Transfer-Meta Framework for Cross-domain Recommendation to Cold-Start Users

Yongchun Zhu[1,2,3], Kaikai Ge[3], Fuzhen Zhuang[4,5,*], Ruobing Xie[3], Dongbo Xi[1,2], Xu Zhang[3],
Leyu Lin[3] and Qing He[1,2]

[1]Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing
Technology, CAS, Beijing 100190, China
[2]University of Chinese Academy of Sciences, Beijing 100049, China
[3]WeChat Search Application Department, Tencent, China
[4]Institute of Artificial Intelligence, Beihang University, Beijing 100191, China
[5]Xiamen Data Intelligence Academy of ICT, CAS, China
{zhuyongchun18s, xidongbo17s, heqing}@ict.ac.cn, {kavinge, ruobingxie, xuonezhang,
goshawklin}@tencent.com,zhuangfuzhen@buaa.edu.cn

ABSTRACT

Cold-start problems are enormous challenges in practical recommender systems. One promising solution for this problem is cross-domain recommendation (CDR) which leverages rich information from an auxiliary (source) domain to improve the performance of recommender system in the target domain. In these CDR approaches, the family of Embedding and Mapping methods for CDR (EMCDR) is very effective, which explicitly learn a mapping function from source embeddings to target embeddings with overlapping users. However, these approaches suffer from one serious problem: the mapping function is only learned on limited overlapping users, and the function would be biased to the limited overlapping users, which leads to unsatisfying generalization ability and degrades the performance on cold-start users in the target domain. With the advantage of meta learning which has good generalization ability to novel tasks, we propose a transfer-meta framework for CDR (TMCDR) which has a transfer stage and a meta stage. In the transfer (pre-training) stage, a source model and a target model are trained on source and target domains, respectively. In the meta stage, a task-oriented meta network is learned to implicitly transform the user embedding in the source domain to the target feature space. In addition, the TMCDR is a general framework that can be applied upon various base models, e.g., MF, BPR, CML. By utilizing data from Amazon and Douban, we conduct extensive experiments on 6 cross-domain tasks to demonstrate the superior performance and compatibility of TMCDR.

CCS CONCEPTS

• **Information systems → Recommender systems**.

KEYWORDS

Cross-domain Recommendation; Meta Learning; Cold-start

1 INTRODUCTION

In the era of information explosion, how to efficiently obtain useful information from massive data is very important. Recommender systems play a major role in alleviating information overload. However, it is difficult to make cold-start recommendations, e.g., new users (user cold-start) and new items (item cold-start). Cross-domain recommendation (CDR) [8, 11, 14] is a promising solution to address the cold-start problem.

CDR which leverages rich information from an auxiliary (source) domain to improve the performance of recommender system in the target domain has gained increasing attention in recent years. Actually, most CDR methods [4, 5, 14, 17, 18] aim to improve the overall performance for a target domain with the help of a source domain. Other methods address the cold-start problem which is more technically challenging as well as has great values from a practical perspective [12]. To address the problem, the Embedding and Mapping approach for CDR (EMCDR) [11] is very effective, which encodes user's preferences of source and target domains on items into two embeddings, respectively, and then **explicitly** learns a mapping function from source embeddings to target embeddings with overlapping users. In other words, it minimizes the distance between the target embedding and the approximated embedding mapped from the source embedding for each overlapping user with Mean Squared Error (MSE) loss. With the advantage of EMCDR, many EMCDR-based approaches [3, 8, 19] have been proposed.

However, most of the EMCDR-based methods suffer from one serious problem. These methods explicitly learn the mapping function by minimizing the distance between the target embedding and the mapped embedding of the overlapping users. In other words,

the number of training samples to learn the mapping function is equal to the number of overlapping users. In practice, the average ratio of the overlapping users to total users of any two domains is very low [8], e.g., in our experiments, the cross-domain tasks have 37388 overlapping users at most and 894 at least. Hence, the mapping function would be biased to the limited overlapping users, which leads to unsatisfying generalization ability and degrades the performance of the model on cold-start users in the target domain.

