Federated Graph Learning for Cross-Domain Recommendation

Ziqi Yang[1,2], **Zhaopeng Peng**[1,2], **Zihui Wang**[1,2], **Jianzhong Qi**[3], **Chaochao Chen**[4], **Weike Pan**[5], **Chenglu Wen**[1,2], **Cheng Wang**[1,2], **Xiaoliang Fan**[1,2*]
[1]Fujian Key Laboratory of Sensing and Computing for Smart Cities, Xiamen University, China.
[2]Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, Xiamen University, China.
[3]University of Melbourne
[4]College of Computer Science and Technology, Zhejiang University Hangzhou, China
[5]College of Computer Science and Software Engineering, Shenzhen University Shenzhen, China
{yangziqi,pengzhaopeng,wangziwei}@stu.xmu.edu.cn
{clwen,cwang,fanxiaoliang}@xmu.edu.cn
jianzhong.qi@unimelb.edu.au, zjuccc@zju.edu.cn, panweike@szu.edu.cn

Abstract

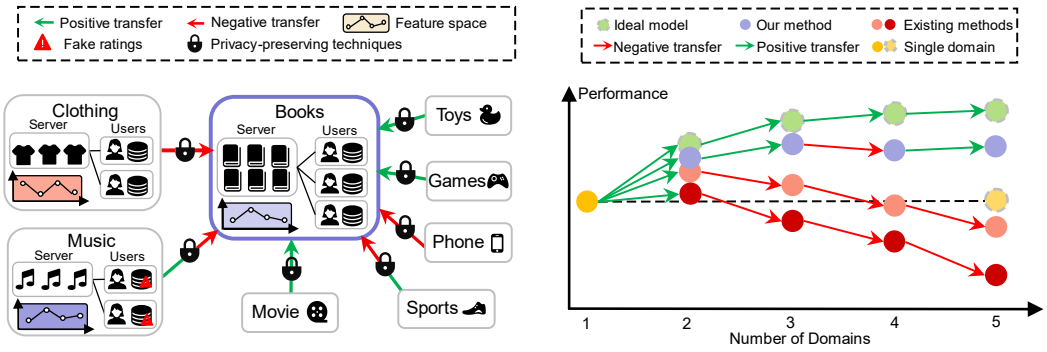
Cross-domain recommendation (CDR) offers a promising solution to the data sparsity problem by enabling knowledge transfer across source and target domains. However, many recent CDR models overlook crucial issues such as privacy as well as the risk of negative transfer (which negatively impact model performance), especially in multi-domain settings. To address these challenges, we propose FedGCDR, a novel federated graph learning framework that securely and effectively leverages positive knowledge from multiple source domains. First, we design a positive knowledge transfer module that ensures privacy during inter-domain knowledge transmission. This module employs differential privacy-based knowledge extraction combined with a feature mapping mechanism, transforming source domain embeddings from federated graph attention networks into reliable domain knowledge. Second, we design a knowledge activation module to filter out potential harmful or conflicting knowledge from source domains, addressing the issues of negative transfer. This module enhances target domain training by expanding the graph of the target domain to generate reliable domain attentions and fine-tunes the target model for improved negative knowledge filtering and more accurate predictions. We conduct extensive experiments on 16 popular domains of the Amazon dataset, demonstrating that FedGCDR significantly outperforms state-of-the-art methods.

1 Introduction

Cross-domain recommendation (CDR) has emerged as an effective solution for mitigating data sparsity in recommender systems [1, 2, 3, 4, 5]. CDR operates by integrating auxiliary information from source domains, thereby enhancing recommendation relevance in the target domain. Recently, to address data privacy constrains, many privacy-preserving CDR frameworks have been proposed [6, 7, 8, 9], which achieve strong performance under the assumptions of **data sparsity and a dual-domain model (i.e., typically involving a single source domain and a single target domain.)**.

*The corresponding author.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).



(a) The BS-CDR scenario. (b) The performance affected by the number of domains

Figure 1: (a) In order to obtain accurate recommendations in the Books domain, we aim to exploit user preferences (i.e., knowledge of foreign domains should be fully utilized, e.g. Movie, Toys, and Games domains). However, with the influence of lossy privacy-preserving techniques, the results of the transfer could be negative (e.g., the Music domain with low-quality data). (b) There is a diminishing marginal effect on the growth rate of the model performance with pure positive knowledge, while NT accumulates with an increasing number of source domains. Consequently, the performance of existing methods declines and is worse than that of a single domain model.

In this paper, we focus on a more generic scenario of **Broader-Source Cross-Domain Recommendation (BS-CDR)**, which integrates knowledge from more than two source domains while preserving privacy. Given the diverse nature of user preferences, it is essential to gain a more holistic understanding of user interests by incorporating user behaviors from diversified domains [10, 11]. For example, in Figure 1a, a user who enjoys certain types of books might also enjoy movies, toys, and games in similar genres. However, incorporating more domains while preserving privacy poses challenges to counteract negative transfer (NT), which is a phenomenon of transferring knowlege from a source domain that negatively impacts the recommender model performance in the target domain [12]. Suppose the Books domain in Figure 1a is the target domain. The Clothing domain is causing NT, because of the domain discrepancy. While the Music domain is supposed to transfer positive knowledge, it might also lead to NT because of lossy privacy-preserving techniques applied to broader source domains. As a result, the influx of negative knowledge accumulated from source domains will poison the model performance of the target domain in BS-CDR scenarios.

To mitigate the NT issue, attention mechanisms have been widely leveraged, either in an explicit (e.g., determine domain attentions by predefined domain features [13, 14]) or implicit manner (e.g., employ hyper-parameters [7, 15]). Another work [16] ensures positive transfer by passing only user-related features. However, existing methods cannot be directly applied to BS-CDR due to two major challenges. **First**, inadequate privacy preservation (**CH1**). Both intra-domain and inter-domain privacy must be carefully considered in BS-CDR. As depicted in Figure 1a, BS-CDR relies on extensive knowledge transfer, risking simultaneous privacy leakages across broader source domains (inter-domain privacy) [9, 17, 18, 19]. Additionally, concerns over centralized data storage may prevent users from sharing sensitive rating data (intra-domain privacy). **Second**, accumulative negative transfer (**CH2**). Adjusting hyper-parameters for a large number of source domains in BS-CDR scenarios is extremely difficult, as well as predefined domain features cannot accommodate complex domain diversities. For instance, the impact of NT can inevitably intensify with an increasing number of source domains (Figure 1b) [2]. In addition, the use of various lossy privacy-preserving techniques can further degrade the quality of transferred knowledge, complicating the achievement of positive transfer. Consequently, the performance of CDR models can decline to levels lower than those of single-domain models, as shown in Figure 1b.

To address the challenges of privacy (CH1) and NT (CH2) in BS-CDR, we propose Federated Graph learning for Cross-Domain Recommendation (FedGCDR). It follows a **horizontal-vertical-horizontal pipeline** [6] and consists of **two key modules**. First, the positive knowledge transfer module aims to safeguard inter-domain privacy and mitigate potential NT before transfer. This module adopts differential privacy (DP) [20] with a theoretical guarantee and aligns the feature spaces to facilitate positive knowledge transfer. Second, the positive knowledge activation module

is engaged to further alleviate NT. Specifically, it expands the local graph of the target domain by incorporating virtual social links, enabling the generation of domain attentions. Additionally, it performs target model fine-tuning to optimize the broader-source CDR. Extensive experiments on 16 popular domains from the Amazon benchmarks demonstrate that FedGCDR outperforms all baseline methods in terms of recommendation accuracy.

Our contributions are summarized as follows:

- We introduce FedGCDR, a novel federated graph learning framework for CDR that provides high-quality BS-CDR recommendations while safeguarding both user privacy and domain confidentiality;
- We propose two key model, i.e., the positive knowledge transfer module and the positive knowledge activation module, to address NT and privacy preservation simultaneously in BS-CDR;
- We conduct extensive experiments on the Amazon datasets that confirm the effectiveness of FedGCDR in terms of recommendation accuracy.

