Journal of Computer Technology and Software

ISSN:2998-2383

Vol. 1, No. 2, 2022

Graph Neural Networks in Financial Markets: Modeling Volatility and Assessing Value-at-Risk

Ke Xu[1], You Wu[2], Haohao Xia[3] , Ningjing Sang[4], Bingxing Wang[5]

[1]Columbia University, New York, USA
[2]College of William & Mary, Williamsburg, USA
[3]University of Southern California, Los Angeles, USA
[4]Columbia University, New York, USA
[5]Illinois Institute of Technology, Chicago, USA

Correspondence should be addressed to **Bingxing Wang**; wanguo0528@gmail.com

Abstract: This paper explores the application of graph neural networks (GNN) in the prediction of volatility and value-at-risk (VaR) calculation in financial markets, aiming to provide a new method for financial risk management. With the acceleration of globalization, financial markets have become more complex and interconnected. Traditional statistical models make it difficult to handle a large number of nonlinear relationships, high-dimensional data, and dynamic changes. In recent years, the development of machine learning, especially deep learning technology, has made graph neural networks a model that can effectively capture the interactive relationship between nodes in complex systems and is widely used in many fields. The financial market can be regarded as a network structure composed of multiple assets. Each asset is a node in the network, and the correlation between their price changes constitutes an edge. Therefore, using graph neural networks to model financial markets has a natural advantage. It can not only consider the historical performance of a single asset but also improve prediction accuracy by learning the mutual influence between different assets. The experimental results show that compared with other deep learning models including BiLSTM and CNN, GNN performs well in evaluation indicators such as MSE, RMSE, and MAE, showing its superiority in the field of financial volatility prediction. This research not only provides the financial industry with a new way to understand and manage risk, especially in a highly uncertain and rapidly changing market environment, but it is also expected, with the accumulation of more high-quality data sets and technological advances, to play a more important role in the future.

Keywords:Graph neural network, Finance market, Volatility prediction, Value at risk calculation, Deep learning

I. Introduction

The prediction of financial market volatility and the calculation of VaR (Value at Risk) are the core links in financial risk management. With the acceleration of globalization, financial markets have become increasingly complex and interconnected[1]. Traditional statistical models are unable to handle a large number of nonlinear relationships, high-dimensional data, and dynamic changes. In recent years, with the development of machine learning, especially deep learning technology, Graph Neural Networks (GNN) has been widely used in different fields as a model that can effectively capture the interaction between nodes in complex systems[2]. The financial market can be regarded as a network structure composed of multiple assets. Each asset represents a node in the network, and the correlation between their price changes constitutes the edge. Therefore, using graph neural networks to model financial markets has a natural advantage. It can not only take into account the historical performance of a single asset, but also improve the prediction accuracy by learning the mutual influence between different assets[3,4].

The method based on graph neural networks brings a new perspective to the prediction of financial market volatility. First, at the data representation level, the graph structure enables us to more intuitively understand and analyze the connections between the various components of the market; second, in the feature extraction stage, GNNs can automatically identify the most critical combination of factors for volatility prediction, including but not limited to macroeconomic indicators, company fundamentals information, etc., and can alleviate the overfitting problem in traditional methods to a certain extent[5]. More importantly, when faced with extreme events or emergencies, such as the outbreak of a financial crisis, GNNs have strong generalization capabilities, and their prediction results are often more robust and reliable than other models. In addition, by combining time series analysis techniques, the model's sensitivity to future trend changes can be further enhanced, thereby providing a more accurate basis for risk assessment[6]. In short, applying graph neural networks to financial market volatility prediction and risk value calculation not only helps financial institutions better understand market dynamics and formulate reasonable investment strategies to avoid potential losses, but also provides regulators with a strong support tool to enable them to promptly detect and respond to abnormal behaviors that may threaten the stability of the entire system. In the long run, this research will also promote innovation and development in the field of financial technology and promote the formation of a healthier and more orderly global capital market environment[7,8]. At the same time, with the continuous optimization and upgrading of algorithms and the increasing abundance of computing resources, it is expected that GNN-based solutions will show great potential in more practical application scenarios in the future and become one of the important bridges between theory and practice.