Meta learning [1, 2] has good generalization ability to novel tasks by training the model on lots of similar training tasks. With the advantage of meta learning, we propose a transfer-meta framework for CDR (TMCDR) which can replace the training procedure of most EMCDR-based methods [3, 8, 11, 19] and be applied upon various base models, e.g., MF [10], BPR [15], CML [7]. In detail, the proposed TMCDR has a transfer stage and a meta stage.

Transfer stage: The transfer stage is similar to the embedding step in EMCDR [11]. However, the embedding step in EMCDR learns a source model and a target model with the user-item interaction of only overlapping users from scratch. However, with limited overlapping users, it is hard to learn the embedding of items which has not been interacted by the overlapping users. In practice, each domain has a unique model for the overall recommendation. To address the problem, the unique model is directly utilized as pre-trained model. Thus, this stage is called as transfer stage. Note that the models should be embedding-based methods [6, 10, 15].

Meta stage: The main idea of the meta stage is that a task-oriented meta network is learned to **implicitly** transform the overlapping user embedding in the source domain to the target feature space. The optimization goal of the cold-start task in CDR is learning knowledge from overlapping users and generalizing to cold-start users in the target domain. Inspired by meta learning (learning to learn), we construct special training tasks to simulate the target task. Each training task consists of a learning phase and a cold-start phase. The learning phase learns knowledge from one overlapping user. The cold-start phase uses another overlapping user to simulate a cold-start user. With the advantage of the popular Model-Agnostic Meta-Learning (MAML) [2], we proposed a gradient-based training method to optimize a meta network. Different from the mapping function of EMCDR, the meta network is task-oriented, which denotes that the optimization goal is rating or ranking, not mapping.

The main contributions of this work are summarized into three folds: (1) To solve the cold-start problem in CDR, we propose a novel transfer-meta framework (TMCDR), which can be applied on most EMCDR-based methods with various base models, e.g., MF, BPR, CML. (2) The proposed method is easy to implement in the online cold-start setting. With existing pre-trained models, once the task-oriented meta network is trained, it can be exploited for cold-start users.

2 MODEL

2.1 Transfer Stage

In practice, each domain has a unique model for the overall recommendation, and we use the source and target models as pre-trained models. The pre-trained model is trained on all data. Compared

---

Algorithm 1 Transfer-Meta framework for CDR (TMCDR)

Input: Given user and item sets of source and target domains, U^s, U^t, V^s, V^t. The overlapping user set U^o. The rating matrix R^s, R^t.

Input: Task-oriented meta network f_θ.

Input: The step size (learning rate) λ, α.

Transfer Stage:
- A pre-trained source model contains u^s, v^s.
- A pre-trained target model contains u^t, v^t.

Meta Stage: utilize the source embedding of overlapping users u^s and the target item embedding v^t to optimize the task-oriented meta network f_θ.

(1) randomly initialize θ.
(2) **while** not converge **do**:
(3)   sample batch of user groups $\{U_1, ..., U_n\}$ from U^o.
(4)   **for** $U_i \in \{U_1, ..., U_n\}$ do:
(5)   divide U_i into two disjoint sets U_a, U_b
(6)   define two training sets D_a, D_b with U_a, U_b
(7)   evaluate loss $\mathcal{L}_\theta$ with D_a
(8)   compute updated parameter $\theta' = \theta - \lambda \frac{\partial \mathcal{L}_\theta}{\partial \theta}$
(9)   evaluate loss $\mathcal{L}_{\theta'_i}$ with D_b
(10)   update $\theta = \theta - \alpha \sum_{U_i \in \{U_1, ..., U_n\}} \frac{\partial \mathcal{L}_{\theta'_i}}{\partial \theta}$
(11) **end while**

Test Stage: for a cold-start user u, we use $f_\theta(u^s)$ as the user embedding for prediction.

