

Dynamic Rank-Aware Aggregation with Graph Contrastive Learning for Federated Foundation Model Fine-Tuning

Zhao Yang*, Xunyun Qiu[†], Hua Cui*,

*School of Future Transportation, Chang'an University, Xi'an, China

[†]Imperial College London, London, UK

Abstract—Foundation Models (FMs) achieve strong performance across natural language tasks but require task-specific adaptation. Federated Learning enables privacy-preserving fine-tuning, yet full-parameter updates at FM scale are costly. Federated Low-Rank Adaptation alleviates this by constraining updates to low-rank subspaces, reducing communication and storage overhead. However, heterogeneous and evolving client data distributions introduce inconsistencies in low-rank representations, causing unstable aggregation and degraded generalization. We propose a graph contrastive learning-enhanced dynamic aggregation strategy to address these challenges. Wasserstein distance is used to quantify distribution disparities, constructing a similarity graph that encodes potential knowledge-sharing relations. Graph Contrastive Learning then models dynamic feature embeddings at the server, capturing temporal shifts in distributions. A consistency-guided weighting mechanism further adapts client contributions during aggregation, suppressing conflicting updates and amplifying effective ones. Extensive experiments on diverse federated benchmarks verify the effectiveness of our approach, demonstrating improved stability, adaptability, and generalization compared to existing methods.

I. INTRODUCTION

Foundation Models (FMs) have demonstrated outstanding performance in tasks such as natural language processing [1], knowledge retrieval [2], and reasoning [3]. However, directly deploying general-purpose pre-trained models in domain-specific scenarios often fails to meet practical requirements, necessitating fine-tuning to improve adaptability and downstream performance [4]. Traditional centralized full-parameter fine-tuning, while effective, incurs prohibitive computation and storage costs and depends on large-scale centralized datasets, which are often incompatible with cross-institutional compliance and privacy constraints.

To alleviate these issues, Federated Learning (FL) has been introduced into FM adaptation [5]. FL enables multiple institutions to collaboratively train a global model without sharing raw data, offering a balance between utility and privacy. Nevertheless, applying full-parameter FL to FMs is still systemically inefficient, as it imposes extreme communication and storage demands on the participating infrastructure. To reduce this burden, Federated Low-Rank Adaptation (Fed-LoRA) [6], [7] has been proposed. By restricting model updates to a low-rank subspace [8], Fed-LoRA significantly compresses communication volume and storage overhead while retaining strong

adaptability. From a systems perspective, Fed-LoRA provides a scalable and resource-efficient paradigm for privacy-preserving FM training. However, its deployment in real-world distributed environments faces two major system-level challenges: data heterogeneity and temporal dynamics.

Data Heterogeneity. Clients participating in cross-institutional federated training often differ in domains, semantics, and data styles, which leads to non-independent and identically distributed (non-IID) data. These discrepancies result in inconsistent low-rank parameter structures, degrading the effectiveness of server-side aggregation, limiting knowledge transfer across participants, and sometimes even causing negative transfer. Under strict privacy constraints—where only low-rank parameter updates are observable—the system must provide mechanisms to quantify and mitigate such distributional differences in order to ensure stable global convergence.

Temporal Dynamics. In addition to static heterogeneity, client data distributions evolve over time due to shifting tasks or environments, driving continuous changes in low-rank feature representations. Static aggregation strategies cannot efficiently adapt to such drift, often causing system-level instability and degraded convergence. Moreover, temporal dynamics exacerbate the difficulty of aligning client updates across training rounds, increasing communication overhead and coordination complexity. From a systems design perspective, the core challenge lies in enabling dynamic and robust aggregation that can adaptively align with evolving client data distributions and amplify effective contributions.

To address these system challenges, this work proposes a graph contrastive learning-based dynamic aggregation strategy for Fed-LoRA. First, Wasserstein distance is used to measure distributional discrepancies among client updates, constructing a similarity graph that captures inter-client knowledge-sharing relationships. Then, Graph Contrastive Learning (GCL) is employed at the server to adaptively embed and model client relationships, dynamically reflecting temporal distribution shifts. Finally, a dynamic weight allocation mechanism is introduced to adjust each client's contribution to aggregation based on its consistency with the global model, thereby enhancing robustness and scalability in heterogeneous, evolving environments. The main contributions of this study are as follows:

- A low-rank feature modeling method is proposed, which

*Hua Cui is the corresponding author. Zhao Yang and Xunyun Qiu are co-first authors and contributed equally to this work.

constructs similarity relationships across clients to capture potential knowledge-sharing structures, providing an interpretable and scalable foundation for dynamic aggregation.

- A dynamic similarity modeling mechanism driven by GCL is introduced, where positive and negative sample contrastive learning is used at the server to model the similarity representations between clients, allowing the feature embeddings to adaptively reflect the dynamic changes in data distribution.
- A dynamic aggregation strategy based on low-rank consistency is designed, which adjusts the contribution of each client adaptively, enhancing the stability and generalization performance of the global model in heterogeneous and time-varying data environments.

