FedLoVA: Value-Only Aggregation for Federated LoRA Fine-Tuning of Large Language Models

Ensieh Khazaei
University of Toronto
ensieh.khazaei@mail.utoronto.ca

Baturalp Buyukates
University of Birmingham
b.buyukates@bham.ac.uk

Dimitrios Hatzinakos
University of Toronto
dimitris@comm.utoronto.ca

Abstract

Low-Rank Adaptation (LoRA) is a popular parameter-efficient fine-tuning (PEFT) method for fine-tuning of large language models (LLMs), especially in federated learning, due to its strong performance and communication efficiency. In practice, LoRA is applied to the query and value projections of transformer layers without considering the distinct roles of these components. In this work, we analyze the impact of these projections in federated fine-tuning of LLMs using LoRA and observe that the value projection converges significantly faster than the query projection. Motivated by this finding, we propose FedLoVA, a federated fine-tuning framework that trains LoRA adapters on both query and value projections locally while sharing only the value projection with the server. This approach reduces communication overhead while improving convergence efficiency. Experiments on natural language understanding, multilingual sentiment analysis, and generation tasks show that FedLoVA reduces the communication cost by up to $7.8\times$ compared to existing federated LLM fine-tuning baselines while achieving comparable or better performance. The code is available at: https://github.com/EnsiehKhazaei/FedLoVA.

1 Introduction

Large Language Models (LLMs) have emerged in recent years and demonstrated remarkable performance in various applications, including conversational systems, finance, and healthcare [38, 9, 37]. To achieve strong performance in these applications, LLMs typically require fine-tuning on task-specific downstream datasets. However, due to their massive number of parameters, full fine-tuning of LLMs requires substantial computational resources and memory capacity, limiting their practical deployment in real-world scenarios with scarce resources.

To address the high computational and memory costs associated with full fine-tuning of LLMs, Parameter-Efficient Fine-Tuning (PEFT) methods have been proposed as promising solutions. Instead of updating all model parameters during fine-tuning, PEFT methods train only a small set of parameters. Some PEFT approaches include Adapter Tuning [20], Prefix-Tuning [33], and Low-Rank Adaptation (LoRA) [34, 29]. Among these methods, LoRA has gained significant attention due to its simplicity and effectiveness. LoRA decomposes weight updates into low-rank matrices A and B, and learns these matrices instead of updating the full parameters. This technique achieves competitive performance compared to full model fine-tuning while reducing the number of trainable parameters.

Although LoRA reduces the number of parameters involved in fine-tuning, applying it in resource-constrained environments such as Federated Learning (FL) frameworks remains challenging. In such

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

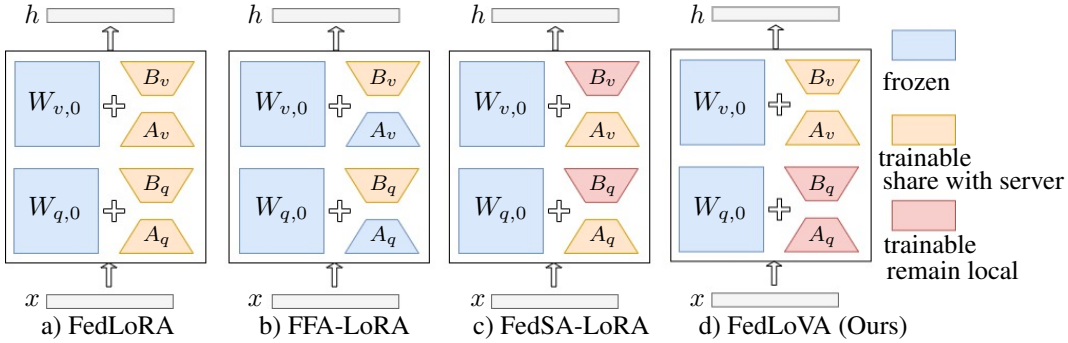


Figure 1: Illustration of communication and update strategies in a) FedLoRA, b) FFA-LoRA, c) FedSA-LoRA, and d) the proposed FedLoVA. FedLoRA updates and shares both A and B for query and value projections. FFA-LoRA freezes A and only updates and shares B. FedSA updates both A and B locally but shares only A, while FedLoVA updates both A and B and shares value projection modules A_v and B_v.

settings, limited computational resources, communication overhead, client heterogeneity, and privacy concerns must be considered. To address these challenges, several studies have explored federated fine-tuning of LLMs using LoRA [36, 2, 1]. Among these, FFA-LoRA [28] freezes the A matrix during training and aggregates only the B matrix across clients. However, freezing A can limit the performance and adaptability of the FL framework. Another approach, FedSA-LoRA [10], trains both A and B locally but shares only A with the server, based on the intuition that A captures global knowledge while B captures personalized knowledge.

Despite these prior studies investigating the roles of LoRA matrices A and B in FL frameworks, it remains underexplored how different attention components, specifically the query (Q) and value (V) projections, affect global and local performance in federated settings. Understanding the roles of these attention modules is crucial in federated fine-tuning of LLMs, where communication efficiency, convergence behavior, and privacy preservation are key constraints. Identifying which attention modules are most beneficial to share with the global server versus to keep local can directly lead to more efficient and effective federated fine-tuning approaches.

In this work, we provide a comprehensive study of attention submodules in federated LLM fine-tuning, quantifying their impact on global generalization and local adaptation. In particular, we analyze how sharing or localizing each component affects performance, convergence behavior, and discuss potential implications for privacy in FL frameworks. Through this analysis, we identify the value (V) projection in the attention mechanism as the most critical module for aggregation among clients. Based on this insight, we propose *FedLoVA (Federated LoRA with Value Aggregation)*, a communication-efficient framework for federated fine-tuning of LLMs using LoRA. FedLoVA applies LoRA adapters to both the query (Q) and value (V) projections within each transformer layer, trains both modules locally and shares only the value-related adapter parameters (A_v and B_v matrices) with the server for aggregation, as shown in Figure 1. This sharing strategy reduces communication overhead and limits the information shared with the server, although we do not claim formal privacy guarantees.

Further, data heterogeneity is a well-known critical challenge in FL. In federated fine-tuning of LLMs, this challenge arises from class-label heterogeneity and linguistic variation, since clients in realistic scenarios may have text in different languages. This linguistic heterogeneity is a source of non-IID data specific to the text setting. Prior work has not examined how these modules behave under multilingual heterogeneity, which is particularly relevant in federated settings for LLMs. In this work, we study how attention modules perform when clients operate in multilingual environments, providing a broader understanding of LoRA adaptation under both class-label and linguistic heterogeneity to test whether the value-aggregation finding holds across different and realistic sources of non-IID data.

We summarize our main contributions as follows:

- We investigate the relationship between attention projections (Q and V) under federated fine-tuning of LLMs using LoRA. We analyze how the FL framework behaves when different modules are shared with the server compared to fully shared settings.

- We introduce FedLoVA, a novel federated fine-tuning framework in which only the value projection modules are aggregated by the server. We show that FedLoVA improves convergence behavior and performance compared to existing federated LLM fine-tuning baselines.
- We conduct comprehensive experiments on five text classification datasets from the GLUE benchmark, a multilingual dataset, and a generative task. Our results show that FedLoVA outperforms prior approaches under both class-label heterogeneity and multilingual heterogeneity, confirming its robustness, communication efficiency, and adaptability.

2 Related Work

2.1 PEFT in Federated Learning (FL)

FL is a distributed machine learning paradigm in which multiple clients collaboratively train a global model without sharing their local data with a central server [13, 14, 15, 18]. With the rapid growth of LLMs, FL has emerged as an appealing framework for fine-tuning these models across distributed data sources. Fine-tuning LLMs typically requires large amounts of data; however, such data are often generated across many sources, and aggregating them on a central server is challenging due to privacy concerns and communication costs. Full fine-tuning of LLMs in federated settings introduces additional challenges, including substantial computational demands on clients and communication overhead.

To address these challenges, PEFT methods have been introduced to adapt large pretrained models by updating only a small subset of parameters or by introducing lightweight trainable components, while keeping the majority of the model frozen. PEFT methods significantly reduce local computation costs and the number of parameters exchanged between clients and the server. PEFT methods can be categorized into different approaches. Adapter-based methods insert small bottleneck layers into each transformer block and train only these modules [11]. In [39], an adapter-based federated multimodal learning framework is proposed that freezes large pretrained backbones and trains lightweight task- and modality-specific adapters via split FL. Prompt-tuning approaches are another line of work that optimize a limited number of continuous prompts with a frozen backbone model [8, 6]. FedBPT [27] is an FL framework that learns shared prompts via gradient-free optimization, which significantly reduces computation costs, and shares only prompts with the server instead of the model parameters, thereby reducing communication overhead. LoRA is another category that has attracted special attention due to its simplicity and competitive performance, which is also the focus of this work. LoRA-based methods reduce the number of communicated parameters through low-rank matrix decompositions [12]. Some FL works [3, 28, 35] employ LoRA, which we discuss next.

2.2 LoRA in Federated Learning

LoRA is a PEFT technique that adapts pretrained models by injecting low-rank trainable matrices A and B into existing weight matrices, while keeping the original model parameters frozen [12]. LoRA significantly reduces the number of trainable parameters, which is beneficial in bandwidth-limited and resource-constrained federated settings. FDLoRA [24] is an approach that uses dual LoRA modules and a gradient-free adaptive fusion strategy to obtain both global and personalized knowledge. Although this method is beneficial in heterogeneous scenarios, it requires higher computational resources, since two LoRA modules must be trained.

LoRA-FAIR [2] addresses the aggregation inconsistency inherent in standard federated LoRA, where averaging LoRA matrices diverges from the ideal global update due to the nonlinearity of their product. LoRA-FAIR introduces a server-side correction term to the averaged LoRA matrices, improving aggregation accuracy. FFA-LoRA [28] is proposed to address the instability of standard LoRA under privacy-preserving federated learning by fixing A and training only B. However, fixing A can limit the ability to reach the optimal solution. FedSA-LoRA [10] further investigates the roles of A and B and finds that A is responsible for capturing global knowledge, while B captures personalized knowledge. In FedSA-LoRA, both A and B are trained locally, and only the A matrices containing global knowledge are aggregated at the server. Unlike previous works, which investigated the roles of A and B to some extent, we adopt a different perspective and aim to study the role of query and value projections in LoRA adapters within the FL framework.

