Intelligent Decision Technologies 18 (2024) 717–729
DOI 10.3233/IDT-240352
IOS Press

Transfer contrast learning based on model-level data enhancement for cross-domain recommendation

Chenyun Yu[a,*] and Xiwei Feng[b]
[a]*Network Information Center, Liaoning Petrochemical University, Fushun, Liaoning, China*
[b]*School of Innovation and Entrepreneurship, Liaoning Petrochemical University, Fushun, Liaoning, China*

Received 19 March 2024
Accepted 11 April 2024

Abstract. A cross-domain recommendation system is an intelligent recommendation technology that integrates multiple fields or types of data. It can cross independent information islands, effectively integrate and complement data resources, and improve recommendation performance. This paper proposes a transfer contrast learning method based on model-level data enhancement for cross-domain recommendations. This method first obtains the initial embeddings of the two domains using item-based collaborative filtering, after which it enhances the transformer network with model-level data through contrastive learning to pre-train the source domain data. The pre-trained transformer network parameters are then transferred and fine-tuned before being applied to tasks on the target domain data. The information link from the source domain to the target domain is effectively constructed, and it has been proven to improve the accuracy and effectiveness of the target domain on real datasets.

Keywords: Cross-domain recommendation system, model-level, collaborative filtering, pre-training, fine-tuned

1. Introduction

With the advent of the Internet and the popularity of intelligent applications, recommendation systems have become an essential component of many application platforms, including e-commerce sites, social media, and news aggregation applications, which strive to provide users with personalized and accurate content through recommendation algorithms [1,2,3,4,5,6,7,8].

Conventional recommendation systems mainly rely on data from a single field for modeling and recommendation [9,10,11]. However, in the real world, users' behavioral data is often scattered across multiple platforms and fields, and the data from these fields is correlated and complementary. Therefore, how to integrate this scattered data, break data silos, and provide users with more comprehensive and accurate recommendation services has become a hot topic in current research. Under such a background, a cross-domain recommendation (CDR) system emerges [12,13,14]. Its objective is to integrate data from various fields and use correlation and complementarity to improve the accuracy and satisfaction of recommendations. Cross-domain recommendation can not only improve sparse data, but also assist the platform in better understanding users, optimizing the user experience, and increasing user stickiness [15, 16].

*Corresponding author: Chenyun Yu, Network Information Center, Liaoning Petrochemical University, Fushun, Liaoning 113001, China. E-mail: chyyu@lnpu.edu.cn. ORCID: 0009-0009-3037-4576.

In recent years, cross-domain recommendation systems have received extensive attention and research. Researchers have proposed a variety of cross-domain recommendation methods from various perspectives. For instance, [17] developed a collective matrix factorization (CMF) model that embedded data into a global matrix spanning all domains and then decomposed the matrix to extract hidden features of users and items in each. [18] developed a CDR system that mined each domain's features through embedding and mapping, resulting in cross-domain recommendation mapping from rich domain to sparse domain. [19] regarded the behavior of multiple users in multiple domains as a view as the user's embedding, and then performed feature extraction on multiple views using the multi-branch deep neural network (MVDNN), which is particularly effective on user cold startup. [20] solved the problem of the CDR system through multi-task learning. It constructs two branches to extract features from users and items in two domains. During the feature extraction process, common user information in both domains is shared, and the recommendation effect of the two domains improves synchronously. [21] used transfer learning to borrow useful information from the source domain and used implicit orthogonal mapping during the transfer process to maintain the similarity between it and the target domain. [22] used the idea of joint training to embed the cross-domain system's users and items, allowing them to focus on domain-sharing information while limiting domain-specific information. [23] developed a meta-learner to generate a personalized feature bridge for each user, thereby personalizing the representation for each user across domains. [24] considered the decoupling of domain-shared and domain-specific information, which greatly increases the model's migration efficiency, as well as the use of mutual information rules to improve cross-domain recommendation performance.

The cross-domain recommendation system approach that does not use graph structure ignores the high-level structural features implied by users and items in the interaction graph, making the model unable to fully capture the responsible interaction between users and items, and also limiting the performance of the recommendation system [25]. To overcome the limitation that algorithms cannot mine the high-order hidden features between users and items, the graph neural network has emerged as the mainstream feature extraction model for recommendation systems [26,27,28]. For instance, [29] used a graph convolutional network for collaborative filtering, simply weighting and aggregating user nodes or item nodes to obtain the final embedded features of each node. In the context of a multi-objective cross-domain recommendation system, a framework combining graph structure and attention was introduced in [30] to mine the heterogeneous graph composed of each domain, thereby improving the quality of the final embedded feature vector of users and items. [31] constructed a bridge of information communication between the source domain and the target domain through bidirectional transfer technology. They used the graph collaborative filtering network in the feature mining process, demonstrating that transfer learning can improve cross-domain recommendation performance. [32] proposed an extended framework for the graph relationship between users and items, which synthesizes the bipartite graph composed of the source domain and target domain into one graph through node similarity, enhances the graph data, and employs a hierarchical attention network for feature extraction, resulting in improved recommendation performance.

