Federated Low-Rank Adaptation for Foundation Models: A Survey

Yiyuan Yang[1] , **Guodong Long**[1] , **Qinghua Lu**[2] , **Liming Zhu**[2] , **Jing Jiang**[1] and **Chengqi Zhang**[3]
[1]University of Technology Sydney, Australia
[2]CSIRO's Data61, Australia
[3]The Hong Kong Polytechnic University, China
Yiyuan.Yang-1@student.uts.edu.au, {guodong.long, jing.jiang}@uts.edu.au, {Qinghua.Lu, Liming.Zhu}@data61.csiro.au, chengqi.zhang@polyu.edu.hk,

Abstract

Effectively leveraging private datasets remains a significant challenge in developing foundation models. Federated Learning (FL) has recently emerged as a collaborative framework that enables multiple users to fine-tune these models while mitigating data privacy risks. Meanwhile, Low-Rank Adaptation (LoRA) offers a resource-efficient alternative for fine-tuning foundation models by dramatically reducing the number of trainable parameters. This survey examines how LoRA has been integrated into federated fine-tuning for foundation models—an area we term FedLoRA—by focusing on three key challenges: distributed learning, heterogeneity, and efficiency. We further categorize existing work based on the specific methods used to address each challenge. Finally, we discuss open research questions and highlight promising directions for future investigation, outlining the next steps for advancing FedLoRA.

1 Introduction

With the increasing limitations of centralized learning in handling large-scale and privacy-sensitive data, Federated Learning (FL) [Zhang *et al.*, 2021] has emerged to collaboratively learn model across distributed clients without direct access to data. One of the primary challenges in FL lies in the computational and communication overhead [Almanifi *et al.*, 2023], as clients must train models locally and periodically exchange updates with the central server. This challenge becomes even more pronounced with modern foundation models, which are increasingly growing in depth and size, e.g., large language models (LLMs) often reaching millions or even billions of parameters. The substantial resource demands of such models necessitate the development of efficient learning techniques in federated learning.

Parameter-efficient fine-tuning (PEFT) methods have been proposed as an effective solution for tuning foundation models, which optimize only a subset of parameters to enable efficient learning while maintaining performance comparable to full fine-tuning, thereby making it particularly suitable for resource-constrained environments. Inspired by this, recent studies [Kuang *et al.*, 2024; Zhang *et al.*, 2023] have explored the integration of PEFT methods into the FL framework to enhance training efficiency. Among these methods, Federated Low-Rank Adaptation (FedLoRA), integrating LoRA [Hu *et al.*, 2021] in FL, has gained significant attention due to its efficiency and parallel ability during training processes. By decomposing model updates into low-rank matrices for learning and communicating, FedLoRA significantly reduces computation and communication costs while maintaining model performance.

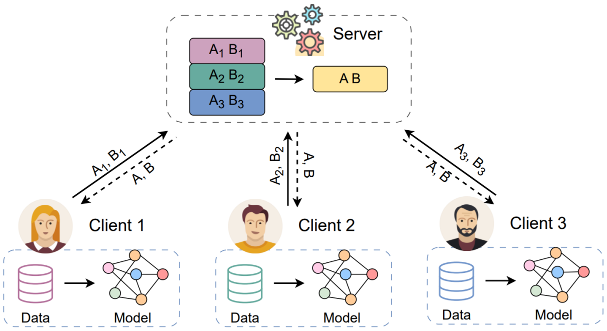


Figure 1: The overall framework of FedLoRA, where only the LoRA parameters (A and B) are communicated for efficient learning.

Although FedLoRA addresses efficiency concerns to some extent, numerous challenges such as heterogeneity remain when aggregating LoRA in FL. In response, extensive research has been conducted to enhance the effectiveness and efficiency of FedLoRA. To systematically analyze and consolidate these advancements, we present a comprehensive survey on FedLoRA, providing an in-depth exploration of its methodologies, challenges, and solutions. Unlike existing surveys that primarily focus on broader aspects of efficient FL training [Almanifi *et al.*, 2023; Woisetschläger *et al.*, 2024] or general federated foundation models [Zhuang *et al.*, 2023; Yu *et al.*, 2023; Ren *et al.*, 2024], our study specifically examines the detailed mechanisms and unique challenges of LoRA aggregation in FL. By offering a focused and structured analysis, this survey aims to bridge existing research gaps and provide insights into future directions for FedLoRA[1].

Contributions. Our contributions can be summarized as: 1) A structured taxonomy: we present a systematic classifica-

[1] https://github.com/Lydia-yang/Awesome-Federated-LoRA

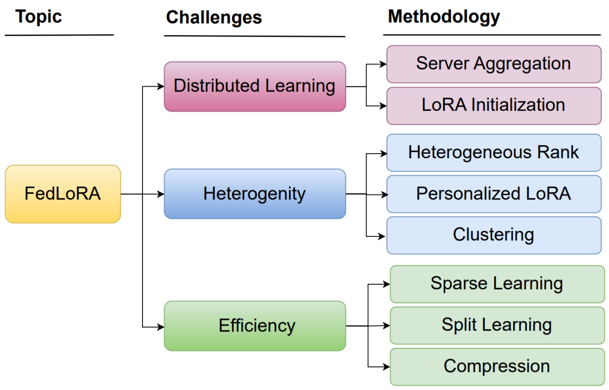


Figure 2: Taxonomy of FedLoRA focusing on distributed learning, heterogeneity and efficiency with further classified subcategories.

tion of FedLoRA based on different technologies; 2) A comprehensive review: we conduct an extensive survey of recent advancements in FedLoRA within the proposed taxonomy; 3) Some future directions: we highlight open challenges and emerging research opportunities in FedLoRA.

2 Preliminary

2.1 Federated Learning

FL [Zhang *et al.*, 2021] is a decentralized machine learning paradigm designed to enable multiple devices or clients to collaboratively train models while preserving privacy data by ensuring that data remains localized on each client. Typically, in an FL scenario, there are K clients in an FL scenrio, where each client k has access to its own local dataset D_k. The objective of FL is to optimize a global model by minimizing a weighted sum of local objective functions across all clients:

$$\min_{\boldsymbol{W}} f(\boldsymbol{W}) = \sum_{k=1}^{K} p_k f_k(\boldsymbol{W}; D_k), \tag{1}$$

where $f(\boldsymbol{W})$ represents the global objective function parameterized by $\boldsymbol{W}$, $f_k(\boldsymbol{W}; D_k)$ is the local objective function computed on client k's dataset D_k, and p_k is the weight assigned to client k. Building on this formulation, numerous FL methods have been developed to tackle a variety of challenges inherent to the paradigm, including data heterogeneity arising from varying client data distributions, communication efficiency necessitated by frequent model updates, and so on.

2.2 Low-Rank Adaptation

With the rapid growth in the size and depth of modern foundation models, PEFT methods have emerged as a practical solution to adapt these large-scale models efficiently by learning only a small subset of parameters. Among these PEFT methods, LoRA [Hu *et al.*, 2021] distinguishes itself through its inherent parallelism and efficiency, achieving impressive results without introducing new parameters into the model. LoRA leverages low-rank decomposition by reparameterizing the weight updates for each layer, thereby significantly reducing memory consumption and computational overhead while maintaining model performance. Formally, given a weight matrix $\boldsymbol{W} \in \mathbb{R}^{m \times n}$, LoRA introduces two low-rank matrices, $\boldsymbol{A} \in \mathbb{R}^{r \times n}$ and $\boldsymbol{B} \in \mathbb{R}^{m \times r}$, where $r \ll \min(m, n)$, which can be formulated as:

$$\boldsymbol{W} = \boldsymbol{W}' + \Delta \boldsymbol{W} = \boldsymbol{W}' + \boldsymbol{B}\boldsymbol{A}. \tag{2}$$

By restricting the rank r, LoRA ensures that the number of additional parameters is significantly smaller than the original model size, enabling efficient fine-tuning of large-scale models without sacrificing performance.

