Mutual Fund Recommendation Based on Robust Negative Sampling with Graph-Cut Algorithm

Chiao-Ting Chen
rsps971130.cs09@nycu.edu.tw
National Yang Ming Chiao Tung University
Hsinchu, Taiwan

Szu-Hao Huang
szuhaohuang@nycu.edu.tw
National Yang Ming Chiao Tung University
Hsinchu, Taiwan

Wen-Chih Peng
wcpeng@cs.nycu.edu.tw
National Yang Ming Chiao Tung University
Hsinchu, Taiwan

ABSTRACT

In recent years, large financial institutions, such as banks, have been collecting extensive fund transaction records to analyze customer preferences and identify potential future buyers. Many traditional recommendation algorithms in the field of data mining research have been extensively employed in the development of personalized fund recommendation systems. However, learning from raw sparse and low-frequency purchase records can result in a significant data imbalance problem. Practically speaking, most participants in the financial market only engage with a limited number of financial products, remaining unfamiliar with the majority of available options. This further increases the difficulty of negative sampling and simultaneously reduces the accuracy of preference classifier learning in recommendation system. In this paper, a robust negative sampling strategy is proposed based on heterogeneous graph embedding and homogeneous graph cut techniques. Financial expert knowledge is captured from unstructured data through graph embedding, allowing us to extract richer information and learn meaningful representations from heterogeneous graphs. By leveraging graph cut techniques, we can partition all the items into a positive item pool and a negative item pool, representing the items that users like and dislike, respectively. The goal of graph cut is to minimize the similarity between the positive and negative item categories, which allows the negative item pool to be used for robust negative sampling. The developed method was applied on a real-world dataset, and the results revealed that the method had a higher precision score than several state-of-the-art methods.

KEYWORDS

Negative Sampling, Graph Embedding, Graph Cut

ACM Reference Format:
Chiao-Ting Chen, Szu-Hao Huang, and Wen-Chih Peng. 2023. Mutual Fund Recommendation Based on Robust Negative Sampling with Graph-Cut Algorithm. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (kdf-workshop).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

With the explosive growth of technology and the availability of vast amounts of data, deep learning has become effective tools to deal with different kinds of tasks in various fields. In recent year, there are more and more people apply the advanced techniques in recommendation systems [2] [17] owing to their extraordinary performance in wide range of applications. Deep learning is good at discover nontrivial relationships among various items by virtue of a lot of nonlinear combinations. It can capture intricate and implicit connections in an efficiency and effective way which can not only mitigate the information overload problem but also help customers to find their interested items.

Customized financial product recommendation is an important application within the robo-advisory services in the era of financial technology. Conventional recommendation strategies often apply collaborative filtering techniques to analyze the relationship between customers and items. However, obtaining enough transaction records in real-world scenarios is challenging. Only a limited portion of data is available in the form of labeled records because the majority of individuals interact with few number of financial product. Learning from such imbalanced data can pose challenges in the context of negative sampling, especially when we lack information about a customer's preferences for items they did not purchase. Inaccurate representation of negative samples may result in less reliable recommendations, leading to lower user satisfaction and reduced trust in the system.

Recommendation for financial instruments are different from traditional products since financial products are equipped with a unique characteristic. For traditional e-commerce recommendation problems, the feature of products and customers are fixed most of time and they seldom change over time. However, in financial industries, most of information concerning financial products and customers will changes over time frequently [4] [1]. Take stock for example, the opening price and the closing price of a stock will change every day. Besides, there are lots of factors influencing investment decisions such as news, risk tolerance, profitability, investment term, just to name a few. Therefore, making recommendations on financial products is more difficult than other products. Recently, some deep learning approaches have been applied in financial recommendation system [14] [9]. In addition, numerous researches adopted long short term memory networks [18] to do stock price prediction and use reinforcement learning for portfolio management [8] [10]. They all obtained extraordinary performance than traditional statistical methods or machine learning approaches.

In this research, we propose a negative sampling strategy based on homogeneous graph cut and heterogeneous graph embedding

kdf-workshop, July 27, 2023, Taipei, Taiwan

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

techniques. Graph is a common data structure [6] and has been applied in various fields [19]. Differ from general data structure, graph can provide people with more information since it can model complicated interactions among all members through nodes and edges. For example, in a social network, we may discover a potential relation with another person and make recommendation suggestions to that person. Due to the progress of computation ability and growing volume of information, graph analysis has caught lots of research attention [12] [11]. Graph embedding enables us to capture intricate relationships among various data using a graph structure. Financial domain know-how and expert knowledge can be extracted from the unstructured data.

