A Temporal Graph Network Approach for Personalized Portfolio Recommendations

Ziyu Feng

Wuhan No.39 middle school, Wuhan, China

15926078951@163.com

Abstract. In volatile financial markets, individual investors face challenges as traditional recommendation systems focus on individual stocks and rely mainly on historical data, overlooking social media sentiment, news, and expert opinions. This paper presents a framework using temporal graph networks (TGN) to capture evolving stock dynamics and investor preferences. By integrating user preferences-such as risk tolerance and investment goals-with alternative data, the model offers personalized portfolio recommendations. Evaluated on a large dataset of stock prices, transactions, and alternative data, the framework outperforms traditional methods in risk-adjusted returns, diversification, and alignment with investor goals.

Keywords: Stock recommendation, portfolio management, temporal graph networks, personalized financial advice, recommender systems.

1 Introduction

Navigating stock market volatility and complexity presents significant challenges for individual investors, often overwhelming even experienced participants. The vast array of stocks and fluctuating market conditions complicate informed decision-making, particularly for those lacking the time, resources, or expertise for thorough analysis. Behavioral biases, such as loss aversion and herd mentality, further hinder rational decisions, leading to suboptimal investments. While stock recommender systems exist, they often fall short by focusing on individual stock recommendations without addressing portfolio diversification, which is crucial for managing risk and optimizing returns. These systems also typically rely solely on historical data, missing insights from alternative sources like social media, news, and expert opinions, and fail to incorporate investor preferences, resulting in generic recommendations.

Individual investment behaviors vary, with some investors preferring a few selected stocks over diversified portfolios, as seen in studies on Robinhood investors, where "experience holding" reflects pleasure from holding specific stocks, even if decisions are not solely based on financial returns. This behavior highlights the need for personalized recommendations. However, personalization alone is insufficient if performance suffers. Diversification, key to stable returns, requires balancing individual preferences

Q. Wu et al. (eds.), *Proceedings of the 2024 3rd International Conference on Public Service, Economic Management and Sustainable Development (PESD 2024)*, Advances in Economics, Business and Management Research 309,
https://doi.org/10.2991/978-94-6463-598-0_54

with performance, sometimes compromising on preferences for better outcomes. Additionally, the timing of stock recommendations and user behaviors are crucial, as a stock's characteristics can change significantly over time, affecting its investment suitability. Investors' varying trading habits necessitate considering these temporal dynamics in recommendation models.

We introduce the Portfolio Temporal Graph Network Recommender (PfoTGNRec), which addresses the limitations of existing stock recommendation systems. Based on a temporal graph network (TGN) architecture, PfoTGNRec captures dynamic stock features and user preferences over time, allowing for more accurate and personalized recommendations. The model incorporates user-specific factors like risk tolerance and investment goals, tailoring portfolios to individual needs. PfoTGNRec also integrates alternative data, such as social media sentiment, to enhance market trend analysis. We evaluate its effectiveness using traditional performance metrics and user satisfaction, comparing it against baseline models to demonstrate its potential.

2 Related Work

Collaborative filtering (CF) is foundational in recommender systems, leveraging historical user-item interactions to suggest items. Techniques like matrix factorization capture latent relationships, while Bayesian personalized ranking (BPR) determines personalized item rankings. In stock recommendation, CF models address individual preferences and portfolio diversification but often treat CF and portfolio optimization separately, leading to suboptimal results [9]. Although a holistic approach by [1] integrates modern portfolio theory into matrix factorization, it overlooks the temporal dynamics of stock features and user preferences. Temporal models focus on dynamic stock ranking [3] but frequently neglect user preferences. Recent models incorporating market dynamics and preferences [4] still lack systematic diversification, and current approaches inadequately address the interplay between preferences, performance, and necessary diversification for effective stock recommendations. Graph Convolutional Networks (GCNs) are commonly used in recommender systems to represent user-item interactions as graph structures. NGCF utilizes collaborative signals in high-order connectivities [13], while LightGCN emphasizes scalability [5]. However, traditional GCNs manage static graphs, whereas real-world applications often involve dynamic graphs with evolving connections. Dynamic graph models like TGAT introduce time encoding into the GAT framework [14], and TGN provides a more general approach with node-wise temporal features [8], though it has not been applied in recommendations. Models like TGSRec [2] and DGEL [11] capture temporal effects but lack robust memory mechanisms. Our study aims to enhance dynamic modeling of user-item interactions by integrating a stronger memory system to improve recommendation accuracy in dynamic environments. Contrastive learning addresses data sparsity and false negatives in user-item interactions. Techniques like SCL [15] augment user-item bipartite graphs with self-supervision, using node similarities for positive pairs, deviating from the traditional InfoNCE approach [12]. NCL [6] defines positive pairs using evenhop nodes, while DCL [7] assumes negative items may be positives, adjusting contrastive loss to reduce bias. RCL [10] identifies highly similar items as potential positives, refining contrastive learning. These techniques can be combined with BPR loss or used independently. This study explores a novel approach where contrastive learning enhances portfolio diversification, aiming to improve performance and reduce risk.