2 Related work

2.1 Cross-domain recommendation

CDR utilizes auxiliary information from external domains to alleviate the data sparsity problem and effectively improve recommendation quality. Li et al. [21] enrich domain knowledge by transferring user-item rating patterns from source domains to target domains. Man et al. [15] and Elkahky et al. [22] augment entities' embeddings in the target domain by employing a linear or multi layer perceptron (MLP)-based nonlinear mapping function across domains. Liu et al. [23] address the review-based non-overlapped recommendation problem by attribution alignment. Zhao et al.[24] improve the recommendation quality of multi-sparse-domains by mining domain-invariant preferences. Liu et al. [25] achieve knowledge transfer without overlapping users by mining joint preferences. Chen et al. [17] and Liu et al. [26] avoid intermediate result privacy leakage during cross-domain knowledge transfer by employing DP. In these works, the NT problem is often ignored because most of them assume a carefully selected dual-domain scenarios or limited multi-domain scenarios where NT is not evident. We aim to solve the NT problem in complex BS-CDR scenarios.

2.2 Federated recommendation

Recently, FL [27, 28, 29, 30, 31] has been widely adopted to tackle the privacy issue in recommender system. Chai et al. [32] adopts FL to classic matrix factorization algorithm and utilize homomorphic encryption to avoid the potential threat of privacy disclosure. Later, Wu et al. [33] explores the application of federated graph neural networks (GNN) models to improve the recommendation quality and ensure user privacy. To utilize sensitive social information, Liu et al. [8] adopts local differential privacy (LDP) and negative sampling. More recent studies use VFL to protect company's privacy in recommender system. Mai et al. [34] utilizes random projection and ternary quantization to ensure privacy preservation in VFL. In CDR, Chen et al. [9] designs a dual-target VFL CDR model with orthogonal mapping matrix and LDP for organizations' privacy preservation. Liu et al. [35] designs a graph convolutional networks (GCN)-based federated framework to learn user preference distributions for more accurate recommendations. To ensure user privacy in CDR, Liu et al. [6] utilizes a VAE-based federated model to mine user preference with data stored locally. Wu et al. [7] designs a personal module and a transfer module to provide personalised recommendation while preserving user privacy. These existing works, especially federated CDR frameworks, only protect the privacy of either individual users (intra-domain privacy) or the organizations (inter-domain privacy) but not both at the same time. We aim to provide both intra-domain and inter-domain privacy.

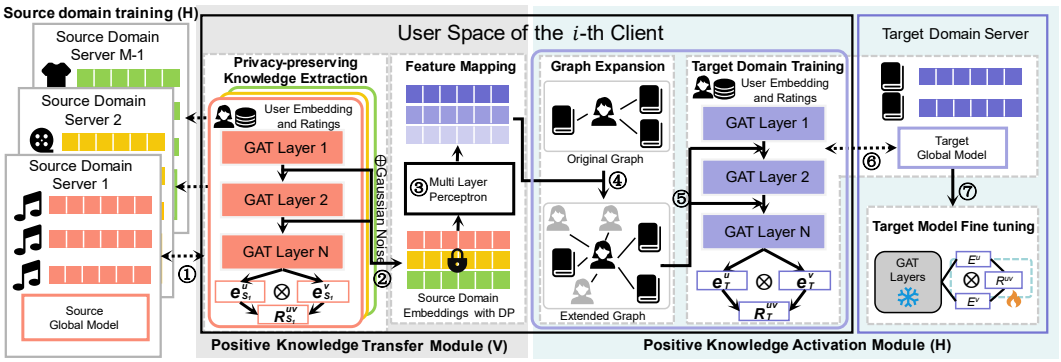


Figure 2: An overview of FedGCDR. It consists of two key modules and follows a HVH pipeline: (1) Source Domain Training (Horizontal FL): ① Each source domain maintains its graph attention network (GAT)-based federated model. (2) Positive Knowledge Transfer Module (Vertical FL): ② Source domain embeddings are extracted from GAT layers and perturbed with Gaussian noise. ③ The Multi layer Perceptron aligns the feature space of source domain embeddings and target domain embeddings. (3) Positive Knowledge Activation Module (Horizontal FL): ④ Local graph is expanded with source domain embeddings. ⑤ Enhanced federated training of the target domain is achieved through the expanded graph. ⑥ The target domain maintains its GAT-based federated model. ⑦ The target domain freezes the GAT layer and fine tunes the model.

3 Methodology

3.1 Problem definition

We consider M (M>3) domains participating in the CDR process. The domains are divided into M-1 source domains $\mathcal{D}^{S_1}, \mathcal{D}^{S_2}, ..., \mathcal{D}^{S_{M-1}}$ and one target domain $\mathcal{D}^{T}$. Each domain is assigned a domain server to conduct intra-domain model training. $\mathcal{U}$ is the user set across all the domains, $\mathcal{U} = \mathcal{U}_1 \bigcup \mathcal{U}_2 \bigcup ... \bigcup \mathcal{U}_M$, where $\mathcal{U}_i$ denotes the user set of domain i. We assume that users partially overlap between domains. Each user is treated as an individual client. User space refers to the virtual space in the user's device containing domain models distributed from each domain server. Meanwhile, $\mathcal{V}_i$ is the item set of domain i. Let $\mathbf{R}^i \in \mathbb{R}^{|\mathcal{U}_i| \times |\mathcal{V}_i|}$ be the observed rating matrix of the i-th domain. We consider top-K recommendation, i.e., we learn a function to estimate the scores of unobserved entries in the rating matrix, which are later used for item ranking and recommendations. Our goal is to achieve highly accurate recommendations in the target domain.

3.2 Framework of FedGCDR

3.2.1 Overview

The overall framework of FedGCDR is shown in Figure 2. FedGCDR follows a Horizontal-Vertical-Horizontal (HVH) pipeline and its two horizontal stages ensure the intra-domain privacy (i.e., individual user level) privacy. Our two key modules focus on the vertical stage and the second horizontal stage: (1) The positive knowledge transfer module preserves the inter-domain privacy by DP and alleviates NT by feature mapping. (2) The positive knowledge activation module filters out potential harmful or conflicting knowledge from the source domains. Specifically, we expand the local graph of the target domain by virtual social links, such that the target domain GAT model could generate reliable domain attention based on the expanded graph. After target domain GAT model training, we further mitigate NT by adopting a fine-tuning stage.

Horizontal-Vertical-Horizontal pipeline The HVH pipeline contains three stages with switching federated settings. The first horizontal stage refers to the source domain training in which source domain servers individually interacts with its domain users (clients). The private rating matrices are stored within each client, while the clients exchange model and gradients to train a domain-specific global model. The next two stages correspond to our two key modules (vertical positive knowledge transfer module and horizontal positive knowledge activation module), which we will cover in detail in the following subsections. It's important to note that the vertical positive knowledge transfer

module is completely computed in each client's user space (their personal devices). This is because the needed source domain knowledge can be extracted from local source models on each client which are distributed during the first horizontal source domain training stage.

Following the HVH pipeline, we achieve: (1) Privacy enhancement. The two horizontal stages can provide intra-domain privacy preservation, while we further ensure inter-domain privacy by applying privacy technism to the vertical stage. In the mean time, servers are not involved in the knowledge transfer process (i.e., the positive knowledge transfer module), making them unaware of user interactions in other source domains. (2) Communication efficiency. Cross-domain knowledge transfer does not require additional communication overhead.

