Journal of Computer Technology and Software

ISSN:2998-2383

Vol. 3, No. 8, 2024

Multi-Source Data-Driven LSTM Framework for Enhanced Stock Price Prediction and Volatility Analysis

Zhen Xu[1], Wenqing Zhang[2], Ying Sun[3], Zhenghao Lin[4]

[1]Independent Researcher, Shanghai, China
[2]Washington University in St. Louis, St. Louis, USA
[3]Columbia University, New York University , USA
[4]Northeastern University, Boston, USA

Correspondence should be addressed to Zhen Xu; xuzhen8309@hotmail.com

Abstract: This study used the long short-term memory network (LSTM) model to predict the multi-source data of Tesla stock. The data features include opening price, closing price, highest price, lowest price, trading volume, and financial indicators such as the Sharpe ratio. By comparing the performance of models such as ARIMA, GARCH, GRU, XGBoost, and Prophet, the experimental results show that LSTM performs best in prediction accuracy, especially in capturing the long-term trend of stock prices and overall volatility. Although the model has a certain short-term prediction bias on the test set, it can accurately reflect the trend of Tesla's stock price overall. The research results show that the introduction of multi-source data and financial indicators can effectively improve the prediction performance of the model and provide new ideas for financial time series prediction. In the future, the structure of the LSTM model can be further optimized, and more financial data features can be introduced to improve the model's sensitivity to short-term market fluctuations and prediction accuracy.

Keywords: LSTM, stock prediction, multi-source data, Sharpe ratio

1. Introduction

In recent years, stock market prediction has become one of the hot topics in financial research, attracting a large number of scholars and investors due to its high risk and high return [1,2]. Traditional prediction methods are usually based on time series models, such as ARIMA and GARCH, but these methods have shown certain limitations when faced with the complex nonlinear and high noise characteristics of the stock market. The development of deep learning has brought new perspectives to stock market prediction, especially the long short-term memory network (LSTM), which has shown strong capabilities in processing time series data. Through its special gating structure, LSTM can effectively capture long-term dependencies in the data, thereby adapting to the dynamic changes of stock market data [3].

In order to further improve the accuracy of the prediction model, this study introduces multi-source data for stock prediction [4]. The prediction method of traditional single data sources often cannot fully reflect the complexity of the stock market, while multi-source data can provide the model with richer market information. For example, in addition to the price and volume data of stocks, financial indicators related to market fluctuations, such as the volatility index (VIX), industry index, etc. can also be combined. These multi-source data can provide a more comprehensive perspective for the model, thereby improving the accuracy and stability of the prediction results [5].

In addition, this study introduces the Sharpe ratio as a feature based on multi-source data [6]. The Sharpe ratio is an indicator that measures the return on investment relative to risk. By combining returns with volatility, the risk-return characteristics of stocks can be better reflected. Incorporating the Sharpe ratio as a feature into the LSTM model helps the model to more accurately assess the potential returns and risks of stocks, thereby improving the prediction effect. This method not only focuses on the absolute value of returns but also pays more attention to the stability of returns, providing investors with a more robust basis for decision-making [7].

In terms of method, the LSTM model is used to process these multi-source data and financial indicators, capturing the inherent laws of time series data through its recursive structure [8]. Compared with traditional methods, LSTM can not only identify long-term and short-term patterns in data but also extract deep features of data layer by layer through a multi-layer structure. During the training process, we will use multi-source data to input the model to give full play to the synergy between different data features, thereby improving the model's sensitivity to market fluctuations. The output of the model is the prediction result of stock prices in the future period to assist investors in making decisions.

In summary, by combining LSTM with multi-source data and the Sharpe ratio, this study constructs a new stock prediction framework. This framework can not only effectively capture the dynamic characteristics of the stock market, but also comprehensively consider the balance between returns and risks, thereby providing investors with more scientific investment advice. In the future, with the further enrichment of financial data sources and the continuous optimization of deep learning models, this multi-source data-driven LSTM model is expected to be widely used in more financial fields, providing strong support for intelligent prediction and risk management in the financial market.