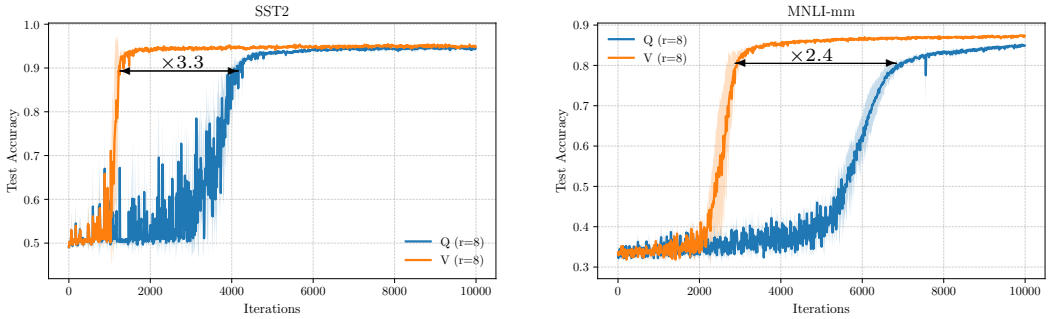


Figure 2: Convergence behavior of LoRA applied separately to Q and V matrices using RoBERTa-large on SST2 and MNLI-mm datasets. Convergence speed multipliers are computed based on reaching 95% of the final accuracy of Q, which is 89.76% for SST2 and 80.65% for MNLI-mm.

3 Problem Statement

3.1 LoRA in Transformer Attention Mechanism

LLMs require extensive computational resources for training. LoRA helps reduce the number of trainable parameters during fine-tuning. In transformer-based LLMs, LoRA is typically applied to the query (Q) and value (V) projection matrices within the attention mechanism [12]. Given an input X, the attention output is computed as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V, \tag{1}$$

where the projections are defined as:

$$Q = W_q X, \quad W_q \in \mathbb{R}^{d_q \times d_{\text{in}}},\ X \in \mathbb{R}^{d_{\text{in}}},\ Q \in \mathbb{R}^{d_q}, \tag{2}$$

$$K = W_k X, \quad W_k \in \mathbb{R}^{d_k \times d_{\text{in}}},\ K \in \mathbb{R}^{d_k}, \tag{3}$$

$$V = W_v X, \quad W_v \in \mathbb{R}^{d_v \times d_{\text{in}}},\ V \in \mathbb{R}^{d_v}. \tag{4}$$

Here, d_{in} denotes the input feature dimension, d_q, d_k, and d_v are the dimensions of the query, key, and value vectors, respectively. The dot product $QK^\top$ produces the attention scores normalized by $\sqrt{d_k}$. W_q, W_k, W_v are trainable weight matrices. In LoRA, W_q and W_v are expressed as the sum of the pretrained weights and a trainable low-rank update:

$$W_q = W_{q,0} + B_q A_q, \quad A_q \in \mathbb{R}^{r \times d_{\text{in}}},\ B_q \in \mathbb{R}^{d_q \times r}, \tag{5}$$

$$W_v = W_{v,0} + B_v A_v, \quad A_v \in \mathbb{R}^{r \times d_{\text{in}}},\ B_v \in \mathbb{R}^{d_v \times r}, \tag{6}$$

where $W_{q,0}$ and $W_{v,0}$ are the pretrained frozen weights, and $r \ll d_q, d_v$ is the LoRA rank. The matrices B_q and B_v are zero-initialized, while A_q and A_v are randomly initialized. All LoRA parameters are trained during the fine-tuning procedure to update W_q and W_v.

3.2 Empirical Motivation for Query and Value Projections

Following the original LoRA paper [12], it has been shown that training both Q and V matrices simultaneously provides the best performance under a fixed parameter budget. Hu et al. [12] also studied the cases where LoRA is applied to only Q or only V in centralized settings, and found that assigning all trainable parameters to either Q or V leads to significantly lower performance, whereas adapting both yields the best results. We also provide a comprehensive analysis of LoRA applied to different attention modules and their combinations in Appendix A.2 to justify the effectiveness of the $Q + V$ design choice. However, [12] did not investigate how these matrices affect the convergence behavior of the model. To this end, we first analyze the convergence behavior of Q and V independently. Specifically, we apply LoRA with rank $r = 8$ once on the Q and once on the V projection and evaluate performance on two datasets from the GLUE benchmark, namely SST2 and MNLI-mm, using the RoBERTa-large model with 355M parameters. As shown in Figure 2,

LoRA applied to the V projection exhibits much faster convergence than when applied to the Q projection. To quantify this, we define the target accuracy as 95% of the final accuracy achieved by the Q projection (89.76% for SST2 and 80.65% for MNLI-mm), and measure the convergence speed multiplier as the ratio of communication rounds required by Q to reach the target accuracy relative to that required by the V projection. The value projection reaches in $3.3\times$ fewer rounds on SST2 and $2.4\times$ fewer rounds on MNLI-mm to the target accuracy. We argue that this difference stems from a structural asymmetry in how Q and V interact with the optimization landscape. When optimizing V alone, the attention weights are fixed for Q and K, making the model output approximately linear in V. The loss can thus be viewed as the composition of a linear map and the downstream objective, which is expected to yield a better-conditioned optimization landscape with more stable, well-aligned gradients, and consequently faster convergence. When optimizing Q alone, the situation is different, Q influences the model output indirectly through the softmax operation, introducing strong nonlinearity. The resulting gradients may be poorly conditioned, particularly when attention logits fall into saturated regimes, leading to greater sensitivity and likely slowing convergence. To verify that this observation is not dependent on model scale or task type, we conduct the same experiments using a smaller model (BERT with 110M parameters) on the GLUE benchmark and a larger model (Llama2 with 7B parameters) on a generative task, with results reported in Appendix A.3. For generative task, we evaluate on the GSM8K dataset, a mathematical reasoning benchmark, to assess whether this behavior generalizes across task types. This observation motivates our hypothesis that, although training both Q and V can achieve the best performance, sharing only V with the global server during aggregation may yield faster convergence and better communication efficiency, since Q converges much more slowly and may contribute less effectively in federated aggregation.

3.3 Theoretical Motivation for Sharing the Value Projection

The convergence gap observed in Section 3.2 is consistent with recent theoretical results on the curvature of self-attention. Ormaniec et al. [25] derive the exact loss Hessian of a self-attention layer in closed form and decompose it, via the Gauss–Newton split, into an outer-product term $\mathbf{H}_o$ and a functional term $\mathbf{H}_f$, where the latter can be indefinite. Building on their analysis, we identify two asymmetries between the query and value projections. Note that we consider the curvature with respect to each projection matrix separately, i.e., while all other parameters are held fixed, and with respect to the full matrix rather than its low-rank LoRA factors.

Curvature asymmetry. Because the output of self-attention is linear in W_v, the functional term vanishes on the value block, $\mathbf{H}_f(W_v, W_v) = \mathbf{0}$ [25, Theorems 3.1–3.2], and hence $\mathbf{H}(W_v, W_v) = \mathbf{H}_o(W_v, W_v)$. Since $\mathbf{H}_o$ is a Gram matrix of the network Jacobian, the value block is positive semi-definite for a convex loss (e.g., cross-entropy). In contrast, both $\mathbf{H}_o(W_q, W_q)$ and $\mathbf{H}_f(W_q, W_q)$ are generically nonzero and depend on the first- and second-order derivatives of the softmax, so the query block is generically indefinite. Ormaniec et al. [25] further show that removing the softmax nonlinearity collapses $\mathbf{H}_f$ to a block-hollow (all-zero-diagonal) structure and the Hessian blocks become positive semi-definite, whereas the softmax increases its indefiniteness. This asymmetry offers a mechanistic explanation for the slower convergence of Q, as gradient-based methods are known to converge substantially more slowly near saddle points and plateaus associated with indefinite curvature [7]. Such curvature-induced difficulties cannot arise within a positive semi-definite Hessian block such as $\mathbf{H}(W_v, W_v)$.

Data-dependence asymmetry. The same analysis shows that the query Jacobian scales cubically with the input norm, $\|\partial F/\partial W_q\|_F \in \mathcal{O}(\|X\|_F^3)$, whereas the value Jacobian scales only linearly, $\|\partial F/\partial W_v\|_F \in \mathcal{O}(\|X\|_F)$ [25]. Consequently, the outer-product Hessian blocks, which are quadratic in these Jacobians, satisfy $\mathbf{H}_o(W_q, W_q) \in \mathcal{O}(\|X\|_F^6)$ and $\mathbf{H}_o(W_v, W_v) \in \mathcal{O}(\|X\|_F^2)$. In FL , where local data distributions differ across clients, a parameter for which the local curvature depends on X only quadratically should exhibit less cross-client variation in its local optimization landscape than one with sixth-order dependence. This gives a direct motivation for aggregating V rather than Q in federated settings.

Implications for federated aggregation. Taken together, these results indicate that V is both better conditioned and less sensitive to client-specific data than Q. For each client k, holding its locally trained Q_k fixed, the curvature of the local objective along V is positive semi-definite, and its dependence on the client's data is mild. This makes V a natural candidate for aggregation, while Q, characterized by a more indefinite and strongly data-dependent optimization landscape, is better

Algorithm 1 FedLoVA: Federated LoRA with Value Aggregation

Require: Pretrained LLM with frozen backbone, number of clients K, communication rounds T, local epochs E, learning rate η, client datasets $\{\mathcal{D}_k\}_{k=1}^{K}$

1: Initialize global LoRA parameters for value projections $(A_v^{(0)}, B_v^{(0)})$ and local LoRA parameters for query projections $(A_{q,k}^{(0)}, B_{q,k}^{(0)})$ for each client k
2: **for** $t = 1$ to T **do**
3: Selects a subset of clients $\mathcal{S}_t \subseteq \{1, \ldots, K\}$
4: **for all** clients $k \in \mathcal{S}_t$ **in parallel do**
5: $(A_{v,k}^{(t)}, B_{v,k}^{(t)}) \leftarrow (A_v^{(t-1)}, B_v^{(t-1)})$
6: **for** $e = 1$ to E **do**
7: Update $(A_{q,k}^{(t)}, B_{q,k}^{(t)}, A_{v,k}^{(t)}, B_{v,k}^{(t)})$ using SGD on local data $\mathcal{D}_k$ with learning rate η
8: **end for**
9: Send $(A_{v,k}^{(t),E}, B_{v,k}^{(t),E})$ to server
10: **end for**
11: $(A_v^{(t)}, B_v^{(t)}) \leftarrow \sum_{k \in \mathcal{S}_t} \frac{|\mathcal{D}_k|}{\sum_{j \in \mathcal{S}_t} |\mathcal{D}_j|} (A_{v,k}^{(t),E}, B_{v,k}^{(t),E})$
12: **end for**
13: **Output:** Global value LoRA parameters $(A_v^{(T)}, B_v^{(T)})$ and personalized query LoRA parameters $\{(A_{q,k}^{(T)}, B_{q,k}^{(T)})\}_{k=1}^{K}$

suited to remaining local. We view this analysis as structural support for our design rather than a formal convergence guarantee, which we leave to future work.

3.4 Problem Formulation

To investigate the hypothesis in Section 3.2, we design several selective-sharing schemes for LoRA-based federated fine-tuning of LLMs. Our proposed framework, FedLoVA (detailed in Section 4.1) trains both Q and V but shares only V (A_v and B_v) with the server for aggregation. As a counterpart baseline, FedLoQA trains both Q and V locally but shares only Q (A_q and B_q). For reference, FedLoRA denotes the standard approach where both projections are shared. Using the latter schemes as baselines, we evaluate FedLoVA and analyze the impact of Q and V on the performance and convergence behavior of federated fine-tuning of LLMs. Figure 1 illustrates the comparison between FedLoVA and related approaches, including FedLoRA, FFA-LoRA [28], and FedSA-LoRA [10].