II. RELATED WORKS

In the context of federated low-rank adaptation for FMs under heterogeneous data, existing studies can be broadly categorized into three directions:

Low-Rank dimension selection. Prior works aim to reduce communication overhead by constraining the tunable subspace. For example, [9] proposes FFA-LoRA, which updates only the zero-initialized B matrix while freezing the randomly initialized A matrix, trading efficiency for degraded performance [10]. To balance efficiency and adaptability, [11] introduces FedSA-LoRA, where both A and B matrices are trained locally but only A is shared for aggregation. More advanced approaches, such as FedARA [12], employ truncated singular value decomposition to increase structural flexibility and mitigate heterogeneity, while Fed-piLot [13] adaptively determines the low-rank dimension by considering both information gain and device memory constraints. These methods effectively reduce communication costs, but they remain limited to static optimization of low-rank dimensions without addressing temporal variation in client updates.

Heterogeneous aggregation optimization. Another line of work focuses on enabling aggregation across clients with different low-rank configurations. A simple approach is heterogeneous LoRA [14], which applies zero-padding to align different ranks before aggregation, but this often leads to unstable training and redundant parameters. Replication- and stacking-based strategies [15], [16] improve aggregation flexibility but at the cost of increased model redundancy. Weighted aggregation (RBLA [17]) and server-side correction (LoRA-FAIR [18]) further enhance accuracy, yet they require either careful parameter tuning or accurate correction at the server. FlexLoRA [19] transforms each client’s LoRA module into a local update, aggregates these updates into a global representation, and employs singular value decomposition (SVD) to refine the local LoRA parameters. While these methods tackle structural heterogeneity, they assume relatively stable distributions and cannot efficiently capture evolving client characteristics in dynamic environments.

Personalized Low-Rank fine-tuning. A third direction incorporates personalization into LoRA to balance global knowledge with client-specific adaptation. Methods such as FedDPA

[20] and FDLORA [21] introduce separate global and personalized LoRA modules, improving adaptability but increasing computation and storage overhead. Similarly, personalized joint task adjustment [22] leverages a global adapter with local LoRA modules to achieve collaborative fine-tuning, though its fixed interaction limits flexibility in non-stationary settings. These methods highlight the importance of personalization but still fall short of dynamically adapting to continuous distribution shifts.

In summary, existing approaches primarily target static heterogeneity, assuming relatively fixed data distributions and system conditions. However, in realistic federated deployments, temporal dynamics drive continuous evolution of client distributions, which exacerbates aggregation instability and increases communication and computation costs. Current methods are not designed to track such dynamics, making knowledge sharing across clients less effective. This motivates our work: to investigate the evolution of parameter features within low-rank subspaces and to design efficient, adaptive aggregation strategies that enhance both robustness and long-term effectiveness in dynamic, heterogeneous environments.

III. PROBLEM FORMULATION

We consider an FL setting consisting of a central server and a set of clients $\mathcal{C} = 1, 2, \dots, N$. Let θ_0 denote the parameters of a pre-trained FM. Under the LoRA framework, each client constrains its update for a weight matrix $W \in \mathbb{R}^{d \times k}$ within a low-rank subspace:

$$\Delta W_i = A_i B_i, \quad A_i \in \mathbb{R}^{d \times r}, B_i \in \mathbb{R}^{r \times k}, \quad r \ll \min(d, k). \quad (1)$$

Thus, the effective update of client i is represented as

$$\Delta \theta_i = \Delta W_i W_{i \in \mathcal{S}}, \quad \theta_i = \theta_0 + \Delta \theta_i, \quad (2)$$

where \mathcal{S} denotes the set of LoRA-adapted layers. This formulation ensures that the adaptation remains communication- and storage-efficient, since only low-rank modules need to be trained and transmitted.

Given its local dataset \mathcal{D}_i , client i minimizes the following local training objective:

$$\min_{A_i, B_i} \mathcal{L}_i(\theta_0 + \Delta \theta_i; \mathcal{D}_i) = \frac{1}{|\mathcal{D}_i|} \sum_{(x, y) \in \mathcal{D}_i} \ell(f(x; \theta_0 + \Delta \theta_i), y), \quad (3)$$

where $\ell(\cdot)$ denotes the task loss and $f(\cdot)$ the FM predictor.

On the server side, raw data is inaccessible due to privacy constraints, and only low-rank parameter updates $\Delta \theta_i$ are collected. To construct the global update at round t , the server performs a dynamically weighted aggregation:

$$\Delta \theta^{(t+1)} = \sum_{i=1}^N \alpha_i^{(t)} \Delta \theta_i^{(t)}, \quad (4)$$

where the aggregation weight $\alpha_i^{(t)}$ must adapt not only to cross-client heterogeneity in low-rank update distributions but also to their temporal evolution across training rounds. This formulation highlights the central challenge: designing

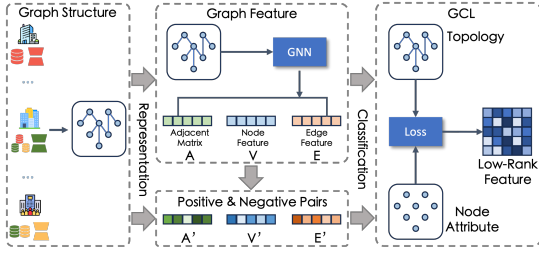


Fig. 1. Local device updates are represented as graph structures and processed by a GNN to obtain low-rank embeddings. GCL is then applied by constructing positive and negative pairs to capture device-level similarity under heterogeneous and time-varying conditions. The learned embeddings are used to compute dynamic aggregation weights, enabling adaptive and robust global model updates.

aggregation strategies that are both robust to heterogeneous updates and sensitive to temporal dynamics, ensuring stable convergence in real-world federated environments.