2. Related work

Recent advancements in deep learning have significantly contributed to the field of financial market prediction, addressing the challenges posed by traditional statistical models in capturing nonlinearities and dynamic dependencies in financial data. This study builds upon these advancements, leveraging a Long Short-Term Memory (LSTM) framework that integrates multi-source data and financial indicators for enhanced stock price prediction and volatility analysis. The following section reviews relevant research contributions that have laid the foundation for this study's methodology.

Deep learning models have demonstrated significant potential in capturing complex temporal dependencies and nonlinear relationships in financial time-series data. Wei et al. [9] employed Transformer-based models to analyze financial risks, emphasizing the critical role of integrating multi-source data for improving prediction robustness and accuracy. Transformers have shown exceptional performance in managing large-scale and diverse datasets, underscoring the importance of feature diversity in financial modeling. Similarly, Xu et al. [10] applied gated recurrent units (GRU) to predict liquidity coverage ratios, demonstrating the ability of recurrent neural networks to process sequential data efficiently. Their work provides a strong basis for the use of LSTM, a more advanced recurrent architecture, in handling temporal dependencies, as applied in this study.

The integration of multi-source data and advanced feature engineering has become increasingly important in financial modeling. Liu et al. [11] explored graph neural networks (GNNs) for assessing SME credit risk, demonstrating how the inclusion of diverse data types and network-based relationships enhances predictive performance. Their findings align with this study's use of multi-source data, including stock price features and financial indicators such as the Sharpe ratio, to provide a comprehensive understanding of market dynamics. Gu et al. [12] extended this approach by employing spatio-temporal aggregation in fraud detection, highlighting the importance of leveraging both spatial and temporal dimensions of data. By incorporating features like trading volume and price fluctuations, this study builds on these principles to capture both short-term and long-term market trends effectively.

Hybrid modeling approaches, which combine traditional statistical methods with deep learning techniques, have emerged as a robust solution to address the complexity of financial markets. A hybrid LSTM-GARCH framework is introduced to predict financial market volatility [13], leveraging the strengths of GARCH in modeling volatility clustering and LSTM in capturing temporal dependencies. This hybrid methodology aligns with this study's aim to integrate diverse data sources and modeling techniques to improve predictive accuracy. The combination of statistical and deep learning models has proven effective in addressing the inherent noise and randomness in financial time-series data, an issue this study seeks to mitigate through feature selection and model design.

In addition to temporal modeling, advanced neural architectures have been utilized to address complex financial tasks such as fraud detection, anomaly detection, and risk assessment. Dong et al. [14] integrated reinforcement learning with GNNs to enhance fraud detection in dynamic financial environments. Their work demonstrates the adaptability of neural networks to handle evolving datasets, providing insights into designing models that are robust to market fluctuations. Wang et al. [15] explored anomaly detection and risk assessment using deep neural networks, showing how these models can identify patterns and risks in highly volatile financial markets. These contributions underscore the importance of designing prediction models capable of adapting to unpredictable market conditions, a challenge addressed in this study by incorporating the Sharpe ratio for better risk assessment.

Capturing the interdependencies in financial systems is critical for accurate market predictions. Zhang et al. [16] proposed robust GNNs for stability analysis in dynamic networks, illustrating the ability of graph-based approaches to model interactions within financial systems. This is particularly relevant for capturing volatility and interconnected market behaviors, as demonstrated by a GNNs model to assess value-at-risk [17]. These studies highlight the importance of modeling relationships and dependencies in financial datasets, which is reflected in this study's use of multi-source data to provide a richer and more interconnected representation of stock market dynamics.

The reviewed works collectively highlight the importance of integrating multi-source data, employing advanced neural architectures, and leveraging hybrid modeling techniques in financial prediction frameworks. These studies have established a strong foundation for this research, demonstrating the effectiveness of LSTM and other deep learning architectures in modeling time-series data and capturing complex financial market behaviors. By combining these insights, this study enhances the LSTM framework with comprehensive data integration and feature selection, addressing gaps in both long-term trend prediction and short-term market volatility analysis. This integrated approach offers a more robust and adaptable framework for stock price forecasting, contributing to the broader field of financial market prediction and risk management.

