

# Learning Accurate and Bidirectional Transformation via Dynamic Embedding Transportation for Cross-Domain Recommendation

Weiming Liu<sup>1</sup>, Chaochao Chen<sup>1\*</sup>, Xinting Liao<sup>1</sup>, Mengling Hu<sup>1</sup>, Yanchao Tan<sup>2</sup>, Fan Wang<sup>1</sup>, Xiaolin Zheng<sup>1</sup>, Yew-Soon Ong<sup>3</sup>

<sup>1</sup>College of Computer Science and Technology, Zhejiang University, China

<sup>2</sup>College of Computer and Data Science, Fuzhou University, Fuzhou, China

<sup>3</sup>School of Computer Science and Engineering, Nanyang Technological University, 50 Nanyang Ave, Singapore  
{21831010, zjuccc, xintingliao, humengling}@zju.edu.cn, yctan@fzu.edu.cn, {fanwang97, xlzheng}@zju.edu.cn, asysong@ntu.edu.sg

## Abstract

With the rapid development of Internet and Web techniques, Cross-Domain Recommendation (CDR) models have been widely explored for resolving the data-sparsity and cold-start problem. Meanwhile, most CDR models should utilize explicit domain-shareable information (e.g., overlapped users or items) for knowledge transfer across domains. However, this assumption may not be always satisfied since users and items are always non-overlapped in real practice. The performance of many previous works will be severely impaired when these domain-shareable information are not available. To address the aforementioned issues, we propose the Joint Preference Exploration and Dynamic Embedding Transportation model (**JPEDET**) in this paper which is a novel framework for solving the CDR problem when users and items are non-overlapped. **JPEDET** includes two main modules, i.e., joint preference exploration module and dynamic embedding transportation module. The joint preference exploration module aims to fuse rating and review information for modelling user preferences. The dynamic embedding transportation module is set to share knowledge via neural ordinary equations for dual transformation across domains. Moreover, we innovatively propose the dynamic transport flow equipped with linear interpolation guidance on barycentric Wasserstein path for achieving accurate and bidirectional transformation. Our empirical study on Amazon datasets demonstrates that **JPEDET** outperforms the state-of-the-art models under the CDR setting.

## Introduction

Cross-Domain Recommendation (CDR) has been widely investigated nowadays since it is an effective approach for tackling data sparsity and cold-start issues in the recommender system (Zang et al. 2022; Lu et al. 2015; Zhu et al. 2021a). Leveraging useful knowledge (e.g., user-item ratings and reviews) across domains can enhance the model performance. Meanwhile, most current CDR models (Man et al. 2017) assume that users or items are overlapped across domains for knowledge sharing. However, explicit domain-shareable information (e.g., overlapped users

or items) might be difficult to obtain, and thus, the contributions of these CDR models might be insignificant (Li, Yang, and Xue 2009; Moreno et al. 2012; Gao et al. 2013; Choi et al. 2022). How to obtain preeminent recommendation results without explicit domain-shareable information has become an urgent problem.

In this paper, we focus on the dual-target non-overlapped CDR problem. That is, we aim to provide source (or target) users with target (or source) items according to preference characteristics. We further assume that both source and target users are non-overlapped. Meanwhile, each domain has multiple types of information, e.g., user-item ratings and reviews information, which is commonly available in real practice (Yi and et al 2018; Wang, Ounis, and Macdonald 2021; Chen et al. 2019; Dong et al. 2020). This problem is rather challenging since (1) there is no explicit transferring bridges (e.g., via overlapped users) for dual knowledge sharing and (2) the existence of embedding discrepancy across domains that strongly hurdles the model performance.

Previous CDR models cannot better resolve these challenges well, resulting in limited performance. On the one hand, most CDR models should rely on overlapped users or items to develop reliable representations via embedding mapping and alignment mechanism (Zang et al. 2022). However, these model performance could be severely degraded when explicit domain-shareable information (e.g., overlapped users) are not available. What is worse, different domains with diverse kinds of items are always heterogeneous which always involves domain bias (Guerraoui et al. 2017; Li et al. 2021). Although commonly-used Maximum Mean Discrepancy (MMD) is easier to be implemented for domain adaptation without overlapped users, it fails to provide accurate matching results among complicated latent embedding spaces (Korba et al. 2021). Meanwhile previous papers have pointed out that adversarial training with domain discriminators could be unstable under some circumstances (Shu et al. 2018) which degrades the model performance. On the other hand, recent CDR models (Yu et al. 2020; Li et al. 2023) still mainly focus on unidirectional mapping from rich to sparse domains. However, unidirectional mapping cannot satisfy dual-target recommendation task which limits their potentials. Thus how to fully exploit

\*Chaochao Chen is the corresponding author.

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

**accurate** and **bidirectional** domain alignment method for recommendation still needs more investigation.

To address the aforementioned issues, in this paper, we propose Joint Preference Exploration and Dynamic Embedding Transportation model (**JPEDET**) for solving dual-target non-overlapped cross domain recommendation problem. We devise two modules in **JPEDET**, i.e., joint preference exploration module and dynamic embedding transportation module for fusing users' rating and review information and share general knowledge across domains. The joint preference exploration module fuses rating and review information via dual autoencoder frameworks for exploiting user general preference. The dynamic embedding transportation module aims to share general but implicit preference among non-overlapped users across domains. To fulfill this task, we innovatively propose dynamic transport flow with matching regularization and moving correction stages. Specifically, dynamic transport flow provides an accurate and bidirectional transformation across domains via linear interpolation guidance on barycentric Wasserstein path, providing monotonous and straight trajectories with a constant moving speed. Utilizing these two modules, we can enhance recommendation performance for both domains via realizing the embedding mapping and transportation process of non-overlapped users across domains. We summarize our contributions: (1) We propose a novel framework consisting of the joint preference exploration module and dynamic embedding transportation module, i.e., **JPEDET**, for solving the dual-target non-overlapped cross domain recommendation problem. (2) We are the first to propose accurate and bidirectional transformation via dynamic embedding transportation, for enhancing dual transferring user embeddings across domains with theoretical guarantees. (3) Extensive empirical studies demonstrate that **JPEDET** significantly improves the state-of-the-art model.

## Related Work

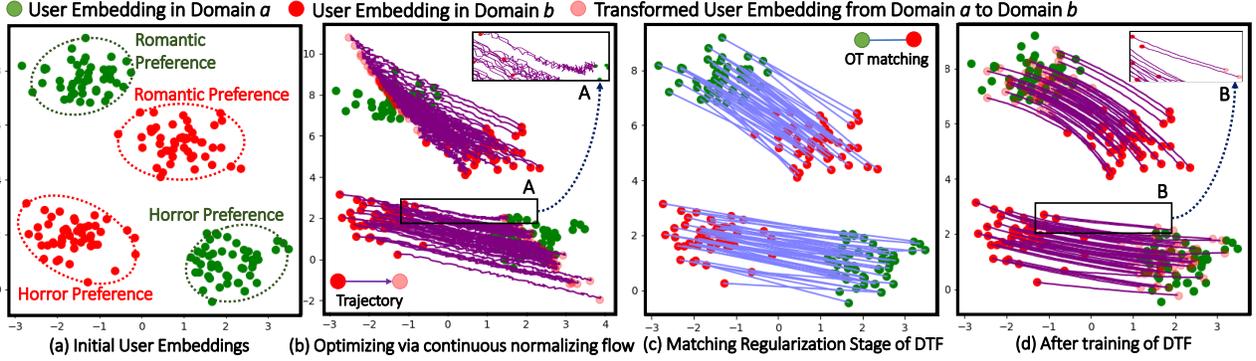
**Cross Domain Recommendation.** Cross Domain Recommendation (CDR) mainly involves source and target domains for solving data sparsity and cold-start problem (Zhao and et al 2021; Khan and et al 2017; Sun et al. 2022). Existing CDR approaches have two main types, i.e., *overlapped-based* methods and *non-overlapped-based* methods. Overlapped-based methods assume that users or items are overlapped across domains and regard them as the bridge for knowledge sharing. Most overlapped-based methods (Man et al. 2017; Zhu et al. 2022; Kang et al. 2019) utilize linear or non-linear mapping functions on the overlapped users to transfer useful information. Some overlapped-based methods also adopt cross-connection components (Hu, Zhang, and Yang 2018) or orthogonal transformation unit (Chen et al. 2023; Li and Tuzhilin 2020, 2021) to enhance the model performance. Non-overlapped-based methods investigate the more general and challenging case when users and items are non-overlapped. Most non-overlapped-based methods utilize some other auxiliary useful information (e.g., reviews) (Yu et al. 2020; Choi et al. 2022) to enhance the model perfor-

