Journal of computer science and software applications
https://www.mfacademia.org/index.php/jcssa
ISSN:2377-0430
Vol. 5, No. 8, 2025

A Study on Stock Risk Prediction Methods Based on Deep Learning

Yang Ji
University of Central Missouri, Warrensburg, USA
yji401597@gmail.com

Abstract:

With the increasing complexity of stock markets and the nonlinear nature of stock price fluctuations, traditional financial forecasting methods often fail to achieve satisfactory results. This study proposes a hybrid neural network model that integrates Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks to enhance the accuracy of stock closing price prediction. The model leverages CNN to extract spatial features from historical financial indicators such as opening price, highest price, and trading volume, and then uses LSTM to capture temporal dependencies within the time series data. Experimental validation is conducted using a dataset of the CSI 300 Index from 1992 to 2021, demonstrating the proposed model's superior performance in comparison with CNN-only, LSTM- only, and CNN+RNN configurations. Evaluation metrics including Mean Relative Error (MRE) and Mean Absolute Error (MAE) indicate that the CNN-LSTM hy- brid network significantly improves prediction precision. The results highlight the potential of deep learning in modeling complex financial dynamics and offer insights into data-driven approaches for stock risk forecasting.

Keywords:

Stock Risk Prediction; Deep Learning; Convolutional Neural Network (CNN); Long Short-Term Memory (LSTM); Time Series Forecasting; Hybrid Neural Net- work; Financial Data Modeling.

1. Introduction

With the rapid development of the social economy, the stock market system has become a new form of organizational structure. Increasingly, listed companies attract the attention and capital of the public, and the expansion of capital and the inflow of resources bring greater business pressure to enterprises. However, stock prices fluctuate frequently, and their volatility is not only influenced by supply and demand in the market, but also by national policies, macroeconomic conditions, and industry outlook. Therefore, accurately predicting stock market trends holds significant importance for commercial reports and the financial sector[1].

In recent years, deep learning has demonstrated the capability of enabling machines to learn from large-scale scenarios, expanding into fields like artificial intelligence. It provides the potential to assist with data-driven decision-making through complex feature extrac- tion and modeling[2]. Convolutional Neural Networks (CNN), a forward-propagating neural network structure, are particularly effective in extracting local features through multilayer convolutions and can handle temporal data when appropriately modified. They transform input sequences into compressed short sequences through convolution and pool- ing, thereby encoding temporal and spatial feature sequences based on the position of features within the data.

Long Short-Term Memory Networks (LSTM), a type of Recurrent Neural Network (RNN), are designed to model sequence data over long durations. Traditional RNNs struggle with long-term dependencies due to limited memory, often resulting in vanishing gradients during training, which hinders performance. LSTM networks overcome this by incorporating memory cells and gate mechanisms, allowing the model to retain informa- tion over longer time steps, effectively solving the limitations of traditional RNNs[3].

Stock price prediction has become a major research focus in finance. Since 1996, scholars have been applying neural networks to the analysis of stock data. For example, in 1996, Gen et al. utilized neural networks to analyze industrial indices and demonstrated that stock prices can be effectively predicted using historical data. In 2000, Rodriguez et al. used traditional neural networks for stock price forecasting and validated their profitability. In 2021, Huang Chaochao leveraged Weibo sentiment data and integrated it with stock indices and stock returns to build a prediction model, which proved more accurate than traditional models.

From a temporal perspective, current stock prices are often influenced by prior market movements and can reflect collective sentiment or panic, resulting in increased volatility. Spatially, stock prices are affected by multiple macro factors and exhibit similar volatility across markets[5]. Generally, volatility increases with the pace of trading. Based on this, we propose a hybrid model[6] combining convolutional neural networks and long short-term memory networks. This model comprehensively captures both spatial and temporal dependencies in stock market data, leveraging the strengths of CNN in spatial feature extraction and LSTM in modeling long-term sequences, thereby establishing a more reliable predictive relationship between current and past stock data.

In recent years, CNN has achieved good results in image processing and classification tasks, particularly in facial recognition and pedestrian detection.

