Joint Item Embedding Dual-view Exploration and Adaptive Local-Global Fusion for Federated Recommendation

Pengyang Zhou
zhoupy@zju.edu.cn
Zhejiang University
Hangzhou, China

Chaochao Chen*
zjuccc@zju.edu.cn
Zhejiang University
Hangzhou, China

Weiming Liu
21831010@zju.edu.cn
Zhejiang University
Hangzhou, China

Wenkai Shen
shenwenkai@mail.nwpu.edu.cn
Northwestern Polytechnical University
Xi'an, China

Xinting Liao
xintingliao@zju.edu.cn
Zhejiang University
Hangzhou, China

Huarong Deng
codydeng@gmail.com
OPPO
Shenzhen, China

Zhihui Fu
luca@oppo.com
OPPO
Shenzhen, China

Jun Wang
junwang.lu@gmail.com
OPPO
Shenzhen, China

Wu Wen
Xiaolin Zheng
wuwen@intl.zju.edu.cn
xlzheng@zju.edu.cn
Zhejiang University
Hangzhou, China

Abstract

Federated Recommendation (FedRec) enables joint training across a large number of clients without centralizing user interaction data. However, existing FedRec methods overlook two key challenges, i.e., (1) sufficiently explore the global item embedding space, and (2) effectively achieve local and global collaboration. The former is caused by client sparsity, which leads to suboptimal item embeddings and subsequently impacts the global item embedding in both the dimension and sample views. The latter arises from the lack of modeling the relative importance of local and global contributions to personalized user preferences. To address the above challenges, we propose `FedIAR` which contains two modules, i.e., item embedding dual-view exploration and adaptive local-global fusion. The first module enhances the global item embedding by reducing redundancy in the dimension view and capturing latent item relationships in the sample view, improving representational capacity. The second module enables the adaptive fusion of local and global item embeddings based on the user preference representation, achieving personalized optimum for recommendation. Extensive experiments on six datasets demonstrate the effectiveness of `FedIAR` in improving federated recommendation performance.

*Corresponding author.

SIGIR '25, Padua, Italy

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3730016

CCS Concepts

• **Information systems** → **Recommender systems**.

Keywords

Federated Learning, Recommender Systems

ACM Reference Format:

Pengyang Zhou, Chaochao Chen, Weiming Liu, Wenkai Shen, Xinting Liao, Huarong Deng, Zhihui Fu, Jun Wang, Wu Wen, and Xiaolin Zheng. 2025. Joint Item Embedding Dual-view Exploration and Adaptive Local-Global Fusion for Federated Recommendation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3726302.3730016

1 Introduction

Personalized recommendation systems [43] have been widely adopted to provide customized content to users. However, traditional methods rely on centralized storage of user data for training, which raises significant privacy concerns [42]. Federated Learning (FL) [35] offers a privacy-preserving alternative by enabling collaborative learning without sharing raw data. In Federated Recommendation (FedRec), each client holds sparse, personal interaction data locally and only shares model parameters with others [13]. This approach operates within a cross-device FL setting [5], where large-scale client participation exacerbates data sparsity and personalization diversity, increasing the difficulty of FedRec.

Existing federated recommendation research can be broadly categorized into three types. *Firstly*, several work focuses on server-side global aggregation [12, 32, 34, 36] by user groups. They cluster similar users based on user embedding [33] or item embedding updated by clients [12], and share recommendation model parameters within each cluster. *Secondly*, some studies perform client-specific local adaptation [49, 50] for personalized learning. These methods personalize the global model for each client through local fine-tuning [49] or adjusting the item embedding based on inter-user relationships [50]. *Thirdly*, some methods separately maintain local and global information [19, 20]. For example, [19] designs a framework where the global item embedding aggregated across clients captures general item characteristics, while the local item embedding kept on each client reflects personalized user preferences. This approach demonstrates promising performance in combining global aggregation and local personalization, making it a potential solution for improving federated recommendation. However, the above three categories of methods still overlook two key challenges, i.e., ***CH1****: sufficiently explore the global item embedding space* and ***CH2****: effectively achieve local and global collaboration*.

The first challenge is attributed to client sparsity, which leads to suboptimal item embeddings that subsequently affect the global item embedding in both dimension and sample views. From the *dimension* view, the sparsity of interaction data on individual clients causes redundancy [48] in the item embeddings. This occurs because the limited data prevents each dimension from capturing distinct aspects of item characteristics. When these client-side item embeddings are aggregated, the redundancy persists and propagates to the global item embedding on the server and across other clients, leading to less diverse and informative embeddings. From the *sample* view, the insufficient local data hinders the capture of inter-item relationships. Simple aggregation techniques [19, 49] employed on the server further restrict the discovery of latent relationships within the embedding space. Without explicitly modeling these relationships, the global item embedding lacks the necessary relational structure, ultimately degrading the performance.

The second challenge arises from the lack of modeling the relative importance of local and global contributions to personalized user preferences. Although the third category of approaches has attempted to capture both local and global aspects of the item embeddings, they directly add them with a fixed weighting across all clients [19]. However, this approach does not account for the fact that the relevance of global information varies across users, depending on their interaction histories and individual preferences. Specifically, some users may rely more on global embeddings to capture general representations, while others may prioritize local embeddings that better reflect their unique preferences. This misalignment between user preferences and embedding integration can result in less optimal personalization.

To fill this gap, we propose a federated recommendation framework named FedIAR, which consists of two modules, **item embedding dual-view exploration** and **adaptive local-global fusion**. To address **CH1**, the dual-view global aggregation module optimizes the global item embedding from both dimension and sample views. Dimension-view optimization reduces redundancy by enforcing autocovariance uniformity through matrix information theory, ensuring that each dimension contributes uniquely to the diverse representations. Sample-view optimization uncovers latent categories with a Bayesian nonparametric method and strengthens item relationships through clustering enhancement. To address **CH2**, the adaptive local-global fusion module enhances the fusion of local and global item embeddings by modeling their relative importance. Instead of a simple combination, we employ adaptive gate control through a gate network that dynamically assigns personalized weights to local and global item embeddings. A personalized hypernetwork further enhances this process by generating gate network parameters based on user embeddings, allowing the fusion to adapt to individual users and achieve a personalized optimum for recommendation.

The main contributions are summarized as follows: (1) We propose a novel framework, i.e., FedIAR, for the federated recommendation problem, which contains item embedding dual-view exploration and adaptive local-global fusion. (2) The proposed item embedding dual-view exploration module improves global item embedding by reducing redundancy and capturing latent item relationships, enhancing its representational capacity. Meanwhile, the adaptive local-global fusion module enables the personalized fusion of local and global embeddings through adaptive weighting, ensuring effective local and global collaboration and achieving a personalized optimum. (3) We conduct extensive experiments on six datasets and validate the effectiveness of FedIAR.

2 Related Work

2.1 Personalized Federated Learning

Federated Learning (FL) [18, 35] has gained significant popularity in recent years as a collaborative machine learning framework, enabling multiple participants to jointly train models while preserving the privacy of their local datasets. Due to the non-independent and identically distributed (non-IID) nature of data [53], client datasets often exhibit heterogeneous distributions, presenting a key challenge in FL [21, 23]. Personalized Federated Learning (PFL) has been widely applied to address this heterogeneity, particularly in domains such as computer vision [22], natural language processing [45], and graph learning [54]. Techniques like distillation [6, 40, 56], clustering [9, 15, 25], personalized aggregation [14, 24, 51, 55] have been developed to tailor models to the specific data distributions of individual clients. However, when applying PFL to recommendation systems, additional unresolved challenges emerge due to the large number of clients involved, which exacerbates data sparsity and personalization diversity.

2.2 Federated Recommendation

Recommendation systems [26, 28] provide customized content by modeling user interaction history and preferences [17, 27, 31]. In Federated Recommendation (FedRec), each user is treated as a client, possessing their own interaction data [13]. Existing FedRec research can be broadly divided into three categories. Firstly, some studies focus on server-side global aggregation [12, 32, 34, 36], where users are grouped based on similarity, either through user embeddings [33] or item embeddings [12]. Recommendation model parameters are then shared within each group. However, obtaining reliable clustering results is difficult due to the insufficient representations of sparse individual client data. [29] Secondly, certain approaches emphasize client-specific local adaptation [49, 50] for personalized learning. These methods first share the global model in one communication round, then customize it for each client by local fine-tuning [49] or by modifying item embeddings based on relationships between users [50]. However, relying on a single round of adaptation using global model parameters limits their ability to effectively preserve local information. Thirdly, some methods maintain both local and global information separately [19, 20]. For instance, FedRAP [19] introduces a framework where the global item embeddings are aggregated across clients to capture general item characteristics, while the local item embeddings reflect personalized user preferences. However, this approach assumes that global and local embeddings have equal importance across all clients, without considering how the reliance on global information varies among users. Unlike existing methods, our framework effectively tackles the previously overlooked challenges of sufficiently exploring the global item embedding space and achieving effective local and global collaboration.