4 Methodology

4.1 Algorithm

Based on our observations across different model architectures, shown in Figures 2-5, and discussed in Section 3, we propose FedLoVA, a federated fine-tuning framework that applies LoRA to both the query and value projections to achieve strong performance, while only the LoRA parameters associated with the value projection are transmitted to the server for aggregation and the query projection parameters remain local. This design is motivated by our empirical finding that the query projection converges significantly more slowly than the value projection. The overall training and aggregation procedure of FedLoVA is summarized in Algorithm 1. At initialization, the server maintains global LoRA parameters for the value projections, $(A_v^{(0)}, B_v^{(0)})$, while each client maintains its own local LoRA parameters for the query projections, $(A_{q,k}^{(0)}, B_{q,k}^{(0)})$ (line 1). At the beginning of each communication round, the server broadcasts the current global value projection parameters to the selected clients (line 5). Each client then performs local fine-tuning by jointly updating its query and value projection parameters using local data $\mathcal{D}_k$ (lines 6–8). After completing local training, only the updated value projection parameters $(A_{v,k}^{(t),E}, B_{v,k}^{(t),E})$ are transmitted back to the server (line 9), while the query projection parameters remain local. The server aggregates the received value parameters to form the updated global value LoRA parameters (line 11). Compared to FedLoRA, which shares all LoRA parameters with the server, FedLoVA offers three key advantages. First, FedLoVA reduces communication overhead by half, since the Q modules remain local and are not transmitted to

the server. Second, FedLoVA improves convergence speed by excluding the slowly converging Q modules from the global aggregation process. Finally, as a potential side benefit, FedLoVA may enhance system privacy by sharing only a subset of the trainable parameters, potentially reducing the exposure of client-specific information compared to FedLoRA, which communicates the full set of trainable parameters. A detailed privacy analysis of FedLoVA is beyond the scope of this paper. In comparison to FFA-LoRA, FedLoVA provides greater flexibility for local optimization, as FFA-LoRA keeps the A matrices fixed across communication rounds, which may constrain the ability to reach optimal solutions. Furthermore, since these fixed components (A matrices) are distributed to all clients at the beginning of training and remain unchanged across rounds, the server can access the whole model parameters. In contrast, FedLoVA shares only the value projection parameters, while other components remain local, which may help reduce the risk of exposing client-specific information. Compared to FedSA-LoRA, FedLoVA has the same number of exchanged parameters and similar local training procedure, while achieving improved convergence behavior and performance, as demonstrated in Section 5.

4.2 Convergence Analysis

In this section, the convergence analysis of FedLoVA is theoretically investigated under smooth and non-convex assumptions, which are standard in similar works [10, 16].

Assumption 4.1. $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_K$ are L-smooth with respect to all trainable parameters $\mathbf{W}_i \triangleq (W_{q,i}, W_{v,i})$ on client i. That is, for any $\mathbf{V}_i$ and $\mathbf{W}_i$,

$$\mathcal{L}_i(\mathbf{V}_i) \leq \mathcal{L}_i(\mathbf{W}_i) + \langle \mathbf{V}_i - \mathbf{W}_i, \nabla \mathcal{L}_i(\mathbf{W}_i) \rangle_F + \frac{L}{2} \|\mathbf{V}_i - \mathbf{W}_i\|_F^2.$$

where K is the number of clients, and $\nabla \mathcal{L}_i(\mathbf{W}_i)$ is the gradient of $\mathcal{L}_i$ with respect to $\mathbf{W}_i$.

Assumption 4.2. For all clients $i \in [K]$, the stochastic gradient is unbiased and has bounded second moment:

$$\mathbb{E}_{\xi_i \sim \mathcal{D}_i}\big[\nabla \mathcal{L}_i(\mathbf{W}_i, \xi_i)\big] = \nabla \mathcal{L}_i(\mathbf{W}_i), \qquad \mathbb{E}_{\xi_i \sim \mathcal{D}_i}\big[\|\nabla \mathcal{L}_i(\mathbf{W}_i, \xi_i)\|_F^2\big] \leq G^2.$$

Assumption 4.3. There exist constants $K_{r,A} > 0$, $K_{r,B} > 0$, $k_{r,A} > 0$, and $k_{r,B} > 0$ for $r \in \{q, v\}$ such that, for all clients $i \in [K]$,

$$\|A_{r,i}\|_F \leq K_{r,A}, \qquad \|B_{r,i}\|_F \leq K_{r,B},$$

and for the corresponding gradient blocks of $\nabla \mathcal{L}_i(\mathbf{W}_i)$,

$$\langle A_{r,i}^\top A_{r,i}, G_{r,i}^\top G_{r,i} \rangle_F \geq k_{r,A} \|G_{r,i}\|_F^2, \qquad \langle B_{r,i} B_{r,i}^\top, G_{r,i}^\top G_{r,i} \rangle_F \geq k_{r,B} \|G_{r,i}\|_F^2.$$

These conditions hold when $A_{r,i}$ and $B_{r,i}$ are full-rank, where $G_{r,i} = \nabla_{W_r} \mathcal{L}_i(\mathbf{W}_i)$, and $\lambda_{\min}(\cdot)$ denotes the smallest eigenvalue of a matrix. In addition, we assume $\lambda_{\min}(A_{r,i}^\top A_{r,i}) \geq k_{r,A}$, $\lambda_{\min}(B_{r,i} B_{r,i}^\top) \geq k_{r,B}$.

Assumptions 4.1 and 4.2 are widely used in previous FL works [21, 16], and Assumption 4.3 is used when LoRA is applied to FL [10].

Theorem 4.4. *Let Assumptions 4.1–4.3 hold and T denote the number of communication rounds. Then, for a constant learning rate η, E local epochs, and $t = 1, \ldots, T$, we have*

$$\begin{aligned}
\frac{1}{T} \sum_{t=1}^{T} \Big\| \nabla \mathcal{L}_i(\mathbf{W}_i^{(t)}) \Big\|_F^2 \leq{}& \frac{\mathcal{L}_i(\mathbf{W}_i^{(1)}) - \mathcal{L}_i(\mathbf{W}_i^{(T)})}{\eta T k_{\min}} + \frac{1}{k_{\min}} \left(\frac{2E}{K} \big(K_{v,A}^4 + K_{v,B}^4\big) + \frac{1}{2} \right) G^2 \\
&+ \frac{\eta}{k_{\min}} \Bigg(K_{q,A} K_{q,B} G + \frac{3L}{2} (K_{q,A}^2 + K_{q,B}^2) + \frac{2EL}{K} (K_{v,A}^4 + K_{v,B}^4) \\
&+ K_{v,A} K_{v,B} G + \frac{3L}{2} (K_{v,A}^2 + K_{v,B}^2) \Bigg) G^2 \\
&+ \frac{3L}{2k_{\min}} \eta^3 \big(K_{q,A}^2 K_{q,B}^2 + K_{v,A}^2 K_{v,B}^2\big) G^4, \qquad (7)
\end{aligned}$$

where $k_{min} = \min\{k_{q,A} + k_{q,B}, k_{v,A} + k_{v,B}\}$.

Table 1: Performance of different methods on the GLUE benchmark. Accuracy is evaluated across 5 seeds with mean and standard deviation.

	MNLI-m	MNLI-mm	SST2	QQP	RTE	Avg.
FedLoRA	87.48 ± 1.16	87.63 ± 0.44	94.15 ± 0.74	85.90 ± 0.57	83.29 ± 0.27	87.69
FFA-LoRA [28]	87.74 ± 0.53	87.58 ± 0.19	93.87 ± 0.78	85.38 ± 1.03	80.99 ± 0.87	87.11
FedSA-LoRA [10]	87.09 ± 1.57	86.95 ± 0.32	94.18 ± 0.71	82.82 ± 2.77	83.07 ± 1.14	86.82
LoRA-FAIR [2]	87.54 ± 0.36	87.39 ± 0.19	93.87 ± 0.39	85.97 ± 0.69	83.71 ± 2.33	87.70
FedLoQA	87.76 ± 1.45	86.39 ± 0.90	93.81 ± 1.57	83.05 ± 1.65	83.43 ± 1.36	86.89
FedLoVA	$\mathbf{88.04} \pm 1.46$	$\mathbf{87.81} \pm 0.41$	$\mathbf{94.45} \pm 0.47$	$\mathbf{86.23} \pm 0.65$	$\mathbf{84.13} \pm 1.31$	**88.13**

As the learning rate diminishes, all η-dependent terms in (7) vanish, leaving only the η-independent constant term. Therefore, as in [16], FedLoVA exhibits practical convergence and converges to a neighborhood of a stationary point.

Unlike standard FL approaches, FedLoVA does not maintain a single global model for all parameters. In particular, the query LoRA parameters are kept local and are never communicated to the server. Consequently, the convergence analysis focuses on the local client objectives rather than a global objective. The detailed proof of Theorem 4.4 is provided in Appendix A.4.

5 Experiments

In this section, we evaluate the performance of FedLoVA against baseline approaches from five key aspects: 1) comparing performance in natural language understanding under class-label heterogeneity; 2) evaluating performance under linguistic heterogeneity; 3) investigating the performance for natural language generation; 4) assessing the efficiency and convergence behavior in different scenarios; and 5) analyzing sensitivity to heterogeneity level, number of clients, and LoRA rank. All experiments are implemented using the FederatedScope-LLM framework [17] on an NVIDIA GeForce RTX 3090 GPU, PyTorch 2.1.0, and CUDA 12.5. The main results in Table 1 are averaged over five independent seeds, while additional analyses are based on three independent runs for each configuration.

5.1 Performance Comparison for Natural Language Understanding

For natural language understanding (NLU) tasks, we use the RoBERTa-large model (355M parameters) [22], evaluated on four datasets from the GLUE benchmark [32], including MNLI, SST2, QQP, and RTE. We use the pretrained weights from Hugging Face for model initialization. All experiments are performed in an FL setup with three clients, similar to other federated fine-tuning of LLMs works [28, 10]. The datasets are partitioned among clients using a non-IID Dirichlet distribution with $\beta = 0.5$. This setting ensures strong heterogeneity across local clients, consistent with prior FL literature [10]. We set the batch size to $B = 128$, the total number of communication rounds to 1000, local steps to 10, and the optimizer to SGD for training. The sequence length is fixed to $l_{\text{seq}} = 128$ across all methods for consistency with previous studies [12, 28, 10]. Following [12], we integrate LoRA adapters only into the query (Q) and value (V) projection matrices of the attention layers. The LoRA rank is set to $r = 8$, and the scaling factor to $\alpha = 16$. We report the best performance among experiments conducted with learning rates $\eta \in \{0.005, 0.01, 0.02, 0.05, 0.1\}$. The selected learning rate for each method is detailed in Appendix A.5.