mance with MMD (Long and et al 2015), and domain adversarial training strategy (Ganin et al. 2016; Zhang et al. 2021). Nonetheless, MMD fails to obtain reliable estimation when domains are biased, while adversarial learning is proven to too unstable to provide promising results (Korba et al. 2021; Shu et al. 2018). How to provide more accurate predictions under non-overlapped scenario still needs more exploration.

**Dynamic Flow and Optimal Transport.** Dynamic flow aims to provide an accurate and invertible transformation approach between different probability distributions. Discrete normalized flow (Rezende and Mohamed 2015; Tabak and Turner 2013) was first been proposed for density estimation with logarithm probability calculation. RTQ Chen et al. (Richter-Powell and et al 2022) further adopted neural ordinary equation into dynamic flow which made it simpler for dual forward and backward processes. To make the continuous trajectory become simpler for faster convergence, researchers have utilized dynamic optimal transport techniques for achieving straight and smooth results (Onken et al. 2021; Finlay et al. 2020; Huang and Yeh 2021; Tong and et al 2020; Yang and Karniadakis 2020; Huguet and et al 2022; Liu and et al 2023; Tong and et al. 2023). However, these models requires either heavy computation on gradient and matrix trace, or non-trivial estimation for domain distribution, which is not practical for real applications.

## Methodology

First, we describe notations. We assume there are two domains, i.e., a source domain  $a$  and a target domain  $b$ . We assume each domain have  $N_U^x$  users and  $N_V^x$  items where  $x \in \{a, b\}$ .  $\mathbf{r}^x \in \mathbb{R}^{N_U^x \times N_V^x}$  is the observed rating matrices in  $x$ -th domain. For the  $i$ -th user and  $j$ -th item in the  $x$ -th domain, it consists of the tuples  $(u_i^x, v_j^x, r_{ij}^x, h_{ij}^x)$ . Here  $r_{ij}^x$  and  $h_{ij}^x$  denote the rating and review information respectively. Meanwhile, the source and target users/items are non-overlapped. We aim to provide dual transfer cross domain recommendation for non-overlapped users, i.e., providing items in domain  $b$  to users in domain  $a$  who they did not have rating interactions in domain  $b$  and vice versa. The task is more general and challenging since (1) rating and review are diverse and heterogeneous for user modeling, and (2) no explicit domain-shareable information, e.g., overlapped users/items, is available to serve as the bridge for knowledge sharing. We then introduce the overview of **JPEDET** framework. **JPEDET** mainly has two modules, i.e., *joint preference exploration module* and *dynamic embedding transportation module*. The joint preference exploration module aims to better exploit user general preference embeddings according to the rating and review information. The dynamic embedding transportation module is set to dual transform the users across different domains. To achieve this goal, we firstly propose *dynamic transport flow* with the combination of neural ordinary equation and optimal transport technique. Specifically, dynamic transport flow includes matching regularization and moving correction stages to learn an accurate and bidirectional domain adaptation approach. The model framework is shown in Fig. 2 and we will introduce **JPEDET** in details.


 Figure 1: The toy examples of dynamic embedding transportation from domain  $a$  to domain  $b$  with two-dimension data

## The Framework JPEDET

**Joint Preference Exploration Module.** Firstly, we provide the details of the joint preference exploration module. In this module, we aim to combine both rating and auxiliary review information for generating user general preference embeddings. To start with, we first adopt the rating encoder  $E_R^x(\cdot)$  to obtain the user rating preference embeddings as  $E_R^x(\mathbf{r}_{i*}^x) = \mathbf{z}_{R,i}^x \in \mathbb{R}^d$  where  $d$  denotes the dimension. Then we can utilize the rating decoder  $D_R^x(\cdot)$  to reconstruct the user-item rating information as  $\tilde{\mathbf{r}}_{i*}^x = D_R^x(\mathbf{z}_{R,i}^x)$ . For the single review information  $h_{ij}^x$ , we first adopt the sentence segmentation component Sentencizer to split  $h_{ij}^x$  into several individual sentences following (Pugoy and Kao 2021). After that we adopt SentenceBERT (Reimers and Gurevych 2019) to obtain the embeddings for each sentence. Then we average these sentence embeddings to obtain the review representations on  $h_{ij}^x$ . Finally, we average all review representations belonging to the  $i$ -th user in domain  $x$  as  $\mathbf{H}_i^x$ . Likewise, we adopt the review encoder  $E_H^x(\cdot)$ , decoder  $D_H^x(\cdot)$  to model the user review preference embeddings as  $E_H^x(\mathbf{H}_i^x) = \mathbf{z}_{H,i}^x \in \mathbb{R}^d$  and reconstruct the review via  $\tilde{\mathbf{H}}_i^x = D_H^x(\mathbf{z}_{H,i}^x)$ . Besides, we further exploit the user general preference embeddings which contain both rating and review information. To fulfill this task, we concatenate the user review and rating information and use a full-connected network  $W^x(\cdot)$  to obtain the user general preference as  $\mathbf{u}_i^x = W^x(\mathbf{r}_{i*}^x \oplus \mathbf{H}_i^x)$ . Inspired by the multi-view consensus learning strategy (Zhang, Liu, and Fu 2019), we regard ratings and reviews information as multiple views for representing the user tastes and characteristics. Thus we tend to learn the neural networks  $G_R^x(\cdot)$ ,  $G_H^x(\cdot)$  for modeling the relationship between user general and specific (e.g., rating and review) preference embeddings. Then we propose the preference exploration loss  $L_R^x$  as:

$$L_R^x = \frac{1}{N} \sum_{i=1}^N \left[ \|\mathbf{r}_i^x - \tilde{\mathbf{r}}_i^x\|_2^2 + \|\mathbf{H}_i^x - \tilde{\mathbf{H}}_i^x\|_2^2 \right] + \frac{1}{N} \sum_{i=1}^N \left[ \|\mathbf{z}_{R,i}^x - G_R^x(\mathbf{u}_i^x)\|_2^2 + \|\mathbf{z}_{H,i}^x - G_H^x(\mathbf{u}_i^x)\|_2^2 \right].$$

where  $N$  denotes the batchsize. The first and second terms in  $L_R^x$  represent the reconstruction loss on ratings and reviews.

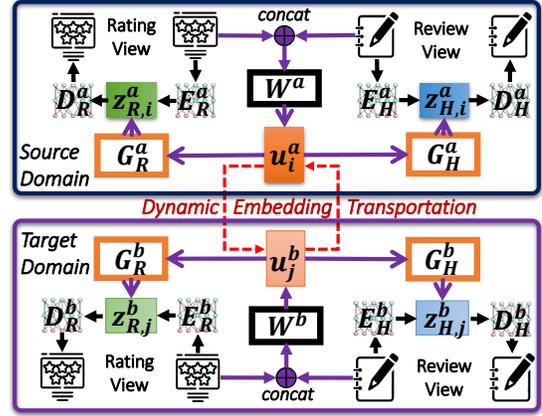


Figure 2: The framework of JPEDET

The others denote the regression loss among the user general and rating/review preference embeddings. We adopt average weights among these terms following (Xin et al. 2022).  $L_R^x$  will be applied in both source and target domains to explore user general preference  $\mathbf{u}^a$  and  $\mathbf{u}^b$ .

