Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/44/20230485

# Multi-source serialization cross-domain recommendation algorithm based on deep learning

**Zhehan Chen[1,2], Zhisen Wang[1,3,5], and Xinzhe Wang[1,4]**

[1]Dalian Polytechnic University, China

[2]zhehan1213@163.com
[3]z_s_wang@dlpu.edu.cn
[4]wxzagm@dlpu.edu.cn
[5]corresponding author

**Abstract.** Cross-domain recommendation is an effective approach to solve the cold start and data sparsity problems in recommendation systems. Sequential recommendation can model user behavior sequences and improve the accuracy of recommendation. Currently, few recommendation algorithms consider both aspects together, and most of them do not utilize multi-source information sufficiently. In view of this, this paper proposes a multi-source serialization cross-domain recommendation model, which fully considers the temporal and contextual relationships in two domains, and fuses multi-source information on the basis of achieving cross-domain recommendation tasks, and reinforce the embedding representation by fitting the interest forgetting function. Finally, use a Multilayer Perceptron as the mapping function to learn the nonlinear mapping relationship between the source domain and the target domain, subsequently enabling recommendations for new users in the target domain. On Amazon dataset, this model can significantly enhance the accuracy of recommendation.

**Keywords:** Cross-Domain Recommendation, Multi-Source Information, Serialization, Interest Forgetting Curve.

## 1. Introduction

Recommendation system is a tool to solve information overload in the digital era. Traditional recommendation systems face the challenge of sparse data, where the amount of user-generated behavior is very small compared to the overall item count. Additionally, there is also a user cold start issue, where new users have little interaction on the platform, making it difficult to provide accurate recommendations for them.

In recent times, with the emergence of cross-domain recommendation, these two problems have been effectively addressed. Cross-domain recommendation can capture the knowledge of target domain users in the source domain and apply it to the target domain for more accurate recommendations. Currently, research on cross-domain recommendation is booming. Liu et al. [1] integrated collaborative filtering through matrix factorization with deep adversarial domain adaptation utilizing attention networks. They considered both common and domain-specific knowledge to enhance recommendation accuracy. Zhang et al. [2] merged knowledge graphs for cross-domain recommendation to enable different domain projects to share knowledge. Li et al. [3] suggested applying adversarial learning to merge global and specific domain user preferences to provide satisfactory cross-domain recommendations. Sahu [4] proposed a method that can transfer knowledge learned from two distinct source domains. In the first domain, user-side knowledge was extracted, while the second domain focused on extracting item-side knowledge, and could control the dominant behavior of the source domain knowledge. While these cross-domain recommendation models had shown promising results, most of them relied on static user behavior data and seldom considered the changes in user interests over time. In practice, user behavior exhibits many sudden changes, and the recent and distant behavior of users has a different degree of influence on the next behavior. There are also deep connections between user behaviors. Therefore, we propose to introduce sequence recommendation-related techniques.

In real scenarios, the interactions between users and items are recorded according to time. By modeling this dynamic information and deeply mining users' potential interests, thereby improving the accuracy of recommendations. Ai et al. [5] aimed at the problem that user behaviors of long sequences often contained numerous implicit and noisy preference signals, and most models ignored the time interval between interactions. They explicitly extracted those interactions related to user interests from implicit feedback information and adaptively prioritized users' core interests. At the same time, they considered the time interval to retain effective information. Guo et al. [6] proposed a novel multi-channel orthogonal decomposition attention network that applied two channels: one that only focused on pure dependency relationships between items and another that captured feature transfer patterns. Kheldouni et al. [7] proposed a bidirectional encoder based on self-attention Transformer mechanism that learned to predict masked items in user sequences using cloze completion objectives. The model uncovered randomly obscured items by capturing attention states from both left and right sides, considering short-term and long-term contexts. Chen et al. [8] introduced a dual utility model called Commodity Utility-Behavior Sequence (CUBS), comprising two components: Commodity Utility (CU) assessed consumers' psychological motivations and translated them into commodity utility, while BS predicted preferences by analyzing behavior sequences. Hou et al. [9] aimed at the limitation of explicitly modeling item IDs, proposed a novel generic sequential representation learning method that leveraged item-related descriptive text to learn transferable representations across different recommendation scenarios.