Intra-domain GAT-based federated model We adopt a GAT-based [36, 37, 38] federated framework as the underlying model for our intra-domain recommender system. The horizontal paradigm avoids centralized storage of user ratings to ensure intra-domain privacy (**CH1**). In the initial step, each user and item is offered an ID embedding of size d, denoted by $\mathbf{e}_u^0$, $\mathbf{e}_v^0 \in \mathbb{R}^d$ respectively. The embedding is passed through L message propagation layers [39, 40, 41]. For the l-th layer:

$$\mathbf{e}_u^{l+1} = \sigma(\mathbf{W}^l(a_{uu}^l \mathbf{e}_u^l + \sum_{v \in N_u} a_{uv}^l \mathbf{e}_v^l)), \tag{1}$$

where N_u is the neighbor set of u, $\mathbf{W}^l$ is a learnable weight matrix, a_{uu}^l and a_{uv}^l are the importance coefficients computed by the attention mechanism:

$$a_{uv}^l = \frac{exp(LeakyReLU(\alpha(\mathbf{W}\mathbf{e}_u^l || \mathbf{W}\mathbf{e}_v^l))}{\sum_{v' \in N_u \bigcup u} exp(LeakyReLU(\alpha[\mathbf{W}\mathbf{e}_u^l || \mathbf{W}\mathbf{e}_{v'}^l])}, \tag{2}$$

where α is the weight vector. Inspired by LightGCN [42], we discard feature transformation and nonlinear activation for better model efficiency and learning effectiveness:

$$e_u^{l+1} = a_{uu}^l \mathbf{e}_u^l + \sum_{v \in N_u} a_{uv}^l \mathbf{e}_v^l, \tag{3}$$

$$a_{uv} = \frac{exp(\alpha(\mathbf{e}_u^l || \mathbf{e}_v^l))}{\sum_{v' \in N_u \bigcup u} exp(\alpha(\mathbf{e}_u^l || \mathbf{e}_{v'}^l)}. \tag{4}$$

In each source domain, the domain server and corresponding users collaboratively train a GAT-based federated model. The training process follows the horizontal federated learning (HFL) paradigm in which only the model and gradients are exchanged considering intra-domain privacy. We will not detail the horizontal federation model (e.g., further privacy guarantee and more high-order information) as it is a well established FL model and not our novel contribution. This model can be replaced by other GAT-based FL models [33, 43] as well.

3.2.2 Positive knowledge transfer module

After the source domain training, we obtain a series of source models in individual client's user space. Our positive knowledge transfer module then prepares positive knowledge to be transferred from each source domains $\mathcal{D}^S$ to the target domain $\mathcal{D}^T$, while protecting inter-domain privacy (**CH1**). Specifically, suppose a individual user (client) u and a source domain $\mathcal{D}^{S_i}$, we transfer the user u's embedding matrix $\mathbf{X}_{S_i}$ in $\mathbb{R}^{L \times d}$. Take the row l of the matrix (i.e., $\mathbf{x}_{S_i}^l$) as an example, it is the user u's embedding output by the l-th message propagation layer. In an ideal scenario (i.e., we transfer totally positive knowledge without taking inter-domain privacy into account) [6], embedding matrices from different source domains can be directly used to enhance target domain local training in client u. By utilizing the source domain embeddings, u's final target domain embedding $\mathbf{e}_T^l$ of layer l is:

$$\mathbf{e}_T^l = f_T(\mathbf{x}_T^l, \mathbf{x}_{S_1}^l, ..., \mathbf{x}_{S_{M-1}}^l), \quad l \in [1, L] \tag{5}$$

where $f_T(\cdot)$ is the function that the target domain aggregates the knowledge of the source domains and we will give its expression in Subsection 3.2.3. In this process, the transfer of knowledge between domains takes place entirely in the user u's local space. Such a fully localized mode of knowledge transfer avoiding the additional communication overhead and potential privacy issues [6]. However, this direct emebddings transfer does not meet the privacy and NT constrains in BS-CDR scenarios.

Privacy-preserving knowledge extraction In existing CDR frameworks, the user or item embedding was shared as knowledge [9, 15, 6], which neglects inter-domain privacy. In a GNN-based approach, such direct transfers are subject to privacy attacks. Each message propagation layer can be viewed as a function with user and item embeddings as input. An attacker can easily obtain the user's private rating matrix based on these embeddings. We apply DP to the souce domain embeddings $\mathbf{x}_{S_i}$ [20, 44] to safeguard inter-domain privacy.

THEOREM 1. *By perturbing the source domain embeddings with Gaussian noise, the reconstructed data of the ideal attack deviates from the real data and prevents a perfect reconstruction.*

In FedGCDR, we adopt the Gaussian mechanism to the source domain embedding $\mathbf{x}_{S_i}$ to obtain $\hat{\mathbf{x}}_{S_i}$ for knowledge transfer. Detailed privacy analysis is included in Appendix A.

Feature mapping User features (i.e., user embeddings) could represent personal preferences and are influenced by domain features (e.g., items). The discrepancy of domains leads to the heterogeneity of feature space between domains which means that source domain embeddings cannot be utilized directly by the target domain. Man et al. [15] show that there exists an underlying mapping relationship between the latent user matrix of different domains, which can be captured by a mapping function. In order to alleviate NT, we adopt a series of MLP to explore mapping functions for each source domain. Adding Gaussian noise and feature mapping, Equation (5) becomes:

$$\mathbf{e}_T^l = f_T(\mathbf{x}_T^l, MLP_1(\hat{\mathbf{x}}_{S_1}^l), ..., MLP_{M-1}(\hat{\mathbf{x}}_{S_{M-1}}^l)). \tag{6}$$

To learn more effective mapping function, we adopt a mapping loss term:

$$l_m = \sum_{i=1}^{M-1} \sum_{l=1}^{L} ||\mathbf{x}_T^l - MLP_i(\hat{\mathbf{x}}_{S_i}^l)||^2, \tag{7}$$

3.2.3 Positive knowledge activation module

After the aforementioned operations, the target domain obtains a list of source domain matrices $\hat{\mathbf{X}}_{S_1}, \hat{\mathbf{X}}_{S_2}, ..., \hat{\mathbf{X}}_{S_{M-1}}$. The row of the matrices represent $MLP_i(\hat{\mathbf{x}}_{S_i}^l)$. It is worth noting that for source domains where a user has no rating, $\hat{\mathbf{X}}_{S_i}$ is a Gaussian noise matrix. Our consideration for doing this is: (1) no rating may also suggest a preference; (2) this is beneficial for enhancing the model's capability to filter noise and identify NT. With the knowledge from the source domains, the positive knowledge activation module is to alleviate NT after the knowledge transfer (**CH2**). Although we have aligned the feature space in the previous module, the Gaussian noise that has been fed to the target domain with source domain embedding matrices leads to potential NT. How to utilize the transferred knowledge remains a great challenge.

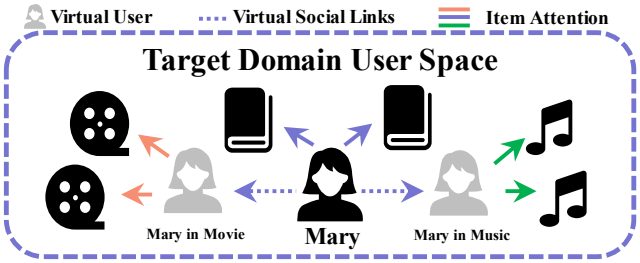


Figure 3: Illustration of target domain graph expansion. The virtual users are constructed with the source domain embeddings from the Movie domain and the Music domain. The attentions generated by social links to the virtual user can be regarded as the domain attentions.