Subsequently, based on graph embedding obtaining from the previous stage, a relation graph will be constructed. We then calculate the distance between two nodes and apply Gaussian kernel to obtain similarity. Later, graph cut technique is applied to identify potential positive samples and improve the representative of negative samples. The goal of graph cut is to divide all vertices of a graph into several disjoint subsets and let each sets have few connection. Therefore, in a well-clustered graph, there should be considerable edges within each cluster while few edges between different clusters. In the past few decades, graph cut has attracted lots of attention due to its computation efficiency [13]. Ratio cut [5] was first proposed by Hagen and Kahng in 1992. Ratio cut considers the size of each vertex sets by their number of vertices when doing graph clustering. Normalized cut was first proposed by shi and Malik [15] and it measured the size of each vertex by the weights of its edge. The above two algorithms have been used in several field recent year [16]. By integrating the above two techniques, the recommendation model can achieve better performance by solving data sparsity problem.

The major contributions of this research can be summarized as follows:

- A negative sampling strategy is designed for customized fund recommendation system based on graph embedding and graph cut techniques.
- Meaningful representations of financial domain knowledge can be captured from unstructured data through graph embedding.
- By leveraging graph cut techniques, representative negative samples are discovered to help constructing a more trustful fund recommendation system.
- We conduct experiments on a real world data set, and the results prove that our proposed framework has better performance than state-of-the-art fund recommendation approaches.

2 PROPOSED METHOD

There are two stages in our proposed method including graph embedding and graph cut. We will provide detailed descriptions on them in the following paragraph.

2.1 Fund embedding Using Financial Graph

Graph embedding is a common approach used to represent data characteristic on a large graph. The purpose of graph embedding is to convert high dimensional vectors into low dimensional vectors while preserving the original information. In our financial product recommendation scenario, we first construct a graph with various heterogeneous nodes including funds and customers. The graph $G = \langle F, C, E_{fc}, \rangle$ where $F = \{f_i\}$ represents fund nodes, $C = \{c_j\}$ represents customer nodes. i, j represent the total number of funds and customers respectively. $E_{fc} = \{e_{ij} \mid f_i \in F, c_j \in C\}$, symbolize the edges between funds and customers. In other words, if a customer c_j buys a fund f_i, $e_{ij} = 1$; otherwise $e_{ij} = 0$. Fund features such as registration country, investment market and customer features such as personal income range, level of holding credit cards are all considered as node features in the graph. Through financial product and customer graph, we can model different relationship among multiple types of nodes. The graph structure helps us to capture implicit relationship among those nodes and generate better representation of each nodes.

In the research, we apply the concept of graphSAGE to design our network since the advantage of graphSAGE is to handle the problem of large scale graph. The main concept of GraphSAGE is that it generate embedding by sampling and aggregating information which comes from neighbors. In other words, during the process of producing node embedding, it will first sample the neighbors of the target node and aggregate those neighbors' information to be a target's new embedding. With the mechanism of sampling and aggregating, the algorithm can handle with inductive node embedding problem. Through GraphSAGE algorithm each node will have stronger relationship with its neighbors. Embedding function is shown as equation 1 where *AGG* means aggregation function using in the process of aggregating.

$$h_v^k = \sigma([W_k \cdot AGG(h_u^{k-1}, \forall u \in N(v)), B_k h_v^{k-1}]) \tag{1}$$

2.2 Negative Sampling via Graph Cut

After obtaining the embedding of each node, relation graph will be constructed based on the information in this stage. The purpose of building the graph is to solve negative sampling problems by improving All missing as negative (AMAN) strategy. In our proposed method, we apply graph cut techniques to distinguish between positive examples and negative examples and thus raise the opportunity to select truly negative examples to be negative examples and put them into recommendation system to train.

