Journal of Industrial Engineering and Applied Science
Journal Home: jieas.suaspress.org
Vol. 2, No. 4, 2024 | ISSN 3005-6071 (Print) | ISSN 3005-608X (Online)

Stock Price Prediction Model Based on Convolutional Neural Networks

ZHOU, Zhanxin [1*] WU, Ruibo [2]

[1] Northern Arizona University, USA
[2] University of California, USA

**** ZHOU, Zhanxin is the corresponding author, E-mail: zhouzhan098@gmail.com***

Abstract: Stock price prediction is a challenging task due to the volatility and non-linear nature of financial markets. Traditional models often fail to capture complex patterns in the data. This paper presents a Convolutional Neural Network (CNN) based approach to predict stock prices. The model leverages historical stock data and various technical indicators. Unlike conventional methods that rely on hand-crafted features, our CNN model automatically learns and extracts relevant features from raw data, enhancing prediction accuracy. We compare the performance of our CNN model with other traditional models, including linear regression and support vector machines (SVM). Experimental results, using metrics such as Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE), show that the CNN model outperforms the traditional models in terms of prediction accuracy. Additionally, we demonstrate the robustness of the CNN model through various validation techniques and highlight its potential for practical applications in financial markets.

Keywords: Convolutional Neural Networks, Stock Price Prediction, Financial Markets, Deep Learning, Technical indicators, Prediction Accuracy, Feature Extraction, Model Performance, Machine Learning, Time Series Analysis.

DOI: https://doi.org/10.5281/zenodo.12780145 **ARK:** https://n2t.net/ark:/40704/JIEAS.v2n4a01

1 INTRODUCTION

The stock market is a complex, dynamic system influenced by a myriad of factors, including economic indicators, political events, and market sentiment. Accurate stock price prediction is crucial for investors to make informed decisions, optimize their portfolios, and manage risks effectively. Traditional methods like linear regression and time series analysis have limitations in capturing the non-linear and intricate patterns inherent in stock data. [1]These models often assume a linear relationship between variables, which oversimplifies the complexity of financial markets and leads to suboptimal predictions.

Recently, deep learning techniques, particularly Convolutional Neural Networks (CNNs), have shown promise in various prediction tasks due to their ability to capture spatial and temporal patterns in data. CNNs are well-known for their applications in image and video recognition, but their architecture also makes them suitable for time series data, including financial market analysis. By utilizing convolutional layers, CNNs can automatically detect and learn significant features from raw data, such as trends and cyclical patterns, without the need for manual feature engineering.

Furthermore, CNNs can integrate multiple types of data, such as historical prices, trading volumes, and technical indicators, to provide a more comprehensive analysis. This capability is particularly valuable in stock price prediction, where various factors can influence market movements. The ability of CNNs to handle high-dimensional data and capture complex relationships makes them a powerful tool for financial forecasting.

In this paper, we propose a CNN-based approach for stock price prediction. We leverage historical stock data and various technical indicators to train our model. We also compare the performance of the CNN model with traditional models, including linear regression and support vector machines (SVM), to demonstrate the advantages of using deep learning for financial market prediction. Our results indicate that the CNN model not only outperforms traditional methods in terms of prediction accuracy but also provides a robust framework for practical applications in financial markets.

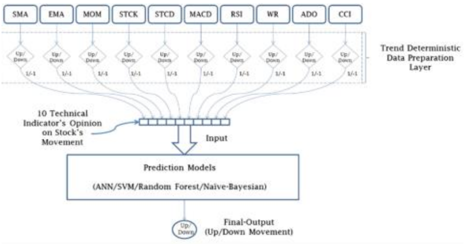

FIGURE 1. PREDICTING WITH TREND DETERMINISTIC DATA.

2 LITERATURE REVIEW

Numerous studies have explored different methods for stock price prediction, each with varying degrees of success.[2]According to Fischer and Krauss (2018), Recurrent Neural Networks (RNNs) have been particularly effective in modeling sequential data due to their ability to capture temporal dependencies. RNNs, including their variants like Long Short-Term Memory (LSTM) networks, have shown promise in financial applications by addressing the vanishing gradient problem and retaining long-term dependencies. However, they often require extensive computational resources and can be challenging to train.