Most existing research on sequential recommendation is based on modeling user behavior data in single-domain scenarios without considering cross-domain recommendation situations under user cold start scenarios. In order to make up for this deficiency, this paper studies a sequential behavior data recommendation method based on cross-domain recommendation and proposes a multi-source sequential cross-domain recommendation algorithm based on deep learning (MSSCDR), which aims to further enhance recommendation accuracy under the premise of solving cold start problems and data sparsity problems.

MSSCDR can learn more accurate embedding representations by fitting interest forgetting curve and integrating multi-source information, then train a Multilayer Perceptron with learned source domain and target domain embeddings to provide more precise recommendations for cold start users.

In conclusion, we present the following contributions: (1) we consider the problem of insufficient utilization of sequential information in current cross-domain recommendation algorithms, and try to learn embedding representations of user sequential information, and strengthen embedding representations by fitting interest forgetting curve; (2) considering the instability of rating information, we fine-tune BERT to score user review information. We then merge the scored review data with the rating data to enhance the embedding learning process, significantly improving the algorithm's robustness; (3) we execute experiments on two authentic datasets compare with other algorithms. The experimental findings illustrate that the model introduced in this paper has more accurate recommendation effect.

The rest of this work is as follows: in Section 2, we introduce model structure and related techniques; in Section 3, we introduce experimental settings and improved model; in Section 4, we present experimental outcomes and analysis; in Section 5, we summarize our findings in this work.

## 2. Model

### *2.1. Problem description*

Cross-domain recommendation task is to transform user behavior characteristics in one domain into recommendations for content or resources in other domains; sequence recommendation task is to predict items or content that users may be interested in based on their historical behavior sequence. This study integrates the two subtasks of cross-domain recommendation and sequence recommendation, modeling them into an end-to-end joint task from source domain to target domain, and enriches the embedding representation in the joint task with fitting interest forgetting curve and multi-source information fusion ideas, striving to provide the most accurate personalized recommendation for cold start users. The task can be described as: given user behavior sequence information $U_a$={ $U_{1t_1}$ , $U_{2t_2}$ ,…, $U_{it_i}$ } and $V_a$={ $V_{1t_1}$ , $V_{2t_2}$ ,…, $V_{it_i}$ } of source domain and partial target domain, where $t_1<t_2<t_j, j \in (2,i]$, $U_{it_i}$ and $V_{it_i}$ represent user behavior at time $t_i$ in source domain and target domain respectively, learning user behavior embedding representations $U_{im}$ and $V_{in}$ of source domain and target domain through MSSCDR model, where $U_{im}$ represents embedding representation of m-th behavior of i-th user in source domain, $V_{in}$ represents embedding representation of n-th behavior of i-th user in target domain. This paper goal is to learn a bridging function through user behavior sequence embedding representations obtained from two domains, and then provide recommendations for cold start users.

### *2.2. Model structure*

Figure 1 depicts the complete structure of the model, comprising three levels: multi-source fusion layer, project embedding layer, and mapping layer. The input of the multi-source fusion layer is the comment matrix $U_b$ and $V_b$ of the source domain and target domain, and a BERT fine-tuning model is trained to score the comments between 1-5, resulting in a comment rating matrix $U_{b'}$ and $V_{b'}$ as the output of this layer.In the project embedding layer, the comment matrices $U_{b'}$ and $V_{b'}$ obtained from the fusion layer are combined with the rating matrices $U_a$ and $V_a$ by taking their averages. This gives us the source domain and target domain multi-element fusion matrices $U_s$ and $V_t$, then use Item2vec method to embed all projects into representations to obtain project embedding table $I_{U_p}$ and $I_{V_q}$, where $I_{U_p}$ represents source domain p-th project embedding representation, $I_{V_q}$ represents target domain q-th project embedding representation; then map each user's behavior sequence in project embedding table to obtain each user's behavior sequence embedding, considering that people's interests will change over time, borrowing Ebbinghaus memory forgetting curve idea, fitting interest forgetting function to weight user behavior sequence embedding, obtaining $U_{im}$ and $V_{in}$ as input for mapping layer. In mapping layer, use Multilayer Perceptron as mapping function to learn source domain and target domain embedding relationship and achieve cold start user embedding representation, then compare with project embedding table for similarity, select TopN as final recommendation results.