Graph expansion and target domain training To alleviate NT, common approaches are to generate domain attention by predefined domain features [13] or to control the transfer ratio of source domains by hyper-parameters [7]. These methods are only applicable to a limited number of domains and have excessive human intervention. In FedGCDR, we take an attention-based approach. First, we expand the u's (Mary's) local graph of the target domain as shown in Figure 3. For the source

domain embedding matrices $\hat{\mathbf{X}}_{S_1}, \hat{\mathbf{X}}_{S_2}, ..., \hat{\mathbf{X}}_{S_{M-1}}$, we represent them as $M-1$ virtual users. Since the virtual users constructed by source domain embeddings are actually the same person u, they have continuity in preference and their features (i.e., embeddings) characterize u's preference from different dimensions. Inspired by social recommendation [45, 46, 47], we consider that there is a implicit social relationship between virtual users and the actual user u, because of the continuity in their preferences. Then, we build virtual social links between them to expand the original target domain graph. Second, by applying this expanded graph in target domain training, GAT generates the corresponding attention coefficient for the virtual users, which can be understood as domain attentions. Leveraging the domain attention coefficients, the target domain can focus on domains that transfer positive knowledge and we can finally give $f_T(\cdot)$:

$$f_T(x_T^l, MLP_1(\hat{\mathbf{x}}_{S_1}^l), ..., MLP_{M-1}(\hat{\mathbf{x}}_{S_{M-1}}^l)) = a_{uu}^l \mathbf{x}_T^l + \sum_{v \in N_u} a_{uv}^l \mathbf{e}_v^l + \sum_{i=1}^{M-1} a_i^l MLP_i(\hat{\mathbf{x}}_{S_i}^l), \quad (8)$$

where a_i^l is the domain attention of source domain i generated by the l-th layer. Beside, we introduce a social regularization term to strengthen the virtual social links:

$$l_s = \sum_{l=1}^{L} \parallel \mathbf{x}_T^l - \frac{\sum_{i=1}^{M-1} Sim(\mathbf{x}_T^l, \hat{\mathbf{x}}_{S_i}^l) \times \hat{\mathbf{x}}_{S_i}^l}{\sum_{i=1}^{M-1} Sim(\mathbf{x}_T^l, \hat{\mathbf{x}}_{S_i}^l)} \parallel^2, \quad (9)$$

Since we do not have direct access to the rating matrix of each source domain, the function $Sim(\cdot)$ calculates the cosine similarity [45].

Through the graph expansion, we achieve: (1) dynamic domain attentions focusing on positive source domain knowledge to alleviate NT; (2) domain attentions are generated by GAT to avoid human interventions such as hyper-parameter tuning or feature engineering.

For top-k recommendation, we adopt a widely-used inner product model to estimate the value of target domain rating $\mathbf{R}_{uv}^T$, which is the interaction probability between a pair of user u and item v:

$$\hat{\mathbf{R}}_{uv}^T = Sigmoid(\mathbf{e}_T^u \cdot \mathbf{e}_T^v), \quad (10)$$

where $\mathbf{e}_T^u$ and $\mathbf{e}_T^v$ are the final user and item embeddings output by GAT. Our objective function consists of three terms as follows:

$$L_{GAT} = BCELoss(\hat{\mathbf{R}}_{uv}^T, \mathbf{R}_{uv}^T) + \frac{\alpha}{2} l_m + \frac{\beta}{2} l_s, \quad (11)$$

where α and β are hyper-parameters, and $BCELoss(\cdot)$ is the binary cross-entropy loss [48]. The target domain federated GAT training with the expanded graph following the HFL paradigm.

Target model fine-tuning After target domain training with the expanded graph, the target domain GAT model assimilated knowledge from the source domains. However, NT may still be inevitable, and negative knowledge may have accumulated in the target domain. An example is that the Gaussian noise matrices from the source domains where the user has no rating data. On the basis of this consideration, we adopt an additional fine-tuning stage: First, we freeze the message propagation layer of GAT to isolate the influence of source domains. This is because the Gaussian noise still floods through the transfer process. Second, we directly train the well-informed embeddings generated by the target domain GAT. These steps adapt the learned external knowledge for predicting the target domain ratings. In this process, we use the loss of prediction in Equation (11) as the object function:

$$L_{ft} = BCELoss(\hat{\mathbf{R}}_{uv}^T, \mathbf{R}_{uv}^T). \quad (12)$$

We provide a computational analysis and a communication analysis of FedGCDR in Appendix B.

4 Experiments

4.1 Experimental setup

Datasets We study the effectiveness of FedGCDR with 16 popular domains on a real-world dataset **Amazon** [49]. To study the impact of the number of domains on model performance, we divide these domains into three subsets containing 4, 8, and 16 domains respectively and denote them as **Amazon-4**, **Amazon-8**, and **Amazon-16** respectively. We filter the original data in different ways, and more details are given in Appendix C.1. In our experiments, Books and CDs are selected as the target domains. For the ratings in each domain, we first convert them to implicit data, where entries corresponding to existing user-item interactions are marked as 1 and others are marked as 0.

Baselines We compare FedGCDR with the following state-of-the-art models: (1) **FedGNN** [33] is an attempt to adopt FL graph learning to recommender systems. Its recommendation performance could represent the data quality of the target domain and reflect negative transfer. In Tables 1 and 2, in order to distinguish **FedGNN** from the CDR baselines, we denote it by **Single Domain**. (2) **EMCDR** [15] is a conventional embedding-mapping CDR framework. We adjust it to the HFL framework following [6]. (3) **PriCDR** [17] is a privacy-preserving CDR framework, which adopted DP on the rating matrices rating matrix to ensure privacy. (4) **FedCT** [6] is a VAE-based federated framework that is the first attempt to protect intra-domain privacy in cross-domain recommendations. (5) **FedCDR** [7] is a dual-target federated CDR framework, where the user embeddings are transferred as knowledge to enhance the other domain's model training. To adapt to the BS-CDR scenarios, we modify **FedCDR** by applying embedding averaging when receiving source domain embeddings.

Table 1: The recommendation performance on Amazon@Books. The best result for the same setting is marked in bold and the second best is underlined. Single Domain represents FedGNN and its performance is exactly the same on three sub-datasets.

Model	Amazon-4@Books				Amazon-8@Books				Amazon-16@Books			
	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10
Single Domain	0.4693	0.3188	0.6067	0.3634	0.4693	0.3188	0.6067	0.3634	0.4693	0.3188	0.6067	0.3634
EMCDR	0.4633	0.3075	0.6179	0.3191	0.4678	0.3268	0.5990	0.3518	0.3140	0.2184	0.4207	0.2348
PriCDR	0.4061	0.3159	0.5275	0.3550	0.4409	0.3196	0.5913	0.3681	0.3699	0.2650	0.4914	0.3042
FedCT	0.2911	0.2044	0.4276	0.2482	0.4665	0.3516	0.6002	0.3939	0.2779	0.2335	0.3580	0.2593
FedCDR	0.4115	0.3153	0.5415	0.3570	0.4791	0.3538	0.6182	0.3967	0.3926	0.2907	0.5626	0.3403
FedGCDR-DP	0.4903	0.3417	0.6717	0.3733	0.5224	0.3608	0.6727	0.3973	0.4928	0.3509	0.6510	0.3742
FedGCDR	**0.4941**	**0.3592**	**0.6732**	**0.3920**	**0.5300**	**0.3686**	**0.6752**	**0.3985**	**0.5016**	**0.3600**	**0.6516**	**0.3854**

Table 2: The recommendation performance on Amazon@CDs.

Model	Amazon-4@CDs				Amazon-8@CDs				Amazon-16@CDs			
	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10
Single Domain	0.4119	0.2751	0.5031	0.3040	0.4119	0.2751	0.5031	0.3040	0.4119	0.2751	0.5031	0.3040
EMCDR	0.4074	0.2651	0.5591	0.2972	0.2882	0.1828	0.4361	0.2199	0.4704	0.3683	0.5740	0.3937
PriCDR	0.3987	0.2838	0.5114	0.3202	0.2946	0.1988	0.4229	0.2400	0.4405	0.3689	0.5399	0.4011
FedCT	0.2681	0.1603	0.3774	0.1956	0.1801	0.1282	0.3001	0.1681	0.3522	0.2963	0.4326	0.3219
FedCDR	0.4299	0.2949	0.5636	0.3381	0.3088	0.2109	0.4620	0.2600	0.4823	0.3983	0.5808	0.4297
FedGCDR-DP	0.4359	0.2960	0.5779	0.3520	0.4122	0.2983	0.5064	0.3106	0.4963	0.4061	0.6135	0.4453
FedGCDR	**0.4588**	**0.3282**	**0.5819**	**0.3679**	**0.4276**	**0.3142**	**0.5270**	**0.3464**	**0.5267**	**0.4382**	**0.6208**	**0.4684**

We provide implement details in Appendix C.2. and additional experimental results in Appendix D.

4.2 Recommendation performance

We report the model performance results in Tables 1 and 2. **Single domain** shows that the Book domain has better single-domain recommendation accuracy than the Music domain, which represents higher data quality and quantity. Under BS-CDR settings, **FedGCDR** outperforms all CDR baselines on all three sub-datasets, which confirms the effectiveness of the proposed model on real-world data.

