Graph Neural Network in Stock Returns Prediction: A Case Study of the Taiwan 0050 ETF

1st Ti-Wen Chen
Institute of International Business
National Taiwan University
Taipei, Taiwan
r10724050@ntu.edu.tw

***Abstract*—In this study, I analyze the constituent stocks of the Taiwan ETF 0050 and construct a long-short investment portfolio based on predictions generated by a deep learning model. Specifically, I utilize a Gated Recurrent Unit (GRU) model to process historical time series data. Additionally, I employ Graph Neural Networks (GNNs) to model each stock as a node within a graph, where edges represent upstream-downstream or competitive relationships between companies. This method allows for the consideration of the influence of suppliers, competitors, and customers on a company's stock price movements. The results indicate that incorporating inter-company relationships significantly improves the accuracy of return predictions. Moreover, the long-short investment portfolio based on the model's predictions yields superior returns.**

***Index Terms*—Deep Learning Factor Investment, Graph Neural Network, Gated Recurrent Unit**

I. Introduction

Extensive research has been conducted on stock market data to identify patterns that forecast future stock trends. The evolution of machine learning and deep learning technologies in recent times has significantly enhanced the capabilities for such investigations. Shen et al. [1] employed Support Vector Machines (SVM) to predict the subsequent day's stock trend. Meanwhile, Rather [2] explored the use of Long Short-Term Memory (LSTM) networks for predicting stock prices and optimizing portfolios. Kelly et al. [3] utilized Convolutional Neural Networks (CNN) to analyze candlestick charts, classify cross-sectional stocks, and generate alpha that is not attributable to risk factors. Chen et al. [4] used past financial statements and stock return data to predict future returns of companies, employing Generative Adversarial Networks (GANs) to create a testing portfolio to enhance prediction results. Finally, Kelly et al. [5] applied text mining techniques to analyze sentiment for various stocks, subsequently forecasting stock returns.

However, most of these studies rely on historical data for individual stocks and often overlook the relationships between stocks, such as upstream and downstream connections and competitive interactions. Wu et al. [6] highlighted that changes in the stock returns of upstream companies can influence the returns of downstream firms. Similarly, Chen et al. [7] mentioned that the social responsibility practices of suppliers can impact the stock price crash risk of the focal company.

In this study, considering the constraints of training resources and the time-consuming nature of manually establishing graph connections, the focus is on the constituent stocks of the 0050.TW ETF. Historical data for each company in 0050.TW, including stock returns, market data, and macroeconomic indicators, is utilized. This data is fed into a Gated Recurrent Unit (GRU) model to produce time-dependent encoded features for each company. Each company is represented as a node, with edges connecting companies that have upstream, downstream, or competitive relationships, forming a complete graph. This graph is processed using a Graph Attention Network (GAT) model, which outputs the predicted stock returns for each company over the next n days. Additionally, a ***homogeneous graph*** that considers only competitive relationships and a ***heterogeneous graph*** that considers only upstream and downstream relationships are constructed and analyzed. The results indicate that incorporating the behavior of related companies enhances the predictive power of the model.

The structure of the remainder of this paper is as follows. Section 2 introduces the data and features utilized in this study. Section 3 is divided into four parts: firstly, it introduces the GRU model; secondly, it details the GAT architecture; thirdly, it describes the construction of each graph; and lastly, it demonstrates the framework used in this study. Section 4 presents the results obtained during the testing period. Finally, Section 5 provides the conclusion of this research.

II. Data

The data for this study is sourced from the TEJ database, which provides comprehensive information on stock prices, shareholding structures, financial reports, and news related to Taiwanese stocks. For the purpose of constructing the graph, I select the fifty constituent stocks of the 0050 ETF as of May 2024, in addition to the 0050 ETF itself. However, I exclude two companies, 6669.TW and 5876.TW, due to incomplete data in the TEJ database, resulting in a total of 49 stocks. The selection of excluded companies is based on the insufficiency of their data within the TEJ database.