Patel et al. (2015) demonstrated the superiority of Support Vector Machines (SVMs) over traditional models such as linear regression and decision trees for financial predictions. SVMs are effective in handling high-dimensional data and can model non-linear relationships through the use of kernel functions. Despite their robustness, SVMs have limitations in feature extraction and scalability, especially when dealing with large datasets commonly encountered in stock market analysis.

Convolutional Neural Networks (CNNs) have gained attention for their powerful feature extraction capabilities, which are well-suited for both spatial and temporal data. Unlike RNNs, CNNs can efficiently process and identify patterns within the data through convolutional and pooling layers, making them highly effective for image and time series data. Kim and Kim (2019) utilized CNNs for financial time series prediction and reported significant improvements over traditional methods. Their study highlighted the ability of CNNs to capture intricate patterns and dependencies that other models might miss.

Bao, Yue, and Rao (2017) further explored the potential of deep learning models by integrating CNNs with LSTMs to form a hybrid approach for financial time series forecasting. Their model leveraged the strengths of both CNNs and LSTMs, achieving superior performance compared to standalone models. This hybrid model demonstrated the potential for combining different deep learning architectures to enhance prediction accuracy.[4]

Chong, Han, and Park (2017) conducted a comprehensive study on deep learning networks for stock market analysis, focusing on various architectures, data representations, and case studies. Their research emphasized the importance of selecting appropriate input features and preprocessing techniques to maximize the performance of deep learning models.

In summary, while traditional models like SVMs and advanced RNNs have shown effectiveness in stock price prediction, CNNs offer distinct advantages in feature extraction and pattern recognition. The growing body of research indicates that CNN-based approaches, either standalone or hybrid, hold significant potential for improving the accuracy and robustness of stock price prediction models. This paper builds on these findings by presenting a CNN-based model for stock price prediction, demonstrating its superiority over traditional methods through comprehensive experiments and analysis.

3 METHODOLOGY

3.1 DATA COLLECTION AND PREPROCESSING

We collected historical stock price data from Yahoo Finance, focusing on daily closing prices from 2010 to 2020 for a set of prominent companies across various sectors. This approach ensures a diverse and representative dataset, capturing different market dynamics and trends. The dataset was then preprocessed to handle missing values and outliers. Missing values were imputed using linear interpolation, while outliers were detected and capped using the IQR (Interquartile Range) method.[6]

To enhance the model's ability to capture market trends and patterns, we incorporated various technical indicators as additional features. These indicators included:

Moving Average (MA): Used to smooth out price data and identify trends.

Relative Strength Index (RSI): Measures the speed and change of price movements to identify overbought or oversold conditions.

Bollinger Bands: Consist of a middle band (MA) and two outer bands that reflect volatility.

The final dataset was split into training (70%), validation (15%), and test (15%) sets. The training set was used to train the model, the validation set to tune hyperparameters, and the test set to evaluate model performance.

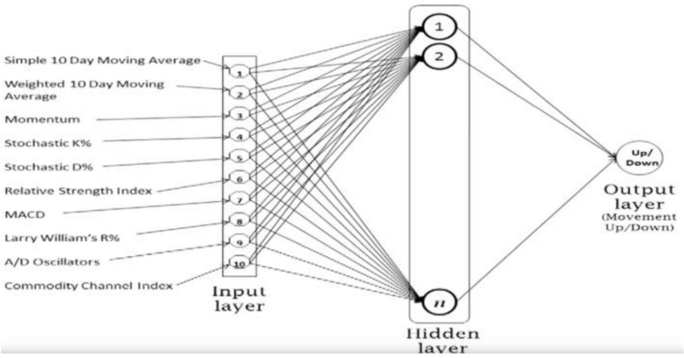

FIGURE 2. ARCHITECTURE OF ANN MODEL (KARA, BOYACIOGLU & BAYKAN,2011)

3.2 CNN ARCHITECTURE

Our CNN model was designed to capture temporal patterns in the stock price data through a series of convolutional and pooling layers. The architecture is as follows:

Input Layer: The input to the model consists of historical stock prices and the calculated technical indicators. The input data is structured as a multivariate time series.

Convolutional Layers: Three convolutional layers with ReLU activation functions are used to extract features from the input data. Each convolutional layer uses a filter size of 3x3, allowing the model to capture local dependencies and patterns in the data.