Variations of LoRA. Various adaptations of LoRA have been proposed to address different challenges and applications, including dynamic rank adjustment for adaptability [Valipour *et al.*, 2022], quantization techniques for further efficiency [Dettmers *et al.*, 2024], and advanced strategies such as Bayesian inference [Yang *et al.*, 2023] to enhance overall effectiveness.

2.3 Federated Low-Rank Adaptation

With the increasing demand for efficient learning in FL, recent research has started to explore the adaptation of LoRA within FL (FedLoRA). FedLoRA leverages the core idea of LoRA by learning and communicating only a small subset of parameters, thereby enhancing both computational and communication efficiency. The overall architecture can be seen in Figure 1, where each client maintains its own model for local training while only the LoRA parameters are transmitted to the server for aggregation. The overall objective is:

$$\min_{\Delta \boldsymbol{W}} f(\boldsymbol{W}) = \sum_{k=1}^{K} p_k f_k(\boldsymbol{W}', \Delta \boldsymbol{W}; D_k), \tag{3}$$

where $\Delta \boldsymbol{W} = \boldsymbol{B}\boldsymbol{A}$ denotes the learnable LoRA parameters.

Taxonomy. Unlike previous surveys [Almanifi *et al.*, 2023; Woisetschläger *et al.*, 2024] that provide a broad overview of efficient FL, our paper focuses specifically on the adaptation of LoRA in FL and delves into the unique challenges introduced by FedLoRA. We propose a taxonomy in Figure 2 that categorizes these challenges into three main aspects: 1) enabling effective distributed learning for stable convergence and knowledge sharing, 2) addressing heterogeneity arising from non-IID data distributions among clients to ensure stable performance, and 3) further optimizing computational and communication efficiency to improve resource utilization. Various approaches have been proposed to address these, which are further classified into subcategories based on the underlying technologies used, providing a detailed examination of the current landscape.

3 Distributed Learning

As a distributed learning paradigm, FL relies on effective client aggregation and proper initialization—both before training and at each communication round—to ensure model convergence and performance consistency. This section systematically examines FedLoRA from these perspectives.

3.1 Server Aggregation

To facilitate knowledge sharing in distributed learning systems, conventional FL often employs specific weighting algorithms, such as FedAVG [McMahan *et al.*, 2017], for server aggregation to balance contributions from diverse clients and improve global model convergence. Recent research [Zhang *et al.*, 2024b] has extended these algorithms to FedLoRA for distributed learning. However, due to the unique structure of LoRA, directly applying FedAVG in FedLoRA can result in suboptimal aggregation outcomes [Sun *et al.*, 2024].

LoRA Aggregation Discordance. As introduced in Section 2.2, LoRA contains two low-rank matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ for learning and communicating. However, applying weighting algorithms separately to matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ is inconsistent with the objective of joint optimization, potentially leading to performance degradation. To illustrate this, consider a scenario with two clients, each with LoRA parameters $(\boldsymbol{A}_1, \boldsymbol{B}_1)$ and $(\boldsymbol{A}_2, \boldsymbol{B}_2)$ for aggregation, this discordance between separate aggregation of $\boldsymbol{A}$ and $\boldsymbol{B}$ and their intended joint optimization can be mathematically formulated as:

$$\begin{aligned} &\underbrace{\boldsymbol{W} = \boldsymbol{W}' + \frac{1}{2}(\boldsymbol{B}_1 + \boldsymbol{B}_2) \times \frac{1}{2}(\boldsymbol{A}_1 + \boldsymbol{A}_2)}_{\text{Separate aggregation of } \boldsymbol{A} \text{ and } \boldsymbol{B}} \\ &\neq \underbrace{\boldsymbol{W}' + \frac{1}{2}(\boldsymbol{B}_1\boldsymbol{A}_1 + \boldsymbol{B}_2\boldsymbol{A}_2) = \boldsymbol{W}^*.}_{\text{Ideal aggregation for joint optimization}} \end{aligned} \tag{4}$$

Single Low-Rank Matrix Aggregation. One simple and intuitive approach to addressing this LoRA aggregation discordance is to learn and aggregate only one low-rank matrix per communication round. Specifically, during each communication round, only one low-rank matrix (either $\boldsymbol{A}$ or $\boldsymbol{B}$) is learned and sent to the server for aggregation, while the other matrix remains fixed and consistent across all clients. This strategy ensures that the ideal aggregation can be equivalently achieved by aggregating a single low-rank matrix, thereby simplifying the process and maintaining computational efficiency. It can be formulated as follows:

$$\begin{aligned} \boldsymbol{W}' + \frac{1}{2}(\boldsymbol{B}\boldsymbol{A}_1 + \boldsymbol{B}\boldsymbol{A}_2) = \boldsymbol{W}' + \frac{1}{2}\boldsymbol{B}(\boldsymbol{A}_1 + \boldsymbol{A}_2) \\ \text{or } \boldsymbol{W}' + \frac{1}{2}(\boldsymbol{B}_1\boldsymbol{A} + \boldsymbol{B}_2\boldsymbol{A}) = \boldsymbol{W}' + \frac{1}{2}(\boldsymbol{B}_1 + \boldsymbol{B}_2)\boldsymbol{A}. \end{aligned} \tag{5}$$

The study [Sun *et al.*, 2024] first analyzed the discordance issue in FedLoRA and proposed FFA-LoRA, which freezes the low-rank matrix $\boldsymbol{A}$ and only updates the matrix $\boldsymbol{B}$ for aggregation. This approach not only achieves more consistent performance by aligning with the objective of joint optimization but also significantly reduces computational costs. Similarly, CoLR [Nguyen *et al.*, 2024] introduced a novel strategy where, in each communication round, clients learn a newly initialized matrix $\boldsymbol{A}$ from their local data for aggregation, while keeping $\boldsymbol{B}$ unified across clients via decomposing full matrix W on the server. To further enhance the performance of FedLoRA, RoLoRA [Chen *et al.*, 2024b] employed an alternating minimization approach, learning and aggregating only $\boldsymbol{B}$ in odd communication rounds and $\boldsymbol{A}$ in even rounds. This alternating strategy effectively addresses the discordance issue while providing more robust performance across heterogeneous scenes. Meanwhile, LoRA-A^2 [Koo *et al.*, 2024] also explored the alternating minimization approach and incorporated an adaptive rank selection strategy to further reduce communication costs by dynamically selecting the most important LoRA ranks for learning and aggregation.