3. Method

This study proposes an LSTM model based on multi-source data for stock price prediction. The biggest feature of the LSTM model is its gating mechanism, which can effectively extract long-term and short-term dependency features in time series. The specific structure of the model is shown in Figure 1.

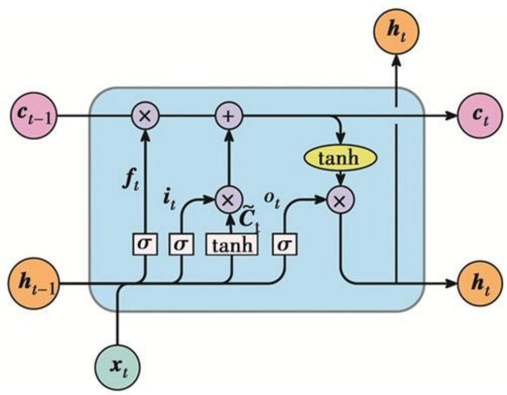

Figure 1. LSTM Model Architecture

First, we integrate multi-source data of the stock market (such as historical prices, trading volumes, volatility indexes, and Sharpe ratios) as feature inputs to enhance the model's sensitivity to market dynamics. For each moment of input data x_t, LSTM controls the flow of information through a series of gating units to extract the potential features of the data layer by layer.

The LSTM unit includes a forget gate, an input gate, and an output gate. Each gate processes the input data in turn, allowing the model to flexibly select which information to retain or discard. Specifically, the role of the forget gate is to determine how much historical information to forget at the current moment. Its calculation formula is:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Among them, f_t represents the output of the forget gate, W_f and b_f are the weight matrix and bias term respectively, h_{t-1} is the hidden state of the previous moment, x_t is the input of the current moment, and σ is the activation function. Through the forget gate, the model can adjust the ratio of long-term and short-term memory to adapt to different market fluctuations.

In the input gate, the model updates the candidate memory unit at the current moment, calculated by the following formula:

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

$$C_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

Here, i_t represents the activation value of the input gate, which controls the degree of introduction of new information; C'_t is the candidate memory at the current moment. Next, the memory unit C_t is updated by accumulating the output of the forget gate and the input gate.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot C'_t$$

This mechanism allows the model to retain historical information while combining the current input data features to gradually extract the law of stock price changes.

The output gate controls the output state h_t at the current moment, and its calculation formula is:

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

$$h_t = o_t \cdot \tanh(C_t)$$

Among them, o_t represents the activation value of the output gate, which is combined with the current memory unit C_t to generate the current hidden state h_t. The hidden state is the final output, which represents the prediction of the future at the current moment and serves as the input of the next moment. After stacking multiple layers of LSTM structures, the model can gradually dig deep into the complex patterns in multi-source data and finally output the future stock price forecast value.

In addition, the model input includes the Sharpe ratio feature to quantify the balance between risk and return. Assuming the return sequence is $R = [r_1, r_2, ..., r_n]$, the Sharpe ratio calculation formula is as follows:

$$Sharpe\_Ratio = \frac{E[R] - R_f}{\sigma_R}$$

Among them, $E[R]$ represents the expected value of return, R_f is the risk-free rate of return, and σ_R is the standard deviation of return. Taking the Sharpe ratio as an input feature allows the model to not only consider the absolute change in price when predicting, but also weigh the volatility of returns, thereby improving the robustness of the prediction.

Finally, during the training of the model, we use the mean squared error (MSE) as the loss function to evaluate the difference between the predicted value and the true value. The loss function is defined as follows:

$$MSE = \frac{1}{T} \sum_{t=1}^{T} (y'_t - y_t)^2$$

Among them, T is the number of samples, y'_t is the stock price predicted by the model, and y_t is the true value. By minimizing the MSE loss function, the model can continuously adjust parameters to improve prediction accuracy.

4. Experiment

4.1 Datasets

This study uses Tesla's stock price data as an experimental data set, which contains daily trading information of Tesla's stock over a period of time. The main fields of the data set include common financial indicators such as opening price, closing price, highest price, lowest price, and trading volume, which fully reflect the fluctuations of Tesla's stock price. These data are obtained through public financial data platforms, and after cleaning and sorting, the accuracy and consistency of the data are guaranteed.