For information on the upstream and downstream relationships of companies, I utilize the corporate supply chain data compiled by XQ, which offers detailed insights into the supply chains and products of Taiwanese enterprises. For mapping

the global corporate supply chain, the Bloomberg database is recommended as a valuable resource.

The training period starts from 2018/06/01 to 2022/12/31, testing data start from 2023/01/01 to 2024/05/10. The reason for this start date is that one of the 50 constituent stocks of the 0050.TW was listed after this period. Therefore, to ensure the inclusion of all relevant stocks, this start date was chosen. It is important to note that this study uses the constituent stocks of the 0050.TW as of May 2024. However, the constituent stocks of the 0050.TW have changed over the period starting from June 2018. To simplify the model, these changes were not considered.

Lastly, the features I use in this study include daily prices, shareholding structures, and macroeconomics indicators which are detailed in the following:

- log return
- Rogers & Satchell volatility [8]
- trading shares / total shares outstanding
- net buy/sell of foreign investment
- net buy/sell of investment banking
- net buy/sell of proprietary
- marginal ratio
- marginal trading volume / total trading volume
- short trading volume / total trading volume
- day trading ratio
- US 10yr treasury interest rate
- VIX index
- USDTWD

III. Methodology

In this section, I will introduce the implementation of the mode in this project. The first subsection will focus on the sequential model GRU. Following this, the GAT model will be detailed. Then, I will introduce the construction of the companies networks, including total network, homogeneous and heterogeneous network. Lastly, the final subsection will demonstrate the framework of GRUGAT model in this study.

A. Gated Recurrence Unit

The GRU is a type of recurrent neural network (RNN) architecture that was introduced as an improvement over traditional RNNs. GRUs were proposed by Kyunghyun Cho et al. [9] in 2014 as a simpler alternative to LSTM networks, which are another popular RNN variant.

GRUs address the vanishing gradient problem commonly encountered in standard RNNs, which can hinder the learning of long-term dependencies in sequential data. This is achieved through the use of gating mechanisms that control the flow of information within the unit. The key components of a GRU are:

1. **Reset Gate**: This gate determines how much of the past information to forget. It controls the influence of the previous hidden state on the current input.

2. **Update Gate**: This gate decides how much of the past information to retain and how much of the current input to incorporate. It essentially blends the previous hidden state with the new candidate state.

3. **Candidate State**: This is a potential new state, computed based on the current input and the previous hidden state, modulated by the reset gate.

4. **Hidden State**: The final output of the GRU unit, which is a combination of the previous hidden state and the candidate state, modulated by the update gate.

In this study, GRUs are utilized to process sequential data and generate temporal embeddings for the nodes in a graph. These embeddings capture the temporal dynamics of the data, which are then analyzed by a Graph Attention Network (GAT) to predict future outcomes based on cross-sectional impacts.

B. Graph Attention Network

GAT is a type of neural network architecture designed to work with graph-structured data. Introduced by Velickovic et al. [10] in 2018, GAT extends the concept of attention mechanisms, and also effectively address the challenge of capturing complex relationships and varying importance between nodes in a graph.

The key innovation of GAT lies in their use of attention mechanisms to compute the importance, or attention coefficients, of neighboring nodes when aggregating information. This allows GAT to dynamically focus on the most relevant parts of the graph, leading to more effective and flexible representations. The main components and steps involved in a GAT are:

1. **Input Features**: Each node in the graph has a set of input features h_i.

2. **Linear Transformation**: Input features are first linearly transformed by a learnable weight matrix $\mathbf{W}$:

$$h_i^{'} = \mathbf{W}h_i.$$

3. **Attention Mechanism**: For each pair of nodes *i* and *j* connected by an edge, an attention coefficient e_{ij} is computed:

$$e_{ij} = LeakyReLU(a^T[h_i^{'}||h_j^{'}]),$$

where a is a learnable weight vector and $||$ denotes concatenation.