DOI: 10.54254/2755-2721/44/20230485

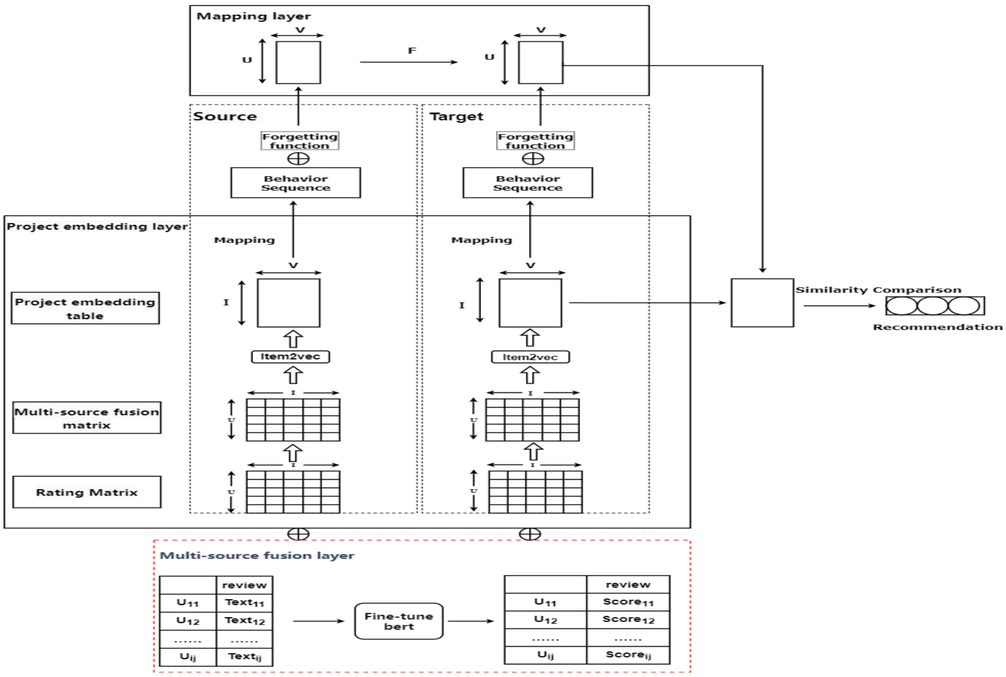



**Figure 1.** Model structure diagram.

## 2.3. Involving technology

*2.3.1. BERT.* BERT is a pre-trained language model designed for natural language processing tasks. BERT had been pre-trained on a large amount of text data to learn the language representation of the text, and adapted to different NLP tasks. BERT uses bidirectional pre-training, which means that it uses both left and right direction information of the text during the pre-training process, thus capturing more semantic relationships. It also uses “masked language modeling” pre-training, which randomly masks some words during the pre-training process and predicts these masked words. Fine-tuning BERT refers to retraining the already pre-trained BERT model on a specific task and dataset with a small learning rate and fewer training steps, so that it adapts to the features of the specific task and enhances the model’s performance on that particular task.

*2.3.2. Item2Vec.* Item2Vec is a recommendation algorithm based on Word2Vec that is used to generate embedding vector representations for items. Like Word2Vec, it uses the Skip-Gram model for training, but replaces words with items. During training, Item2Vec uses known user-item interaction data to generate embedding vectors, and the goal of these embedding vectors is to make similar items closer in the embedding space.

In this study, in order for our model to learn the embedding representation, we need to obtain the corresponding relationship between “words” and “sentences” from the data. In this case, we can treat each item as a “word”, and those items that are rated similarly by the same user are considered to be in the same “sentence”. More specifically, we can generate “sentences” as follows: For each user, generate two lists that store items that users “like” and “dislike”. The first list contains all items with ratings above 4 points, while the second list contains items with ratings below 4 points. These lists will serve as input for training the Gensim Word2Vec model to generate embedding representations for each item.

*2.3.3. Interest forgetting curve.* The interest forgetting curve stems from Ebbinghaus’s forgetting curve idea and applies it to recommendation systems. Ebbinghaus’s forgetting curve was discovered by German psychologist Ebbinghaus and describes how human brains forget new things. In recommendation systems, a series of user behaviors have different utilities for the final recommendation result due to different time intervals and order of behaviors. Therefore, Ebbinghaus's forgetting curve idea has great reference value for user interest changes in recommendation systems. The forgetting curve is expressed by an exponential formula. Here we choose a basic, widely accepted formula of exponential. As shown in Equation (1).

$$\mathrm{R} = \mathrm{e}^{-\mathrm{at}} \tag{1}$$

where t is the interval from user behavior to current time, $\alpha$ is interest forgetting speed, it is a non-negative parameter, $\alpha$ larger means faster forgetting speed with time passing, $\alpha$ equals 0 means this model is a uniform dispersion model. R describes the interest degree retained from an event occurrence to now. Chen et al. [10] proposed in their study of this model that the optimal setting of $\alpha$ is about 0.5. Based on this, in this study, we set $\alpha$ value of model uniformly as 0.5.

