Dual Autoencoder Network with Swap Reconstruction for Cold-Start Recommendation

Bei Wang*[1], Chenrui Zhang[1], Hao Zhang[2], Xiaoqing Lyu*[1], Zhi Tang[1]
[1] Wangxuan Institute of Computer Technology, Peking University, Beijing, China
[2] Beijing Institute of Technology, Beijing, China
{wang_bei,chenrui.zhang,lvxiaoqing,tangzhi}@pku.edu.cn,gcrth@outlook.com

ABSTRACT

Cold-start is a long-standing and challenging problem in recommendation systems. To tackle this issue, many cross-domain recommendation approaches are proposed. However, most of them follow a two-stage embedding-and-mapping paradigm, which is hard to be optimized. Besides, they ignore the structure information of the user-item interaction graph, resulting in that the embedding is insufficient to capture the latent collaborative filtering effect. In this paper, we propose a Dual Autoencoder Network (DAN), which implements cross-domain recommendations to cold-start users in an end-to-end manner. The graph convolutional network (GCN) based encoder in DAN explicitly captures high-order collaborative information in user-item interaction graphs. The two-branched decoder is proposed for fully exploiting the data across domains, and therefore the elaborate reconstruction constraints are obtained under a domain swapping strategy. Experiments on two pairs of real-world cross-domain datasets demonstrate that DAN outperforms existing state-of-the-art methods.

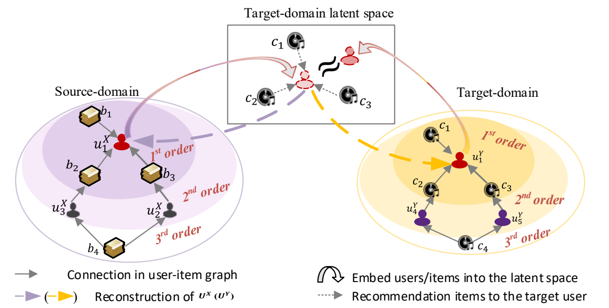

Figure 1: Illustration of our solution for cross-domain recommendation. The purple circle and yellow circle denote the book-domain (the source domain) and the CD-domain (the target domain), respectively. Transferred overlapping user u_1 is drived to be close to the ground-truth in the latent space of the target-domain for accurate recommendation via designed reconstruction constrains.

CCS CONCEPTS

• **Information systems → Recommender systems**; • **Computing methodologies → Neural networks**.

KEYWORDS

Cold-Start Recommendation, Dual Autoencoder Network, Swap Reconstruction, High-order Connectivity

ACM Reference Format:
Bei Wang*[1], Chenrui Zhang[1], Hao Zhang[2], Xiaoqing Lyu*[1], Zhi Tang[1]. 2020. Dual Autoencoder Network with Swap Reconstruction for Cold-Start Recommendation. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20), October 19–23, 2020, Virtual Event, Ireland.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3340531.3412069

1 INTRODUCTION

With an increasing number of users interacting with multi-modal information spanning from multiple domains, cross-domain recommendation (CDR) has increased more attention to alleviating the cold-start and data sparsity problem by leveraging information from multiple domains [1, 2, 5, 6, 9, 10, 13]. Most existing CDR methods [1, 5] are designed to improve the recommendation quality of a target domain with the help of the overlapped users who also exist in other domains. However, these methods only apply to the condition that users involved in both domains are fully shared, which is unrealistic in real world recommendation scenarios.

In this paper, we focus on a more challenging and practical cross-domain recommendation task, that is, recommending items to cold-start users in the target domain based on the learned preference of the users in other domains.