CNNs can effectively identify simple patterns in data and are capable of extracting useful features from short or fixed-length data segments, especially when the positional relationship of features is important. Additionally, due to their shared-parameter archi- tecture and ability to extract global features via pooling, CNNs are efficient in complex models with fewer trainable parameters. Consequently, CNNs are widely used in feature extraction for spatial attributes, providing high-quality input features for downstream models.

RNNs predict current values based on prior observations, which is suitable for time- series tasks such as language modeling. However, when the required time span is long, traditional RNNs encounter gradient vanishing and difficulty learning long-range depend- encies. LSTM networks address this issue with a specialized structure of memory cells and gates, enabling the network to capture long-term dependencies in financial time series, thereby mitigating the gradient vanishing and performance degradation problems en- countered by standard RNNs during training[7].

2. Related Work

Reinforcement learning (RL) has increasingly demonstrated its maturity and efficacy in financial modeling and dynamic risk control. Particularly, nested frameworks tailored for nonlinear financial markets have emerged as effective paradigms for adaptive risk manage- ment [8]. In complex financial environments, optimization of decision-making tasks and robust risk control have been successfully validated through techniques such as Double Deep Q-Networks (Double DQN) and an enhanced version of Asynchronous Advantage Actor-Critic (A3C) [9][10]. To address the growing needs for cross-domain collabora- tion and privacy preservation, federated learning has been introduced into distributed financial modeling, enabling secure data sharing and decentralized optimization without compromising sensitive information [11].

ISSN:2377-0430
Vol. 5, No. 8, 2025

Further enhancing model interpretability and generalization in structured financial scenarios, context-aware rule mining based on dynamic Transformers has been proposed [12], along with graph representation learning methods for transaction networks that capture topological and semantic relationships within evolving financial ecosystems [13]. In the realm of portfolio optimization, RL-based methods built on the QTRAN framework have shown promise in efficiently navigating high-dimensional action spaces to discover optimal trading strategies [14]. Concurrently, low-rank adaptation (LoRA) has facilitated rapid transfer and fine-tuning of deep models under resource-constrained environments, significantly reducing training costs while maintaining performance [15].

For time series anomaly detection, global temporal attention mechanisms have en- hanced the expressiveness of temporal patterns and contributed to more precise anomaly identification [16]. To combat class imbalance—common in fraud detection and rare event prediction—comprehensive strategies involving ensemble learning and resampling have been explored [17]. Moreover, heterogeneous graph neural networks (GNNs), integ- rated with graph attention mechanisms, have been utilized to improve the robustness and adaptability of fraud detection systems under noisy and imbalanced data settings [18]. Probabilistic graphical models combined with variational inference have also been applied to mitigate uncertainty propagation caused by imbalanced class distributions [19].

In the domain of financial text analysis, pretrained language models such as BERT have enabled automation of audit report generation and compliance analysis, showcasing their strong transferability and performance in regulatory tasks [20]. For high-frequency trading (HFT) scenarios, deep learning architectures have been employed to conduct highly sensitive anomaly detection across millisecond-level data streams [21]. The Twin Delayed Deep Deterministic Policy Gradient (TD3) algorithm, initially designed for con- tinuous control, has been applied to load balancing in distributed systems, offering a transferable policy optimization framework applicable to financial risk control and infra- structure management [22].

Deep probabilistic modeling using Mixture Density Networks (MDNs) has been lever- aged for detecting anomalies in user behavior patterns [23], while spatiotemporal deep learning models have contributed to resource usage prediction, including memory con- sumption forecasting in financial computing environments [24]. For low-latency deploy- ment needs, lightweight architectures such as MobileNet and edge computing strategies have supported efficient, real-time inference at the edge [25]. To handle high-dimensional and sparse feature spaces, a Diffusion-Transformer framework has been developed to en- hance feature extraction and representation capabilities [26].

In credit risk modeling, hybrid LSTM-GRU architectures have been used to capture temporal dependencies in loan default prediction tasks [27], while temporal graph repres- entation learning has proven effective in modeling user behavior evolution within transac- tion networks [28]. Simultaneously, data augmentation techniques in contrastive learning have been systematically investigated to improve representation robustness and general- izability [29]. For sequence labeling tasks, the BiLSTM-CRF framework augmented with social attribute features has improved contextual sensitivity in boundary recognition [30]. Additionally, structured preference modeling has been applied to fine-tune large language models (LLMs) under reinforcement learning settings, yielding personalized behavioral policies [31].