**Dynamic Embedding Transportation Module.** After we obtain the user general preference, we should consider how to provide proper recommendation results on the  $j$ -th item in domain  $b$  to the  $i$ -th user in domain  $a$ , and vice versa. That is, one can directly adopt  $D_R^b(G_R^b(\mathbf{u}^a))$  or  $D_R^a(G_R^a(\mathbf{u}^b))$  to make the cross-domain predictions. However, we cannot obtain satisfactory results since different domains always exist the domain discrepancy and it will lead to poor model performance (Li et al. 2021; Yu et al. 2020). It has been shown in Fig. 1(a) where red and green dots denote the preference of source and target users, respectively. The red and green dots are separated which indicates the existence of the domain discrepancy among source and target users. Since we cannot obtain explicit bridge (e.g., overlapped users) for knowledge sharing, we should learn an **accurate** and **bidirectional** embedding transportation module across source and target domains to reduce the discrepancy for solving dual-target non-overlapped CDR problem. To fulfill this task, we propose *Dynamic Transport Flow*

(DTF) in the dynamic embedding transport module. DTF is equipped with neural ordinary equations (Chen et al. 2018; Grathwohl and et al 2018) which further includes two stages, i.e., matching regularization stage and moving correction stage. DTF first adopts discrete optimal transport mechanism among the user embeddings in the matching regularization stage, then utilizes moving correction stage to obtain a more smooth trajectory.

**Neural Ordinary Equations of DTF.** To start with, we first introduce the neural ordinary equations of DTF which includes the forward and backward process. Specifically, the forward process denotes the procedure that transports the source user embeddings  $\mathbf{u}^a$  to target domain. We utilize a neural ordinary equation with learnable parameters  $\mathbf{f}(\cdot, \boldsymbol{\theta})$  to fulfill the task as  $\mathbf{u}_{i,t+\Delta t}^a = \mathbf{u}_{i,t}^a + \mathbf{f}(\mathbf{u}_{i,t}^a, t; \boldsymbol{\theta}) \cdot \Delta t$  and  $\mathbf{u}_{i,T}^a = \mathcal{F}(\mathbf{u}_{i,0}^a) = \text{ODE.forward}(\mathbf{u}_{i,0}^a)$  where  $\mathbf{u}_{i,t}^a$  denotes the transformed source user embedding at the  $t$ -th iteration and  $\Delta t$  denotes the step size.  $\mathbf{u}_{i,T}^a$  denotes the final transformed user embedding from source to target domains at the  $T$ -th iteration. Note that  $\mathbf{u}_{i,0}^a$  equals to the initial  $i$ -th user embeddings  $\mathbf{u}_i^a$  and  $\mathbf{u}_{i,T}^a$  denotes the final transformed user embeddings where  $T$  denotes the total number of iterations. We regard the whole forward moving process, i.e.,  $\mathcal{F}(\cdot)$ , as the transferring bridge, which can be shown in Fig. 1(b) for a more intuitive reflection. Specifically, the red and pink dots represent the  $\mathbf{u}_{i,0}^a$  user embeddings and  $\mathbf{u}_{i,T}^a$  transformed user embeddings respectively. The purple line denotes the moving trajectory from  $\mathbf{u}_{i,0}^a$  to  $\mathbf{u}_{i,T}^a$ . Likewise, we can not only transport user embeddings from domain  $a$  to  $b$ , but also make reverse transport from domain  $b$  to  $a$  using backward process as  $\mathbf{u}_{j,t-\Delta t}^b = \mathbf{u}_{j,t}^b - \mathbf{f}(\mathbf{u}_{j,t}^b, t; \boldsymbol{\theta}) \cdot \Delta t$  and  $\mathbf{u}_{j,T}^b = \mathcal{F}^{-1}(\mathbf{u}_{j,0}^b) = \text{ODE.backward}(\mathbf{u}_{j,0}^b)$  where  $\mathbf{u}_{j,T}^b$  denotes the  $j$ -th user's final transformed embeddings from domain  $b$  to domain  $a$ . It is obvious that utilizing the neural ordinary equations can satisfy the bidirectional conditions.

**Matching Regularization Stage of DTF.** Although we have obtained the final transformed user embeddings  $\mathbf{u}_{i,T}^a$  and  $\mathbf{u}_{j,T}^b$  via neural ordinary equations, they still have bias and discrepancy among the origin user embeddings  $\mathbf{u}^a$  and  $\mathbf{u}^b$ , respectively. Therefore, we aim to match the final transformed user embeddings  $\mathbf{u}_{i,T}^a$  and  $\mathbf{u}_{j,T}^b$  with original user embeddings  $\mathbf{u}_{i,0}^a$  and  $\mathbf{u}_{j,0}^b$  respectively in the matching stage of DTF. To tackle this issue, previous methods (Chen et al. 2018; Onken et al. 2021; Richter-Powell and et al 2022; Finlay et al. 2020) always adopted continuous normalizing flow which can be formulated as below:

$$\log \mathbb{P}(\mathbf{u}_{i,T}^x) = \log \mathbb{P}(\mathbf{u}_{i,0}^x) - \int_0^T \text{tr} \left( \frac{\partial \mathbf{f}(\mathbf{u}_{i,t}^x, t; \boldsymbol{\theta})}{\partial \mathbf{u}_{i,t}^x} \right) dt, \quad (1)$$

where  $\mathbb{P}(\cdot)$  denotes the probability distribution. However, it is difficult to obtain the probability distribution on domain  $a$  or  $b$ , since these distributions are empirically observed but unknown. Although one can adopt some non-parametric methods (e.g., Kernel Density Estimation) to estimate the probability distribution, it is sensitive to find a suitable hyper-parameters to obtain accurate results. As the example in Fig. 1(b), the pink dots (transformed source user embeddings) and green dots (target user embeddings) are still not

well aligned, when they are directly optimized via continuous normalized flow and kernel density estimation. What is worse, they are easier to obtain arbitrary mapping across domains while degrading the model performance (Korotin and et al 2019; Onken et al. 2021). To overcome such obstacles, we further utilize the optimal transport techniques for dynamic domain adaptation with Theorem 1 on optimizing the moving trajectories (Seguy et al. 2018; Makkuva and et al 2020; Huang and et al 2020; Mikami and Thieullen 2008).

**Theorem 1.** *Given the probability densities of  $\mu_a$  and  $\mu_b$  in the source and target domains respectively, the dynamic optimal transport problem can be formulated as follows (Finlay et al. 2020; Onken et al. 2021; Tong and et al 2020):*

$$\begin{aligned} \min_{(\rho, \mathbf{f})} & \int_0^T \int_{\mathbb{R}^d} \frac{1}{2} \|\mathbf{f}(\mathbf{u}^x, t)\|_2^2 \cdot \rho(\mathbf{u}^x, t) d\mathbf{u}^x dt, \\ \text{s.t.} & \frac{d\rho(\mathbf{u}^x, t)}{dt} + \nabla \cdot [\rho(\mathbf{u}^x, t) \cdot \mathbf{f}(\mathbf{u}^x, t)] = 0, \rho(\cdot, 0) = \mu_a, \rho(\cdot, T) = \mu_b, \end{aligned}$$

where  $\rho(\cdot, t)$  denotes the probability densities of the transformed user embeddings at the  $t$ -th step.  $\mathbf{f}(\mathbf{u}^x, t)$  has the optimal solution as  $\mathbf{f}(\mathbf{u}^x, t) = -\nabla \lambda(\mathbf{u}^x, t)$  where  $\lambda(\mathbf{u}^x, t)$  denotes the potential function. Optimizing dynamic optimal transportation problem is equivalent to minimize the following two loss functions, i.e., continuity constraint loss  $\ell_M$  and path-length constraint loss  $\ell_S$  as follows:

$$\begin{aligned} \ell_M &= \int_0^T \int_{\mathbb{R}^d} \left| \frac{d\lambda(\mathbf{u}^x, t)}{dt} - \|\nabla \lambda(\mathbf{u}^x, t)\|_2^2 \right| d\mathbf{u}^x dt, \\ \ell_S &= \int_0^T \int_{\mathbb{R}^d} \frac{1}{2} \|\mathbf{f}(\mathbf{u}^x, t)\|_2^2 d\mathbf{u}^x dt. \end{aligned} \quad (2)$$