Full-size Matrix Aggregation. Another class of approaches to address the discordance issue is leveraging the full-size matrix $\Delta\boldsymbol{W}$, reconstructed from the product of two low-rank matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. Instead of aggregating $\boldsymbol{A}$ and $\boldsymbol{B}$ separately, this approach aggregates the full-size matrix $\Delta\boldsymbol{W} = \boldsymbol{B}\boldsymbol{A}$, achieving ideal aggregation directly:

$$\begin{aligned} \boldsymbol{W}' + \Delta\overline{\boldsymbol{W}} &= \boldsymbol{W}' + \frac{1}{2}(\Delta\boldsymbol{W}_1 + \Delta\boldsymbol{W}_2) \\ &= \boldsymbol{W}' + \frac{1}{2}(\boldsymbol{B}_1\boldsymbol{A}_1 + \boldsymbol{B}_2\boldsymbol{A}_2), \end{aligned} \tag{6}$$

where $\Delta\overline{\boldsymbol{W}}$ represents the newly aggregated full-size matirx, and the updated low-rank matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ can be obtained by decomposing$\Delta\overline{\boldsymbol{W}}$. FlexLoRA [Bai *et al.*, 2024] was the first to adopt this approach, simultaneously learning both $\boldsymbol{A}$ and $\boldsymbol{B}$, aggregating the full-size LoRA weight $\Delta\boldsymbol{W} = \boldsymbol{B}\boldsymbol{A}$ based on individual client contributions, and then employing Singular Value Decomposition (SVD) for weight redistribution. This method not only addressed the discordance issue but also the heterogeneous rank configurations across clients to enhance aggregation effectiveness. Similarly, FedPipe [Fang *et al.*, 2024] also explored full-size LoRA weight aggregation and integrated quantization techniques to improve training efficiency by quantizing local models into different bit levels. At the same time, FloRA [Wang *et al.*, 2024b] implemented the full-size LoRA weight aggregation in another way, which stacks all clients' $\boldsymbol{A}$ and $\boldsymbol{B}$ respectively to derive the final aggregated full-size LoRA weight.

Corrective Mechanism. To preserve the advantages of initialization with averaged low-rank matrices [Bian *et al.*, 2024], a corrective mechanism is proposed to address the inaccuracies in separately aggregated LoRA matrices, bringing them closer to the ideal aggregation. FedEx-LoRA [Singhal *et al.*, 2024] first introduced a corrective mechanism by calculating the residual $\Delta\hat{\boldsymbol{W}}$ between the ideal aggregated weight and the weight obtained from separate aggregation, and then adding the residual to the pre-trained weight matrix $\boldsymbol{W}'$ to rectify inaccuracies in the aggregation process, improving the alignment with the ideal aggregation. It can be formulated as:

$$\begin{aligned} \Delta\hat{\boldsymbol{W}} &= (\boldsymbol{B}_1\boldsymbol{A}_1 + \boldsymbol{B}_2\boldsymbol{A}_2) - (\boldsymbol{B}_1 + \boldsymbol{B}_2)(\boldsymbol{A}_1 + \boldsymbol{A}_2), \\ \boldsymbol{W} &= \boldsymbol{W}' + \Delta\hat{\boldsymbol{W}} + \boldsymbol{B}\boldsymbol{A}, \end{aligned} \tag{7}$$

where $\Delta\hat{\boldsymbol{W}}$ represents the residual error, and $\boldsymbol{A}$ and $\boldsymbol{B}$ are the separately aggregated LoRA matrices. Additionally, LoRA-FAIR [Bian *et al.*, 2024] also adapted the corrective mechanism by incorporating a corrective term $\Delta\boldsymbol{B}$ to refine the matrix $\boldsymbol{B}$, which is to minimize the residual error $\Delta\hat{\boldsymbol{W}}$ by optimizing the similarity between the ideal aggregated LoRA weights $\Delta\overline{\boldsymbol{W}}$ and the corrected LoRA weights $(\boldsymbol{B} + \Delta\boldsymbol{B})\boldsymbol{A}$.

Rank Clustering Aggregation. Rather than treating the low-rank matrix as a single aggregation unit, FedInc [Qin and Li, 2024] proposed considering each rank of LoRA as the smallest semantic unit and introduced a clustering-based aggregation algorithm. This approach combines the two low-rank matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ based on their ranks to determine the newly aggregated LoRA, enabling more fine-grained and adaptive aggregation. Specifically, $\boldsymbol{A} \in \mathbb{R}^{r\times n}$ and $\boldsymbol{B} \in \mathbb{R}^{m\times r}$ are merged into $\boldsymbol{C} \in \mathbb{R}^{r\times(m+n)}$, which represents the combined low-rank space. From K clients, a set of vectors $\{\boldsymbol{z}_i\}_{i=1}^{K\times r}$ is collected and clustered into N clusters, resulting in new rank $r = N$ for the aggregated LoRA, formulated as:

$$\begin{gathered}\{\boldsymbol{C}_k = [\boldsymbol{A}_k, \boldsymbol{B}_k^T]\}_{k=1}^{K} \to \{\boldsymbol{z}_i\}_{i=1}^{K\times r}, \\ \min_{\boldsymbol{\mu}} \sum_{n=1}^{N} \sum_{\boldsymbol{z}\in\mathbb{Z}_n} ||\boldsymbol{z} - \boldsymbol{\mu}_n||^2,\end{gathered} \tag{8}$$

where $\mathbb{Z}$ denotes a cluster of vectors from $\{\boldsymbol{z}_i\}_{i=1}^{K\times r}$, and $\boldsymbol{\mu}_n$ is the centroid of cluster $\mathbb{Z}_n$. The final set of centroids $\{\boldsymbol{\mu}_n\}_{n=1}^{N}$is used as the aggregated parameters and can be further decomposed into the newly updated matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ with a new rank $r = N$. This method not only effectively addresses the discordance issue but also allows the LoRA rank to adapt dynamically to the inherent heterogeneity in FL.

3.2 LoRA Initialization

In FL, initialization plays a crucial role in determining training efficiency, convergence speed, and model performance, and it can be broadly categorized into server-side and client-side initialization. Server-side initialization focuses on the initial setup of LoRA parameters before training begins to ensure consistency across clients, while client-side initialization deals with the reinitialization of LoRA for individual clients at the beginning of each communication round, balancing global consistency and local adaptability. We detail these initialization strategies of FedLoRA in this section.

Server-Side Initialization. For server-side initialization, FedLoRA typically uses a random Gaussian initialization for $\boldsymbol{A}$ and zero initialization for $\boldsymbol{B}$, as formulated below:

$$\boldsymbol{A} = \mathcal{N}(0, \sigma^2) \quad \text{and} \quad \boldsymbol{B} = 0, \tag{9}$$

where $\mathcal{N}$ denotes a Gaussian distribution with mean 0 and variance σ^2. However, in the distributed FL setting, this standard server-side initialization can lead to significant weight update drift, widening the performance gap compared to fully fine-tuned models. To address this challenge, SLoRA [Babakniya *et al.*, 2023] introduced a novel data-driven initialization technique, which begins with sparse fine-tuning to find a mature starting point for LoRA, and then applies SVD to the fine-tuned parameters to initialize the low-rank matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ for continual conventional LoRA tuning in FL. Similarly, FeDeRA [Yan *et al.*, 2024] addressed this challenge by initializing these matrices directly from the SVD of the pre-trained weight matrices, effectively mitigating weight divergence. Overall, these methods can be generalized into:

$$\begin{gathered} W_{pre} \xrightarrow{SVD} \boldsymbol{U}\Sigma\boldsymbol{V}^T, \\ \boldsymbol{A} = \boldsymbol{V}[:, : r] \quad \text{and} \quad \boldsymbol{B} = \boldsymbol{U}[: r, :]\Sigma[: r],\end{gathered} \tag{10}$$

where W_{pre} is the pre-trained weight matrix for initialization.