---

with the embedding model of EMCDR trained on samples of overlapping users, the pre-trained model has two advantages: 1) with more data, the model is more robust, which is hard to be bothered by data noise. 2) In EMCDR, the model can only learn embeddings of items interacted by the overlapping users. In contrast, with all data, the pre-trained model can capture information from all items. Hence, our TMCDR directly utilizes the pre-trained model. In our experiments, we simulate the pre-training procedure, i.e., a model is trained with all data as the pre-trained model. In practice, various models are exploited for different applications. To testify our TMCDR is compatible with various models, we implement four popular embedding models into TMCDR framework, including MF [10], BPR [15], ListRank-MF [16], and CML [7]. Note that we have defined a rating matrix $R \in \{0, 1\}$ above, and the problem is a binary classification task.

2.2 Meta Stage

After the transfer stage, we can obtain the pre-trained source and target models (the users' and items' embeddings u^s, v^s, u^t, v^t). With the pre-trained embeddings fixed, we put forward a task-oriented meta network that can implicitly transform the source embeddings into the target feature space.

Recall that the main idea of cold-start in CDR is to learn knowledge from overlapping users and generalizing to cold-start users in the target domain. Meta learning [1, 2] has good generalization ability to novel tasks by training the model on a variety of similar training tasks. Inspired by meta learning, to simulate the target

task, we construct similar training tasks, and each training task consists of a learning phase and a cold-start phase.

The goal of the learning phase is to simulate that learning knowledge from overlapping users. Hence, each learning phase contains all user-item interaction samples of an overlapping user. In addition, the cold-start phase is to simulate the cold-start users in the target domain. However, the real cold-start users have no interaction behavior in the target domain, and the model cannot directly learn knowledge from these users. Thus, we utilize another overlapping user to simulate a cold-start user.

Each training task only has training samples of two users. This suffers from a challenge that a user could have very limited interaction samples. We find that it may lead to unstable training. In [13], they use users whose number of interaction samples exceeds a certain threshold for training. In this paper, we propose a group strategy that both the learning phase and cold-start phase contain several users, but the users of the two phases are disjoint. We denote the users of the learning phase and cold-start phase as U_a and U_b, respectively. Two training sets in the target domain with all samples of U_a and U_b are denoted as D_a and D_b, respectively.

We define meta network as $f_\theta(\cdot)$, and θ denotes the parameters. Besides, $f_\theta(u_i^s)$ represents the transformed embedding of u_i^s. The meta network should be optimized on lots of training tasks. We define the loss function to be the same as the optimization goal of the pre-training task, so we call it task-oriented loss. The pre-training task could be one of MF [10], BPR [15], ListRank-MF [16], and CML [7] as mentioned above. **Task-oriented loss utilizes the transformed user embedding** $f_\theta(u_i^s)$ **not** u_i^t. The overall training procedure of the task-oriented meta network is following the meta learning paradigm [2]. Firstly, the loss of the learning phase can be formulated as:

$$\mathcal{L}_\theta = \sum_{x \in D_a} L_{task}(x), \tag{1}$$

where $task \in MF, BPR, ListRank-MF, CML$, and $x \in D_a$ denotes one sample. By computing the gradient of $\mathcal{L}_\theta$ and taking a step of gradient descent, we get a new adapted parameter:

$$\theta' = \theta - \lambda \frac{\partial \mathcal{L}_\theta}{\partial \theta}, \tag{2}$$

where $\lambda > 0$ is the step size of gradient descent (learning rate). Now that we have a new parameter θ' which is trained with the overlapping users U_a, and we can test the adapted model $f_{\theta'}$ on the cold-start users U_b. Similarly, the loss of the cold-start phase is:

$$\mathcal{L}_{\theta'} = \sum_{x \in D_b} L_{task}(x). \tag{3}$$

Then we minimize the $\mathcal{L}_{\theta'}$ to update θ:

$$\theta = \theta - \alpha \frac{\partial \mathcal{L}_{\theta'}}{\partial \theta} \tag{4}$$

where $\frac{\partial \theta'}{\partial \theta}$ can be computed by the Equation (2). Note that the meta-optimization is performed over the model parameters θ, whereas the objective is computed using the updated model parameters θ'. Actually, the meta stage aims to optimize the parameters of task-oriented meta network such that one or a small number of gradient steps on a group of simulated cold-start users will produce maximally effective behavior on that the real-world cold-start users.