Based on Theorem 1, one should first consider path-length constraint by figuring out the optimal mapping between the source and target domains to determine the moving directions. However, previous methods (Onken et al. 2021; Yang and Karniadakis 2020; Zhang, Weinan, and Wang 2018) should involve complex gradient and trace computation during the optimization. Meanwhile Discrete Optimal Transport (DOT) with entropy regularization term (Courty and et al 2017; Flamary et al. 2016) enjoys the benefits of providing cyclic monotonous mapping efficiently for providing accurate matching results (Makkuva and et al 2020; Paty, d'Aspremont, and Cuturi 2020; Villani et al. 2009). Therefore, we adopt DOT on  $\mathbf{u}^a$  and  $\mathbf{u}^b$  as follows:

$$\min_{\boldsymbol{\pi} \in \Gamma} \mathcal{J} = \sum_{i,j=1}^N [\boldsymbol{\pi}_{i,j} \cdot \|\mathbf{u}_i^a - \mathbf{u}_j^b\|_2^2 + \epsilon \boldsymbol{\pi}_{i,j} (\log(\boldsymbol{\pi}_{i,j}) - 1)], \quad (3)$$

where  $\Gamma = \{\sum_{i=1}^N \boldsymbol{\pi}_{i,j} = 1/N, \sum_{j=1}^N \boldsymbol{\pi}_{i,j} = 1/N\}$  denotes the constraints on  $\boldsymbol{\pi}$ .  $\boldsymbol{\pi}_{i,j}$  denotes the coupling matrix between  $\mathbf{u}_i^a$  and  $\mathbf{u}_j^b$  accordingly.  $\epsilon$  is the balanced hyper parameter and  $\sum_{i,j=1}^N \boldsymbol{\pi}_{i,j} \cdot (\log(\boldsymbol{\pi}_{i,j}) - 1)$  denotes the entropy regularization term. We can adopt Sinkhorn algorithm (Cuturi 2013) to solve the problem on  $\boldsymbol{\pi}$  iteratively with time complexity of  $O(N^2)$ . Note that the matching solution based on discrete optimal transport is monotonous. We depict the optimal matching solution between user in domain  $a$  (red dots) and user in domain  $b$  (green dot) with blue lines in Fig. 1(b).

**Moving Correction Stage of DTF.** After we obtain the optimal solution of coupling matrix  $\pi_{i,j}$ , we achieve the moving directions for each user. Meanwhile, we wish these data sample embeddings will travel in straight lines to other domains since it is the optimal trajectory for convex cost ideally as illustrated in Theorem 1. However, the moving trajectory depicted as the purple lines between the original and transformed user embeddings will rapidly vary and bring truncation error as shown in Fig. 1(b). Apparently, it will bring about noise and lead to inaccurate transportation plans across domains, resulting in the limited model performance (Finlay et al. 2020; Onken et al. 2021; Tong and et al 2020). The main reason is there always exists many possible transporting solutions across domains. The models may find out an arbitrary transporting result among all possible solutions and make the trajectory become fluctuated with poorly conditioned (Korotin and et al 2019). Previous methods (Finlay et al. 2020; Yang and Karniadakis 2020; Onken et al. 2021) mainly used kinetic (energy) and potential regularization to enforce the straight trajectories. However, these constraints only guide the model implicitly with heavy computation. How to provide a simple but efficient way for guiding the model to obtain straight trajectories explicitly is still challenging. To alleviate such issues, we propose moving correction stage of DTF which can provide a robust and smooth moving trajectory. For achieving this goal, we first propose *Barycentric Wasserstein Path* to represent an ideal straight trajectory. Then we propose *Linear Interpolation Guidance* to further constrain the moving trajectory.

**Barycentric Wasserstein Path.** To start with, we first introduce the newly proposed barycentric Wasserstein path which is the basis of moving correction stage. We first apply the matching regularization stage of DTF to exploit the mapping solution across domains. Then we figure out the barycentric mapping embeddings  $\tilde{\mathbf{u}}_i^a$  and  $\tilde{\mathbf{u}}_j^b$  across domains respectively. Specifically, barycentric mapping is set to project the user embeddings  $\mathbf{u}^x$  in domain  $x$  to another domain via discrete optimal transport which can be computed as  $\tilde{\mathbf{u}}_i^a = N\pi_{i*}\mathbf{u}_*^a$  and  $\tilde{\mathbf{u}}_j^b = N\pi_{*j}^\top\mathbf{u}_*^b$  respectively. We then define the line segment vectors between the original user embeddings  $\mathbf{u}_i^a$  and barycentric mapping embeddings  $\tilde{\mathbf{u}}_i^a$  as the corresponding barycentric Wasserstein path  $\mathcal{W}_i^a$  in source domain. Likewise, we obtain the barycentric Wasserstein path  $\mathcal{W}_j^b$  in target domain which can be depicted as:

$$\mathcal{W}_i^a := \tilde{\mathbf{u}}_i^a - \mathbf{u}_i^a, \quad \text{and} \quad \mathcal{W}_j^b := \tilde{\mathbf{u}}_j^b - \mathbf{u}_j^b. \quad (4)$$

Apparently,  $\mathcal{W}_i^a$  and  $\mathcal{W}_j^b$  can be viewed as an ideal straight trajectory that transforms the user embeddings across domains. We can utilize the corresponding barycentric Wasserstein path for optimizing our coarse trajectory during the training stage.

**Linear Interpolation Guidance.** Then we will introduce our proposed Linear Interpolation Guidance strategy on barycentric Wasserstein path for optimizing the trajectory.

**Theorem 2.** *Given the probability densities of  $\mu_a$  and  $\mu_b$  in source and target domains respectively, data samples should move at a constant speed for achieving optimal solutions.*

Based on Theorem 2, we not only enforce the transformed user embeddings moving towards a straight and smooth trajectory, but also let them move at a constant speed to achieve the optimal solution. Suppose that the source and target domains have the probability densities of  $\mu_a$  and  $\mu_b$  respectively, McCann and Moser proposed a simple but efficient interpolation method  $\alpha_\eta = (1 - \eta)\mu_b + \eta\mu_a$  for mass transportation where  $\eta \in [0, 1]$  and  $\alpha_\eta$  denotes the interpolant (Dacorogna and Moser 1990; McCann 1997; Lei and Gu 2021; Rozen et al. 2021; Moser 1965; Gu and Yau 2020). This method also gives geodesics in Wasserstein space with lower transportation cost (Lei and Gu 2021; Liu and et al 2023). Based on the above observations, we first uniformly divide the barycentric Wasserstein path  $\mathcal{W}_i^a$  and  $\mathcal{W}_j^b$  into  $T$  segments as follows:

$$\gamma_{i,t}^a = \mathbf{u}_i^a + (t/T) \cdot \mathcal{W}_i^a, \quad \gamma_{j,t}^b = \mathbf{u}_j^b + (t/T) \cdot \mathcal{W}_j^b, \quad (5)$$

where  $\gamma_{i,t}^a$  and  $\gamma_{j,t}^b$  denote the linear interpolation points at the  $t$ -th time step on  $\mathcal{W}_i^a$  and  $\mathcal{W}_j^b$ , respectively. Then we aim to reduce the distance between the transformed user embedding  $(\mathbf{u}_{i,t}^a, \mathbf{u}_{j,t}^b)$  and the linear interpolation points  $(\gamma_{i,t}^a, \gamma_{j,t}^b)$  simultaneously. Therefore, we propose interpolation guidance loss for minimizing the pairwise distance as below:

$$\min L_G = \sum_{i,j=1}^N \sum_{t=1}^T (\|\gamma_{i,t}^a - \mathbf{u}_{i,t}^a\|_2^2 + \|\gamma_{j,t}^b - \mathbf{u}_{j,t}^b\|_2^2). \quad (6)$$

The linear interpolation on barycentric Wasserstein path is easy to compute and provides explicit guidance for training straight moving trajectories with a constant speed for optimization. That is, interpolation guidance loss satisfies both continuity constraint loss  $\ell_M$  and path-length constraint loss  $\ell_S$  as mentioned in Theorem 1 and Theorem 2.