*2.3.4. Multilayer Perceptron.* In this experiment's mapping layer, we select Multilayer Perceptron as mapping function. Multilayer Perceptron is a machine learning model based on neural networks. It contains multiple fully connected layers (layers composed of multiple neurons), and each neuron uses nonlinear activation function to process input. It can be used for classification, regression and many other tasks. It can handle nonlinear problems. And you can add more hidden layers and neurons to increase model complexity. In this experiment, we construct a four-layer perceptron to learn mapping relationship, two hidden layers are set respectively 128 and 64 Neurons, dropout set as 0.2, loss function set as MAE, activation function set as ReLU, optimizer choose Adam.

## 3. Experimental setting

### *3.1. Description of the dataset*

The experimental data comes from Amazon review data, which is a set of datasets containing a large number of product reviews released by Amazon. These reviews come from different categories of products on the Amazon website. In this study, two groups of datasets were selected: CDs_and_Vinyl and Magazine_Subscriptions, Digital_Music and Movies_and_TV, some data were cut from them. The specific structure is shown in Table 1 and Table 2.

**Table 1.** Music-Movies dataset.

| | Music | Moives |
|---|---|---|
| data volume | 169623 | 500000 |
| user volume | 16561 | 166392 |
| Overlapping users | 3275 | 3275 |
| merchandise volume | 11797 | 7189 |
| rating value | {1,2,3,4,5} | {1,2,3,4,5} |
| sparsity | larger | lesser |

**Table 2.** CDs-Video dataset.

| | CDs | Video |
|---|---|---|
| data volume | 100000 | 200000 |
| user volume | 39874 | 39302 |
| Overlapping users | 1496 | 1496 |
| merchandise volume | 4406 | 8526 |
| rating value | {1,2,3,4,5} | {1,2,3,4,5} |
| sparsity | larger | lesser |

### 3.2. Evaluation standard

Mean Absolute Error (MAE), which signifies the mean of the absolute differences between the predicted and actual values. Its calculation formula is as follows (2).

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_u - y_u| \tag{2}$$

Root Mean Squared Error (RMSE), which signifies the root mean square deviation between the predicted model values and the actual values. Its calculation formula is as follows (3).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}((\hat{y}_u - y_u))^2} \tag{3}$$

Where n represents the number of cold start users, $\hat{y}_u$represents the embedding predicted value of user u, and $y_u$ represents the actual embedding value of user u.

### 3.3. Contrastive model

In this paper, in addition to comparing the proposed complete model with other excellent models, we also consider using the two weighting methods proposed in this paper: multi-source data fusion and fitting interest forgetting function separately to observe the degree of influence of these two methods on the recommendation effect. The following are all the models involved in this paper:

**EMCDR**: EMCDR technology enables the mapping of users and items from distinct domains into a common low-dimensional vector space. This allows users and items originating from distinct domains to share identical vector representations, thereby simplifying cross-domain recommendation [11].

**SSCDR**: SSCDR solves the recommendation problem for cold-start users by utilizing cross-domain data and semi-supervised learning. It improves the accuracy of cold-start user recommendation by sharing features and knowledge across different domains and training on partially labeled and unlabeled data [12].

**TMCDR**: TMCDR utilizes the idea of meta-learning, and by using a small amount of data from the target domain, it adapts and adjusts the knowledge and experience learned in the source domain. By doing so, it can improve the recommendation effect for cold-start users [13].