IV. LOW-RANK FEATURE REPRESENTATION WITH GCL

In the process of federated low-rank adaptation for FMs, the heterogeneity and temporal variation of local data distributions across devices result in significant differences in the uploaded low-rank updates:

$$\Delta\theta_i^{(t)} = \{\Delta W_{i,W}^{(t)} \mid W \in \mathcal{S}\}, \quad (5)$$

where each update matrix is factorized as $\Delta W_{i,W}^{(t)} = A_{i,W}^{(t)} B_{i,W}^{(t)}$ with rank $r_{i,W}$. This decomposition provides a set of rank-1 components $a_{i,W,k}^{(t)}(b_{i,W,k}^{(t)})^\top$, where $a_{i,W,k}^{(t)}$ and $b_{i,W,k}^{(t)}$ denote the k -th column of $A_{i,W}^{(t)}$ and row of $B_{i,W}^{(t)}$, respectively. These components naturally serve as low-rank feature vectors that capture the essential variations of client updates. Directly performing weighted aggregation on the server side fails to adequately capture both the similarity relationships among devices and the dynamic characteristics of low-rank updates over time, which undermines the adaptability and stability of the global model.

To effectively measure the heterogeneity in low-rank feature representations across devices, this study introduces GCL to construct a device-level low-rank feature similarity graph, based on which aggregation weights are dynamically adjusted. As illustrated in Fig. 1, compared with traditional low-rank similarity measures, GCL has distinct advantages: (1) device-specific low-rank features can be naturally modeled as a graph structure, where inter-device relations are explicitly captured by the graph topology, thereby enabling more fine-grained similarity modeling in the embedding space; (2) traditional distance-based metrics usually capture only static similarities and struggle with evolving distributions, whereas GCL leverages the construction of positive and negative pairs to adaptively learn device-level similarity, thus improving robustness under heterogeneous and time-varying environments.

To quantify the differences between devices, we represent the update of client i as a weighted empirical distribution over its rank-1 components:

$$P_i^{(t)} = \sum_{m=1}^{M_i} w_{i,m}^{(t)} \delta_{u_{i,m}^{(t)}}, \quad (6)$$

where $u_{i,m}^{(t)}$ is the normalized vectorized form of the m -th low-rank component extracted from $\Delta\theta_i^{(t)}$, and $w_{i,m}^{(t)}$ is proportional to its energy $\|a_{i,m}^{(t)}\|_2 \|b_{i,m}^{(t)}\|_2$. The distributional discrepancy between devices i and j is then measured via the Wasserstein distance:

$$D(P_i^{(t)}, P_j^{(t)}) = \min_{\Gamma \geq 0} \sum_{m,n} \Gamma_{m,n} c(u_{i,m}^{(t)}, u_{j,n}^{(t)}) \quad (7)$$

s.t. $\Gamma \mathbf{1} = b, \Gamma^\top \mathbf{1} = a.$

Where a and b are the marginal weights from $\{w_{i,m}^{(t)}\}$ and $\{w_{j,n}^{(t)}\}$, respectively, and $c(\cdot, \cdot)$ is the angular-based transport cost defined on normalized feature vectors. Unlike pairwise metrics such as Euclidean or cosine distance that only measure point-to-point similarity, the Wasserstein distance operates at the distributional level. In representation learning, it aligns entire feature distributions derived from LoRA components rather than individual samples, thereby providing a geometry-aware notion of similarity. This makes the Wasserstein distance robust to temporal variations and heterogeneous data sources commonly observed in FL systems. Based on this measure, the similarity weight is defined as:

$$S_{i,j}^{(t)} = \exp(-D(P_i^{(t)}, P_j^{(t)})), \quad (8)$$

from which we construct the similarity matrix $S^{(t)} = [S_{i,j}^{(t)}]$ and the device similarity graph $\mathcal{G}^{(t)} = (\mathcal{V}, \mathcal{E}, S^{(t)})$.

On this graph structure, a Graph Neural Network (GNN) is employed to learn device-level low-rank feature embeddings. Let $H^{(l,t)}$ denote the feature matrix at the l -th layer. The update rule is given by:

$$H^{(l+1,t)} = \sigma\left((D^{(t)})^{-1/2} S^{(t)} (D^{(t)})^{-1/2} H^{(l,t)} W^{(l)}\right), \quad (9)$$

where $D^{(t)}$ is the degree matrix, $W^{(l)}$ is the learnable parameter matrix, and $\sigma(\cdot)$ denotes a nonlinear activation function. Through multi-layer propagation, each device i obtains its final embedding representation $z_i^{(t)}$, which aggregates the low-rank features of neighboring devices and thus improves adaptability to heterogeneous distributions.