To further study our model capacity in alleviating negative transfer, we first define two types of negative transfer: (1) Soft Negative Transfer (SNT), where recommendation models' performance under the multi-domain setting is worse than that under the single-domain setting. This means that the knowledge from source domains poisoning the target domain's model training. (2) Hard Negative Transfer (HNT), where recommended performance of a large number of source domains is lower than that of a small number of source domains. This means that the newly added domains are not conducive to the training of the target domain or conflict with the already added source domain.

Taking the Books domain as the target domain, **EMCDR**, **PriCDR**, **FedCT** and **FedCDR** both have serious negative transfer problems and lower performance on the three data subsets. From the SNT perspective, their performances is much worse than that of **Single Domain** as shown in Figure 4. From the HNT perspective, their performances under 16-domain settings is worse than that under the 8-domain and 4-domain settings, which suggests it is not appropriate to recklessly transfer knowledge to a well-informed domain. Our **FedGCDR** model successfully alleviates NT with consistently best

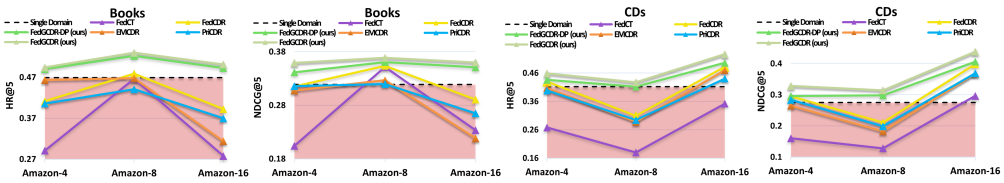


Figure 4: Illustrations of negative transfer on HR@5 and NDCG@5. Metric values lower than single-domain (dotted line and red area) mean severe negative soft negative transfer. The figure on HR@10 and NDCG@10 is shown in Appendix D.1.

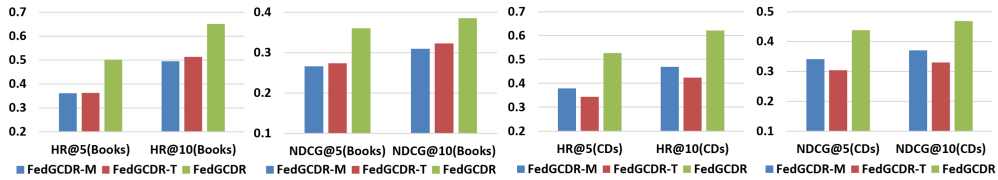


Figure 5: Ablation study on Amazon-16@CDs and Amazon-16@Books.

and stable performance results. For the CDs domain, which has lower-quality data, the performance of the CDR models greatly improves with less NT in Figure 4. From the SNT perspective, negative transitions now has a lighter impact for all models. From the HNT perspective, on Amazon-8, the performance of all models decrease, which we believe is due to the strong negative knowledge generated by the four additional domains on top of Amazon-4. On Amazon-16, all methods achieve best performance which indicates more knowledge from the source domains helpd improve the model performance in the CDs domain. Overall, the capability of **EMCDR**, **PriCDR** and **FedCDR** to alleviate negative transfer is much higher than that of **FedCT**. This is because the proportion of target domain features in the final feature is guaranteed by tuning hyper-parameters to control the transfer ratio of the source domain. Meanwhile, **FedGCDR** avoids this kind of human involvement and maintains performance optimality on three sub-datasets. In conclusion, our experiments show the superiority of **FedCDR** in recommendation performance and the effectiveness of alleviating NT.

4.3 Ablation study

To study the contribution of each module of **FedGCDR**, we implement two model variants, **FedGCDR-M** and **FedGCDR-T**. **FedGCDR-T** transfers the source domain embeddings without mapping. **FedGCDR-M** replaces the attention graph expansion with the average sum of source domain embeddings and removes the fine-tuning. We experiment with Books and CDs as target domains on the Amazon-16 dataset. The experimental results are shown in Figure 5. We make the following observations: (1) The two variants perform differently on different target domains. On the Books domain, **FedGCDR-T** performs better than **FedGCDR-M**, which indicates that for domains with higher data quality, preventing the transfer of negative knowledge from other domains is more important than mapping this knowledge better, and the Positive Knowledge Activation module meets the requirements of such domains. On the CDs domain, **FedGCDR-M** performs better than **FedGCDR-T**, which indicates that for domains that are deficient in information, mapping knowledge correctly is more important than preventing inter-domain negative knowledge, and the Positive Knowledge Transfer module meets these requirements. (2) Compared to **FedGCDR**, the absence of either module can cause a significant drop in performance. This indicates that in cross-domain recommendation, we should not only focus on transferring positive knowledge, but also control the spread of negative knowledge to the target domain, especially when a large number of domains.

5 Limitations

Our experiments were conducted on 16 domains of the Amazon dataset. While this extensive dataset covers broader source domains, relying on a single dataset may limit the generalizability of our model to data from other sources. Our approach uses overlapping users as a cross-domain bridge. Indeed, there are no widely-recognized cross-domain recommendation (CDR) datasets with more

than three domains, aside from the Amazon dataset. Despite this limitation, we firmly believe that the significant improvements in privacy preservation and model performance demonstrated by FedGCDR underscore its superiority.

6 Conclusion

We proposed FedGCDR, a federated graph learning framework designed for BS-CDR. FedGCDR addresses the critical challenge of privacy preservation and negative transfer by employing a positive knowledge transfer module and a positive knowledge activation module. Our privacy-preserving method achieves best recommendation quality results on 16 domains of the Amazon dataset. In the future, we aim to extend FedGCDR to improve the recommendaiton performance of both the target and the source domains.

References

[1] Weiming Liu, Jiajie Su, Chaochao Chen, and Xiaolin Zheng. Leveraging distribution alignment via stein path for cross-domain cold-start recommendation. *Advances in Neural Information Processing Systems*, 34:19223–19234, 2021.

[2] Tianzi Zang, Yanmin Zhu, Haobing Liu, Ruohan Zhang, and Jiadi Yu. A survey on cross-domain recommendation: taxonomies, methods, and future directions. *ACM Transactions on Information Systems*, 41(2):1–39, 2022.

[3] Feng Zhu, Yan Wang, Chaochao Chen, Jun Zhou, Longfei Li, and Guanfeng Liu. Cross-domain recommendation: challenges, progress, and prospects. *arXiv preprint arXiv:2103.01696*, 2021.

[4] Meng Liu, Jianjun Li, Guohui Li, and Peng Pan. Cross domain recommendation via bi-directional transfer graph collaborative filtering networks. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 885–894, 2020.

[5] Jiangxia Cao, Jiawei Sheng, Xin Cong, Tingwen Liu, and Bin Wang. Cross-domain recommendation to cold-start users via variational information bottleneck. In *2022 IEEE 38th International Conference on Data Engineering*, pages 2209–2223. IEEE, 2022.

[6] Shuchang Liu, Shuyuan Xu, Wenhui Yu, Zuohui Fu, Yongfeng Zhang, and Amelie Marian. Fedct: Federated collaborative transfer for recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 716–725, 2021.

[7] Wu Meihan, Li Li, Chang Tao, Eric Rigall, Wang Xiaodong, and Xu Cheng-Zhong. Fedcdr: federated cross-domain recommendation for privacy-preserving rating prediction. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 2179–2188, 2022.

[8] Zhiwei Liu, Liangwei Yang, Ziwei Fan, Hao Peng, and Philip S Yu. Federated social recommendation with graph neural network. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(4):1–24, 2022.

[9] Gaode Chen, Xinghua Zhang, Yijun Su, Yantong Lai, Ji Xiang, Junbo Zhang, and Yu Zheng. Win-win: a privacy-preserving federated framework for dual-target cross-domain recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 4149–4156, 2023.

[10] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big Data*, 3:1–40, 2016.

[11] Nidhi Agarwal, Akanksha Sondhi, Khyati Chopra, and Ghanapriya Singh. Transfer learning: Survey and classification. *Smart Innovations in Communication and Computational Sciences 2020*, pages 145–155, 2021.