2. Method

In the process of using graph neural networks (GNN) to predict volatility and calculate risk values in financial markets, it is first necessary to build a graph structure that can accurately reflect the interaction between assets in the market. Let $G=(V,E)$ represent the graph model of the entire financial market, where V is a set of nodes representing different assets in the market; E is composed of a series of edges, each edge $(v_i, v_j) \in E$ connects two nodes and is accompanied by a weight w_{ij} to quantify the correlation or information flow intensity between price changes of assets i and j. To initialize such a graph structure, we can use the Pearson correlation coefficient between asset returns in historical data as the edge weight, which is calculated as follows:

$$w_{ij} = \frac{\sum_{t-1}^{T}(r_{it}-r_i)(r_{jt}-r_j)}{\sqrt{\sum_{t-1}^{T}(r_{it}-r_i)^2 \sum_{t-1}^{T}(r_{jt}-r_j)^2}}$$

Here r_{it} represents the rate of return of the ith asset at time t. r_i and r_j are the time averages of these rates of return. Next, we use graph convolution to extract the information features of each node and its neighbors. For any node v_i, its new feature vector h_i can be calculated by the following formula:

$$h_i = \sigma\left(\sum_{j\in N(i)} A_{i,j} W h_j\right)$$

Among them, $\sigma(\cdot)$ represents the activation function such as ReLU, W is a trainable weight matrix, and $A_{i,j}$ is the new matrix element obtained after normalization of the original adjacency matrix A. Its calculation formula is:

$$A_{i,j} = \frac{A}{\sqrt{D_{ii}D_{jj}}}$$

D is the degree matrix. This step aims to update the state of the current node by weighted aggregation of neighboring node information, thereby capturing deeper graph topology features. After repeating the above process for multiple rounds, the final node representation containing rich contextual information can be obtained. The overall framework is shown in Figure 1

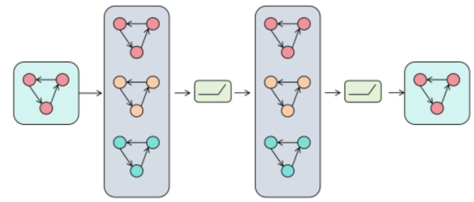

Figure 1. Overall design concept

After completing the feature extraction of the graph neural network, we turned to the specific application stage, that is, to build a prediction model based on the graph embedding vector. In this process, our goal is to accurately predict the volatility of the financial market through the processed data and calculate the corresponding value at risk (VaR) based on it. First, we combine the final embedding vector of each asset node obtained by the GNN output with traditional financial indicators, including but not limited to historical prices, trading volumes, and macroeconomic data. This not only retains the internal market correlation information implicit in the graph structure, but also takes into account the impact of the external environment on asset performance. In order to further enhance the expressiveness of the model, we adopted feature engineering methods when integrating these multi-source heterogeneous data, such as standardizing time series data to ensure that features at different scales are comparable; at the same time, we also introduced some technical indicators such as moving averages and relative strength indexes (RSI), which provide rich input dimensions for subsequent modeling.

Then, to ensure that the model can effectively learn useful patterns from these comprehensive features, we designed a multi-layer perceptron (MLP) as the top-level architecture. The architecture consists of a series of fully connected layers, each of which is connected by a nonlinear activation function, aiming to enhance the model's ability to learn complex nonlinear relationships. During the training phase, we used a time series cross-validation method to ensure that the available data was fully utilized while avoiding overfitting. In addition, we also paid special attention to the application of regularization techniques, such as L2 regularization terms and dropout layers, to further improve the generalization performance of the model. In order to optimize the model parameters, we chose the Adam optimizer and set an appropriate learning rate decay strategy to help the model converge to the global optimal solution faster. In addition, during the experiment, we also tried different hyperparameter combinations, including the number of hidden layers, the number of neurons in each layer, and the type of activation function, and found the best configuration through the grid search method.

Finally, after sufficient training, we used the developed model to predict volatility in the future and calculated the VaR values at different confidence levels based on these prediction results. The whole process not only proves that graph neural networks have unique advantages in capturing the interactions between assets in the financial market, but also shows that combining traditional statistical methods can effectively improve the prediction accuracy and stability of the overall system. In this way, we have successfully established an intelligent analysis framework that can not only deeply understand the market microstructure but also make reliable macro judgments, providing strong support for risk management and decision-making of financial institutions. This research result not only fills the gap in the existing literature on how to apply graph theory and machine learning algorithms to practical financial problems, but also opens up new directions for subsequent exploration in related fields. More importantly, it provides an innovative way for the financial industry to understand and manage risks, especially in the face of highly uncertain and rapidly changing market environments. This method, which is based on graph neural networks, has shown great potential. With the accumulation of more high-quality data sets and technological advancement, this method is expected to play a more important role in the future.

