evaluating the overall performance of the U.S. stock market. It comprises 500 large-cap companies listed on major U.S. exchanges, covering diverse sectors such as technology, finance, healthcare, and energy, and is therefore widely regarded as an indicator of overall economic health. This dataset provides daily trading data for the S&P 500 Index in 2018, including key variables such as opening price, closing price, highest price, lowest price, and trading volume. As shown in Figure 2, the closing price exhibits significant fluctuations throughout the year, reflecting dynamic market conditions. These data offer a detailed view of market movements and can be utilized for analyzing trends, designing investment strategies, and conducting risk management studies.

The core variable in this dataset is the daily closing price, which captures the overall market trajectory and is commonly used in technical analysis and time series forecasting. In addition, trading volume reflects market liquidity and investor activity, while daily returns, calculated from changes in closing prices, provide insight into market volatility. Together, these features support a wide range of quantitative finance applications, including price prediction, volatility modeling, and portfolio optimization.

Notably, 2018 was characterized by heightened market volatility. Influenced by factors such as Federal Reserve interest rate hikes, Sino-U.S. trade tensions, and global economic uncertainty, the S&P 500 experienced substantial fluctuations. As illustrated in Figure 2, the index underwent multiple upward and downward movements over the year. This dataset therefore enables in-depth analysis of market behavior under varying macroeconomic conditions and supports advanced modeling approaches, including deep learning and statistical methods, for applications such as risk forecasting, asset allocation, and market sentiment analysis..

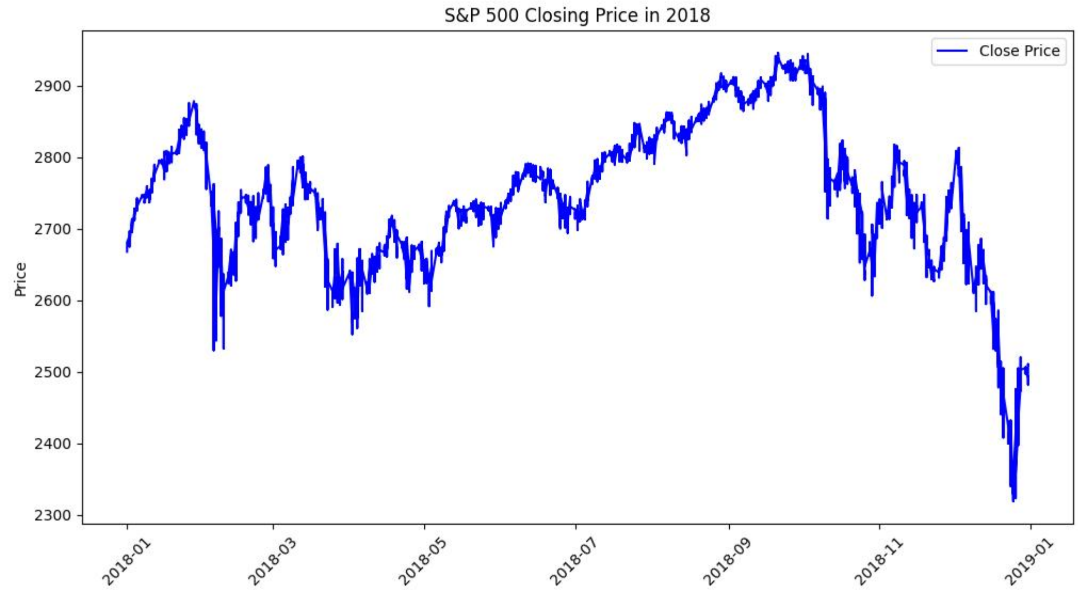

Figure 2. S&P 500 Closing Price in 2018

Furthermore, this paper also gives the daily return rate curve, and the experimental results are shown in Figure 3.

