
Efficient Knowledge Transfer in Federated Recommendation for Joint Venture Ecosystem

Yichen Li¹, Yijing Shan¹, Yi Liu², Haozhao Wang¹, Cheng Wang¹,
Wei Wang², Yi Wang², Ruixuan Li^{1*}

¹School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China ²Chongqing Ant Consumer Finance Co., Ltd, Ant Group, Chongqing, China
{ycli0204, M202474082, hz_wang}@hust.edu.cn

Abstract

The current Federated Recommendation System (FedRS) focuses on personalized recommendation services and assumes clients are personalized IoT devices (e.g., mobile phones). In this paper, we deeply dive into new but practical FedRS applications within the joint venture ecosystem. Subsidiaries engage as participants with their users and items. However, in such a situation, merely exchanging item embedding is insufficient, as user bases always exhibit both overlaps and exclusive segments, demonstrating the complexity of user information. Meanwhile, directly uploading user information is a violation of privacy and unacceptable. To tackle the above challenges, we propose an efficient and privacy-enhanced Federated Recommendation for the Joint Venture Ecosystem (FR-JVE) that each client transfers more common knowledge from other clients with a distilled user's *rating preference* from the local dataset. More specifically, we first transform the local data into a new format and apply model inversion techniques to distill the rating preference with frozen user gradients before the federated training. Then, a bridge function is employed on each client side to align the local rating preference and aggregated global preference in a privacy-friendly manner. Finally, each client matches similar users to make a better prediction for overlapped users. From a theoretical perspective, we analyze how effectively FR-JVE can guarantee user privacy. Empirically, we show that FR-JVE achieves superior performance compared to state-of-the-art methods.

1 Introduction

Recommendation systems have emerged as crucial tools and products, significantly impacting daily lives by offering tailored suggestions of new items that may interest users. These systems fundamentally rely on centralized servers to consolidate user data, digital behaviors, and preferences, thereby training models for precise recommendation generation [9, 39, 8]. However, transmitting local user information to central servers poses considerable privacy and security concerns. Moreover, recent stringent government regulations on privacy protection underscore the necessity of storing user data locally on devices rather than uploading it to a global server. As a potential remedy to this dilemma, federated learning (FL) emerges as a promising approach, enforcing data localization and enabling the distributed training of a globally shared model [18, 12, 37, 13]. This framework has achieved remarkable success and has been applied to various fields, such as recommendation systems [38, 27, 14] and smart healthcare [6, 34].

In recent years, researchers have studied federated recommendation systems (FedRS), which enable different clients to yield optimal recommendations without breaching user privacy. FCF proposed

*Ruixuan Li is the corresponding author.

in [1] is the first FL-based collaborative filtering method, which employs the stochastic gradient approach to update the local model, and FedAvg is adopted to update the global model. The authors in [4] propose to adapt distributed matrix factorization to the FL setting and introduce the homomorphic encryption technique on gradients before uploading to the server. FedNCF [10] adapts neural collaborative filtering to the federated setting which introduces neural network to learn user-item interaction function to enhance model learning ability. To better protect user privacy, the authors in [27] offer a bipartite personalization mechanism for personalized recommendations via keeping the user embedding local. Based on it, FedRAP [21] further emphasizes the differences between user clients with additive personalization techniques.

While these methods achieve great success in user-level federated recommendation in which the user engages as the participant in the federated training, we focus on another more challenging and realistic scenario of federated recommendation, specifically within a joint venture ecosystem, where each participant is no longer an individual user but rather a subsidiary or business group. These subsidiaries, each possessing their own users and items, are managed under a unified controlling corporation. For instance, Alibaba [23], as the parent company, oversees various businesses such as Alipay [26], Tmall Shopping [11], and Digital Media and Entertainment Group [31]. These business groups need to collaborate while also protecting their respective user privacy information due to potential conflicts of interest. Each group will engage in federated training as a participant, sharing partially overlapping user and item information (e.g., credits, red envelopes, discounts) while maintaining their own exclusive user bases and item offerings. The intuitive explanation can also be found in a simple 3-client example illustrated in Figure 1.

Compared to traditional federated recommendation scenarios, our ecosystem poses additional challenges. Firstly, there is a significantly larger volume of data involved, as the participants are no longer individual users but rather business groups. Secondly, partial overlap exists between users and items, necessitating selective knowledge sharing among participants to enhance recommendation performance for both shared and unique users. Lastly, there is a trade-off between the privacy of user information and the performance of the recommendation model. Some existing methods upload user embedding to improve model performance, inadvertently leaking private information [10, 4]. Others exclusively utilize item embedding while keeping user embedding locally private [27, 21]. While this approach can yield satisfactory results in traditional federated recommendation settings, in our scenario, where a large data volume is involved, users exhibit partial overlap, resulting in intricate and overlapping user information. Relying solely on the sharing of item embedding falls short of achieving the expected performance.

To address these challenges, we in this paper investigate both a privacy-friendly and efficient method for the joint venture ecosystem. Inspired by [28], most cross-domain recommendation research transfers the common knowledge in the latent space between source and target domains to enhance recommendation performance. To facilitate this transfer, bridge functions are trained to map the embedding of the user and item from the source domain onto the target domain. In our ecosystem, we consider that the local data in each client constitutes a specific domain and study how to more effectively and securely transfer user and item information between clients.

To explore this idea, we first follow the hypothesis of sharing item embedding across clients in a FedAvg manner. While directly sharing user embedding with other clients is unacceptable due to the privacy leakage, we attempt to apply the differential privacy technique [35] to the user embedding to

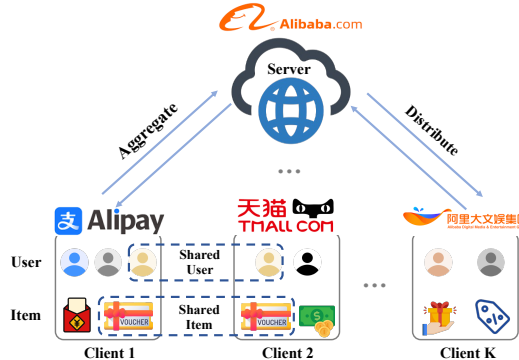


Figure 1: An example of 3-client in the joint venture ecosystem. Each participant in the study represents a distinct business group, encompassing a subset of users and items. Notably, while Client 1 and Client 2 share a partial overlap in their user and item sets, they each maintain a unique set of personalized users and items. The primary objective of FedRS is to leverage collaborative training to develop a comprehensive global model, especially enhancing prediction accuracy specifically for those users who are common across the participating clients.

protect user privacy and employ a bridge function for each client to identify similar users between clients. However, the common understanding of differential privacy is that as the privacy budget decreases, the effectiveness of the model decreases, whereas increasing the budget results in increased privacy risks. Then, we propose an efficient and privacy-enhanced federated recommendation framework for the joint venture ecosystem dubbed FR-JVE that constructs the rating preference for each client as the filtering signal to transfer the common knowledge from other clients. Specifically, before the federated training, each client will transform the local dataset into a new data format. Based on it, a secure user’s rating preference will be constructed locally via the model inversion technique with frozen user gradients, which will not disclose user privacy. Then, the fixed rating preference will be uploaded, and the central server will transmit the aggregated rating preferences of other clients to each client. During the training process, each client only needs to update and communicate with the item embedding. A bridge function is trained locally to map the downloaded aggregated rating preference to the local rating preference. At the inference stage, each client searches the top-k users well-matched on the rating preference mapped by the bridge function for recommendation.