3 Method

3.1 Problem Statement

In a federated recommendation system, there are K users, denoted by $\mathcal{U} = \{u_1, u_2, \ldots, u_K\}$, and n items, denoted by $\mathcal{I} = \{i_1, i_2, \ldots, i_n\}$. Each user is treated as an individual client, possessing their interaction records $\mathcal{D}_k$. The recommendation model deployed at each client k includes a user embedding $\boldsymbol{u}_k \in \mathbb{R}^d$, a score function Θ_k, a local item embedding $Z_k \in \mathbb{R}^{n\times d}$, and the global item embedding $Z_g \in \mathbb{R}^{n\times d}$. Because of the privacy regularization, only the global item embedding is uploaded and aggregated on the server. The goal is to collaboratively train recommendation models to predict ratings between items and decentralized users, without sharing individual user-item interaction records. The overall objective is:

$$\min \sum_{k=1}^{K} \mathbb{E}_{i\sim\mathcal{I}} \left[\mathcal{L}\left(Z_g, Z_k, \Theta_k, r_{ki}\right) \right], \tag{1}$$

where $\mathcal{L}$ is the loss function, r_{ki} represents the rating between user k and item i.

3.2 Framework Overview

The overall framework of `FedIAR` is illustrated in Figure. 1, which shows the communication between the server and the k-th client. `FedIAR` consists of two modules, i.e., item embedding dual-view exploration and adaptive local-global fusion. *Firstly*, the item embedding dual-view exploration module on the server collects the set of trained global item embeddings $\{Z_{g_k}\}$ from all clients and conducts exploration from both the dimension-view and sample-view. From the dimension view, matrix information theory is applied to align the autocovariance of the global item embedding with the identity matrix, effectively reducing redundancy across different dimensions. From the sample view, a Bayesian nonparametric clustering method is used to capture the latent distribution of items. Each item is then aligned with its corresponding centroid $\boldsymbol{V}$, enhancing the relational structure among items. The dual-view updated global item embedding $\boldsymbol{Z}_g$ is then transmitted to all clients. *Secondly*, the adaptive local-global fusion module on each client k employs a gate network to predict the personalized optimal weight $\lambda_{k,i}$ for combining the local and global item embeddings for each item i. The gate network parameters $\boldsymbol{\Phi}_k$ are generated by a personalized hypernetwork $\mathcal{H}(\boldsymbol{u}_k; \boldsymbol{\Psi}_k)$, which takes local user embedding $\boldsymbol{u}_k$ as input and has $\boldsymbol{\Psi}_k$ as trainable parameters. The output of the gate network forms a fused item embedding $\hat{\boldsymbol{z}}_i$. Each client uses this personalized embedding $\hat{\boldsymbol{z}}_i$ and the local user embedding $\boldsymbol{u}_k$ to predict the rating for the item. This communication process between the server and clients iterates until the performance converges.

3.3 Item Embedding Dual-view Exploration

Motivation. Existing federated recommendation methods aggregate the global item embedding through simple averaging [19, 49], which causes the global embedding to be influenced by suboptimal item embeddings derived from sparse local data. To enhance the aggregated global item embedding and further explore the embedding space, we propose optimizing it from both the dimension and sample views. From the dimension view, minimizing redundancy helps maximize the utility of the embedding space [48], ensuring that each dimension contributes uniquely and promotes a more uniform representation. We apply matrix information theory to enforce autocovariance uniformity, resulting in a more informative embedding. From the sample view, we apply a Bayesian nonparametric method to uncover latent categories and group samples with similar characteristics. Clustering enhancement is then used to strengthen these sample-level relationships, further improving the representational capacity.

Dimension-view Redundancy Reduction. In traditional federated recommendation approaches, each client updates the global item embedding $Z_{g_k}^{t+1}$ during every training round and sends it to the server for aggregation. The standard federated averaging process can be expressed as:

$$\begin{aligned} Z_g^{t+1} = \frac{1}{K}\sum_{k=1}^{K} Z_{g_k}^{t+1} &= Z_g^t - \frac{1}{K}\sum_{k=1}^{K}\left(Z_g^t - Z_{g_k}^{t+1}\right) \\ &= Z_g^t - \frac{1}{K}\sum_{k=1}^{K}\Delta Z_{g_k}^t, \end{aligned} \tag{2}$$

where Z_g^{t+1} represents the updated global item embedding and $\Delta Z_{g_k}^t$ denotes the change in the global item embedding due to the update from client k. This simple aggregation, however, does not fully explore the embedding space, especially when the item embeddings provided by clients are suboptimal due to the sparsity of client interaction data. To maximize the utilization of each dimension in the global item embedding space, we propose incorporating a uniformity regularization into Eq. (2). This regularization is supported by a matrix information-theoretic guarantee [52]. Specifically, we introduce Matrix Cross-Entropy (MCE), which is defined for a divergence between two positive semi-definite matrices $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{d\times d}$, measuring the Matrix Kullback-Leibler divergence and Matrix Entropy simultaneously:

$$\mathrm{MCE}(\boldsymbol{X}, \boldsymbol{Y}) = \mathrm{MKL}(\boldsymbol{X}||\boldsymbol{Y}) + \mathrm{ME}(\boldsymbol{X}) = \mathrm{tr}(-\boldsymbol{X}\log \boldsymbol{Y} + \boldsymbol{Y}), \tag{3}$$

where the Matrix Kullback-Leibler divergence quantifies the difference between two matrices, defined as $\mathrm{MKL}(\boldsymbol{X}||\boldsymbol{Y}) = \mathrm{tr}(\boldsymbol{X}\log\boldsymbol{X} - \boldsymbol{X}\log\boldsymbol{Y} - \boldsymbol{X} + \boldsymbol{Y})$, and the Matrix Entropy is a measure of the uncertainty of the matrix $\boldsymbol{X}$, given by $\mathrm{ME}(\boldsymbol{X}) = -\,\mathrm{tr}(\boldsymbol{X}\log\boldsymbol{X}) + \mathrm{tr}(\boldsymbol{X})$ where log denotes the principal matrix logarithm. In order to properly assess redundancy in the global embedding space, we first compute the centered autocovariance matrix, which is defined as:

$$C_g = \frac{1}{n}\bar{Z}_g^{t\top}\bar{Z}_g^t = \frac{1}{n}\left(\boldsymbol{H}_n Z_g^t\right)^\top\left(\boldsymbol{H}_n Z_g^t\right) = \frac{1}{n} Z_g{}^{t\top}\boldsymbol{H}_n Z_g^t, \tag{4}$$

where $\bar{Z}_g^t = \boldsymbol{H}_n Z_g^t$ is the centered global item embedding. $\boldsymbol{H}_n = \boldsymbol{I}_n - \frac{1}{n}\boldsymbol{1}_n\boldsymbol{1}_n^\top$ is a centering matrix removing the global mean from

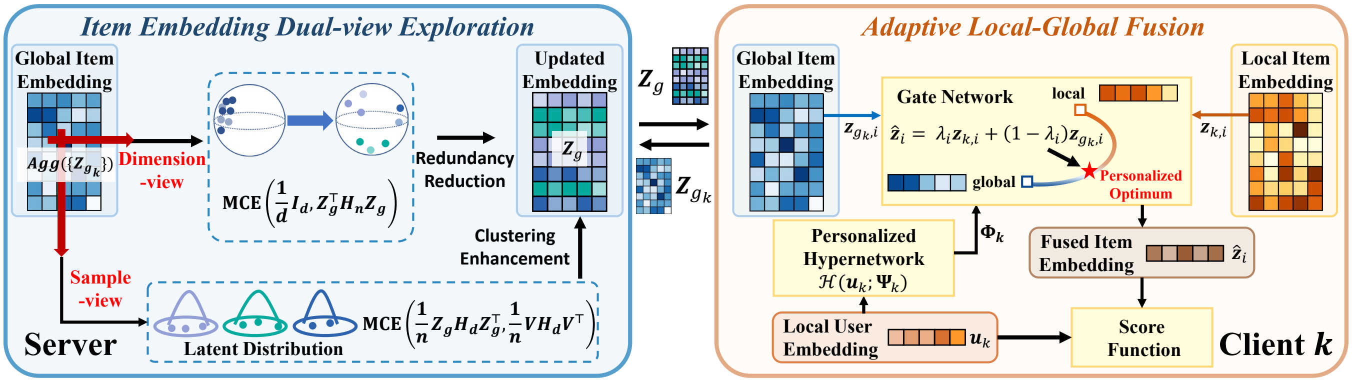


Figure 1: The main framework of the proposed FedIAR.

the embedding, which satisfies $H_n^\top = H_n$ and $H_n^2 = H_n$. I_n denotes the identity matrix with size $n \times n$. The autocovariance matrix C_g captures the dependence between the dimensions of the embedding, indicating their redundancy degree. Hence, we can define the dimension-wise autocovariance uniformity loss, i.e.,

$$\mathcal{L}_d = \mathrm{MCE}\left(\frac{1}{d}I_d + \gamma I_d, C_g + \gamma I_d\right), \tag{5}$$

where a small term γI_d is added to ensure the matrix is non-singular. In this formulation, we penalize dimensions that exhibit unnecessary correlations by aligning the autocovariance matrix with the target uniformity matrix. The target matrix represents the ideal case where all dimensions are independent and free of redundancy.

