Privacy-Preserving Federated Cross-Domain Social Recommendation

Jianping Cai^1, Yang Liu^2, Ximeng Liu^{1*}, Jiayin Li^1, Hongbin Zhuang^1

^{1}College of Computer and Data Science, Fuzhou University, Fuzhou 350108, China
^{2}Institute for AI Industry Research, Tsinghua University, Beijing 100084, China
jpingcai@163.com, liuy03@air.tsinghua.edu.cn, {snbnix,lijiayin2019,hbzhuang476}@gmail.com

Abstract

By combining user feedback on items with social networks, cross-domain social recommendations provide users with more accurate recommendation results. However, traditional cross-domain social recommendations require holding both data of ratings and social networks, which is not easy to achieve for both information-oriented and social-oriented websites. To promote cross-domain social network collaboration among the institutions holding different data, we propose a federated cross-domain social recommendation (FCSR) algorithm. The main innovation is applying Random Response mechanism to achieve sparsely maintained differential privacy for user connections and proposing Matrix Confusion Method to achieve efficient encrypted user feature vector updates. Our experiments on three datasets show the practicality of FCSR in social recommendation and significantly outperforms baselines.

1 Introduction

Nowadays, recommendation systems are playing an essential role in modern business. It can accurately predict users' preferences and recommend items of interest to them, which undoubtedly brings excellent business value to websites that hold users' feedback. Besides users' feedback, the social connections of users have been proven to improve the quality of recommendations [Guo *et al.*, 2015], which has attracted the attention of several studies [Fan *et al.*, 2019; Liu *et al.*, 2020; Wang *et al.*, 2017] for cross-domain social recommendations. Unfortunately, user feedback and social networks are often not in the hands of one site. The information-oriented websites (IOW) that only hold users' feedback have to seek cooperation with social-oriented websites to improve the quality of recommendations. However, collaborative modeling is often not easy to implement. Most websites cannot freely share user information due to policy or privacy concerns, which leads to traditional centralized modeling techniques failing to be implemented due to a lack of necessary data.

Federated learning [Yang *et al.*, 2019] (FL) has recently been shown to be a promising learning framework for facilitating collaborative modeling among multiple parties without sharing any raw training data. For cross-domain social recommendations, FL allows the information-oriented websites and the social-oriented websites to collaboratively build more accurate recommendation models without exposing their respective data. However, federated cross-domain social recommendations are currently facing the following multifaceted challenges. First, although some centralized social recommendation methods [Fan *et al.*, 2019; Wang *et al.*, 2017] and purely federated recommendation algorithms [Yang *et al.*, 2020] have been proposed, they cannot be directly applied to federated cross-domain recommendations. The key challenge is how to ensure the security of users' individual privacy in social networks. Second, FL requires applying privacy-preserving techniques to secure local data, but some techniques trade off at the cost of significant computational overhead, such as homomorphic encryption [Paillier, 1999]. Since federated cross-domain social recommendations involve complex algebraic operations, achieving high enough computational efficiency while ensuring security is another important challenge we face. Besides, the sparsity of data is an essential characteristic of social recommendations [Cui *et al.*, 2021]. Effectively exploiting the sparsity to achieve high efficiency is a fundamental problem faced in social recommendations.

In response to the challenges, we contribute the following:

1) We propose a federated cross-domain social recommendation (FCSR) algorithm that treats the social-oriented website as a social services platform (SSP) and applies a FL framework to train social recommendation models without each participant's data.

2) We introduce a Random Response Mechanism to preserve individual privacy in SSP and design an efficient social network perturbation method, which avoids perturbing all possible social connections.

3) We propose a Matrix Confusion Method, which enables SSP to correctly update user feature vectors with high efficiency while the encrypted user feature vectors cannot be identified.

4) Facing the challenge of computing equations with large-scale sparse matrices, we propose a scheme to apply LU decomposition to improve the computational efficiency and study the impact of various decomposition strategies on the sparsity of the decomposed matrices.

*Contact Author

2 Preliminaries

2.1 Social Recommendation System

Given a set of user ratings for items such as (u_i, v_j, r_{ij}), indicating that user u_i rates item v_j as r_{ij}, a typical recommendation system tries to predict the users' potential ratings for each item and then recommends items of interest to users from the predicted ratings.

To predict the ratings more accurately, collaborative filtering methods based on feature representations are widely used. The basic idea is to represent each user u_i and item v_j as feature vectors $\mathbf{u}_i$ and $\mathbf{v}_j$, respectively, and then train a model $\mathcal{M}$ such that the predicted rating $\widehat{r}_{ij} = \mathcal{M}(\mathbf{u}_i, \mathbf{v}_j)$ is as close as possible to the true rating r_{ij}.

The work close to ours is the cross-domain social recommendation approach proposed by [Wang *et al.*, 2017]. The main idea is to build training models for user ratings from the information-oriented domain and social networks from the social-oriented domain separately, and then exchange the common users' (bridge users in [Wang *et al.*, 2017]) feature vectors of both to achieve cross-domain social recommendation. However, the work only focuses on centralized learning, meaning rating data and social networks have to be aggregated together before learning. Unfortunately, users' ratings or social connections are often privacy-sensitive. The cross-domain social recommendation will not be achieved when data providers are reluctant to share data due to policy or commercial competition.