**Model Summary.** We first minimize the preference exploration loss  $L_R^x$  for model pretraining in both source and target domains to obtain user preference embeddings. Then we minimize the interpolation guidance loss  $L_G$  for learning the dynamic transport flow. After the training procedure, we can make accurate and bidirectional cross domain predictions via  $D_R^b(G_R^b(\mathcal{F}(\mathbf{u}^a)))$  or  $D_R^a(G_R^a(\mathcal{F}^{-1}(\mathbf{u}^b)))$  respectively.

## Empirical Study

**Datasets and Tasks.** We conduct extensive experiments on the popularly used real-world *Amazon* datasets (Ni, Li, and McAuley 2019). **Amazon** dataset has five domains, i.e., Movies (Movies and TV), Books (Books), CD (CDs and Vinyl), Phone (Cell Phones and Accessories), and Elec (Electronics) which are commonly used in the cross domain recommendation (Liu et al. 2022; Yu et al. 2020; Fu et al. 2019; Zhao et al. 2020). Specifically, we conduct four corresponding tasks as (T1) Book  $\leftrightarrow$  CD, (T2) Book  $\leftrightarrow$  Movie, (T3) Movie  $\leftrightarrow$  CD, and (T4) Phone  $\leftrightarrow$  Elec. We filter out users whose number of interactions is less than 5 in each domain following (Chen et al. 2023; Zhu et al. 2021b; Yuan, Yao, and Benatallah 2019). We keep the origin user-item ratings and set the unobserved or not clicked as 0. To establish the non-overlapped datasets, we first figure out the overlapped users among the source and target domains. Then we

	Book->CD		CD->Book		Book->Movie		Movie->Book		Movie->CD		CD->Movie		Phone->Elec		Elec->Phone	
	RMSE	MAE														
NeuMF	4.720	4.216	3.697	3.044	3.525	3.227	3.683	3.251	4.862	4.311	4.102	3.531	4.977	4.013	4.469	3.540
DeepCoNN	4.275	4.080	3.101	2.469	2.902	2.734	2.856	2.743	4.453	4.022	3.264	2.519	4.382	3.500	3.974	3.192
VCM	4.123	3.969	2.758	2.202	2.591	2.435	2.660	2.353	4.312	3.904	3.048	2.282	4.174	3.395	3.623	3.017
NARRE	4.044	3.871	2.632	2.152	2.474	2.360	2.376	2.133	4.198	3.871	2.850	2.226	4.034	3.162	3.523	2.807
UBERT	3.811	3.723	2.140	1.964	2.152	1.935	2.093	1.854	3.666	3.579	2.284	1.895	3.772	2.943	3.153	2.501
ESCOFILT	3.756	3.694	2.115	1.863	2.080	1.859	2.101	1.792	3.710	3.608	2.326	1.953	3.741	2.893	3.130	2.465
RC-DFM	3.919	3.825	2.303	1.936	2.314	2.001	2.087	1.812	3.786	3.632	2.431	2.012	3.945	2.920	3.397	2.539
GA-DTCDR	3.762	3.704	2.146	1.881	2.159	1.803	2.032	1.768	3.754	3.615	2.243	1.920	3.713	2.851	2.962	2.278
Rec-DAN	3.502	3.267	1.968	1.725	2.086	1.874	1.922	1.703	3.521	3.349	2.005	1.866	3.620	2.658	2.742	1.973
DDTCDR	3.431	3.303	2.094	1.755	2.047	1.792	1.823	1.576	3.253	3.115	1.964	1.801	3.456	2.327	2.633	1.842
TDAR	2.918	2.483	1.835	1.583	1.903	1.665	1.731	1.459	3.042	2.856	1.909	1.774	3.195	2.032	2.521	1.734
DisAlign	2.342	2.139	1.907	1.628	1.793	1.505	1.664	1.430	2.751	2.352	1.653	1.302	3.242	2.119	2.407	1.686
CATN	2.287	1.950	1.714	1.514	1.704	1.426	1.580	1.362	2.153	1.917	1.718	1.484	3.002	1.745	2.390	1.556
CFAA	1.902	1.498	1.751	1.541	1.631	1.349	1.548	1.211	1.832	1.551	1.696	1.265	2.813	1.297	2.104	1.303
SRTrans	1.765	1.245	1.493	1.380	1.552	1.204	1.463	1.147	1.692	1.274	1.500	1.163	2.762	1.204	1.961	1.187
SER	1.456	1.031	1.420	1.047	1.382	1.050	1.465	1.087	1.435	1.040	1.396	1.059	2.798	1.173	1.975	1.148
<b>JPEDET-B</b>	<b>3.685</b>	<b>3.539</b>	<b>2.023</b>	<b>1.810</b>	<b>2.264</b>	<b>1.713</b>	<b>1.982</b>	<b>1.770</b>	<b>3.684</b>	<b>3.538</b>	<b>2.045</b>	<b>1.832</b>	<b>3.616</b>	<b>2.700</b>	<b>2.205</b>	<b>1.531</b>
<b>JPEDET-M</b>	<b>1.714</b>	<b>1.352</b>	<b>1.585</b>	<b>1.396</b>	<b>1.521</b>	<b>1.299</b>	<b>1.606</b>	<b>1.187</b>	<b>1.626</b>	<b>1.203</b>	<b>1.680</b>	<b>1.296</b>	<b>2.869</b>	<b>1.258</b>	<b>2.047</b>	<b>1.361</b>
<b>JPEDET-A</b>	<b>1.596</b>	<b>1.168</b>	<b>1.551</b>	<b>1.261</b>	<b>1.469</b>	<b>1.188</b>	<b>1.524</b>	<b>1.125</b>	<b>1.507</b>	<b>1.163</b>	<b>1.400</b>	<b>1.180</b>	<b>2.954</b>	<b>1.385</b>	<b>2.121</b>	<b>1.423</b>
<b>JPEDET-M1</b>	<b>1.388</b>	<b>0.973</b>	<b>1.357</b>	<b>0.976</b>	<b>1.358</b>	<b>0.968</b>	<b>1.403</b>	<b>1.035</b>	<b>1.383</b>	<b>0.945</b>	<b>1.353</b>	<b>0.961</b>	<b>2.641</b>	<b>1.109</b>	<b>1.867</b>	<b>1.103</b>
<b>JPEDET-M2</b>	<b>1.374</b>	<b>0.965</b>	<b>1.362</b>	<b>0.980</b>	<b>1.362</b>	<b>0.975</b>	<b>1.408</b>	<b>1.038</b>	<b>1.362</b>	<b>0.936</b>	<b>1.347</b>	<b>0.958</b>	<b>2.607</b>	<b>1.096</b>	<b>1.875</b>	<b>1.110</b>
<b>JPEDET-M3</b>	<b>1.310</b>	<b>0.918</b>	<b>1.313</b>	<b>0.931</b>	<b>1.310</b>	<b>0.927</b>	<b>1.297</b>	<b>0.947</b>	<b>1.320</b>	<b>0.926</b>	<b>1.277</b>	<b>0.940</b>	<b>2.581</b>	<b>1.072</b>	<b>1.782</b>	<b>1.034</b>
<b>JPEDET</b>	<b>1.267</b>	<b>0.879</b>	<b>1.273</b>	<b>0.919</b>	<b>1.206</b>	<b>0.893</b>	<b>1.264</b>	<b>0.923</b>	<b>1.203</b>	<b>0.854</b>	<b>1.235</b>	<b>0.917</b>	<b>2.556</b>	<b>1.064</b>	<b>1.769</b>	<b>1.023</b>

Table 1: Experimental results on Amazon datasets with different tasks.

randomly select users to appear in source domain and the others in target domain, so that the users of the two domains are non-overlapping following the setting of (Wang, Niepert, and Li 2019). We only use (a) source user-source item rating and review information and (b) target user-target item rating and review information during the training phase.