Table 1: Recommendation performance on 6 CDR tasks. * indicates 0.05 level, paired t-test of TMCDR_MF vs. the best baselines.

Method	AUC	NDCG@10	AUC	NDCG@10
	Scenario1		Scenario2	
CMF	0.6490	0.1696	0.6996	0.2076
BPR	0.7226	0.2182	0.7160	0.2379
ListRank-MF	0.6648	0.1709	0.7232	0.2204
CML	0.6470	0.1408	0.6986	0.2147
CST	0.7240	0.2137	0.7124	0.2324
SSCDR	0.7245	0.0089	0.6745	0.0013
EMCDR_MFori	0.6942	0.1978	0.6511	0.1747
EMCDR_MF	0.7271	0.2103	0.6923	0.1985
TMCDR_MF	**0.7501***	**0.2246***	**0.7253***	**0.2427***
	Scenario3		Scenario4	
CMF	0.7769	0.3066	0.7295	0.2349
BPR	0.7737	0.3065	0.7199	0.2150
ListRank-MF	0.7640	0.2902	0.7409	0.2277
CML	0.8191	**0.3548**	0.7857	0.2647
CST	0.7995	0.2960	0.7842	0.2563
SSCDR	0.7956	0.3080	0.6545	0.1628
EMCDR_MFori	0.7273	0.2284	0.7307	0.1990
EMCDR_MF	0.8011	0.3055	0.7936	0.2670
TMCDR_MF	**0.8282***	0.3334	**0.8056***	**0.2775***
	Scenario5		Scenario6	
CMF	0.8465	0.3420	0.8339	0.3764
BPR	0.8108	0.3283	0.8138	0.3659
ListRank-MF	0.8136	0.3106	0.8191	0.3281
CML	0.8466	0.3409	0.8405	0.3707
CST	0.8524	0.3405	0.8406	0.3742
SSCDR	0.8144	0.2925	0.8317	0.3644
EMCDR_MFori	0.7307	0.1990	0.7627	0.2703
EMCDR_MF	0.8438	0.3322	0.8297	0.3702
TMCDR_MF	**0.8589***	**0.3483***	**0.8442***	**0.3778***

Finally, we come to the overall training algorithm of TMCDR, which can update the meta-parameters by stochastic gradient descent in a mini-batch manner, see Algorithm 1.

3 EXPERIMENTS

3.1 Experimental Settings

Dataset. Two real-world datasets are adopted for evaluation, Amazon[1] and Douban[2]. Both datasets have been used for the CDR problem.

The first dataset is a public Amazon dataset, which has various versions. And we use the Amazon-5cores dataset that each user or item has at least five ratings. The dataset contains 24 different item domains. Among them, we choose the seven popular categories: apps_for_android, video_games, home_and_kitchen,

---

[1] http://jmcauley.ucsd.edu/data/amazon/
[2] https://www.douban.com

movies_and_tv, cds_and_vinyl, books and tools_and_home_improvement. Then, we define four CDR scenarios as Scenario 1: apps_for_android → video_games, Scenario 2: home_and_kitchen → tools_and_home_improvement, Scenario 3: movies_and_tv → cds_and_vinyl, and Scenario 4: books → movies_and_tv.

The second dataset is Douban dataset, which includes three recommendation tasks like movies, music, and book recommendations. We utilize the Douban dataset to construct another two cross-domain tasks: Scenario 5: movie → music, and Scenario 6: music → book.

Evaluation Protocol: To evaluate the performance of the proposed framework on the CDR tasks for cold-start users, for each task, we randomly select about 20% overlapping users as cold-start users, and use all samples of these users in target domain for the test stage. we adopt three standard metrics, AUC and NDCG@K, which are widely used in recommendation [4, 11] to evaluate the ranking performance of each method.