2.2 Vertical Federated Learning

As an emerging machine learning paradigm, FL can build a learning model exploiting distributed datasets of all participants without revealing private datasets [Yang *et al.*, 2019]. There are three categories of FL, i.e., horizontal federated learning (HFL), vertical federated learning (VFL) and federated transfer learning (FTL).

VFL can mainly be applied in two or more different collaborating institutions, which hold heterogeneous user data, but some of the users involved are common. In the VFL scenario, participating institutions are assumed to be untrustworthy and attempt to mine others' privacy through the learning process. For data privacy and security reasons, the participants of VFL cannot directly exchange data. Some privacy security methods are applied to VFL to preserve each participant's privacy, such as homomorphic encryption [Paillier, 1999], differential privacy [Dwork *et al.*, 2014], secure multi-party computation [Bogdanov *et al.*, 2008], and Diffie-Hellman Key Exchange [Raymond and Stiglic, 2000].

2.3 Differential Privacy

Differential privacy (DP) [Dwork *et al.*, 2006a] is a theoretically provable technique for protecting individual privacy and is widely used in Federated Learning. It protects individual privacy through data perturbation, defined as follows:

Definition 1 (ε-Differential Privacy)**.** *Given two neighboring datasets $\mathcal{D}$ and $\mathcal{D}'$, a random algorithm $\mathcal{A}$ satisfies ε-DP if its all outputs $O \in \mathrm{Range}(\mathcal{A})$ satisfies*

$$\Pr(\mathcal{A}(\mathcal{D}) = O) \leq e^{\varepsilon} \Pr(\mathcal{A}(\mathcal{D}') = O). \tag{1}$$

Traditional DP mechanisms are mostly for continuous and discrete data, such as Laplace mechanism [Dwork *et al.*, 2006b] for continuous data and Exponential mechanism [McSherry and Talwar, 2007] for discrete data. By studying the Rényi Differential Privacy [Mironov, 2017], Mironov proposed a random response mechanism (RRM) [Mironov, 2017] applicable to the binary release, which is defined as follows.

Definition 2 (Random Response Mechanism)**.** *Given a function* $\mathrm{f} : \mathcal{D} \mapsto \{0, 1\}$, *RRM achieves* $\widetilde{\mathrm{f}}(\mathcal{D})$ *satisfying* ε*-DP by*

$$\widetilde{\mathrm{f}}(\mathcal{D}) = \begin{cases} \mathrm{f}(\mathcal{D}) & \textit{with probability } p \\ 1 - \mathrm{f}(\mathcal{D}) & \textit{with probability } 1-p \end{cases}, \tag{2}$$

where $p = \frac{e^{\varepsilon}}{1+e^{\varepsilon}} \geq 0.5$.

In general, DP algorithm consists of several sub-algorithms. They satisfy the serial combination theorem [McSherry, 2009] as follows.

Theorem 1 (Serial Combination)**.** *Given k random algorithms $A_1, A_2, \ldots, A_k$ where A_i satisfies ε_i-DP, their combination satisfies $\left(\sum_{i=1}^{k} \varepsilon_i\right)$-DP.*

3 Problem Formulation

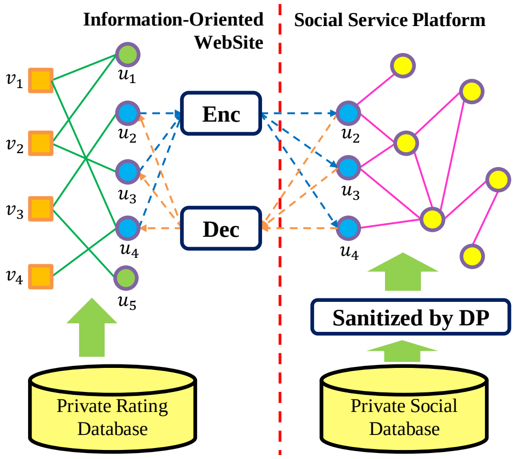


Figure 1: System model of FCSR

3.1 System Model

The system model of our FCSR is shown in Fig. 1. In the system model, we consider two FL participants: the information-oriented website (IOW) and the social service platform (SSP).

IOW: IOWs hold a large amount of user feedback data by providing information services. Using these data, they can realize recommendation systems by user ratings. IOW is the service requester, which initiates social service requests to SSP to improve recommendation quality.

SSP: SSP holds the users' social connections by providing social services. It is the service provider and is in charge of applying the privacy-preserving social relationships to improve the quality of the recommendation of collaborators.

As shown in Fig. 1, an IOW and SSP consist of a VFL system with two parties. They employ appropriate privacy security schemes to protect their respective local data. As a service requester, IOW encrypts the submitted data via encryption to ensure that SSP cannot obtain any available information. However, SSP cannot wholly hide its information as a service provider. Thus, we introduce DP to protect users' individual privacy in social networks.

3.2 Threat Model

In FCSR, SSP is considered to be semi-honest, i.e., SSP follows protocols to return the correct results, but actively try to learn the private information form the uploaded data. Meanwhile, we consider IOW as potentially malicious. IOW expects to mine users' social connections through machine learning or algebraic analysis. For this purpose, IOW may violate protocols by uploading some special constructed user features to capture users' social connections, or even directly invading SSP to illegally access the serving social network.

4 The Scheme of Federated Cross-Domain Social Recommendation

4.1 The Algorithm Framework

As shown in Fig. 1, our scheme consists of two parts: training the secure recommendation model on IOW and the individual privacy-secure social network service for SSP.