Client-Side Initialization. For client-side initialization, LoRA-FAIR [Bian *et al.*, 2024] investigated three strategies:

- Avg-Initial follows conventional FL approaches, initializing clients with aggregated LoRA parameters for next communication round. For each communication round t with K clients, this can be formulated as $\boldsymbol{A}_t^k = \boldsymbol{A}_t = \sum_{k=1}^{K} p_k \boldsymbol{A}_{t-1}^k$ and $\boldsymbol{B}_t^k = \boldsymbol{B}_t = \sum_{k=1}^{K} p_k \boldsymbol{B}_{t-1}^k$.
- Re-Initial reinitializes the LoRA parameters for each client k in every communication round t by using the formulation in Equation 9 and updates each client's local pre-trained matrix by adding the aggregated LoRA parameters, denoted as $\boldsymbol{W}'_{t,k} = \boldsymbol{W}'_{t-1,k} + \boldsymbol{B}_t\boldsymbol{A}_t$, where $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ are aggregated matrices from all clients.
- Local-Initial randomly selects one client c's local LoRA parameters from the previous round $t-1$ as the initialization for all clients in the next round t, represented as $\boldsymbol{A}_t^k = \boldsymbol{A}_{t-1}^c, \boldsymbol{B}_t^k = \boldsymbol{B}_{t-1}^c, \forall k \in [K]$.

Experiments demonstrated that Avg-Initial yields the best performance due to its ability to balance continuity and unification across clients for reducing initialization drift. Building on this insight, LoRA-FAIR maintained separate aggregation for $\boldsymbol{A}$ and $\boldsymbol{B}$ while learning a corrective term $\Delta\boldsymbol{B}$ to address the discordance introduced by separate aggregation.

Addtionaly, FedLoRU [Park and Klabjan, 2024] introduced a momentum-based initialization approach, extending the Re-Initial strategy from every round to periodic every τ rounds. Specifically, after each τ rounds, the server aggregates low-rank updates from clients to compute the global update $\boldsymbol{B}\boldsymbol{A}$, accumulates this global update with the pre-trained weight $\boldsymbol{W}'$, and reinitializes $\boldsymbol{A}$ and $\boldsymbol{B}$ for the next cycle. The global model at communication round T can be expressed as:

$$\boldsymbol{W}_T = \boldsymbol{W}' + \sum_{\substack{t=1 \\ t \bmod \tau = 0}}^{T} \boldsymbol{B}_t\boldsymbol{A}_t. \tag{11}$$

This approach constrains client-side optimization to a low-rank subspace by reinitializing LoRA parameters every τ communication rounds, and tailors the global model to a higher-rank space by accumulating updates from previous rounds, effectively balancing the trade-off between local adaptability and global consistency.

3.3 Discussion

Effective LoRA adaptation in FL relies on advanced aggregation and initialization, and there are already numerous studies addressing these challenges as summarized in Table 1. Despite their effectiveness, gaps remain in the determination of importance weighting for aggregation and the theoretical understanding of its impact on model convergence and stability, which prompts future research to explore more adaptive aggregation mechanisms and establish theoretical guarantees for robustness and efficiency in FedLoRA.

4 Heterogeneity

Heterogeneity is another key challenge in FL, mainly arising from non-IID data, which could lead to performance degradation and slower convergence. To address this, personalization is introduced to tailor global models to individual client's alignment. Early work [Yi *et al.*, 2023] adapted LoRA as a personalization method for heterogeneous FL, which treats full-size parameters as client-specific and aggregates LoRA globally. However, as models scale, recent research has shifted toward advanced personalization techniques that focus exclusively on adapting LoRA while keeping the rest model frozen, ensuring both computational efficiency and effective learning, and we detail these methods in this section.

4.1 Heterogeneous Rank

Considering the heterogeneous system capabilities and data distributions in FL, HETLORA [Cho *et al.*, 2024] introduced heterogeneous LoRA ranks, allowing each client to select a personalized rank based on its task complexity and available computational resources. These heterogeneous LoRA are efficiently aggregated and distributed by local rank self-pruning and sparsity-weighted aggregation at server with objective:

$$\min_{\{\boldsymbol{A}_k,\boldsymbol{B}_k\}} \sum_{k=1}^{K} p_k f_k(\boldsymbol{A}_k, \boldsymbol{B}_k, \boldsymbol{W}'; D_k). \tag{12}$$

Building on this, subsequent research [Chen *et al.*, 2024a; Byun and Lee, 2024] enhanced it by replacing zero-padding strategy with replication-based padding strategy during aggregation, which better preserves valuable information from clients with high-quality data for better performance. Moreover, full-size matrix aggregation and rank clustering methods (Section 3.1) can naturally accommodate heterogeneous ranks for personalization, because full-size matrix $\Delta\boldsymbol{W}$ retains a fixed size regardless of individual ranks and rank clustering treats rank as the fundamental unit for aggregation, ensuring the compatibility of heterogeneous rank in FedLoRA.

4.2 Personalized LoRA

Another promising approach for personalization is to introduce an additional personalized LoRA alongside the global LoRA, enhancing client-specific adaptation while preserving global knowledge sharing. This section categorizes methods by how personalized LoRA is obtained and integrated in FL.

Dual-LoRA. As personalization and global learning often aim to align with different data distributions, learning an additional personalized model is proposed to tackle this challenge. Specifically, for each client k, the framework involves learning both a global model $\boldsymbol{\theta}$ to capture shared knowledge across all clients and a personalized model $\boldsymbol{\theta}_k$ tailored to the client's unique data. This dual-objective can be formulated:

$$\min_{\boldsymbol{\theta}_k} f_k(\boldsymbol{\theta}^*, \boldsymbol{\theta}_k; D_k) \text{ s.t., } \boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \sum_{k=1}^{K} p_k f_k(\boldsymbol{\theta}; D_k), \tag{13}$$

FedDPA [Yang *et al.*, 2024] first proposed a dual-adapter framework in which one global LoRA $\boldsymbol{\theta} = (\boldsymbol{A}, \boldsymbol{B})$ is trained and aggregated across clients to capture global knowledge, while another LoRA $\boldsymbol{\theta}_k = (\boldsymbol{A}_k, \boldsymbol{B}_k)$ is locally trained for personalization without communication. These two LoRAs are dynamically combined during inference to balance global generalization and local adaptation. Similarly, FDLoRA [Qi *et al.*, 2024] adopted a dual-LoRA framework but incorporated periodic synchronization between the personalized and global LoRAs every few rounds, and a novel adaptive fusion method to merge these LoRAs. Building on the symmetry analysis of LoRA matrices, FedSA-LoRA [Guo *et al.*, 2024a] proposed a novel approach where $\boldsymbol{\theta} = \boldsymbol{A}$ is for global aggregation and knowledge sharing, while $\boldsymbol{\theta}_k = \boldsymbol{B}_k$ is reserved for local personalization, which not only addresses the personalization but also resolves the LoRA aggregation discordance.