The results are summarized in Table 1. As shown, FedLoVA achieves comparable or better performance compared to other methods across all datasets, demonstrating its effectiveness in federated fine-tuning of LLMs. LoRA-FAIR and FedLoRA achieve the second-best performance but require approximately twice the communication overhead compared to FedLoVA. Since Table 1 compares all methods at the same number of rounds, it does not account for the difference in communication cost. We therefore provide a communication-matched comparison in Appendix A.6, where FedLoRA and LoRA-FAIR are evaluated at the same cumulative communication budget as FedLoVA. Under this setting, FedLoVA improves the average accuracy by 1.4% and 2.0% over FedLoRA and LoRA-FAIR, respectively, and a paired t-test across tasks confirms that FedLoVA improvement is statistically significant. Therefore, FedLoVA provides two key advantages over FedLoRA and LoRA-FAIR: 1)

Table 2: Performance of different methods on the MLS dataset.

FedLoRA	FFA-LoRA	FedSA-LoRA	LoRA-FAIR	FedLoQA	FedLoVA
68.90 ± 0.24	69.63 ± 0.18	68.61 ± 0.12	69.05 ± 0.36	67.79 ± 0.19	$\mathbf{70.18} \pm 0.28$

Table 3: Performance of different methods on the GSM8K dataset

FedLoRA	FFA-LoRA	FedSA-LoRA	FedLoQA	FedLoVA
43.59 ± 0.02	44.95 ± 1.08	44.04 ± 2.61	38.74 ± 2.83	$\mathbf{45.79} \pm 1.46$

achieves comparable accuracy in the same number of rounds and, under an equal communication budget, achieves higher accuracy with a larger margin, 2) reduces communication cost.

FedLoQA performs worse than both FedLoVA and FedLoRA on average. This suggests that sharing the query projection alone is less effective for aggregation and may slow down convergence due to its slower adaptation behavior, as observed in Section 3.2. In addition, we observe that FedLoQA is significantly more sensitive to the choice of learning rate, whereas FedLoVA remains stable across a wider range. As shown in Appendix A.5, FedLoQA consistently achieves its best performance at a low learning rate (0.005) across all datasets, while FedLoVA attains strong performance even at higher learning rates. Detailed results for SST2 and MNLI-mm are provided in Appendix A.7, where we show that the performance of FedLoQA degrades as the learning rate increases, whereas FedLoVA remains comparatively stable. This sensitivity to the learning rate could be related to the structure of the attention mechanism. In Eq. 1, the value projection contributes linearly to the output, making its aggregation across clients more stable and less sensitive to heterogeneity. In particular, due to linearity, averaging the value projections across clients yields outputs that are consistent with the average of the client outputs. In contrast, the query projection affects the attention distribution through the softmax nonlinearity, which is highly sensitive to its scale and direction. As a result, aggregating Q across heterogeneous clients may lead to inconsistent attention patterns, increasing sensitivity to the learning rate and negatively impacting convergence. Our observations are also consistent with prior work [26], which reports that query projections exhibit larger deviations from ideal updates compared to value projections in federated LoRA settings.

5.2 Performance Comparison for Multilingual Sentiment Analysis

In federated settings, clients may use different languages. To evaluate the robustness of FedLoVA under linguistic heterogeneity, we perform multilingual sentiment analysis using the Amazon reviews dataset from the Hugging Face repository.[1] The Multilingual Sentiment (MLS) dataset contains data from 27 domains, including Amazon reviews, book reviews, hotel reviews, and others, across 17 languages. Among these, the Amazon reviews constitute the largest share of the dataset (approximately 28% of the total). Therefore, we select this domain for our experiments to ensure sufficient data per client. As we have three clients, we use Amazon review data from English, German, and Chinese languages, assigning one language to each client. The experimental settings for MLS are the same as those described in Section 5.1. The results are presented in Table 2. FedLoVA outperforms other methods, achieving an accuracy of 70.18% under multilingual conditions. This demonstrates its effectiveness and robustness in linguistically heterogeneous environments.

5.3 Performance Comparison for Natural Language Generation

For natural language generation (NLG) tasks, we employ the Llama3-8B model [30] and evaluate its performance on the GSM8K dataset [5]. GSM8K is a dataset of grade-school mathematical problems that require multi-step numerical reasoning, which is a challenging dataset for evaluating reasoning capabilities in language models. The number of communication rounds is set to 400, with three participating clients. For data distribution and optimization parameters, we follow the experimental settings in [17]. The results are presented in Table 3. FedLoVA shows improved performance compared to other methods in this setting, indicating its potential effectiveness in complex generation tasks. To further analyze the reasoning behavior of different methods, we provide several qualitative examples with varying question lengths and complexity in Appendix A.8. FedLoVA exhibits stronger

[1] `https://huggingface.co/datasets/clapAI/MultiLingualSentiment`

Table 4: Comparison of total communication cost, computed as transmitted parameters per round multiplied by communication rounds to reach the target accuracy.

	MNLI-m	MNLI-mm	SST2	QQP	RTE	MLS
Target Accuracy	87%	86%	93%	82%	80%	67%
FedLoRA	215.3 M (3.2×)	132.6 M (3.1×)	156.0 M (2.2×)	237.1 M (3.3×)	230.1 M (2.0×)	287.0 M (3.1×)
FFA-LoRA	184.9 M (2.7×)	161.5 M (3.8×)	352.2 M (5.0×)	121.3 M (1.7×)	186.0 M (1.6×)	111.5 M (1.2×)
FedSA-LoRA	129.9 M (1.9×)	128.7 M (3.0×)	102.6 M (1.5×)	111.9 M (1.6×)	128.3 M (1.1×)	140.4 M (1.5×)
LoRA-FAIR	411.8 M (6.0×)	297.2 M (6.9×)	237.9 M (3.4×)	266.8 M (3.8×)	280.0 M (2.5×)	279.2 M (3.0×)
FedLoQA	185.3 M (2.7×)	333.1 M (7.8×)	113.1 M (1.6×)	101.4 M (1.4×)	115.1 M (1.0×)	151.7 M (1.6×)
FedLoVA	**68.3 M** (1×)	**42.9 M** (1×)	**69.8 M** (1×)	**71.0 M** (1×)	**113.9 M** (1×)	**93.2 M** (1×)

and more consistent reasoning ability across both short and long questions, while other methods often show confusion even on relatively simple mathematical problems.

5.4 Efficiency and Convergence Analysis

In this section, we compare the efficiency of FedLoVA against the baselines in terms of communication cost and convergence behavior. To provide a unified measure of communication efficiency, Table 4 reports the total communication cost, computed as the product of transmitted parameters per round and the number of communication rounds required to reach the target accuracy. For each dataset, the target accuracy is set to the floor of the lowest accuracy achieved among all methods. As shown in Table 4, FedLoVA has the lowest communication cost among all datasets and demonstrates a significant reduction in communication cost, while other methods incur up to $7.8\times$ higher cost. Additional metrics such as the number of communication rounds required to achieve a target accuracy, local trainable parameters, and transmitted parameters per round are reported in Appendix A.9.

5.5 In-Depth Analysis

We conduct ablation studies to investigate the impact of data heterogeneity level, number of clients, and LoRA rank on the performance of FedLoVA, where the detailed results are reported in Appendix A.10. To assess scalability, we increase the number of clients to 10 and 50. FedLoVA achieves the highest accuracy in all four settings. On MNLI-m, it reaches 89.18% with 50 clients, compared to 88.60% for the strongest baseline (FedSA-LoRA), and 86.90% with 10 clients, compared to 86.59% (FFA-LoRA). On SST-2 with 10 clients, it improves over the strongest baseline (FedLoRA) from 96.19% to 96.53%. In addition, we evaluate an IID setting and a mild non-IID setting ($\beta = 1.0$). FedLoVA achieves the highest accuracy in all cases, with the largest gain on MNLI-m under mild non-IID data, where it improves over the strongest baseline from 87.63% to 88.51%. Varying the LoRA rank from the default $r = 8$ to $r = 4$ and $r = 16$, FedLoVA matches or outperforms all baselines, with the largest gain on MNLI-m at $r = 4$ (88.32% versus 87.80% for FedLoRA).

In Appendix A.10, we also evaluate the robustness of all methods when differential privacy (DP) noise is added to the transmitted local updates. FedLoVA achieves the highest accuracy on SST-2 (92.83% versus 92.21% for FedSA-LoRA) and ties for the highest on MNLI-m (85.73%). This indicates that FedLoVA retains high utility under a standard DP mechanism.

6 Conclusion

In this paper, we investigate the role of attention modules in LoRA-based federated fine-tuning of LLMs. Our analysis shows that value projections converge significantly faster than query projections. Motivated by this, we introduce FedLoVA, which applies LoRA to both projections but aggregates only the value projection at the server. Experiments on natural language understanding, multilingual sentiment analysis, and generation tasks demonstrate that FedLoVA achieves faster convergence and comparable or superior performance while significantly reducing communication costs.

Limitations and Future Work. Our method assumes a fixed aggregation strategy (value-only aggregation). As a future work, we plan to explore layer-wise selective aggregation, where value projections are aggregated selectively across layers.

Broader Impact. A brief discussion of broader impact is provided in the Appendix A.1.

Acknowledgments and Disclosure of Funding

We thank the area chairs and reviewers for their valuable comments and suggestions, which substantially improved this paper. Ensieh Khazaei is supported by a doctoral funding package from the University of Toronto.

References

[1] Sara Babakniya, Ahmed Elkordy, Yahya Ezzeldin, Qingfeng Liu, Kee-Bong Song, Mostafa El-Khamy, and Salman Avestimehr. SLoRA: Federated parameter efficient fine-tuning of language models. In *International Workshop on Federated Learning in the Age of Foundation Models in Conjunction with NeurIPS 2023*, 2023.

[2] Jieming Bian, Lei Wang, Letian Zhang, and Jie Xu. LoRA-FAIR: Federated lora fine-tuning with aggregation and initialization refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3737–3746, 2025.

[3] Yuji Byun and Jaeho Lee. Towards federated low-rank adaptation of language models with rank heterogeneity. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 356–362, 2025.

[4] Yae Jee Cho, Luyang Liu, Zheng Xu, Aldi Fahrezi, and Gauri Joshi. Heterogeneous lora for federated fine-tuning of on-device foundation models. *arXiv preprint arXiv:2401.06432*, 2024.

[5] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[6] Tianyu Cui, Hongxia Li, Jingya Wang, and Ye Shi. Harmonizing generalization and personalization in federated prompt learning. *arXiv preprint arXiv:2405.09771*, 2024.

[7] Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. *Advances in neural information processing systems*, 27, 2014.

[8] Wenlong Deng, Christos Thrampoulidis, and Xiaoxiao Li. Unlocking the potential of prompt-tuning in bridging generalized and personalized federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6087–6097, 2024.

[9] Sagar Goyal, Eti Rastogi, Sree Prasanna Rajagopal, Dong Yuan, Fen Zhao, Jai Chintagunta, Gautam Naik, and Jeff Ward. Healai: A healthcare llm for effective medical documentation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 1167–1168, 2024.