4. **Attention Coefficients**: The attention coefficients are normalized using the softmax function:

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k\in N(i)}\exp(e_{ik})},$$

where $N(i)$ denotes the set of neighbors of node i.

5. **Aggregation**: The normalized attention coefficients are used to compute a weighted sum of the neighboring node features:

$$h_i^{''} = \sigma(\sum_{j\in N(i)} \alpha_{ij}h_j^{'}),$$

where σ is a non-linear activation function.

GATs can also incorporate multi-head attention, where multiple independent attention mechanisms are used, and their outputs are concatenated or averaged to form the final node representations. This enhances the model's expressiveness and robustness.

In this study, GATs are used to analyze the cross-sectional impacts of the temporal embeddings generated by a GRU for each node in the graph. By leveraging attention mechanisms, GATs can effectively capture the intricate relationships and varying importance between nodes, leading to more accurate predictions of future outcomes.

C. Complete, Homogeneous and Heterogeneous graph

In the complete graph, nodes represent companies, and edges connect nodes based on either upstream/downstream supply chain relationships or competitive relationships. This graph is comprehensive, encompassing both types of relationships to provide a full picture of the interconnections within the market. To construct the complete graph:

1. **Node Embeddings**: Each node's embedding is generated using a GRU that processes historical stock data. This embedding captures the temporal dynamics and trends of each company's stock performance.

2. **Edge Creation**: Edges are added between nodes that have either a supply chain (upstream/downstream) relationship or a competitive relationship. This results in a graph where each node is connected to all relevant nodes, forming a complete network of inter-company relationships.

Figure 1 demonstrates the connections of the nodes in the complete network.

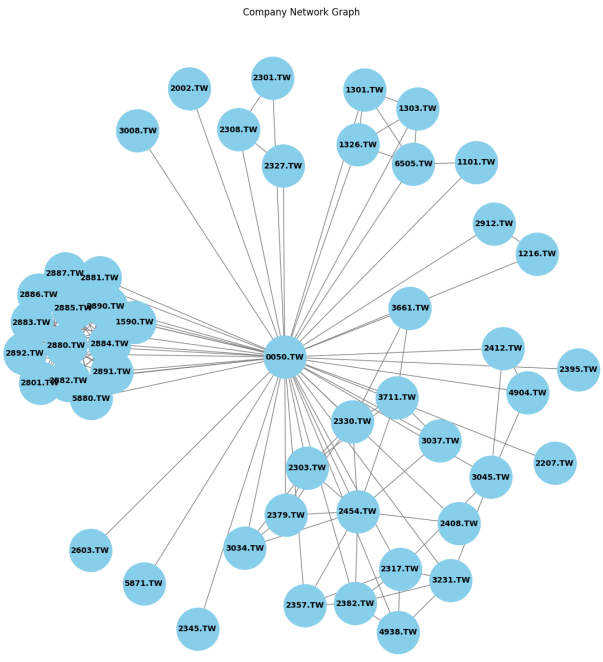

Fig. 1. Connections of nodes in complete graph

As for the homogeneous graph, it focuses solely on competitive relationships between companies. This type of graph isolates the impact of market competition by excluding supply chain relationships. Finally, the heterogeneous graph is centered on the supply chain relationships, connecting companies based on their upstream and downstream interactions. This graph emphasizes the dependencies and influences within the supply chain network.

D. Framework of GRUGAT

Figure 2 illustrates the framework of this study. First, after preprocessing the data, it is structured into a tensor $x \in \mathbb{R}^{D\times c\times s\times f}$, where D represents the number of data instances, c represents the number of companies, s represents sequential days, and f represents features. Since the time series data for each company is fed into the same GRU model to generate the time series embeddings, for each d_i the dimension of batch size that the model is given is c.