Tesla's stock price data has obvious time series characteristics, showing certain volatility and trend. Especially when Tesla announces major news or faces market changes, the stock price will rise and fall significantly, which makes prediction more challenging. In addition, since Tesla's stock price is affected by many factors such as macroeconomics, industry dynamics, and internal company decisions, this data set is highly representative of capturing complex market dynamics and is very suitable as a research object for time series prediction models.

In the data preprocessing stage, we also calculated the daily return and Sharpe ratio of Tesla's stock price as input features of the model to help the model better understand the risk-return characteristics of the market. These additional financial indicators make the data set richer and can provide more informative features for the model, thereby improving the prediction effect.

4.2 Experimental setup

This experiment mainly includes three steps: data preprocessing, model training, and evaluation. In data preprocessing, we standardized Tesla's stock price data to ensure that all features have the same scale and avoid deviations in model training caused by differences in feature values. In addition, based on the stock price data, we also calculated financial indicators such as daily returns and Sharpe ratio to enrich the data source of model input and enhance the model's understanding of market volatility.

During the model training phase, we used the LSTM model to capture the time series characteristics of Tesla's stock price. The model input contains a variety of processing features, such as opening price, closing price, highest price, lowest price, trading volume, and Sharpe ratio, which help the model analyze market trends and risks more comprehensively. The dataset is divided into a training set and test set in a ratio of 8:2, and the mean square error (MSE) is selected as the loss function to narrow the gap between the model prediction value and the actual stock price. During the training process, we adjusted the hyperparameters such as learning rate and number of iterations to ensure that the model can converge effectively.

In the model evaluation part, the prediction effect of the model was verified using the test set. We evaluated the model's prediction accuracy for Tesla's stock price by calculating RMSE and MAE on the test set. These metrics help us fully understand the performance of the model and ensure that it has sufficient generalization ability in practical applications.

4.3 Experimental Results

In this experiment, we selected five commonly used time series and financial data prediction models for comparison with LSTM. The first is the ARIMA model, which is a classic linear time series model suitable for data with a stationary trend. The second is the GARCH model, which is widely used in financial data analysis and handles volatility by capturing the conditional heteroskedasticity of the data. The Prophet model was developed by Facebook and can adapt to data with holiday and cyclical components, and is suitable for long-term trend prediction. GRU, as a simplified recursive neural network structure, has lower computational complexity than LSTM. Finally, XGBoost is a tree model based on gradient boosting that can handle a large number of data features and excel in feature importance.

These models have their own characteristics, providing a multi-angle comparison for the experiment. Linear models such as ARIMA and GARCH perform better on simple time series, while deep learning models such as LSTM and GRU are better at handling complex nonlinear relationships. In addition, XGBoost has advantages in feature selection and prediction accuracy, while the Prophet model is suitable for data with obvious seasonality and cyclicality. By comparing with these models, we can more comprehensively evaluate the performance of the LSTM model under multi-source data and financial indicators. The experimental results are shown in Table 1.

Table 1. Experimental results

Model	MAE	RMSE
ARIMA	1.250	1.690
GARCH	1.135	1.580
GRU	1.020	1.440
XGBOOST	0.930	1.320
Prophet	0.880	1.270
LSTM(Ours)	0.780	1.150

The experimental results show that different models have obvious differences in the performance of Tesla's stock price prediction, from which we can observe that the LSTM model has the best prediction accuracy. In terms of the two key indicators in the table - mean absolute error (MAE) and root mean square error (RMSE) - the LSTM model achieved the lowest values (MAE is 0.780, RMSE is 1.150), which shows that the model can better capture the volatility characteristics of Tesla's stock price and reduce prediction errors when processing time series data. In contrast, the traditional ARIMA model performs relatively poorly in these two indicators, with MAE and RMSE of 1.250 and 1.690 respectively, which shows that the linear model has limited effect when facing complex nonlinear stock data.