Some efforts are made to design CDR approaches for tackling the cold-start issue [6, 9]. These methods generally follow a two-stage embedding-and-mapping framework for CDR, which first train a model to learn the latent factor of users/items in each domain, and then optimize a mapping function to capture the cross-domain relationships of users/items. Though encouraging results are observed, these paradigms suffer several limitations. On the one hand, the first stage of embedding learning for such two-stage methods has a large impact on the final performance, but it is not optimized for the cross-domain recommendation objective, leading to suboptimal representation. Furthermore, optimizing the networks in a two-stage manner is rather time-consuming. On the other hand, such existing methods fail to model the high-order structure information among users and items when learning embeddings, leading to insufficient to capture the complex collaborative information. Thirdly, these approaches only focus on how to leverage the data of source domain to learn a mapping function. However, in fact,

CIKM '20, October 19–23, 2020, Virtual Event, Ireland

ACM ISBN 978-1-4503-6859-9/20/10…$15.00
https://doi.org/10.1145/3340531.3412069

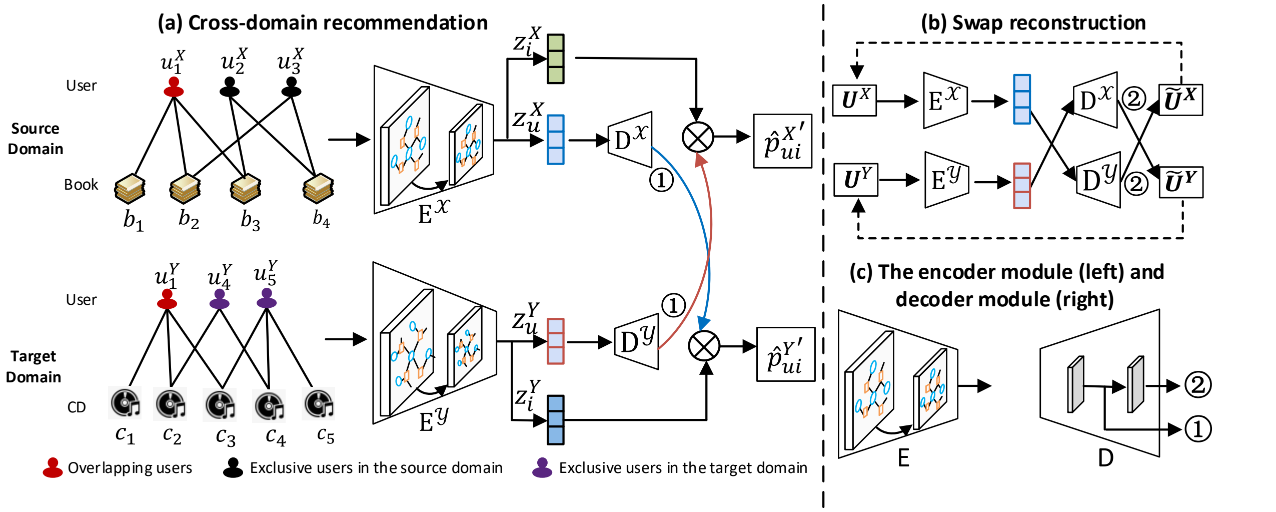

Figure 2: (a) The architecture of DAN for cross-domain recommendations. (b) The illustration of swap reconstruction. (c) Details of the encoder and decoder modules. The outputs of decoders in (a) and (b) are from branch ①, and branch ②, respectively.

the paired domains are each other's target domain, and thus, we can leverage the data in both domains to boost the optimization of the networks.

To address the above problems, in this paper, we propose an end-to-end Dual Autoencoder Network (DAN) for user cold-start recommendations with a pair of encoder-decoder networks. Figure 1 gives an overview of our solution. Conceptually, the proposed encoder in each domain adopts a graph neural network to embed the high-order collaborative information among users and items in the interaction graph via multi-hop propagation for effective user preference learning. Additionally, unlike the classic autoencoder model whose input and output are in the same domain, our decoder transforms the user information to the other domain for recommendations. As Figure 1 shows, given an overlapping user u_1, our model is optimized to make the transferred representations of u_1 close to the ground-truth embeddings in the target latent space for making accurate recommendation to it. In the cross-domain autoencoder framework, the reconstruction constraint is designed elaborately, which provides extra training optimization for the autoencoder to enhance the performance based on the data in both domains. The contributions of our work include: (1) We propose a framework DAN, which improves cold-start recommendations by an end-to-end manner in the cross-domain scenario. (2) We adopt a domain swapping strategy for designing reconstruction constraints in DAN with the two-branched decoders. (3) The experimental results show that the proposed DAN outperforms existing approaches by a significant margin on two pairs of real-world datasets.