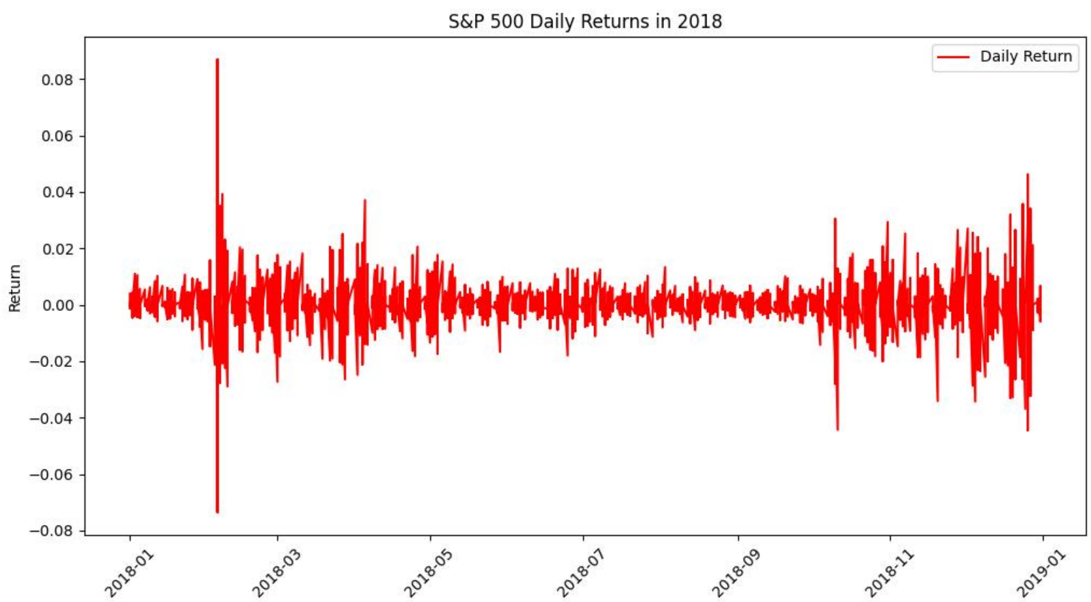

Figure 3. S&P 500 Daily Returns in 2018

4.2 Experimental setup

In this experiment, we use the stock price data of the S&P 500 index in 2018 to conduct experiments and use the LSTM+GNN method to predict stocks. The main goal of the experiment is to evaluate the performance of this method in stock market time series prediction and verify the effectiveness of GNN in modeling market network relationships. The experiment includes three parts: data preprocessing, model training and testing, and performance evaluation.

The experimental environment includes Intel i9-13900K CPU, 64GB DDR5 memory, NVIDIA RTX 4090D GPU, Ubuntu 22.04 operating system, PyTorch 2.0.1 deep learning framework, and CUDA 12.1 for accelerated computing. The experimental data comes from the daily stock trading data of S&P 500 in 2018 obtained by akshare, including opening price, closing price, highest price, lowest price, trading volume and other features. We first normalize the data and construct training samples using a time sliding window, where the first 30 days of data are used to predict the stock price on the 31st day.

During model training, we use the Adam optimizer, set the learning rate to 0.001, the batch size to 64, and use the mean square error (MSE) as the loss function. During training, we use 80% of the data as the training set, 10% as the validation set, and 10% as the test set, and the number of training rounds is set to 50. After training, we use the mean square error (MSE), root mean square error (RMSE), and determination coefficient (R^2) to evaluate the model performance and compare it with baseline models (such as LSTM, random forest, XGBoost) to verify the contribution of the GNN structure in the stock prediction task.

4.3 Experimental results

In order to comprehensively evaluate the performance of the LSTM+GNN model in the stock prediction task, this experiment uses a variety of baseline models for comparison, including MLP (multi-layer perceptron), 1D-CNN (single-dimensional convolutional neural network), GRU (gated recurrent unit) and Transformer. MLP, as a traditional neural network model, can process structured data, but lacks the ability to model time series; 1D-CNN extracts local time features through convolution operations and is suitable for capturing short-term patterns; GRU, as a variant of RNN, can model time dependencies and reduce computational overhead; and Transformer relies on the self-attention mechanism to capture long-range dependencies and performs well in time series modeling. The purpose of the comparative experiment is to analyze the performance of different models in the stock prediction task and verify the advantages of LSTM+GNN in combining time series information with market network relationship modeling. The experimental results are shown in Table 1.