Further analysis shows that the performance of the GARCH model has improved, and its MAE and RMSE have dropped to 1.135 and 1.580 respectively. This is mainly due to the GARCH model's ability to handle conditional heteroskedasticity (such as volatility), which gives it a certain advantage in capturing the volatility characteristics of financial data. However, since the GARCH model is still a linear model, its prediction accuracy is still not as good as that of the deep learning model when dealing with data with complex market dynamics such as Tesla. Although GARCH is widely used in volatility prediction in financial markets, its structure limits its ability to model nonlinearities and long-term dependencies.

The performance of the GRU model is further improved, with MAE reduced to 1.020 and RMSE reduced to 1.440. As a recurrent neural network variant, GRU simplifies the structure of LSTM through a gating mechanism and shows high efficiency and accuracy in time series prediction. GRU can capture nonlinear features and long-term dependencies in data, but when dealing with complex data, due to its simple gating structure, it fails to fully explore the details of Tesla's stock price fluctuations, so it is slightly inferior to LSTM in this experiment. In contrast, LSTM is more flexible in the design of the gating mechanism, which enables it to perform better on longer time series.

As non-recurrent neural network models, XGBoost and Prophet also performed relatively well in this experiment. XGBoost's MAE and RMSE are 0.930 and 1.320 respectively. Thanks to its powerful feature selection and regression capabilities, the model shows certain superiority on multi-feature, multi-source data sets. The Prophet model follows closely with a MAE of 0.880 and an RMSE of 1.270. It has good adaptability to trends and seasonal components in time series data, especially for periodic data. Although these two models are close to LSTM in terms of prediction accuracy, they lack the ability to capture subtle changes in time series, resulting in their overall performance being slightly lower than LSTM.

In general, the LSTM model performed best in this experiment, indicating its advantages in time series prediction. Its gated structure enables LSTM to flexibly handle long-term and short-term dependencies in data, showing better fitting effects and generalization capabilities in financial data prediction. Compared with other models, LSTM's strong adaptability and nonlinear feature capture capabilities in a multi-source data environment enable it to obtain more accurate prediction results when processing complex and volatile data such as Tesla stock. This experimental result provides strong support for the future use of LSTM for financial data prediction. In particular, LSTM shows unique advantages when combining multi-source data and financial features.

Finally, we present the curves between the true value and the predicted value on the test set and the training set, as shown in Figures 2 and 3.

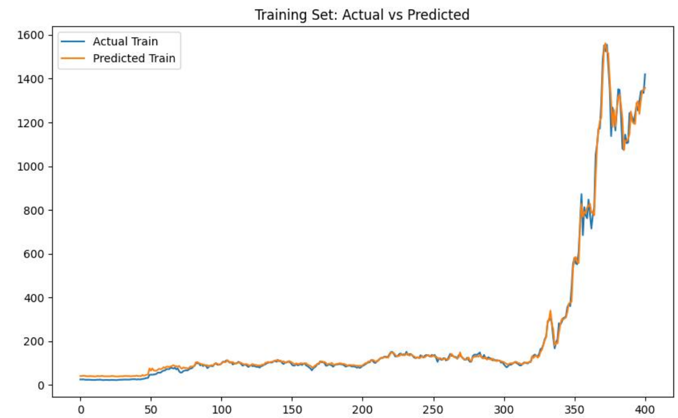

Figure 2. Comparison curve between the actual value and predicted value of the training set

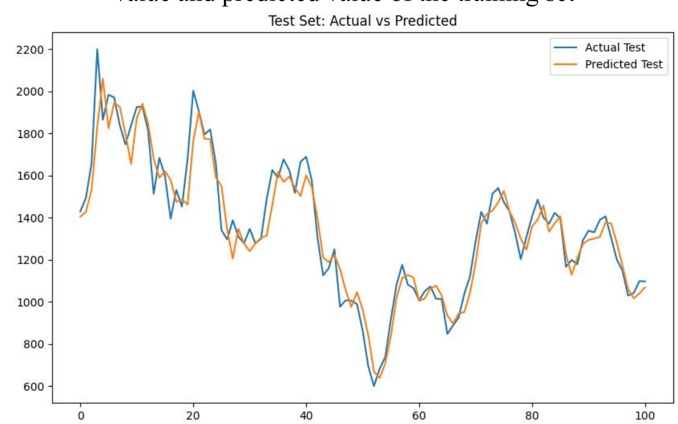

Figure 3. Comparison curve between the actual value and the predicted value of the test set