Recently, contrast learning has made significant advances in all areas of deep learning [33,34,35,36]. Because contrast learning combined with a data enhancement module can bring more samples and learn better-classified features by maximizing the consistency between the original feature and the enhanced feature, as well as the distance between the original feature and the other feature, recommendation performance has greatly improved [37,38,39,40,41]. To further improve the recommendation performance of a cross-domain recommendation system, this paper combined comparative learning with a GraphSAGE neural network to dig deeper features of data in the source domain, and transfer learning as an information bridge between two domain data to improve recommendation accuracy in the target domain. The main contributions of this paper are as follows:

Because of the complex relationships between different fields or platforms in current technologies, the key to cross-domain recommendation is how to effectively map and transfer information from different

fields or platforms. Therefore, this article uses transfer learning and fine-tuning techniques from the NLP field, as well as contrastive learning, to design transfer contrast learning based on model-level data enhancement for CDR. This method not only effectively enhances the embedding representation of users and items in the two domains, but it can also be applied to a variety of target domains to enhance their recommendation performance. The innovation points are as follows:

First, this paper combined comparative learning with the GraphSAGE neural network to dig deeper into the source domain's features, and then used transfer learning as an information bridge between two domains to improve the target domain's recommendation accuracy.
Second, in contrast to learning based on model-level data enhancement, two domain data are trained. It can not only effectively mine the hidden associations between users and items, but it can also improve the robustness of the model and the ability to resist noise without destroying the original data's internal association, resulting in improved accuracy.
Third, after transferring the pre-trained GraphSAGE parameters to the GraphSAGE network in the source domain, the fine-tuning training method of Low-Rank Adaptation (LoRA) is used to reduce the model's trainable parameters.

In this paper, the model is comprehensively evaluated on two real-world large-scale recommendation system data sets, Amazon and MovieLens, and the results show that its recommendation performance is significantly superior.

2. The framework of the CDR system

Figure 1 depicts a detailed description of transfer contrast learning based on a model-level data enhancement framework for the CDR system. The framework consists primarily of two stages: pre-training and transfer learning. In the first stage, model-level data enhancement is adopted to train data comparison learning in the source domain; in the second stage, comparative learning and supervised learning training are carried out on the target domain simultaneously through the transfer learning stage, which mainly transfers the parameters from the previous stage. Through the above two stages, the hidden relationship between users and items in the source domain data can be effectively fused into the target domain data.

2.1. Initial vector encoding

Before constructing the cross-domain recommendation system model, the initial vector encoding of data in the two domains should be used as the input features of the CDR system. Assume that U^s and U^d are the user sets of the source domain and the target domain, respectively, and V^s and V^d are the item sets of the source domain and the target domain, respectively, where there are some overlapping users between U^s and U^d. Where each user's initial vector is encoded as its rating sequence for all items, and the initial vector encoding of each item is calculated by the cosine similarity between it and all other items, which is as follows:

$$S_{i,j} = \frac{X_{V_i} \times X_{V_j}}{\|X_{V_i}\| \cdot \|X_{V_j}\|} \tag{1}$$

Where $S_{i,j}$ is the cosine similarity, X_{V_i} and X_{V_j} are the rating sequences for all users, respectively. If the number of items in the source and target domains is N, then the initial vector encoding dimension for each user or item is $R^{1\times N}$.

2.2. The stage of pre-training

After obtaining the initial vector coding of each user and item, the source domain data is pre-trained

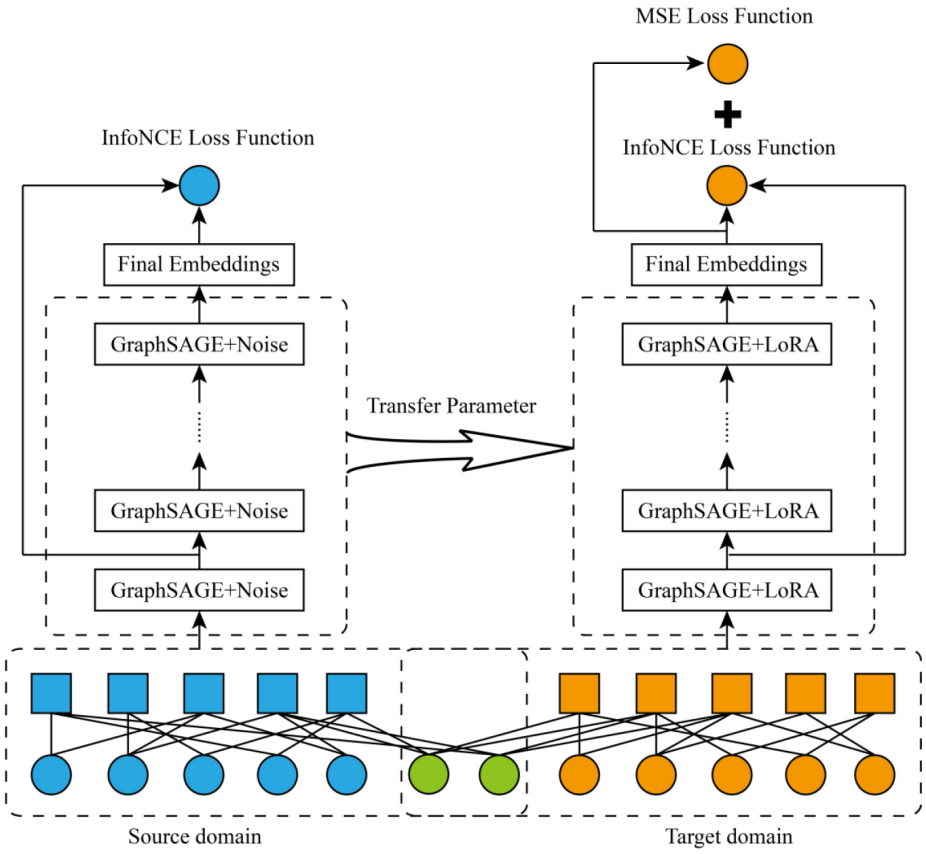


Fig. 1. The structure of the cross-domain recommendation system.

using contrast learning, allowing the feature extraction model to effectively mine the relationship between users and items in the domain.

First, this paper generates a bipartite graph structure $G(N, E)$ from the data in the source domain to represent the relationship between all users and items, where N represents the node set in the graph, which is the number of users plus the number of items, and the feature of each node is its initial vector code. E is the edge set of the graph and indicates that there is a relationship between users and items.