Regarding graph cut algorithm, the easiest way to solve these problems is using minimum cut approach. The concept of minimum cut is to let all vertices in the graph cluster to K disjoint vertex sets and make the edges between the vertex sets have the smallest sum of weight. Equation 2 illustrates the minimization equation of minimum cut approach where $\bar{A_i}$ is the complement of A.

$$cut(A_1, ..., A_k) := \frac{1}{2} \sum_{i=1}^{k} W(A_i, \bar{A_i}) \tag{2}$$

Normalized cut was first proposed by shi and Malik and it measured the size of each vertex by the weights of its edge $assoc(A, V)$. The definition of $assoc(A, V)$ is showed as equation 3. The equation represents the weights of all edges which is attached to vertices in the subset A.

$$assoc(A, V) = \sum_{u \in A, t \in V} w(u, t) \tag{3}$$

Besides, since the definition of equation 2, equation 3 and $V = A + B$, we have equation shown as equation 4. The value of $assoc(A, V)$ will be restricted to one to zero.

$$\begin{aligned} assoc(A, V) &= \sum_{u \in A, t \in V} w(u, t) \\ &= \sum_{u \in A, t \in B} w(u, v) + \sum_{u_1, u_2 \in A} w(u_1, u_2) \geq cut(A, B) \geq 0 \end{aligned} \tag{4}$$

In our situation, we have applied minimum cut to see which graph cut algorithm will generate better results. In this stage, each customer will have the same graph which all funds are nodes in the graph. Later graph cut techniques will be applied in the graph to divide the whole graph into two groups. One is stand for positive examples, the other represent negative examples. It is noticing that cutting methods are different for each customer. Therefore, after optimization, a suitable cutting energy will be found to meet every customer's demand. The process of generating relation graph is shown as fig. 1.

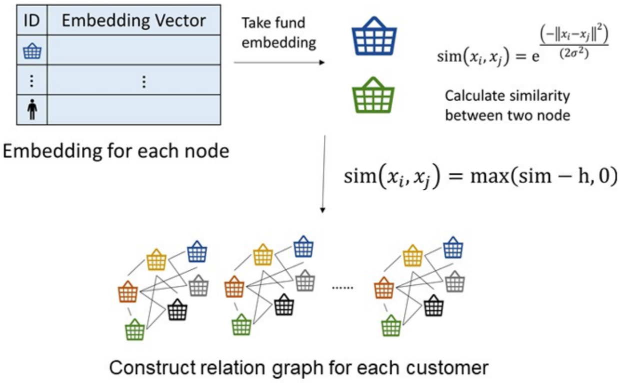
ID
Embedding Vector
Embedding for each node
Take fund embedding
$\mathrm{sim}(x_i, x_j) = e^{\frac{(-\|x_i - x_j\|^2)}{(2\sigma^2)}}$
Calculate similarity between two node
$\mathrm{sim}(x_i, x_j) = \max(\mathrm{sim} - \mathrm{h}, 0)$
......
Construct relation graph for each customer

Figure 1: The Process of Generating Relation Graph

3 EXPERIMENT

we conduct experiments to verify the effectiveness of the proposed method. Historical transaction records of mutual funds from a Taiwan commercial bank is used in our experiments. The range of data starts from 2016/09/01 to 2019/06/30. There are 2, 697 funds and 113, 891 customers and 9, 746, 667 transaction records. Original transaction records includes purchase, convert, redeem, and other operations. In this research, we only keep the purchase transactions in our model to be analyze. There are customer data and fund data in the financial product database. We divided both features into two categories. One is categorical features, the other is continuous features. For categorical features, one hot encoding techniques will be adopted to preprocess those features. While for continuous features, normalization approach such as min-max normalization or Z-score normalization will be applied to prevent outliers which will probably effect the model performance.

3.1 Experimental Setting

The training period we adopt is six month and testing period will be one month. The dimension of the embedding space where we calculate graph cut loss is set to 16. The learning rate is set to 0.001 and the optimizer we applied is Adam. We apply precision scores to measure the recommendation results of financial product recommendation. Top 1, 3, 5, 10, K of precision are conducted since they are common numbers of fund recommendation. In equation 5, TP denotes the correctly predicted examples and K represent number of positive examples.

$$Precision = \frac{number\ of\ TP\ examples}{K} \tag{5}$$

In addition, we design several methods as baselines in the experiment. The detailed description about them are listed below.