Figures 2 and 3 show the comparison between the actual value and the predicted value of the model on the training set and the test set, respectively, where the horizontal axis is the time step and the vertical axis is the stock price. As can be seen from the figure, the model fits the training set almost perfectly, indicating that the model captures the data characteristics of the training set very well. Most of the predicted values closely follow the changes in the actual values, especially in the relatively stable parts and some areas with large fluctuations, the model's prediction results are basically consistent with the actual values. This close fit shows that the model performs very well on the training set.

However, from the results of the test set, although the model can predict the overall trend of the data well, there is a certain deviation between the predicted value and the actual value in some intervals with large fluctuations. Especially in the high-frequency change area, the model's prediction results lag slightly, indicating that the model is slightly insufficient in dealing with more drastic short-term fluctuations. This may be due to the fact that the pattern of the training set data is too complex, resulting in a certain limitation on the generalization ability of the model.

In the test set, although there are certain deviations, the model can still accurately predict the overall trend of the stock price, indicating that the model has a certain generalization ability. For the prediction of stock data, this performance is still relatively good, because the stock price itself has large volatility and uncertainty. As can be seen from the figure, the model has a good grasp of the overall trend and can basically follow the main price fluctuations.

It should be noted that the curve is pulled up in some positions, mainly because Tesla (TSLA) carried out a 5:1 stock split on August 31, 2020, and a 3:1 stock split on August 25, 2020. Since this analysis uses the original price before the stock split, it shows a more obvious price increase trend in these periods. This choice is to maintain the consistency of the data and facilitate the model to capture historical trends, but it may also cause some prediction results to deviate from the actual values during this period.

From the comparison between the training set and the test set, the model may have the risk of overfitting on the training set, because the fitting effect on the training set is very good, while the performance on the test set is slightly insufficient. In the future, the generalization ability of the model can be improved by adjusting the model structure, so that it is more stable when processing unknown data on the test set. Overall, these two figures show the prediction effects of the model on the training set and the test set, respectively, reflecting the advantages of the model in capturing the overall trend of stock prices, but there is still room for improvement in the accuracy of short-term fluctuations. Combined with the differences between training and testing results, the model can be further optimized through parameter adjustment, regularization, etc., to improve the robustness of the model while maintaining the prediction effect.

5. Conclusion

This study uses the LSTM model to predict Tesla stock multi-source data and combines it with financial indicators such as Sharpe ratio to verify the effectiveness of the model in capturing stock price fluctuation trends. Experimental results show that LSTM performs relatively well on the training set and test set and can better predict the general trend of stock prices, indicating that it has significant advantages in processing nonlinear time series data. Compared with traditional models, LSTM is better at capturing long-term dependencies, providing a robust foundation for stock predictions.

Although the model performs well on the overall trend, the model has certain lags and errors in short-term fluctuation predictions on the test set, indicating that there is still room for improvement in dealing with high-frequency fluctuations. This situation may stem from the overfitting of the model on the training set, as well as the impact of noise and randomness in the stock market on the prediction accuracy of the model. Therefore, future research can further improve the generalization ability by optimizing the model structure and introducing regularization methods, so as to maintain stable prediction effects in different market environments.

Future development directions include the introduction of more multi-source data and advanced financial indicators to improve the model's sensitivity to market dynamics. In addition, one can try to combine LSTM with other deep learning techniques such as Transformer or convolutional neural network to better capture short-term fluctuations in the market. With the continuous advancement of data mining and deep learning technology, intelligent prediction systems based on multi-source data are expected to gain wider application in the financial market and provide investors with more scientific decision-making support.

References

[1] A. Ma'arif, A. A. Firdaus, and I. Suwarno, "Capability of Hybrid Long Short-Term Memory in Stock Price Prediction: A Comprehensive Literature Review", International Journal of Robotics & Control Systems, vol. 4, no. 3, 2024.