[12] Wen Zhang, Lingfei Deng, Lei Zhang, and Dongrui Wu. A survey on negative transfer. *IEEE/CAA Journal of Automatica Sinica*, 10(2):305–329, 2022.

[13] Hongwei Zhang, Xiangwei Kong, and Yujia Zhang. Selective knowledge transfer for cross-domain collaborative recommendation. *IEEE Access*, 9:48039–48051, 2021.

[14] Xu Yu, Dingjia Zhan, Lei Liu, Hongwu Lv, Lingwei Xu, and Junwei Du. A privacy-preserving cross-domain healthcare wearables recommendation algorithm based on domain-dependent and domain-independent feature fusion. *IEEE Journal of Biomedical and Health Informatics*, 26(5):1928–1936, 2021.

[15] Tong Man, Huawei Shen, Xiaolong Jin, and Xueqi Cheng. Cross-domain recommendation: An embedding and mapping approach. In *IJCAI*, volume 17, pages 2464–2470, 2017.

[16] Zhen Liu, Jingyu Tian, Lingxi Zhao, and Yanling Zhang. Attentive-feature transfer based on mapping for cross-domain recommendation. In *2020 International Conference on Data Mining Workshops (ICDMW)*, pages 151–158. IEEE, 2020.

[17] Chaochao Chen, Huiwen Wu, Jiajie Su, Lingjuan Lyu, Xiaolin Zheng, and Li Wang. Differential private knowledge transfer for privacy-preserving cross-domain recommendation. In *Proceedings of the ACM Web Conference 2022*, pages 1455–1465, 2022.

[18] Xinting Liao, Weiming Liu, Xiaolin Zheng, Binhui Yao, and Chaochao Chen. Ppgencdr: A stable and robust framework for privacy-preserving cross-domain recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 4453–4461, 2023.

[19] Zhongxuan Han, Xiaolin Zheng, Chaochao Chen, Wenjie Cheng, and Yang Yao. Intra and inter domain hypergraph convolutional network for cross-domain recommendation. In *Proceedings of the ACM Web Conference 2023*, pages 449–459, 2023.

[20] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[21] Bin Li, Qiang Yang, and Xiangyang Xue. Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction. In *Twenty-First International Joint Conference on Artificial Intelligence*, 2009.

[22] Ali Mamdouh Elkahky, Yang Song, and Xiaodong He. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In *Proceedings of the 24th International Conference on World Wide Web*, pages 278–288, 2015.

[23] Weiming Liu, Xiaolin Zheng, Mengling Hu, and Chaochao Chen. Collaborative filtering with attribution alignment for review-based non-overlapped cross domain recommendation. In *Proceedings of the ACM Web Conference 2022*, pages 1181–1190, 2022.

[24] Xiaoyun Zhao, Ning Yang, and Philip S Yu. Multi-sparse-domain collaborative recommendation via enhanced comprehensive aspect preference learning. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pages 1452–1460, 2022.

[25] Weiming Liu, Chaochao Chen, Xinting Liao, Mengling Hu, Yanchao Tan, Fan Wang, Xiaolin Zheng, and Yew Soon Ong. Learning accurate and bidirectional transformation via dynamic embedding transportation for cross-domain recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, number 8, pages 8815–8823, 2024.

[26] Weiming Liu, Xiaolin Zheng, Chaochao Chen, Mengling Hu, Xinting Liao, Fan Wang, Yanchao Tan, Dan Meng, and Jun Wang. Differentially private sparse mapping for privacy-preserving cross domain recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 6243–6252, 2023.

[27] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

[28] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020.

[29] Zheng Wang, Xiaoliang Fan, Jianzhong Qi, Chenglu Wen, Cheng Wang, and Rongshan Yu. Federated learning with fair averaging. In Zhi-Hua Zhou, editor, *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 1615–1623. International Joint Conferences on Artificial Intelligence Organization, 8 2021. Main Track.

[30] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020.

[31] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210, 2021.

[32] Di Chai, Leye Wang, Kai Chen, and Qiang Yang. Secure federated matrix factorization. *IEEE Intelligent Systems*, 36(5):11–20, 2020.

[33] Chuhan Wu, Fangzhao Wu, Yang Cao, Yongfeng Huang, and Xing Xie. Fedgnn: Federated graph neural network for privacy-preserving recommendation. *arXiv preprint arXiv:2102.04925*, 2021.

[34] Peihua Mai and Yan Pang. Vertical federated graph neural network for recommender system. In *International Conference on Machine Learning*, pages 23516–23535. PMLR, 2023.

[35] Weiming Liu, Chaochao Chen, Xinting Liao, Mengling Hu, Jianwei Yin, Yanchao Tan, and Longfei Zheng. Federated probabilistic preference distribution modelling with compactness co-clustering for privacy-preserving multi-domain recommendation. In *Proceedings of the 32rd International Joint Conference on Artificial Intelligence*, pages 2206–2214, 2023.

[36] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.

[37] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[38] Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. Graph neural networks in recommender systems: a survey. *ACM Computing Surveys*, 55(5):1–37, 2022.

[39] Chen Gao, Xiang Wang, Xiangnan He, and Yong Li. Graph neural networks for recommender system. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pages 1623–1625, 2022.

[40] Teng Xiao, Zhengyu Chen, Donglin Wang, and Suhang Wang. Learning how to propagate messages in graph neural networks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1894–1903, 2021.

[41] Yifei Zhang, Hao Zhu, Zixing Song, Piotr Koniusz, Irwin King, et al. Mitigating the popularity bias of graph collaborative filtering: A dimensional collapse perspective. *Advances in Neural Information Processing Systems*, 36:67533–67550, 2023.

[42] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 639–648, 2020.

[43] Chuhan Wu, Fangzhao Wu, Lingjuan Lyu, Tao Qi, Yongfeng Huang, and Xing Xie. A federated graph neural network framework for privacy-preserving personalization. *Nature Communications*, 13(1):3091, 2022.

[44] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

[45] Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. Recommender systems with social regularization. In *Proceedings of the fourth ACM International Conference on Web Search and Data Mining*, pages 287–296, 2011.

[46] Chaochao Chen, Liang Li, Bingzhe Wu, Cheng Hong, Li Wang, and Jun Zhou. Secure social recommendation based on secret sharing. *arXiv preprint arXiv:2002.02088*, 2020.

[47] Suman Deb Roy, Tao Mei, Wenjun Zeng, and Shipeng Li. Socialtransfer: cross-domain transfer learning from social streams for media applications. In *Proceedings of the 20th ACM international conference on Multimedia*, pages 649–658, 2012.

[48] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*, pages 173–182, 2017.

[49] Jianmo Ni, Jiacheng Li, and Julian McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 188–197, 2019.

[50] Xinjian Luo, Yuncheng Wu, Xiaokui Xiao, and Beng Chin Ooi. Feature inference attack on model predictions in vertical federated learning. In *2021 IEEE 37th International Conference on Data Engineering*, pages 181–192. IEEE, 2021.

[51] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

A Privacy analysis

Due to the algorithmic nature of GNN, the source domain embeddings we pass are a function result on the user embeddings and item embeddings. This means that in the event of a successful inference attack, our user item interaction matrix is exposed to the threat of privacy disclosure. So we use DP to further protect privacy.

Threat model In this paper, we assume the threat model to be semi-honest (honest-but-curious). Under this threat model, the participants adhere strictly to the FL protocol for collaborative model training. However, they are interested in the sensitive rating data and may attempt to extract as much information as possible from the transferred embeddings. Specifically, these semi-honest parties, i.e. the target domain, may employ inference attacks [50] on the embeddings to reconstruct or infer sensitive user-item interaction matrix of other domains.

DEFINITION 1 (THE GAUSSIAN MECHANISM). *Given a function $f : D \to \mathbb{R}^d$ over a dataset D, the Gaussian mechanism is defined as*:

$$F_G(x, f(\cdot), \epsilon) = f(x) + (r_1, ...r_k), \tag{13}$$

where r_i is the random noise drawn from $\mathcal{N} \sim (0, \sigma^2 \Delta_2 f^2)$ and $\sigma = \frac{\sqrt{2ln(1.25/\delta)}}{\epsilon}$. In FedGCDR, the intra-domain GAT-based federated model is considered as the function $f(\cdot)$.