- **AMAN**: We set up the baseline model which adopt the AMAN strategy in negative sampling problems.
- **Dynamic Sampling Strategy (DS)**: In this benchmark, we change the negative examples based on prediction results every ten epoch. By doing so, we have higher probability to obtain truly negative examples.
- **GraphDCF [3]**: The graph-structured network is constructed by connecting customer nodes based on their similar purchase and redeemed trading orders, allowing for the modeling of various latent relationships among customers with similar shopping habits.
- **KGknowledge [7]**: This is the knowledge graph method, which employed a knowledge graph structure and deep learning techniques to embed features of customers and funds into a unified latent space.
- **Proposed**: The proposed model we use to overcome data sparsity problems.

3.2 Analysis of Experimental Results

The experimental results are shown in Table 1. The original dataset separate customers into four categories based on their income. G1 is the group that has the lowest income whereas G4 is the group that has the highest income. As we can observe from the table, the precision performance of our proposed method is better than two control groups including AMAN model and dynamic sampling strategy model. The lowest value 1.48% appears in group four when k equals to one and the approach is AMAN model. The highest value 10.47% appears in group three when k equals to one and the approach is graph cut. It is noticing that in group three, the proposed method will achieve extraordinary performance than other groups. In addition, we can observe group four sometimes will have worse performance. We conclude that since people in group four have highest income, they may not totally follow the recommended suggestion and have their own investing strategy.

After doing experiments on different groups of customers, we merge all customers to test the overall performance. The experimental results are shown as Table 2. The row stands for different baselines we adopt and the proposed method in this research. The column presents all our measurement in precision@k where k is 1, 3, 5 respectively. The number in the table is calculated by equation 5. The results in the table represents one-month testing results.

Take the number 7.28% for example, it means the precision outcome that make one customized mutual fund to every investor with our proposed method. The highest value appears in k equals to 1 and the approach is our proposed method. The lowest value appears in k equals to 5 and the approach is AMAN. It is noticing that the recommendation results is worse when k is 5. As we can observe from the table, the precision performance of our proposed method is higher than dynamic sampling strategy, AMAN model and state-of-the-art approaches GraphDCF and KGknowledge. We can conclude that our proposed method obtain rich information through graph embedding and discover representative negative samples during graph cut stage.

Table 1: Recommendation Performance Comparison for Different Groups

Measurement	1	3	5	10	K
G1 AMAN	1.94%	1.96%	2.22%	2.50%	2.06%
G1 DS	4.06%	3.74%	3.41%	2.82%	3.75%
G1 Proposed	**8.69%**	**7.39%**	**6.02%**	**4.89%**	**7.61%**
G2 AMAN	2.18%	2.32%	2.53%	2.69%	2.45%
G2 DS	4.67%	3.73%	3.36%	2.94%	3.70%
G2 Proposed	**8.82%**	**7.50%**	**6.33%**	**5.02%**	**7.24%**
G3 AMAN	3.49%	1.94%	2.09%	2.56%	1.54%
G3 DS	5.81%	4.65%	4.65%	3.49%	5.10%
G3 Proposed	**10.47%**	**6.98%**	**5.81%**	**5.23%**	**9.09%**
G4 AMAN	1.48%	1.89%	2.07%	2.46%	1.74%
G4 DS	4.43%	3.61%	3.40%	2.78%	4.32%
G4 Proposed	**7.88%**	**7.47%**	**6.50%**	**4.90%**	**8.08%**

Table 2: Recommendation Performance Comparison for Overall Customers

	1	3	5
AMAN	2.43%	2.34%	2.21%
DS	2.94%	4.82%	4.90%
GraphDCF	6.82%	6.02%	5.17%
KGknowledge	7.19%	5.62%	4.90%
Proposed	**7.28%**	**6.69%**	**5.79%**

4 CONCLUSION

The problem we tackle with in this paper is negative sampling problems which there is few positive training data and lots of unknown data. It is difficult to identify representative negative examples under such data since all of truly negative data and possibly positive examples are mixed together during the training process of constructing a recommendation system. In order to solve the problem, we fully leverage graph embedding and graph cut techniques. First of all, we construct a graph and apply graph embedding technique to obtain the embedding of each node. The process help us discover implicit relationship among each node which the valuable information cannot be obtain without using graph structure. In addition, we adopt graph cut techniques for each customer to divided all unknown examples into positive examples and negative examples. The goal of graph cut aims at discovering those possibly positive examples and representative negative examples. By integrating the above two techniques, the recommendation model can achieve better performance. To verify the effectiveness of our proposed method, we apply our method on a financial product data and the results also work well than other baselines. In the future, We hope to deploy our proposed method in real-world recommendation systems and test its effectiveness in practical applications.