Through extensive experiments on various datasets with rating tasks, we show that FR-JVE significantly improves the recommendation performance compared to state-of-the-art approaches. The major contributions of this paper are summarized as follows:

- We study the problem of federated recommendation for joint venture ecosystem, in which the main challenges are the massive data volume and complex user/item distribution. Different from traditional user-level federated recommendation, the subsidiary or business group with its own users and items participates in the federated training. The balance between performance and privacy does matter here.
- Then, to address this problem, we propose a novel framework named FR-JVE which can be seen as an off-the-shelf personalization add-on for standard item embedding transmission-based federated recommendation system and it involves the user information with a secure rating preference in a privacy-friendly manner to improve the performance.
- Finally, we theoretically show that FR-JVE can efficiently guarantee user privacy and empirically conduct extensive experiments on various datasets. Experimental results illustrate that our proposed model outperforms the state-of-the-art methods on rating tasks.

2 Related Work

Federated Learning Federated Learning (FL) aims to train a global model collaboratively by aggregating locally trained models from multiple clients, each utilizing its own private dataset [17, 16, 15, 19]. FedAvg [29] is a widely known FL framework that enhances the global model by combining parameters from locally trained models. Methods like [25] employ knowledge distillation with unlabeled samples as a proxy dataset. Recently, data-free knowledge distillation techniques leveraging adversarial methods for data generation have garnered attention [45, 43, 36]. In this manuscript, the FL framework has been adopted to the recommendation system to provide privacy protection for user information and recommendation services.

Federated Recommendation System Federated recommendation systems have garnered significant interest lately, fueled by heightened privacy concerns [40, 24, 22]. Recent efforts have primarily concentrated on leveraging the interaction matrix, the cornerstone of basic recommendation scenarios [42, 2]. FCF [1], a pioneering collaborative filtering method under federated learning (FL), utilizes stochastic gradients to update local models and FedAvg for global model aggregation. To protect user privacy, FedMF [4] integrates distributed matrix factorization into the FL framework, encrypting gradients before transmission to the server. Additionally, federated recommendation methods utilizing diverse data sources have emerged, accounting for multiple information streams. FedFast [33] enhances FedAvg with an active aggregation strategy to accelerate convergence, while Efficient-FedRec partitions the model into a server-side news model and a client-side user model, minimizing computational and communication overheads. Both approaches go beyond the interaction matrix, incorporating user features and new attributes. Meanwhile, PFedRec[27] offers a bipartite personalization mechanism for personalized recommendations. Existing research usually focuses on the user-level federated recommendation, where each user is involved as a participant. FedCORE [20] proposes a cross-organization federated recommendation framework for cold-start users. In this

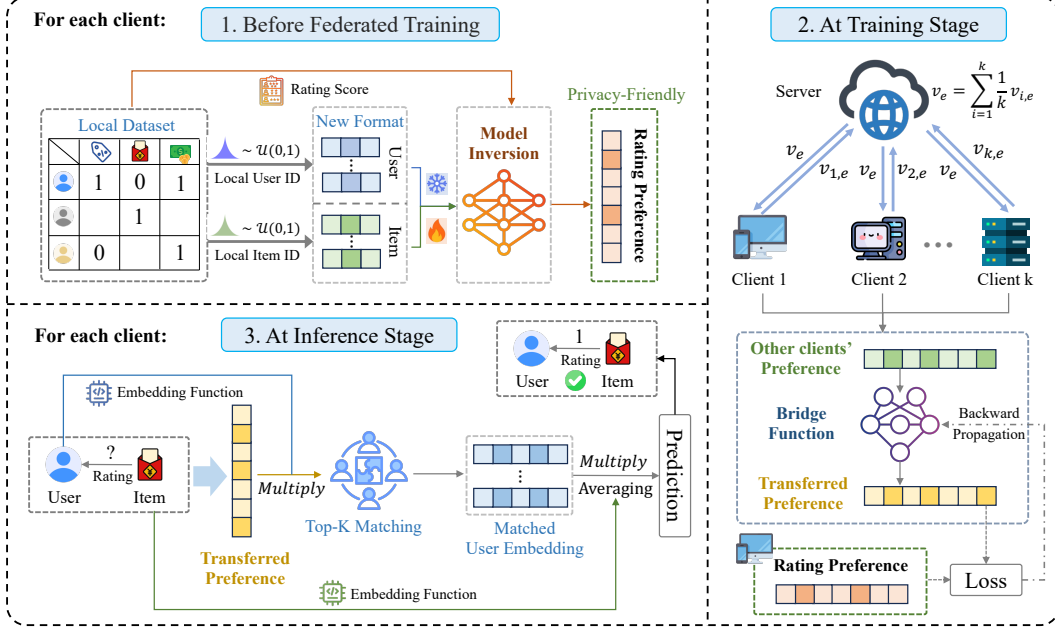


Figure 2: The framework of FR-JVE. Before the federated training, each client first locally transforms the local data into a new data format. Then, the model inversion is applied to distill the privacy-friendly rating preference for each user with the frozen gradient of user information. At the training stage, each client communicates only with the item embedding in a FedAvg manner. A bridge function will be trained to align the local rating preference and other clients' rating preferences. At the inference stage, each client matches similar users to obtain a better prediction.

paper, we study a novel federated recommendation training framework for a joint venture ecosystem and better enhance user privacy while improving recommendations.

3 Problem Formulation

Federated Recommendation. We aim to collaboratively train a global model for K total clients in FedRS. We consider clients to have a partially shared user set and item set. Each client k can only access to his local private dataset $D_k = \{U_k, V_k, R_k\}$, where $U_k = \{u_k^1, u_k^2, \dots, u_k^i\}$ has i users, $V_k = \{v_k^1, v_k^2, \dots, v_k^j\}$ has j items and $R_k = \{r_{11}, r_{12}, \dots, r_{ij}\}$, $r_{ij} \in \mathbf{R}$ denotes the interaction of user u_i to the item v_j . When original local data $\{U_k, V_k, R_k\}$ enters the forward process, matrix factorization will transform it into the representation $(u_k, v_k)\mathbf{E}_k = (u_{k,e}, v_{k,e})$. The global dataset is considered as the composition of all local datasets $D = \{D_1, D_2, \dots, D_K\} = \sum_{k=1}^K D_k$. The objective of the FedRS is to learn a global embedding model $\mathbf{E}$ that minimizes the total empirical loss over the entire dataset D :

$$\min_{\mathbf{E}} \mathcal{L}(\mathbf{E}) := \sum_{k=1}^K \frac{|D_k|}{|D|} \mathcal{L}(\mathbf{E}_k). \quad (1)$$

$$\mathcal{L}(\mathbf{E}_k) = \begin{cases} \sum_{(i,j) \in R_k} \frac{1}{D_k} (r_{ij} - \hat{r}_{ij})^2, & \hat{r}_{ij} = [u_{k,e}^i]^T v_{k,e}^j. \quad \textcircled{1} \\ \sum_{(i,j) \in R_k} \frac{1}{D_k} - (r_{ij} \log \hat{r}_{ij} + (1 - r_{ij}) \log(1 - \hat{r}_{ij})). & \textcircled{2} \end{cases} \quad (2)$$

where $\mathcal{L}(\mathbf{E}_k)$ is the loss in the k -th client. Here we define two loss functions for ① rating prediction and ② top-k recommendation.