**Experiment Settings.** We set batch size  $N = 128$  for source and target domains during the training. The latent dimension of user rating/review/general preference embeddings is set to  $d = 128$ . We set the step size as  $\Delta t = 0.01$  and the total number of iterations as  $T = 30$  in the moving stage of dynamic transport flow. We set the balanced hyper-parameter  $\epsilon = 0.1$  in matching stage of dynamic transport flow for calculating the discrete optimal transport. For all experiments, we perform five random experiments and report the average results. We choose Adam as optimizer and set the learning rate as 0.01. We adopt the commonly-used Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) as the evaluation metrics following previous papers (Fu et al. 2019; Zhu et al. 2022; Zhao et al. 2020).

**Baseline.** We compare **JPEDET** with the following models. (1) **Single domain models:** **NeuMF** (He et al. 2017), **DeepCoNN** (Zheng and et al 2017), **VCM** (Cui et al. 2018), **NARRE** (Chen et al. 2018), **UBERT** (Qiu et al. 2021), **ESCOFILT** (Pugoy and Kao 2021). (2) **Cross domain models:** **RC-DFM** (Fu et al. 2019), **GA-DTCDR** (Zhu et al. 2021b), **Rec-DAN** (Wang, Niepert, and Li 2019), **DDTCDR** (Li and Tuzhilin 2020), **TDAR** (Yu et al. 2020), **DisAlign** (Liu et al. 2021), **CATN** (Zhao et al. 2020), **CFAA** (Liu et al. 2022), **SER** (Choi et al. 2022), **SRTrans** (Li et al. 2023).

**Recommendation Performance.** The comparison results

on several datasets are shown in Table 1. From them, we can find that: (1) Single domain recommendation models equipped with ratings and reviews (e.g., **DeepCoNN** and **NARRE**) can obtain better results than the models which only use ratings (e.g., **NeuMF**). However, they cannot provide satisfactory results since they cannot reduce the domain bias and discrepancy. (2) Conventional cross-domain recommendation models (e.g., **RC-DFM** and **GA-DTCDR**) can obtain better results than most single domain recommendation models. However, they mainly rely on the domain-shareable information for knowledge transfer which limits the performance when users are non-overlapped. (3) Some latest cross-domain recommendation models (e.g., **Rec-DAN**, **TDAR** and **SER**) also utilize adversarial learning strategy to reduce domain discrepancy when users and items are non-overlapped. Nonetheless, adversarial learning with domain discriminator is unstable and hard to train in practice (Shu et al. 2018) and thus they cannot achieve better results. (4) **JPEDET** achieves more satisfied results than the runner-up models (e.g., **SER**) with improvement from 7.4% to 17.9%, which proves that joint preference exploration and dynamic embedding transportation can boost the model potential.

**Ablation.** To study how does each module of **JPEDET** contribute on the final performance, we compare **JPEDET** with its several variants, including **JPEDET-B**, **JPEDET-M**, **JPEDET-A**, **JPEDET-M1**, **JPEDET-M2** and **JPEDET-M3**. **JPEDET-B** only adopts the joint preference exploration module during the training procedure and it directly applies  $D_R^b(G_R^b(\mathbf{u}^a))$  or  $D_R^a(G_R^a(\mathbf{u}^b))$  for testing. **JPEDET-M** and **JPEDET-A** replace the dynamic embedding transportation module with MMD and domain ad-

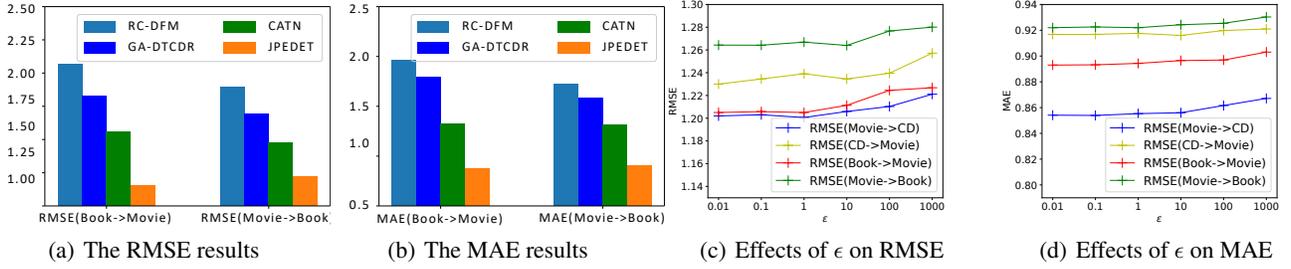
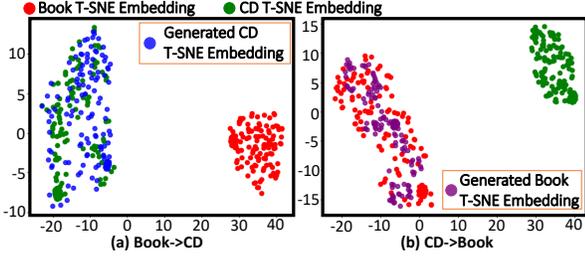


Figure 3: The model extension and effect of hyper-parameters

Figure 4: The T-SNE visualization of DTF in **JPEDET**

versarial training respectively. **JPEDET-M1**, **JPEDET-M2** and **JPEDET-M3** replace the dynamic embedding transportation module with OT-Flow (Onken et al. 2021), TPR (Huang and Yeh 2021) and Rectified Flow (Liu and et al 2023) respectively. The comparison results are shown in Table 1. From it, we can observe that (1) **JPEDET-B** cannot reduce biases and discrepancy and reaches poor performance which indicates the significance of embedding transformation across domains. (2) **JPEDET-M** and **JPEDET-A** both achieve better results than **JPEDET-B**. However MMD only provides ambiguous matching and thus it cannot better reduce domain discrepancy. **JPEDET-A** is difficult to train with domain discriminator in real practice, leading to the limited performance of **JPEDET-A**. (3) Although **JPEDET-M1** and **JPEDET-M2** both achieve better results than **JPEDET-B**, these methods cannot well adapt to the scenario when the probability distribution of source and target domains are hard to estimate then leading to limited performance. (4) **JPEDET-M3** achieves competitive results than **JPEDET** while **JPEDET-M3** should involve multiple iterations for re-training. When the source and target probability distributions are rather complex, it could be difficult for **JPEDET-M3** to fulfill the rectification. (5) The above ablation study shows that our proposed **JPEDET** is effective in solving the cross-domain recommendation problem.

**Model Extension.** We further analyse the general extension of **JPEDET** on the scenario that some source and target users are overlapped. Specifically, we randomly choose 5% users as overlapped across domains. The cross-domain user-item rating and review information for the rest of non-overlapped users are removed during the training phase and they will be used for evaluation in the testing phase.