The rating data of IOW can be represented as the set $\mathcal{D}_r = \left\{\left(u_i^{(k)}, v_j^{(k)}, r_{ij}^{(k)}\right)\right\}_{k=1}^{m}$ with m elements. We denote the set of users and items involved as $\mathcal{U}_r$, $\mathcal{V}_r$ respectively. Thus, IOW expects to train a group of user feature vectors $\{\mathbf{u}_i\}_{i\in\mathcal{U}_r}$ and a group of item feature vectors $\{\mathbf{v}_j\}_{j\in\mathcal{V}_r}$ to accurately predict the rating (u_i, v_j, r_{ij}) that are not in $\mathcal{D}_r$, where $\mathbf{u}_i, \mathbf{v}_j \in \mathbb{R}^{d\times 1}$, which are both d-dimensional feature column vectors.

In contrast, the social network data held by the SSP is represented as $\mathcal{D}_s = \left\{\left(u_i^{(k)}, u_j^{(k)}\right)\right\}_{k=1}^{n}$ with n elements, where $\left(u_i^{(k)}, u_j^{(k)}\right)$ satisfying $u_i^{(k)} < u_j^{(k)}$ indicates that the user $u_i^{(k)}$ and $u_j^{(k)}$ are friends or know each other. Thus, the social network in SSP can be represented as an undirected graph $\mathcal{G}$. We denote the set of users involved in the social network as $\mathcal{U}_s$, which contains p_s users. The adjacency matrix of $\mathcal{G}$ can be denoted as $\mathbf{S}_\mathcal{G} \in \mathbb{R}^{p_s\times p_s}$, which satisfies

$$\mathbf{S}_\mathcal{G} = \begin{cases} 1 & \text{if } (u_i, u_j) \text{ or } (u_j, u_i) \in \mathcal{D}_s \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

By Exp. (3), we know that $\mathbf{S}_\mathcal{G}$ is symmetric and contains only 0 or 1. Besides, in practical social networks, $\mathbf{S}_\mathcal{G}$ is usually highly sparse. The density of 1 can be expressed as

$$\rho_\mathcal{G} = \frac{n}{C_{p_s}^2} = \frac{\mathbf{1}^T\mathbf{S}_\mathcal{G}\mathbf{1}}{2C_{p_s}^2}, \tag{4}$$

where $C_{p_s}^2 = \frac{1}{2}p_s\,(p_s - 1)$, indicates the all possible social relationship pairs.

The federated cross-domain social recommendation should work on the common users between IOW and SSP. Thus, we denote the common users as the set $\mathcal{U}_c = \mathcal{U}_r \cap \mathcal{U}_s$. As the overall algorithmic workflow shown in Fig. 2, IOW trains a collaborative filtering model and obtains a group of user feature vectors of $\mathcal{U}_r$. And then, it submits the feature vectors $\{\mathbf{u}_i\}_{i\in\mathcal{U}_c}$ of common users to SSP and receives $\{\mathbf{u}'_i\}_{i\in\mathcal{U}_c}$ from SSP after updating by the social network. To describe the update of $\{\mathbf{u}_i\}_{i\in\mathcal{U}_c}$ more effectively, we stacks $\{\mathbf{u}_i\}_{i\in\mathcal{U}_c}$, $\{\mathbf{u}_i\}_{i\in\mathcal{U}_c}$ vertically and denote them as feature matrices $\mathbf{U}_c, \mathbf{U}'_c \in \mathbb{R}^{p_c\times d}$.

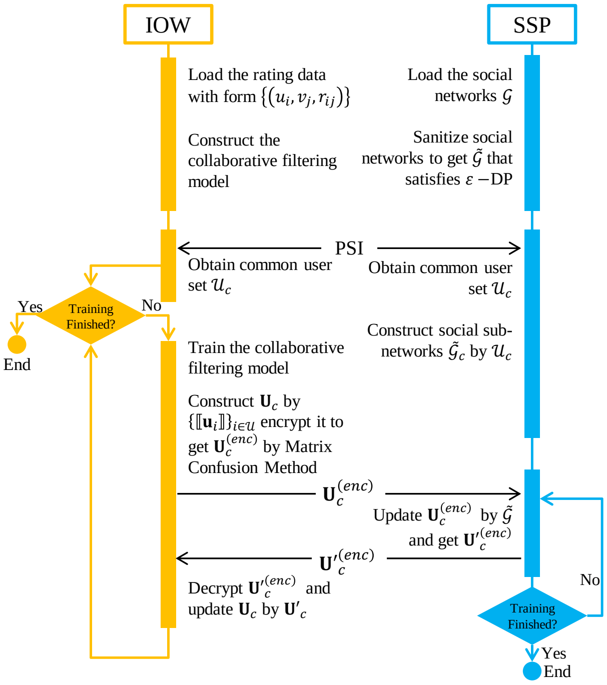

IOW
SSP
Load the rating data with form $\{(u_i, v_j, r_{ij})\}$
Load the social networks $\mathcal{G}$
Construct the collaborative filtering model
Sanitize social networks to get $\tilde{\mathcal{G}}$ that satisfies ε −DP
PSI
Obtain common user set $\mathcal{U}_c$
Obtain common user set $\mathcal{U}_c$
Yes
Training Finished?
No
End
Train the collaborative filtering model
Construct social sub-networks $\tilde{\mathcal{G}}_c$ by $\mathcal{U}_c$
Construct $\mathbf{U}_c$ by $\{[\![\mathbf{u}_i]\!]\}_{i\in\mathcal{U}}$ encrypt it to get $\mathbf{U}_c^{(enc)}$ by Matrix Confusion Method
$\mathbf{U}_c^{(enc)}$
Update $\mathbf{U}_c^{(enc)}$ by $\tilde{\mathcal{G}}$ and get $\mathbf{U'}_c^{(enc)}$
$\mathbf{U'}_c^{(enc)}$
Decrypt $\mathbf{U'}_c^{(enc)}$ and update $\mathbf{U}_c$ by $\mathbf{U'}_c$
No
Training Finished?
Yes
End