Then, the depth feature extraction of the two-part graph G is carried out by GraphSAGE with Noise, which mainly consists of two parts: sampling neighborhood and aggregation features from neighbors, as shown in Fig. 2. Sampling neighbors entails randomly sampling a fixed number of neighbors for each node in the graph. Sampling is used to reduce the consumption of computing resources, particularly at nodes with many neighbors. Aggregation features from neighbors entails using an aggregation function to combine the neighbor feature representations for each node based on its sampled neighbors. This article uses the pooling aggregation function to mine the hidden features of each node, as shown in Fig. 3. The calculation process includes two branches: one branch calculates the fusion features of the target node and Noise.

$$\begin{cases} z_0^l = h_0^l + n_0^l \\ n_0^l = \varepsilon \cdot sign(h_0^l) \end{cases} \tag{2}$$

Where z_0^l is the fusion feature of the target node and noise at layer l, h_0^l and n_0^l are the features of the target node and the features of the noise node at layer l, respectively. ε represents the scaling factor, a

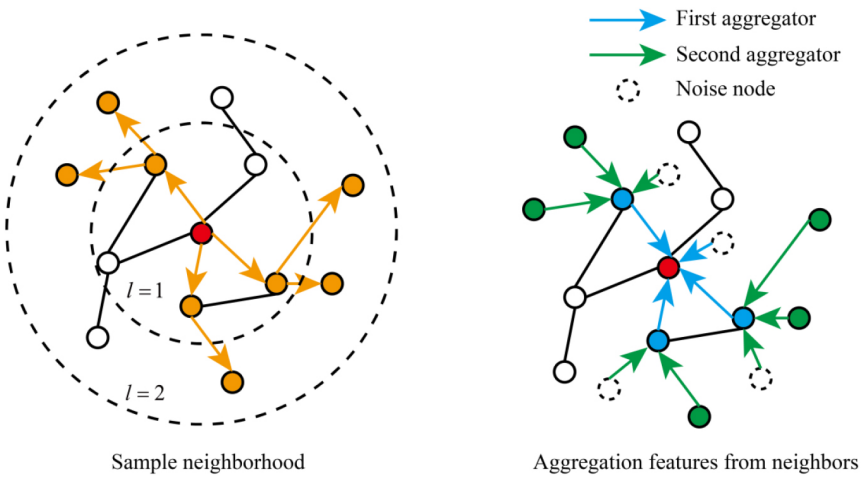


Fig. 2. The specific steps of the GraphSAGE.

small hyperparameter that ensures that the target node's characteristics, when combined with the noise characteristics, do not change significantly but do introduce subtle differences.

The other branch calculates the fusion features of all sampled neighbor nodes. The specific calculation process is as follows:

$$z_p^l = \text{Pooling}(\sigma(w_i^l h_i^l) | i \in s) \tag{3}$$

where z_p^l is the fusion features of all sampled neighbor nodes at layer l, h_i^l represents the features of the i-th neighbor node at layer l, w_i^l is the weight, and s is the total number of all neighbor nodes.

After obtaining the features of the two branches, they are fused using concat, and the features of the target node's next layer are obtained using a fully connected layer. The calculation is as follows:

$$h_0^{l+1} = [w_c^l(z_0^l \| z_p^l)] \tag{4}$$

where h_0^{l+1} is the feature of the target node at layer $l+1$, and w_c^l is the weight of the fully connected layer.

Pre-training is a process of contrast learning, so InfoNEC is used as a loss function:

$$L_{pre} = \sum_{i \in (U^s \cap V^s)} -\log \frac{\exp[(h_i^1)^{\mathrm{T}} h_i^L / \tau]}{\sum_{j \in (U^s \cap V^s)} \exp[(h_i^1)^{\mathrm{T}} h_j^L / \tau]} \tag{5}$$

Where h_i^1 and h_i^L are the node features in the first and L (last) layers of the i node, respectively, and h_j^L is the node features in the L (last) layers of j node. τ represents the temperature coefficient, which is a set hyperparameter, and its main function is to control the model's differentiation of negative samples. The infoNEC function can make the positive samples h_i^1 and h_i^L consistent, and make the negative samples h_i^1 and h_j^L non-consistent.

2.3. The stage of transfer learning

The GraphSAGE network used for feature extraction in the source domain is acquired through pre-training. During the transfer learning stage, the trained weight parameters are transferred to the target domain to extract features, but the bipartite graph data composed of data in the target domain is trained through fine-tuning. This paper uses Low-Rank Adaptation (LoRA) for fine-tuning. LoRA works by freezing all of the weights in the pre-trained GraphSAGE and then injecting a dimensionality-reducing matrix and a dimensionality-increasing matrix, resulting in a significant reduction in trainable parameters

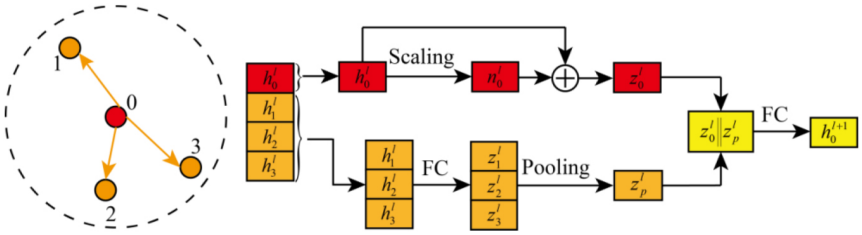


Fig. 3. The node aggregation in GraphSAGE.