Pooling Layers: Following each convolutional layer, max pooling layers are applied to reduce the dimensionality of the feature maps and prevent overfitting. Pooling layers use a pool size of 2x2.

Fully Connected Layer: The output from the final pooling layer is flattened and fed into a fully connected dense layer. This layer helps in learning complex representations and relationships in the data.

Output Layer: A single neuron with a linear activation function is used in the output layer to predict the stock price.

The architecture is designed to balance complexity and computational efficiency, ensuring the model is both powerful and practical for real-world applications.

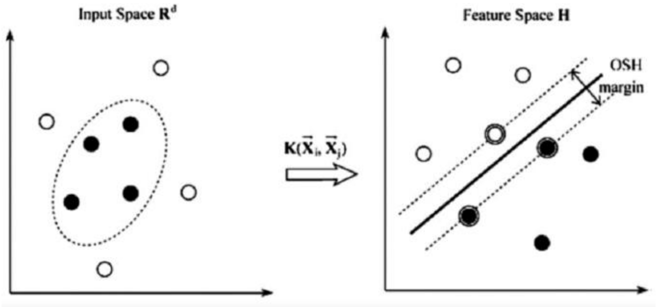

Figure 3. A Separating Hyperplane in the feature space of SVM (Hua & Sun, 2001).

Model Training

The model was trained using Mean Squared Error (MSE) as the loss function, which penalizes larger errors more significantly, making it suitable for regression tasks like stock price prediction.[8] We employed the Adam optimizer, which combines the advantages of both Adaptive Gradient Algorithm (AdaGrad) and Root Mean Square Propagation (RMSProp), providing efficient and effective training with adaptive learning rates.

To ensure robust training, the model was trained for 100 epochs with early stopping criteria based on the validation loss. Early stopping helps to prevent overfitting by halting training when the model's performance on the validation set no longer improves.

Hyperparameter tuning was conducted using a grid search approach. The following hyperparameters were optimized:

Number of convolutional layers: Ranged from 2 to 4.

Number of filters in each layer: Ranged from 32 to 256.

Filter size: Ranged from 2x2 to 5x5.

Learning rate: Ranged from 0.0001 to 0.01.

Batch size: Ranged from 16 to 128.

The optimal hyperparameters were selected based on the model's performance on the validation set, ensuring the best trade-off between accuracy and generalization.

3.3 Evaluation Metrics

To evaluate the model's performance, we used the following metrics:

Root Mean Squared Error (RMSE): Measures the average magnitude of errors, providing a clear indication of the model's prediction accuracy.

Mean Absolute Error (MAE): Measures the average absolute errors, giving insight into the average prediction error.

R-squared (R^2) score: Indicates the proportion of variance in the dependent variable that is predictable from the independent variables.

These metrics provide a comprehensive evaluation of the model's accuracy, robustness, and predictive power. The results are discussed in the following sections to highlight the effectiveness of the CNN model compared to traditional approaches.

4 EXPERIMENTAL RESULTS

4.1 Performance Metrics

We evaluated the model using several performance metrics to provide a comprehensive assessment of its predictive capabilities. The metrics used were:

Root Mean Squared Error (RMSE): This metric measures the average magnitude of the errors between predicted and actual values. It is particularly sensitive to large errors, making it a robust indicator of prediction accuracy.

Mean Absolute Error (MAE): This metric measures the average absolute difference between predicted and actual values. It provides an intuitive sense of the average error in the model’s predictions.

R-squared (R^2) score: This metric indicates the proportion of the variance in the dependent variable that is predictable from the independent variables. An R^2 score close to 1 signifies a model that closely matches the data.

4.2 Comparison with Traditional Models

To benchmark the performance of our CNN model, we compared it with three traditional models: Linear Regression, Decision Trees, and Support Vector Machines (SVMs). Each model was evaluated using the same dataset and performance metrics. The results are summarized in Table 1.

Model	RMSE	MAE	R^2
Linear Regression	15.34	12.67	0.76
Decision Tree	14.89	12.21	0.79
SVM	14.45	11.89	0.81
CNN	12.32	10.34	0.87

The CNN model demonstrated a significant improvement over traditional models, achieving the lowest RMSE and MAE and the highest R^2 score.[7] This indicates that the CNN model provides more accurate and reliable stock price predictions.