**MS-EMCDR**: Adding the multi-source data fusion mentioned in this paper on the basis of EMCDR.

**MSSCDR**: The complete model of multi-source serialization cross-domain recommendation proposed in this paper.

**MSSCDR(-FF)**: The complete model proposed in this paper does not consider fitting interest forgetting function.

**SCDR**: The complete model proposed in this paper does not consider multi-source data fusion.

**SCDR(-FF)**: SCDR model does not consider fitting interest forgetting function.

## 4. Experimental assessment and analysis

### 4.1. Fine-tune BERT during data preprocessing

In this study, we fine-tune BERT on the Yelp dataset to achieve rating of review data. The Yelp dataset is a large online business review dataset that contains businesses, reviews, and star ratings from around the world. Due to the large size of the original dataset, we cut 200K data to perform the BERT fine-tuning task, setting the transformer layer to 12 layers and the hidden layer dimension to 768 dimensions. The training setting has a learning rate of 3e-5, the input sequence's maximum length is set to 150, and the number of iterations is set to 20, the training batch size is set to 32, and the gradient accumulation parameter is set to 2. Two metrics are selected here to evaluate the performance of the fine-tuned model. Accuracy (exact) is the ratio of correctly predicted star ratings (PS) to total star ratings (TS), as shown in equation (4).

$$Accuracy(exact) = \frac{PS}{TS} \tag{4}$$

Accuracy (off-by-1) is the percentage of reviews where the star rating predicted by the model differs by at most 1 from the rating given by human reviewers.

Compared with nlptown's excellent model bert-base-multilingual-uncased-sentiment, our model's training results have improved on both metrics, as shown in Table 3.

**Table 3.** Accuracy (exact) and Accuracy (off by 1) of the two models.

| Model | Accuracy | Accuracy (off-by-1) |
|---|---|---|
| bert-base-multilingual-uncased-sentiment | 67% | 95% |
| bert-finetune-base-yelp20w | 72.716% | 96.88% |

*4.2. Experimental results and analysis*

In the experiment, the Music-Movies dataset was first used with Music as the source domain and Movie as the target domain, and the CDs-Video dataset was used with CD as the source domain and Video as the target domain. In order to assess the model's generalization capability, for the two datasets, then swap their original two domains, that is, for the Music-Movies dataset, Movie was used as the source domain and Music as the target domain; for the CDs-Video dataset, Video was used as the source domain and CD as the target domain. Furthermore, in order to test the performance of the model under different proportions of cold start users, the proportion of cold start users was set to: 20%, 30%, 40%, 50%. Table 4 and Table 5 show the RMSE and MAE of each model after experiments on the Music-Movies dataset.

**Table 4.** MAE and RMSE on Amazon dataset ($M_u$-$M_o$).

| | 20% | | 30% | | 40% | | 50% | |
|---|---|---|---|---|---|---|---|---|
| Model | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE |
| EMCDR | 0.2315 | 0.2771 | 0.2337 | 0.2786 | 0.2326 | 0.2783 | 0.2338 | 0.2892 |
| SSCDR | 0.2307 | 0.2768 | 0.2311 | 0.2779 | 0.2315 | 0.2781 | 0.2321 | 0.2813 |
| TMCDR | 0.2279 | 0.2738 | 0.2299 | 0.2753 | 0.2298 | 0.2761 | 0.2309 | 0.2874 |
| MS-EMCDR | 0.2220 | 0.2464 | 0.2229 | 0.2469 | 0.2253 | 0.2478 | 0.2257 | 0.2497 |
| SCDR(-FF) | 0.1183 | 0.1523 | 0.1171 | 0.1517 | 0.1174 | 0.1527 | 0.1180 | 0.1519 |
| MSSCDR(-FF) | 0.1073 | 0.1442 | 0.1067 | 0.1428 | 0.1070 | 0.1435 | 0.1071 | 0.1432 |
| SCDR | 0.0627 | 0.0885 | 0.0621 | 0.0883 | 0.0633 | 0.0886 | 0.0635 | 0.0900 |
| MSSCDR | 0.0584 | 0.0764 | 0.0581 | 0.0757 | 0.0582 | 0.0764 | 0.0584 | 0.0767 |