THEOREM 2. *The Gaussian mechanism defined in Definition 1 preserves (ϵ, δ)-DP for each publication step [20].*

First, we give the definition of the inverse function:

DEFINITION 2 (INVERSE FUNCTION). *Given a function $f : D \to \mathbb{R}^d$ over a dataset D, the inverse function f^{-1} is defined as*:

$$f^{-1} = argmin_g \sum_{i \in u \bigcup v} \parallel \mathbf{e}_i - g(f(\mathbf{e}_u, \mathbf{e}_v)) \parallel_2, \tag{14}$$

where

$$\mathbf{e}_u, \mathbf{e}_v = Embedding(x), x \in D. \tag{15}$$

For the target domain, the embeddings received form source domains can be regarded as the functional result of their models. Let the function be $f(\cdot, \cdot)$ and the input e_u, e_v is the user embedding and item embedding respectively. The embeddings is the output $f(e_u, e_v)$. The target domain attempts to find a inference attack function $I(\cdot)$ which is as close to the inverse function as possible.

DEFINITION 3 (PRIVACY LEAKAGE). *Given a function $f : E \to \mathbb{R}^d$ over a Embedding set E and an inference function I, the privacy leakage Λ is defined as*:

$$\Lambda = \frac{1}{1 + \frac{1}{U} \sum Leak_u + \frac{1}{V} \sum Leak_v}. \tag{16}$$

where

$$Leak_u = \parallel \mathbf{e}_i - I_u(f(\mathbf{e}_u, \mathbf{e}_v)) \parallel_2, i \in U, Leak_v = \parallel \mathbf{e}_j - I_v(f(\mathbf{e}_u, \mathbf{e}_v)) \parallel_2, j \in V. \tag{17}$$

$\parallel e - I(f(e_u, e_v)) \parallel_2$ reflects the closeness of the reconstructed input to the true input. Therefore, privacy leakage (PLeak) Λ is able to reflect privacy leakage of FedGCDR with the inference function $I(\cdot)$. PLeak equal to 1 means a perfect reconstruction, and being close to zero means a bad reconstruction. DP on the embeddings further ensures that attackers cannot perfectly reconstruct the raw data.

THEOREM 2. *If PLeak equals to 1 with the inference function $I(\cdot)$, the function $f(\cdot, \cdot)$ is bijection.*

Proof. If the function $f(\cdot, \cdot)$ is not bijection, there are $i, j \in D$ and $i \neq j$, but $f(\mathbf{e}_u^i, \mathbf{e}_v^i) = f(\mathbf{e}_u^j, \mathbf{e}_v^j)$ and $I(f(\mathbf{e}_u^i, \mathbf{e}_v^i))$ and $I(f(\mathbf{e}_u^j, \mathbf{e}_v^j))$. This is a contradiction as the perfect reconstruction requires both $\mathbf{e}_u^i, \mathbf{e}_v^i = I(f(\mathbf{e}_u^i, \mathbf{e}_v^i))$ and $\mathbf{e}_u^j, \mathbf{e}_v^j = I(f(\mathbf{e}_u^j, \mathbf{e}_v^j))$ to achieve $\Lambda = 1$. Therefore, the function $f(\cdot, \cdot)$ must be is bijection.

THEOREM 3. *Given the lipschiz constant L of the function f at $x \in D$ with the noise generated by Gaussian mechanism $\mathcal{N}$ on embeddings. If $f(\mathbf{e}_u, \mathbf{e}_v) + \mathcal{N} \in f$, the distance between x to the reconstructed data of the attack $I(\cdot)$ which achieves $\Lambda = 1$ is bounded by $\frac{|\mathcal{N}|}{L}$.*

Proof. By Theorem 2, we have for $x \in D$ and $v \in f, I(f(\mathbf{e}_u, \mathbf{e}_v)) = (\mathbf{e}_u, \mathbf{e}_v)$ and $f(I(v)) = v$, From the Lipschitz continuous,

$$|e - I(f(\mathbf{e}_u, \mathbf{e}_v) + \mathcal{N})| \geq \frac{|f(\mathbf{e}_u, \mathbf{e}_v) - (f(\mathbf{e}_u, \mathbf{e}_v) + \mathcal{N})|}{L} = \frac{|\mathcal{N}|}{L}. \tag{18}$$

Therefore, by perturbing the source domain embedding with Gaussian mechanism, the reconstructed data of the ideal attack deviates from the real data and prevents a perfect reconstruction ($i.e., \Lambda = 1$).

B Cost analysis

Due to the complexity of the FedGCDR pipeline, we perform a theoretical analysis of the computational and communication cost of FedGCDR accordance with the HVH pipeline including horizontal source domain training, vertical positive knowledge tranfer module and horizontal positive knowledge activation module.

B.1 Computational cost

Given a GAT model, let V be the node set, E be the edge set, and F the embedding size. The computational cost of one propagation layer of classic GAT framework is $O(|V|FF' + |E|F')$ [37]. In the horizontal source domain training, our model is a simplified GAT variants which discard feature transformation and non-linear activation. For a N^K layer model, the simplified computational cost is $O(N^K|E|F)$. In the vertical positive knowledge tranfer module, space mapping is carried by a N^m layers' MLP with computational cost $O(N^m F^2)$. In the horizontal positive knowledge activation module, the first part is the simplified GAT model and the second part is the fine-tuning model with computational cost $O(F^2)$. In conclusion, for the FedCDR framework with $N^{\mathcal{D}}$ domains, T^G GAT-based federated model traing epochs, and T^F fine-tuning epochs, the total computational cost is $O(T^G(N^{\mathcal{D}}N^K|E|F + N^m F^2) + T^F F^2))$. Cause $N^m F^2 \ll N^{\mathcal{D}}N^K|E|F$, we get the final computational cost $O(T^G N^{\mathcal{D}}N^K|E|F + T^F F^2))$.

B.2 Communication cost

In FedGCDR, the global model and item embeddings are held by the domain server. Let I be the item set and F be the embedding size. The space complexity of global model and item embeddings are $O(F)$ and $O(|I|F)$ respectively. In the horizontal source domain training, the domain server distributes the global model and item embeddings and get the gradient with the same size. The communication cost is $O(F + |I|F)$. In the vertical positive knowledge tranfer module, the N^m layers' MLP and its gradients are transmitted with the commnication cost $O(N^m F^2)$. In the horizontal positive knowledge activation module, the target domain additionally perform a fine-tuning stage with comunication cost $O(|I|F)$. In conclusion, for the FedCDR framework with N^u users, T^G GAT-based federated model traing epochs, and T^F fine-tuning epochs, the total communication cost is $O(T^G N^u(N^m F^2 + F + |I|F) + T^F N^u|I|F)$. Cause $N^m F^2 + F \ll |I|F$, we get $O(N^u|I|F(T^G + T^F))$. According to the expression, the communication cost of our FedGCDR is basically equivalent to the cost of two HFL progress. The cost is reduced because knowledge transfer takes place in user space, thus avoiding large-scale information exchange.

C Experimental details

C.1 Dataset details

The Amazon dataset we used is the 2018 version and can be easily accessd in `https://cseweb.ucsd.edu/~jmcauley/datasets/amazon_v2/`. In addition to multi-domain experiments, we randomly selected 2500 overlapping users in the Books domain and CDs domain to construct the dataset **Amazon-Dual**, so as to validate the performance of our FedGCDR in the conventional dual-domain scenarios where users full-overlap. The statistics of sub-datasets and processing details

are shown in Table 3 and Table 4. The bottleneck time of FedGCDR is the federated-GAT training time in each domain, and we also show it in Table 4.

Table 3: Statistics on the Amazon Dataset. (min-median-max) values are provided for $|U_d|$, $|I_d|$ and $|R_d|$.