[10] Pengxin Guo, Shuang Zeng, Yanran Wang, Huijie Fan, Feifei Wang, and Liangqiong Qu. Selective aggregation for low-rank adaptation in federated learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

[11] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR, 2019.

[12] 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. In *International Conference on Learning Representations*, 2022.

[13] Ensieh Khazaei, Alireza Esmaeilzehi, Bilal Taha, and Dimitrios Hatzinakos. CDFL: efficient federated human activity recognition using contrastive learning and deep clustering. *IEEE Sensors Journal*, 24(22):38196–38208, 2024.

[14] Ensieh Khazaei and Dimitrios Hatzinakos. Forget less, learn more: Contrastive-based federated class incremental learning with a low-dimensional projection layer. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1782–1791. IEEE, 2025.

[15] Ensieh Khazaei, Bilal Taha, Alireza Esmaeilzehi, and Dimitrios Hatzinakos. OSR: Toward developing efficient federated learning-based human activity recognition using optimal server representations. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[16] Jisoo Kim, Sungmin Kang, and Sunwoo Lee. Layer-wise update aggregation with recycling for communication-efficient federated learning. *arXiv preprint arXiv:2503.11146*, 2025.

[17] 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.

[18] Seungjun Lee, Ensieh Khazaei, Dimitrios Hatzinakos, Baturalp Buyukates, and Sunwoo Lee. Latent information sharing for accelerating federated learning. *arXiv preprint arXiv:2610.01126*, 2026.

[19] Huaicheng Li, Junhui Zhao, Haoyu Quan, and Xiaoming Wang. Fed-DLoRA: Efficient wireless federated learning with dynamic low-rank adaptation. *IEEE Transactions on Vehicular Technology*, 2026.

[20] Minglei Li, Peng Ye, Yongqi Huang, Lin Zhang, Tao Chen, Tong He, Jiayuan Fan, and Wanli Ouyang. Adapter-x: A novel general parameter-efficient fine-tuning framework for vision. *arXiv preprint arXiv:2406.03051*, 2024.

[21] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. *arXiv preprint arXiv:1907.02189*, 2019.

[22] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[23] Guodong Long, Tao Shen, Jing Jiang, Michael Blumenstein, et al. Dual-personalizing adapter for federated foundation models. *Advances in Neural Information Processing Systems*, 37:39409–39433, 2024.

[24] Yao Lu, 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.

[25] Weronika Ormaniec, Felix Dangel, and Sidak Pal Singh. What does it mean to be a transformer? insights from a theoretical hessian analysis. In *International Conference on Learning Representations*, volume 2025, pages 17046–17081, 2025.

[26] Raghav Singhal, Kaustubh Ponkshe, and Praneeth Vepakomma. FedEx-LoRA: Exact aggregation for federated and efficient fine-tuning of large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1316–1336, 2025.

[27] Jingwei Sun, Ziyue Xu, Hongxu Yin, Dong Yang, Daguang Xu, Yiran Chen, and Holger R Roth. Fedbpt: Efficient federated black-box prompt tuning for large language models. *arXiv preprint arXiv:2310.01467*, 2023.

[28] Youbang Sun, Zitao Li, Yaliang Li, and Bolin Ding. Improving lora in privacy-preserving federated learning. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun, editors, *International Conference on Representation Learning*, volume 2024, pages 17978–17994, 2024.

[29] Chunlin Tian, Zhan Shi, Zhijiang Guo, Li Li, and Cheng-Zhong Xu. Hydralora: An asymmetric lora architecture for efficient fine-tuning. *Advances in Neural Information Processing Systems*, 37:9565–9584, 2024.

[30] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[31] Linh Tran, Ana Milanova, and Stacy Patterson. Improving parameter-efficient federated learning with differentially private refactorization. *arXiv preprint arXiv:2605.08443*, 2026.

[32] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

[33] Haonan Wang, Brian K Chen, Li Siquan, Liang Xinhe, Tianyang Hu, Hwee Kuan Lee, and Kenji Kawaguchi. Prefix-Tuning+: Modernizing prefix-tuning by decoupling the prefix from attention. In *Second Workshop on Test-Time Adaptation: Putting Updates to the Test! at ICML 2025*, 2025.

[34] Runqian Wang, Soumya Ghosh, David Cox, Diego Antognini, Aude Oliva, Rogerio Feris, and Leonid Karlinsky. Trans-LoRA: towards data-free transferable parameter efficient finetuning. *Advances in Neural Information Processing Systems*, 37:61217–61237, 2024.

[35] 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. *Advances in Neural Information Processing Systems*, 37:22513–22533, 2024.

[36] Panlong Wu, Kangshuo Li, Ting Wang, Yanjie Dong, Victor CM Leung, and Fangxin Wang. FedFMSL: Federated learning of foundation models with sparsely activated lora. *IEEE Transactions on Mobile Computing*, 23(12):15167–15181, 2024.

[37] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, et al. Autogen: Enabling next-gen llm applications via multi-agent conversations. In *First Conference on Language Modeling*, 2024.

[38] Yangyang Yu, Zhiyuan Yao, Haohang Li, Zhiyang Deng, Yuechen Jiang, Yupeng Cao, Zhi Chen, Jordan Suchow, Zhenyu Cui, Rong Liu, et al. Fincon: A synthesized llm multi-agent system with conceptual verbal reinforcement for enhanced financial decision making. *Advances in Neural Information Processing Systems*, 37:137010–137045, 2024.

[39] Zixin Zhang, Fan Qi, and Changsheng Xu. Enhancing storage and computational efficiency in federated multimodal learning for large-scale models. In *Forty-first International Conference on Machine Learning*, 2024.

A Appendix

A.1 Broader Impact

This work advances communication-efficient federated fine-tuning of LLMs, with direct implications for deploying AI in privacy-sensitive and resource-constrained environments such as healthcare, finance, and multilingual applications. By identifying the value projection as the most effective module for server aggregation, FedLoVA reduces communication overhead by up to 7.8× compared to existing baselines while maintaining competitive performance, lowering the barrier for organizations with limited bandwidth to participate in collaborative model training without sharing raw data. Beyond the proposed method, our systematic analysis of attention submodules under both class-label and linguistic heterogeneity provides an answer to the question that which components of transformer-based models are most suitable for federated aggregation.

A.2 Analysis of LoRA Placement in Attention Modules

In this section, we investigate the effect of applying LoRA to different attention modules and their combinations. While LoRA could be applied to K, Q, and V attention modules, most prior works [10, 28, 23, 4] follow the original LoRA paper [12] and apply LoRA only to the query and value projections. The original LoRA study shows that adapting both Q and V captures the downstream task while maintaining parameter efficiency. To further study this design choice, we conduct additional experiments by applying LoRA to different attention modules and their combinations on SST2 and MNLI-mm using RoBERTa-large. All experiments are averaged over three random seeds. We evaluate both the convergence behavior and the final performance of each configuration. The results are summarized in Table 5 and Figure 3.

Based on the results in Figure 3, we observe that among individual modules, the V projection converges the fastest, while the K projection shows the slowest convergence. Applying LoRA only on MLP attains slightly better final performance at a much higher parameter cost: with r=8, $Q/K/V$-only LoRA uses 0.39M trainable parameters vs. 2.36M for MLP (6× larger). Given slower convergence and higher overhead of MLP, MLP adaptation is less suitable for federated settings with limited client resources and it is not included in module combinations. In addition, combining modules generally improves convergence compared to the best individual component within each pair. For example, the $Q+K$ combination converges faster than Q alone (which itself converges faster than K). Since V shows the fastest convergence, combinations involving V, including $Q+V$ and $K+V$ exhibit better convergence behavior than $Q+K$ combination that exclude it. Moreover, under joint training, we observe that combinations involving V, $Q+V$ and $K+V$ exhibit very similar convergence behavior, even though Q and K alone have noticeably different convergence patterns. This suggests that V plays a dominant role in determining the overall convergence behavior when trained jointly with other components. In addition, when we train all three modules $Q+K+V$ together, we observe a similar convergence behavior to joint training of $Q+V$ and $K+V$, indicating that the contribution of V remains dominant even in the presence of additional components. Therefore, the joint training of all three modules does not improve in convergence compared to pairwise combination of $Q+V$, while it has higher computational cost and more trainable parameters.

In terms of accuracy, a similar pattern is observed. Pairwise combinations outperform the best individual module within that pair. Among these, the $Q+V$ combination achieves the strongest performance. While combining all three modules of $Q+K+V$ can provide marginal improvements in MNLI-mm, it comes at the cost of increased computational overhead. As FL is constrained by limited local compute and communication bandwidth, efficiency is critical. Since the only advantage of $Q+K+V$ is a small accuracy gain only on MNLI-mm with no convergence benefit, its $1.5\times$ overhead compared to $Q+V$ is not justified in our setting. These results suggest that the combination of Q and V provides a favorable trade-off between accuracy, convergence behavior, and computational cost. This observation is also consistent with the findings of the original LoRA work [12]. Therefore, we adopt this setting in our main experiments.

A.3 Convergence Analysis of Query and Value Projections in LoRA

In this section, we present a convergence analysis of the query and value projections when LoRA is applied exclusively to either query (Q) or value (V) projections. We consider a small model like

Table 5: Performance comparison of applying LoRA to different attention modules and their combinations on SST2 and MNLI-mm (averaged over three random seeds).

	K	Q	V	MLP	$K+Q$	$K+V$	$Q+V$	$Q+K+V$
SST2	91.13 ± 1.43	94.49 ± 0.32	94.84 ± 0.09	95.70 ± 0.05	94.34 ± 0.32	95.03 ± 0.10	95.49 ± 0.05	95.49 ± 0.10
MNLI-mm	41.81 ± 3.00	84.90 ± 0.27	87.31 ± 0.11	88.53 ± 0.22	85.15 ± 0.41	87.61 ± 0.09	87.80 ± 0.12	88.03 ± 0.11

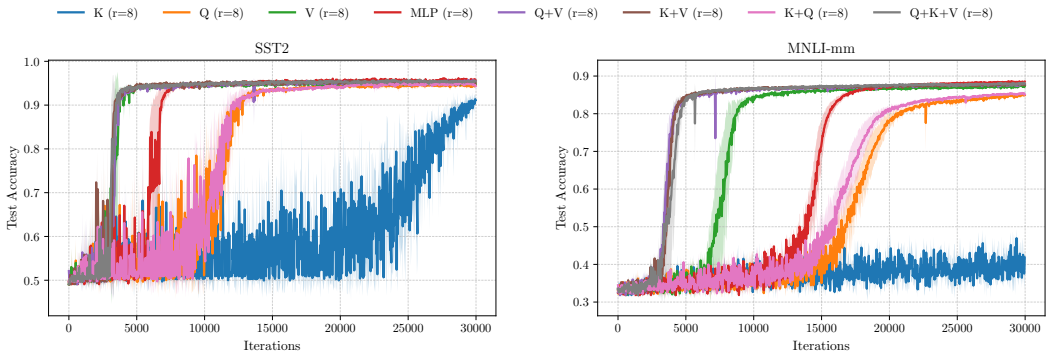


Figure 3: Convergence behavior of LoRA applied to different attention modules and their combinations on SST2 and MNLI-mm using RoBERTa-large.