To further capture temporal dynamics, a GCL mechanism is employed by constructing positive and negative pairs in the embedding space. Specifically, if $S_{i,j}^{(t)} \geq \tau_p$, then $(z_i^{(t)}, z_j^{(t)})$ is considered a positive pair; if $S_{i,j}^{(t)} \leq \tau_n$, then it is considered a negative pair. Based on this, the contrastive loss is defined as:

$$L_{\text{GCL}}^{(t)} = -\log \frac{\sum_{(i,j) \in \text{Pos}} \exp(\text{sim}(z_i^{(t)}, z_j^{(t)})/\tau)}{\sum_{(i,j) \in \text{Pos} \cup \text{Neg}} \exp(\text{sim}(z_i^{(t)}, z_j^{(t)})/\tau)}, \quad (10)$$

where $\text{sim}(\cdot, \cdot)$ denotes cosine similarity, and τ is the temperature parameter. By minimizing this loss, devices with similar low-rank features are mapped closer in the embedding space, thereby enabling dynamic learning of inter-device similarity structures.

This procedure is executed entirely on the server side, which learns similarity-aware embeddings and aggregation weights.

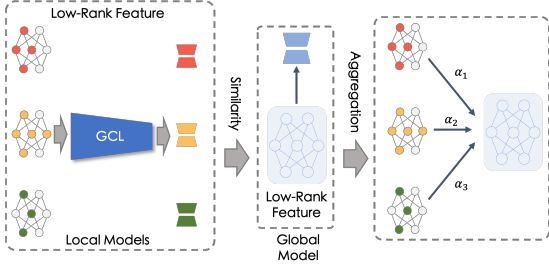


Fig. 2. Local model updates are transformed into low-rank features and aligned in a unified embedding space through GCL. This space enables similarity measurement with the global representation, from which dynamic aggregation weights are derived.

Clients are only responsible for local low-rank feature extraction and model training, without incurring additional computation. Thus, the framework maintains low communication cost, preserves client-side efficiency, and enhances robustness to both heterogeneous distributions and temporal drift in federated FM adaptation.

V. DYNAMIC AGGREGATION BASED ON LOW-RANK FEATURE DISCREPANCY

In federated low-rank adaptation of FMs, each device trains locally and uploads its device-specific low-rank updates to the server. Due to heterogeneous local data distributions, these updates often exhibit substantial discrepancies, which undermine stable global training if directly aggregated. To address this, we introduce a GCL-based embedding space where device-level low-rank feature representations H_k are mapped into a unified low-dimensional space. This mapping achieves two goals: (1) It enables all devices to be expressed in the same dimension for direct comparison and similarity measurement. (2) It captures semantic-level alignment by modeling inter-device relations, mitigating inconsistencies caused by heterogeneous local updates. As illustrated in Fig. 2, this unified embedding space provides the foundation for a dynamic aggregation mechanism.

In this framework, the global model is associated with a low-rank feature representation denoted as H_G , which can be obtained from the LoRA parameters of the aggregated global model in the previous round. The server computes the cosine similarity between each device embedding and the global representation as:

$$\text{sim}(H_k, H_G) = \frac{H_k^\top H_G}{\|H_k\|_2 \|H_G\|_2}, \quad (11)$$

where the similarity reflects the degree to which the low-rank features learned on device k align with the global model's representation. Based on this similarity, the dynamic aggregation weight of device k is defined as:

$$\alpha_k = \frac{\exp(\text{sim}(H_k, H_G))}{\sum_i \exp(\text{sim}(H_i, H_G))}. \quad (12)$$

This softmax-based normalization assigns higher weights to devices whose embeddings are more consistent with the global representation, while automatically down-weighting devices whose updates deviate due to distribution shifts or noise.

The global model update at round t is then computed as:

$$\Delta\theta^{(t)} = \sum_k \alpha_k \Delta\theta_k^{(t)}, \quad (13)$$

where $\Delta\theta_k^{(t)}$ denotes the low-rank update of device k and α_k is its dynamic aggregation weight. Unlike conventional weighted averaging, this mechanism adaptively adjusts device contributions in each communication round. For FMs with billions of parameters, such an approach significantly reduces communication and computation overhead, while maintaining robust aggregation under heterogeneous conditions. By jointly leveraging low-rank embeddings, distribution-aware similarity, and adaptive weighting, the proposed method enables stable and semantically consistent global model training in federated FM fine-tuning, which is particularly beneficial for cross-domain text modeling, personalized language generation, and multilingual adaptation.

The proposed dynamic aggregation strategy provides a semantically consistent and system-efficient mechanism for federated FM fine-tuning. It enables stable global convergence in practical deployments and is particularly advantageous for applications such as cross-domain text modeling, personalized language generation, and multilingual adaptation.

VI. EXPERIMENTS

A. Experimental Setup

Foundation Models. In our experiments, we employ two representative foundation models for computer vision tasks. Specifically, we use the Vision Transformer (ViT) [23], adopting the “vit-base-patch16-224” variant with 12 transformer layers pre-trained on ImageNet-21k [24], and the MLP-Mixer [25], using the “mixer-b16-224” model with 12 layers pre-trained on the same dataset. Both models are fine-tuned following the procedure in [26], with the LoRA rank fixed at 16 across all experiments.