Figure 2: The workflow of our algorithm framework

4.2 Learning of Social Networks with Differential Privacy

To learn user feature vectors by social networks, we adopt the semi-supervised learning method on graph proposed by [Wang *et al.*, 2017]. It considers that if two users are strongly connected, they are more likely to have similar feature representations. Based on this consideration, two objective functions are proposed for optimizing $\mathbf{U}'_c$. One is the objective function for smoothness constraint, which is expressed as

$$\mathrm{f}_1\left(\mathbf{U}'_c\right) = \frac{1}{2}\sum_{i,j\in\mathcal{U}_c} s_{ij}\left\|\mathbf{u}'_i/\sqrt{d_i} - \mathbf{u}'_j/\sqrt{d_j}\right\|^2, \tag{5}$$

where s_{ij} is the i-th row and j-column element of $\mathbf{S}_{\mathcal{G}_c}$ and d_i denotes the degree of user u_i. As shown in Fig. 2, $\mathbf{S}_{\mathcal{G}_c}$ is the adjacency matrix of sub-networks $\mathcal{G}_c$ involved the users in $\mathcal{U}_c$, i.e., a sub-matrix of $\mathbf{S}_\mathcal{G}$. Another objective function is to keep the consistency of the user feature representations, which is expressed as

$$\mathrm{f}_2\left(\mathbf{U}'_c\right) = \frac{1}{2}\left\|\mathbf{U}'_c - \mathbf{U}_c\right\|_F^2. \tag{6}$$

Combining the above two objective functions, [Wang *et al.*, 2017] developed the following optimization equation to obtain $\mathbf{U}'$, which is

$$\min_{\mathbf{U}_c'} \quad \mathrm{f}_1\left(\mathbf{U}_c'\right) + \mu \mathrm{f}_2\left(\mathbf{U}_c'\right), \tag{7}$$

where $\mu > 0$ is a parameter to control the tradeoff between two objective functions. Since (7) is a quadratic equation, it has a closed-form solution as

$$\mathbf{U}_c' = \frac{\mu}{(2+\mu)}\left(\mathbf{I} - \frac{2}{(2+\mu)}\mathbf{D}^{-1/2}\,\mathbf{S}_{\mathcal{G}_c}\mathbf{D}^{-1/2}\right)^{-1}\mathbf{U}_c \tag{8}$$

where $\mathbf{D}$ is a diagonal matrix whose diagonal elements are consisted of $d_i, i \in \mathcal{U}_c$. Note that we corrected an error in [Wang *et al.*, 2017] on computing $\mathbf{U}_c'$. Thus, our Exp. (8) is different from Exp. (15) in [Wang *et al.*, 2017].

Since the calculation of $\mathbf{U}_c'$ involves $\mathbf{S}_{\mathcal{G}_c}$, unprotected $\mathbf{S}_{\mathcal{G}}$ will lead to leakages of users' social connections. Considering the characteristics of $\mathbf{S}_{\mathcal{G}}$, we adopt the random response mechanism (RRM) as Def. 2 to provide individual privacy guarantees. The process of obtaining the perturbed $\mathbf{S}_{\widetilde{\mathcal{G}}}$ satisfying ε-DP is shown in Fig. 3.

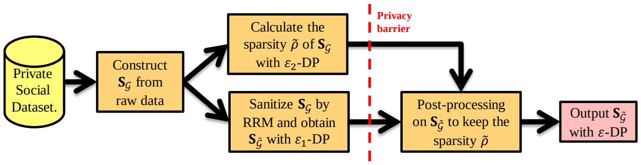

Private Social Dataset.
Construct $\mathbf{S}_{\mathcal{G}}$ from raw data
Calculate the sparsity $\tilde{\rho}$ of $\mathbf{S}_{\mathcal{G}}$ with ε_2-DP
Sanitize $\mathbf{S}_{\mathcal{G}}$ by RRM and obtain $\mathbf{S}_{\widetilde{\mathcal{G}}}$ with ε_1-DP
Privacy barrier
Post-processing on $\mathbf{S}_{\widetilde{\mathcal{G}}}$ to keep the sparsity $\tilde{\rho}$
Output $\mathbf{S}_{\widetilde{\mathcal{G}}}$ with ε-DP

Figure 3: Flowchart to Achieve ε-DP with RRM

In Fig. 3, we divide the perturbation process into two parts, satisfying ε_1-DP and ε_2-DP, respectively, where $\varepsilon_1 + \varepsilon_2 = \varepsilon$ by Thm. 1. The first part satisfies ε_1-DP. Regarding all elements in $\mathbf{S}_{\mathcal{G}}$ above (or below) the main diagonal as a series of independent responses, we directly apply RRM on $\mathbf{S}_{\mathcal{G}}$ to achieve ε_1-DP. The element $\widetilde{s}_{ij}$ of $\mathbf{S}_{\widetilde{\mathcal{G}}}$ can be calculated by

$$\widetilde{s}_{ij} = \begin{cases} s_{ij} & \text{with probability } p \text{ if } i < j \\ 1 - s_{ij} & \text{with probability } 1-p \text{ if } i < j \\ \widetilde{s}_{ji} & \text{otherwise} \end{cases}. \tag{9}$$

where $p = \frac{e^{\varepsilon_1}}{1+e^{\varepsilon_1}}$.