Heterogeneous Structure. Diverging from previous works that optimize personalized LoRA with a bi-level objective, PerFIT [Zhang *et al.*, 2024c] employed pruning-oriented neural architecture search (NAS) to discover a personalized LoRA structure tailored to each client and transformed the global aggregated LoRA into personalized LoRA with the NAS-searched structures, eliminating the need for additional optimization objectives. Therefore, for K clients with LoRA $\boldsymbol{\theta} = (\boldsymbol{A}, \boldsymbol{B})$, the objective can be formulated as follows:

$$\min_{\boldsymbol{\theta}, \{\mathcal{A}_k\}} \sum_{k=1}^{K} p_k f_k(\boldsymbol{\theta}(\mathcal{A}_k); D_k) \quad \text{s.t.,}\ R_k(\mathcal{A}_k) \le B_k, \tag{14}$$

where $\mathcal{A}_k$ is the personalized architecture of client k, and R_k and B_k represent the resource consumption and budget limitation for client k. More recently, researchers [Zhang *et al.*, 2024c; Mei *et al.*, 2024] have explored mixture-of-experts with LoRA, enabling dynamic selection and combination of expert LoRAs, with $\boldsymbol{\theta} = \{\boldsymbol{A}_i, \boldsymbol{B}_i\}_{i=1}^{N}$ denoting the set of expert LoRAs and $\mathcal{A}_k$ denoting the selected experts for client k, enhancing both global performance and personalization.

Hypernetwork. Beyond previously discussed approaches, HyperFloRA [Lu *et al.*, 2024] introduced hypernetworks to generate personalized LoRA parameters for each client based on its unique representation vector. Specifically, each client k is assigned an indicator vector $\boldsymbol{r}_k$ derived from its local data D_k, and the server trains the hypernetwork ψ to generate personalized LoRA for each client with the following objective:

$$\min_{\psi} \sum_{k=1}^{K} p_k f_k(h(\psi; \boldsymbol{r}_k); D_k). \tag{15}$$

This enables HyperFloRA to efficiently personalize models, particularly advantageous for training-incapable clients.

4.3 Clustering

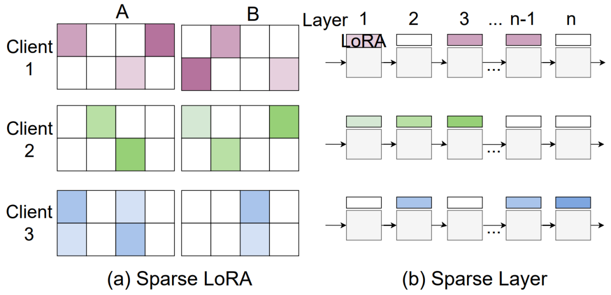
Clustering-based approaches offer another promising solution for addressing heterogeneity in FL by grouping clients with similar data distributions or preferences. In this framework, clients are assigned to N clusters corresponding to a set of LoRA $\{\boldsymbol{\theta}_n = (\boldsymbol{A}_n, \boldsymbol{B}_n)\}_{n=1}^{N}$ for optimization:

$$\begin{aligned} &\min_{\{\boldsymbol{\theta}_n\}} \sum_{k=1}^{K} \sum_{n=1}^{N} \alpha_{k,n} p_k f_k(\boldsymbol{\theta}_n; D_k) \\ &\text{s.t.,} \quad \alpha_{k,n} \in \arg\min_{\alpha_{k,n}} \sum_{k=1}^{K} \sum_{n=1}^{N} \alpha_{k,n} d(\boldsymbol{\theta}_k, \boldsymbol{\theta}_n), \end{aligned} \tag{16}$$

where $\alpha_{k,n}$ is the assignment matrix ($\alpha_{k,n} = 1$ if $k \in n$ else $\alpha_{k,n} = 0$), $\boldsymbol{\theta}_n$ is the centroid of cluster n, and d is the distance function to measure the distance between a client's parameter and the cluster centroid. Recognizing that data from the same task shares similar distributions, FL-TAC [Ping *et al.*, 2024] proposed a server-side clustering approach to group similar LoRA modules, enabling task-specific aggregation and improving model performance for diverse tasks. Similarly, FedLFC [Guo *et al.*, 2024c] introduced a clustering strategy for LoRA modules based on different languages, effectively addressing the challenges of Multilingual FL by tailoring the aggregation process to language-specific characteristics. FedHLT [Guo *et al.*, 2024b] extended FedLFC by incorporating a hierarchical language tree for Multilingual FL, where the server maintains a set of LoRA parameters for each node in the language tree and aggregates clients' LoRA based on their positions in the tree, ensuring more structured and efficient knowledge sharing across related languages.

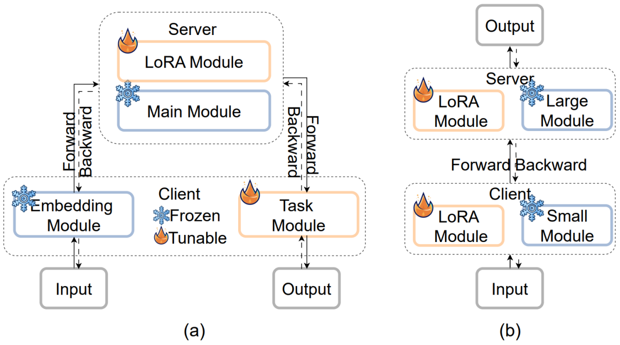
4.4 Discussion

Personalization has emerged as a promising approach to mitigate heterogeneity in FedLoRA, where each approach presents trade-offs as summarized in Table 1. Future research should focus on developing more LoRA-derived personalization strategies like enabling automatic rank selection, as well as enhancing the generalization ability of personalized FedLoRA across diverse client distributions to align with the versatile demands of foundation models in FL applications.



(a) Sparse LoRA (b) Sparse Layer

Figure 3: Two types of sparse learning frameworks in FedLoRA.



(a) (b)

Figure 4: Two types of split learning frameworks in FedLoRA.

5 Efficiency

Despite FedLoRA's efficiency in learning with fewer parameters, modern foundation models still pose significant storage, communication, and computational challenges. Recent research addresses these issues by developing advanced efficiency-enhancing methods, which can be categorized into three key aspects based on the technologies employed.

Sparse Learning. Sparse learning is an effective approach to enhancing the efficiency of learning in FL by identifying and utilizing only the most essential parameters. One type of sparse learning focuses on applying sparsity at the LoRA parameter level, as illustrated in Figure 3 (a). FLASC [Kuo *et al.*, 2024] first employed pruning methods to transmit sparsified LoRA parameters, reducing communication overhead while allowing clients to locally fine-tune the entire LoRA module for superior utility with minimal computational overhead compared to sparse tuning methods. To further balance efficiency and heterogeneity in FedLoRA, HAFL [Su *et al.*, 2024] introduced a method that selectively updates only the most important decomposed rank-1 LoRA matrices while keeping the rest frozen, allowing clients to have heterogeneous ranks for resource-aware optimization and task-specific alignment. Another type of sparse learning applies sparsity at the layer level, as shown in Figure 3 (b). To enhance both communication and computation efficiency, FibecFed [Liu *et al.*, 2024] and Fed-piLot [Zhang *et al.*, 2024d] proposed selecting the most important layers of LoRA for tuning and aggregation based on Fisher Information or Local-Global Information Gain Scores. Similarly, FedFMSL [Wu *et al.*, 2024c] introduced a sparsely activated LoRA layer framework, which progressively tunes specific LoRA layers over communication rounds based on training accuracy, optimizing resource usage dynamically. To better align with clients' resource constraints in real-world applications, FedRA [Su *et al.*, 2025] proposed randomly selecting layers to construct client-specific local models, tailored to their computational and storage capabilities using an allocation matrix.