Cross-Domain Recommendation. Cross-Domain Recommendation (CDR) focuses on transferring preference knowledge from an auxiliary source domain to a target domain, effectively addressing

the data sparsity issue inherent in conventional recommendation systems. Given the necessity of knowledge transfer in CDR, recent research has predominantly emphasized user embedding and mapping relationships, leveraging latent space bridge functions to transfer user preferences. Suppose that $\{u_{s,e}, v_{s,e}\}$ denotes the user/item embedding of the source domain and $\{u_{t,e}, v_{t,e}\}$ denotes the user embedding of the target domain. Embedding bridge functions $\{\phi_u, \phi_v\}$ are tasked with mapping the user/item embedding from the source domain to the target domain. The learning process is formalized as a supervised regression problem that minimizes the following loss:

$$\min_{\phi_u} \sum_{\bar{u}} \mathcal{L}_{map}(\phi(u_{s,e}), u_{t,e}), \min_{\phi_v} \sum_{\bar{v}} \mathcal{L}_{map}(\phi(v_{s,e}), v_{t,e}). \quad (3)$$

where the $\bar{u}$ denotes the shared users and $\bar{v}$ denotes the shared items, the bridge function is defined with a linear layer, and the loss function $\mathcal{L}_{map}$ with the mean square error loss. During the local inference stage, the final rate is predicted for the user u_i and item v_j by averaging the ratings:

$$\hat{r}_{ij} = \sum_{n=1}^{\frac{|\bar{u}|}{|\bar{v}|}} \frac{1}{|\bar{u}|} [\phi_u(u_{s,e}^i)]^T \phi_v(v_{s,e}^j). \quad (4)$$

where $\hat{r}_{ij}$ denotes the predicted rating for the user $u_{t,e}^i$ and item $v_{t,e}^j$.

4 Federated Recommendation for the Joint Venture Ecosystem

In this section, we first analyze the knowledge transfer techniques used in the centralized cross-domain recommendation and examine their application in our ecosystem with partially overlapped user bases and item offerings. We then discuss an important issue: directly uploading the user embedding will breach user privacy. We adopt differential privacy to user embedding, ensuring privacy. Unfortunately, it inevitably results in an unbalance between performance and privacy. Motivated by these findings, we finally present our both efficient and privacy-enhanced FR-JVE method.

4.1 Exploration of Cross-Domain Recommendation for Our Ecosystem

In traditional user-based federated recommendation systems, each user participates in federated training to obtain better recommendation results tailored to themselves. In such scenarios, user information is relatively scarce, and promising results can be achieved solely through interactions based on item information. However, in our joint venture ecosystem, where participants act as subsidiaries, there exist vast numbers of users and items. It necessitates federated training to recommend suitable products to all users, where user information plays a pivotal role.

Inspired by [28], the cross-domain recommendation has achieved great success in transferring knowledge from other domains to enhance recommendation performance. Here we try to adopt this technique into the user-based FedRS. Since the client can communicate with the item embedding without breaching user privacy, we just employ a single bridge function to map the user embedding. Suppose that the client k implements an embedding bridge function ϕ_k which maps the local user embedding with user uid. The learning process is formalized as a supervised regression problem that aims to minimize the following loss:

$$\min_{\phi_k} \sum_{u_k^i \in \bar{U}} \mathcal{L}_{map}(\phi_k(u_{k,t}), u_{k,e}), \quad (5)$$

$$\text{where } u_{k,t} = \sum_{i=1}^{K/\{k\}} \frac{|D_i|}{|D|/|D_k|} \cdot u_{k,e}.$$

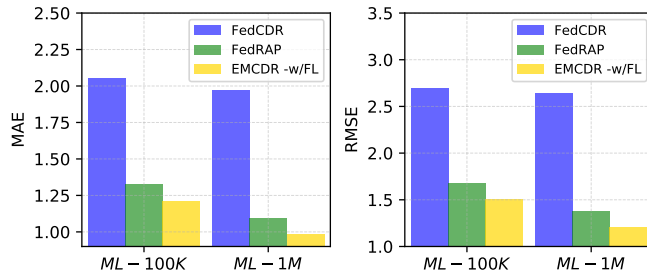


Figure 3: Performance of traditional CDR technique for the joint venture ecosystem.

Table 1: Evaluation of differential privacy with the user embedding on two recommendation datasets. Here we underline the budget used in subsequent experiments.

Dataset	Method	Metric	Privacy Budget ϵ				
			$\epsilon = 0.08$	$\epsilon = 0.1$	$\epsilon = 0.15$	<u>$\epsilon = 0.2$</u>	$\epsilon \geq 1.0$
ML-100K	FedMF	MAE	5.9516	5.4994	4.9770	4.7998	3.6019
		RMSE	12.977	10.6839	8.1233	7.1237	4.3780
	EMCDR -w/FL	MAE	1.8382	1.5983	1.3591	1.2693	1.2123
		RMSE	3.5265	3.1064	2.3998	2.0796	1.5072
ML-1M	FedMF	MAE	$\epsilon = 0.02$	$\epsilon = 0.03$	$\epsilon = 0.04$	<u>$\epsilon = 0.05$</u>	$\epsilon \geq 0.1$
		RMSE	5.2194	4.8106	4.5018	4.3850	3.6809
	EMCDR -w/FL	MAE	13.6630	10.0847	7.8181	6.9106	4.3610
		RMSE	1.4919	1.2322	1.1329	1.0869	0.9824
			4.2600	3.2920	2.5695	2.1763	1.2028

where the $u_{k,t}$ denotes the global user embedding calculated by weighing other user embedding in a FedAvg manner and $\bar{U}$ denotes the shared user. For simplicity, we define the bridge function with a linear layer and the loss function $\mathcal{L}_{map}$ with the mean square error loss. During the local inference stage, each client makes the final rate prediction for the user u_k^i and item v_k^j by averaging the ratings by similar users:

$$\hat{r}_{ij} = \sum_{k=1}^{|\bar{U}|} \frac{1}{|\bar{U}|} [\phi_k(u_{k,t})]^T v_{k,e}^j. \quad (6)$$

where $\hat{r}_{ij}$ denotes the predicted rating. As shown in Fig. 3, we test three methods on two datasets, and EMCDR -w/FL achieves a significant improvement on both metrics that verifies the effectiveness of traditional CDR methods in our scenario. Although precise matching through user-embedding can ensure the maximum transfer of common knowledge for the shared user, directly exposing user-embedding to clients in federated learning severely violates user privacy and is undesirable.