After feeding the organized data into the GRU model, it produces embeddings $x \in \mathbb{R}^{c\times h}$, where h represents the GRU hidden dimension. Next, the embeddings for each company, combined with the pre-defined edge indices based on upstream, downstream, and competitive relationships, are fed into the GAT model. The output is $\hat{y} \in \mathbb{R}^{c\times 5}$, representing the predicted cumulative log returns for the next five days. Finally, the parameters for the GRUGAT model are configured as follows:

GRU Parameters:

- Hidden Dimension: 10
- Number of Layers: 10

GAT Parameters:

- Number of Layers: 2
- Hidden Dimension: 10
- Number of Heads: 2

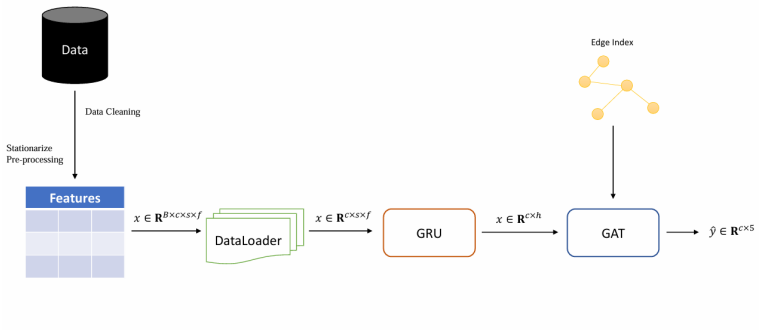

Fig. 2. Framework of the GRUGAT

IV. Empirical Results

First, we can analyze the performance of each model on the testing data by calculating the Mean Squared Error (MSE) for each of the five days following the prediction. This allows us to evaluate the accuracy of the model's predictions over time

and identify any trends or patterns in the errors. The definition of the average MSE at $T+i$ of each model is as follow:

$$MSE(T+i) = \frac{1}{D}\sum_{d=1}^{D} MSE(y^{T+i}, \hat{y}^{T+i}), y^{T+i} \in \mathbb{R}^c.$$

Table 1 presents the results of our analysis. From the results, we observe that models incorporating GAT to account for inter-company influences consistently outperform models that consider only the company's own time series data when predicting cumulative returns. Notably, the best-performing model is the one that considers only competitive relationships, represented by the homogeneous graph. This model demonstrates superior accuracy in predicting cumulative returns compared to the complete graph, which includes both upstream and downstream relationships. The likely reason for this outcome is the limited amount of data available. Including both upstream and downstream relationships introduces a greater number of parameters, which increases the risk of overfitting. Consequently, the model's performance on the testing data may suffer. In contrast, focusing solely on competitive relationships helps mitigate this issue, resulting in better generalization and improved performance on the testing data.

TABLE I
THE AVERAGE MSE OF NEXT FIVE DAYS' CUMULATIVE RETURNS IN TESTING PERIOD

Models	T+1	T+2	T+3	T+4	T+5
Complete Graph	3.071	6.176	9.280	12.148	15.195
Homogeneous Graph	3.053	6.147	9.189	12.033	14.921
Heterogeneous Graph	3.053	6.137	9.191	12.028	14.927
GRU only	3.095	6.501	9.898	12.09	15.571

Additionally, due to the inclusion of the 0050.TW node in the graph, which is connected to all other nodes, this model enables us to analyze the prediction errors for the cumulative returns of 0050.TW over the next five days. By comparing the performance of the GRUGAT model (which incorporates the relationships and interactions between constituent stocks) with a GRU model that solely relies on the historical time series data of 0050.TW, we can evaluate the benefits of incorporating graph-based relational data. Table 2 displays the comparative results. The findings indicate that the model incorporating the GAT outperforms the standalone GRU model. This suggests that considering the interdependencies among constituent stocks enhances the prediction accuracy for 0050.TW's cumulative returns over the next five days.

Furthermore, we consider a baseline model, which assumes that stock prices follow a martingale process, implying that the cumulative returns over the next five days are zero. The results show that while the GRU model alone fails to outperform this baseline assumption, the models incorporating the homogeneous graph and the heterogeneous graph do outperform the baseline. This represents a significant improvement, demonstrating the added value of integrating relational data through GAT in predicting future stock performance.