Then we add a new alignment loss for these overlapped users as  $\min L_N = \sum_{i \in \mathcal{O} \cup \mathcal{U}} \|\mathbf{u}_{i,T}^a - \mathbf{u}_{i,0}^b\|_2^2 + \|\mathbf{u}_{i,T}^b - \mathbf{u}_{i,0}^a\|_2^2$  and the total loss for domain adaptation is given as  $\min[L_G + L_N]$ . We conduct the experiment on Amazon Book  $\leftrightarrow$  Amazon Movie and report the result of RMSE, MAE in Fig. 3(a)-(b). From that we can observe utilizing overlapped users as domain-shareable information can further boost the model performance. Moreover, our proposed **JPEDET** even achieves the best performance against some other baseline models, indicating that **JPEDET** can also be used when users are overlapped.

**Effect of hyper-parameters.** We finally study the effects of hyper-parameters  $\epsilon$  on **JPEDET** empirically. We vary the  $\epsilon \in \{0.01, 0.1, 1, 10, 100, 1000\}$  in DTF of dynamic embedding transportation module on Movie  $\leftrightarrow$  CD, Book  $\leftrightarrow$  Movie and report the results Fig. 3(c)-(d). From that we can observe that **JPEDET** is not sensitive to  $\epsilon$  especially when  $\epsilon = \{0.01, 0.1, 1\}$ . Meanwhile, smaller  $\epsilon$  could lead to relatively sparse and robust solutions on  $\pi$ . When  $\epsilon$  becomes larger (e.g.,  $\epsilon = \{100, 1000\}$ ), the coupling matrix  $\pi$  will become dense and thus provides less accurate matching results. Therefore, we set  $\epsilon = 0.1$  for DTF in **JPEDET**.

**Visualization.** To provide a more comprehensive insight into **JPEDET**, we adopt T-SNE to visualize the origin and transformed user embeddings on Movie  $\leftrightarrow$  CD as shown in Fig. 4(a)-(b). We observe that DTF provides accurate and bidirectional transformation across domains for reducing the discrepancy, showing the efficacy of DTF in **JPEDET**.

## Conclusion and Future Work

In this paper, we propose Joint Preference Exploration and Dynamic Embedding Transportation model (**JPEDET**), with the joint preference exploration module and the dynamic embedding transportation module. The dynamic embedding transportation module aims to provide an accurate and invertible embedding transformation approach between the source and target domains. We propose a simple but efficient approach namely Dynamic Transport Flow (DTF) with matching regularization stage and moving correction stage. Moreover, we adopt barycentric Wasserstein path with linear interpolation guidance to obtain straight moving trajectories. We conduct experiments to show the superior performance of **JPEDET** on several tasks.

## Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (No. 62172362), Leading Expert of “Ten Thousands Talent Program” of Zhejiang Province, China (No. 2021R52001), Distributed Smart Value Chain programme which is funded under the Singapore RIE2025 Manufacturing, Trade and Connectivity (MTC) Industry Alignment Fund-Pre-Positioning (Award No: M23L4a0001).

## References

- Chen, R. T.; Rubanova, Y.; Bettencourt, J.; and Duvenaud, D. K. 2018. Neural ordinary differential equations. *NeurIPS*.
- Chen, X.; Zhang, Y.; Tsang, I. W.; Pan, Y.; and Su, J. 2023. Toward Equivalent Transformation of User Preferences in Cross Domain Recommendation. *TOIS*, 41(1): 1–31.
- Chen, Z.; Wang, X.; Xie, X.; Wu, T.; Bu, G.; Wang, Y.; and Chen, E. 2019. Co-attentive multi-task learning for explainable recommendation. In *IJCAI*, 2137–2143.
- Choi, Y.; Choi, J.; Ko, T.; Byun, H.; and Kim, C.-K. 2022. Based Domain Disentanglement without Duplicate Users or Contexts for Cross-Domain Recommendation. In *CIKM*.
- Courty, N.; and et al. 2017. Joint distribution optimal transportation for domain adaptation. *NeurIPS*.
- Cui, K.; Chen, X.; Yao, J.; and Zhang, Y. 2018. Variational collaborative learning for user probabilistic representation. *WWW*.
- Cuturi, M. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *NeurIPS*.
- Dacorogna, B.; and Moser, J. 1990. On a partial differential equation involving the Jacobian determinant. In *Annales de l’Institut Henri Poincaré C, Analyse non linéaire*.
- Dong, X.; Ni, J.; Cheng, W.; Chen, Z.; Zong, B.; Song, D.; Liu, Y.; Chen, H.; and De Melo, G. 2020. Asymmetrical hierarchical networks with attentive interactions for interpretable review-based recommendation. In *AAAI*.
- Finlay, C.; Jacobsen, J.-H.; Nurbekyan, L.; and Oberman, A. 2020. How to train your neural ODE: the world of Jacobian and kinetic regularization. In *ICML*, 3154–3164. PMLR.
- Flamary, R.; Courty, N.; Tuia, D.; and Rakotomamonjy, A. 2016. Optimal transport for domain adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1.
- Fu, W.; Peng, Z.; Wang, S.; Xu, Y.; and Li, J. 2019. Deeply fusing reviews and contents for cold start users in cross-domain recommendation systems. In *AAAI*.
- Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. 2016. Domain-adversarial training of neural networks. *JMLR*, 17(1).
- Gao, S.; Luo, H.; Chen, D.; Li, S.; Gallinari, P.; and Guo, J. 2013. Cross-domain recommendation via cluster-level latent factor model. In *PKDD*, 161–176. Springer.
- Grathwohl, W.; and et al. 2018. FFIJORD: Free-Form Continuous Dynamics for Scalable Reversible Generative Models. In *ICLR*.
- Gu, X.; and Yau, S.-T. 2020. *Computational Conformal Geometry*. International Press and Higher Education Press.
- Guerraoui, R.; Kermarrec, A.-M.; Lin, T.; and Patra, R. 2017. Heterogeneous recommendations: what you might like to read after watching interstellar. *VLDB*, 10(10): 1070–1081.
- He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; and Chua, T.-S. 2017. Neural collaborative filtering. In *WWW*, 173–182.
- Hu, G.; Zhang, Y.; and Yang, Q. 2018. Conet: Collaborative cross networks for cross-domain recommendation. In *CIKM*.
- Huang, C.-W.; and et al. 2020. Convex Potential Flows: Universal Probability Distributions with Optimal Transport and Convex Optimization. In *ICLR*.
- Huang, H.-H.; and Yeh, M.-Y. 2021. Accelerating continuous normalizing flow with trajectory polynomial regularization. In *AAAI*, volume 35.
- Huguet, G.; and et al. 2022. Manifold interpolating optimal-transport flows for trajectory inference. *NeurIPS*.
- Kang, S.; Hwang, J.; Lee, D.; and Yu, H. 2019. Semi-supervised learning for cross-domain recommendation to cold-start users. In *CIKM*, 1563–1572.
- Khan, M. M.; and et al. 2017. Cross domain recommender systems: a systematic literature review. *CSUR*, 50(3): 1–34.
- Korba, A.; Aubin-Frankowski, P.-C.; Majewski, S.; and Ablin, P. 2021. Kernel stein discrepancy descent. In *ICML*, 5719–5730. PMLR.
- Korotin, A.; and et al. 2019. Wasserstein-2 Generative Networks. In *ICLR*.
- Lei, N.; and Gu, X. 2021. *Computational Conformal Geometry*. International Press and Higher Education Press.
- Li, B.; Yang, Q.; and Xue, X. 2009. Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction. In *IJCAI*.
- Li, P.; and Tuzhilin, A. 2020. Dtdcdr: Deep dual transfer cross domain recommendation. In *WSDM*, 331–339.
- Li, P.; and Tuzhilin, A. 2021. Dual metric learning for effective and efficient cross-domain recommendations. *TKDE*.
- Li, S.; Yao, L.; Mu, S.; Zhao, W. X.; Li, Y.; Guo, T.; Ding, B.; and Wen, J.-R. 2021. Debiasing learning based cross-domain recommendation. In *KDD*, 3190–3199.
- Li, Z.; Amagata, D.; Zhang, Y.; Hara, T.; Haruta, S.; Yonekawa, K.; and Kurokawa, M. 2023. Semantic Relation Transfer for Non-overlapped Cross-domain Recommendations. In *PAKDD*, 271–283. Springer.
- Liu, W.; Su, J.; Chen, C.; and Zheng, X. 2021. Leveraging distribution alignment via stein path for cross-domain cold-start recommendation. *NeurIPS*, 34: 19223–19234.
- Liu, W.; Zheng, X.; Hu, M.; and Chen, C. 2022. Collaborative Filtering with Attribution Alignment for Review-based Non-overlapped Cross Domain Recommendation. In *WWW*.
- Liu, X.; and et al. 2023. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. *ICLR*.
- Long, M.; and et al. 2015. Learning transferable features with deep adaptation networks. In *ICML*.