In the following, we provide more insights on the effectiveness of Eq. (5). To measure the exploration of the embedding space, we adopt total coding rate (TCR) [46], which calculates the minimal number of bits required to encode a matrix with distortion up to the constraint ϵ. TCR for the centered global item embedding $\bar{Z}_g^t$ is given as:

$$\mathrm{TCR}(\bar{Z}_g^t, \epsilon) = \frac{1}{2}\log\det\left(I_d + \frac{d}{\epsilon^2 n}\bar{Z}_g^{t\top}\bar{Z}_g^t\right). \tag{6}$$

Maximizing TCR promotes more sufficient exploration of the embedding space [30].

Theorem 1 (Sufficient Dimensional Exploration). *Minimizing the loss function in Eq. (5) is equivalent to maximizing the exploration of the global item embedding space, indicated by the relation* $\mathcal{L}_d = (1+d\gamma)(-\log(\gamma) + 1 - 2\,\mathrm{TCR}(\bar{Z}_g^t, \epsilon))$.

Proof. $\mathrm{TCR}(\bar{Z}_g^t, \epsilon)$ can be expressed as:

$$\mathrm{TCR}(\bar{Z}_g^t, \epsilon) = \frac{1}{2}\mathrm{tr}\left(\log\left(I_d + \frac{1}{\gamma}C_g\right)\right), \tag{7}$$

with γ set to $\frac{\epsilon^2}{d}$. Using the definition of MCE, Eq. (5) can be rewritten as:

$$\begin{aligned}\mathcal{L}_d &= \mathrm{tr}\left(-\left(\frac{1}{d}I_d + \gamma I_d\right)\log\left(C_g + \gamma I_d\right) + C_g + \gamma I_d\right)\\ &= -\mathrm{tr}\left(\left(\frac{1}{d}I_d + \gamma I_d\right)(\log(\gamma) I_d)\right) - 2(1+d\gamma)\mathrm{TCR}(\bar{Z}_g^t, \epsilon) + \\ &\quad \mathrm{tr}\left(C_g + \gamma I_d\right)\\ &= (1+d\gamma)(-\log(\gamma) + 1 - 2\mathrm{TCR}(\bar{Z}_g^t, \epsilon)),\end{aligned} \tag{8}$$

where the second equation follows the fact that $\mathrm{tr}\left(\log\left(C_g + \gamma I_d\right)\right) = \mathrm{tr}\left(\log(\gamma) I_d\right) + 2\,\mathrm{TCR}(\bar{Z}_g^t, \epsilon)$. □

Due to the equivalence between minimizing $\mathcal{L}_d$ and maximizing the TCR, dimension-view redundancy reduction improves the effective exploration of the embedding space.

Sample-view Clustering Enhancement. To capture latent structures within the global item embedding space, we perform clustering to group samples with similar characteristics, enabling structure understanding of sample-level relationships. We employ a Bayesian nonparametric approach based on the Dirichlet Process [41], which allows for flexible and adaptive clustering by discovering an unknown number of clusters in the global item embedding space. A Dirichlet Process $P \sim \mathrm{DP}(\alpha, G_0)$ is defined as a random probability measure on the sample space, such that for any finite measurable partition $\{A_1, A_2, \ldots, A_S\}$,

$$(P(A_1), P(A_2), \ldots, P(A_S)) \sim \mathrm{Dir}(\alpha G_0(A_1), \alpha G_0(A_2), \ldots, \alpha G_0(A_S)), \tag{9}$$

where α is the concentration parameter, G_0 is the base distribution, and $\mathrm{Dir}(\cdot)$ represents the Dirichlet distribution. To approximate the Dirichlet Process, we employ the stick-breaking construction, which provides a concrete representation of cluster probabilities. For the m-th cluster, the probability ϕ_m is assigned using the following process:

$$\beta_m \sim \mathrm{Beta}(1, \alpha), \quad \phi_m = \beta_m \prod_{j=1}^{m-1}(1-\beta_j), \tag{10}$$

where β_m is drawn from a Beta distribution, and the cumulative product reflects the stick-breaking mechanism for assigning probabilities to clusters. The base distribution of each cluster is characterized by a set of parameters $\boldsymbol{\omega}_m = (\boldsymbol{\mu}_m, \Sigma_m)$, where $\boldsymbol{\mu}_m$ is the mean vector and Σ_m is the covariance matrix that models the items within the cluster. To ensure robust and flexible modeling of the item clusters, we place a Normal-Inverse-Wishart (NIW) prior on $\boldsymbol{\omega}_m$, where the precision matrix Σ_m^{-1} follows a Wishart distribution [37]. For computational feasibility, we truncate the number of clusters in the Dirichlet Process to M, as this truncation does not alter the essential statistical properties of the resulting mixture distribution [2]. We represent the clustering potential using $\boldsymbol{\pi} \in \mathbb{R}^{n\times M}$, where $\boldsymbol{\pi}_{i,m}$ denotes the degree to which i-th item belongs to the cluster m in the global item embedding space. The cluster assignment is denoted as $\boldsymbol{c} \in \mathbb{R}^d$, where the assignment for the i-th item is given by $\boldsymbol{c}_i = \mathrm{argmax}_m\, \boldsymbol{\pi}_{i,m}$. To fit the Dirichlet Process, we approximate the true posterior distribution using a variational posterior q, which is factorized for efficient computation. Specifically, we model the variational family as:

$$q(\boldsymbol{\beta}, \boldsymbol{\omega}, \boldsymbol{\pi}) = \prod_{m=1}^{M} q(\beta_m) \prod_{m=1}^{M} q(\boldsymbol{\omega}_m) \prod_{i=1}^{n} q(\boldsymbol{\pi}_i). \tag{11}$$

The variational parameters are optimized by maximizing the evidence lower bound (ELBO), which is computed as:

$$\begin{aligned}\mathrm{ELBO} &= \mathbb{E}_q\left[\log p(z, \boldsymbol{\beta}, \boldsymbol{\omega}, \boldsymbol{\pi})\right] - \mathbb{E}_q\left[\log q(\boldsymbol{\beta}, \boldsymbol{\omega}, \boldsymbol{\pi})\right] \\ &= \mathbb{E}_q[\log p(z|\boldsymbol{\beta}, \boldsymbol{\omega}, \boldsymbol{\pi})] - \sum_{m=1}^{M} \mathrm{KL}(q(\beta_m)\|p(\beta_m)) \\ &\quad - \sum_{m=1}^{M} \mathrm{KL}(q(\boldsymbol{\omega}_m)\|p(\boldsymbol{\omega}_m)) - \sum_{i=1}^{n} \mathrm{KL}(q(\boldsymbol{\pi}_i)\|p(\boldsymbol{\pi}_i)),\end{aligned} \tag{12}$$

where $\mathrm{KL}(q(x)||p(x)) = \int q(x) \log \frac{q(x)}{p(x)} dx$ denotes the Kullback-Leibler divergence between the variational and true posterior distributions. Practically, the mean and covariance matrix parameters $\boldsymbol{\omega}_m$ are encoded by the trainable parameters ξ_m, and we collect these parameters in the set $\Xi = \{\xi_m\}_{m=1}^{M}$. Following [7], the clustering potential $\boldsymbol{\pi}$ can be computed as:

$$\begin{aligned}\boldsymbol{\pi}_{i,m} = \exp\Big(&\mathbb{E}_{\beta\sim q}[\log \phi_m] + \mathbb{E}[\log p(\boldsymbol{z}_i)] \\ &+ \mathbb{H}[q_{\psi_m}(\cdot \mid \boldsymbol{c}_i = m, \boldsymbol{z}_i)] + \mathrm{const}\Big),\end{aligned} \tag{13}$$

where $\mathbb{E}_{\beta\sim q}[\log \phi_m]$ is the expected log of the cluster probability, $\mathbb{E}[\log p(\boldsymbol{z}_i)]$ is the expected log-likelihood of the item $\boldsymbol{z}_i$ under the current model, and $\mathbb{H}[q(x)] = -\int q(x) \log q(x) dx$ represents the entropy term, which acts as a regularization to ensure diversity in cluster assignments. The sum of the clustering potential for each item is normalized, i.e., $\sum_{m=1}^{M} \boldsymbol{\pi}_{i,m} = 1$. The first term in Eq. (13), can be computed as:

$$\mathbb{E}_{\beta\sim q}[\log \phi_m] = \frac{\Gamma'(\boldsymbol{\alpha}_{1,m})}{\Gamma(\boldsymbol{\alpha}_{1,m})} - \frac{\Gamma'(\boldsymbol{\alpha}_{2,m})}{\Gamma(\boldsymbol{\alpha}_{2,m})}, \tag{14}$$

where Γ denotes the gamma function, $\alpha_{1,m}$ and $\alpha_{2,m}$ are the parameters of the Beta distribution for β_m. These parameters are the closed-form solutions for minimizing the Kullback-Leibler divergence between the variational distribution and the $\mathrm{Beta}(\beta_m; \alpha_{1,m}, \alpha_{2,m})$. Specifically, $\alpha_{1,m}$ and $\alpha_{2,m}$ are defined as:

$$\boldsymbol{\alpha}_{1,m} = 1 + \sum_{i=1}^{n} \boldsymbol{\pi}_{i,m}, \quad \boldsymbol{\alpha}_{2,m} = \alpha + \sum_{i=1}^{n} \sum_{j=m+1}^{M} \boldsymbol{\pi}_{i,j}. \tag{15}$$