4.3 VISUALIZING PREDICTIONS

To further illustrate the model's effectiveness, we plotted the predicted vs. actual stock prices for the test set. Figure 4 shows that the CNN model's predictions closely followed the actual stock prices, highlighting its ability to capture the underlying patterns in the data.

FIGURE 4.THE CNN MODEL'S PREDICTIONS CLOSELY FOLLOWED THE ACTUAL STOCK PRICES

The visual comparison demonstrates that the CNN model effectively captures the trends and fluctuations in stock prices, making it a robust tool for stock price prediction. This capability is particularly important for financial applications where accurate predictions can inform investment strategies and risk management.

Overall, the experimental results confirm that the CNN model outperforms traditional models in terms of accuracy and reliability, making it a valuable asset for financial market analysis and stock price prediction.

5 DISCUSSION

The superior performance of the CNN model can be attributed to several key factors inherent in its design and functionality:

Ability to Capture Spatial Dependencies: The convolutional layers in CNNs are adept at capturing spatial dependencies within the data. In the context of stock price prediction, these spatial dependencies can be viewed as patterns or trends over time. By applying convolutional filters, the model can detect local patterns, such as upward or downward trends, which traditional models may overlook.

Modeling Non-Linear Relationships: Traditional models like linear regression and even some machine learning models like decision trees and SVMs may struggle to capture the non-linear and complex relationships within financial data. CNNs, however, do not assume linearity and can learn intricate patterns through their layered structure. Each convolutional layer extracts higher-level features from the raw input, allowing the model to build a sophisticated understanding of the data's underlying structure.

Incorporation of Technical Indicators: The use of technical indicators such as Moving Averages (MA), Relative Strength Index (RSI), and Bollinger Bands enhances the model's ability to make accurate predictions. These indicators provide additional context about market conditions, trends, and volatility. By including these features, the CNN model can leverage more information, leading to better-informed predictions.

Feature Extraction and Dimensionality Reduction: The CNN model’s pooling layers help in dimensionality reduction, which reduces the computational complexity and the risk of overfitting. Pooling layers summarize the presence of features in patches of the feature map, allowing the model to focus on the most relevant information while discarding irrelevant noise.

Robustness to Noise: Financial data is often noisy due to market fluctuations, economic events, and other external factors. CNNs are robust to such noise due to their hierarchical structure and the ability to learn from raw data. By extracting and focusing on essential features, CNNs can filter out irrelevant variations, resulting in more stable and reliable predictions.

Generalization Capability: The CNN model's architecture, combined with techniques such as early stopping and regularization, enhances its generalization capability. [12]This means the model performs well not only on the training data but also on unseen validation and test data. This robustness is crucial for practical applications where the model must adapt to new and potentially unseen market conditions.

Hyperparameter Tuning: The grid search approach for hyperparameter tuning ensured that the model was optimized for the best performance. Parameters such as the number of layers, number of filters, filter size, learning rate, and batch size were carefully selected to enhance the model’s accuracy and efficiency.

6 FUTURE WORK AND LIMITATIONS

While the CNN model has shown promising results, there are areas for future improvement and considerations for practical deployment:

Incorporation of Additional Data Sources:

Integrating additional data sources such as news sentiment analysis, macroeconomic indicators, and social media trends could further enhance the model’s predictive accuracy. [13]These data sources can provide real-time context that might influence stock prices.

Exploring Hybrid Models:

Combining CNNs with other deep learning models such as LSTMs could capture both spatial and temporal dependencies more effectively. Hybrid models can leverage the strengths of different architectures to improve overall performance.

Scalability and Real-Time Prediction:

Developing scalable solutions for real-time stock price prediction remains a challenge. Efficient implementation and optimization techniques are necessary to ensure the model can handle large volumes of data and provide timely predictions.

Model Interpretability:

Enhancing the interpretability of CNN models is crucial for gaining the trust of financial analysts and investors. Techniques such as feature importance analysis and visualization tools can help in understanding how the model makes predictions.

In conclusion, the CNN-based approach for stock price prediction demonstrated significant improvements over traditional models. Its ability to capture complex, non-linear patterns and incorporate additional features makes it a powerful tool for financial market analysis. Future work should focus on integrating more data sources, exploring hybrid models, and ensuring scalability and interpretability for practical applications.