- Lu, J.; Wu, D.; Mao, M.; Wang, W.; and Zhang, G. 2015. Recommender system application developments: a survey. *Decision support systems*, 74: 12–32.
- Makkuva, A.; and et al. 2020. Optimal transport mapping via input convex neural networks. In *ICML*.
- Man, T.; Shen, H.; Jin, X.; and Cheng, X. 2017. Cross-domain recommendation: An embedding and mapping approach. In *IJCAI*, volume 17, 2464–2470.
- McCann, R. J. 1997. A convexity principle for interacting gases. *Advances in mathematics*, 128(1): 153–179.
- Mikami, T.; and Thieullen, M. 2008. Optimal transportation problem by stochastic optimal control. *SIAM Journal on Control and Optimization*.
- Moreno, O.; Shapira, B.; Rokach, L.; and Shani, G. 2012. Talmud: transfer learning for multiple domains. In *CIKM*.
- Moser, J. 1965. On the volume elements on a manifold. *Transactions of the American Mathematical Society*.
- Ni, J.; Li, J.; and McAuley, J. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *EMNLP*, 188–197.
- Onken, D.; Fung, S. W.; Li, X.; and Ruthotto, L. 2021. Ot-flow: Fast and accurate continuous normalizing flows via optimal transport. In *AAAI*, volume 35, 9223–9232.
- Paty, F.-P.; d’Aspremont, A.; and Cuturi, M. 2020. Regularity as regularization: Smooth and strongly convex brenier potentials in optimal transport. In *International Conference on Artificial Intelligence and Statistics*. PMLR.
- Pugoy, R. A.; and Kao, H.-Y. 2021. Unsupervised extractive summarization-based representations for accurate and explainable collaborative filtering. In *ACL*, 2981–2990.
- Qiu, Z.; Wu, X.; Gao, J.; and Fan, W. 2021. U-BERT: Pre-training user representations for improved recommendation. In *AAAI*, volume 35, 4320–4327.
- Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP*, 3982–3992.
- Rezende, D.; and Mohamed, S. 2015. Variational inference with normalizing flows. In *ICML*. PMLR.
- Richter-Powell, J.; and et al. 2022. Neural Conservation Laws: A Divergence-Free Perspective. In *NeurIPS*.
- Rozen, N.; Grover, A.; Nickel, M.; and Lipman, Y. 2021. Moser flow: Divergence-based generative modeling on manifolds. *NeurIPS*, 34: 17669–17680.
- Seguy, V.; Damodaran, B. B.; Flamary, R.; Courty, N.; Rolet, A.; and Blondel, M. 2018. Large Scale Optimal Transport and Mapping Estimation. In *ICLR*.
- Shu, R.; Bui, H. H.; Narui, H.; and Ermon, S. 2018. A dirt-t approach to unsupervised domain adaptation. *ICLR*.
- Sun, Z.; Fang, H.; Yang, J.; Qu, X.; Liu, H.; Yu, D.; Ong, Y.-S.; and Zhang, J. 2022. DaisyRec 2.0: Benchmarking Recommendation for Rigorous Evaluation. *TPAMI*.
- Tabak, E. G.; and Turner, C. V. 2013. A family of nonparametric density estimation algorithms. *Communications on Pure and Applied Mathematics*, 66(2).
- Tong, A.; and et al. 2020. Trajectorynet: A dynamic optimal transport network for modeling cellular dynamics. In *ICML*.
- Tong, A.; and et al. 2023. Improving and generalizing flow-based generative models with minibatch optimal transport. In *ICML Workshop*.
- Villani, C.; et al. 2009. *Optimal transport: old and new*, volume 338. Springer.
- Wang, C.; Niepert, M.; and Li, H. 2019. Recsys-dan: discriminative adversarial networks for cross-domain recommender systems. *TNNLS*, 31(8): 2731–2740.
- Wang, X.; Ounis, I.; and Macdonald, C. 2021. Leveraging review properties for effective recommendation. In *WWW*.
- Xin, D.; Ghorbani, B.; Gilmer, J.; Garg, A.; and Firat, O. 2022. Do Current Multi-Task Optimization Methods in Deep Learning Even Help? *NeurIPS*, 35: 13597–13609.
- Yang, L.; and Karniadakis, G. E. 2020. Potential flow generator with L 2 optimal transport regularity for generative models. *TNNLS*, 33(2): 528–538.
- Yi, J.; and et al. 2018. Rating prediction in review-based recommendations via adversarial auto-encoder. In *WI*.
- Yu, W.; Lin, X.; Ge, J.; Ou, W.; and Qin, Z. 2020. Semi-supervised collaborative filtering by text-enhanced domain adaptation. In *KDD*, 2136–2144.
- Yuan, F.; Yao, L.; and Benatallah, B. 2019. DAREc: deep domain adaptation for cross-domain recommendation via transferring rating patterns. In *IJCAI*, 4227–4233.
- Zang, T.; Zhu, Y.; Liu, H.; Zhang, R.; and Yu, J. 2022. A survey on cross-domain recommendation: taxonomies, methods, and future directions. *TOIS*, 41(2): 1–39.
- Zhang, C.; Liu, Y.; and Fu, H. 2019. Ae2-nets: Autoencoder in autoencoder networks. In *CVPR*, 2577–2585.
- Zhang, L.; Weinan, E.; and Wang, L. 2018. Monge-Ampère Flow for Generative Modeling.
- Zhang, Y.; Liu, Y.; Han, P.; Miao, C.; Cui, L.; Li, B.; and Tang, H. 2021. Learning personalized itemset mapping for cross-domain recommendation. In *IJCAI*, 2561–2567.
- Zhao, C.; Li, C.; Xiao, R.; Deng, H.; and Sun, A. 2020. CATN: Cross-domain recommendation for cold-start users via aspect transfer network. In *SIGIR*, 229–238.
- Zhao, W. X.; and et al. 2021. Recbole: Towards a unified, comprehensive and efficient framework for recommendation algorithms. In *CIKM*, 4653–4664.
- Zheng, L.; and et al. 2017. Joint deep modeling of users and items using reviews for recommendation. In *WSDM*.
- Zhu, F.; Wang, Y.; Chen, C.; Zhou, J.; Li, L.; and Liu, G. 2021a. Cross-domain recommendation: challenges, progress, and prospects. In *IJCAI*, 4721–4728.
- Zhu, F.; Wang, Y.; Zhou, J.; Chen, C.; Li, L.; and Liu, G. 2021b. A unified framework for cross-domain and cross-system recommendations. *TKDE*.
- Zhu, Y.; Tang, Z.; Liu, Y.; Zhuang, F.; Xie, R.; Zhang, X.; Lin, L.; and He, Q. 2022. Personalized transfer of user preferences for cross-domain recommendation. In *WSDM*, 1507–1515.