2 PROPOSED METHOD

2.1 Preliminaries

Suppose that we are given two domains, a source domain $\mathcal{X}$, and a target domain $\mathcal{Y}$. In this paper, we take the CDR scenario that the 'CD-domain' is the target domain, and 'book-domain' is the source domain as an example. We denote $\mathcal{U}^X$, $\mathcal{I}^X$, $\mathcal{U}^Y$, and $\mathcal{I}^Y$ as the set of users and items in the source-domain and target-domain, respectively. Partial of the users in both domains are overlapped, and the items are disjoint in the two domains. The set of overlapping users are denoted as $\mathcal{U}^O = \mathcal{U}^X \cap \mathcal{U}^Y$. Given the implicit feedback of each domain, we represent the interaction data as a bipartite user-item graph $\mathcal{G}^* = \{(u,i)|u \in \mathcal{U}^*, i \in \mathcal{I}^*\}$ in each domain, where $* \in \{\mathcal{X}, \mathcal{Y}\}$. Users and items are attached with initial feature matrix $\mathbf{U}^X$, $\mathbf{I}^X$, $\mathbf{U}^Y$, $\mathbf{I}^Y$. An edge $p_{ui} = 1$ indicates an observed user-book reading interaction in the source domain, and user-CD purchase interaction in the target domain; otherwise, $p_{ui} = 0$. We aim to make top-N recommendations for those cold-start users in the target domain, by leveraging the information in the source domain. Through our paper, we denote vectors by lowercase boldface letters (e.g. $\mathbf{z} \in \mathbb{R}^d$), matrices by uppercase boldface letters (e.g. $\mathbf{Z} \in \mathbb{R}^{N\times d}$), and scalars as non-boldface letters.

2.2 The DAN Architecture

Figure 2 depicts the architecture of our proposed DAN model, which is realized by a pair of encoder-decoder networks. Note that the network structure in the two domains is the same, so for simplicity, we focus on source domain $\mathcal{X}$, and the model for $\mathcal{Y}$ is analogous.

2.3 Encoder Module

Inspired by the message-passing mechanism of graph neural networks [8], we employ graph convolution in the encoder module to encode the user preference by exploiting the high-order connectivity in the interaction graph of each domain. That is, we refine the representation of each node by aggregating its own feature and the message propagated from its interacted neighborhoods. With stacking multiple embedding propagation layers, the representation of a user can be augmented by integrating the knowledge propagated from its multi-hop neighbors. The representations for users and items at the layer l are summarized as follows.

$$\mathbf{Z}^{(l)} = LeakyReLU(\tilde{\mathbf{D}}^{-1}\tilde{\mathbf{A}}\mathbf{Z}^{(l-1)}\mathbf{W}_g) \tag{1}$$

where matrix $\mathbf{Z}^{(l)} \in \mathbb{R}^{N\times d}$ is the representation of users and items in the l-th layer, in which the information of the l-th hop neighbors are gathered. $\tilde{\mathbf{D}}$ is the degree matrix of $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, where $\mathbf{A}$ is the adjacent matrix and $\mathbf{I}$ is the identity matrix, which means adding the self-connection to retain the information of the user itself. $\mathbf{W}_g$ is the trainable parameter.