Table 1. Experimental results

Model	MSE	RMSE	R^2
MLP	0.0156	0.1249	0.7423
GRU	0.0128	0.1131	0.7845
1D-CNN	0.0143	0.1196	0.7632
Transformer	0.0112	0.1058	0.8104
Ours	0.0095	0.0975	0.8457

The experimental results show that in the task of stock market time series prediction, the performance of different models is significantly different. The traditional MLP is the worst because it cannot capture the dependencies of the time series, with an MSE of 0.0156, an RMSE of 0.1249, and an R^2 of only 0.7423. This shows that simple fully connected neural networks have limitations when processing highly nonlinear financial data and cannot fully explore the dynamic change patterns of market prices. In addition, 1D-CNN extracts local time features through convolution operations, which is better than MLP. The MSE drops to 0.0143, the RMSE drops to 0.1196, and the R^2 increases to

0.7632, indicating that the extraction of short-term patterns has a certain effect on stock price prediction, but it is still not enough to handle long-term dependency information.

In the recurrent neural network architecture, GRU performs significantly better than MLP and 1D-CNN, with MSE further reduced to 0.0128, RMSE down to 0.1131, and R^2 up to 0.7845. This shows that GRU better captures the temporal dependency of stock prices through the gating mechanism and improves prediction accuracy. Transformer, due to its powerful self-attention mechanism, is able to model long-term temporal dependencies, with MSE further reduced to 0.0112, RMSE only 0.1058, and R^2 up to 0.8104, showing its advantages in time series modeling. However, due to the low sensitivity of Transformer to local features, it may not fully utilize the short-term volatility information in the market, and there is still room for optimization.

The LSTM+GNN model proposed in this study achieved the best performance in all indicators, with MSE reduced to 0.0095, RMSE reduced to 0.0975, and R^2 reached 0.8457, which are significantly improved compared with other models. This shows the effectiveness of LSTM in capturing time series features, and GNN can model network relationships in the stock market, providing additional information support for prediction. Compared with models that rely solely on time series, LSTM+GNN can more comprehensively model the pattern of stock price changes by combining historical trading data of individual stocks and overall market structure information, thereby improving prediction accuracy.

Overall, the experimental results show that simple MLP and 1D-CNN cannot effectively capture the complex dynamics of the stock market. RNN and Transformer have strong time series modeling capabilities, but still fail to fully consider the network structure in the market. The LSTM+GNN proposed in this study achieved the best performance in all evaluation indicators by combining time-dependent features and market network information, proving the effectiveness and feasibility of this method in stock market prediction.

Furthermore, this paper gives the results of ablation experiments, and the experimental settings are single LSTM and single GNN. The experimental results are shown in Table 2.

Table 2. Ablation experiments

Model	MSE	RMSE	R^2
LSTM	0.0118	0.1086	0.8053
GNN	0.0109	0.1043	0.8187
Ours	0.0095	0.0975	0.8457

The results of the ablation experiment show that both LSTM and GNN can achieve good performance in the stock market prediction task when used alone, but the combination of the two further improves the prediction accuracy. As a common time series modeling method, LSTM can effectively capture the historical trading patterns of individual stocks, with an MSE of 0.0118, an RMSE of 0.1086, and an R^2 of 0.8053. However, since LSTM only relies on the historical data of individual stocks and fails to consider the correlation between stocks in the market, there is still room for improvement in prediction accuracy. In contrast, GNN makes up for the limitations of the time series model to a certain extent by modeling the relationship network of the stock market and learning the mutual influence between different stocks. Its MSE is reduced to 0.0109, RMSE is reduced to 0.1043, and R^2 is increased to 0.8187, indicating that market structure information plays an important role in stock price prediction.

Although GNN performs well in modeling stock market relationships, it only relies on the structural information of the market and does not directly model the temporal dynamic characteristics of individual stocks, which still limits it in time series prediction tasks. In contrast, the LSTM+GNN combined model we proposed achieved the best results in MSE, RMSE, and R^2, with MSE reduced to 0.0095, RMSE reduced to 0.0975, and R^2 increased to 0.8457. This shows that LSTM and GNN complement each other. LSTM is responsible for extracting the time-dependent information of individual stocks, while GNN supplements the mutual influence between stocks through market network modeling. The combination of the two can more comprehensively model the complex dynamics of the stock market and improve prediction accuracy.