4.2 Why does Differential Privacy Fail to Work?

To better protect user privacy, we try to adopt differential privacy to the uploaded user embedding. First, we conduct experiments with the LDP technique on MovieLens-100K in Table 1. As expected, with a smaller privacy budget, user privacy can be guaranteed and when the privacy budget exceeds a certain threshold, the effectiveness of privacy information protection will no longer be achieved. However, due to the relatively small number of parameters in the embedding, when perturbation is applied to the model parameters to effectively protect user information, the recommendation performance decreases significantly. Achieving a balance between user privacy and recommendation performance is challenging. To address this issue, we next propose a novel FedRS framework with enhanced privacy and satisfying performance.

4.3 FR-JVE: Secure Preference-based Framework

To better balance the privacy and recommendation performance, we avoid directly exposing the user embedding and construct a secure rating preference for each client, serving as the transferable collaborative filtering signals to match similar users rather than the user embedding. Intuitively, a user's rating preference is suitable to represent the main characteristics of that user. For example, in [44], the rating preference of the user is the weighted sum of item embedding that is rated by this user. To ensure user privacy, here we apply model inversion techniques to distill the rating preference from both the user and item information for each client. Furthermore, we observe that the data format of each user and item is indifferntiable one-hot representation, impeding the optimization of the model inversion training. The framework of our proposed method is illustrated in Fig. 2. Consequently, we first apply the normal distribution and softmax function to initialize the data format for model inversion. Suppose that the user u_k^i and item v_k^j in client k is recognized as :

$$u_k^i = (\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_{|U_k|}), \quad v_k^j = (\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{|V_k|}). \quad (7)$$

where $\mathbf{u}_i, \mathbf{v}_j \in \{0, 1\}$. Then we transform the one-hot representation with normal distribution and softmax function:

$$\mathbf{u}_i = \frac{\exp(\hat{\mathbf{u}}_i)}{\sum_{i=1}^{|\mathcal{U}_k|} \exp(\hat{\mathbf{u}}_i)}, \quad \mathbf{v}_j = \frac{\exp(\hat{\mathbf{v}}_j)}{\sum_{j=1}^{|\mathcal{V}_k|} \exp(\hat{\mathbf{v}}_j)}, \text{ where } \hat{\mathbf{u}}_i, \hat{\mathbf{v}}_j \sim \mathcal{U}(0, 1). \quad (8)$$

With the new format defined in Eq.(8), each client can distill the rating preference for the user u_k^i with the model f_k :

$$(p_k^i | u_k^i, r_i) = \min_{v_k} \mathcal{L}(f_k(u_k^i, v_k), r_i). \quad (9)$$

During the training process, the gradient of u_k^i is frozen. Each element of the rating preference $p_k^i = (p_1, p_2, \dots, p_{|\mathcal{V}_k|})$ is a weight for the whole item embedding V_k . Then, the client k will train the matrix factorization with the new data format like Eq.(1) to obtain the item embedding. It may raise concerns that an individual user might overlook numerous items. However, many other users involved in the federated training have rated these neglected items. As a result, these ratings contribute to well-generalized item embeddings. The overall item embedding embodies enriched knowledge, thereby facilitating the informative user characteristics. Inspired by the attention mechanism, which enables each component to contribute distinctively when compressing various elements into a singular representation, we obtain the final rating preference P_k^i for the user u_k^i as follows:

$$P_k^i = \sum_{j=1}^{|\mathcal{V}_k|} p_j v_{k,e}^j. \quad (10)$$

In this work, the rating preference will be used to transfer the common knowledge from other clients, which acts as a collaborative filtering signal enhanced with secure user information. Like Eq.(5), each client locally trains a bridge function to map the aggregated rating preferences from other clients to the local rating preference.

However, simply employing the rating preference cannot match similar users like Eq.(6). Here we search the top-k similar users with the rating preference for better prediction:

$$S_k^i = \text{top}_k \left(|r - \min_{u_{k,e}^i} [u_{k,e}^i]^T \phi_k(P_{k,t}^i)| \right). \quad (11)$$

Where S_k^i is the matrix where each column is a searched user embedding, K and r are hyper-parameters that denote the number of collected users and the expected recommendation score, $\phi(\cdot)$ is a bridge function defined in Eq.(5) to map the rating preference here. Then, the final prediction for the user u_k^i will be implemented like Eq.(10). The workflow and privacy analysis of FR-JVE are provided in Appendix A and B.

5 Experiments

5.1 Datasets & Baselines

A thorough experimental study has been conducted to assess the performance of the FR-JVE in two popular scenarios with four recommendation datasets: (1) **Rating Prediction**: MovieLens-100K (ML-100K)[7] and MovieLens-1M (ML-1M). (2) **Top-K Recommendation**: LastFM-2K(LastFM) [3] and QB-article [41]. The details of these datasets are outlined in Appendix D.

For a fair comparison with other works, we follow the same protocols proposed by [29] to stimulate FL settings. We evaluate our method with the following baselines. **1. User-based Federated Recommendation**: FedMF [4], FedNCF [10], PFedRec [27], FedRAP [21]; **2. Customed Methods**: FedMF [4]-w/DP, EMCDR [28]-w/FL+DP; **3. Federated Cross-Domain Recommendation**: FedCDR [30], P2FCDR [5]. More details are provided in Appendix C.

5.2 Configurations

Unless otherwise mentioned, to build the system with partially overlapped users and items, we first assign each client with random private users and items and then use the Dirichlet distribution

Table 2: Performance comparison of various methods for top-k recommendation.

Categories	Methods	Metrics	LastFM-2K			QB-article-10000			QB-article		
			@ 10	@ 20	@ 50	@ 10	@ 20	@ 50	@ 10	@ 20	@ 50
Customed Methods	FedMF-w/DP	HR	0.1954	0.2725	0.4907	0.2140	0.3102	0.5020	0.2350	0.3188	0.4971
		NDCG	0.1176	0.1369	0.1797	0.1263	0.1505	0.1883	0.1350	0.1561	0.1910
	EMCDR	HR	0.3162	0.4085	0.6087	0.3773	0.4724	0.6421	0.4667	0.5523	0.7029
		NDCG	0.2146	0.2378	0.2770	0.2315	0.2555	0.2889	0.2996	0.3212	0.3509
	EMCDR-w/FL+DP	HR	0.4721	0.5501	0.7232	0.5426	0.6405	0.7899	0.6997	0.7686	0.8685
		NDCG	0.3536	0.3732	0.4072	0.3316	0.3563	0.3859	0.4411	0.4586	0.4783
Federated CDR	FedCDR	HR	0.2839	0.4073	0.5890	0.4243	0.5387	0.6845	0.2891	0.3858	0.7666
		NDCG	0.1614	0.1924	0.2284	0.2388	0.2678	0.2967	0.1756	0.1998	0.2745
	P2FCDR	HR	0.3466	0.4620	0.6808	0.3389	0.4338	0.6133	0.3690	0.4712	0.6502
		NDCG	0.2182	0.2472	0.2902	0.1981	0.2221	0.2572	0.2168	0.2426	0.2778
User-Based FedRS	FedMF	HR	0.2087	0.2940	0.4963	0.2460	0.3296	0.4960	0.2744	0.3506	0.4953
		NDCG	0.1256	0.1470	0.1865	0.1428	0.1638	0.1965	0.1652	0.1843	0.2127
	FedNCF	HR	0.3115	0.4394	0.6723	0.3291	0.4755	0.7176	0.4230	0.5588	0.7844
		NDCG	0.1958	0.2278	0.2736	0.1825	0.2191	0.2668	0.2445	0.2787	0.3234
	PFedRec	HR	0.5818	0.6404	0.7779	0.6355	0.6952	0.8123	0.7066	0.7478	0.8851
		NDCG	0.4247	0.4394	0.4663	0.3922	0.4074	0.4303	0.4583	0.4687	0.4954
	FedRAP	HR	0.6010	0.6528	0.7524	0.7325	0.8022	0.8311	0.8276	0.8951	0.9115
		NDCG	0.4358	0.4489	0.4683	0.4320	0.4501	0.4558	0.4999	0.5175	0.5207
	FR-JVE (Ours)	HR	0.6578	0.6866	0.8093	0.7517	0.8490	0.8909	0.8635	0.9148	0.9421
		NDCG	0.4684	0.4882	0.5022	0.4524	0.4871	0.4950	0.5141	0.5421	0.5633