BERT with 110M parameters, and evaluate it on two classification datasets, SST2 and MNLI-mm. Its results are shown in Figure 4. To assess the behavior on larger models and more complex reasoning tasks, we conduct experiments using Llama2 with 7B parameters on a generative task from the GSM8K dataset (mathematical reasoning), with results reported in Figure 5. As evaluation of Llama2 is time consuming, we do it every 300 iterations. Across both small and large models, and across different task types (classification and reasoning), we consistently observe that LoRA applied to the V projection exhibits a substantially faster convergence compared to LoRA applied to the Q projection.

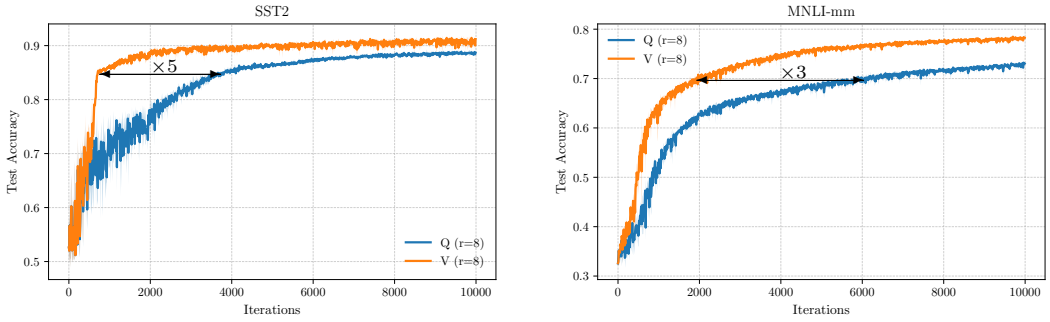


Figure 4: Convergence behavior of LoRA applied separately to Q and V matrices using BERT on SST2 and MNLI-mm datasets (averaged over three random seeds). Convergence speed multipliers are computed based on reaching 95% of the final accuracy of Q, which is 84.25% for SST2 and 69.39% for MNLI-mm.

For Figures 2 and 4, convergence is defined as reaching γ percent of the final Q projection accuracy, where $\gamma = 0.95$. To evaluate sensitivity to this choice, we vary $\gamma \in \{0.95, 0.9, 0.85, 0.8\}$. As shown in Table, the resulting convergence speed multipliers remain consistently above 1.6 across all settings, indicating that our conclusions are robust to the choice of threshold. For GSM8K, due to the limited accuracy range, we report convergence speed using a fixed target accuracy of 13%.

A.4 Proof of Theorem 4.4

Proof. In this proof, we assume full and uniform client participation, i.e., all K clients participate in every communication round with equal aggregation weights, which is adopted to simplify the

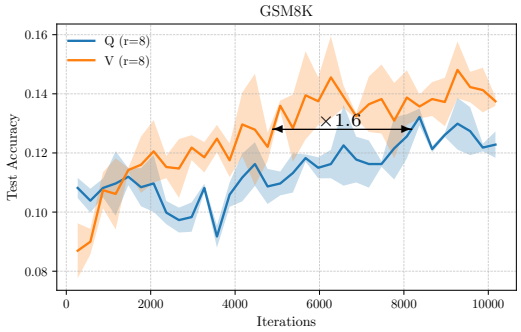


Figure 5: Convergence behavior of LoRA applied separately to Q and V matrices using Llama2 on the GSM8K dataset (averaged over three random seeds). Convergence speed multipliers are computed based on reaching the target accuracy of 13%.

Table 6: Convergence speed multipliers between V and Q projections for different target accuracy thresholds γ.

	MNLI-mm		SST2	
	BERT	RoBERTa-large	BERT	RoBERTa-large
$\gamma = 0.8$	2.1	2.3	1.6	2.8
$\gamma = 0.85$	2.3	2.3	2.0	3.2
$\gamma = 0.9$	2.6	2.3	3.4	3.2
$\gamma = 0.95$	3.0	2.4	5.0	3.3

analysis. Let us denote the full set of trainable parameters for client i at iteration t as:

$$\mathbf{W}_i^{(t)} \triangleq \left(W_{q,i}^{(t)}, W_{v,i}^{(t)}\right).$$

We parametrize the query and value projection matrices of the i-th client at iteration t as:

$$W_{r,i}^{(t)} = W_{r,0} + B_{r,i}^{(t)} A_{r,i}^{(t)}, \quad r \in \{q, v\}.$$

Clients communicate with the central server every E local iterations. We denote the set of communication rounds by $\mathcal{I}_E = \{nE \mid n = 1, 2, \ldots\}$. Accordingly, the query projection of client i at iterations t and $t+1$ is given by:

$$W_{q,i}^{(t)} = W_{q,0} + B_{q,i}^{(t)} A_{q,i}^{(t)}, \qquad W_{q,i}^{(t+1)} = W_{q,0} + B_{q,i}^{(t+1)} A_{q,i}^{(t+1)}.$$

For the value projection, we define the following intermediate matrices:

$$U_{v,i}^{(t)} = W_{v,0} + B_{v,i}^{(t+1)} A_{v,i}^{(t+1)},$$

and

$$V_{v,i}^{(t)} = W_{v,0} + \left(\frac{1}{K}\sum_{j=1}^{K} B_{v,j}^{(t+1)}\right)\left(\frac{1}{K}\sum_{j=1}^{K} A_{v,j}^{(t+1)}\right),$$

which correspond to the value projection after one local update and after global aggregation, respectively.

The update rule for the value projection matrix of client i at iteration $t+1$ is then given by:

$$W_{v,i}^{(t+1)} = \begin{cases} U_{v,i}^{(t)}, & \text{if } t+1 \notin \mathcal{I}_E, \\ V_{v,i}^{(t)}, & \text{if } t+1 \in \mathcal{I}_E. \end{cases}$$

For $r \in \{q, v\}$, gradients of the local loss with respect to the LoRA parameters B_r and A_r at iteration t are given by:

$$\nabla_{B_r} \mathcal{L}_i(\mathbf{W}_i^{(t)}, \xi_{i,t}) = \nabla_{W_r} \mathcal{L}_i(\mathbf{W}_i^{(t)}, \xi_{i,t}) A_{r,i}^{(t)\top},$$

$$\nabla_{A_r}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t}) = {B_{r,i}^{(t)}}^\top \nabla_{W_r}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t}).$$

Under Assumption 4.2, the stochastic gradients of the LoRA parameters are unbiased estimators of the full gradients, i.e.,

$$\nabla_{B_r}\mathcal{L}_i(\mathbf{W}_i^{(t)}) = \mathbb{E}_{\xi_{i,t}}\big[\nabla_{B_r}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t})\big], \qquad \nabla_{A_r}\mathcal{L}_i(\mathbf{W}_i^{(t)}) = \mathbb{E}_{\xi_{i,t}}\big[\nabla_{A_r}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t})\big],$$

where $\xi_{i,t}$ is a mini-batch sampled from client i at iteration t.

By Assumption 4.1, for any client i and iteration t, the L-smoothness of the local objective with respect to all trainable parameters implies the following:

$$\mathcal{L}_i(\mathbf{W}_i^{(t+1)}) \le \mathcal{L}_i(\mathbf{W}_i^{(t)}) + \left\langle \mathbf{W}_i^{(t+1)} - \mathbf{W}_i^{(t)}, \nabla\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F + \frac{L}{2}\left\|\mathbf{W}_i^{(t+1)} - \mathbf{W}_i^{(t)}\right\|_F^2. \quad (8)$$

Here, the full gradient with respect to all trainable parameters admits the decomposition:

$$\nabla\mathcal{L}_i(\mathbf{W}_i^{(t)}) \triangleq \big(\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}), \nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\big), \quad (9)$$

and the parameter difference satisfies $\mathbf{W}_i^{(t+1)} - \mathbf{W}_i^{(t)} = \big(W_{q,i}^{(t+1)} - W_{q,i}^{(t)}, W_{v,i}^{(t+1)} - W_{v,i}^{(t)}\big)$.

Therefore, we can split the second and third terms of (8) into query and value contributions:

$$\begin{aligned}\left\langle \mathbf{W}_i^{(t+1)} - \mathbf{W}_i^{(t)}, \nabla\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F &= \left\langle W_{q,i}^{(t+1)} - W_{q,i}^{(t)}, \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F \\ &\quad + \left\langle W_{v,i}^{(t+1)} - W_{v,i}^{(t)}, \nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F, \quad (10)\end{aligned}$$

$$\frac{L}{2}\|\mathbf{W}_i^{(t+1)} - \mathbf{W}_i^{(t)}\|_F^2 = \frac{L}{2}\|W_{q,i}^{(t+1)} - W_{q,i}^{(t)}\|_F^2 + \frac{L}{2}\|W_{v,i}^{(t+1)} - W_{v,i}^{(t)}\|_F^2. \quad (11)$$

In the ensuing analysis, we bound the query and value projection parts separately.

A.4.1 Query Projection Bound

We express the updates of the LoRA parameters $A_{q,i}$ and $B_{q,i}$ using stochastic gradient descent:

$$A_{q,i}^{(t+1)} = A_{q,i}^{(t)} - \eta\nabla_{A_q}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t}) = A_{q,i}^{(t)} - \eta {B_{q,i}^{(t)}}^\top \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t}), \quad (12)$$

$$B_{q,i}^{(t+1)} = B_{q,i}^{(t)} - \eta\nabla_{B_q}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t}) = B_{q,i}^{(t)} - \eta\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t}){A_{q,i}^{(t)}}^\top. \quad (13)$$

Then, the difference between consecutive query projection matrices can be written as:

$$\begin{aligned}W_{q,i}^{(t+1)} - W_{q,i}^{(t)} &= W_{q,0} + B_{q,i}^{(t+1)}A_{q,i}^{(t+1)} - W_{q,0} - B_{q,i}^{(t)}A_{q,i}^{(t)} \\ &= B_{q,i}^{(t+1)}A_{q,i}^{(t+1)} - B_{q,i}^{(t)}A_{q,i}^{(t)} \\ &= \Big(B_{q,i}^{(t)} - \eta\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t}){A_{q,i}^{(t)}}^\top\Big)\Big(A_{q,i}^{(t)} - \eta {B_{q,i}^{(t)}}^\top\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t})\Big) \\ &\quad - B_{q,i}^{(t)}A_{q,i}^{(t)} \\ &= \eta^2\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t}){A_{q,i}^{(t)}}^\top {B_{q,i}^{(t)}}^\top\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t}) \\ &\quad - \eta\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t}){A_{q,i}^{(t)}}^\top A_{q,i}^{(t)} - \eta B_{q,i}^{(t)}{B_{q,i}^{(t)}}^\top\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)},\xi_{i,t}). \quad (14)\end{aligned}$$