Split Learning. As resource-constrained clients struggle to maintain and tune large models locally, the study [Wang *et al.*, 2024a] integrated a split learning framework into FedLoRA as shown in Figure 4 (a), where only the embedding and task-specific modules are retained on clients, while the main model body, comprising the majority of parameters, is hosted on the server for efficient training. Similarly, FedsLLM [Zhao *et al.*, 2024] and SplitLoRA [Lin *et al.*, 2024] also extended FedLoRA with split learning while distributing the first several layers of model to clients, placing the majority of the remaining layers to the server, and introducing a local aggregation server for efficient client-side LoRA aggregation, as shown in Figure 4 (b).

Compression. In addition to previous methods, CG-FedLLM [Wu *et al.*, 2024b] introduced an autoencoder-based compression framework, where clients encode gradient features locally and transmit compressed representations to the server for decoding, effectively reducing communication overhead. Similarly, FedBiOT [Wu *et al.*, 2024a] applied model compression techniques to obtain a compact model for tuning in FedLoRA, further leveraging knowledge distillation to align its performance with that of a fully fine-tuned model.

Discussion. With the increasing scale of foundation models, recent research has focused on adapting advanced efficiency methods in FedLoRA to address computational

	Methods	Advantages	Disadvantages	Reference
Distributed Learning	FedAVG	Simple, widely used in FL	Causes LoRA aggregation discordance	[McMahan *et al.*, 2017]
	Single Low-Rank Matrix	Simple, efficient	Slow convergence	[Sun *et al.*, 2024; Nguyen *et al.*, 2024], [Chen *et al.*, 2024b; Koo *et al.*, 2024]
	Full-size Matrix	Faster convergence, supports heterogeneous ranks	Higher computational costs	[Bai *et al.*, 2024; Fang *et al.*, 2024], [Wang *et al.*, 2024b]
	Corrective Mechanism	Improves initialization consistency	High computation overhead, additional parameters	[Bian *et al.*, 2024; Singhal *et al.*, 2024]
	Rank Clustering	Fine-grained aggregation, supports heterogeneous rank	Additional hyperparameters, high computation costs	[Qin and Li, 2024]
	Server Standard Initial	Simple to implement	Introduce weight update drift	[Hu *et al.*, 2021]
	Server Data-Driven Initial	Enhanced data alignment	Additional computation costs	[Babakniya *et al.*, 2023; Yan *et al.*, 2024]
	Client Avg-Initial	Balances stability and adaptability	LoRA aggregation discordance	[Bian *et al.*, 2024]
	Client Re-Initial	Learn of a higher-rank space	Introduce initialization drift	[Bian *et al.*, 2024; Park and Klabjan, 2024]
	Client Local-Initial	Customizes initialization	Lack global consistency	[Bian *et al.*, 2024]
Heterogeneity	Heterogeneous Rank	Simple, supports resource heterogeneity	Difficult to automatically select rank	[Cho *et al.*, 2024; Chen *et al.*, 2024a], [Byun and Lee, 2024]
	Dual-LoRA	Balances personalization and global learning	Additional parameters	[Yang *et al.*, 2024; Qi *et al.*, 2024], [Guo *et al.*, 2024a]
	Heterogeneous Structure	Offers flexibility, supports resource heterogeneity	Implementation complexity increases with model size	[Zhang *et al.*, 2024c; Zhang *et al.*, 2024c], [Mei *et al.*, 2024]
	Hypernetwork	Generalized to new clients	Difficult to optimize and train effectively	[Lu *et al.*, 2024]
	Cluster	Good for application	High computation costs, additional hyperparameters	[Ping *et al.*, 2024; Guo *et al.*, 2024c], [Guo *et al.*, 2024b]
Efficiency	Sparse Learning	Advanced LoRA specific techs	Slow convergence	[Kuo *et al.*, 2024; Su *et al.*, 2024], [Liu *et al.*, 2024; Zhang *et al.*, 2024d], [Wu *et al.*, 2024c; Su *et al.*, 2025]
	Split Learning	Good for large model's application	Synchronization and load-balancing issues	[Wang *et al.*, 2024a; Zhao *et al.*, 2024], [Lin *et al.*, 2024]
	Compression	Compatible with existing FL frameworks	Risk performance degradation if improper optimized	[Wu *et al.*, 2024b; Wu *et al.*, 2024a]

Table 1: Comparison of different FedLoRA methods for distributed learning, heterogeneity and Efficiency.

and communication challenges, as summarized in Table 1. Despite these, FwdLLM [Xu *et al.*, 2024] introduced a backpropagation-free approach in FedLoRA for computational efficiency by replacing backpropagation with perturbed inferences and output validation. However, further research could explore more advanced methods, particularly LoRA-specific optimizations for large-scale FL scenarios, and also consider real-world applications for FedLoRA like healthcare analytics to enhance scalability and practical deployment.

6 Future Directions

Theory Analysis. Existing work has empirically validated the effectiveness of various FedLoRA methods, but their theoretical convergence remains underexplored. Recently, a study [Malinovsky *et al.*, 2024] analyzed the convergence rates of FedLoRA with different optimizers, and the other work [Mahla and Ramakrishnan, 2024] highlighted the potential instability of previous FedLoRA methods via convergence analysis. While these provide initial theoretical insights, further research is needed to rigorously examine FedLoRA's convergence properties, particularly under heterogeneous settings, considering factors such as different aggregation algorithms, personalization models, and initializations.

LoRA-derived Methods. Recently, various enhanced LoRA methods (Section 2.2) have emerged. These advancements encourage future research to adapt them within FL to develop more sophisticated FedLoRA approaches. For example, LoRA with adaptive rank adjustment [Valipour *et al.*, 2022] could be employed to dynamically adjust the LoRA rank with individual client's need for heterogeneity, while pruning enhanced LoRA [Dettmers *et al.*, 2024] could be integrated into FedLoRA to further reduce communication and computational costs. Future research could explore the integration of these enhanced LoRA methods within FedLoRA and propose derived frameworks to further improve efficiency and performance.

Unified Benchmark. As LoRA is a PEFT method for large foundation models, datasets and models used in these foundation models are highly diverse, leading to significant benchmark variations in FedLoRA studies. Unlike conventional FL, which often uses standardized datasets like ImageNet [Deng *et al.*, 2009], FedLoRA lacks a unified benchmark and metrics for fair comparison of existing approaches. Moreover, its definition of heterogeneity remains underexplored, as foundation models face broader heterogeneity beyond label distribution shifts [Ren *et al.*, 2024]. Future research should focus on standardizing benchmarks and refining heterogeneity definitions for fair evaluation and broader applicability.

Various Application. While FedLoRA has shown promise in general machine learning tasks like image classification and language understanding, its potential in other domains remains underexplored. Recent studies [Zhang *et al.*, 2024a; Nguyen *et al.*, 2024] have applied FedLoRA to recommendation systems, enabling personalized models with LoRA learning for efficiency at scale. Beyond this, FedLoRA could be leveraged in other areas, like weather prediction and financ to efficiently process time series data with long history. Further exploration of FedLoRA in these domains could unlock new opportunities for efficient and secure model development.

7 Conclusion

As modern foundation models grow in size and complexity, FedLoRA has been proposed by integrating LoRA into FL for efficient learning. This paper provides a comprehensive survey of FedLoRA, primarily focusing on distributed learning, heterogeneity and efficiency challenges, further categorizing existing research into granular subcategories based on applied technologies. For each subcategory, we introduce underlying methodologies with detailed mathematical formulations and Figures, comparing their advantages and limitations. Finally, we discuss promising future directions for FedLoRA to guide further research and development.