Here, $\boldsymbol{\alpha}_{1,m}$ represents the count of items assigned to the m-th cluster, while $\boldsymbol{\alpha}_{2,m}$ accounts for the remaining items distributed among all clusters greater than m. By iteratively optimizing Eq. (11) to update Ξ, and using Eq. (13) and Eq. (15) to update $\boldsymbol{\pi}$, $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$, we approximate the true posterior distribution and obtain the cluster potential $\boldsymbol{\pi}$, determining the cluster assignments for all items in the global item embedding space.

After clustering, we encourage the global item embedding to be similar to the corresponding cluster centroids. The embedding of the cluster centroids is $\boldsymbol{v}_{\boldsymbol{c}_i} = \frac{1}{|\mathcal{Z}_{\boldsymbol{c}_i}|} \Sigma_{z\in\mathcal{Z}_{\boldsymbol{c}_i}} z$, where $\mathcal{Z}_{\boldsymbol{c}_i}$ represents the set of items assigned to cluster $\boldsymbol{c}_i$. Now we construct the matrix $\boldsymbol{V} \in \mathbb{R}^{n\times d}$, where each row is assigned with the embedding of the corresponding cluster centroid, such that the i-th row of $\boldsymbol{V}$ is $\boldsymbol{v}_{\boldsymbol{c}_i}$. Then we perform the sample-wise enhancement by:

$$\mathcal{L}_n = \mathrm{MCE}\left(\frac{1}{d} \boldsymbol{Z_g}^t \boldsymbol{H}_d \boldsymbol{Z_g}^{t\top}, \frac{1}{d} \boldsymbol{V} \boldsymbol{H}_d \boldsymbol{V}^\top\right), \tag{16}$$

which aligns each item embedding with its corresponding centroids. This alignment encourages items that belong to the same cluster to be positioned closer in the embedding space, thereby preserving the relational structure and improving the representational capacity.

With the exploration of the global item embedding space from both the dimension and sample views, we combine them into the aggregation process, i.e.:

$$\boldsymbol{Z}_g^{t+1} = \boldsymbol{Z}_g^t - \eta\left(\frac{1}{\eta K} \sum_{k=1}^{K} \Delta \boldsymbol{Z}_{g_k}^t + \mu_d \nabla \mathcal{L}_d + \mu_n \nabla \mathcal{L}_n\right). \tag{17}$$

Here, μ_d and μ_n are hyperparameters that control the influence of the dimension-view and sample-view exploration on the global item embedding update. This optimization process can be performed using a standard stochastic gradient descent (SGD) with a learning rate η, updating the global item embedding based on the gradients of both the dimension-view and sample-view losses. By minimizing these losses, the module reduces redundancy between dimensions while simultaneously capturing the latent relationships between items, thereby enhancing the exploration of the embedding space. The enhanced global item embedding is then transmitted to all clients for the next communication round, further improving the item representations in subsequent updates.

3.4 Adaptive Local-Global Fusion

Motivation. Existing methods have attempted to integrate local and global information into personalized recommendations. However, the simple combination of item embeddings and the insufficient modeling of their relative importance results in suboptimal personalization. To address this, we propose adaptive gate control, implemented through a gate network that dynamically assigns personalized weights to local and global item embeddings. By adaptively combining these two types of information, the gate network outputs a fused item embedding, which ensures a more customized recommendation experience for individual users. To further enhance the effectiveness of the gate network, we introduce a personalized hypernetwork that generates gate network parameters based on user embeddings. This allows the gate network to adjust its behavior for different users, making the fusion process contextually relevant and achieving a personalized optimum for recommendation.

Adaptive Gate Control. Each client k holds a local item embedding $\boldsymbol{Z}_k^t$ and receives the aggregated global item embedding $\boldsymbol{Z}_g^t$ from the server at communication round t. The global item embedding $\boldsymbol{Z}_g^t$ optimized in client k is denoted as $\boldsymbol{Z}_{g_k}^t$. These embeddings are trained to provide complementary perspectives for the items, with $\boldsymbol{Z}_k^t$ reflecting user-specific preferences and $\boldsymbol{Z}_{g_k}^t$ capturing general item representations across all users. To combine these embeddings effectively, we introduce an adaptive gate control that fuses the local and global embeddings. The fused item embedding $\hat{\boldsymbol{z}}_i$ is expressed as a weighted combination:

$$\hat{\boldsymbol{z}}_i = \lambda_{k,i}^t \boldsymbol{Z}_{k,i}^t + (1 - \lambda_{k,i}^t) \boldsymbol{Z}_{g_k,i}^t, \tag{18}$$

where $\lambda_{k,i}^t \in [0, 1]$ serves as a control parameter that dynamically adjusts the contribution of each embedding. The value of $\lambda_{k,i}^t$ is computed adaptively using a gate network $\mathcal{G}$ parameterized by $\boldsymbol{\Phi}_k$. The gate network takes the concatenation of the local and global embeddings $[\boldsymbol{Z}_{k,i}||\boldsymbol{Z}_{g,i}]$ as input and outputs a value that determines

Algorithm 1 Training procedure of FedIAR

Input: communication rounds T, number of clients K, learning rate η, weight of dimension-view exploration μ_d, weight of sample-view exploration μ_n

Output: global item embedding Z_g^T

1: Server initializes Z_g^0
2: Each client k initializes $Z_k^0, \boldsymbol{u}_k^0, \Theta_k^0, \Psi_k^0$
3: **for** round $t = 0, 1, ..., T-1$ **do**
4:   **for** client $k = 1, 2, ..., K$ **in parallel do**
5:     $Z_{g_k}^{t+1} \leftarrow$ **Client executes**(k, Z_g^t)
6:   **end for**
7:   Dimension-view exploration $\mathcal{L}_d$ by Eq. (5)
8:   Sample-view exploration $\mathcal{L}_n$ by Eq. (16)
9:   $Z_g^{t+1} = Z_g^t - \eta \left(\frac{1}{\eta K} \sum_{k=1}^{K} \Delta Z_{g_k}^t + \mu_d \nabla \mathcal{L}_d + \mu_n \nabla \mathcal{L}_n \right)$
10: **end for**
11:
12: **Client executes**(k, Z_g^t):
13: Receive updated global item embedding $Z_{g_k}^t \leftarrow Z_g^t$
14: **for** each local epoch $e = 1, 2, \cdots, E$ **do**
15:   **for** batch of interaction $(k, i) \in \mathcal{D}_k$ **do**
16:     Calculate gate network parameters $\Phi_k^e = \mathcal{H}(\boldsymbol{u}_k^e; \Psi_k^e)$
17:     Calculate the adaptive control $\lambda_{k,i}$ by Eq. (19)
18:     Get fused item embedding $\hat{\boldsymbol{z}}_i$ by Eq. (18)
19:     Compute $\mathcal{L}_{\text{REC}}$ by Eq. (20), update $Z_{g_k}^e, Z_k^e, \boldsymbol{u}_k^e, \Theta_k^e, \Psi_k^e$
20:   **end for**
21: **end for**
22: **return** $Z_{g_k}^E$ to server

the relative importance of the local and global embeddings:

$$\lambda_{k,i}^t = \mathcal{G}\left(\left[Z_{k,i}^t || Z_{g_k,i}^t \right]; \Phi_k \right). \tag{19}$$

This adaptive control explicitly models the interplay between local and global information, dynamically adjusting their contributions to achieve a personalized optimum for each user. For each local interaction, represented as (k, i, r_{ki}), where i denotes the i-th item and r_{ki} is the observed rating, a score function $\mathcal{S}$ with parameters Θ_k predicts the rating based on the fused item embedding $\hat{\boldsymbol{z}}_i$. The prediction process is optimized by minimizing the binary cross-entropy loss:

$$\begin{aligned} \mathcal{L}_{\text{REC}} = - \sum_{(k,i,r_{ki})} \Big[& r_{ki} \log \mathcal{S}\left(\boldsymbol{u}_k, \hat{\boldsymbol{z}}_i; \Theta_k\right) \\ & + (1 - r_{ki}) \log\left(1 - \mathcal{S}\left(\boldsymbol{u}_k, \hat{\boldsymbol{z}}_i; \Theta_{\boldsymbol{k}}\right)\right) \Big], \end{aligned} \tag{20}$$

where $\boldsymbol{u}_k$ is the unique user embedding maintained and trained locally for each client, which serves as a compact representation of individual user preferences. By combining local and global embeddings through the gate network, an adaptive control between personalized and generalized recommendations is achieved. This adaptive fusion ensures that the client-side recommendation leverages the effective collaboration between local and global contributions.