Table 3: Performance comparison of various methods for rating prediction.

Dataset	Metric	Customed Methods			Federated CDR		User-based Federated Recommendation				
		FedMF-w/DP	EMCDR	EMCDR-w/FL+DP	FedCDR	P2FCDR	FedMF	FedNCF	PFedRec	FedRAP	FR-JVE
MovieLens-100K	MAE	4.7998	2.1779	1.2693	2.0535	1.4178	3.6019	1.6982	1.2503	1.3248	1.1765
	RMSE	7.1237	2.6074	2.0796	2.6883	1.7954	4.3780	2.1153	1.6465	1.6725	1.4725
MovieLens-1M-3000	MAE	5.2519	1.7794	1.2596	2.3624	1.2620	3.7927	1.6276	1.2060	1.1260	1.0496
	RMSE	10.8743	2.1641	2.7304	3.1643	1.6493	4.5506	2.0689	1.5565	1.4253	1.2554
MovieLens-1M	MAE	4.3850	1.5118	1.0869	1.9701	1.1593	3.6809	1.4435	1.1689	1.0912	1.0150
	RMSE	6.9106	1.8612	2.1763	2.6428	1.4699	4.3610	1.8457	1.5021	1.3722	1.2210

$\text{Dir}(\alpha = 10)$ [32] to distribute shared users to yield data heterogeneity for all datasets where a smaller α indicates higher data heterogeneity. Here, we report the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) as the metrics for the rating prediction [28] and Hit Rate (HR@K) and Normalized Discounted Cumulative Gain (NDCG@K) for the top-k recommendation [27, 21]. In this work, we set $K = 10$. We illustrate all the settings with all the benchmark parameters in Appendix D.

5.3 Performance Overview

Main Results. Table 2 and 3 comprehensively showcase the efficacy of various methods on two tasks. For the top-k recommendation task, when examining the HR@10 metric, FR-JVE outperforms other methods by significant margins. As the evaluation metrics extend to higher positions, FR-JVE continues to exhibit robust performance, indicating its ability to provide relevant recommendations even when considering a larger pool of candidates. Notably, PFedRec and FedRAP outperform the Federated CDR method across most datasets and achieve relatively good results due to the gain of the linear layer within their frameworks. The performance of EMCDR-w/FL+DP is closely related to the privacy budget, and the model undergoes a significant degradation under strict privacy requirements. Here, we report the case with $\epsilon = 0.2$, and the remaining analysis is provided in Table 1.

Communication Efficiency. Fig. 4 shows the evaluation of various methods in terms of convergence and communication efficiency. Here, we record evaluations for these methods for every two iterations. Under this joint venture ecosystem, the convergence of the FedMF method is extremely challenging, making it difficult to achieve effective recommendation performance. While FedRAP is more complex than other baselines, it needs more iterations to converge. FR-JVE achieves the best evaluations in less than 20 iterations and displays a fast convergence trend within a shorter time of 80 iterations.

Ablation Study. As shown in Table 5, we evaluate the effects of each module in our model via ablation studies. The -w/o bridge function denotes the performance without using the bridge function but directly employing the rating preference from other clients. The -w/o rating preference means

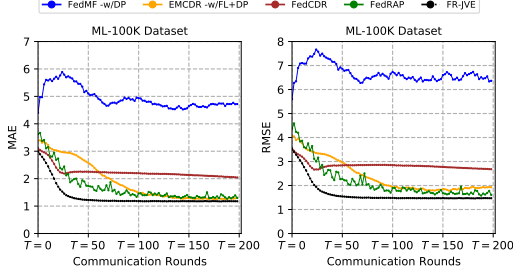


Figure 4: Convergence and efficiency comparison of various methods on the ML-100K dataset.

the client communicates with the only item embedding without user information. Compared with FR-JVE, the performance of FR-JVE -w/o rating preference degrades evidently. Specifically, the relatively less prominent role of the bridge function module may be constrained by issues related to the dataset, specifically that different clients did not introduce many new items or items that are sparse among clients. Experiment results verify the effectiveness of all modules.

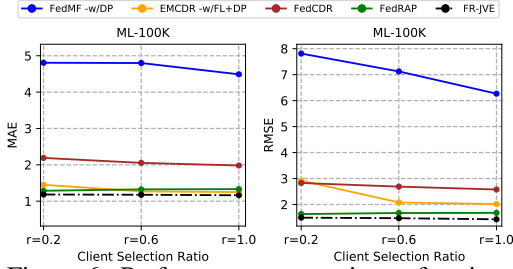


Figure 6: Performance comparison of various methods w.r.t. ratio r between active clients and total clients in each round.

Data Distribution. Fig. 5 displays two metrics with different ratios between local unique users and overlapped users on MovieLens-100K. As shown in this figure, we select three different ratios to divide the dataset and experimental results show that all methods achieve an improvement with the decline in local unique users. The underlying reason may be that the more overlapped users there are, the more associations between users can be captured, and different clients can obtain useful information from other clients to improve recommendation performance. However, when the number of locally unique users is large, the differences in data distribution among different clients are relatively significant, leading to a decline in the effectiveness of federated learning.

Table 5: Ablation study with two tasks.

Methods	MovieLens-100K		LastFM-2K	
	MAE	RMSE	HR@10	NDGC@10
FR-JVE (Ours)	1.1765	1.4725	0.6578	0.4684
-w/o bridge function	2.4200	2.7768	0.6281	0.4361
-w/o rating preference	2.9596	3.4988	0.5416	0.3317

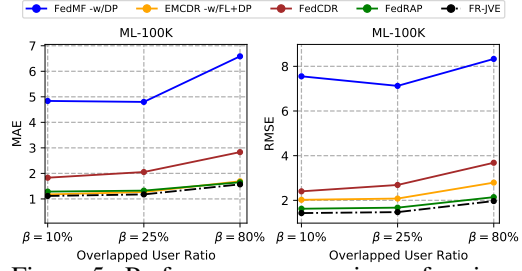


Figure 5: Performance comparison of various methods w.r.t. ratio β between local unique users and shared overlapped users.