Dataset	$\|U\|$	$\|U_d\|$	$\|I_d\|$	$\|R_d\|$	avg (sparsity)
Amazon-4	55,518	6,632 - 12,626 - 27,402	53,082 - 134,438 - 501,153	623,420 - 646,266 - 5,481,801	0.0802%
Amazon-8	99,506	6,632 - 13,978 - 27,402	53,082 - 106,985 - 501,153	186,016 - 618,539 - 5,481,801	0.0399%
Amazon-16	117,672	1,036 - 9,038 - 27,402	17,209 - 64,624 - 501,153	41,427 - 379,657 - 5,481,801	0.0928%
Amazon-Dual	2,500	2,500 - 2,500 - 2,500	17,889 - 28,649 - 39,510	106,741 - 128,601 - 150,461	0.1955%

Table 4: Processing details on Amazon Dataset.

Domain	$\|U\|$	$\|I\|$	user core	item core	time per epoch (mm:ss)	
Clothing, Shoes and Jewelry	11,558	197,677	24	10	03:31	Amazon-4
Books	27,402	501,153	96	10	11:02	
CDs and Vinyl	13,694	71,199	24	10	04:02	
Movies and TV	6,632	53,082	48	10	01:55	
Home and Kitchen	15,772	135,182	48	10	04:36	Amazon-8
Electronics	16,836	120,876	32	10	04:40	
Sports and Outdoors	14,262	93,095	32	10	03:41	
Cell Phones and Accessories	9,312	55,312	24	10	02:40	
Tools and Home Improvement	9,899	65,378	16	10	02:47	Amazon-16
Toys and Games	5,267	63,870	32	10	01:10	
Automotive	6,135	62,188	32	10	01:32	
Pet Supplies	4,280	31,853	32	10	00:55	
Kindle Store	8,756	82,874	48	10	02:06	
Office Products	1,266	17,209	32	10	00:16	
Patio, Lawn and Garden	1,036	17,605	32	10	00:16	
Grocery and Gourmet Food	3,415	36,292	32	10	00:50	

C.2 Implement details

We provide the implemented details of our proposed model and baselines. We set batch size = 256 and latent dim = 8 for all domains. The number of propagation layer of GAT-base federated model is set to 2. The MLP has two hidden layers with size={16, 4}.Considering the trade-off between recommendation performance and privacy preservation, we set ϵ to 8 and σ to 10^{-5}. We set α=0.01 and β=0.01 which are the two hyper-parameters of the objective function $L_{GAT}(\cdot)$. When training our models, we choose Adam as the optimizer, and set the learning rate to 0.01 both in GAT-based federated model training and the fine-tuning stage. To evaluate the recommendation performance, we use the leave-one-out method which is widely used in recommender systems [48]. Specifically, we held out the latest interaction as the test set and utilized the remaining data for training. Then, we follow the common strategy which randomly samples 99 negative items that are not interacted with by the user for the rank list generation of the test set. We consider the top-k recommendation task as the main experiment so we choose metrics including Hit Ratio (HR)@K score and the Normalized Discounted Cumulative Gain (NDCG)@K [51] of the top-K ranked items with K=5, 10. We conduct the experiments on three groups of random seeds and report the average results. We conduct all the experiments on NVIDIA 3090 GPUs.

D Additional experimental results

D.1 Neagtive transfer on HR@10 and NDCG@10.

For HR@10 and NDCG@10 in Figure 6, our method and baselines show similar trends to the previous HR@5 and NDCG@5. Compared to Figure 6, the slight difference is that FedCT's HR@10 performance is better on Amazon-8@CDs than on Amazon-4@CDs. We believe that the reason is the poor performance of FedCT on Amzon-4@CDs lowers the threshold for negative transfer of the newly added source domain.

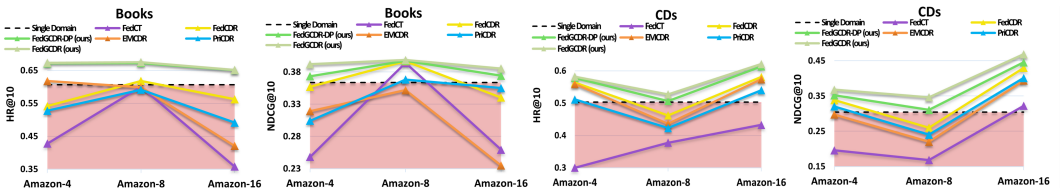


Figure 6: Illustrations of negative transfer on HR@10 and NDCG@10

D.2 Dual-domain scenario

Table 5: Dual-domain CDR performance.

Model	Books → CDs		Books ← CDs	
	HR@10	NDCG@10	HR@10	NDCG@10
Single Domain	0.2713	0.1429	0.2594	0.1524
EMCDR	0.2816	0.1409	0.2596	0.1540
PriCDR	0.2903	0.1446	0.2662	0.1583
FedCT	0.2384	0.1239	0.2570	0.1551
FedCDR	0.2566	0.1376	0.2657	0.1554
FedGCDR-DP	0.3076	0.1552	0.2749	0.1602
FedGCDR	**0.3323**	**0.1838**	**0.2958**	**0.1797**

According to the experimental results shown in Table 5, our **FedGCDR** achieved the best experimental metrics in both knowledge transfer directions. This shows that our approach is also suitable for traditional scenarios where users full-overlap and have only a single source domain and a single target domain.

D.3 Privacy budget

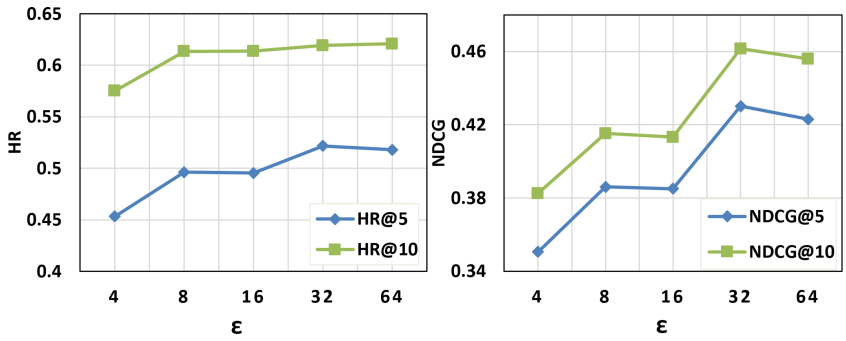


Figure 7: The effect of ϵ in DP on model performance.

To study the effects of privacy budget ϵ on the model performance, we vary the privacy budget $\epsilon = \{4, 8, 16, 32, 64\}$ to affect the σ. We experimented on Amazon-16 with CDs as the target domain and fix $\delta = 10{-}5$. We report the results Figure 7. From that we can observe that the model's performance decreases as ϵ decreases, but the model performance is not completely destroyed by Gaussian noise. Thus, there is a trade-off between accuracy and privacy, where a smaller ϵ value adds more noise to embeddings for stronger privacy preservation but leads to more prediction error. Therefore, to balance the data privacy preservation capacity and the model performance, we set it as $\epsilon = 8$.

E Broader impacts

Our proposed FedGCDR is tailored for BS-CDR, focusing on the privacy and negative transfer problems. CDR is widely uesd, while BS-CDR is generic and close to the reality. Our approach can

better mine user preferences and effectively protect privacy. On the one hand, users will benefit from more accurate recommendations and thus have a better experience in shopping, watching movies, etc. On the other hand, various economic entities can gain more profits.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have claimed the contributions and scope in lines 5-7 and 69-75.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have discussed the limitations in lines 318-325.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We have given the full set of assumptions and proof in Appendices A and B.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have provided experimental setup in section 4 and more experimental details in Appendix C.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide the code in supplemental material. For datasets, we provide the data processing code and the public benchmark is easy to access via the link in code file.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We have provided implement details in Appendix C.2 which contains data splits, hyper-parameters, type of optimizer, etc.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Our reported results are averaged over 3 runs with different random seeds.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We have provided information the computer resources in Appendices C.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer:
   answerYes

   Justification: We have fully reviewed the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have discussed social impacts in Appendix E.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our method well-address the privacy issure and poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have cited the original paper and proviced necessary information in the line 246 and Appendix C.1.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We totally use public benchmarks.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.