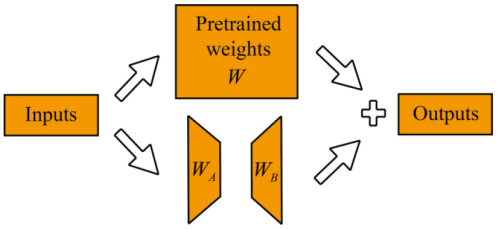


Fig. 4. The structure of the LoRA module.

in the downstream task. The specific structure is shown in Fig. 4, and in fact, a branch is added. The branch first applies a linear layer to reduce the dimension of the input feature, followed by another linear layer to restore the dimension to its original dimension. Finally, the results of the double-branch are added and fused to produce the output.

Assuming the pre-trained weight matrix is $W \in R^{d\times k}$, its update can be expressed as follows:

$$W + \Delta W = W + W_B W_A, W_B \in R^{d\times k}, W_A \in R^{r\times k} \tag{6}$$

Where W_A and W_B are the weights of changing dimensions. In the fine-tuning process of LoRA, W is fixed, and only W_A and W_B are training parameters. In the forward propagation process, W and ΔW will be multiplied by the same input feature X, and finally added to obtain the next layer feature H:

$$H = WX + \Delta WX = WX + BAX \tag{7}$$

In the transfer learning stage, this paper uses the joint loss function L_J to train the target domain data, as follows:

$$L_J = L_{rec} + \lambda L_{cl} \tag{8}$$

Where L_{rec} and L_{cl} are InfoNEC loss for comparison learning and mean square error loss for users and items scoring, respectively. The specific calculation process is shown in Eq. (9), and λ is the proportional coefficient.

$$\begin{cases} L_{cl} = \sum\limits_{i\in(U^d\cap V^d)} -\log \frac{\exp[(h_i^1)^{\mathrm{T}} h_i^L/\tau]}{\sum\limits_{j\in(U^d\cap V^d)} \exp[(h_i^1)^{\mathrm{T}} h_j^L/\tau]} \\ L_{rec} = \frac{1}{E} \sum\limits_{i\in U^d, j\in V^d} (y_{i,j} - \hat{y}_{i,j})^2 \end{cases} \tag{9}$$

Where $y_{i,j}$ and $\hat{y}_{i,j}$ are actual and calculated scores, and E is the number of node pairs.

3. Experimental analysis

All experiments in this article are conducted on a computer equipped with an Nvidia GeForce RTX3090 GPU.

Table 1
A specific description of the two data setkua

Dataset	MovieLens				Amazon				
Domain	Comedy	Drama	Action	Thriller	Books	CDs	Music	Movies	Beauty
User	2113	2113	2113	2113	12761	12761	12761	12761	30000
Item	3029	3975	1277	1460	7346	2541	778	8270	302782
Score	332038	381616	241211	226975	81986	82612	27050	184133	375781
Sparsity	5.19	4.54	8.94	7.36	0.09	0.27	0.29	0.18	0.01

Table 2
Setting cross-domain recommendation system tasks

Dataset	MovieLens			Amazon		
Task	1	2	3	1	2	3
Source domain	Comedy	Drama	Action	Books	CDs	Music
Target domain	Drama	Action	Thriller	CDs	Music	Movies

Table 3
The recommended score error for each task on the MovieLens dataset

Dataset	Task	Metric	CMF	L-GCN	DTCDR	PGPRec	PTUP	DisenCDR	Our
MovieLens	Task1	MAE	0.73	0.72	0.75	0.70	0.71	0.99	0.62
		RMSE	0.96	0.99	1.06	0.91	0.93	1.12	0.83
	Task2	MAE	0.70	0.82	0.84	0.74	0.68	0.96	0.67
		RMSE	0.92	1.01	0.99	0.94	0.89	1.04	0.85
	Task3	MAE	0.71	0.88	0.95	0.77	0.71	0.92	0.68
		RMSE	0.90	1.11	1.09	0.96	0.92	0.98	0.87

3.1. Data description

In this experiment, two true-real data sets with multiple domains are used [3] from the MovieLens dataset and Amazon. Table 1 shows that MovieLens has four item domains: comedy, drama, action, and thriller, while the Amazon data set has four item domains: books, CDs, music, movies, and beauty.

In this paper, three pairs of source domain and target domain combinations from MovieLens and Amazon are selected as CDR system tasks, as shown in Table 2. The performance of the designed CDR system is then evaluated in these six tasks.

3.2. Performance superiority analysis

To demonstrate the superiority of our method, we will compare it based on the depth study to six advanced baseline models: the CMF [13], L-GCN [25], DTCDR [16], PGPRec [33], PTUP [19], and DisenCDR [20].

First, the accuracy of each recommendation model is compared. The mean absolute error (*MAE*) and root mean squared error (*RMSE*) are used as evaluation metrics.

$$MAE = \frac{1}{E_{test}} \sum_{i \in U^{test}} \sum_{j \in V^{test}} |y_{i,j} - \hat{y}_{i,j}| \tag{10}$$

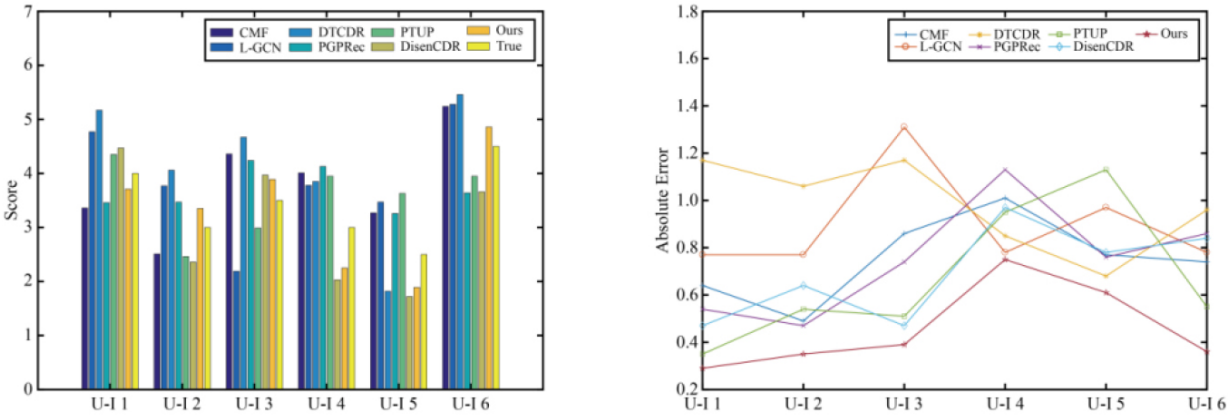
$$RMSE = \sqrt{\frac{1}{E_{test}} \sum_{i \in U^{test}} \sum_{j \in V^{test}} (y_{i,j} - \hat{y}_{i,j})^2} \tag{11}$$