References

[Almanifi *et al.*, 2023] Omair Rashed Abdulwareth Almanifi, Chee-Onn Chow, Mau-Luen Tham, Joon Huang Chuah, and Jeevan Kanesan. Communication and computation efficiency in federated learning: A survey. *Internet of Things*, 22:100742, 2023.

[Babakniya *et al.*, 2023] Sara Babakniya, Ahmed Roushdy Elkordy, Yahya H Ezzeldin, Qingfeng Liu, Kee-Bong Song, Mostafa El-Khamy, and Salman Avestimehr. Slora: Federated parameter efficient fine-tuning of language models. *arXiv preprint arXiv:2308.06522*, 2023.

[Bai *et al.*, 2024] Jiamu Bai, Daoyuan Chen, Bingchen Qian, Liuyi Yao, and Yaliang Li. Federated fine-tuning of large language models under heterogeneous language tasks and client resources. *arXiv e-prints*, pages arXiv–2402, 2024.

[Bian *et al.*, 2024] Jieming Bian, Lei Wang, Letian Zhang, and Jie Xu. Lora-fair: Federated lora fine-tuning with aggregation and initialization refinement. *arXiv preprint arXiv:2411.14961*, 2024.

[Byun and Lee, 2024] Yuji Byun and Jaeho Lee. Towards federated low-rank adaptation of language models with rank heterogeneity. *arXiv preprint arXiv:2406.17477*, 2024.

[Chen *et al.*, 2024a] Shuaijun Chen, Omid Tavallaie, Niousha Nazemi, and Albert Y Zomaya. Rbla: Rank-based-lora-aggregation for fine-tuning heterogeneous models in flaas. In *International Conference on Web Services*, pages 47–62. Springer, 2024.

[Chen *et al.*, 2024b] Shuangyi Chen, Yue Ju, Hardik Dalal, Zhongwen Zhu, and Ashish Khisti. Robust federated finetuning of foundation models via alternating minimization of lora. *arXiv preprint arXiv:2409.02346*, 2024.

[Cho *et al.*, 2024] Yae Jee Cho, Luyang Liu, Zheng Xu, Aldi Fahrezi, and Gauri Joshi. Heterogeneous lora for federated fine-tuning of on-device foundation models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12903–12913, 2024.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[Dettmers *et al.*, 2024] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36, 2024.

[Fang *et al.*, 2024] Zihan Fang, Zheng Lin, Zhe Chen, Xianhao Chen, Yue Gao, and Yuguang Fang. Automated federated pipeline for parameter-efficient fine-tuning of large language models. *arXiv preprint arXiv:2404.06448*, 2024.

[Guo *et al.*, 2024a] Pengxin Guo, Shuang Zeng, Yanran Wang, Huijie Fan, Feifei Wang, and Liangqiong Qu. Selective aggregation for low-rank adaptation in federated learning. *arXiv preprint arXiv:2410.01463*, 2024.

[Guo *et al.*, 2024b] Zhihan Guo, Yifei Zhang, Zhuo Zhang, Zenglin Xu, and Irwin King. Fedhlt: Efficient federated low-rank adaption with hierarchical language tree for multilingual modeling. In *Companion Proceedings of the ACM on Web Conference 2024*, pages 1558–1567, 2024.

[Guo *et al.*, 2024c] Zhihan Guo, Yifei Zhang, Zhuo Zhang, Zenglin Xu, and Irwin King. Fedlfc: Towards efficient federated multilingual modeling with lora-based language family clustering. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1519–1528, 2024.

[Hu *et al.*, 2021] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[Koo *et al.*, 2024] Jabin Koo, Minwoo Jang, and Jungseul Ok. Towards robust and efficient federated low-rank adaptation with heterogeneous clients. *arXiv preprint arXiv:2410.22815*, 2024.

[Kuang *et al.*, 2024] Weirui Kuang, Bingchen Qian, Zitao Li, Daoyuan Chen, Dawei Gao, Xuchen Pan, Yuexiang Xie, Yaliang Li, Bolin Ding, and Jingren Zhou. Federatedscope-llm: A comprehensive package for fine-tuning large language models in federated learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5260–5271, 2024.

[Kuo *et al.*, 2024] Kevin Kuo, Arian Raje, Kousik Rajesh, and Virginia Smith. Federated lora with sparse communication. *arXiv preprint arXiv:2406.05233*, 2024.

[Lin *et al.*, 2024] Zheng Lin, Xuanjie Hu, Yuxin Zhang, Zhe Chen, Zihan Fang, Xianhao Chen, Ang Li, Praneeth Vepakomma, and Yue Gao. Splitlora: A split parameter-efficient fine-tuning framework for large language models. *arXiv preprint arXiv:2407.00952*, 2024.

[Liu *et al.*, 2024] Ji Liu, Jiaxiang Ren, Ruoming Jin, Zijie Zhang, Yang Zhou, Patrick Valduriez, and Dejing Dou. Fisher information-based efficient curriculum federated learning with large language models. *arXiv preprint arXiv:2410.00131*, 2024.

[Lu *et al.*, 2024] Qikai Lu, Di Niu, Mohammadamin Samadi Khoshkho, and Baochun Li. Hyperflora: Federated learning with instantaneous personalization. In *Proceedings of the 2024 SIAM International Conference on Data Mining (SDM)*, pages 824–832. SIAM, 2024.

[Mahla and Ramakrishnan, 2024] Navyansh Mahla and Ganesh Ramakrishnan. Why gradient subspace? identifying and mitigating lora's bottlenecks in federated fine-tuning of large language models. *arXiv preprint arXiv:2410.23111*, 2024.

[Malinovsky *et al.*, 2024] Grigory Malinovsky, Umberto Michieli, Hasan Abed Al Kader Hammoud, Taha Ceritli, Hayder Elesedy, Mete Ozay, and Peter Richtárik. Randomized asymmetric chain of lora: The first meaningful theoretical framework for low-rank adaptation. *arXiv preprint arXiv:2410.08305*, 2024.

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

[Mei *et al.*, 2024] Hanzi Mei, Dongqi Cai, Ao Zhou, Shangguang Wang, and Mengwei Xu. Fedmoe: Personalized federated learning via heterogeneous mixture of experts. *arXiv preprint arXiv:2408.11304*, 2024.

[Nguyen *et al.*, 2024] Ngoc-Hieu Nguyen, Tuan-Anh Nguyen, Tuan Nguyen, Vu Tien Hoang, Dung D Le, and Kok-Seng Wong. Towards efficient communication and secure federated recommendation system via low-rank training. In *Proceedings of the ACM on Web Conference 2024*, pages 3940–3951, 2024.

[Park and Klabjan, 2024] Haemin Park and Diego Klabjan. Communication-efficient federated low-rank update algorithm and its connection to implicit regularization. *arXiv preprint arXiv:2409.12371*, 2024.

[Ping *et al.*, 2024] Siqi Ping, Yuzhu Mao, Yang Liu, Xiao-Ping Zhang, and Wenbo Ding. Fl-tac: Enhanced fine-tuning in federated learning via low-rank, task-specific adapter clustering. *arXiv preprint arXiv:2404.15384*, 2024.

[Qi *et al.*, 2024] Jiaxing Qi, Zhongzhi Luan, Shaohan Huang, Carol Fung, Hailong Yang, and Depei Qian. Fdlora: Personalized federated learning of large language model via dual lora tuning. *arXiv preprint arXiv:2406.07925*, 2024.