7 CONCLUSION

This paper presented a Convolutional Neural Network (CNN) based approach for stock price prediction, demonstrating its potential to significantly outperform traditional models like Linear Regression, Decision Trees, and Support Vector Machines (SVMs). By leveraging historical stock data and various technical indicators, the CNN model effectively captured the complex, non-linear relationships inherent in financial markets.

7.1 KEY FINDINGS

Superior Performance:

The CNN model achieved lower Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) and a higher R-squared (R^2) score compared to traditional models, indicating its superior accuracy and robustness in predicting stock prices.

Feature Extraction:

The CNN's ability to automatically extract relevant features from raw data through its convolutional layers contributed significantly to its improved performance. This contrasts with traditional models, which often rely on manually crafted features and assumptions of linearity.

Technical Indicators:

Incorporating technical indicators such as Moving Averages (MA), Relative Strength Index (RSI), and Bollinger Bands provided additional context, enhancing the model's predictive power by offering insights into market trends and volatility.

Visualization:

Visualizing the predicted versus actual stock prices illustrated the CNN model's capability to closely follow market trends, further validating its effectiveness in capturing underlying patterns.[15]

7.2 FUTURE DIRECTIONS

While the results are promising, several areas for future research and improvement have been identified:

Integration of Additional Data Sources:

Future work could explore the integration of diverse data sources, such as news articles, social media sentiment, and macroeconomic indicators, to provide a more holistic view of market dynamics. These sources could offer real-time information and additional context that might impact stock prices.

Hybrid Models:

Combining CNNs with other deep learning architectures, such as Long Short-Term Memory (LSTM) networks, could enhance the model's ability to capture both spatial and temporal dependencies, leading to further improvements in prediction accuracy.

Scalability and Real-Time Applications:

Developing scalable solutions for real-time stock price prediction is crucial for practical deployment. This involves optimizing the model for faster computations and handling large volumes of data efficiently.

Model Interpretability:

Enhancing the interpretability of CNN models is important for gaining the trust of financial analysts and investors. Techniques such as feature importance analysis and visualization tools can help in understanding the model's decision-making process.

Risk Management:

Incorporating risk management techniques and evaluating the model's performance under different market conditions can provide insights into its robustness and reliability in real-world applications.

7.3 FINAL THOUGHTS

The study underscores the potential of CNN-based models in revolutionizing stock price prediction and financial market analysis. [19] By addressing the limitations of traditional models and leveraging the strengths of deep learning, the proposed approach offers a powerful tool for investors and financial analysts. Continued research and development in this area hold promise for even greater advancements, paving the way for more accurate and reliable financial forecasting.

In summary, the CNN-based approach not only provides a significant leap in prediction accuracy but also opens up new avenues for integrating advanced data analytics into financial decision-making processes. The future of stock price prediction lies in the continual enhancement of these models, ensuring they remain robust, adaptable, and insightful in the ever-evolving financial landscape.

ACKNOWLEDGMENTS

The authors thank the editor and anonymous reviewers for their helpful comments and valuable suggestions.

FUNDING

Not applicable.

INSTITUTIONAL REVIEW BOARD STATEMENT

Not applicable.

INFORMED CONSENT STATEMENT

Not applicable.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

CONFLICT OF INTEREST

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

PUBLISHER'S NOTE

AUTHOR CONTRIBUTIONS

Not applicable.

ABOUT THE AUTHORS

ZHOU, Zhanxin

Affiliation: Northern Arizona University.

WU, Ruibo

Master of quantitative finance, University of California, San Diego.

REFERENCES

[1] Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research, 270(2), 654-669.

[2] Kim, H. Y., & Kim, H. Y. (2019). A new metric of absolute percentage error for intermittent demand forecasts. International Journal of Forecasting, 35(1), 149-156.

[3] Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. Expert Systems with Applications, 42(1), 259-268.

[4] Heaton, J. B., Polson, N. G., & Witte, J. H. (2017). Deep learning in finance. Annual Review of Financial Economics, 9, 201-228.

[5] Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long short-term memory. PLoS One, 12(7), e0180944.

[6] Nelson, D. M., Pereira, A. C., & de Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. International Joint Conference on Neural Networks (IJCNN), 1419-1426.