REFERENCES

[1] Prasad N Achyutha, Sushovan Chaudhury, Subhas Chandra Bose, Rajnish Kler, Jyoti Surve, and Karthikeyan Kaliyaperumal. 2022. User Classification and Stock Market-Based Recommendation Engine Based on Machine Learning and Twitter Analysis. *Mathematical Problems in Engineering* 2022 (2022).

[2] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*. 7–10.

[3] Yi-Ching Chou, Chiao-Ting Chen, and Szu-Hao Huang. 2022. Modeling behavior sequence for personalized fund recommendation with graphical deep collaborative filtering. *Expert Systems with Applications* 192 (2022), 116311.

[4] Shibo Feng, Chen Xu, Yu Zuo, Guo Chen, Fan Lin, and Jianbing XiaHou. 2022. Relation-aware dynamic attributed graph attention network for stocks recommendation. *Pattern Recognition* 121 (2022), 108119.

[5] Lars Hagen and Andrew B Kahng. 1992. New spectral methods for ratio cut partitioning and clustering. *IEEE transactions on computer-aided design of integrated circuits and systems* 11, 9 (1992), 1074–1085.

[6] William L Hamilton, Rex Ying, and Jure Leskovec. 2017. Representation learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584* (2017).

[7] Pei-Ying Hsu, Chiao-Ting Chen, Chin Chou, and Szu-Hao Huang. 2022. Explainable mutual fund recommendation system developed based on knowledge graph embeddings. *Applied Intelligence* (2022), 1–26.

[8] Szu-Hao Huang, Yu-Hsiang Miao, and Yi-Ting Hsiao. 2021. Novel deep reinforcement algorithm with adaptive sampling strategy for continuous portfolio optimization. *IEEE Access* 9 (2021), 77371–77385.

[9] Tsan-Yin Hung and Szu-Hao Huang. 2022. Addressing the cold-start problem of recommendation systems for financial products by using few-shot deep learning. *Applied Intelligence* 52, 13 (2022), 15529–15546.

[10] Yu-Cen Lin, Chiao-Ting Chen, Chuan-Yun Sang, and Szu-Hao Huang. 2022. Multiagent-based deep reinforcement learning for risk-shifting portfolio management. *Applied Soft Computing* 123 (2022), 108894.

[11] Hai Liu, Chao Zheng, Duantengchuan Li, Zhaoli Zhang, Ke Lin, Xiaoxuan Shen, Neal N Xiong, and Jiazhang Wang. 2022. Multi-perspective social recommendation method with graph representation learning. *Neurocomputing* 468 (2022), 469–481.

[12] Yujie Mo, Liang Peng, Jie Xu, Xiaoshuang Shi, and Xiaofeng Zhu. 2022. Simple unsupervised graph representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 7797–7805.

[13] Maria CV Nascimento and Andre CPLF De Carvalho. 2011. Spectral methods for graph clustering–a survey. *European Journal of Operational Research* 211, 2 (2011), 221–231.

[14] Samkit Shah and Harshal Trivedi. 2021. Social Media Analytics and Mutual Fund Recommendation. In *Proceedings of International Conference on Communication and Computational Technologies: ICCCT-2019*. Springer, 287–303.

[15] Jianbo Shi and Jitendra Malik. 2000. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence* 22, 8 (2000), 888–905.

[16] Yangtao Wang, Xi Shen, Shell Xu Hu, Yuan Yuan, James L Crowley, and Dominique Vaufreydaz. 2022. Self-supervised transformers for unsupervised object discovery using normalized cut. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14543–14553.

[17] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 974–983.

[18] Xukuan Zhan, Yuhua Li, Ruixuan Li, Xiwu Gu, Olivier Habimana, and Haozhao Wang. 2018. Stock Price Prediction Using Time Convolution Long Short-Term Memory Network. In *International Conference on Knowledge Science, Engineering and Management*. Springer, 461–468.

[19] Daokun Zhang, Jie Yin, Xingquan Zhu, and Chengqi Zhang. 2017. User profile preserving social network embedding. In *IJCAI International Joint Conference on Artificial Intelligence*.