[Qin and Li, 2024] Huangsiyuan Qin and Ying Li. Fedinc: One-shot federated tuning for collaborative incident recognition. In *International Conference on Artificial Neural Networks*, pages 174–185. Springer, 2024.

[Ren *et al.*, 2024] Chao Ren, Han Yu, Hongyi Peng, Xiaoli Tang, Anran Li, Yulan Gao, Alysa Ziying Tan, Bo Zhao, Xiaoxiao Li, Zengxiang Li, et al. Advances and open challenges in federated learning with foundation models. *arXiv e-prints*, pages arXiv–2404, 2024.

[Singhal *et al.*, 2024] Raghav Singhal, Kaustubh Ponkshe, and Praneeth Vepakomma. Exact aggregation for federated and efficient fine-tuning of foundation models. *arXiv preprint arXiv:2410.09432*, 2024.

[Su *et al.*, 2024] Yang Su, Na Yan, and Yansha Deng. Federated llms fine-tuned with adaptive importance-aware lora. *arXiv preprint arXiv:2411.06581*, 2024.

[Su *et al.*, 2025] Shangchao Su, Bin Li, and Xiangyang Xue. Fedra: A random allocation strategy for federated tuning to unleash the power of heterogeneous clients. In *European Conference on Computer Vision*, pages 342–358. Springer, 2025.

[Sun *et al.*, 2024] Youbang Sun, Zitao Li, Yaliang Li, and Bolin Ding. Improving lora in privacy-preserving federated learning. *arXiv preprint arXiv:2403.12313*, 2024.

[Valipour *et al.*, 2022] Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. Dylora: Parameter efficient tuning of pre-trained models using dynamic search-free low-rank adaptation. *arXiv preprint arXiv:2210.07558*, 2022.

[Wang *et al.*, 2024a] Zixin Wang, Yong Zhou, Yuanming Shi, Khaled Letaief, et al. Federated fine-tuning for pre-trained foundation models over wireless networks. *arXiv preprint arXiv:2407.02924*, 2024.

[Wang *et al.*, 2024b] Ziyao Wang, Zheyu Shen, Yexiao He, Guoheng Sun, Hongyi Wang, Lingjuan Lyu, and Ang Li. Flora: Federated fine-tuning large language models with heterogeneous low-rank adaptations. *arXiv preprint arXiv:2409.05976*, 2024.

[Woisetschläger *et al.*, 2024] Herbert Woisetschläger, Alexander Isenko, Shiqiang Wang, Ruben Mayer, and Hans-Arno Jacobsen. A survey on efficient federated learning methods for foundation model training. *arXiv preprint arXiv:2401.04472*, 2024.

[Wu *et al.*, 2024a] Feijie Wu, Zitao Li, Yaliang Li, Bolin Ding, and Jing Gao. Fedbiot: Llm local fine-tuning in federated learning without full model. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3345–3355, 2024.

[Wu *et al.*, 2024b] Huiwen Wu, Xiaohan Li, Deyi Zhang, Xiaogang Xu, Jiafei Wu, Puning Zhao, and Zhe Liu. Cg-fedllm: How to compress gradients in federated fune-tuning for large language models. *arXiv preprint arXiv:2405.13746*, 2024.

[Wu *et al.*, 2024c] Panlong Wu, Kangshuo Li, Ting Wang, Yanjie Dong, Victor CM Leung, and Fangxin Wang. Fedfmsl: Federated learning of foundations models with sparsely activated lora. *IEEE Transactions on Mobile Computing*, 2024.

[Xu *et al.*, 2024] Mengwei Xu, Dongqi Cai, Yaozong Wu, Xiang Li, and Shangguang Wang. Fwdllm: Efficient fedllm using forward gradient. *arXiv. Available at: hjp://arxiv. org/abs/2308.13894 (Accessed: 11 March 2024)*, 2024.

[Yan *et al.*, 2024] Yuxuan Yan, Qianqian Yang, Shunpu Tang, and Zhiguo Shi. Federa: Efficient fine-tuning of language models in federated learning leveraging weight decomposition. *arXiv preprint arXiv:2404.18848*, 2024.

[Yang *et al.*, 2023] Adam X Yang, Maxime Robeyns, Xi Wang, and Laurence Aitchison. Bayesian low-rank adaptation for large language models. *arXiv preprint arXiv:2308.13111*, 2023.

[Yang *et al.*, 2024] Yiyuan Yang, Guodong Long, Tao Shen, Jing Jiang, and Michael Blumenstein. Dual-personalizing adapter for federated foundation models. *arXiv preprint arXiv:2403.19211*, 2024.

[Yi *et al.*, 2023] Liping Yi, Han Yu, Gang Wang, and Xiaoguang Liu. Fedlora: Model-heterogeneous personalized federated learning with lora tuning. *arXiv preprint arXiv:2310.13283*, 2023.

[Yu *et al.*, 2023] Sixing Yu, J Pablo Muñoz, and Ali Jannesari. Federated foundation models: Privacy-preserving and collaborative learning for large models. *arXiv preprint arXiv:2305.11414*, 2023.

[Zhang *et al.*, 2021] Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, and Yuan Gao. A survey on federated learning. *Knowledge-Based Systems*, 216:106775, 2021.

[Zhang *et al.*, 2023] Zhuo Zhang, Yuanhang Yang, Yong Dai, Qifan Wang, Yue Yu, Lizhen Qu, and Zenglin Xu. Fedpetuning: When federated learning meets the parameter-efficient tuning methods of pre-trained language models. In *Annual Meeting of the Association of Computational Linguistics 2023*, pages 9963–9977. Association for Computational Linguistics (ACL), 2023.

[Zhang *et al.*, 2024a] Chunxu Zhang, Guodong Long, Hongkuan Guo, Xiao Fang, Yang Song, Zhaojie Liu, Guorui Zhou, Zijian Zhang, Yang Liu, and Bo Yang. Federated adaptation for foundation model-based recommendations. *arXiv preprint arXiv:2405.04840*, 2024.

[Zhang *et al.*, 2024b] Jianyi Zhang, Saeed Vahidian, Martin Kuo, Chunyuan Li, Ruiyi Zhang, Tong Yu, Guoyin Wang, and Yiran Chen. Towards building the federatedgpt: Federated instruction tuning. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6915–6919. IEEE, 2024.

[Zhang *et al.*, 2024c] Pengyu Zhang, Yingbo Zhou, Ming Hu, Junxian Feng, Jiawen Weng, and Mingsong Chen. Personalized federated instruction tuning via neural architecture search. *arXiv preprint arXiv:2402.16919*, 2024.

[Zhang *et al.*, 2024d] Zikai Zhang, Jiahao Xu, Ping Liu, and Rui Hu. Fed-pilot: Optimizing lora assignment for efficient federated foundation model fine-tuning. *arXiv preprint arXiv:2410.10200*, 2024.

[Zhao *et al.*, 2024] Kai Zhao, Zhaohui Yang, Chongwen Huang, Xiaoming Chen, and Zhaoyang Zhang. Fedsllm: Federated split learning for large language models over communication networks. In *2024 International Conference on Ubiquitous Communication (Ucom)*, pages 438–443. IEEE, 2024.

[Zhuang *et al.*, 2023] Weiming Zhuang, Chen Chen, and Lingjuan Lyu. When foundation model meets federated learning: Motivations, challenges, and future directions. *arXiv preprint arXiv:2306.15546*, 2023.