TABLE II
THE MSE OF NEXT FIVE DAYS' 0050.TW CUMULATIVE RETURNS IN TESTING PERIOD

Models	T+1	T+2	T+3	T+4	T+5
Baseline	1.025	2.060	2.966	3.893	4.859
Complete Graph	1.054	2.066	2.952	3.846	4.945
Homogeneous Graph	1.027	2.026	2.884	3.746	4.749
Heterogeneous Graph	1.032	2.038	2.901	3.768	4.762
GRU only	1.042	2.077	3.019	4.013	4.947

Finally, we can utilize the model's predictions to form portfolios and evaluate their performance during the testing period. Figure 3 illustrates the results of forming long-short portfolios by longing the top five stocks predicted to have the highest returns and shorting the bottom five stocks predicted to have the lowest returns. The analysis reveals that only the model using the heterogeneous graph achieves cumulative returns exceeding those of 0050.TW, and this outperformance is primarily due to an abnormal surge in returns from May 2023 to August 2023.

The reason for this could be that the 50 stocks included in this study are among the largest by market capitalization in Taiwan, and since the beginning of 2023, the Taiwanese stock market has been in a bullish phase. Therefore, shorting stocks predicted to have lower returns might not be appropriate in this context. To verify the assumption, Figure 4 shows the results of forming portfolios by only longing the top five stocks predicted to have the highest returns. The results indicate that the cumulative returns of almost every model outperform 0050.TW, with the heterogeneous graph model still performing the best.

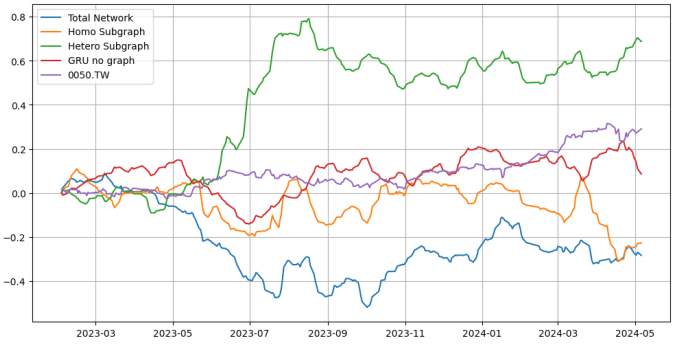

Fig. 3. Cumulative returns of long-short portfolio for each model in testing period

V. CONCLUSION

This study explored the integration of GRU and GAT to predict stock price movements of the 0050.TW constituent stocks. By leveraging both temporal and relational data, the GRUGAT model aims to enhance the accuracy of stock return predictions.

The study employed three types of graphs: complete, homogeneous, and heterogeneous graphs. The complete graph included all types of relationships, the homogeneous graph

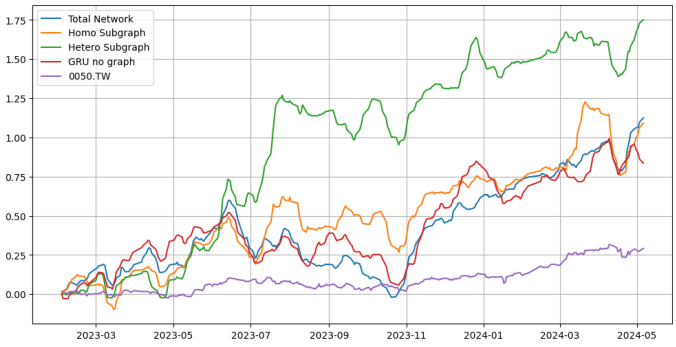

Fig. 4. Cumulative returns of long-only portfolio for each model in testing period

focused solely on competitive relationships, and the heterogeneous graph concentrated on supply chain relationships. Each company's embedding was connected to the 0050.TW node, ensuring a comprehensive representation of the market segment.