Table 4: Evaluation of the Top-K searching of FR-JVE in two federated recommendation scenarios. Here we underline the value used in our experiments.

Dataset	Metric	Top-K Search			
		$K = 100$	$K = 200$	$K = 400$	$K = 800$
ML-100K	MAE	1.2374	1.1765	1.2248	1.8145
	RMSE	1.5861	1.4725	1.4995	2.1663
LastFM	HR@10	0.6135	0.6394	0.6506	0.6578
	NDGC@10	0.4211	0.4458	0.4547	0.4684

Table 6: Time cost of Top-K searching with ML-100K (100 users, 200 items).

Time (ms)	$K = 100$	$K = 200$	$K = 400$	$K = 800$
Other baselines	9.9449	9.3909	9.6094	10.1382
FR-JVE (Ours)	12.4295	11.9481	12.1262	12.7664

Parameter Sensitivity Analysis. Fig. 6 provides the two metrics under different ratios between active clients and total clients. FR-JVE performs best with different ratios, and the lower MAE and RMSE loss is achieved by applying more active clients. Furthermore, we conduct additional research on the number of searching users K . When the value of K is too small, the client fails to migrate sufficient knowledge from limited similar users, leading to insignificant improvement in performance. Conversely, if an excessive amount of user knowledge is migrated, it may introduce redundant information that impacts the model’s judgment.

To discuss the potential extra time cost introduced by the top-k search, we conducted time tests for different k values while controlling other variables in Table 6. We assume other methods incur equal time costs with similar inference approaches. FR-JVE requires an extra step for a top-k search. The cost for "Other baselines" should theoretically be independent of the k value, but there are fluctuations. The experiments show that our top-k search does not significantly increase the cost.

Table 7: Three different bridge functions.

FR-JVE (Bridge Function)	MovieLens-100K		LastFM-2K	
	MAE	RMSE	HR@10	NDCG@10
Linear Trans.	1.1765	1.4725	0.6578	0.4684
Logarithmic Trans.	1.1782	1.4801	0.6534	0.4609
Logistic Trans.	1.1698	1.4592	0.6488	0.4577

Table 8: Privacy enhancement with DP.

Privacy Budget	MovieLens-100K		LastFM-2K	
	MAE	RMSE	HR@10	NDCG@10
FR-JVE	1.1765	1.4725	0.6578	0.4684
-w/DP($\epsilon = 0.1$)	1.1904	1.4918	0.6411	0.4572
-w/DP($\epsilon = 0.2$)	1.2057	1.4987	0.6251	0.4430

We use a linear function for the mapping since our experiments are conducted on standard datasets. Although real-world data may require more complex bridge functions, the results in Table 7 show limited performance gains from more complex architectures. Complex bridge functions risk overfitting, especially with sparse or noisy transferred knowledge, whereas a simple linear network effectively captures essential cross-client relationships while maintaining good generalization.

Privacy Enhancement. In our method, each client transfers privacy-friendly common knowledge from other clients without using the LDP algorithm, but with a distilled user’s rating preference from the local dataset. Furthermore, we also provide experiments in Table 8 that apply differential privacy to our rating preferences for reference. However, our method itself can guarantee user information privacy. Considering that differential privacy can impact performance, we did not utilize it further.

Table 9: Scalability of participants.

Methods	MovieLens-100K		LastFM-2K	
	MAE	RMSE	HR@10	NDCG@10
FR-JVE ($C = 5$)	1.1765	1.4725	0.6578	0.4684
$C = 10$	1.1979	1.5101	0.6132	0.4152
pFedRec ($C = 5$)	1.2503	1.6465	0.5818	0.4247
$C = 10$	1.2923	1.7524	0.5387	0.3859

Table 10: Add-on of Other baselines.

Methods	MovieLens-100K		LastFM-2K	
	MAE	RMSE	HR@10	NDCG@10
FedNCF	1.6892	2.1153	0.3115	0.1958
FR-JVE -w/FedNCF	1.5696	1.9842	0.5950	0.4203
pFedRec	1.2503	1.6465	0.5818	0.4247
FR-JVE -w/pFedRec	1.1938	1.4937	0.6018	0.4409

Scalability and Efficiency. In this paper, we focus on federated recommendation within a joint-venture ecosystem, where each participant holds substantial user-item data with natural overlaps. This setting aligns with cross-silo FL, featuring few participants but complex data distributions. In contrast, cross-device FL (e.g., $C > 20$) involves numerous users with limited data, which differs from our target scenario. To validate our approach, we vary the number of participants C across three configurations on two datasets. Results in Table 9 confirm the effectiveness of our method.

Flexibility and Lightweight. The key of FR-JVE is to allow each client to transfer more common knowledge from other clients with a distilled user’s rating preference from the local dataset. Since this rating preference is unrelated to the backbone of federated recommendation algorithms, we can integrate it with other basic algorithms. In our paper, we employed the most common collaborative filtering. We will supplement the integration with well-known FedNCF and pFedRec algorithms. Experimental results 10 demonstrate that our algorithm can be combined with any basic federated recommendation algorithm to enhance performance.

6 Conclusion

In this paper, we delve into the challenges of developing a federated recommendation system tailored for joint venture ecosystems. We propose an innovative and privacy-enhancing framework, FR-JVE, which leverages the user’s rating preferences as a filtering signal to transmit common knowledge, thereby enhancing recommendation performance while safeguarding user privacy. Extensive experiments conducted on various settings and baselines show that FR-JVE achieves significant improvement in recommendation performance.

Acknowledgments and Disclosure of Funding

This work is supported by the National Key Research and Development Program of China under grant 2024YFC3307900; the National Natural Science Foundation of China under grants 62376103, 62302184, 62436003 and 62206102; Major Science and Technology Project of Hubei Province under grant 2024BAA008; Hubei Science and Technology Talent Service Project under grant 2024DJC078; Ant Group through CCF-Ant Research Fund; and Fundamental Research Funds for the Central Universities under grant YCJJ20252319. The computation is completed in the HPC Platform of Huazhong University of Science and Technology.