Datasets. We evaluate our approach on two real-world vision datasets that reflect diverse client distributions. DomainNet [27] contains roughly 600k images from 345 categories spanning six visual domains (clipart, infograph, painting, quickdraw, real, sketch). Following [26], we restrict the experiments to the first 100 categories. NICO++ [28], an extension of the NICO dataset, includes around 90k samples from 60 categories, covering six styles such as autumn, dim, grass, outdoor, rock, and water. To model realistic federated scenarios, we adopt non-IID setting. In the feature + label non-IID case, we set 30 clients, dividing each domain/style across five clients, while label imbalance is introduced using a Dirichlet distribution [29] with concentration parameter 0.5. To further mimic partial participation in real-world federated learning, only 18 clients were randomly selected to participate in each communication round, with the number of local iterations set to 5. To account for temporal dynamics, we allow the data distribution of each client to evolve across training rounds by gradually shifting category proportions within a domain, thereby emulating non-stationary environments in which client data changes over time.

TABLE I

EVALUATION RESULTS OF OUR METHOD AGAINST BASELINES ON DOMAINNET AND NICO++ WITH ViT AND MLP-MIXER BACKBONES IN A NON-IID SETTING WHERE FEATURES AND LABELS ARE HETEROGENEOUS. THE ‘‘AVERAGE’’ METRIC CORRESPONDS TO THE MEAN ACCURACY ACROSS DOMAINS.

		Clipart	Infograph	Painting	Quickdraw	Real	Sketch	Average	
DomainNet	ViT	FFA-LoRA	78.28 \pm 0.018	48.51 \pm 0.022	74.39 \pm 0.051	58.08 \pm 0.025	84.71 \pm 0.106	72.12 \pm 0.091	69.35
		FedIT	80.78 \pm 0.029	49.14 \pm 0.024	76.22 \pm 0.067	57.05 \pm 0.019	86.60 \pm 0.074	73.14 \pm 0.125	70.49
		FLoRA	80.44 \pm 0.042	50.25 \pm 0.043	76.77 \pm 0.063	59.22 \pm 0.058	85.61 \pm 0.060	72.49 \pm 0.082	70.80
		FlexLoRA	80.93 \pm 0.018	50.42 \pm 0.036	75.76 \pm 0.084	58.37 \pm 0.079	85.94 \pm 0.055	73.75 \pm 0.085	70.86
		LoRA-FAIR	81.06 \pm 0.024	51.52 \pm 0.058	77.00 \pm 0.038	59.26 \pm 0.059	86.10 \pm 0.026	73.54 \pm 0.053	71.41
		Ours	83.75 \pm 0.020	56.72 \pm 0.041	79.85 \pm 0.019	62.39 \pm 0.047	88.28 \pm 0.018	77.11 \pm 0.038	74.10
	MLP-Mixer	FFA-LoRA	60.82 \pm 0.025	31.44 \pm 0.062	61.84 \pm 0.022	23.59 \pm 0.024	77.85 \pm 0.040	48.02 \pm 0.017	50.59
		FedIT	69.45 \pm 0.072	36.56 \pm 0.089	65.99 \pm 0.084	40.36 \pm 0.060	79.75 \pm 0.021	57.80 \pm 0.071	58.32
		FLoRA	67.32 \pm 0.052	34.62 \pm 0.095	65.05 \pm 0.095	38.71 \pm 0.090	78.88 \pm 0.016	57.49 \pm 0.106	57.01
		FlexLoRA	68.71 \pm 0.052	36.84 \pm 0.033	66.69 \pm 0.024	41.56 \pm 0.091	79.80 \pm 0.012	58.57 \pm 0.096	58.70
		LoRA-FAIR	70.33 \pm 0.013	38.02 \pm 0.043	66.65 \pm 0.064	43.59 \pm 0.073	79.95 \pm 0.054	59.15 \pm 0.059	59.61
		Ours	71.12 \pm 0.018	39.45 \pm 0.037	67.35 \pm 0.049	51.82 \pm 0.051	81.22 \pm 0.012	62.32 \pm 0.041	62.50
NICO++	ViT	FFA-LoRA	88.76 \pm 0.013	84.08 \pm 0.056	89.58 \pm 0.045	86.10 \pm 0.048	87.54 \pm 0.065	84.39 \pm 0.035	86.74
		FedIT	88.63 \pm 0.035	84.90 \pm 0.034	89.96 \pm 0.091	86.05 \pm 0.093	87.01 \pm 0.040	84.99 \pm 0.068	86.92
		FLoRA	88.51 \pm 0.035	84.66 \pm 0.062	89.63 \pm 0.064	86.62 \pm 0.064	87.88 \pm 0.056	84.51 \pm 0.019	86.97
		FlexLoRA	89.05 \pm 0.031	84.82 \pm 0.046	90.11 \pm 0.007	87.03 \pm 0.021	86.94 \pm 0.033	84.59 \pm 0.071	87.09
		LoRA-FAIR	88.80 \pm 0.040	85.34 \pm 0.060	90.52 \pm 0.016	87.33 \pm 0.028	88.32 \pm 0.025	84.83 \pm 0.018	87.52
		Ours	91.02 \pm 0.052	88.10 \pm 0.061	92.10 \pm 0.023	89.36 \pm 0.046	89.58 \pm 0.029	88.84 \pm 0.050	89.70
	MLP-Mixer	FFA-LoRA	77.65 \pm 0.075	70.95 \pm 0.048	79.31 \pm 0.081	75.32 \pm 0.081	73.54 \pm 0.051	69.14 \pm 0.068	74.32
		FedIT	78.65 \pm 0.021	72.17 \pm 0.092	80.24 \pm 0.075	76.31 \pm 0.084	76.12 \pm 0.019	72.20 \pm 0.081	75.95
		FLoRA	78.79 \pm 0.018	72.10 \pm 0.055	80.59 \pm 0.075	76.29 \pm 0.053	75.62 \pm 0.035	71.91 \pm 0.122	75.88
		FlexLoRA	79.55 \pm 0.032	72.49 \pm 0.015	81.19 \pm 0.021	76.10 \pm 0.012	76.23 \pm 0.030	71.26 \pm 0.098	76.14
		LoRA-FAIR	80.17 \pm 0.049	73.93 \pm 0.027	81.74 \pm 0.040	77.37 \pm 0.041	77.78 \pm 0.046	72.71 \pm 0.091	77.28
		Ours	81.12 \pm 0.044	74.75 \pm 0.052	82.41 \pm 0.021	78.56 \pm 0.039	78.90 \pm 0.031	81.52 \pm 0.047	79.80