We let the observed user-item interactions guide the embedding learning in each domain by adding Bayesian Personalized Ranking (BPR) [11] loss. The objective function is formulated as follows:

$$\mathcal{L}^X = \sum_{(u,i,i')\in\mathcal{R}^X} -ln\sigma(\mathbf{z}_u^\top \mathbf{z}_i - \mathbf{z}_u^\top \mathbf{z}_{i'}) + \lambda||\Theta||_2^2 \tag{2}$$

where $\mathcal{R}^X = \{(u,i,i')|(u,i) \in \mathcal{G}^X, (u,i') \notin \mathcal{G}^X\}$ denotes the set of triple training data. $\top$ means transposition. To simplify the notation, in Equation (2), we replace $\mathbf{z}_u^X$ and $\mathbf{z}_i^X$ with $\mathbf{z}_u$ and $\mathbf{z}_i$. $\sigma(\cdot)$ is the sigmoid function. Θ is all trainable model parameters and λ is the regularization weight to prevent overfitting. Embeddings in the target domain are updated analogously.

2.4 Two-Branched Decoder

Different from the classic autoencoder, our DAN utilizes a decoder across two domains, for instance, decoding the users in the source domain to the target domain for recommendations. As Figure 2 (c) depicts, our decoder is designed with two branches, the outputs of which belong to different feature spaces. Concretely, the first branch (branch ①) aims to map the user embedding into latent space of the other domain for recommendations (corresponding to the output space of encoder), and the second branch (banch ②) aims to reconstruct the initial feature representations (the input of the encoder). The implementation of the decoder is as follows:

$$\mathbf{z}_u^{Y'} = \mathbf{W}_{d1}^\top(Tanh(\mathbf{W}_{d2}^\top \mathbf{z}_u^X + \mathbf{b}_{d2})) + \mathbf{b}_{d1} \tag{3}$$

$$\widetilde{u}^Y = D^X(\mathbf{z}_u^X) = \mathbf{W}_{d3}Tanh(\mathbf{z}_u^{Y'}) + \mathbf{b}_{d3} \tag{4}$$

where $\mathbf{z}_u^{Y'} \in \mathbb{R}^d$ is the mapped user latent vector corresponding to the first branch of output. The output of the second branch, $\widetilde{u}^Y$, is the reconstructed input feature of the overlapped user in the target domain. $\mathbf{W}_{d*}$ is the trainable weights, and $\mathbf{b}_{d*}$ is the bias, respectively, where $* \in \{1,2,3\}$. Note that the output of $D^X(\cdot)$ means the second branch output in this paper.

The quality of reconstructed input feature in Equation (4) indicts the performance of our model, which can be employed to guide the optimzation. Moreover, as the ground-truths about input features of the overlapping users in two domains are available, the source and target domains become relative, thus we explore a new design of reconstruction losses according to the characteristic of our decoder. Specifically, we achieve this goal from two perspectives with mutual enhancements of domain data. On the one side, we force the reconstructed feature by the decoder D^X in the source domain to be as close to its ground-truth initial feature vector $\mathbf{u}^Y$ in the target domain as possible. To this end, a paired reconstruction loss is performed as follows:

$$\mathcal{L}_p^X = \sum_{u\in\mathcal{U}^O} ||D^X(\mathbf{z}_u^X) - \mathbf{u}^Y||_2^2 \tag{5}$$

On the other side, as Figure 2 (b) shows, we design a swap reconstruction constraint. Concretely, we feed the latent representation in domain $\mathcal{X}$ to the decoder in domain $\mathcal{Y}$ to reconstruct the input feature of domain $\mathcal{X}$, and vice versa. That is, we swap the data from another domain as additional information to enhance the performance of the decoder itself in one domain. Based on this insight, we design the swap loss of the cross-domain reconstruction:

$$\mathcal{L}_s^X = \sum_{u\in\mathcal{U}^O} ||D^Y(\mathbf{z}_u^X) - \mathbf{u}^X||_2^2 \tag{6}$$

2.5 Cold-Start Recommendation

After transferring the vectors of the cold-start users in the source-domain latent space to the target-domain latent space, we conduct the inner product (Equation (7)) to estimate the probability that a cold-start user u would adopt item i in target-domain, and recommend the top-N items with the highest probability to user u.