Personalized Hypernetwork. To further enhance the flexibility and personalization of the adaptive gate control, we introduce a hypernetwork to discover the personalized optimum. Hypernetworks [16] are specialized networks that generate the weights for another network, enabling it to perform specific tasks. By leveraging this approach, the gate network can adjust its behavior based on user-specific embeddings $\boldsymbol{u}_k$, ensuring that the fusion of local and global information is tuned to individual user preferences and contextual needs. In this framework, the hypernetwork, denoted as $\mathcal{H}$, takes the user embedding $\boldsymbol{u}_k$ as input and generates the parameters Φ_k for the gate network:

$$\Phi_k = \mathcal{H}(\boldsymbol{u}_k; \Psi_k), \tag{21}$$

where Ψ_k represents the trainable parameters of the hypernetwork. The dynamic generation of parameters ensures that the gate network $\mathcal{G}$ is finely tuned for each user, allowing precise control over the fusion of local and global embeddings, and ultimately guiding the recommendation process toward a personalized optimum. The optimization of the hypernetwork is seamlessly integrated into the overall training process. Specifically, the parameters Ψ_k of the hypernetwork are updated to minimize the recommendation loss, as the generated parameters have a direct impact on the performance of the gate network. Thus, the optimal parameters Ψ_k^* are found by solving:

$$\Psi_k^* = \text{argmin}_{\Psi_k} \mathcal{L}_{\text{REC}}. \tag{22}$$

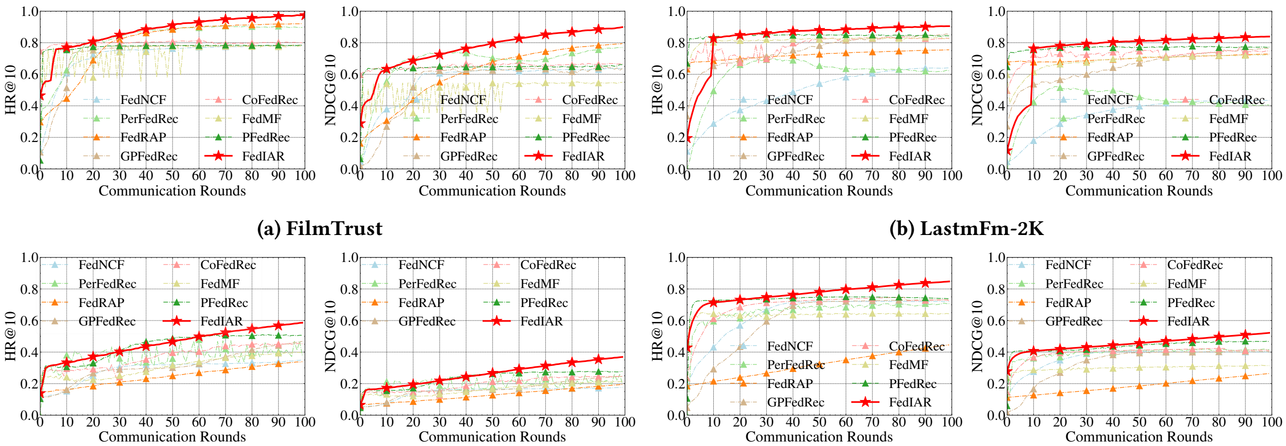
For practical implementation, we begin with a warm-up phase. In the initial few communication rounds, the local and global embeddings are set to be identical, ensuring a stable start. Afterward, the local embedding remains specific to each client, while the global embedding stays aggregated, maintaining separate local and global information. This gradual adjustment enhances the overall effectiveness of the fusion process. With the integration of the hypernetwork, the adaptive gate control becomes highly personalized, tailoring the gate network to the unique needs of each user. This enables effective collaboration between local and global information, leading to a personalized optimum for more accurate recommendations.

The overall algorithm of FedIAR is provided in Alg. 1, where lines 1-10 are the main process for the communication between server and clients. Lines 7-9 refer to the item embedding dual-view exploration module on the server, which enhances the global item embedding from dimension-view redundancy reduction and sample-view clustering enhancement, addressing **CH1**. Lines 12-22 refer to the adaptive local-global fusion module on the client, where each client leverages a hypernetwork with user embeddings as input to generate the gate network parameters, enabling the calculation of adaptive fusion weights for the local and global item embeddings, addressing **CH2**.

4 Experiments

4.1 Experimental Setup

Datasets. We conduct experiments on six widely used recommendation datasets, adapting them to federated learning scenarios. In this setup, each user acts as a client, maintaining their interaction records locally. Specifically, FilmTrust [8] is a movie recommendation dataset. Lastfm-2K [3] is a music dataset, where each user retains a list of listened artists along with their listening frequency. Foursquare [44] is a location-based dataset that captures user check-ins at various venues. Yelp [32] is a benchmark dataset derived from the 2018 edition of the Yelp challenge. Amazon-Video [38] consists



(a) FilmTrust **(b) LastmFm-2K**

(c) Yelp **(d) QB-article**

Figure 2: Comparison of model convergence.

Table 1: Dataset statistics.

Datasets	# Users (# Clients)	# Items	# Interactions	Sparsity
Filmtrust	1,227	2,059	34,889	98.62%
Lastfm-2K	1,600	12,454	185,650	99.07%
Foursquare	1,083	38,333	227,428	99.45%
Yelp	5,224	7,690	12,3024	99.69%
Amazon-Video	8,072	11,830	63,836	99.93%
QB-article	24,516	7,355	348,736	99.81%

of product reviews and metadata collected from the Amazon website. QB-article [47] records user click behaviors on articles. We remove the users with less than 5 interactions. The overall dataset statistics are shown in Table. 1.

Baselines. To assess FedIAR and demonstrate its superiority, we choose the following algorithms as baselines:

- **FedNCF** [39]: This is the federated adaptation of neural collaborative filtering [11]. It updates user embedding locally on each client and synchronizes the item embedding globally on the server.
- **FedMF** [4]: It applies matrix factorization in a federated learning environment to prevent information leakage by encrypting gradients of both user and item embeddings.
- **PerFedRec** [32]: It integrates a federated GNN for joint representation learning, user clustering, and model adaptation.
- **PFedRec** [49]: It incorporates the user embeddings into the score function and performs local adaptation to generate personalized item representations for each user.
- **FedRAP** [19]: It learns a global view of items via FL and a personalized view locally on each user, and enforces the two views to be complementary.
- **CoFedRec** [12]: This approach employs a co-clustering strategy, first grouping items into clusters and then clustering users based on their preferences for specific item categories.
- **GPFedRec** [50]: It constructs a user relationship graph using the received item embeddings and learns user-specific item embeddings through graph-guided aggregation.

We refer to the open-sourced code provided by these methods, adopt the recommended hyperparameters, and tune them accordingly.

Implementation Details. Following [49], we randomly sample $N = 4$ negative instances for each positive sample during training. The local epoch is set to 1 and the total communication round is set to 100. We set both the user and item embedding sizes to 32, with the batch size fixed at 256. For validation and testing, we employ the leave-one-out strategy. Model performance is evaluated using Top-K evaluation metrics [10], including Hit Ratio (HR) and Normalized Discounted Cumulative Gain (NDCG). For each user, we randomly select 99 unobserved items and perform a ranking evaluation among 100 items, including the test item. The test results are reported at the epoch with the best validation performance.

4.2 Performance Comparison

Main Results. Table. 2 shows the average performance results and standard deviation obtained from five runs with different seeds. We have the following observations and insights: (1) Directly combining centralized recommendation methods with federated learning, where model parameters are shared directly, does not lead to optimal results. For instance, FedNCF performs poorly in most cases, as it only personalizes user embeddings while neglecting the personalization of item embeddings and model parameters. (2) Server-side global aggregation methods show varying performance across different datasets. For example, PerFedRec achieves good results in the Amazon-Video dataset, whereas in the Foursquare dataset, it underperforms out of expectation. This can be attributed to its heavy reliance on clustering results, which can be unreliable due to client sparsity and insufficient user representation. (3) Client-specific local adaptation methods achieve better performance in most cases, especially for PFedRec in low-sparsity datasets, e.g., Lastfm-2K. This is due to its ability to fine-tune item embeddings using local interaction data. Nevertheless, single-round adaptation fails to effectively preserve personalized local information and does not consistently achieve runner-up in other datasets. (4) The method maintaining separate representations of local and global information gains significant improvements in small datasets, i.e., Filmtrust and Foursquare, but does not achieve consistent improvements in larger datasets.