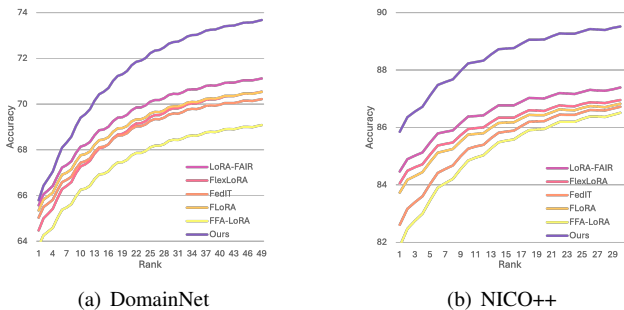


Fig. 3. Convergence comparison of different methods for ViT on DomainNet and NICO++.

Training Details. All results are averaged over three independent runs. We adopt a mini-batch size of 128 and use SGD with a learning rate of 0.01 for local training. The number of local iterations is set to 2 under the feature non-IID configuration and 5 under the feature + label non-IID configuration. For DomainNet, training proceeds for 50 global rounds, while for NICO++ we use 30 rounds. In the feature non-IID case, all six clients participate in every round, whereas in the feature + label non-IID case, 18 clients are randomly selected at each round to emulate partial participation.

Baselines. To assess the effectiveness of our proposed LoRA-FAIR method, we benchmark it against several representative approaches for federated LoRA fine-tuning: FedIT [7], FFA-LoRA [9], FLoRA [16], FlexLoRA [19], and LoRA-FAIR [18].

B. Experimental Results

Performance Evaluation. On the DomainNet and NICO++ datasets, using ViT and MLP-Mixer as FMs respectively, the experimental results are reported in Table I. Overall, our method consistently achieves the best performance across all domains as well as in terms of overall average accuracy. In addition to higher final accuracy, our approach also exhibits a faster and more stable convergence rate, reaching competitive performance in fewer communication rounds compared to existing methods (see Fig. 3).

In contrast, existing approaches demonstrate various limitations under different non-IID settings. FedIT, which simply combines FedAVG with LoRA, fails to explicitly model cross-device distributional heterogeneity and temporal dynamics, thus showing limited performance in complex scenarios. FFA-LoRA mitigates aggregation bias partially by fixing the A matrix and updating only the B matrix to reduce local computation, yet this design weakens the model’s expressive capacity and hinders its adaptability to heterogeneous client data. FLoRA, by stacking and transmitting local LoRA modules at the server, alleviates aggregation bias to some extent; however, its performance improvement remains unstable under highly imbalanced label distributions. FlexLoRA leverages SVD to decompose aggregated updates, enhancing the flexibility of low-rank representations, but it still fails to capture temporal dynamics across devices. Even LoRA-FAIR, one of the more advanced methods, introduces correction terms on the server

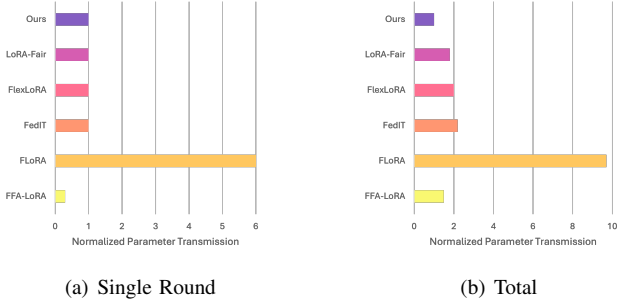


Fig. 4. Comparison of normalized communication parameters of different methods.

side to address aggregation bias and initialization lag; nonetheless, due to its lack of fine-grained modeling of inter-client low-rank feature similarities and their temporal evolution, its performance remains suboptimal in highly heterogeneous and dynamically shifting scenarios.