where E_{test} represents the number of the test set, $y_{i,j}$ and $\hat{y}_{i,j}$ are the actual score and calculated score. *MAE* and *RMSE* analyze error values; the lower their values, the higher the accuracy of the model.

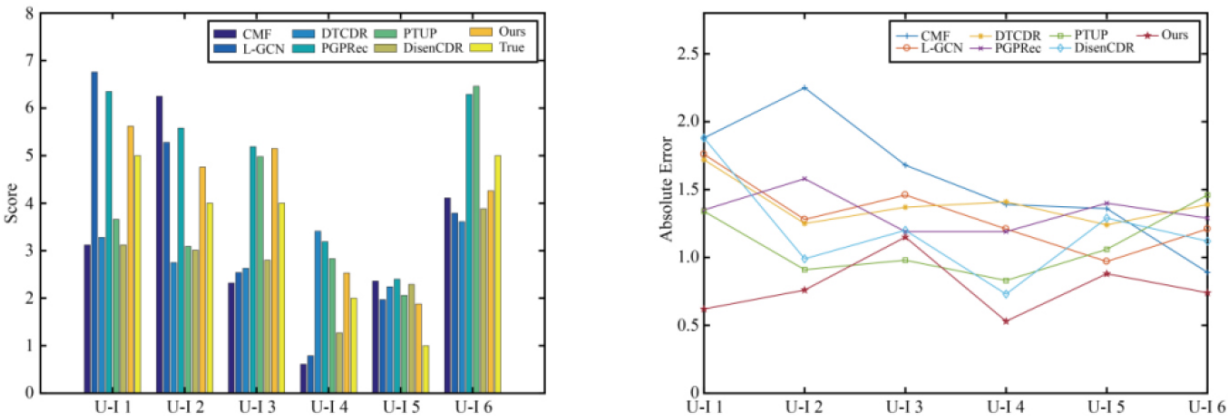
Tables 3and 4 present the error indicators for all recommended models on the MovieLens and Amazon

Table 4
The recommended score error for each task on the Amazon dataset

Dataset	Task	Metric	CMF	L-GC	DTCDR	PGPRec	PTUP	DisenCDR	Our
Amazon	Task1	MAE	2.08	1.39	1.44	0.97	1.02	1.15	0.76
		RMSE	2.91	1.48	1.55	1.15	1.16	1.33	0.95
	Task2	MAE	1.07	1.35	1.46	1.22	0.81	1.27	0.74
		RMSE	1.41	1.60	1.68	1.49	1.16	1.39	0.93
	Task3	MAE	1.04	1.34	1.44	1.28	1.08	1.23	0.77
		RMSE	1.27	1.52	1.61	1.43	1.23	1.31	0.96