Table 2: Comparison of federated recommendation performance.

Datasets	Filmtrust				Lastfm-2K			
Methods	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10
FedNCF	66.67±0.30	53.09±0.36	69.49±0.17	54.20±0.39	52.08±0.74	38.98±0.28	63.85±0.49	42.82±0.26
FedMF	65.23±0.30	47.83±0.35	69.38±0.45	49.21±0.40	73.10±0.28	65.02±0.16	78.63±0.18	66.71±0.23
PerFedRec	81.77±1.64	66.35±1.67	90.52±0.91	76.95±1.71	45.94±1.07	37.83±2.13	63.96±3.70	44.24±3.79
PFedRec	66.40±0.33	54.30±0.09	78.76±0.10	64.83±0.36	76.40±0.45	70.59±0.21	81.35±0.62	72.34±0.12
FedRAP	87.83±0.77	73.80±1.73	91.74±0.30	77.67±1.29	68.33±0.91	64.55±0.69	75.71±0.31	71.53±0.24
CoFedRec	68.87±0.35	55.58±0.53	81.26±0.24	67.29±0.09	73.85±0.94	68.68±0.55	77.23±1.23	68.96±0.97
GPFedRec	66.56±0.57	53.32±0.79	78.57±0.13	64.07±0.21	70.92±1.58	61.41±2.57	80.35±0.40	70.70±0.71
FedIAR	**92.26±0.12**	**84.55±0.08**	**95.63±0.08**	**85.64±0.14**	**83.88±0.67**	**79.02±0.24**	**87.88±0.81**	**80.42±0.31**
Datasets	Foursquare				Yelp			
Methods	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10
FedNCF	37.18±0.89	26.12±0.59	48.94±2.82	29.81±1.17	18.94±0.52	12.04±0.41	31.30±0.88	16.00±0.57
FedMF	74.27±0.41	69.92±0.12	76.30±0.70	70.68±0.40	23.95±0.31	15.54±0.24	37.92±0.37	20.03±0.24
PerFedRec	22.62±0.64	17.09±0.56	49.25±0.56	29.41±0.79	29.57±3.71	19.68±2.52	44.77±2.59	24.57±1.63
PFedRec	75.90±0.66	72.82±0.62	80.33±0.35	74.91±0.10	31.07±1.01	20.52±0.74	47.05±0.88	26.03±1.13
FedRAP	77.13±0.74	73.57±0.55	78.39±0.38	72.89±0.19	21.31±0.18	14.86±0.14	32.91±0.33	18.59±0.03
CoFedRec	75.41±0.48	72.70±0.70	79.35±0.48	74.68±0.13	26.74±1.06	17.74±0.74	42.20±1.12	22.83±0.69
GPFedRec	70.08±5.19	66.90±5.12	73.68±2.80	67.44±4.15	23.60±2.96	15.38±1.78	36.98±4.42	19.65±2.21
FedIAR	**81.35±0.39**	**78.39±0.33**	**86.24±0.53**	**79.95±0.38**	**41.52±1.42**	**30.30±0.86**	**56.47±2.26**	**35.05±1.13**
Datasets	Amazon-Video				QB-article			
Methods	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10
FedNCF	48.79±1.09	35.04±1.38	60.88±0.34	38.86±1.30	32.58±0.04	19.79±0.31	54.11±0.32	26.56±0.40
FedMF	38.54±1.87	23.74±0.19	45.57±2.17	26.02±0.24	30.72±0.52	16.40±0.29	49.64±0.38	22.51±0.17
PerFedRec	45.82±2.76	33.90±2.63	59.54±1.55	40.07±0.98	36.42±0.37	23.09±0.27	55.02±0.73	29.19±0.34
PFedRec	46.73±0.32	34.04±0.16	59.59±0.28	38.13±0.06	41.65±0.34	27.51±0.24	58.87±0.34	32.86±0.26
FedRAP	31.90±1.12	25.16±0.75	43.88±1.14	31.47±0.93	20.68±0.72	14.47±0.51	41.57±0.53	24.38±0.36
CoFedRec	48.24±0.29	35.11±0.29	60.82±0.20	40.50±0.20	35.68±0.36	21.96±0.21	56.02±1.06	28.06±0.67
GPFedRec	45.66±1.37	32.96±1.06	57.26±2.17	36.74±1.19	34.53±1.23	21.38±1.42	55.20±0.23	27.97±0.92
FedIAR	**50.91±1.53**	**39.15±1.25**	**63.59±0.52**	**42.66±0.93**	**49.33±0.38**	**30.91±0.22**	**69.81±1.24**	**37.59±0.50**

Table 3: Ablation Study

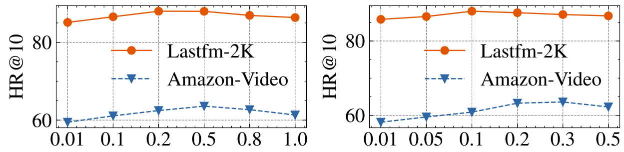
Lastfm-2K	HR@5	NDCG@5	HR@10	NDCG@10
FedIAR-w/o-IEDE	80.25±1.06	76.25±0.60	84.25±1.10	78.54±0.42
FedIAR-w/o-ALGF	73.71±0.46	69.56±0.36	79.19±0.59	71.32±0.32
FedIAR	**83.88±0.67**	**79.02±0.24**	**87.88±0.81**	**80.42±0.31**
Yelp	HR@5	NDCG@5	HR@10	NDCG@10
FedIAR-w/o-IEDE	39.34±1.57	28.44±0.87	54.31±2.15	33.61±1.06
FedIAR-w/o-ALGF	36.01±0.50	25.39±0.20	49.81±0.39	29.32±0.66
FedIAR	**41.52±1.42**	**30.30±0.86**	**56.47±2.26**	**35.05±1.13**
QB-article	HR@5	NDCG@5	HR@10	NDCG@10
FedIAR-w/o-IEDE	45.19±0.16	27.98±0.13	66.54±1.06	34.93±0.42
FedIAR-w/o-ALGF	42.01±0.25	24.13±0.14	62.79±1.01	31.66±0.39
FedIAR	**49.33±0.38**	**30.91±0.22**	**69.81±1.24**	**37.59±0.50**

This suggests that directly combining local and global information is useful in less complex scenarios, but in larger-scale datasets, a more refined and effective approach to collaborating between local and global information is required for better performance. (5) FedIAR consistently outperforms all other methods across various scenarios. Notably, the performance improvements become more significant with larger datasets. In the largest dataset, QB-article, the improvement in HR@10 reaches 18.58%, which verifies the effectiveness of FedIAR with global item embedding enhancement and effective local and global collaboration.

Convergence Comparison. Figure. 2 provides a detailed comparison of the convergence behavior between FedIAR and the baseline methods. The figure presents the validation results of HR@10 and NDCG@10 across four different datasets, offering a clear illustration of how each method evolves during training. Client-specific model adaptation methods, e.g., PFedRec, show a significant initial improvement due to fine-tuning on local data, enabling them to quickly capture client-specific preferences. However, this initial boost is followed by a noticeable slowdown in performance growth over time, indicating that relying solely on local adaptation is not sufficient for achieving long-term improvement. Meanwhile, CoFedRec, which uses clustering for both items and users, exhibits a more fluctuating convergence curve. This variability highlights the inherent challenges in obtaining stable and reliable clustering results. In comparison, FedIAR consistently outperforms the others. With its enhanced global item embedding and the effective collaboration between the global and local item embeddings, FedIAR maintains stable and superior performance throughout the training process.

4.3 In-depth Analysis

Ablation Study. To evaluate the effectiveness of each module in FedIAR, we conduct an ablation study using three datasets, Lastfm-2K, Yelp, and QB-article, as shown in Table 3. We assess two versions



(a) Dimension-view μ_d **(b) Sample-view μ_n**

Figure 3: Effect of hyperparameters.

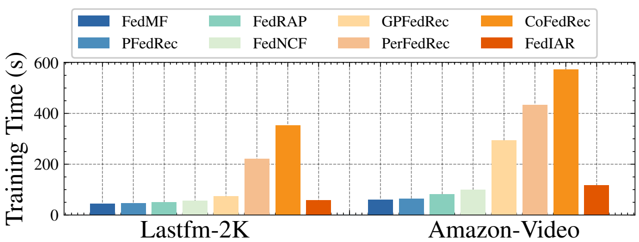


Figure 4: Comparison of training efficiency.

of FedIAR for comparison. The first version, FedIAR-w/o-IEDE, excludes the item embedding dual-view exploration module, which aggregates the global item embedding using the conventional federated averaging aggregation process. This helps evaluate the impact of enhancing the global item embedding by considering both the dimensional and sample perspectives. The second version, FedIAR-w/o-ALGF, removes the adaptive local-global fusion module, which simply adds the local and global item embeddings with equal weight. This version verifies the importance of better collaboration by finding the personalized optimum. The results in Table. 3 clearly demonstrate the effectiveness of each module, validating their contributions to the overall performance. Notably, FedIAR-w/o-ALGF suffers from a more significant performance drop, highlighting the critical role of effective local and global collaboration.