$$\widehat{p}_{ui}^{Y'} = z_u^{Y'} \cdot z_i^Y \tag{7}$$

where $z_u^{Y'}$ denotes the user latent factor mapped from domain $\mathcal{X}$ to domain $\mathcal{Y}$, and z_i^Y is the latent factor of item i in domain $\mathcal{Y}$.The cross-domain recommendation objective is optimized by BPR loss in the training process with overlapped users as Equation (8).

$$\mathcal{L}^{X2Y} = \sum_{u\in\mathcal{U}^O} -ln\sigma(\widehat{p}_{ui}^{Y'} - \widehat{p}_{ui'}^{Y'}) + \lambda||\Theta||_2^2 \tag{8}$$

Finally, we train our bi-directional model jointly in an end-to-end manner, and the objective is summarized as follows:

$$\begin{aligned}\mathcal{L} &= (\mathcal{L}^X + \mathcal{L}^Y) + (\mathcal{L}^{X2Y} + \mathcal{L}^{Y2X}) \\ &+ (\mathcal{L}_p^X + \mathcal{L}_p^Y) + (\mathcal{L}_s^X + \mathcal{L}_s^Y)\end{aligned} \tag{9}$$

where $\mathcal{L}^Y$, $\mathcal{L}_p^Y$, $\mathcal{L}_s^Y$ and $\mathcal{L}^{Y2X}$, are obtained with the similar algorithms described in formula (2), (5), (6) and (8), respectively.

3 EXPERIMENTS

Table 1: Statistics of two cross-domain scenarios (Inter. denotes interactions and Overlap. denotes overlapping users)

Datasets	#Users	#Items	#Inter	density	#Overlap
Books	10,040	22,044	672,664	0.30%	5,739
CDs and Vinly	6,557	7,433	178,905	0.37%	
CDs and Vinly	10,992	14,680	185,041	0.11%	2,397
Movies and TV	10,514	8,184	149,480	0.17%	

Datasets. We evaluate our DAN and baselines on two pairs of real-world cross-domain datasets from Amazon[1], including *Books & CDs and Vinyl*; *CDs and Vinyl & Movies and TV*. For each pair, we randomly select 50% of the total overlapping users as cold-start users by removing their information in the target domain for testing. The detailed statistics are provided in Table 1.

Implementation Details. We implement our DAN with Pytorch[2]. For a fair comparison, we set the embedding size as 64 for all models. In DAN, the size of the initial pretrained embedding with CML[4] is 64. The configuration of networks is the same in both source and target domains. In the encoder module, we employ two-layer GCN with hidden dimensions [64, 64].The first branch network of decoder adopts two-layer MLPs with size [64, 64], and

[1]https://nijianmo.github.io/amazon/index.html
[2]https://pytorch.org/

Table 2: Cold-start recommendation performances of two cross-domain scenarios. Best results are in boldface.

Dataset	Metrics	EMCDR-BPR	SSCDR	DAN
Books CDs and Vinly	HR@10	0.0959	0.1143	**0.1459**
	HR@20	0.1184	0.1353	**0.1728**
	NDCG@10	0.0921	0.1137	**0.1417**
	NDCG@20	0.1006	0.1206	**0.1508**
CDs and Vinly Movie	HR@10	0.2249	0.2565	**0.2864**
	HR@20	0.2713	0.3019	**0.3353**
	NDCG@10	0.2314	0.2632	**0.2941**
	NDCG@20	0.2507	0.2814	**0.3134**

another fully connected layer is utilized as the second branch network. The whole DAN model is trained end-to-end via Adam [7] optimizer with learning rate 0.0001, and the regularization coefficient λ is 0.0005. We randomly sample 999 items that have not interacted with the user as the negative items, and evaluate the top-N recommendation performance via hit ratio (HR@k) and normalized discounted cumulative gain (NDCG@k) of the average scores of 10 rounds.