Substituting (14) into the inner product term for the query projection and taking expectations with respect to the stochastic sample $\xi_{i,t}$ yields:

$$\begin{aligned}&\left\langle W_{q,i}^{(t+1)} - W_{q,i}^{(t)}, \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F \\ &= \eta^2\left\langle \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}){A_{q,i}^{(t)}}^\top {B_{q,i}^{(t)}}^\top\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}), \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F \\ &\quad - \eta\left\langle \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}){A_{q,i}^{(t)}}^\top A_{q,i}^{(t)}, \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F \\ &\quad - \eta\left\langle B_{q,i}^{(t)}{B_{q,i}^{(t)}}^\top\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}), \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F. \quad (15)\end{aligned}$$

We bound the first term in (15):

$$\begin{aligned}
&\left\langle \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) A_{q,i}^{(t)^\top} B_{q,i}^{(t)^\top} \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}),\, \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F \\
&= \mathrm{Tr}\left[\left(\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) A_{q,i}^{(t)^\top} B_{q,i}^{(t)^\top} \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right)^\top \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right] \\
&= \mathrm{Tr}\Big[\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top B_{q,i}^{(t)} A_{q,i}^{(t)} \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \Big] \\
&= \mathrm{Tr}\Big[\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top B_{q,i}^{(t)} A_{q,i}^{(t)} \Big] \\
&= \left\langle \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}),\, B_{q,i}^{(t)} A_{q,i}^{(t)} \right\rangle_F . \qquad (16)
\end{aligned}$$

Applying the Cauchy–Schwarz inequality, we obtain:

$$\begin{aligned}
&\left\langle \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) A_{q,i}^{(t)^\top} B_{q,i}^{(t)^\top} \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}),\, \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F \leq \\
&\quad \left\| \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\|_F \left\| B_{q,i}^{(t)} A_{q,i}^{(t)} \right\|_F \leq \\
&\quad \left\| \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\|_F^3 \left\| B_{q,i}^{(t)} \right\|_F \left\| A_{q,i}^{(t)} \right\|_F \leq K_{q,A} K_{q,B} G^3 . \qquad (17)
\end{aligned}$$

We next bound the second term of (15). Using Assumption 4.3, we have:

$$\begin{aligned}
&\left\langle \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) A_{q,i}^{(t)^\top} A_{q,i}^{(t)},\, \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F \\
&= \mathrm{Tr}\Big[A_{q,i}^{(t)^\top} A_{q,i}^{(t)} \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \Big] \\
&= \left\langle A_{q,i}^{(t)^\top} A_{q,i}^{(t)},\, \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F \\
&\geq k_{q,A} \left\| \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\|_F^2 . \qquad (18)
\end{aligned}$$

The third term of (15) can be similarly bounded using Assumption 4.3:

$$\begin{aligned}
&\left\langle B_{q,i}^{(t)} B_{q,i}^{(t)^\top} \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}),\, \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F \\
&= \mathrm{Tr}\left[\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top B_{q,i}^{(t)} B_{q,i}^{(t)^\top} \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right] \\
&= \mathrm{Tr}\left[\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top B_{q,i}^{(t)} B_{q,i}^{(t)^\top} \right] \\
&= \mathrm{Tr}\left[B_{q,i}^{(t)} B_{q,i}^{(t)^\top} \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top \right] \\
&= \left\langle B_{q,i}^{(t)} B_{q,i}^{(t)^\top},\, \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})^\top \right\rangle_F \\
&\geq k_{q,B} \left\| \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\|_F^2 . \qquad (19)
\end{aligned}$$

Combining the bounds derived in (17), (18), and (19), we obtain:

$$\begin{aligned}
\left\langle W_{q,i}^{(t+1)} - W_{q,i}^{(t)}, \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\rangle_F \leq \eta^2 K_{q,A} K_{q,B} G^3 - \eta k_{q,A} \left\| \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\|_F^2 \\
- \eta k_{q,B} \left\| \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}) \right\|_F^2 . \qquad (20)
\end{aligned}$$

We next bound the squared Frobenius norm of the update. Taking expectation with respect to the stochastic sample $\xi_{i,t}$ yields:

$$\begin{aligned}\mathbb{E}\left[\left\|W_{q,i}^{(t+1)} - W_{q,i}^{(t)}\right\|_F^2\right] &= \mathbb{E}\Big[\big\|\eta^2 \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}, \xi_{i,t}) {A_{q,i}^{(t)}}^\top {B_{q,i}^{(t)}}^\top \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}, \xi_{i,t}) \\ &\quad - \eta \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}, \xi_{i,t}) {A_{q,i}^{(t)}}^\top A_{q,i}^{(t)} - \eta B_{q,i}^{(t)} {B_{q,i}^{(t)}}^\top \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)}, \xi_{i,t})\big\|_F^2\Big] \\ &\le 3\eta^4 K_{q,A}^2 K_{q,B}^2 G^4 + 3\eta^2 K_{q,A}^2 G^2 + 3\eta^2 K_{q,B}^2 G^2. \end{aligned} \tag{21}$$

Therefore, the final bound for query projection is:

$$\begin{aligned}\left\langle W_{q,i}^{(t+1)} - W_{q,i}^{(t)}, \nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\rangle_F + \frac{L}{2}\left\|W_{q,i}^{(t+1)} - W_{q,i}^{(t)}\right\|_F^2 &\le -\eta(k_{q,A} + k_{q,B})\left\|\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2 \\ &\quad + \eta^2 K_{q,A} K_{q,B} G^3 + \frac{3L}{2}\eta^2 K_{q,A}^2 G^2 \\ &\quad + \frac{3L}{2}\eta^2 K_{q,B}^2 G^2 + \frac{3L}{2}\eta^4 K_{q,A}^2 K_{q,B}^2 G^4. \end{aligned} \tag{22}$$

A.4.2 Value Projection Bound

For the value projection, we follow a similar approach to the query projection. Since the value projection involves two cases—local updates and global aggregation—we first analyze the relation between $U_{v,i}^{(t)}$ and $W_{v,i}^{(t)}$, and then between $V_{v,i}^{(t)}$ and $U_{v,i}^{(t)}$. The relation between $U_{v,i}^{(t)}$ and $W_{v,i}^{(t)}$ is similar to that of the query projection, and thus we can leverage the same bounding techniques.

Specifically, for the inner product term, we have:

$$\begin{aligned}\left\langle U_{v,i}^{(t+1)} - W_{v,i}^{(t)}, \nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\rangle_F &\le \eta^2 K_{v,A} K_{v,B} G^3 - \eta k_{v,A}\left\|\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2 \\ &\quad - \eta k_{v,B}\left\|\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2. \end{aligned} \tag{23}$$

For the squared update norm, we have:

$$\mathbb{E}\Big[\left\|U_{v,i}^{(t+1)} - W_{v,i}^{(t)}\right\|_F^2\Big] \le 3\eta^4 K_{v,A}^2 K_{v,B}^2 G^4 + 3\eta^2 K_{v,A}^2 G^2 + 3\eta^2 K_{v,B}^2 G^2. \tag{24}$$

Next, we work on the relation between $V_{v,i}^{(t)}$ and $U_{v,i}^{(t)}$, we have:

$$\begin{aligned} V_{v,i}^{(t)} - U_{v,i}^{(t)} &= W_{v,0} + \left(\frac{1}{K}\sum_{j=1}^K B_{v,j}^{(t+1)}\right)\left(\frac{1}{K}\sum_{j=1}^K A_{v,j}^{(t+1)}\right) - W_{v,0} - B_{v,i}^{(t+1)} A_{v,i}^{(t+1)} \\ &= \left(\frac{1}{K}\sum_{j=1}^K B_{v,j}^{(t+1)}\right)\left(\frac{1}{K}\sum_{j=1}^K A_{v,j}^{(t+1)}\right) - B_{v,i}^{(t+1)} A_{v,i}^{(t+1)} \end{aligned} \tag{25}$$

For clarity of presentation, we redefine the average of $A_{v,i}^{(t+1)}$ and $B_{v,i}^{(t+1)}$ among clients as:

$$\bar{A}_v^{(t+1)} = \frac{1}{K}\sum_{j=1}^K A_{v,j}^{(t+1)}, \qquad \bar{B}_v^{(t+1)} = \frac{1}{K}\sum_{j=1}^K B_{v,j}^{(t+1)}.$$

Then,

$$\begin{aligned} V_{v,i}^{(t)} - U_{v,i}^{(t)} &= \bar{B}_v^{(t+1)}\bar{A}_v^{(t+1)} - B_{v,i}^{(t+1)} A_{v,i}^{(t+1)} \\ &= \big(\bar{B}_v^{(t+1)} - B_{v,i}^{(t+1)}\big)\bar{A}_v^{(t+1)} + B_{v,i}^{(t+1)}\big(\bar{A}_v^{(t+1)} - A_{v,i}^{(t+1)}\big). \end{aligned} \tag{26}$$

Taking the Frobenius norm and applying the triangle inequality yields:

$$\begin{aligned}\left\|V_{v,i}^{(t)} - U_{v,i}^{(t)}\right\|_F^2 \leq \left\|\bar{B}_v^{(t+1)} - B_{v,i}^{(t+1)}\right\|_F^2 \left\|\bar{A}_v^{(t+1)}\right\|_F^2 \\ + \left\|B_{v,i}^{(t+1)}\right\|_F^2 \left\|\bar{A}_v^{(t+1)} - A_{v,i}^{(t+1)}\right\|_F^2. \end{aligned} \tag{27}$$

We next bound the two disagreement terms. For each client j, the local update of A_v over the last E iterations is:

$$\begin{aligned} A_{v,j}^{(t+1)} &= A_{v,j}^{(t-E+1)} - \sum_{t'=t-E+1}^{t} \eta \nabla_{A_v} \mathcal{L}_j\big(\mathbf{W}_i^{(t')}, \xi_{j,t'}\big) \\ &= A_{v,j}^{(t-E+1)} - \sum_{t'=t-E+1}^{t} \eta B_{v,j}^{(t')\top} \nabla_{W_v} \mathcal{L}_j\big(\mathbf{W}_i^{(t')}, \xi_{j,t'}\big). \end{aligned} \tag{28}$$

Therefore,

$$\begin{aligned} \bar{A}_v^{(t+1)} - A_{v,i}^{(t+1)} &= \frac{1}{K}\sum_{j=1}^{K}\big(A_{v,j}^{(t+1)} - A_{v,i}^{(t+1)}\big) \\ &= \frac{1}{K}\sum_{j=1}^{K}\sum_{t'=t-E+1}^{t} \eta\Big(\nabla_{A_v}\mathcal{L}_i(\mathbf{W}_i^{(t')}, \xi_{i,t'}) - \nabla_{A_v}\mathcal{L}_j(\mathbf{W}_i^{(t')}, \xi_{j,t'})\Big). \end{aligned} \tag{29}$$

In (29), the terms $A_{v,j}^{(t-E+1)}$ vanish when forming $\bar{A}_v^{(t+1)} - A_{v,i}^{(t+1)}$. This follows from the fact that all local variables are synchronized at the beginning of each communication round. Specifically, we assume:

$$A_{v,i}^{(t-E+1)} = A_v^{(t-E+1)}, \quad \forall i \in \{1, \ldots, K\},$$

which corresponds to the standard FL protocol where the global model is broadcast to all clients before performing E local updates. Consequently, $\frac{1}{m}\sum_{j=1}^{m} A_{v,j}^{(t-E+1)} - A_{v,i}^{(t-E+1)} = 0$.