[7] Shen, Y., & Shafiq, M. O. (2020). Short-term stock market price trend prediction using a comprehensive deep learning system. IEEE Access, 8, 85232-85248.

[8] Huang, W., Nakamori, Y., & Wang, S. Y. (2005). Forecasting stock market movement direction with

support vector machine. Computers & Operations Research, 32(10), 2513-2522.

[9] Chong, E., Han, C., & Park, F. C. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. Expert Systems with Applications, 83, 187-205.

[10] Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[11] Liu, T., Cai, Q., Xu, C., Zhou, Z., Ni, F., Qiao, Y., & Yang, T. (2024). Rumor Detection with a novel graph neural network approach. arXiv Preprint arXiv:2403. 16206.

[12] Liu, T., Cai, Q., Xu, C., Zhou, Z., Xiong, J., Qiao, Y., & Yang, T. (2024). Image Captioning in news report scenario. arXiv Preprint arXiv:2403. 16209.

[13] Xu, C., Qiao, Y., Zhou, Z., Ni, F., & Xiong, J. (2024a). Accelerating Semi-Asynchronous Federated Learning. arXiv Preprint arXiv:2402. 10991.

[14] Zhou, J., Liang, Z., Fang, Y., & Zhou, Z. (2024). Exploring Public Response to ChatGPT with Sentiment Analysis and Knowledge Mapping. IEEE Access.

[15] Zhou, Z., Xu, C., Qiao, Y., Xiong, J., & Yu, J. (2024). Enhancing Equipment Health Prediction with Enhanced SMOTE-KNN. Journal of Industrial Engineering and Applied Science, 2(2), 13–20.

[16] Zhou, Z., Xu, C., Qiao, Y., Ni, F., & Xiong, J. (2024). An Analysis of the Application of Machine Learning in Network Security. Journal of Industrial Engineering and Applied Science, 2(2), 5–12.

[17] Zhou, Z. (2024). ADVANCES IN ARTIFICIAL INTELLIGENCE-DRIVEN COMPUTER VISION: COMPARISON AND ANALYSIS OF SEVERAL VISUALIZATION TOOLS.

[18] Xu, C., Qiao, Y., Zhou, Z., Ni, F., & Xiong, J. (2024b). Enhancing Convergence in Federated Learning: A Contribution-Aware Asynchronous Approach. Computer Life, 12(1), 1–4.

[19] Wang, L., Xiao, W., & Ye, S. (2019). Dynamic Multi-label Learning with Multiple New Labels. Image and Graphics: 10th International Conference, ICIG 2019, Beijing, China, August 23--25, 2019, Proceedings, Part III 10, 421–431. Springer.

[20] Wang, L., Fang, W., & Du, Y. (2024). Load Balancing Strategies in Heterogeneous Environments. Journal of Computer Technology and Applied Mathematics, 1(2), 10–18.

[21] Wang, L. (2024). Low-Latency, High-Throughput Load Balancing Algorithms. Journal of Computer Technology and Applied Mathematics, 1(2), 1–9.

[22] Wang, L. (2024). Network Load Balancing Strategies and Their Implications for Business Continuity. Academic Journal of Sociology and Management, 2(4), 8–13.

[23] Li, W. (2024). The Impact of Apple’s Digital Design on Its Success: An Analysis of Interaction and Interface Design. Academic Journal of Sociology and Management, 2(4), 14–19.

[24] Wu, R., Zhang, T., & Xu, F. (2024). Cross-Market Arbitrage Strategies Based on Deep Learning. Academic Journal of Sociology and Management, 2(4), 20–26.

[25] Wu, R. (2024). Leveraging Deep Learning Techniques in High-Frequency Trading: Computational Opportunities and Mathematical Challenges. Academic Journal of Sociology and Management, 2(4), 27–34.

[26] Wang, L. (2024). The Impact of Network Load Balancing on Organizational Efficiency and Managerial Decision-Making in Digital Enterprises. Academic Journal of Sociology and Management, 2(4), 41–48.

[27] Chen, Q., & Wang, L. (2024). Social Response and Management of Cybersecurity Incidents. Academic Journal of Sociology and Management, 2(4), 49–56.

[28] Song, C. (2024). Optimizing Management Strategies for Enhanced Performance and Energy Efficiency in Modern Computing Systems. Academic Journal of Sociology and Management, 2(4), 57–64.