Beyond addressing the aforementioned limitations, our method provides a more comprehensive characterization of inter-client similarity structures and adapts effectively to temporal distribution drift. As a result, it improves the robustness and generalization of the global model while keeping computational and communication costs manageable.

Communication Overhead Comparison. In FL scenarios, communication cost is often the key factor affecting overall training efficiency. Therefore, we conducted a dedicated analysis of the communication cost of our method. As shown in Fig. 4(a), in each training round, our method only needs to distribute the updated \bar{B}' and \bar{A} to server. Compared with FedIT, FlexLoRA, and LoRA-Fair, this does not introduce any additional communication burden. Since the parameter sizes of \bar{B}' and \bar{A} are comparable to those in the conventional LoRA approach, the communication cost remains on the same order of magnitude as mainstream lightweight methods.

In contrast, FLoRA stacks all locally uploaded LoRA modules on the server side before redistributing them, which, although providing greater flexibility in aggregation, significantly increases communication overhead—especially in large-scale client scenarios. On the other hand, FFA-LoRA reduces communication cost by fixing the A matrix and transmitting only the smaller B matrix during communication, making it the most communication-efficient approach. However, as shown in Table I, this method performs noticeably worse than other approaches in terms of model performance.

Moreover, thanks to the better convergence of our method, as illustrated in the Fig. 4(b), when different approaches reach the same level of accuracy, our method requires the least total communication volume. Therefore, compared with the baselines, our method achieves a better trade-off between accuracy and communication cost.

Impact of Different LoRA Rank. We investigate the influence of different LoRA ranks by setting the rank to $\{4, 8, 16, 32\}$ and report the results in Fig. 5. Our method consistently achieves the highest accuracy across all rank settings, demonstrating its robustness and superior adaptability.

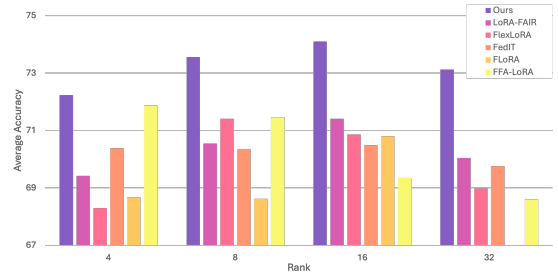


Fig. 5. Average accuracy comparison of ViT on DomainNet under varying LoRA rank settings (4, 8, 16, 32).

Stronger performance at lower ranks indicates that our approach is more parameter-efficient, capable of achieving competitive accuracy with fewer trainable parameters, which is crucial in resource-constrained federated environments. At higher ranks, maintaining superior accuracy reflects that our method can also fully exploit FM capacities without succumbing to overfitting, thereby adapting well to heterogeneous data distributions. Notably, while several baselines such as LoRA-FAIR, FlexLoRA, and FLoRA achieve competitive performance at lower ranks, their accuracy either stagnates or declines as the rank increases, suggesting that a higher rank does not necessarily lead to improved final accuracy. In contrast, our approach maintains steady improvements and avoids overfitting effects that appear in other methods.

Moreover, we observe that FFA-LoRA struggles with performance stability and provides the lowest accuracy across all ranks, reflecting the limitations of fixing the A matrix and relying solely on B updates. Meanwhile, FedIT also shows limited scalability, with accuracy degradation becoming evident at higher ranks. Importantly, FLoRA fails to converge properly at rank 32, highlighting the drawback of its server-side aggregation mechanism when scaling to larger ranks.

These results confirm that our proposed method not only surpasses baselines in terms of accuracy but also remains stable and effective under different rank configurations, thereby validating its practicality in FL scenarios.

VII. CONCLUSION

This work addresses the critical challenges of data heterogeneity and temporal dynamics in federated adaptation of foundation models. By introducing a GCL-based dynamic aggregation strategy for Fed-LoRA, the proposed framework effectively captures inter-client similarity, models evolving distributional shifts, and adaptively adjusts aggregation weights. This approach not only reduces communication overheads but also enhances robustness and scalability in heterogeneous and evolving environments.

ACKNOWLEDGEMENT

This work was supported by National Key R&D Program of China (No. 2024YFB4303400), Natural Science Basic Research Plan in Shaanxi Province of China (Program No. 2025JC-YBQN-811), Key Science and Technology Program of Shaanxi Province, China (Grant No. 2025CY-YBXM-197), and National Natural Science Foundation of China (Grant No. 12171234).