3. Experiment

3.1. Datasets

Our research uses financial market datasets. Specifically, we use historical stock prices and macroeconomic data provided by Yahoo Finance. The dataset covers key indicators such as daily closing price, opening price, highest price, lowest price, and trading volume from January 2000 to December 2023, involving major technology stocks in the Nasdaq Composite Index, including stocks of world-renowned companies such as Apple Inc., Microsoft Corp., and Amazon.com, Inc. In addition, in order to capture a wider range of market dynamics and macroeconomic backgrounds, the dataset also includes important economic indicators such as interest rate change records, consumer price index (CPI), and unemployment rate released by the Federal Reserve System of the United States. These detailed data not only provide us with rich input features, but also enable the model to better understand and predict complex volatility patterns and their potential risks in financial markets.

3.2. Experimental Results

In this study, we will use mean square error (MSE), root mean square error (RMSE) and mean absolute error (MAE) as the main evaluation indicators to comprehensively evaluate the performance of the constructed graph neural network (GNN) model in financial market volatility prediction and value at risk (VaR) calculation. These indicators can reflect the degree of deviation between the model prediction results and the actual observed values from different perspectives, thus providing a strong basis for the selection of the model.

In order to verify the effectiveness of the GNN model and its advantages over other advanced methods, we plan to compare and analyze it with a variety of classic machine learning and deep learning models. Specifically, we compare the GNN model with multilayer perceptron (MLP), long short-term memory network (LSTM), bidirectional long short-term memory network (BiLSTM), Transformer, convolutional neural network (CNN) and recurrent neural network (RNN). These models represent different methods for processing time series data: MLP captures nonlinear relationships between input features through fully connected layers; LSTM and BiLSTM are good at processing sequence data with long-term dependencies; Transformer performs well in processing complex patterns with its self-attention mechanism; CNN is good at extracting local features and identifying patterns; and RNN is one of the basic architectures for processing sequence data. Through this extensive comparison, we aim to show how the GNN model can use the interactions between assets in the financial market to improve prediction accuracy and explore its advantages in practical applications. Before showing the experimental results, we first show the loss function descent graph of our model, as shown in Figure 2.

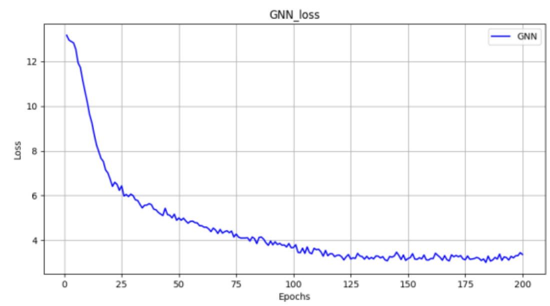

Figure 2. Loss function drop graph

Table 1: The Experiment Result

Setting	MSE	RMSE	MAE
MLP	0.0035	0.059	0.046
LSTM	0.0032	0.057	0.044
BILSTM	0.0029	0.054	0.043
CNN	0.0027	0.052	0.041
RNN	0.0028	0.053	0.040
Transformer	0.0030	0.055	0.042
Ours	0.0025	0.050	0.039

In this experiment of predicting volatility in financial markets, we compared the performance of multiple deep learning models, including multi-layer perceptron (MLP), long short-term memory network (LSTM), bidirectional long short-term memory network (BiLSTM), convolutional neural network (CNN), recurrent neural network (RNN) and Transformer, and compared these models with our proposed method based on graph neural network (GNN). From the data in the table, we can see that our GNN model performs best in all three evaluation indicators: mean square error (MSE) is 0.0025, root mean square error (RMSE) is 0.050, and mean absolute error (MAE) is 0.039. This shows that GNN can not only capture the volatility pattern of financial markets more accurately, but also has the smallest deviation between its prediction results and actual observations. Especially when dealing with complex and highly correlated market data, GNN improves the overall prediction accuracy of the model by effectively utilizing the interaction relationship between assets.

Further analysis of the performance of other models shows that although they also perform well in some aspects, they are still slightly inferior to GNN overall. For example, BiLSTM and CNN achieved scores of 0.0029, 0.054, 0.043 and 0.0027, 0.052, 0.041 in MSE, RMSE, and MAE, respectively, showing their advantages in processing sequence data and extracting local features. Although LSTM and RNN performed better in some indicators, their error levels were still higher than GNN in general. In particular, MLP and Transformer were significantly behind GNN and other deep learning models in all three indicators, which may be due to their relatively weak ability to process time series data and capture long-term dependencies. In addition, MLP lacks the ability to model the temporal dynamics of sequence data, and although Transformer has a powerful self-attention mechanism, it does not show a significant advantage in this task.