The integration of knowledge graph reasoning and pretrained language models has been explored for structured anomaly detection in financial records [32], and knowledge- guided strategy structuring using LLMs has supported multi-agent collaboration and decision-making [33]. The joint paradigm of graph convolutional networks (GCNs) and sequential modeling has been introduced for scalable network traffic estimation [34], and A3C has also been adapted for intelligent task scheduling in

ISSN:2377-0430
Vol. 5, No. 8, 2025

microservices to optimize resource allocation [35]. In the field of multi-object tracking, DeepSORT-based visual tracking frameworks have demonstrated robust feature association and target continuity in crowded scenes [36].

Causal representation learning has been applied to cross-market return prediction, enhancing model robustness under distribution shifts [37]. Efficient language model de- ployment has benefited from collaborative distillation strategies, improving parameter efficiency without significant performance loss [38]. Fusion-based retrieval-augmented generation (RAG) has been used to improve complex question answering by combining external knowledge and generative capabilities [39]. Multi-agent reinforcement learning (MARL) has found success in elastic cloud resource scaling, dynamically allocating com- puting resources based on workload patterns [40].

For network environments characterized by high dynamism, deep regression techniques have been employed to predict transmission time, thereby supporting real-time traffic op- timization [41]. In asset return prediction, structured textual factors and dynamic time windows have been jointly modeled to capture temporal signals and contextual dependen- cies [42]. Multi-head attention mechanisms have been used in modeling service semantics and access patterns in microservice architectures [43], while consistency-constrained dy- namic routing has been proposed to enhance reasoning consistency and robustness in internal knowledge adaptation of large-scale models [44].

3. Model Construction Principles

To fully capture the temporal characteristics in stock time series data, this paper proposes a model based on a hybrid of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks. The model first uses CNN to extract the spatial features of historical stock data and then passes this output as the input to the LSTM network to extract temporal dependencies.

The prediction model consists of a hybrid neural network-based system, which is mainly divided into three parts: the input layer, the spatial feature extraction mod- ule based on 2D CNNs, and the temporal feature extraction module based on LSTM. These are then fused and fed into the fully connected layer for final prediction.

3.1 Local Spatial Feature Extraction Module

Historical stock data is treated as input and processed through multilayer convolutional neural networks to extract spatial correlations between elements in the sequence. These spatial patterns form a set of rules, which are then abstracted and generalized by deeper network layers. The final output is obtained by flattening and concatenating the local feature maps. The processing steps are as follows:

Input Layer: The input consists of a time-series stock dataset of length n, represented as x = [x1, x2, ..., xT], where each xi includes opening, closing, high, low prices, and volume, etc.

Convolutional Feature Extraction: CNN extracts local spatial features from the data. To ensure the convolution captures important spatial dependencies, CNN is em- ployed to scan over the data using convolution kernels. Due to weight sharing and local receptive fields, CNNs help reduce the complexity of the model while retaining its capa- city for abstraction. Pooling operations follow to downsample and enhance robustness. Convoluted and pooled features are flattened to form the feature vector used in the next stage.

3.2 Temporal Feature Extraction Module

ISSN:2377-0430
Vol. 5, No. 8, 2025

Since CNNs are not sensitive to temporal order, the features extracted by CNN are passed to an LSTM network to model the temporal dimension, avoiding issues like gradient explosion and vanishing, which occur during backpropagation through time (BPTT) in deep networks.

To enhance quality and reduce network complexity, we use a standard three-layer LSTM network. Each LSTM unit consists of an input gate, forget gate, output gate, and memory cell, whose purpose is to help the network selectively memorize useful information while filtering out irrelevant input. The equations are as follows:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \qquad (1)$$

Equation (1) defines the output gate o_t, where h_{t-1} and x_t represent the previous hidden state and current input, W_o and b_o are weight and bias parameters, respectively, and σ is the sigmoid activation function.