REFERENCES

- [1] A. Mukanova, M. Milosz, A. Dauletaliyeva, A. Nazyrova, G. Yelibayeva, D. Kuzin, and L. Kussepova, "Llm-powered natural language text processing for ontology enrichment," *Applied Sciences*, 2024.
- [2] X. Long, J. Zeng, F. Meng, Z. Ma, K. Zhang, B. Zhou, and J. Zhou, "Generative multi-modal knowledge retrieval with large language models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.
- [3] M. Ferrag, N. Tihanyi, and M. Debbah, "From llm reasoning to autonomous ai agents: A comprehensive review," in *arXiv preprint arXiv:2504.19678*, 2025.
- [4] X. Wu, M. Chen, W. Li, R. Wang, L. Lu, J. Liu, K. Hwang, Y. Hao, Y. Pan, Q. Meng, and K. Huang, "Llm fine-tuning: Concepts, opportunities, and challenges," *Big Data and Cognitive Computing*, 2025.
- [5] W. Kuang, B. Qian, Z. Li, D. Chen, D. Gao, X. Pan, Y. Xie, Y. Li, B. Ding, and J. Zhou, "Federatedscope-llm: A comprehensive package for fine-tuning large language models in federated learning," in *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024.
- [6] C. Hatfaludi and A. Serban, "Foundational models and federated learning: survey, taxonomy, challenges and practical insights," *PeerJ Computer Science*, 2025.
- [7] J. Zhang, S. Vahidian, M. Kuo, C. Li, R. Zhang, T. Yu, G. Wang, and Y. Chen, "Towards building the federatedgpt: Federated instruction tuning," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2024.
- [8] E. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," in *International Conference on Learning Representations*, 2022.
- [9] Y. Sun, Z. Li, Y. Li, and B. Ding, "Improving lora in privacy-preserving federated learning," in *International Conference on Learning Representations*, 2024.
- [10] L. Zhang, L. Zhang, S. Shi, X. Chu, and B. Li, "Lora-fa: Memory-efficient low-rank adaptation for large language models fine-tuning," in *arXiv preprint arXiv:2308.03303*, 2023.
- [11] P. Guo, S. Zeng, Y. Wang, H. Fan, F. Wang, and L. Qu, "Selective aggregation for low-rank adaptation in federated learning," in *International Conference on Learning Representations*, 2025.
- [12] F. Wu, J. Hu, G. Min, and S. Wang, "Adaptive rank allocation for federated parameter-efficient fine-tuning of language models," in *arXiv preprint arXiv:2501.14406*, 2025.
- [13] Z. Zhang, J. Xu, P. Liu, and R. Hu, "Fed-pilot: Optimizing lora assignment for efficient federated foundation model fine-tuning," in *arXiv preprint arXiv:2410.10200*, 2024.
- [14] Y. Cho, L. Liu, Z. Xu, A. Fahrezi, and G. Joshi, "Heterogeneous lora for federated fine-tuning of on-device foundation models," in *International Workshop on Federated Learning in the Age of Foundation Models in Conjunction with NeurIPS*, 2023.
- [15] Y. Byun and J. Lee, "Towards federated low-rank adaptation with rank-heterogeneous communication," in *NeurIPS 2024 Workshop on Adaptive Foundation Models*, 2024.
- [16] Z. Wang, Z. Shen, Y. He, G. Sun, H. Wang, L. Lyu, and A. Li, "Flora: Federated fine-tuning large language models with heterogeneous low-rank adaptations," in *Advances in Neural Information Processing Systems*, 2024.
- [17] S. Chen, O. Tavallaie, N. Nazemi, and A. Zomaya, "Rbla: Rank-based-lora-aggregation for fine-tuning heterogeneous models in fllaas," in *International Conference on Web Services*, 2024.
- [18] J. Bian, L. Wang, L. Zhang, and J. Xu, "Lora-fair: Federated lora fine-tuning with aggregation and initialization refinement," in *International Conference on Computer Vision*, 2025.
- [19] J. Bai, D. Chen, B. Qian, L. Yao, and Y. Li, "Federated fine-tuning of large language models under heterogeneous language tasks and client resources," in *Advances in Neural Information Processing Systems*, 2024.
- [20] G. Long, T. Shen, J. Jiang, and M. Blumenstein, "Dual-personalizing adapter for federated foundation models," in *Advances in Neural Information Processing Systems*, 2024.
- [21] J. Qi, Z. Luan, S. Huang, C. Fung, H. Yang, and D. Qian, "Fdflora: personalized federated learning of large language model via dual lora tuning," in *arXiv preprint arXiv:2406.07925*, 2024.
- [22] L. Yi, H. Yu, G. Wang, X. Liu, and X. Li, "pfdflora: model-heterogeneous personalized federated learning with lora tuning," in *arXiv preprint arXiv:2310.13283*, 2023.
- [23] A. Dosovitskiy, "An image is worth 16x16 words: Transformers for image recognition at scale," in *arXiv preprint arXiv:2010.11929*, 2020.
- [24] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE conference on computer vision and pattern recognition*, 2009.
- [25] I. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, and M. Lucic, "Mlp-mixer: An all-mlp architecture for vision," in *Advances in neural information processing systems*, 2021.
- [26] B. L. Shangchao Su and X. Xue, "Fedra: A random allocation strategy for federated tuning to unleash the power of heterogeneous clients," in *European Conference on Computer Vision*, 2024.
- [27] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," in *IEEE/CVF international conference on computer vision*, 2019.
- [28] Y. He, Z. Shen, and P. Cui, "Towards non-iid image classification: A dataset and baselines," *Pattern Recognition*, 2021.
- [29] Q. Li, Y. Diao, Q. Chen, and B. He, "Federated learning on non-iid data silos: An experimental study," in *IEEE 38th international conference on data engineering*, 2022.