Hyperparameters Sensitivity. To investigate the sensitivity of various hyper-parameters, we conduct experiments on Lastfm-2K and Amazon. We tune the weight of dimension-view redundancy reduction $\mu_d = \{0.1, 0.2, 0.5, 0.8, 1.0\}$ and sample-view clustering enhancement $\mu_n = \{0.1, 0.2, 0.5, 0.8, 1.0\}$. The results are shown in Figure 3. Both hyperparameters exhibit a bell curve, with different optimal values for different datasets. This suggests that a balanced value is preferred for exploring the global embedding space. A larger value may cause the global item embedding to deviate too much from the global aggregated embedding, while a smaller value may not produce sufficient impact.

Training efficiency. We analyze the training efficiency of FedIAR in terms of communication and computation costs. (1) For *communication cost*, FedIAR only transmits the gradient of the updated global item embeddings to the server and receives the enhanced global item embeddings in return. This results in the same communication cost as the baselines. (2) For *computation cost*, we analyze the complexity of the two modules in FedIAR. The item embedding dual-view exploration module involves operations in both the dimension and sample views. Specifically, dimension-view redundancy reduction has a complexity of $O(n \cdot d^2)$ for centering and autocovariance computation, and $O(d^3)$ for the MCE computation. The sample-view clustering enhancement costs $O(n \cdot M)$ for the variational inference step, and $O(n^2 \cdot d + M \cdot d^2)$ for the alignment computation. The aggregation step, which involves gradient updates, has a complexity of $O(K \cdot n \cdot d + n \cdot d^2)$. Overall, these operations are manageable when executed on a server, ensuring that the computational burden remains within reasonable limits. For the adaptive local-global fusion module, both the hypernetwork and gate network are implemented as small MLPs. They can be optimized together in a single backward pass for the recommendation loss, introducing minimal computational overhead. As shown in Figure 4, we conducted experiments on Lastfm-2k and Amazon-video, which demonstrate the average time cost per communication round. The results indicate that the training time of FedIAR is comparable to most baselines and significantly lower than GPFedRec, PerFedRec, and CoFedRec, which involve more complex operations.

Table 4: Performance of integrating LDP with varying Laplacian noise strength.

Datasets	Noise Strength	$\lambda = 0$	$\lambda = 0.1$	$\lambda = 0.2$	$\lambda = 0.3$	$\lambda = 0.4$
Lastfm-2K	HR@10	87.88	87.81	87.31	86.93	86.69
	NDCG@10	80.42	79.90	79.35	78.93	78.70
Yelp	HR@10	56.47	56.15	56.06	55.81	55.76
	NDCG@10	35.05	34.89	34.86	34.67	34.61
QB-article	HR@10	68.91	68.80	68.67	68.63	68.49
	NDCG@10	37.59	37.48	37.36	37.28	37.14

Privacy Protection. To enhance user privacy protection, we integrate the Local Differential Privacy (LDP) technique [1]. Following existing FedRec practices [49, 50], we introduce zero-mean Laplacian noise to the gradients of the item embeddings, expressed as $\Delta Z_{g_k}^t = {Z_g}^t - Z_k^{t+1} + Laplace(0, \lambda)$, where λ denotes the strength of the noise. We evaluate the impact of different noise strengths $\lambda \in \{0.1, 0.2, 0.3, 0.4\}$ on Lastfm-2K, Yelp and QB-article, reporting the HR@10 and NDCG@10 metrics. The results in Table 4 show that increasing the noise strength leads to a slight decrease in the performance of FedIAR, but the effect is minimal. Even with $\lambda = 0.4$, FedIAR still outperforms all baselines by a significant margin.

5 Conclusion

In this paper, we propose a federated recommendation framework FedIAR which contains two modules, i.e., item embedding dual-view exploration and adaptive local-global fusion. The first module on the server sufficiently explores the global item embedding space from dimension-view redundancy reduction and sample-view clustering enhancement. The second module on each client achieves effective local and global collaboration by leveraging an adaptive gate control mechanism, which calculates personalized fusion weights for the local and global item embeddings based on user-specific representation. Extensive experiments on six datasets demonstrate the effectiveness of FedIAR.

Acknowledgments

This research was supported by Zhejiang Provincial Natural Science Foundation of China under Grant No. LZYQ25F020002, Zhejiang University International Business School (ZIBS) and Zhejiang International Science and Technology Collaboration Center for FinTech and Big Data Analysis.

References

[1] Pathum Chamikara Mahawaga Arachchige, Peter Bertok, Ibrahim Khalil, Dongxi Liu, Seyit Camtepe, and Mohammed Atiquzzaman. 2019. Local differential privacy for deep learning. *IEEE Internet of Things Journal* 7, 7 (2019), 5827–5842.

[2] DM BLEI. 2006. Variational inference for Dirichlet process mixtures. *International Society for Bayesian Analysis* 1 (2006), 121–144.

[3] Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik. 2011. Second workshop on information heterogeneity and fusion in recommender systems (HetRec2011). In *Proceedings of the fifth ACM conference on Recommender systems*. 387–388.

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

[5] Daoyuan Chen, Dawei Gao, Yuexiang Xie, Xuchen Pan, Zitao Li, Yaliang Li, Bolin Ding, and Jingren Zhou. 2023. Fs-real: Towards real-world cross-device federated learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3829–3841.

[6] Zihan Chen, Howard Hao Yang, Tony Quek, and Kai Fong Ernest Chong. 2023. Spectral Co-Distillation for Personalized Federated Learning. In *Thirty-seventh Conference on Neural Information Processing Systems*. https://openreview.net/forum?id=RqjQL08UFc

[7] Amine Echraibi, Joachim Flocon-Cholet, Stéphane Gosselin, and Sandrine Vaton. 2020. On the variational posterior of dirichlet process deep latent gaussian mixture models. *arXiv preprint arXiv:2006.08993* (2020).

[8] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. 2013. A novel Bayesian similarity measure for recommender systems. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence* (Beijing, China) *(IJCAI '13)*. AAAI Press, 2619–2625.

[9] Yongxin Guo, Xiaoying Tang, and Tao Lin. 2024. FedRC: Tackling Diverse Distribution Shifts Challenge in Federated Learning by Robust Clustering. In *Forty-first International Conference on Machine Learning*. https://openreview.net/forum?id=kc4dZYJlJG

[10] Xiangnan He, Tao Chen, Min-Yen Kan, and Xiao Chen. 2015. Trirank: Review-aware explainable recommendation by modeling aspects. In *Proceedings of the 24th ACM international on conference on information and knowledge management*. 1661–1670.

[11] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*. 173–182.

[12] Xinrui He, Shuo Liu, Jacky Keung, and Jingrui He. 2024. Co-clustering for Federated Recommender System. In *Proceedings of the ACM on Web Conference 2024*. 3821–3832.

[13] Jing Jiang, Chunxu Zhang, Honglei Zhang, Zhiwei Li, Yidong Li, and Bo Yang. 2024. A Tutorial of Personalized Federated Recommender Systems: Recent Advances and Future Directions. *arXiv preprint arXiv:2412.08071* (2024).

[14] Meirui Jiang, Anjie Le, Xiaoxiao Li, and Qi Dou. 2024. Heterogeneous Personalized Federated Learning by Local-Global Updates Mixing via Convergence Rate. In *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=7pWRLDBAtc

[15] Heasung Kim, Hyeji Kim, and Gustavo De Veciana. [n. d.]. Clustered Federated Learning via Gradient-based Partitioning. In *Forty-first International Conference on Machine Learning*.

[16] Tao Kong, Anbang Yao, Yurong Chen, and Fuchun Sun. 2016. Hypernet: Towards accurate region proposal generation and joint object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 845–853.

[17] Yuyuan Li, Chaochao Chen, Yizhao Zhang, Weiming Liu, Lingjuan Lyu, Xiaolin Zheng, Dan Meng, and Jun Wang. 2023. Ultrare: Enhancing receraser for recommendation unlearning via error decomposition. *Advances in Neural Information Processing Systems* 36 (2023), 12611–12625.

[18] Zilinghan Li, Shilan He, Ze Yang, Minseok Ryu, Kibaek Kim, and Ravi Madduri. 2024. Advances in appfl: A comprehensive and extensible federated learning framework. *arXiv preprint arXiv:2409.11585* (2024).