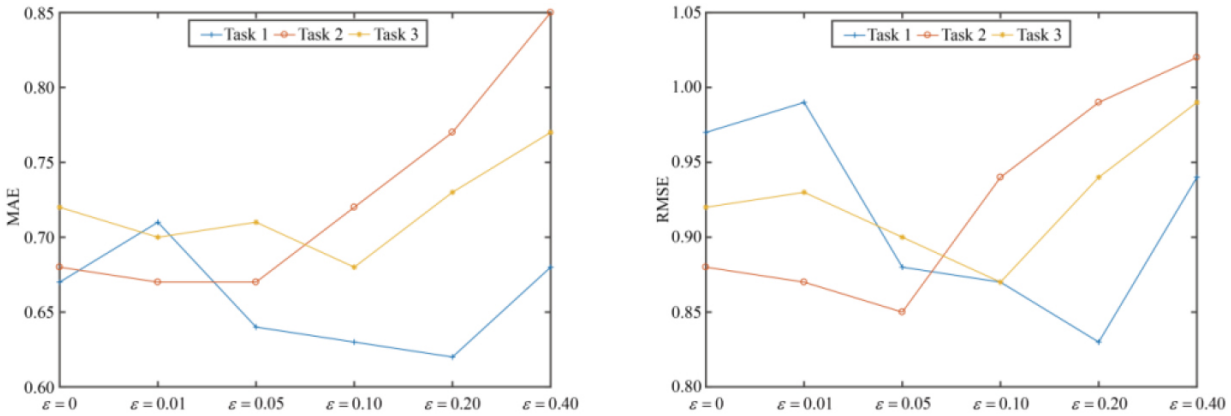
(a) MovieLens



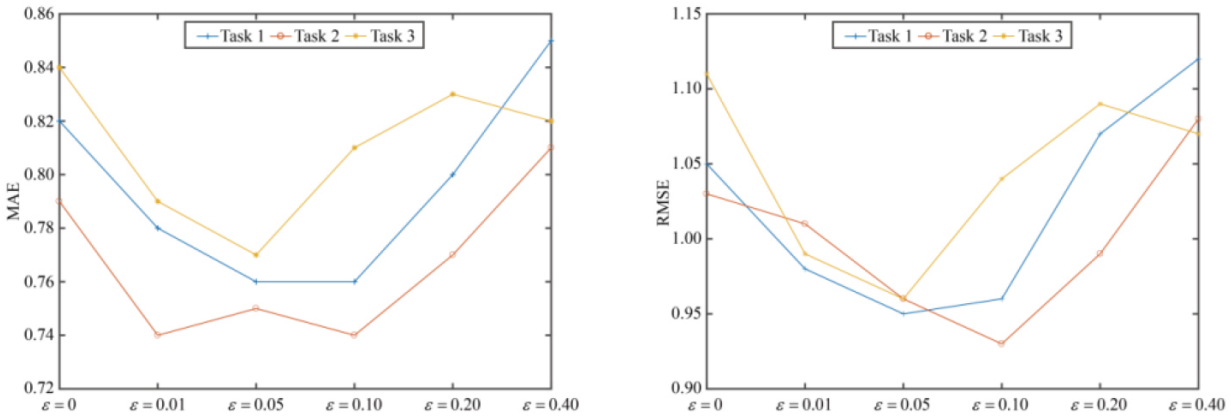
(b) Amazon

Fig. 5. Compare the real and calculated scores for different user and item instances.

datasets, respectively. Table 3 shows that MovieLens reduces MAE by 11.11%, 1.47%, and 4.23%, respectively, as well as RMSE by 8.79%, 4.49%, and 3.33%, when compared to the best-performing comparison model in the MovieLens dataset for tasks 1, 2, and 3. Table 4 shows that MovieLens reduces MAE by 21.64%, 8.64%, and 25.96%, respectively, as well as RMSE by 25.96%, 19.83%, and 24.41%, when compared to the best-performing comparison model in the MovieLens dataset for tasks 1, 2, and 3. For different tasks in different data sets, the error index of the recommendation model designed by us is lower than that of the comparison model on the whole, indicating that its recommendation performance is superior.



(a) MovieLens



(b) Amazon

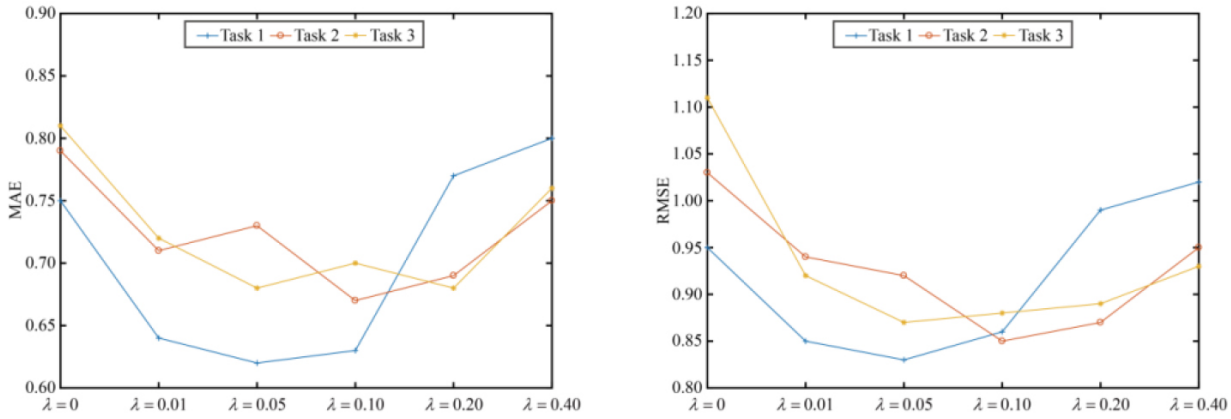
Fig. 6. The recommendation errors of the model vary with the scaling factor.

To more intuitively demonstrate the difference between the real score and the calculated score of specific cases of each model in different tasks in different datasets, this paper randomly selects two paired users and items for each task. The specific values of the real score and the calculated score and the absolute error between them are shown in Fig. 5. It is evident that among the most randomly selected paired users and items, the calculated score of the recommendation model we designed is the closest to the actual score. Only U-I 3 in the Amazon data set has a lower similarity between the calculated and actual scores than the PTUP model. It demonstrates that the recommendation model we designed can still provide good recommendation performance in a single case.

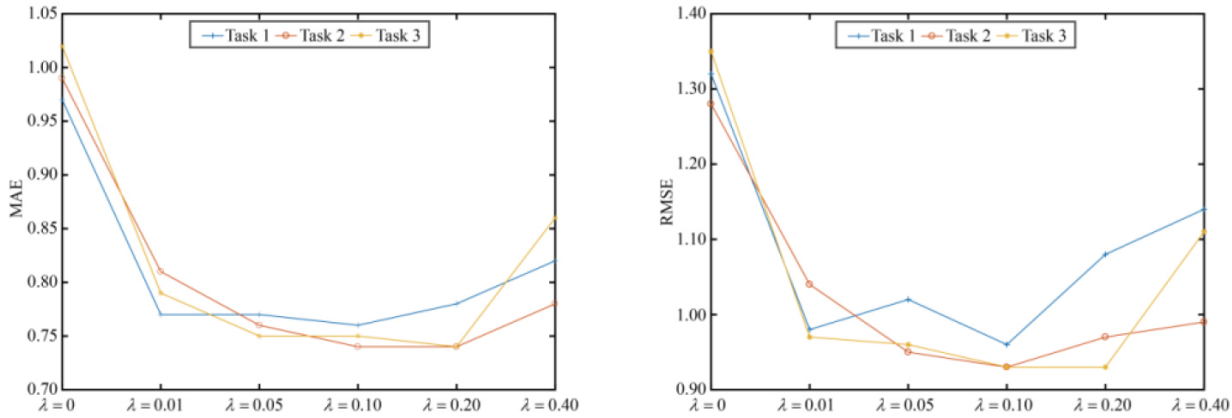
3.3. Key parameter selection

The model designed in this article has two key parameters: scaling factor ε and professional coefficient λ.

n this paper, we first changed the scaling factor to evaluate the model's error indicators on various tasks. Figure 6 shows that when ε of tasks 1, 2, and 3 in the MovieLens dataset selected are 0.20, 0.05, and 0.10, respectively, the error was the lowest, indicating that the model had the best recommendation performance. When ε of tasks 1, 2, and 3 in the Amazon data set is selected as 0.10, 0.05, and 0.05, respectively, the error is the lowest, indicating that the model has the best recommendation performance.