References

- [1] Muhammad Ammad-Ud-Din, Elena Ivannikova, Suleiman A Khan, Were Oyomno, Qiang Fu, Kuan Eeik Tan, and Adrian Flanagan. Federated collaborative filtering for privacy-preserving personalized recommendation system. *arXiv preprint arXiv:1901.09888*, 2019.
- [2] Muhammad Ammad-Ud-Din, Elena Ivannikova, Suleiman A Khan, Were Oyomno, Qiang Fu, Kuan Eeik Tan, and Adrian Flanagan. Federated collaborative filtering for privacy-preserving personalized recommendation system. *arXiv preprint arXiv:1901.09888*, 2019.
- [3] Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik. Second workshop on information heterogeneity and fusion in recommender systems (hetrec2011). In *Proceedings of the fifth ACM conference on Recommender systems*, pages 387–388, 2011.
- [4] Di Chai, Leye Wang, Kai Chen, and Qiang Yang. Secure federated matrix factorization. *IEEE Intelligent Systems*, 36(5):11–20, 2020.
- [5] 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.
- [6] Jiahua Dong, Lixu Wang, Zhen Fang, Gan Sun, Shichao Xu, Xiao Wang, and Qi Zhu. Federated class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10164–10173, 2022.
- [7] F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.
- [8] Chaoqun Hou, Yuanhang Zhou, Yi Cao, and Tong Liu. Ecat: A entire space continual and adaptive transfer learning framework for cross-domain recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2885–2889, 2024.
- [9] Folasade Olubusola Isinkaye, Yetunde O Folajimi, and Bolande Adefowoke Ojokoh. Recommendation systems: Principles, methods and evaluation. *Egyptian informatics journal*, 16(3):261–273, 2015.
- [10] Xueyong Jiang, Baisong Liu, Jiangchen Qin, Yunchong Zhang, and Jiangbo Qian. Fedncf: federated neural collaborative filtering for privacy-preserving recommender system. In *2022 International joint conference on neural networks (IJCNN)*, pages 1–8. IEEE, 2022.
- [11] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. Multi-interest network with dynamic routing for recommendation at tmall. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 2615–2623, 2019.
- [12] Li Li, Yuxi Fan, Mike Tse, and Kuo-Yi Lin. A review of applications in federated learning. *Computers & Industrial Engineering*, 149:106854, 2020.
- [13] Yichen Li, Qunwei Li, Haozhao Wang, Ruixuan Li, Wenliang Zhong, and Guannan Zhang. Towards efficient replay in federated incremental learning, 2024.
- [14] Yichen Li, Yijing Shan, Yi Liu, Haozhao Wang, Wei Wang, Yi Wang, and Ruixuan Li. Personalized federated recommendation for cold-start users via adaptive knowledge fusion. In *Proceedings of the ACM on Web Conference 2025, WWW '25*, page 2700–2709, New York, NY, USA, 2025. Association for Computing Machinery.
- [15] Yichen Li, Haozhao Wang, Yining Qi, Wei Liu, and Ruixuan Li. Re-fed+: A better replay strategy for federated incremental learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.
- [16] Yichen Li, Haozhao Wang, Wenchao Xu, Tianzhe Xiao, Hong Liu, Minzhu Tu, Yuying Wang, Xin Yang, Rui Zhang, Shui Yu, Song Guo, and Ruixuan Li. Unleashing the power of continual learning on non-centralized devices: A survey, 2024.
- [17] Yichen Li, Yuying Wang, Haozhao Wang, Yining Qi, Tianzhe Xiao, and Ruixuan Li. Fedssi: Rehearsal-free continual federated learning with synergistic synaptic intelligence. In *Forty-second International Conference on Machine Learning*.
- [18] Yichen Li, Wenchao Xu, Yining Qi, Haozhao Wang, Ruixuan Li, and Song Guo. Sr-fdil: Synergistic replay for federated domain-incremental learning. *IEEE Transactions on Parallel and Distributed Systems*, 35(11):1879–1890, 2024.

- [19] Yichen Li, Wenchao Xu, Haozhao Wang, Ruixuan Li, Yining Qi, and Jingcai Guo. Personalized federated domain-incremental learning based on adaptive knowledge matching, 2024.
- [20] Zhitao Li, Xueyang Wu, Weike Pan, Youlong Ding, Zeheng Wu, Shengqi Tan, Qian Xu, Qiang Yang, and Zhong Ming. Fedcore: Federated learning for cross-organization recommendation ecosystem. *IEEE Transactions on Knowledge and Data Engineering*, 2024.
- [21] Zhiwei Li, Guodong Long, and Tianyi Zhou. Federated recommendation with additive personalization, 2024.
- [22] Feng Liang, Weike Pan, and Zhong Ming. Fedrec++: Lossless federated recommendation with explicit feedback. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 4224–4231, 2021.
- [23] Huang Lihua, Ganglan Hu, and Xianghua Lu. E-business ecosystem and its evolutionary path: the case of the alibaba group in china. *Pacific Asia Journal of the Association for Information Systems*, 1(4):3, 2009.
- [24] Guanyu Lin, Feng Liang, Weike Pan, and Zhong Ming. Fedrec: Federated recommendation with explicit feedback. *IEEE Intelligent Systems*, 36(5):21–30, 2020.
- [25] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in Neural Information Processing Systems*, 33:2351–2363, 2020.
- [26] Rongbing Liu. The role of alipay in china. *Nijmegen, Radboud University, Nijmegen, The Netherlands*, retrieved from: www.ru.nl/publish/pages/769526/z01_rongbing_liu.pdf, 2015.
- [27] Yuhang Ma, Zhongle Xie, Jue Wang, Ke Chen, and Lidan Shou. Continual federated learning based on knowledge distillation. In Lud De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 2182–2188. International Joint Conferences on Artificial Intelligence Organization, 7 2022. Main Track.
- [28] 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.
- [29] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguerre y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.
- [30] 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.
- [31] Jeffrey Melnik. China’s “national champions” alibaba, tencent, and huawei. *Education About Asia*, 24(2):28–33, 2019.
- [32] Thomas Minka. Estimating a dirichlet distribution, 2000.
- [33] Khalil Muhammad, Qinqin Wang, Diarmuid O’Reilly-Morgan, Elias Tragos, Barry Smyth, Neil Hurley, James Geraci, and Aonghus Lawlor. Fedfast: Going beyond average for faster training of federated recommender systems. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1234–1242, 2020.
- [34] Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):1–7, 2020.
- [35] Stacey Truex, Ling Liu, Ka-Ho Chow, Mehmet Emre Gursoy, and Wenqi Wei. Ldp-fed: Federated learning with local differential privacy. In *Proceedings of the third ACM international workshop on edge systems, analytics and networking*, pages 61–66, 2020.
- [36] Haozhao Wang, Yichen Li, Wenchao Xu, Ruixuan Li, Yufeng Zhan, and Zhigang Zeng. Dafkd: Domain-aware federated knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20412–20421, 2023.
- [37] Haozhao Wang, Haoran Xu, Yichen Li, Yuan Xu, Ruixuan Li, and Tianwei Zhang. FedCDA: Federated learning with cross-rounds divergence-aware aggregation. In *The Twelfth International Conference on Learning Representations*, 2024.

- [38] Yuhan Wang, Qing Xie, Mengzi Tang, Lin Li, Jingling Yuan, and Yongjian Liu. Amazon-kg: A knowledge graph enhanced cross-domain recommendation dataset. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 123–130, 2024.
- [39] Chen Wei, Yixin Ji, Zeyuan Chen, Jia Xu, and Zhongyi Liu. Llmgr: Large language model-based generative retrieval in alipay search. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2847–2851, 2024.
- [40] Liu Yang, Ben Tan, Vincent W Zheng, Kai Chen, and Qiang Yang. Federated recommendation systems. *Federated Learning: Privacy and Incentive*, pages 225–239, 2020.
- [41] Guanghu Yuan, Fajie Yuan, Yudong Li, Beibei Kong, Shujie Li, Lei Chen, Min Yang, Chenyun Yu, Bo Hu, Zang Li, et al. Tenrec: A large-scale multipurpose benchmark dataset for recommender systems. *Advances in Neural Information Processing Systems*, 35:11480–11493, 2022.
- [42] Honglei Zhang, Fangyuan Luo, Jun Wu, Xiangnan He, and Yidong Li. Lightfr: Lightweight federated recommendation with privacy-preserving matrix factorization. *ACM Transactions on Information Systems*, 41(4):1–28, 2023.
- [43] Lin Zhang, Li Shen, Liang Ding, Dacheng Tao, and Ling-Yu Duan. Fine-tuning global model via data-free knowledge distillation for non-iid federated learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 10164–10173.
- [44] Yongchun Zhu, Zhenwei Tang, Yudan Liu, Fuzhen Zhuang, Ruobing Xie, Xu Zhang, Leyu Lin, and Qing He. Personalized transfer of user preferences for cross-domain recommendation. In *Proceedings of the fifteenth ACM international conference on web search and data mining*, pages 1507–1515, 2022.
- [45] Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *International Conference on Machine Learning*, pages 12878–12889. PMLR, 2021.