Note that after sanitization, the density of $\mathbf{S}_{\mathcal{G}}$ will be changed. Denoting $\mathbf{S}_{\widetilde{\mathcal{G}}}$ as $\rho_{\widetilde{\mathcal{G}}}$, we have

$$\mathbb{E}\left(\rho_{\widetilde{\mathcal{G}}}\right) = \rho_{\mathcal{G}} p + \left(1 - \rho_{\mathcal{G}}\right)(1-p). \tag{10}$$

Since $\mathbf{S}_{\mathcal{G}}$ is highly sparse, even if p is close to 1, $\mathbb{E}\left(\rho_{\widetilde{\mathcal{G}}}\right)$ still changes dramatically. For example, $\rho_{\mathcal{G}}$ of FilmTrust is 0.34%. When $\varepsilon_1 = 5$, $p = 0.9933$ is close to 1 (only a tiny part of elements in $\mathbf{S}_{\widetilde{\mathcal{G}}}$ return fake responses). We calculate that $\mathbb{E}\left(\rho_{\widetilde{\mathcal{G}}}\right) = 1.01\%$, which is nearly twice more than the original $\rho_{\mathcal{G}}$. To keep the original density, we introduce the second perturbation process that satisfies ε_2-DP. It computes a noisy connection count $\widetilde{n} = |\mathcal{D}_s| + Lap\left(1/\varepsilon_2\right)$ by Laplace mechanism and then computes $\widetilde{\rho}_{\mathcal{G}}$ satisfying ε_2-DP by $\widetilde{\rho}_{\mathcal{G}} = \widetilde{n}/C_{p_s}^2$. Using $\widetilde{\rho}_{\mathcal{G}}$, we can keep $\rho_{\widetilde{\mathcal{G}}}$ reach $\widetilde{\rho}_{\mathcal{G}}$ by randomly adding or removing 1s in $\mathbf{S}_{\widetilde{\mathcal{G}}}$. It is post-processing and does not lead to any more privacy leakage [Dwork *et al.*, 2014]. Usually, we can allocate most of the privacy budget to ε_1, e.g., 99% of ε, because even if ε_2 is allocated a extremely small privacy budget, the deviation of $\rho_{\widetilde{\mathcal{G}}}$ is still not significant. For instance, $\varepsilon = 1$ and ε_1 is 50% of ε, we have $p = 0.62$ and $\rho_{\widetilde{\mathcal{G}}}$ of the dataset FilmTrust satisfying $(3.4 \pm 0.0074) \times 10^{-3}$. However, if ε_1 is changed to 99% of ε, we have $p = 0.73$ and $\rho_{\widetilde{\mathcal{G}}}$ satisfying $(3.4 \pm 0.37) \times 10^{-3}$, which shows that increasing the ratio of ε_1 can improve p up to 0.11 but the deviation of $\rho_{\widetilde{\mathcal{G}}}$ still remains in a small range.

4.3 Highly efficient $\mathbf{S}_{\widetilde{\mathcal{G}}}$ construction and calculation for $\mathbf{U}_c'$

Since there is only a tiny part of elements in $\mathbf{S}_{\widetilde{\mathcal{G}}}$ return fake responses in general, it is unnecessary to traverse all possible connections and check whether they return a fake response. To construct $\mathbf{S}_{\widetilde{\mathcal{G}}}$ more efficiently, we first generate a random integer N_{fake} satisfying binomial distribution $\mathrm{B}\left(C_{p_s}^2, 1-p\right)$ to denote the number of fake responses. And then, we randomly choose a group of $\left\{s_{ij}^{(t)}\right\}_{t=1}^{N_{fake}}, i < j$ to change their and the symmetric elements' values. Since the generation of N_{fake} is $O(1)$, the fake responses can be achieved by a random sampling algorithm [Meng, 2013]. Thus, the overall computational complexity of generating $\mathbf{S}_{\widetilde{\mathcal{G}}}$ is $O\left(\left(1 - p + \rho_{\mathcal{G}}\right) p_s^2\right)$, which is usually much lower than $O\left(p_s^2\right)$ of traversal.

In addition, Exp. (8) involves the process of solving an equation. Specifically, denoting

$$\mathbf{Q} = \mathbf{I} - \frac{2}{(2+\mu)}\mathbf{D}^{-1/2}\,\mathbf{S}_{\mathcal{G}}\mathbf{D}^{-1/2}, \tag{11}$$

we can describe the calculation of $\mathbf{U}_c'$ as solving the following equation for $\mathbf{U}_c'$,

$$\mathbf{Q}\mathbf{U}_c' = \frac{\mu}{(2+\mu)}\mathbf{U}_c. \tag{12}$$

Solving equations is an algebraic calculation with high computational overhead. Even though $\mathbf{S}_{\widetilde{\mathcal{G}}}$ is usually highly sparse, the traditional linear solution methods for sparse matrices cannot support large-scale social networks. Facing the challenge of solving large-scale sparse equations, we adopt LU decomposition to improve the computational efficiency of $\mathbf{U}_c'$. Since $\mathbf{Q}$ is symmetric, we can express it as the following LU decomposition form