Baselines: The baselines can be divided into two groups: single-domain and cross-domain. In the first group, we consider both source and target domains as a single domain and utilize popular CF method, including CMF [17], BPR [15], ListRank-MF [16], and CML [7]. The second group includes state-of-the-art CDR methods for cold-start users, including CST [14], EMCDR [11], and SS-CDR [8]. Both EMCDR and our TMCDR are general frameworks for many embedding models. Thus, we apply EMCDR and TMCDR on MF, BPR, ListRank-MF, and CML, e.g., TMCDR_MF.

Implementation Details: For simplicity, we intentionally transform the rating data into binary (1/0 indicate whether a user has interacted with an item or not) to fit the problem setting of implicit feedback following [4]. In the training stage, for each positive sample, we randomly sample 4 negative samples. For all methods, we set the dimension of embedding as 256, and mini-batch size of 1280. We employ the Adam [9] optimizer and search its learning rate within $\{0.001, 0.002, 0.005, 0.01\}$. We fix $\lambda = 0.005$ in meta stage. For all EMCDR-based methods and TMCDR, we use a single fully connected layer as mapping function and meta network. We report the average AUC and NDCG@10 on three random trials.

3.2 Results

Recommendation Performance. We demonstrate the effectiveness of TMCDR on six CDR tasks. The experimental results evaluated by AUC and NDCG@10 are shown in Table 1. The experimental results reveal several insightful observations. (1) With various CDR scenarios, TMCDR outperforms most compared methods which demonstrates the effectiveness of TMCDR. The improvement mainly comes from the task-oriented meta network. (2) On all CDR scenarios, EMCDR_MF largely outperforms EMCDR$_{ori}$_MF. The main reason is that the overlapping users only cover a part of items, and the models cannot learn other items which have not been interacted by overlapping users. It also demonstrates that using all samples is more effective than only using samples of overlapping users. (3) We can find that CMF, BPR, ListRank-MF, and CML have different performances on various scenarios, which testifies different tasks should adopt different models. Especially in Scenario 3, CML outperforms all methods on NDCG@10. (4) The confidence intervals of results on the Douban dataset are smaller than the Amazon dataset. We think the reason is that Douban dataset has more
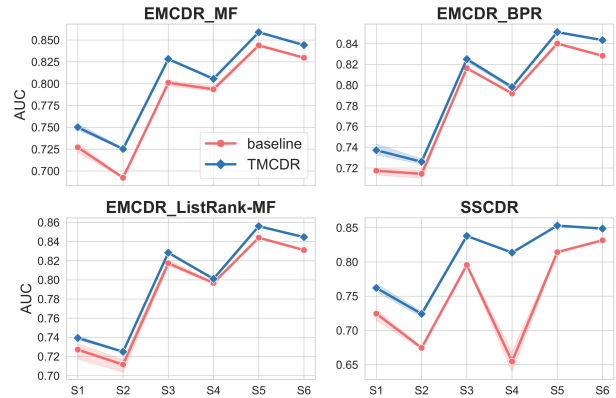

Figure 1: Generalization experiments.

overlapping users, which makes the training process more stable. (5) CST only utilizes the source model as a pre-trained model and regularizes the parameters can obtain remarkable results, which demonstrates fine-tuning recommendation models on a pre-trained model is effective.

Generalization. Our TMCDR is a general framework that can be applied on most EMCDR-based methods with various base models, e.g., MF, BPR, ListRank-MF, and CML. We compare four EMCDR-based methods, EMCDR_MF, EMCDR_BPR, EMCDR_ ListRank-MF, and SSCDR. Note that SSCDR is a combination of CML and EMCDR, so we do not compare TMCDR with EMCDR_CML. To implement our TMCDR, the mapping function of these methods is replaced by the task-oriented meta network. The results are drawn in Figure 1, and the red lines denote the baselines (EMCDR-based methods), while the blue lines represent the modified methods (TMCDR). And, the bands are confidence intervals over three runs. From the Figure 1, we can find that on most tasks our TMCDR can improve the performance of various EMCDR-based methods, which demonstrates the generalization ability of the TMCDR.