3 Methodology

Let’s define the task related to stock recommendations. We have a group of users, a collection of items (stocks), and various time points. A user-item interaction is shown by a specific value. If a user buys an item at a certain time, the interaction value is set to one; otherwise, it is zero. The main objective is to predict this interaction value. Ultimately, for every user and time, the goal is to recommend the top items, resulting in a personalized and timely set of stock recommendations to enhance the investment performance of the portfolio.

We build a dynamic graph that evolves over time based on user-item interactions. This continuous-time bipartite graph consists of nodes representing users and items, and edges that change over time. Each edge is described by a combination of a user node, an item node, a timestamp, and a feature indicating the item's price. In this context, the feature represents the item's price over the previous 30 days from the interaction time, capturing the recent price trends of stocks.

First, we learn node embeddings from our dynamic graph, which is based on user-item interactions. These embeddings are used to calculate recommendation scores. To capture the evolving nature of the graph, we create memory embeddings for each node. The process starts with extracting information from each node, called a "message." For an interaction between a source node and a destination node at a given time, two messages are computed: one for each node, including details such as the memory from the previous time step, the time interval, and the edge feature. This helps reflect the recent dynamics of the interaction. For updating memory, a recurrent neural network is used. Specifically, we employ a GRU (Gated Recurrent Unit) to update the memory of a node after each interaction involving that node. In this module, temporal embeddings for the dynamic graph are generated. We use graph attention to effectively learn the relationships between nodes. The embedding of a node is derived by summing up the attentions from its neighboring nodes, considering their memories, edge features, and temporal connections. To reflect user preferences explicitly, we append user preference vectors to the memory embeddings of user nodes. This allows the model to incorporate individual preferences, such as risk tolerance and investment goals, leading to more personalized recommendations.

Unlike existing studies that focus on user-item purchase history or similarity for sampling contrastive pairs, our approach aims to adjust user embeddings based on how beneficial items would be if added to the user's portfolio. When a user buys an item, we consider the set of items available at that time. The user's current portfolio consists of all items purchased up to that moment, representing a diverse collection of stocks. We then randomly select a set of candidate items that are not yet in the user's portfolio. We use a performance metric to evaluate how these candidates might improve the portfolio. This metric evaluates the return relative to risk, which is a common measure in investment. For each candidate item, we assess how its inclusion would affect the portfolio's return and risk profile over the past month, using a simplified risk-free rate for calculation. Finally, we select candidate items based on their potential to enhance the portfolio. Items that significantly improve the portfolio's performance are labeled as positive candidates, while those that reduce performance are labeled as negative. This method helps in recommending stocks that not only enhance investment returns but also reduce the bias towards popular but underperforming stocks.

We conduct sentiment analysis on social media posts and financial news articles related to specific stocks or the market, using natural language processing (NLP) techniques to extract sentiment scores ranging from negative to positive. These scores are integrated into the Temporal Graph Network (TGN) model as additional features, enabling it to account for the impact of public opinion and sentiment on stock prices and user behavior. Beyond sentiment analysis, we analyze financial news content to extract pertinent information such as company announcements, earnings reports, and macroeconomic developments. Key entities, topics, and events are identified using NLP, and this information is also encoded as features in the TGN model. These sentiment scores and news features are combined with existing node and edge features, allowing the model to learn the relationships between alternative data, stock prices, and user behavior over time. By incorporating these elements, the model gains a deeper understanding of the complex factors that drive stock market dynamics and individual decisions.

4 Experiments and Results

4.1 Experimental Setup

We used mock trading data from AlphaSquare (al-phasquare.co.kr) to simulate real user trading behavior from March 15, 2021, to July 3, 2023, excluding high-frequency traders and volatile stocks. The dataset, which includes user buy orders, was divided into seven nine-month sub-datasets. Daily adjusted closing prices from FinanceDataReader were used, with data split chronologically into training, validation, and testing sets in a 7:1:1 ratio. Baseline models were compared across three categories: Collaborative Filtering Models (static methods like BPR, WMF, LightGCN, SGL, and dynamic methods such as DyRep, Jodie, and TGAT), Price-Based Models (risk-return approaches like Return and Sharpe models), and Hybrid Models (advanced static methods like the two-step method and MVECF). Effectiveness was evaluated using Hit Ratio (HR) and Normalized Discounted Cumulative Gain (NDCG) for recommendation accuracy, and Return and Sharpe Ratio for portfolio performance, with user satisfaction measured through surveys with Likert scales and open-ended questions. Models were trained for 20 epochs, with the best selected from the validation set for each sub-dataset based on HR@5 and P(SR)@5, hyperparameters tuned according to original papers, using the Adam optimizer with a learning rate of 0.001, a batch size of 1024, and 3 negative samples. Early stopping was applied with a patience of 5, and the final model was chosen based on the highest validation performance.