$$\mathbf{Q} = \mathbf{P}\mathbf{L}\mathbf{L}^T\mathbf{P}^T, \tag{13}$$

where $\mathbf{L}$ is a lower triangular matrix; $\mathbf{P}$ is a permutation matrix. Since the multiplication with the permutation matrix $\mathbf{P}$ is equivalent to rearranging the elements of the matrix, the calculation of U can be equivalently converted into solving the equations about the triangular matrix twice after LU decomposition, i.e.,

$$\mathbf{L}\mathbf{U}_c^{(2)} = \mathbf{U}_c^{(1)} \text{ and } \mathbf{L}^T\mathbf{U}_c^{(3)} = \mathbf{U}_c^{(2)}. \tag{14}$$

where $\mathbf{U}_c^{(1)} = \frac{\mu}{(2+\mu)}\mathbf{P}^T\mathbf{U}_c$, $\mathbf{U}_c^{(3)} = \mathbf{P}^T\mathbf{U}_c'$. Compared to the regular sparse equations, solving equations on triangular matrices is far more efficient. Its computational overhead depends on the density of $\mathbf{L}$. That is, fewer non-zero elements imply a smaller computational overhead.

Our study shows that different permutation strategies during LU decomposition will result in different density of $\mathbf{L}$.

Table 1: The Number of Non-zero Elements of $\mathbf{L}$ under Different Permutation Strategies for the Three Datasets

	FilmTrust	**CiaoDVD**	**Epinions**
PS.1	$42,672$	$4,677,004$	Failed
PS.2	$5,236$	$564,080$	$43,108,346$
PS.3	$\mathbf{3,115}$	$\mathbf{232,645}$	$\mathbf{16,030,689}$
PS.4	$5,070$	$839,656$	$142,441,761$

Here, we compared four different strategies, which are natural ordering (PS.1), minimum degree ordering on the structure of $\mathbf{Q}^T\mathbf{Q}$ (PS.2), minimum degree ordering on the structure of $\mathbf{Q}^T + \mathbf{Q}$ (PS.3), as well as approximate minimum degree column ordering (PS.4). As shown in Tab. 1, we test the density of $\mathbf{L}$ after LU decomposition with different strategies (PS.1 to PS.4) for the three classical datasets FilmTrust, CiaoDVD, and Epinions. The results show that $\mathbf{L}$ always has the least number of non-zero elements with PS.3, which means it is the most optimal LU decomposition strategy for social networks and can solve $\mathbf{U}'_c$ with the highest efficiency.

4.4 Privacy-Preserving User Feature Vector Update with Matrix Confusion Method

To avoid privacy leakage by directly uploading user features, IOW needs to encrypt them while maintaining computability before uploading. Completing the update efficiently of $\mathbf{U}_c$ without any privacy available to SSP is a critical challenge faced by federated cross-domain social recommendations. We propose Matrix Confusion Method according to the update of $\mathbf{U}_c$, which enables the SSP to compute $\mathbf{U}'_c$ despite the inability to identify the received $\mathbf{U}_c$.

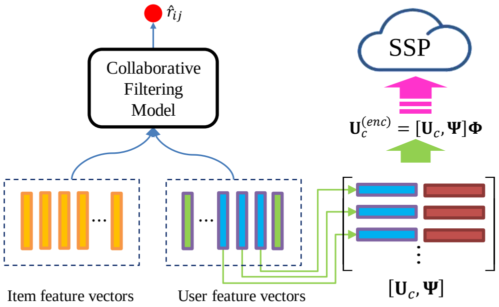

$\hat{r}_{ij}$
Collaborative Filtering Model
SSP
$\mathbf{U}_c^{(enc)} = [\mathbf{U}_c, \mathbf{\Psi}]\mathbf{\Phi}$
...
...
Item feature vectors
User feature vectors
$[\mathbf{U}_c, \mathbf{\Psi}]$

Figure 4: Matrix Confusion before Uploading

As shown in Fig. 2, IOW encrypts $\mathbf{U}_c$ and obtains $\mathbf{U}_c^{(enc)}$, and then replaces $\mathbf{U}_c$ by $\mathbf{U}_c^{(enc)}$ to upload to SSP. As shown is Fig. 4, in our Matrix Confusion Method, $\mathbf{U}_c^{(enc)}$ is constructed as the following step:

1) IOW randomly generates a matrix $\mathbf{\Psi} \in \mathbb{R}^{p_c \times d}$ (the crimson vectors in Fig. 4) of the same shape as $\mathbf{U}_c$ and an invertible random matrix $\mathbf{\Phi} \in \mathbb{R}^{2d \times 2d}$.

2) IOW stacks $\mathbf{U}_c$ and $\mathbf{\Psi}$ horizontally and then multiplies $\mathbf{\Phi}$ right to obtain $\mathbf{U}_c^{(enc)}$.

3) IOW uploads $\mathbf{U}_c^{(enc)}$ to SSP.

Compared to the inefficient calculation on homomorphic encryption, our method only requires twice the calculation of unencrypted $\mathbf{U}_c$ to achieve the calculation on encrypted data. After SSP finishes calculating $\mathbf{U}_c'^{(enc)}$, IOW can restore $\mathbf{U}'_c$ by the following step:

1) IOW multiply $\mathbf{U}_c'^{(enc)}$ right by $\mathbf{\Phi}^{-1}$ to get $[\mathbf{U}'_c, \mathbf{\Psi}']$.