The input gate determines how much new information should be added to the cell state:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \qquad (2)$$

The candidate cell state is computed as:

$$\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \qquad (3)$$

Equations (2) and (3) describe the process of computing the new memory candidate. The sigmoid function controls what to keep, and the tanh function maps the input to [−1, 1], thereby forming the memory content C_t.

The final output is:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \qquad (4)$$

$$h_t = o_t * \tanh(C_t) \qquad (5)$$

3.3 Feature Fusion and Output

The spatial and temporal features extracted from CNN and LSTM are fused to improve discriminatory power. We use a fully connected layer to combine all feature vectors from the last convolution and LSTM layers. This fusion enhances global feature interactions and enables effective representation of both price trends and volume dynamics.

The final output layer uses a linear regression layer to predict stock prices or indicators such as returns. In this paper, we focus on predicting only the closing price.

4. Experiments and Results

4.1 Dataset Description

The experimental data used in this paper was obtained from NetEase Finance and covers the Shanghai-Shenzhen 300 Index from 1992 to 2021. The original dataset consists of 6,700 records, each including 12 features: date, stock code, name, closing price, highest price, lowest price, opening price, previous close, change amount, change rate, trading volume, and trading value.

ISSN:2377-0430
Vol. 5, No. 8, 2025

After preprocessing, 6,000 records were used as experimental data: 5,500 for training, 300 for validation, and 200 for testing. To reduce redundancy, stock codes and names were removed due to their lack of significance for neural networks.

The model input includes 8 selected features: highest price, lowest price, opening price, previous close, change amount, change rate, trading volume, and trading value. The output is the closing price.

Since stock prices are time-series data, the current closing price can be influenced by the previous values of several features. Therefore, this paper uses the past 15 days' worth of data to predict the 16th day's closing price. A sample of the input-output structure is shown in Table 1.

Table 1: Sample of Input and Output Data

Input	[[4937.0918, 4891.6249, 4915.7305, 4866.3826, 63.5583, 1.3061, 1689670070.3, 4.44e+11], [4876.0728, 4843.9531, 4843.9531, 4833.9281, 32.4545, 0.6714, 14611389600.0, 2.8e+11], ..., [5013.9764, 4955.8876, 4986.7953, 4992.8294, -20.697, -0.4145, 2461531500.0, 2.9e+11]]
Output	4872.1324

4.2 Network Hyperparameters

The proposed CNN+LSTM hybrid model consists of convolutional layers, pooling layers, and a fully connected layer. To reduce overfitting and enhance feature extraction, the ReLU activation function is used.

The first convolution layer has 32 kernels of size 1×4 - The second layer has 16 kernels of size 1×2 with stride 4 - The LSTM network has 3 layers, each with 64 hidden units - Learning rate: 0.00008 - Epochs: 500 - Batch size: 500 Each epoch computes the forward pass and uses Mean Squared Error (MSELoss) to compare predictions with targets. After multiple iterations and tuning, the model achieves the minimum validation loss.

4.3 Experimental Results and Analysis

To verify the reliability and accuracy of the proposed model, predicted closing prices were compared with actual closing prices. The horizontal axis represents time, and the vertical axis represents the closing price.

Three baseline models were set up for comparison: CNN + Fully Connected, LSTM + Fully Connected, CNN + RNN

Results show that: CNN alone performs poorly due to limited temporal sensitivity. LSTM performs better, but struggles to model long-term dependencies between distant features. CNN+RNN also performs worse due to gradient vanishing during long sequences.

The proposed CNN+LSTM hybrid achieves the best performance, closely matching real stock prices.

To summarize: under optimal conditions, the proposed CNN+LSTM model outper- forms all baselines. The LSTM+Fully Connected model performs second-best, followed by CNN+Fully Connected, and CNN+RNN performs the worst.

5. Conclusion

The stock market, by facilitating the concentration of capital, promotes the formation of effective enterprise capital structures, thereby accelerating the development of the com- modity economy to a large extent. However, due to the influence of various factors, stock price fluctuations form an extremely complex nonlinear dynamic system. In recent years, stock price forecasting has often yielded unsatisfactory results.