Overall, the experimental results verify the effectiveness of GNN in financial market prediction, and also show that relying solely on market structure or time series characteristics will lead to information loss and affect model performance. LSTM+GNN constructs a more comprehensive stock prediction framework by simultaneously considering time series and market network relationships, further reducing errors and improving prediction stability, providing a better solution for financial market data modeling.

Finally, this paper also gives the results of the hyperparameter sensitivity experiment, intending to analyze the impact of different optimizers on the experimental results. The experimental results are shown in Table 3.

Table 3. Hyperparameter sensitivity experiment results

Optimizer	MSE	RMSE	R^2
Momentum	0.0124	0.1114	0.7932
SGD	0.0117	0.1082	0.8085
Adam	0.0102	0.1010	0.8321
AdamW	0.0095	0.0975	0.8457

The experimental results show that different optimizers have a significant impact on the convergence speed and final prediction performance of the model. Among them, Momentum, as an improved gradient descent method, can accumulate gradient information to accelerate convergence during training, but it performs relatively poorly in this experiment, with MSE of 0.0124, RMSE of 0.1114, and R^2 of 0.7932. This shows that when Momentum processes complex stock time series data, there may be certain convergence instability, and it is difficult to effectively optimize the loss function of the LSTM+GNN combination model. In contrast, the standard SGD (stochastic gradient descent) method performs slightly better, with MSE reduced to 0.0117, RMSE of 0.1082, and R^2 increased to 0.8085, indicating that under appropriate learning rates and training iterations, SGD can still achieve good results, but due to the lack of adaptive adjustment mechanism, the overall performance is still not optimal.

Compared with the traditional gradient descent method, the adaptive optimization method Adam performed better in this experiment, with MSE reduced to 0.0102, RMSE to 0.1010, and R^2 increased to 0.8321, indicating that Adam's adaptive learning rate adjustment mechanism can better adapt to the non-stationary characteristics of stock market data, accelerate convergence and improve prediction accuracy. However, Adam has the problem of unstable gradient update, which may cause certain fluctuations in the later stage of training. In order to further optimize the performance of the model, we used AdamW for experiments, and its MSE was further reduced to 0.0095, RMSE to 0.0975, and R^2 increased to 0.8457, achieving the best result. This shows that AdamW effectively alleviates the parameter overfitting problem of Adam in long-term training by introducing a weight decay mechanism, improves the generalization ability, and makes the model more stable in the stock market prediction task.

Overall, the experimental results verify the advantages of the adaptive optimization algorithm in the LSTM+GNN combination model, especially Adam and AdamW have significantly improved the convergence speed and final prediction effect compared with Momentum and SGD. AdamW achieved the best performance in this experiment, indicating that it can better balance the convergence speed and model stability in financial time series modeling, and provide a more reliable optimization solution for stock market prediction.

5. Conclusion

This study proposes a stock market prediction method that combines LSTM and GNN, aiming to capture the time series characteristics of individual stocks and the market network relationship at the same time. Experimental results show that LSTM can effectively model the time dependence of stock prices, while GNN can utilize the stock correlation information in the market. The combination of the two significantly improves the prediction accuracy. In the comparative experiment, our method achieved the best performance in MSE, RMSE and R^2 indicators, indicating that in the complex environment of the stock market, the fusion modeling method that considers time dynamic characteristics and market structure information is of great value. Further ablation experiments also verified the respective contributions of LSTM and GNN, and proved that the synergy of the two can provide more robust prediction capabilities.

In the hyperparameter sensitivity experiment, we analyzed the impact of different optimizers on the model training effect and found that AdamW performed best in the stock prediction task, further improving the stability and generalization ability of the model. This result shows that in financial market modeling, a suitable optimization algorithm is crucial to the final performance of the model. In addition, comparative experiments with models such as MLP, 1D-CNN, GRU, and Transformer further verified the superiority of the LSTM+GNN combination method. Compared with models that rely solely on time series or market relationships, our method can more comprehensively model stock price change patterns and provide a more accurate solution for financial forecasting.