[19] Zhiwei Li, Guodong Long, and Tianyi Zhou. 2024. Federated Recommendation with Additive Personalization. In *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=xkXdE81mOK

[20] Zhiwei Li, Guodong Long, Tianyi Zhou, Jing Jiang, and Chengqi Zhang. 2024. Personalized Federated Collaborative Filtering: A Variational AutoEncoder Approach. *arXiv preprint arXiv:2408.08931* (2024).

[21] Xinting Liao, Chaochao Chen, Weiming Liu, Pengyang Zhou, Huabin Zhu, Shuheng Shen, Weiqiang Wang, Mengling Hu, Yanchao Tan, and Xiaolin Zheng. 2023. Joint local relational augmentation and global nash equilibrium for federated learning with non-iid data. In *Proceedings of the 31st ACM International Conference on Multimedia*. 1536–1545.

[22] Xinting Liao, Weiming Liu, Chaochao Chen, Pengyang Zhou, Fengyuan Yu, Huabin Zhu, Binhui Yao, Tao Wang, Xiaolin Zheng, and Yanchao Tan. 2024. Rethinking the Representation in Federated Unsupervised Learning with Non-IID Data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 22841–22850.

[23] Xinting Liao, Weiming Liu, Chaochao Chen, Pengyang Zhou, Huabin Zhu, Yanchao Tan, Jun Wang, and Yue Qi. 2023. HyperFed: Hyperbolic Prototypes Exploration with Consistent Aggregation for Non-IID Data in Federated Learning. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, Edith Elkind (Ed.). International Joint Conferences on Artificial Intelligence Organization, 3957–3965. doi:10.24963/ijcai.2023/440 Main Track.

[24] Xinting Liao, Weiming Liu, Pengyang Zhou, Fengyuan Yu, Jiahe Xu, Jun Wang, Wenjie Wang, Chaochao Chen, and Xiaolin Zheng. 2024. FOOGD: Federated Collaboration for Both Out-of-distribution Generalization and Detection. *Advances in Neural Information Processing Systems* 37 (2024), 132908–132945.

[25] Weiming Liu, Chaochao Chen, Xinting Liao, Mengling Hu, Jianwei Yin, Yanchao Tan, and Longfei Zheng. 2023. Federated Probabilistic Preference Distribution Modelling with Compactness Co-Clustering for Privacy-Preserving Multi-Domain Recommendation.. In *IJCAI*. 2206–2214.

[26] Weiming Liu, Xiaolin Zheng, Chaochao Chen, Jiajie Su, Xinting Liao, Mengling Hu, and Yanchao Tan. 2023. Joint internal multi-interest exploration and external domain alignment for cross domain sequential recommendation. In *Proceedings of the ACM web conference 2023*. 383–394.

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

[28] Weiming Liu, Xiaolin Zheng, Jiajie Su, Mengling Hu, Yanchao Tan, and Chaochao Chen. 2022. Exploiting variational domain-invariant user embedding for partially overlapped cross domain recommendation. In *Proceedings of the 45th International ACM SIGIR conference on research and development in information retrieval*. 312–321.

[29] Weiming Liu, Xiaolin Zheng, Jiajie Su, Longfei Zheng, Chaochao Chen, and Mengling Hu. 2023. Contrastive proxy kernel stein path alignment for cross-domain cold-start recommendation. *IEEE Transactions on Knowledge and Data Engineering* 35, 11 (2023), 11216–11230.

[30] Xin Liu, Zhongdao Wang, Ya-Li Li, and Shengjin Wang. 2022. Self-supervised learning via maximum entropy coding. *Advances in Neural Information Processing Systems* 35 (2022), 34091–34105.

[31] Yue Liu, Shihao Zhu, Jun Xia, Yingwei Ma, Jian Ma, Wenliang Zhong, Xinwang Liu, Shengju Yu, and Kejun Zhang. 2024. End-to-end Learnable Clustering for Intent Learning in Recommendation. In *Proc. of NeurIPS*.

[32] Sichun Luo, Yuanzhang Xiao, and Linqi Song. 2022. Personalized federated recommendation via joint representation learning, user clustering, and model adaptation. In *Proceedings of the 31st ACM international conference on information & knowledge management*. 4289–4293.

[33] Sichun Luo, Yuanzhang Xiao, Xinyi Zhang, Yang Liu, Wenbo Ding, and Linqi Song. 2024. Perfedrec++: Enhancing personalized federated recommendation with self-supervised pre-training. *ACM Transactions on Intelligent Systems and Technology* 15, 5 (2024), 1–24.

[34] Xingyuan Mao, Yuwen Liu, Lianyong Qi, Li Duan, Xiaolong Xu, Xuyun Zhang, Wanchun Dou, Amin Beheshti, and Xiaokang Zhou. 2024. Cluster-driven Personalized Federated Recommendation with Interest-aware Graph Convolution Network for Multimedia. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 5614–5622.

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

[36] Khalil Muhammad, Qinqin Wang, Diarmuid O'Reilly-Morgan, Elias Tragos, Barry Smyth, Neil Hurley, James Geraci, and Aonghus Lawlor. 2020. 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*. 1234–1242.

[37] Kevin P Murphy. 2007. Conjugate Bayesian analysis of the Gaussian distribution. *def* 1, $2\sigma 2$ (2007), 16.

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

[39] Vasileios Perifanis and Pavlos S Efraimidis. 2022. Federated neural collaborative filtering. *Knowledge-Based Systems* 242 (2022), 108441.

[40] Yiqing Shen, Yuyin Zhou, and Lequan Yu. 2022. Cd2-pfed: Cyclic distillation-guided channel decoupling for model personalization in federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10041–10050.

[41] Yee Whye Teh et al. 2010. Dirichlet process. *Encyclopedia of machine learning* 1063 (2010), 280–287.

[42] Paul Voigt and Axel Von dem Bussche. 2017. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing* 10, 3152676 (2017), 10–5555.

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

[44] Dingqi Yang, Daqing Zhang, Vincent W Zheng, and Zhiyong Yu. 2014. Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 45, 1 (2014), 129–142.

[45] Ze Yang, Yihong Jin, Yihan Zhang, Juntian Liu, and Xinhe Xu. 2025. Research on Large Language Model Cross-Cloud Privacy Protection and Collaborative Training based on Federated Learning. *arXiv preprint arXiv:2503.12226* (2025).

[46] Yaodong Yu, Kwan Ho Ryan Chan, Chong You, Chaobing Song, and Yi Ma. 2020. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. *Advances in neural information processing systems* 33 (2020), 9422–9434.

[47] Guanghu Yuan, Fajie Yuan, Yudong Li, Beibei Kong, Shujie Li, Lei Chen, Min Yang, Chenyun Yu, Bo Hu, Zang Li, et al. 2022. Tenrec: A large-scale multipurpose benchmark dataset for recommender systems. *Advances in Neural Information Processing Systems* 35 (2022), 11480–11493.

[48] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. 2021. Barlow twins: Self-supervised learning via redundancy reduction. In *International conference on machine learning*. PMLR, 12310–12320.

[49] Chunxu Zhang, Guodong Long, Tianyi Zhou, Peng Yan, Zijian Zhang, Chengqi Zhang, and Bo Yang. 2023. Dual Personalization on Federated Recommendation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, Edith Elkind (Ed.). International Joint Conferences on Artificial Intelligence Organization, 4558–4566. doi:10.24963/ijcai.2023/507 Main Track.

[50] Chunxu Zhang, Guodong Long, Tianyi Zhou, Zijian Zhang, Peng Yan, and Bo Yang. 2024. GPFedRec: Graph-Guided Personalization for Federated Recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Barcelona, Spain) *(KDD '24)*. Association for Computing Machinery, New York, NY, USA, 4131–4142. doi:10.1145/3637528.3671702

[51] Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. 2023. Fedala: Adaptive local aggregation for personalized federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 11237–11244.

[52] Yifan Zhang, Zhiquan Tan, Jingqin Yang, Weiran Huang, and Yang Yuan. [n. d.]. Matrix Information Theory for Self-Supervised Learning. In *Forty-first International Conference on Machine Learning*.

[53] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. 2018. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582* (2018).

[54] Pengyang Zhou, Chaochao Chen, Weiming Liu, Xinting Liao, Wenkai Shen, Jiahe Xu, Zhihui Fu, Jun Wang, Wu Wen, and Xiaolin Zheng. 2025. FedGOG: Federated Graph Out-of-Distribution Generalization with Diffusion Data Exploration and Latent Embedding Decorrelation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 22965–22973.

[55] Pengyang Zhou, Chaochao Chen, Weiming Liu, Xinting Liao, Fengyuan Yu, Zhihui Fu, Xingyu Lou, Wu Wen, Xiaolin Zheng, and Jun Wang. 2025. FedGF: Enhancing Structural Knowledge via Graph Factorization for Federated Graph Learning. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*. 448–456.

[56] Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. 2021. Data-free knowledge distillation for heterogeneous federated learning. In *International conference on machine learning*. PMLR, 12878–12889.