(a) MovieLens



(b) Amazon

Fig. 7. The recommendation errors of the model vary with the professional coefficient.

Then, the proportional coefficient is changed to evaluate the error index of the model on different tasks. As shown in Fig. 7, it can be seen that when λ is selected to be 0.05, 0.10, and 0.05, respectively in tasks 1, 2, and 3 in the MovieLens dataset, the error is the lowest. That is, the model has the best recommendation performance. When λ of tasks 1, 2, and 3 in the Amazon dataset is selected as 0.10, 0.10, and 0.10, respectively, the error is the lowest, which means that the model has the best recommendation performance.

A comprehensive comparison of two key parameters reveals that the model is more sensitive to r than to e, indicating that changes in r are more likely to cause changes in the model's recommendation accuracy.

3.4. Ablation experiment

This paper uses an ablation experiment to verify that the transfer learning approach can effectively construct an information bridge between two domains. Two ablation models are developed and compared to the original model. Ablation model 1 eliminated the need for pre-training with contrast learning and trained the target domain data directly with GraphSAGE. Ablation model 2 is based on the original model but does not use the LoRA fine-tuning process, and the target domain migrates the pre-trained model's GraphSAGE. Tables 5 and 6 show the results, with A_Model 1 and A_Model 2 representing ablation model 1 and ablation model 2, respectively.

Table 5
Results of the ablation experiments in MovieLens dataset

Dataset	Task	Metric	A_Model 1	A_Model 2	Original
MovieLens	Task1	MAE	0.76	1.35	0.62
		RMSE	0.95	1.67	0.83
	Task2	MAE	0.81	1.86	0.67
		RMSE	1.02	2.54	0.85
	Task3	MAE	0.87	2.12	0.68
		RMSE	1.03	2.67	0.87

Table 6
Results of the ablation experiments in Amazon dataset

Dataset	Task	Metric	A_Model 1	A_Model 2	Original
Amazon	Task1	MAE	0.87	1.87	0.76
		RMSE	1.22	2.97	0.95
	Task2	MAE	0.82	1.53	0.74
		RMSE	1.03	2.24	0.93
	Task3	MAE	0.81	1.61	0.77
		RMSE	1.15	2.37	0.96

As shown in Tables 5 and 6, the MAE and RMSE of each ablation model in different tasks on different datasets are higher than the original model; pre-training and fine-tuning are both extremely important for the model's recommended performance; and fine-tuning has a greater impact on the original model.

4. Conclusion

A transfer contrast learning method based on model-level data enhancement is proposed for the CDR system. The method uses transfer learning as an information link between two domains, ensuring that useful information is efficiently transferred between them. At the same time, by combining model-level data enhancement technology and a graph neural network, more effective hidden features of the user and item nodes are mined. By performing cross-domain recommendation tasks on two publicly available datasets, it proved that the method is superior to existing advanced recommendation methods. In addition, the key parameters involved in the method are effectively selected through experiments, and the important roles of transfer learning and fine-tuning are also verified through ablation experiments.

Acknowledgments

This work was supported by the General Project of Liaoning Provincial Department of Education (No. LJKMZ20220741).

Conflict of interest

We declare that we do not have any known interests or personal relationships that could potentially influence the reported work in this paper.

References

[1] Fu W, Peng Z, Wang S, et al. Deeply fusing reviews and contents for cold start users in cross-domain recommendation systems. Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, 2019. pp. 94-101.