Although this study has achieved good experimental results, there are still some issues that deserve further exploration. For example, this study is mainly based on the historical trading data of the S&P 500 index for modeling, while other information in the market, such as news text, investor sentiment, and macroeconomic indicators, may also have an important impact on stock price prediction. In future research, we can try to integrate these multimodal data into the LSTM+GNN framework to further improve the accuracy of prediction. In addition, the composition method of GNN is crucial to modeling market relationships. Different composition methods (such as those based on causal reasoning, industry classification, or trading patterns) may affect the final prediction effect. In the future, we can further explore the optimal market network construction method.

Future research can also focus on more efficient deep learning architectures and more powerful optimization strategies, such as introducing self-supervised learning, graph attention networks (GAT), variational autoencoders (VAE) and other methods to further enhance the model's expressiveness. At the same time, the real-time requirements of the actual financial market are high, so in the future, it is possible to combine reinforcement learning to optimize dynamic trading strategies and explore how to make optimal investment decisions based on predictions. With the continuous development of big data and computing power, the LSTM+GNN-based method is expected to play a role in a wider range of financial market applications and promote the further development of intelligent finance.

References

[1] J. Lai, "Attention Alignment under Logical Constraints for Reliable Financial Statement Reasoning," 2024.

[2] D. Matsunaga, T. Suzumura and T. Takahashi, "Exploring Graph Neural Networks for Stock Market Predictions with Rolling Window Analysis," arXiv preprint arXiv:1909.10660, 2019.

[3] D. Filipović and P. Pasricha, "Empirical Asset Pricing via Ensemble Gaussian Process Regression," arXiv preprint arXiv:2212.01048, 2022.

[4] P. Nithya, C. D. Nandakumar and S. Srinivasan, "Analysis of the Life Insurance Business Performance Based on COVID by Using Machine Learning Algorithms," Advances in Machine Learning and Big Data Analytics I: ICMLBDA 2023, pp. 347, 2023.

[5] W. Bao, J. Yue and Y. Rao, "A Deep Learning Framework for Financial Time Series Using Stacked Autoencoders and Long-Short Term Memory," PLOS ONE, vol. 12, no. 7, e0180944, 2017.

[6] L. Liu, B. Peng and J. Yu, "Stock Price Prediction Based on Machine Learning," Proceedings of the 2022 2nd International Conference on Economic Development and Business Culture (ICEDBC 2022), pp. 1277-1282, 2022.

[7] C. Wang, Y. Chen, S. Zhang and Q. Zhang, "Stock Market Index Prediction Using Deep Transformer Model," Expert Systems with Applications, vol. 208, 118128, 2022.

[8] L. Peng, "Stock Price Prediction of 'Google' Based on Machine Learning," BCP Business & Management, vol. 34, pp. 912-918, 2022.

[9] W. Khan, M. A. Ghazanfar, M. A. Azam, A. Karami, K. H. Alyoubi and A. S. Alfakeeh, "Stock Market Prediction Using Machine Learning Classifiers and Social Media, News," Journal of Ambient Intelligence and Humanized Computing, vol. 13, no. 7, pp. 3433-3456, 2022.

[10] I. Orquin-Serrano, "Predictive Power of Adaptive Candlestick Patterns in Forex Market: EURUSD Case," Mathematics, vol. 8, no. 5, 802, 2020.

[11] D. Daiya and C. Lin, "Stock Movement Prediction and Portfolio Management via Multimodal Learning with Transformer," Proceedings of the 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3305-3309, 2021.

[12] Q. Gan, "Large Language Model Framework for Multi-Document Financial Anomaly Detection in Intelligent Auditing via Semantic Mapping and Risk Reasoning," 2024.

[13] S. Li, J. Wu, X. Jiang and K. Xu, "Chart GCN: Learning Chart Information with a Graph Convolutional Network for Stock Movement Prediction," Knowledge-Based Systems, vol. 248, 108842, 2022.

[14] W. Li, R. Bao, K. Harimoto, D. Chen, J. Xu and Q. Su, "Modeling the Stock Relation with Graph Network for Overnight Stock Movement Prediction," Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, pp. 4541-4547, 2021.

[15] Y. Li, Z. Simon and D. Turkington, "Investable and Interpretable Machine Learning for Equities," The Journal of Financial Data Science, vol. 4, no. 1, pp. 54-74, 2022.