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: The summarized contributions are provided in the abstract and introduction.

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: [\[No\]](#)

Justification: The method may have limitations.

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: [\[NA\]](#)

Justification: We did not include new theorems.

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 could provide more details and codes to support the reproducibility.

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 could provide our code if required.

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 presented the settings in the appendix.

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: The results are accompanied by error bars, confidence intervals, or statistical significance tests.

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 provide the amount of compute required for experiments in the appendix.

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: [Yes]

Justification: We have 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: [NA]

Justification: There is no societal impact of the work performed.

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: The paper 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 cited the original paper that produced the code package or dataset.

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: The paper does not release new assets.

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: The 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: The 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.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: We did not use LLM for the core method.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

A Algorithm Flow for FR-JVE

Algorithm 1: FR-JVE

Input : T : communication round; K : client number; D_k : local dataset for the client k ; $u_{k,e}$: local user embedding; $v_{k,e}$: local item embedding; $\phi_k(\cdot)$: local bridge function; f_k : local inversion model.

Output : v_e : Item Embedding.

- 1 Initialize the parameter w ;
- 2 **Before Federated Training:**
- 3 **for** each selected client $k \in [K]$ **in parallel** **do**
- 4 Transform the local data format with (7)(8);
- 5 Distill the rating preference with (9)(10);
- 6 // Upload/Download rating preference
- 7 **end**
- 8 **At Training Stage:**
- 9 **for** $c = 1$ **to** T **do** // communication round
- 10 Server randomly selects a subset of devices S_t ;
- 11 **for** each selected client $k \in S_t$ **in parallel** **do**
- 12 Train the local matrix factorization with new data format with (1);
- 13 Train the bridge function for rating preference with (5);
- 14 Send the item embedding $v_{k,e}$ back to the server.
- 15 **end**
- 16 $v_e \leftarrow \text{ServerAggregation}(\{v_{k,e}\}_{k \in S_t})$
- 17 **end**
- 18 **At Inference Stage:**
- 19 Each client matches the similar users for the shared users with (2);
- 20 The final prediction will be made by average rating with (6).

B Privacy Analysis for FR-JVE

In our proposed FR-JVE framework, user embedding never leaves the local client. The only transferred knowledge is the rating preference and item embedding, of which the item embedding is necessary and has little user privacy, and the rating preference is the linear combination of the column vectors of item embedding V_k using the elements of p_k as the coefficients. However, the distillation of the rating preference employs the frozen gradient of the user information. We need to figure out whether attackers can recover user embedding information through rating preference. Assuming that attackers know their target rating, inferring the original user embedding can be understood as solving a one-line linear equation with multiple variables, where the number of variables is the dimension size of the embedding. According to Eq.(1), for the user u_k^i at the client k , the rating is calculated as:

$$\underbrace{r_i}_{|U_k| \times |V_k|} = \underbrace{[u_{k,e}^i]}_{d \times |U_k|}^T \underbrace{\phi_k(P_{k,t}^i)}_{d \times |V_k|}.$$

It is clear that when the embedding size of users or items $d > 1$, then determining the value of each dimension in a user embedding involves solving a *multi-variable linear equation*. These variables are interrelated and cannot be solved separately, and thus, attackers fail to infer the user embedding.

C Baselines

1. User-based Federated Recommendation:

FedMF [4]: This approach employs matrix factorization in federated contexts to mitigate information leakage by encrypting the gradients of user and item embedding.

FedNCF [10]: It is the federated version of NCF. It updates user embedding locally on each client and synchronizes the item embedding on the server for global updates.

PFedRec [27]: It aggregates item embedding from clients on the server side and leverages them in

the evaluation to facilitate bipartite personalized recommendations.

FedRAP [21]: This approach further employs personalized techniques to guide the local training by gradually increasing the regularization weights to mitigate performance degradation.

2. Customed Methods:

FedMF [4] - **w/DP**: It conducts local matrix factorization and global aggregation of both user embedding and item embedding. For a fair comparison, the differential privacy will be only added on the uploaded user embedding.

EMCDR [28]- **w/FL+DP**: This method is a popular embedding-and-mapping framework for handling CDR. For a fair comparison, here we only map the user embedding and add the differential privacy to the uploaded user embedding.

3. Federated Cross-Domain Recommendation:

FedCDR [30]: It is a privacy-preserving federated CDR model designed for individual customer scenarios. It builds a cross-domain embedding transformation model on the server side.

P2FCDR [5]: It is the first privacy-preserving federated framework for dual-target cross-domain recommendation that enhances information fusion at the feature level.

D Datasets & Configurations

These datasets are commonly employed for evaluating recommendation systems. Specifically, two MovieLens datasets sourced from the MovieLens platform feature movie ratings spanning 1 to 5, with each user contributing at least 20 ratings. LastFM, a music recommendation dataset, comprises users' preferred artist lists and associated tags. QB-article, an implicit feedback dataset, is derived from user interaction logs.

Each experiment set is run twice, and we take each run's final 10 rounds' accuracy and calculate the average value and standard variance. We use Adam as an optimizer with a linear learning rate schedule. We set the remaining parameters according to the values in the original open-source code.

Table 11: Experimental Details. Analysis of various considered settings of different datasets in the experiments section.

Attributes	Rating Prediction		Top-K Recommendation	
	ML-100K	ML-1M	LastFM	QB-article
Ratings	100,000	1,000,209	92,834	348,736
Users	943	6,040	1,892	24,516
Items	1,682	3,952	17,632	7,355
Sparsity	$s = 93.70\%$	$s = 95.81\%$	$s = 99.72\%$	$s = 99.81\%$
Batch Size	$b = 32$	$b = 64$	$b = 256$	$b = 256$
Learning Rate	$l = 0.001$	$l = 0.001$	$l = 0.04$	$l = 0.01$
Local Users	100	600	200	2,400
Local Items	200	400	1,700	700
Shared Users	443	3,040	892	12,516
Client numbers	$C = 5$	$C = 5$	$C = 5$	$C = 5$
Top-K Search	$K = 200$	$K = 3000$	$K = 1600$	$K = 7000$
Local training epoch	$E = 5$	$E = 5$	$E = 5$	$E = 5$
Client selection ratio	$k = 0.6$	$k = 0.6$	$k = 0.6$	$k = 0.6$
Communication Round	$T = 200$	$T = 200$	$T = 200$	$T = 200$