**Table 5.** MAE and RMSE on Amazon dataset ($M_o$-$M_u$).

| | 20% | | 30% | | 40% | | 50% | |
|---|---|---|---|---|---|---|---|---|
| Model | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE |
| EMCDR | 0.2678 | 0.3174 | 0.2675 | 0.3184 | 0.2679 | 0.3191 | 0.2677 | 0.3212 |
| SSCDR | 0.2639 | 0.3141 | 0.2642 | 0.3153 | 0.2646 | 0.3162 | 0.2651 | 0.3184 |
| TMCDR | 0.2601 | 0.3108 | 0.2605 | 0.3121 | 0.2609 | 0.3131 | 0.2614 | 0.3154 |
| MS-EMCDR | 0.2550 | 0.3061 | 0.2572 | 0.3074 | 0.2585 | 0.3078 | 0.2594 | 0.3105 |
| SCDR(-FF) | 0.2104 | 0.2717 | 0.2103 | 0.2736 | 0.2193 | 0.2722 | 0.2196 | 0.2718 |
| MSSCDR(-FF) | 0.1948 | 0.2548 | 0.1941 | 0.2546 | 0.1954 | 0.2546 | 0.1961 | 0.2555 |
| SCDR | 0.1856 | 0.2509 | 0.1853 | 0.2506 | 0.1857 | 0.2514 | 0.1858 | 0.2512 |
| MSSCDR | 0.1667 | 0.2404 | 0.1663 | 0.2383 | 0.1650 | 0.2401 | 0.1679 | 0.2385 |

From the results on the Music-Movies dataset, it becomes apparent that, regardless of whether the source domain and the target domain are swapped or not, compared with the baseline model, the other improved models proposed in this paper have decreased in both MAE and RMSE. MSSCDR, which is the complete model proposed in this paper, has the best performance, and the cold start users are 30%, the MAE and RMSE on the Music-Movies dataset reach the minimum values. Therefore, on the Music-Movies dataset, the model proposed in this paper has a better effect on recommending to cold start users than the baseline models.

Table 6 and Table 7 show the RMSE and MAE of each model after experiments on the CDs-Video dataset.

**Table 6.** MAE and RMSE on Amazon dataset ($M_c$-$M_v$).

| | 20% | | 30% | | 40% | | 50% | |
|---|---|---|---|---|---|---|---|---|
| Model | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE |
| EMCDR | 0.2580 | 0.3146 | 0.2586 | 0.3145 | 0.2555 | 0.3116 | 0.2620 | 0.3172 |
| SSCDR | 0.2569 | 0.3134 | 0.2573 | 0.3146 | 0.2577 | 0.3158 | 0.2583 | 0.3180 |
| TMCDR | 0.2535 | 0.3103 | 0.2539 | 0.3116 | 0.2544 | 0.3129 | 0.2550 | 0.3152 |
| MS-EMCDR | 0.2521 | 0.2932 | 0.2523 | 0.2932 | 0.2515 | 0.2877 | 0.2546 | 0.2946 |
| SCDR(-FF) | 0.1809 | 0.2480 | 0.1800 | 0.2484 | 0.1810 | 0.2482 | 0.1811 | 0.2476 |
| MSSCDR(-FF) | 0.1108 | 0.1889 | 0.1107 | 0.1889 | 0.1122 | 0.1897 | 0.1127 | 0.1897 |
| SCDR | 0.0927 | 0.1296 | 0.0924 | 0.1285 | 0.0930 | 0.1311 | 0.0940 | 0.1315 |
| MSSCDR | 0.0892 | 0.1221 | 0.0891 | 0.1208 | 0.0902 | 0.1249 | 0.0906 | 0.1253 |

**Table 7.** MAE and RMSE on Amazon dataset ($M_v$-$M_c$).

| Model | 20% | | 30% | | 40% | | 50% | |
|---|---|---|---|---|---|---|---|---|
| | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE |
| EMCDR | 0.2752 | 0.3268 | 0.2776 | 0.3242 | 0.2717 | 0.3239 | 0.2755 | 0.3282 |
| SSCDR | 0.2741 | 0.3256 | 0.2745 | 0.3268 | 0.2749 | 0.3280 | 0.2755 | 0.3288 |
| TMCDR | 0.2707 | 0.3225 | 0.2711 | 0.3238 | 0.2716 | 0.3251 | 0.2722 | 0.3274 |
| MS-EMCDR | 0.2637 | 0.3092 | 0.2648 | 0.3074 | 0.2647 | 0.3045 | 0.2638 | 0.3109 |
| SCDR(-FF) | 0.1888 | 0.2492 | 0.1875 | 0.2468 | 0.1899 | 0.2510 | 0.1919 | 0.2508 |
| MSSCDR(-FF) | 0.1122 | 0.1989 | 0.1119 | 0.1984 | 0.1139 | 0.2000 | 0.1143 | 0.2012 |
| SCDR | 0.0961 | 0.1691 | 0.0952 | 0.1658 | 0.0964 | 0.1679 | 0.0974 | 0.1664 |
| MSSCDR | 0.0913 | 0.1665 | 0.0906 | 0.1636 | 0.0923 | 0.1695 | 0.0925 | 0.1692 |