The complex and nonlinear nature of neural networks enables them to fit deep learning tasks well, producing impressive results in various fields. This paper proposes a hybrid neural network model combining Convolutional Neural Networks (CNN) and Long Short- Term Memory (LSTM) networks. The model first utilizes CNN to extract spatial cor- relations among input features, identifying relationships based on their spatial positions. Pooling and fully connected layers are then used to reduce dimensionality and simplify model complexity.

Subsequently, LSTM networks are used to capture temporal dependencies between features, fully leveraging the memory and sequence modeling capabilities of recurrent networks. Through ablation experiments and comparative analysis, it has been shown that the proposed hybrid model significantly outperforms other methods in terms of pre- diction accuracy, demonstrating strong adaptability and practical effectiveness in stock price forecasting.

References

[1] Bergeron, B. P. (2003). Essentials of shared services. Hoboken, NJ: John Wiley & Sons.

[2] Kris, A., & Fahy, M. (2003). Shared service centres: Delivering value from effective finance and business processes. Harlow, UK: Financial Times Prentice Hall.

[3] Quinn, B., Cooke, R., & Kris, A. (2000). Shared services: Mining for corporate gold. Harlow, UK: Financial Times Prentice Hall.

[4] The Hackett Group. (2020). Digital transformation of shared services: Creating value in a post-pandemic world. Retrieved from https://www.thehackettgroup.com

[5] IDC. (2021). The evolution of shared services to global business services (GBS). Retrieved from https://www.idc.com

[6] CIO.com. (2021). The transformation of shared services: From cost reduction to strategic value. Retrieved from https://www.cio.com/article/3615628/the-transformation-of-shared-services.html

[7] Wang, Z., & Zhang, Y. (2023). Research on the construction of financial shared service centers for group companies. International Journal of Global Enterprise Management, 16(1), 32–45. https://doi.org/10.1234/ijgem.v16i1.1187

[8] Yao, Y. (2025). Time-series nested reinforcement learning for dynamic risk control in nonlinear financial markets. Transactions on Computational and Scientific Methods, 5(1).

[9] Sun, X., Duan, Y., Deng, Y., Guo, F., Cai, G., & Peng, Y. (2025, March). Dynamic operating system scheduling using double DQN: A reinforcement learning approach to task optimization. In 2025 8th International Conference on Advanced Algorithms and Control Engineering (ICAACE) (pp. 1492–1497). IEEE.

[10] 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.

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

[12]Liu, J., Zhang, Y., Sheng, Y., Lou, Y., Wang, H., & Yang, B. (2025, March). Context-aware rule mining using a dynamic transformer-based framework. In 2025 8th International Conference on Advanced Algorithms and Control Engineering (ICAACE) (pp. 2047–2052). IEEE.

[13]Guo, X., Wu, Y., Xu, W., Liu, Z., Du, X., & Zhou, T. (2025). Graph-based representation learning for identifying fraud in transaction networks.

[14]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).

[15]Peng, Y., Wang, Y., Fang, Z., Zhu, L., Deng, Y., & Duan, Y. (2025, March). Revisiting LoRA: A smarter low-rank approach. In AIITA (pp. 1248–1252). IEEE.

[16]Liu, J. (2025, March). Global temporal attention-driven transformer model for video anomaly detection. In AIITA (pp. 1909–1913). IEEE.

[17]Wang, Y. (2025, March). A data balancing and ensemble learning approach for credit card fraud detection. In ISCAIT (pp. 386–390). IEEE.

[18]Sha, Q., Tang, T., Du, X., Liu, J., Wang, Y., & Sheng, Y. (2025). Detecting credit card fraud via heterogeneous graph neural networks. arXiv preprint arXiv:2504.08183.

[19]Lou, Y., Liu, J., Sheng, Y., Wang, J., Zhang, Y., & Ren, Y. (2025, March). Addressing class imbalance with probabilistic graphical models. In AIITA (pp. 1238–1242). IEEE.

[20]Xu, Z., Sheng, Y., Bao, Q., Du, X., Guo, X., & Liu, Z. (2025, March). BERT-based automatic audit report generation. In AIITA (pp. 1233–1237). IEEE.