2) IOW retain the first d columns of $[\mathbf{U}'_c, \mathbf{\Psi}']$ to obtain $\mathbf{U}'_c$.

5 Security Analysis

The security consists of two aspects, i.e., the security of user features for IOW and the social network's security for SSP.

In our scheme, our Matrix Confusion Method avoids $\mathbf{U}_c$ from being identified while updating. Since IOW always keeps the random matrices $\mathbf{\Psi}$ and $\mathbf{\Phi}$ locally during the federated learning, SSP cannot infer the true value of $\mathbf{U}_c$ from the received $\mathbf{U}_c^{(enc)}$. Furthermore, our Matrix Confusion Method provides a non-analyzable guarantee for $\mathbf{U}_c$. Due to the randomness of $\mathbf{\Psi}$, stacking it horizontally with $\mathbf{U}_c$ and engaging in confusion destroys the properties originally held by $\mathbf{U}_c$. Attackers cannot analyze $\mathbf{U}_c^{(enc)}$ effectively, which ensures the security of the uploaded $\mathbf{U}_c$.

The security of social networks comes from the privacy guarantees provided by DP. As a proven DP method in [Mironov, 2017], RRM provides sufficient security. Even if an attacker maliciously accesses $\widetilde{\mathcal{G}}$, he still cannot effectively infer individual privacy. Notice that our proposed DP approach provides static sanitized social networks for FL. Even if IOW requests updates for $\mathbf{U}_c$ multiple times, the privacy budget will not be consumed additionally. It prevents privacy leakage caused by IOW maliciously submitting multiple update requests.

6 Experiments

6.1 Experimental Setting

Table 2: The Number of Non-zero Elements of $\mathbf{L}$ under Different Permutation Strategies for the Three Datasets

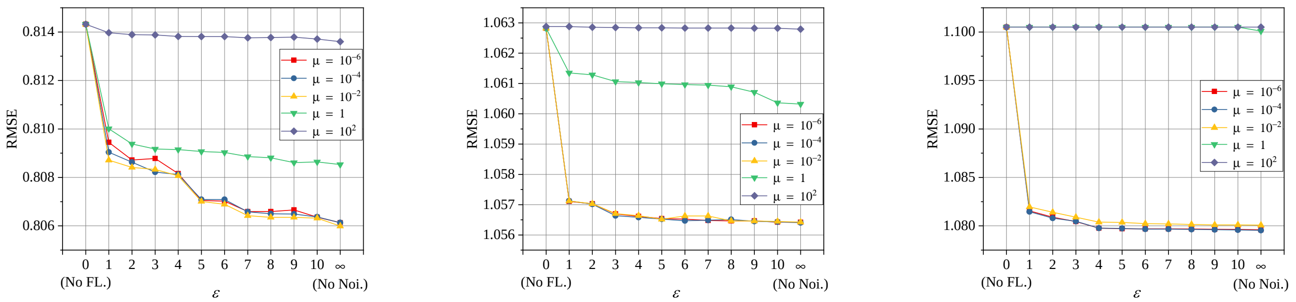
	FilmTrust	**CiaoDVD**	**Epinions**
p_r	$1,508$	$17,615$	$40,163$
p_s	874	$4,658$	$49,287$
p_c	740	$2,740$	$40,162$
v_r	$2,071$	$16,121$	$139,738$
m	$35,497$	$72,665$	$664,823$
n	$1,309$	$33,116$	$381,035$

Notes. "p_r": the rated user number in IOW; "p_s": the social user number in SSP; "p_c": the common user number; "v_r": the item number in IOW; "m": the rating number in IOW; "n": the number of social connection pairs.

In this section, we evaluate the effectiveness of our proposed FCSR by experiments. The experiments run on a computer with Dual 4 Core 3.9GHz AMD Ryzen CPU, 32GB RAM and NVIDIA GeForce RTX 2080 Ti GPU. To fully evaluate our scheme, we experiment with three classic social recommendation datasets with different data scales, which are small-scale FilemTrust[1], medium-scale CiaoDVD[1], and large-scale Epinions[2]. Among them, Filmtrust comes from

[1] https://guoguibing.github.io/librec/datasets.html

[2] https://www.cse.msu.edu/~tangjili/datasetcode/truststudy.htm



(a) FilmTrust

(b) CiaoDVD

(c) Epinions

Figure 5: RMSE for The Three Experimental Datasets

online film rating website which contains the social connections between users; The user ratings of CiaoDVD and Epinions are crawled from shopping websites and the social connections are constructed from the users' trust links. The details are shown in Tab. 2. We adopt NCF proposed by [He *et al.*, 2017] as the recommended model of our FCSR and ε_1 and ε_2 are taken as 99% and 1% of ε. To test the generalizability of the training models, we randomly divide the rating data into a training set $\mathcal{D}_r^{(train)}$, and a test set $\mathcal{D}_r^{(test)}$ with a proportion of $9:1$. Then, we perform multiple experiments on the same experimental parameters and take their average experimental results as our final experimental results.