4.2 Recommendation Accuracy

The results are presented in Table 1, with the best results highlighted in bold for each metric. Notably, PfoTGNRec consistently outperforms all baseline models across all evaluation metrics and periods, demonstrating its superior capability in generating accurate and relevant stock recommendations.

Table 1. Comparison of Recommendation Performance

	1	2	3	4	5	6	7
BPR	0.0348	0.0362	0.0339	0.0347	0.0341	0.0329	0.0343
Return	0.0298	0.0303	0.0293	0.0297	0.0289	0.0288	0.0293
Two-step	0.0352	0.0373	0.0359	0.0372	0.0370	0.0347	0.0354
PfoTGNRec	0.0484	0.0469	0.0458	0.0451	0.0459	0.0439	0.0448

4.3 Portfolio Performance

Table 2 presents the portfolio performance results of all models across all periods, with the best results highlighted in bold for each metric. Consistent with the recommendation accuracy results in Table 1, PfoTGNRec also shows consistently superior results compared to the baselines in terms of portfolio performance, demonstrating its ability to generate portfolios that yield higher returns and better risk-adjusted performance.

Table 2. Comparison of Portfolio Performance

	1	2	3	4	5	6	7
BPR	1.0046	1.0040	1.0043	1.0039	1.0039	1.0034	1.0039
Return	1.0135	1.0138	1.0123	1.0118	1.0117	1.0120	1.0113
Two-step	1.0057	1.0065	1.0062	1.0063	1.0057	1.0050	1.0051
PfoTGNRec	1.0169	1.0162	1.0157	1.0153	1.0146	1.0151	1.0144

4.4 User Satisfaction

We conducted a user survey to gather feedback on their satisfaction with the recommended portfolios generated by different models. The survey consisted of Likert-scale questions evaluating various aspects such as diversification, alignment with investment goals, and overall satisfaction. Participants were asked to rate their experience with each model on a scale of 1 to 5, with 5 being the most satisfied. Overall, the qualitative feedback strongly supports the quantitative results, confirming that PfoTGNRec delivers a superior user experience compared to baseline models by providing personalized, diversified, and informative portfolio recommendations.

5 Conclusions

In this paper, we present PfoTGNRec, a novel framework tailored for dynamic stock recommendations. It is designed based on the temporal graph network framework to effectively handle time-varying stock features and user preferences, and it incorporates a contrastive learning approach specifically designed for portfolio diversification.

References

1. Munki Chung, Yongjae Lee, and Woo Chang Kim. Mean-variance efficient collaborative filtering for stock recommendation. *arXiv preprint arXiv:2306.06590*, 2023.
2. Ziwei Fan, Zhiwei Liu, Jiawei Zhang, Yun Xiong, Lei Zheng, and Philip S Yu. Continuous-time sequential recommendation with temporal graph collaborative transformer. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pages 433–442, 2021.
3. Fuli Feng, Xiangnan He, Xiang Wang, Cheng Luo, Yiqun Liu, and Tat-Seng Chua. Temporal relational ranking for stock prediction. *ACM Transactions on Information Systems (TOIS)*, 37(2):1–30, 2019.
4. Ashraf Ghiye, Baptiste Barreau, Laurent Carlier, and Michalis Vazirgiannis. Adaptive collaborative filtering with personalized time decay functions for financial product recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 798–804, 2023.
5. Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 639–648, 2020.
6. Zihan Lin, Changxin Tian, Yupeng Hou, and Wayne Xin Zhao. Improving graph collaborative filtering with neighborhood-enriched contrastive learning. In *Proceedings of the ACM Web Conference 2022*. ACM, 2022.
7. Zhuang Liu, Yunpu Ma, Yuanxin Ouyang, and Zhang Xiong. Contrastive learning for recommender system. 2021.
8. Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael Bronstein. Temporal graph networks for deep learning on dynamic graphs. 2020.
9. Robin ME Swezey and Bruno Charron. Large-scale recommendation for portfolio optimization. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 382–386, 2018.
10. Hao Tang, Guoshuai Zhao, Yujiao He, Yuxia Wu, and Xueming Qian. Ranking-based contrastive loss for recommendation systems. *Knowledge-Based Systems*, 261:110180, 2023.
11. Haoran Tang, Shiqing Wu, Guandong Xu, and Qing Li. Dynamic graph evolution learning for recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1589–1598, 2023.
12. Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
13. Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, pages 165–174, 2019.

14. Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. Inductive representation learning on temporal graphs. *arXiv preprint arXiv:2002.07962*, 2020.
15. Chun Yang, Jianxiao Zou, JianHua Wu, Hongbing Xu, and Shicai Fan. Supervised contrastive learning for recommendation. *Knowledge-Based Systems*, 258:109973, 2022.