[2] Zhong S, Huang L, Wang C, et al. An autoencoder framework with attention mechanism for cross-domain recommendation. IEEE Transactions on Cybernetics. 2020; 52(6): 5229-5241.
[3] Dubey N, Verma A, Setia S, et al. PerSummRe: Gaze-based personalized summary recommendation tool for wikipedia. Journal of Cases on Information Technology (JCIT). 2022; 24(3): 1-18.
[4] Tiwari S, Kumar S, Jethwani V, et al. PNTRS: Personalized news and tweet recommendation system. Journal of Cases on Information Technology (JCIT). 2022; 24(3): 1-19.
[5] Huang L, Zhao Z, Wang C, et al. LSCD: Low-rank and sparsecross-domain recommendation. Neurocomputing. 2019; 366: 86-96.
[6] Wang C, Deng Z, Lai J, et al. Serendipitous recommendation in e-commerce using innovator-based collaborative filtering. IEEE Transactions on Cybernetics. 2018; 49(7): 2678-2692.
[7] Zheng T, Ding M. Behavior-Aware English Reading Article Recommendation System Using Online Distilled Deep Q-Learning. Journal of Cases on Information Technology (JCIT). 2023; 25(1): 1-21.
[8] Alrashidi M, Selamat A, Ibrahim R, et al. Social Recommender System Based on CNN Incorporating Tagging and Contextual Features. Journal of Cases on Information Technology (JCIT). 2024; 26(1): 1-20.
[9] Shin H, Kim S, Shin J, et al. Privacy enhanced matrix factorization for recommendation with local differential privacy. IEEE Transactions on Knowledge and Data Engineering. 2018; 30(9): 1770-1782.
[10] He X, Du X, Wang X, et al. Outer product-based neural collaborative filtering. Proceedings of the 27th Int Joint Conference on Artificial Intelligence. Freiburg: IJCAI. 2018. pp. 2227-2233.
[11] Chae D, Kang J, Kim S, et al. Rating augmentation with generativeadversarial networks towards accurate collaborative filtering. Proceedings of the 26th World Wide Web Conference. New York: ACM. 2019. pp. 2616-2622.
[12] Babak L, Yue S, Martha L, et al. Cross-domain Collaborative Filtering with Factorization Machines. ECIR. 2014; 656-661.
[13] Feng Z, Chen C, Wang Y, et al. DTCDR: A Framework for Dual-Target Cross-Domain Recommendation. CIKM. 2019; 1533-1542.
[14] Zhu Y, Tang Z, Liu Y, et al. Personalized transfer of user preferences for cross-domain recommendation. Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining. 2022. pp. 1507-1515.
[15] Ren H, Liu B, Sun J, et al. A time and relation-aware graph collaborative filtering for cross-domain sequential recommendation. Journal of Computer Research and Development. 2023; 60(1): 112-124.
[16] Guo L, Li Q, Liu i, et al. Shared-account cross-domain sequential recommendation with self-attention network. Journal of Computer Research and Development. 2021; 58(11): 2524-2537.
[17] Singh A, Gordon G. Relational learning via collective matrix factorization. Proceedings of the 14th ACM SIGKDD Int Conference on Knowledge Discovery and Data Mining. New York: ACM. 2008. pp. 650-658.
[18] Man T, Shen H, Jin X, et al. Cross-domain recommendation: An embedding and mapping approach. Proceedings of the 26th Int Joint Conference on Artificial Intelligence. Freiburg: IJCAI. 2017. pp. 2464-2470.
[19] Elkahky A, Song Y, He X. A multi-view deep learning approach for cross domain user modeling in recommendation systems. Proceedings of the 24th International Conference on World Wide Web. 2015. pp. 278-288.
[20] Zhu F, Chen C, Wang Y, et al. DTCDR: A framework for dual-target cross-domain recommendation. Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2019. pp. 1533-1542.
[21] Li P, Tuzhilin A. DDTCDR: Deep dual transfer cross domain recommendation. Proceedings of the 13th International Conference on Web Search and Data Mining. 2020. pp. 331-339.
[22] Cao J, Sheng J, Cong X, et al. CrossDomain Recommendation to Cold-Start Users via Variational Information Bottleneck. arXiv preprint arXiv2203.16863. 2022.
[23] Zhu Y, Tang Z, Liu Y, et al. Personalized transfer of user preferences for cross-domain recommendation. Proceedings of the 15th ACM International Conference on Web Search and Data Mining. New York: ACM. 2022. pp. 1507-1515.
[24] Cao J, Lin X, Cong X, et al. DisenCDR: Learning disentangled representations for cross-domain recommendation. Proceedings of the 45th International ACM SIGIR Conferenc on Research and Development in Information Retrieval. New York: ACM. 2022. pp. 267-277.
[25] Wu S, Sun F, Zhang W, et al. Graph neural networks inrecommender systems: A survey. ACM Computing Surveys. 2022; 55(5): 1-37.
[26] Ahmed N, Rossi R, Zhou R, et al. Inductive Representation Learning in Large Attributed Graphs. doi: 10.48550/arXiv.1710.09471. 2017.
[27] Zhang S, Tong H, Xu J, et al. Graph convolutional networks: A comprehensive review. Computational Social Network. 2019; 6(1): 1-23.
[28] Wu F, Souza A, Zhang T, et al. Simplifying graph convolutional networks. Proceedings of the 36th Proceedings on Machine Learning. New York: PMLR. 2019. pp. 6861-6871.
[29] He X, Deng K, Wang X, et al. Lightgcn: Simplifying and powering graph convolution network for recommendation. Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2020. pp. 639-648.
[30] Zhu F, Wang Y, Chen C, et al. A Graphical and Attentional Framework for Dual-Target Cross-Domain Recommendation. In IJCAI. 2020; 3001-3008.
[31] Liu M, Li G, Pan P. Cross Domain Recommendation via Bi-directional Transfer Graph Collaborative Filtering Networks. The 29th ACM International Conference on Information and Knowledge Management ACM. 2020.

[32] Xu k, Xie Y, Chen L, et al. Expanding Relationship for Cross Domain Recommendation. Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 2021. pp. 2251-2260.
[33] You Y, Chen T, Sui Y, et al. Graph contrastive learning with augmentations. NeurIPS. 2020; 33: 1-12.
[34] Chen T, Kornblith S, Norouzi M, et al. A simple framework for contrastive learning of visual representations. ICML. 2020; 1597-1607.
[35] He K, Fan H, Wu Y, et al. Momentum contrast for unsupervised visual representation learning. CVPR. 2020; 9729-9738.
[36] Liu X, Zhang F, Hou Z, et al. Self-supervised learning: Generative or contrastive. arXiv preprint arXiv2006.08218. 2020.
[37] Yi Z, Ounis I, Macdonald C. Contrastive Graph Prompt-tuning for Cross-domain Recommendation. ACM Trans. Inf. Syst. 2023; 1(1): 1-26.
[38] Xie X, Sun F, Liu Z, et al. Contrastive Learning for Sequential Recommendation. IEEE 38th International Conference on Data Engineering (ICDE). 2022.
[39] Zhou K, Wang H, Zhao W, et al. S3-Rec: Self-Supervised Learning for Sequential Recommendation with Mutual Information Maximization. Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2020. pp. 1893-1902.
[40] Yu J, Xia X, Chen T, et al. XSimGCL: Towards Extremely Simple Graph Contrastive Learning for Recommendation, IEEE Transactions On Knowledge And Data Engineering. 2023; 1-14.
[41] Chen M, Huang C, Xia L, et al. Heterogeneous Graph Contrastive Learning for Recommendation. arXiv:2303.00995v1. 2023.