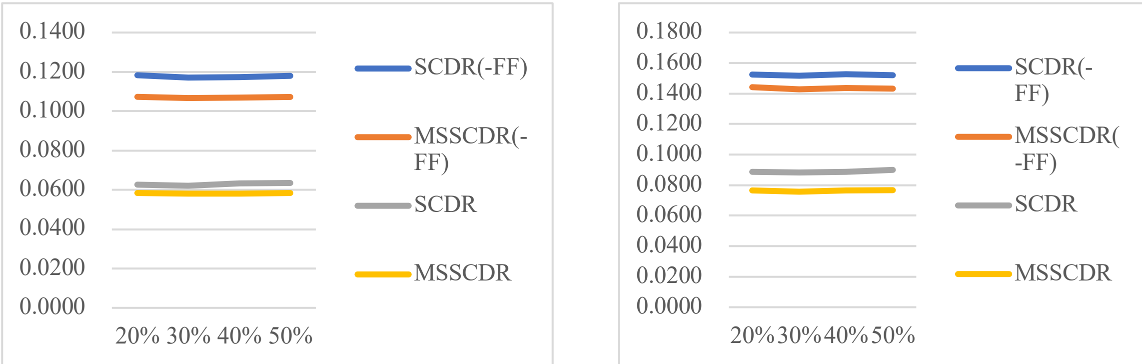
In the experiments on the CDs-Video dataset, the performance of each model is basically consistent with that on the Music-Movies dataset. All the improved models have decreased in both MAE and RMSE. The MSSCDR model still has the best performance. When the cold start users are 30%, the MAE and RMSE on the CDs-Video dataset still reach the minimum values. Therefore, it can prove that the model proposed in this paper has a certain robustness.

By comparing the two sets of experimental results in the two datasets, it can be found that after swapping the source domain and the target domain, using the denser dataset as the source domain and the sparser dataset as the target domain, the MAE and RMSE values are lower. And after swapping the source domain and the target domain in the Music-Movies dataset, the gap between the MAE and RMSE values is larger than that after swapping the source domain and the target domain in the Cds-Video dataset. This is mainly because the difference in sparsity between the two domains in the Music-Movie dataset is greater than that in the Cds-Video dataset.

### *4.3. The effect of two weighting methods on cross-domain recommendation*

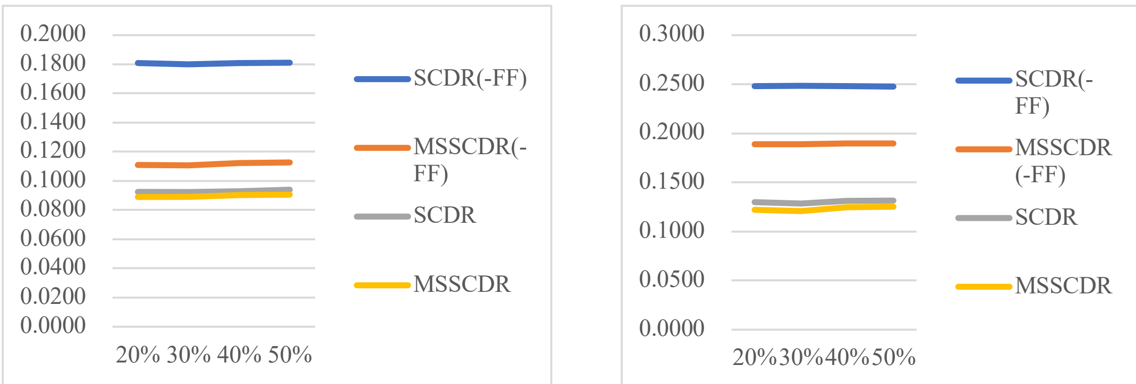
In this paper, we propose two weighting methods: multi-source data fusion and fitting interest forgetting function weighting. Multi-source data fusion is to weight the original rating data after transforming the comments into ratings. Fitting interest forgetting function weighting is to weight the historical behavior data of each user by fitting the interest forgetting curve. In the previous section, the MSSCDR model obtained by using these two methods together had shown good results. This section will analyze the effects of these two methods on cross-domain recommendation separately.