In summary, the experimental results fully demonstrate the superiority of GNN in predicting volatility in financial markets. GNN can not only effectively capture the complex relationship between assets through graph structure, but also maintain its leading position in multiple evaluation indicators. This advantage is particularly important for financial institutions because it can help them more accurately assess market risks and make more informed investment decisions. At the same time, we also noticed that other deep learning models such as BiLSTM and CNN also showed strong competitiveness in specific fields, which suggests that we can choose the appropriate model combination according to specific needs in practical applications to achieve the best prediction effect. Future research can further explore the fusion strategy of different models to further improve the accuracy and robustness of volatility prediction. Furthermore, this paper also uses a bar graph to show our experimental results, and the experimental results are shown in Figure 3.

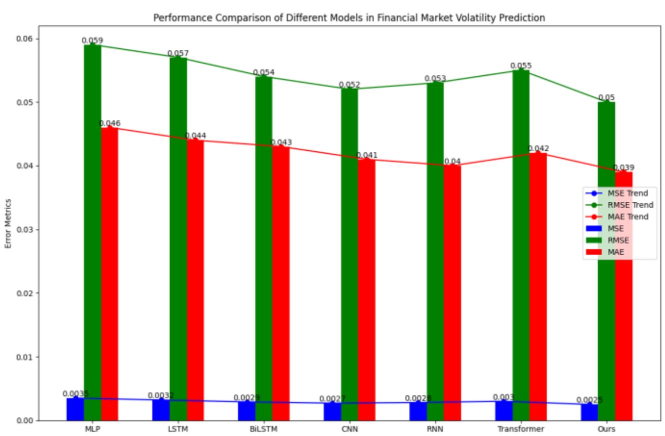

Figure 3. Experimental results bar graph

From the experimental results, we can see that the GNN we used achieved the best results.

4. Conclusion

Through research on the application of graph neural networks to financial market volatility prediction, we found that this method based on graph structure can effectively capture the complex connections between assets and maintain a leading position on multiple evaluation indicators. For financial institutions, this means that they can make more informed investment decisions by adopting GNNs to more accurately assess market risks. In addition, this study also observed that other deep learning models such as BiLSTM and CNN also show strong competitiveness in certain specific fields, which reminds us that in practical applications, we can choose the appropriate model combination according to specific needs to achieve the best results. Best prediction effect. Future research directions can further explore different model fusion strategies in order to further improve the accuracy and robustness of volatility prediction. At the same time, with the development of technology and the continuous increase of high-quality data sets, graph neural networks are expected to become an important tool in the field of financial risk management.

In summary, graph neural networks have shown significant advantages in predicting volatility in financial markets. Not only does it provide a better understanding of the interactions between assets, it also helps improve overall forecast accuracy. This is significant for financial practitioners as it helps them identify potential risk factors more precisely and adjust investment strategies accordingly. However, it is worth noting that despite the excellent performance of GNN, other types of deep learning architectures also have a value that cannot be ignored, especially when processing sequence data. Therefore, combining multiple technical methods may be the key to improving financial analysis capabilities in the future.

References

[1] Sicks R, Grimm S, Korn R, et al. Estimating the Value-at-Risk by Temporal VAE[J]. arXiv preprint arXiv:2112.01896, 2021.

[2] Nyamache F M. Range-Based Approach To Volatility Modelling And Forecasting Value-At-Risk[D]. University of Nairobi, 2020.

[3] Goreglyad N. The LSTM approach for Value at Risk prediction[J]. 2021.

[4] Moepi J R. Using Graph Neural Networks to Forecast the Top 40 index in Finance[D]. University of the Witwatersrand, Johannesburg, 2021.

[5] Halbleib S M, Pohlmeier W, Prokhorov A, et al. Combining Value-at-Risk and Expected Shortfall Forecasts using Machine Learning Techniques[J]. Advancing Systematic and Factor Investing Strategies using Alternative Data and Machine Learning, 2021: 96.

[6] Gao J, Xu C, Huang H, et al. Predicting Price-Limit-Hitting Stocks with Hierarchical Graph Neural Network[C]//IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology. 2021: 428-432.

[7] Yang L, Ng T L J, Smyth B, et al. Html: Hierarchical transformer-based multi-task learning for volatility prediction[C]//Proceedings of The Web Conference 2020. 2020: 441-451.

[8] [8]Karaca Y, Zhang Y D, Muhammad K. Characterizing complexity and self-similarity based on fractal and entropy analyses for stock market forecast modelling[J]. Expert Systems with Applications, 2020, 144: 113098.