Experimental Results and Analysis. To demonstrate the effectiveness of our DAN, we compare it with the state-of-the-art CDR methods, EMCDR-BPR and SSCDR, which can provide top-N recommendations to cold-start users, and apply the same network configuration as suggested in [6] for them. Due to the space limitation, we refer readers to Section 4 for the details of them. As shown in Table 2, our DAN consistently outperforms all the baselines in each cross-domain recommendation scenario. And the performance of SSCDR is better than EMCDR, one possible reason is that our DAN factors the high-order collaborative information into the embedding learning process to provide high-quality preference representation via performing propagation operation, while SSCDR only utilizes the multi-hop neighborhood information when inference instead of directly infusing high-order relationships into the optimization process, as well as EMCDR doesn't consider the high-order relationships. That demonstrates that it is of vital important to model high-order relationships for learning effective user preference. Furthermore, the end-to-end learning manner makes our DAN is capable to directly leverage cross-domain recommendation objective to guide the embedding learning, which is unattainable for the two-stage methods. More importantly, our dual autoencoder network with designed reconstruction constraints mautally enhances the domain data to optimize our model, as a result, the transferred embedding by our model is more colse to the groud-truth vector for recommendation.

4 RELATED WORK

Existing researches for CDR mainly aim to improve recommendation accuracy for existing users in the target domain [5, 12, 13]. However, all these methods are not apply to the cold-start recommendation task. Some efforts have been made to design CDR approaches for cold-start recommendation [1, 6, 9]. EMCDR [9] presents a general embedding-and-mapping framework for CDR, which first adopts latent factor models to find the embedding of users/items in each domain, and then learns a mapping function to capture the cross-domain relationships of users/items in a supervised manner. To alleviate the dependence of EMCDR to overlapping users, SSCDR [6] employs semi-supervised metric space mapping, which takes the overlapping users as labeled data and all the source-domain items as unlabeled data. All these algorithms follow a two-step embedding-and-mapping paradigm, which cannot optimize the embedding procedure based on the quality of the CDR, leading to the suboptimal performance.

5 CONCLUSION

In this paper, we propose a dual autoencoder network model for user cold-start recommendation in an end-to-end manner, which captures the high-order collaborative information for effective user preference learning. Furthermore, we devise cross-domain reconstruction constraints to provide extra training optimization to enhance the performance based on the data across domains. Experiments conducted on real-world datasets demonstrate that DAN makes more effective recommendations to cold-start users than the existing two-stage embedding-and-mapping methods. In future work, it is worth exploring the disentanglement technique to extract high-quality transferable knowledge for cross-domain recommendations.

ACKNOWLEDGMENTS

This work is supported by the National Key R&D Program of China (2019YFB1406303), the National Natural Science Foundation of China under Grant 61876003 and Beijing Natural Science Foundation L192024. It is a research achievement of Key Laboratory of Science, Techonology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology).

REFERENCES

[1] 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*. 94–101.

[2] 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.

[3] Ming He, Jiuling Zhang, Peng Yang, and Kaisheng Yao. 2018. Robust Transfer Learning for Cross-Domain Collaborative Filtering Using Multiple Rating Patterns Approximation. In *WSDM*. 225–233.

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

[5] Guangneng Hu, Yu Zhang, and Qiang Yang. 2018. CoNet: Collaborative Cross Networks for Cross-Domain Recommendation. In *CIKM*. 667–676.

[6] 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.

[7] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

[8] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR*.

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

[10] Nima Mirbakhsh and Charles X. Ling. 2015. Improving Top-N Recommendation for Cold-Start Users via Cross-Domain Information. *ACM Trans. Knowl. Discov. Data* (2015).

[11] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *UAI*. 452–461.

[12] Feng Yuan, Lina Yao, and Boualem Benatallah. 2019. DARec: Deep Domain Adaptation for Cross-Domain Recommendation via Transferring Rating Patterns. In *IJCAI*. 4227–4233.

[13] Cheng Zhao, Chenliang Li, and Cong Fu. 2019. Cross-Domain Recommendation via Preference Propagation GraphNet. In *CIKM*. 2165–2168.