Figure 2 and Figure 3 are the comparisons of MAE and RMSE values of four improved models on Music-Movies dataset and CDs-Video dataset, each group contains two figures, where (a) figure represents MAE, (b) figure represents RMSE.




(a) (b)

**Figure 2.** The impact of interest forgetting function and multi-source data fusion on Music-Movies dataset.




(a) (b)

**Figure 3.** The impact of interest forgetting function and multi-source data fusion on CDs-Video dataset.

From the two groups of figures, it is evident that using multi-source data fusion and fitting interest forgetting function weighting separately will reduce MAE and RMSE to a certain extent, but their impact effects have large differences, SCDR(-FF) and MSSCDR(-FF), SCDR and MSSCDR between the difference compared to SCDR(-FF) and SCDR, MSSCDR(-FF) and MSSCDR difference is generally smaller, this shows that fitting interest forgetting function weighting for cross-domain recommendation effect is better than multi-source data fusion, this is because multi-source data fusion is to increase the robustness of source data to a certain extent, does not directly affect the final recommendation effect, and fitting interest forgetting function weighting will give higher weight to the user's recent interaction behavior, and the user's recent behavior will generally dominate the user's next few times behavior.

## 5. Conclusion

This paper proposes a multi-source serialization cross-domain recommendation model based deep learning. This model avoids the drawbacks of traditional cross-domain recommendation tasks that mostly use static and single data of users, by utilizing sequential information and multi-source information. It effectively improves the overall effectiveness of the task. The proposed model is tested on two groups of datasets from Amazon, and the results obtained have decreased in the relevant evaluation metrics compared to the excellent model. Meanwhile, this paper analyzes the selection of source domains and target domains with different sparsity degrees on the proposed model, and proves that the recommendation effect of dense domains on sparse domains is better. Finally, this paper also separately analyzes the impact of the proposed fitting interest forgetting function and multi-source data fusion techniques on the model, and proves that both can reduce MAE and RMSE values, and the effect of fitting interest forgetting function is greater than that of multi-source data fusion. Multiple experiments demonstrate that the model presented in this paper is rational, efficient, and practical.

**References**

[1] Liu Huiting,Guo Lingling,Li Peipei,Zhao Peng,Wu Xindong. Collaborative filtering with a deep adversarial and attention network for cross-domain recommendation[J]. Information Sciences,2021,565.
[2] Zhang L, Ge Y, Ma J, et al. Knowledge-aware Neural Collective Matrix Factorization for Cross-domain Recommendation[J]. arXiv preprint arXiv:2206.13255, 2022.
[3] Li P, Brost B, Tuzhilin A. Adversarial Learning for Cross Domain Recommendations[J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2022.
[4] Sahu A K. User and item spaces transfer from additional domains for cross-domain recommender systems[J]. Applied Intelligence, 2023, 53(5): 5766-5783.
[5] Ai Z, Wang S, Jia S, et al. Core Interests Focused Self-attention for Sequential Recommendation[C]//International Conference on Database Systems for Advanced Applications. Springer, Cham, 2022: 306-314.
[6] Guo J, Ji W, Yuan J, et al. Multi-channel Orthogonal Decomposition Attention Network for Sequential Recommendation[C]//Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Cham, 2022: 288-300.
[7] Kheldouni A, Boumhidi J. A Study of Bidirectional Encoder Representations from Transformers for Sequential Recommendations[C]//2022 International Conference on Intelligent Systems and Computer Vision (ISCV).IEEE, 2022: 1-5.
[8] Chen L, Zhu H. Behavior prediction based on a Commodity Utility-Behavior Sequence model[J]. Machine Learning with Applications, 2022, 9: 100314.
[9] Hou Y, Mu S, Zhao W X, et al. Towards Universal Sequence Representation Learning for Recommender Systems[C]//Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2022: 585-593.
[10] Chen J, Wang C, Wang J. Modeling the interest-forgetting curve for music recommendation[C]//Proceedings of the 22nd ACM international conference on Multimedia. 2014: 921-924.
[11] Man T, Shen H, Jin X, et al. Cross-domain recommendation: An embedding and mapping approach[C]//IJCAI. 2017, 17: 2464-2470.
[12] Kang S K, Hwang J, Lee D, et al. Semi-supervised learning for cross-domain recommendation to cold-start users[C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2019: 1563-1572.
[13] Zhu Y, Ge K, Zhuang F, et al. Transfer-meta framework for cross-domain recommendation to cold-start users[C]//Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2021: 1813-1817.