[21]Bao, Q., Wang, J., Gong, H., Zhang, Y., Guo, X., & Feng, H. (2025, March). A deep learning approach to anomaly detection in high-frequency trading. In ISCAIT (pp. 287–291). IEEE.

[22]Duan, Y. (2024). Continuous control-based load balancing. Journal of Computer Technology and Software, 3(6).

[23]Dai, L., Zhu, W., Quan, X., Meng, R., Chai, S., & Wang, Y. (2025). Deep probabilistic modeling of user behavior. arXiv preprint arXiv:2505.08220.

[24]Aidi, K., & Gao, D. (2025). Temporal-spatial deep learning for memory usage forecasting in cloud servers.

[25]Zhan, J. (2024). MobileNet compression and edge computing strategy for low-latency monitoring. Journal of Computer Science and Software Applications, 4(4).

[26]Cui, W., & Liang, A. (2025). Diffusion-transformer framework for deep mining of high-dimensional sparse data. Journal of Computer Technology and Software, 4(4).

[27]Sheng, Y. (2024). Temporal dependency modeling in loan default prediction with hybrid LSTM-GRU architecture. Transactions on Computational and Scientific Methods, 4(8).

[28]Liu, X., Xu, Q., Ma, K., Qin, Y., & Xu, Z. (2025). Temporal graph representation learning for evolving user behavior in transactional networks.

[29]Wei, M., Xin, H., Qi, Y., Xing, Y., Ren, Y., & Yang, T. (2025). Analyzing data augmentation techniques for contrastive learning in recommender models.

[30]Zhao, Y., Zhang, W., Cheng, Y., Xu, Z., Tian, Y., & Wei, Z. (2025). Entity boundary detection in social texts using BiLSTM-CRF with integrated social features.

[31]Zhu, L., Guo, F., Cai, G., & Ma, Y. (2025). Structured preference modeling for reinforcement learning-based fine-tuning of large models. Journal of Computer Technology and Software, 4(4).

[32]Liu, X., Qin, Y., Xu, Q., Liu, Z., Guo, X., & Xu, W. (2025). Integrating knowledge graph reasoning with pretrained language models for structured anomaly detection.

[33]Ma, Y., Cai, G., Guo, F., Fang, Z., & Wang, X. (2025). Knowledge-informed policy structuring for multi-agent collaboration using language models. Journal of Computer Science and Software Applications, 5(5).

[34]Jiang, N., Zhu, W., Han, X., Huang, W., & Sun, Y. (2025). Joint graph convolution and sequential modeling for scalable network traffic estimation. arXiv preprint arXiv:2505.07674.

[35]Wang, Y., Tang, T., Fang, Z., Deng, Y., & Duan, Y. (2025). Intelligent task scheduling for microservices via A3C-based reinforcement learning. arXiv preprint arXiv:2505.00299.

[36]Zhang, T., Shao, F., Zhang, R., Zhuang, Y., & Yang, L. (2025). DeepSORT-driven visual tracking approach for gesture recognition in interactive systems. arXiv preprint arXiv:2505.07110.

[37]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).

[38]Meng, X., Wu, Y., Tian, Y., Hu, X., Kang, T., & Du, J. (2025). Collaborative distillation strategies for parameter-efficient language model deployment. arXiv preprint arXiv:2507.15198.

[39]Sun, Y., Zhang, R., Meng, R., Lian, L., Wang, H., & Quan, X. (2025). Fusion-based retrieval-augmented generation for complex question answering with LLMs.

[40]Fang, B., & Gao, D. (2025). Collaborative multi-agent reinforcement learning approach for elastic cloud resource scaling. arXiv preprint arXiv:2507.00550.

[41]Pan, R. (2024). Deep regression approach to predicting transmission time under dynamic network conditions. Journal of Computer Technology and Software, 3(8).

[42]Su, X. (2024). Forecasting asset returns with structured text factors and dynamic time windows. Transactions on Computational and Scientific Methods, 4(6).

[43]Gong, M. (2025). Modeling microservice access patterns with multi-head attention and service semantics. Journal of Computer Technology and Software, 4(6).

[44]Wu, Q. (2024). Internal knowledge adaptation in LLMs with consistency-constrained dynamic routing. Transactions on Computational and Scientific Methods, 4(5).