4 CONCLUSION

In this paper, we studied CDR to cold-start users from the perspective of meta learning. EMCDR is a popular approach in this area, and there are many EMCDR-based methods. However, with limited overlapping users, the mapping function would be overfitting. To address the problem, with the advantage of MAML, we proposed a novel Transfer-Meta Framework for CDR (TMCDR), which learns a task-oriented meta network. Besides, TMCDR is a general framework that can be applied on most EMCDR-based methods with various base models, e.g., MF, BPR, CML. Finally, we conducted extensive experiments on real-world datasets collected from Amazon and Douban to validate the effectiveness and compatibility of our proposed TMCDR.

REFERENCES

[1] Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. 2016. Learning to learn by gradient descent by gradient descent. In *NeurIPS*. 3981–3989.

[2] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*. JMLR. org, 1126–1135.

[3] Wenjing Fu, Zhaohui Peng, Senzhang Wang, Yang Xu, and Jin Li. 2019. Deeply Fusing Reviews and Contents for Cold Start Users in Cross-Domain Recommendation Systems. In *AAAI*, Vol. 33. 94–101.

[4] Chen Gao, Xiangning Chen, Fuli Feng, Kai Zhao, Xiangnan He, Yong Li, and Depeng Jin. 2019. Cross-domain Recommendation Without Sharing User-relevant Data. In *WWW*. 491–502.

[5] Jia He, Rui Liu, Fuzhen Zhuang, Fen Lin, Cheng Niu, and Qing He. 2018. A General Cross-Domain Recommendation Framework via Bayesian Neural Network. In *ICDM*. IEEE, 1001–1006.

[6] Xiangnan He and Tat-Seng Chua. 2017. Neural factorization machines for sparse predictive analytics. In *SIGIR*. 355–364.

[7] Cheng-Kang Hsieh, Longqi Yang, Yin Cui, Tsung-Yi Lin, Serge Belongie, and Deborah Estrin. 2017. Collaborative metric learning. In *WWW*. 193–201.

[8] SeongKu Kang, Junyoung Hwang, Dongha Lee, and Hwanjo Yu. 2019. Semi-Supervised Learning for Cross-Domain Recommendation to Cold-Start Users. In *CIKM*. 1563–1572.

[9] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*, Vol. 5.

[10] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[11] Tong Man, Huawei Shen, Xiaolong Jin, and Xueqi Cheng. 2017. Cross-Domain Recommendation: An Embedding and Mapping Approach. In *IJCAI*. 2464–2470.

[12] Nima Mirbakhsh and Charles X Ling. 2015. Improving top-n recommendation for cold-start users via cross-domain information. *TKDD* 9, 4 (2015), 1–19.

[13] Feiyang Pan, Shuokai Li, Xiang Ao, Pingzhong Tang, and Qing He. 2019. Warm Up Cold-start Advertisements: Improving CTR Predictions via Learning to Learn ID Embeddings. In *SIGIR*. 695–704.

[14] Weike Pan, Evan Wei Xiang, Nathan Nan Liu, and Qiang Yang. 2010. Transfer learning in collaborative filtering for sparsity reduction. In *AAAI*.

[15] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*. 452–461.

[16] Yue Shi, Martha Larson, and Alan Hanjalic. 2010. List-wise learning to rank with matrix factorization for collaborative filtering. In *Recsys*. 269–272.

[17] Ajit P Singh and Geoffrey J Gordon. 2008. Relational learning via collective matrix factorization. In *KDD*. 650–658.

[18] Dongbo Xi, Fuzhen Zhuang, Yongchun Zhu, Pengpeng Zhao, Xiangliang Zhang, and Qing He. 2020. Graph Factorization Machines for Cross-Domain Recommendation. *arXiv preprint arXiv:2007.05911* (2020).

[19] Feng Zhu, Yan Wang, Chaochao Chen, Guanfeng Liu, Mehmet Orgun, and Jia Wu. 2018. A deep framework for cross-domain and cross-system recommendations. In *IJCAI*.