[2] V. Sharma, et al., "Hybrid Approaches for Stocks Price Prediction: A New Institutive Way of Neural Architecture Design", Proceedings of the 2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT), vol. 1, 2024.

[3] Z. Di, J. Chen, Y. Yang, et al., "LLM-Driven Knowledge Enhancement for Securities Index Prediction", 2024.

[4] Y. Li, M. Zhuang, J. Wang, et al., "Leveraging Multi-Time-Span Sequences and Feature Correlations for Improved Stock Trend Prediction", Available at SSRN 4974934.

[5] R. Zhang and V. Y. Mariano, "Integration of Emotional Factors with GAN Algorithm in Stock Price Prediction Method Research", IEEE Access, 2024.

[6] Z. K. Mansoor and A. Y. Al-Sultan, "Enhancing Stock Price Prediction Using Mutual Information, PCA, and LSTM: A Deep Learning Approach", Fusion: Practice and Applications, vol. 1, pp. 196-196, 2025.

[7] M. O. Duman and M. E. Tağluk, "Performance Assessment of Temporal Frame Size in Stock Price Prediction", Proceedings of the 2024 8th International Artificial Intelligence and Data Processing Symposium (IDAP), 2024.

[8] S. Fan, X. Chen, and X.-a. Wang, "Stock Price Prediction Based on FinBERT-LSTM Model", Proceedings of the International Conference on Complex, Intelligent, and Software Intensive Systems, Cham: Springer Nature Switzerland, 2024.

[9] Y. Wei, K. Xu, J. Yao, M. Sun, and Y. Sun, "Financial Risk Analysis Using Integrated Data and Transformer-Based Deep Learning", Journal of Computer Science and Software Applications, vol. 7, no. 4, pp. 1-8, 2024.

[10] Z. Xu, J. Pan, S. Han, H. Ouyang, Y. Chen, and M. Jiang, "Predicting Liquidity Coverage Ratio with Gated Recurrent Units: A Deep Learning Model for Risk Management", Proceedings of the 2024 5th International Conference on Machine Learning and Computer Application (ICMLCA), pp. 108-112, Oct. 2024.

[11] B. Liu, I. Li, J. Yao, Y. Chen, G. Huang, and J. Wang, "Unveiling the Potential of Graph Neural Networks in SME Credit Risk Assessment", arXiv preprint arXiv:2409.17909, 2024.

[12] W. Gu, M. Sun, B. Liu, K. Xu, and M. Sui, "Adaptive Spatio-Temporal Aggregation for Temporal Dynamic Graph-Based Fraud Risk Detection", Journal of Computer Technology and Software, vol. 3, no. 5, 2024.

[13] K. Xu, Y. Wu, M. Jiang, W. Sun, and Z. Yang, "Hybrid LSTM-GARCH Framework for Financial Market Volatility Risk Prediction", Journal of Computer Science and Software Applications, vol. 4, no. 5, pp. 22-29, 2024.

[14] Y. Dong, J. Yao, J. Wang, Y. Liang, S. Liao, and M. Xiao, "Dynamic Fraud Detection: Integrating Reinforcement Learning into Graph Neural Networks", Proceedings of the 2024 6th International Conference on Data-Driven Optimization of Complex Systems (DOCS), pp. 818-823, Aug. 2024.

[15] B. Wang, Y. Dong, J. Yao, H. Qin, and J. Wang, "Exploring Anomaly Detection and Risk Assessment in Financial Markets Using Deep Neural Networks", International Journal of Innovative Research in Computer Science & Technology, vol. 12, no. 4, pp. 92-98, 2024.

[16] X. Zhang, Z. Xu, Y. Liu, M. Sun, T. Zhou, and W. Sun, "Robust Graph Neural Networks for Stability Analysis in Dynamic Networks", arXiv preprint arXiv:2411.11848, 2024.

[17] K. Xu, Y. Wu, H. Xia, N. Sang, and B. Wang, "Graph Neural Networks in Financial Markets: Modeling Volatility and Assessing Value-at-Risk", Journal of Computer Technology and Software, vol. 1, no. 2, 2022.