Performance analysis revealed that models incorporating GAT outperformed those relying solely on GRU. Specifically, the homogeneous graph model demonstrated the best accuracy in predicting cumulative returns. Besides, comparing the GRUGAT model with a baseline model (assuming a martingale process) further highlighted the superiority of integrating relational data. While the GRU model alone could not outperform the baseline, both the homogeneous and heterogeneous graph models did, underscoring the importance of accounting for inter-company relationships.

In conclusion, the GRUGAT model, by combining temporal embeddings from GRU with relational insights from GAT, provides a robust framework for predicting stock returns. This approach demonstrates the significant benefits of incorporating graph-based relational data.

As for future research, one could further enhance this model by considering the dynamic nature of ETF constituent stocks and exploring additional types of inter-company relationships. Secondly, one could further interpret the meaning of edge weights between nodes calculated by the GAT model, Figure 5 demonstrates the edge weights of complete graph in this study. If these two studies are completed, it will not only allow for better predictions based on real-world scenarios but should also explain why the GRUGAT model can achieve better prediction results.

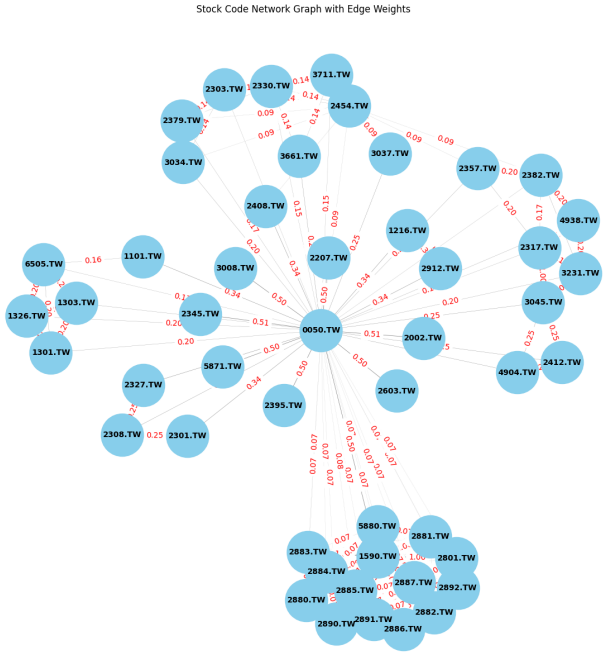

Fig. 5. Connections of nodes in complete graph with calculated edge weights

REFERENCES

[1] S. Shen, H. Jiang and T. Zhang. 2012. Stock market forecasting using machine learning algorithms. Department of Electrical Engineering, Stanford University, Stanford, CA, 1-5.

[2] A. M, Rather. 2021. LSTM-Based deep learning model for stock prediction and predictive optimization model. EURO Journal on Decision Processes, 9, 100001.

[3] J. Jiang, B. Kelly and D. Xiu. 2023. (Re-) Imag (in) ing price trends. The Journal of Finance, 78(6), 3193-3249.

[4] Chen, L., Pelger, M., Zhu, J. (2024). Deep learning in asset pricing. Management Science, 70(2), 714-750.

[5] Ke, Z. T., Kelly, B. T., Xiu, D. (2019). Predicting returns with text data (No. w26186). National Bureau of Economic Research.

[6] Wu, J., Birge, J. R. (2014). Supply chain network structure and firm returns. Available at SSRN 2385217.

[7] Chen, H. A., Karim, K., Tao, A. (2021). The effect of suppliers' corporate social responsibility concerns on customers' stock price crash risk. Advances in accounting, 52, 100516.

[8] Rogers, L. C. G., Satchell, S. E. (1991). Estimating variance from high, low and closing prices. The Annals of Applied Probability, 504-512.

[9] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

[10] Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y. (2017). Graph attention networks. stat, 1050(20), 10-48550.