6.2 Evaluation on Training Effect

We use RMSE on $\mathcal{D}_r^{(test)}$ as the evaluation metric to measure the learning effect of FCSR, which is calculated by

$$RMSE = \sqrt{\frac{1}{\left|\mathcal{D}_r^{(test)}\right|} \sum_{k \in \mathcal{D}_r^{(test)}} \left(r_{ij}^{(k)} - \widehat{r}_{ij}^{(k)}\right)^2}. \tag{15}$$

The results in Fig. 5(a-c) test the impact of various privacy budgets ε and the social learning parameters μ on RMSE. In Fig. 5(a-c), we use the setting $\varepsilon = 0$ to indicate IOW trains the recommendation model independently without interacting with the SSP (non-FL scenario, i.e., No FL.), and $\varepsilon = \infty$ to indicate Federated learning without DP (i.e., SSP directly uses the original social networks, i.e., No Noi.). Fig. 5(a-c) show that the cross-domain federation learning based on social networks effectively reduce the RMSE on $\mathcal{D}_r^{(test)}$. As ε increases, RMSE tends to increase overall, which means more accurate social networks will lead to more accurate predictions.

In addition, according to optimization (7), a smaller μ implies a higher weight of the smoothing constraint. In other words, the more significant the role of social network learning. Our experimental results show that smoothing constraints with higher weights (smaller μ) can reduce the RMSE more, indicating that enhancing the role of social networks in FL can effectively improve the learning effect.

6.3 Comparison with Existing Federated Schemes

We further compare our FCSR with the existing Federated recommendation, which are FedMF [Chai *et al.*, 2020], FedGNN [Wu *et al.*, 2021], and FeSoG [Liu *et al.*, 2022]. Similarly, we use RMSE as the evaluation metric. The experimental results are shown in Fig. 6. Since FCSR introduces DP to protect individual privacy, we set $\varepsilon = 1$ facing the optimal case of μ in Fig. 5 to carry out the comparisons in Fig. 6. The results show that our proposed FCSR significantly improves the accuracy of existing federated schemes on all three datasets. The main reasons are as follows. First, we only apply DP on social networks instead of applying DP on both user feature vectors and social networks, as in [Liu *et al.*, 2022]. It guarantees the introduction of FL will not produce worse results than recommendations without social networks. Second, FCSR requires only one perturbation to achieve ε-DP, which does not accumulate noise compared to the common gradient-based perturbations. In summary, FCSR is effective and practical.

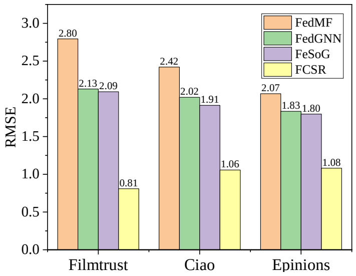


Figure 6: RMSE for Dataset Epinions

7 Conclusions and Future Works

Our proposed FCSR effectively supports the participants to collaboratively train better recommendation models with the private data maintained locally. Through security analysis, we ensure security during the FL process. Meanwhile, our experiments show that FCSR is an effective and practical algorithm. Currently, federated cross-domain social recommendations are a promising direction with substantial potential opportunities. In our future work, we will concentrate on researching more effective federated cross-domain recommendation algorithms and overcoming challenges from security and efficiency.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (project numbers 62072109 and U1804263). Ximeng Liu is the contact author.

References

Dan Bogdanov, Sven Laur, and Jan Willemson. Sharemind: A framework for fast privacy-preserving computations. In *European Symposium on Research in Computer Security*, pages 192–206. Springer, 2008.

Di Chai, Leye Wang, Kai Chen, and Qiang Yang. Secure federated matrix factorization. *IEEE Intelligent Systems*, PP:1–1, 08 2020.

Jinming Cui, Chaochao Chen, Lingjuan Lyu, Carl Yang, and Wang Li. Exploiting data sparsity in secure cross-platform social recommendation. *Advances in Neural Information Processing Systems*, 34:10524–10534, 2021.

Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 486–503. Springer, 2006.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407, 2014.

Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. Graph neural networks for social recommendation. In *The world wide web conference*, pages 417–426, 2019.

Guibing Guo, Jie Zhang, and Neil Yorke-Smith. Trustsvd: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29, 2015.

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.

Yang Liu, Chen Liang, Xiangnan He, Jiaying Peng, Zibin Zheng, and Jie Tang. Modelling high-order social relations for item recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 2020.

Zhiwei Liu, Liangwei Yang, Ziwei Fan, Hao Peng, and Philip Yu. Federated social recommendation with graph neural network. *ACM Transactions on Intelligent Systems and Technology*, 02 2022.

Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. pages 94–103, 11 2007.

Frank D McSherry. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, pages 19–30, 2009.

Xiangrui Meng. Scalable simple random sampling and stratified sampling. In *International Conference on Machine Learning*, pages 531–539. PMLR, 2013.

Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th computer security foundations symposium (CSF)*, pages 263–275. IEEE, 2017.

Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *International conference on the theory and applications of cryptographic techniques*, pages 223–238. Springer, 1999.

Jean-Fransico Raymond and Anton Stiglic. Security issues in the diffie-hellman key agreement protocol. *IEEE Transactions on Information Theory*, 22:1–17, 2000.

Xiang Wang, Xiangnan He, Liqiang Nie, and Tat-Seng Chua. Item silk road: Recommending items from information domains to social users. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 185–194, 2017.

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.

Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology*, 10, 2019.

Liu Yang, Ben Tan, Vincent Zheng, Kai Chen, and Qiang Yang. *Federated Recommendation Systems*, pages 225–239. 11 2020.