We begin by expanding the disagreement between the averaged and local LoRA factor:

$$\left\|\bar{A}_v^{(t+1)} - A_{v,i}^{(t+1)}\right\|_F^2 = \left\|\frac{1}{K}\sum_{j=1}^{K}\sum_{t'=t-E+1}^{t} \eta\Big(\nabla_{A_v}\mathcal{L}_i\big(\mathbf{W}_i^{(t')}, \xi_{i,t'}\big) - \nabla_{A_v}\mathcal{L}_j\big(\mathbf{W}_j^{(t')}, \xi_{j,t'}\big)\Big)\right\|_F^2 \tag{30}$$

$$\leq \frac{\eta^2}{K^2}\sum_{j=1}^{K}\sum_{t'=t-E+1}^{t}\left\|\nabla_{A_v}\mathcal{L}_i\big(\mathbf{W}_i^{(t')}, \xi_{i,t'}\big) - \nabla_{A_v}\mathcal{L}_j\big(\mathbf{W}_j^{(t')}, \xi_{j,t'}\big)\right\|_F^2 \tag{31}$$

$$\leq \frac{\eta^2}{K^2}\sum_{j=1}^{K}\sum_{t'=t-E+1}^{t}\left\|B_{v,i}^{(t')\top}\nabla_{W_v}\mathcal{L}_i\big(\mathbf{W}_i^{(t')}, \xi_{i,t'}\big) - B_{v,j}^{(t')\top}\nabla_{W_v}\mathcal{L}_j\big(\mathbf{W}_j^{(t')}, \xi_{j,t'}\big)\right\|_F^2 \tag{32}$$

$$\leq \frac{\eta^2}{K^2}\sum_{j=1}^{K}\sum_{t'=t-E+1}^{t}\left\|B_{v,i}^{(t')\top}\big(\nabla_{W_v}\mathcal{L}_i\big(\mathbf{W}_i^{(t')}, \xi_{i,t'}\big) - \nabla_{W_v}\mathcal{L}_j\big(\mathbf{W}_j^{(t')}, \xi_{j,t'}\big)\big)\right\|_F^2 \tag{33}$$

$$\leq \frac{\eta^2}{K^2}\sum_{j=1}^{K}\sum_{t'=t-E+1}^{t}\left\|B_{v,i}^{(t')\top}\right\|_F^2\left\|\big(\nabla_{W_v}\mathcal{L}_i\big(\mathbf{W}_i^{(t')}, \xi_{i,t'}\big) - \nabla_{W_v}\mathcal{L}_j\big(\mathbf{W}_j^{(t')}, \xi_{j,t'}\big)\big)\right\|_F^2. \tag{34}$$

In (32), we substitute the gradient structure induced by the LoRA parameters $\nabla_{A_v}\mathcal{L}_j = B_{v,j}^{(t')\top}\nabla_{W_v}\mathcal{L}_j$. By applying the inequality $\|a-b\|_F^2 \leq 2\|a\|_F^2 + 2\|b\|_F^2$ and taking expectation

with respect to the stochastic sample $\xi_{i,t}$, we obtain:

$$\mathbb{E}[\Big\|\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t')},\xi_{i,t'})-\nabla_{W_v}\mathcal{L}_j(\mathbf{W}_j^{(t')},\xi_{j,t'})\Big\|_F^2] \tag{35}$$

$$\leq 2\mathbb{E}[\big\|\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t')},\xi_{i,t'})\big\|_F^2]+2\mathbb{E}[\big\|\nabla_{W_v}\mathcal{L}_j(\mathbf{W}_j^{(t')},\xi_{j,t'})\big\|_F^2] \tag{36}$$

$$=2\big\|\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t')})\big\|_F^2+2\big\|\nabla_{W_v}\mathcal{L}_j(\mathbf{W}_j^{(t')})\big\|_F^2\leq 4G^2. \tag{37}$$

By substitution of (37) and using Assumption 4.3, we obtain:

$$\big\|\bar{A}_v^{(t+1)}-A_{v,i}^{(t+1)}\big\|_F^2\leq\frac{\eta^2}{K^2}\sum_{j=1}^{K}\sum_{t'=t-E+1}^{t}4K_{v,B}^2G^2=\frac{4E\eta^2}{K}K_{v,B}^2G^2. \tag{38}$$

In a similar approach, we obtain a corresponding bound for the value projection B_v:

$$\big\|\bar{B}_v^{(t+1)}-B_{v,i}^{(t+1)}\big\|_F^2\leq\frac{\eta^2}{K^2}\sum_{j=1}^{K}\sum_{t'=t-E+1}^{t}4K_{v,A}^2G^2=\frac{4E\eta^2}{K}K_{v,A}^2G^2. \tag{39}$$

Substituting the bounds in (38) and (39) into (27) yields:

$$\begin{aligned}\big\|V_{v,i}^{(t)}-U_{v,i}^{(t)}\big\|_F^2&\leq\big\|\bar{B}_v^{(t+1)}-B_{v,i}^{(t+1)}\big\|_F^2\big\|\bar{A}_v^{(t+1)}\big\|_F^2+\big\|B_{v,i}^{(t+1)}\big\|_F^2\big\|\bar{A}_v^{(t+1)}-A_{v,i}^{(t+1)}\big\|_F^2\\&\leq\frac{4E\eta^2}{K}K_{v,A}^4G^2+\frac{4E\eta^2}{K}K_{v,B}^4G^2.\end{aligned} \tag{40}$$

Next, applying Young's inequality, we get:

$$\begin{aligned}\big\langle V_{v,i}^{(t)}-U_{v,i}^{(t)},\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\big\rangle_F&\leq\frac{1}{2\eta}\big\|V_{v,i}^{(t)}-U_{v,i}^{(t)}\big\|_F^2+\frac{\eta}{2}\big\|\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\big\|_F^2\\&\leq\frac{1}{2\eta}\left(\frac{4E\eta^2}{K}K_{v,A}^4G^2+\frac{4E\eta^2}{K}K_{v,B}^4G^2\right)+\frac{1}{2}\eta G^2\\&=\frac{2\eta E}{K}\left(K_{v,A}^4+K_{v,B}^4\right)G^2+\frac{1}{2}\eta G^2.\end{aligned} \tag{41}$$

Then we have:

$$\begin{aligned}\big\langle W_{v,i}^{(t+1)}-W_{v,i}^{(t)},\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\big\rangle_F+\frac{L}{2}\|W_{v,i}^{(t+1)}-W_{v,i}^{(t)}\|_F^2&\leq\big\langle U_{v,i}^{(t)}-W_{v,i}^{(t)},\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\big\rangle_F\\&+\big\langle V_{v,i}^{(t)}-U_{v,i}^{(t)},\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\big\rangle_F\\&+\frac{L}{2}\|U_{v,i}^{(t)}-W_{v,i}^{(t)}\|_F^2+\frac{L}{2}\|V_{v,i}^{(t)}-U_{v,i}^{(t)}\|_F^2.\end{aligned} \tag{42}$$

By substituting (23), (24), (40), and (41) in (42), we get:

$$\begin{aligned}\big\langle W_{v,i}^{(t+1)}-W_{v,i}^{(t)},\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\big\rangle_F+\frac{L}{2}\|W_{v,i}^{(t+1)}-W_{v,i}^{(t)}\|_F^2\leq&\\&-\eta(k_{v,A}+k_{v,B})\Big\|\nabla_{W_v}\mathcal{L}_i\big(W_{v,i}^{(t)}\big)\Big\|_F^2\\&+\frac{2\eta E}{K}\left(K_{v,A}^4+K_{v,B}^4\right)G^2+\frac{1}{2}\eta G^2\\&+\eta^2G^2\Bigg(\frac{2EL}{K}K_{v,A}^4+\frac{2EL}{K}K_{v,B}^4+K_{v,A}K_{v,B}G\\&+\frac{3L}{2}K_{v,A}^2+\frac{3L}{2}K_{v,B}^2\Bigg)+\frac{3L}{2}\eta^4K_{v,A}^2K_{v,B}^2G^4.\end{aligned} \tag{43}$$

A.4.3 Combined Bound for Query and Value Projections

Here, we substitute the query and value projection bounds in (22) and (43) in (8) to get the final bound:

$$\begin{aligned}\mathcal{L}_i(\mathbf{W}_i^{(t+1)}) \leq \mathcal{L}_i(\mathbf{W}_i^{(t)}) &- \eta(k_{q,A}+k_{q,B})\left\|\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2 + \eta^2 K_{q,A}K_{q,B}G^3 \\ &+ \frac{3L}{2}\eta^2 K_{q,A}^2G^2 + \frac{3L}{2}\eta^2 K_{q,B}^2G^2 + \frac{3L}{2}\eta^4 K_{q,A}^2K_{q,B}^2G^4 \\ &- \eta(k_{v,A}+k_{v,B})\left\|\nabla_{W_v}\mathcal{L}_i\big(W_{v,i}^{(t)}\big)\right\|_F^2 + \frac{2\eta E}{K}\left(K_{v,A}^4+K_{v,B}^4\right)G^2 + \frac{1}{2}\eta G^2 \\ &+ \eta^2G^2\Bigg(\frac{2EL}{K}K_{v,A}^4 + \frac{2EL}{K}K_{v,B}^4 + K_{v,A}K_{v,B}G \\ &+ \frac{3L}{2}K_{v,A}^2 + \frac{3L}{2}K_{v,B}^2\Bigg) + \frac{3L}{2}\eta^4K_{v,A}^2K_{v,B}^2G^4 \end{aligned} \tag{44}$$

By rearranging terms in (44), we obtain

$$\begin{aligned}\eta(k_{q,A}+k_{q,B})\left\|\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2 + \eta(k_{v,A}+k_{v,B})\left\|\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2 \leq \\ \mathcal{L}_i(\mathbf{W}_i^{(t)}) - \mathcal{L}_i(\mathbf{W}_i^{(t+1)}) \\ + \eta\left(\frac{2E}{K}\big(K_{v,A}^4+K_{v,B}^4\big)+\frac{1}{2}\right)G^2 \\ + \eta^2\Bigg(K_{q,A}K_{q,B}G + \frac{3L}{2}K_{q,A}^2 + \frac{3L}{2}K_{q,B}^2 \\ + \frac{2EL}{K}K_{v,A}^4 + \frac{2EL}{K}K_{v,B}^4 + K_{v,A}K_{v,B}G \\ + \frac{3L}{2}K_{v,A}^2 + \frac{3L}{2}K_{v,B}^2\Bigg)G^2 \\ + \frac{3L}{2}\eta^4\big(K_{q,A}^2K_{q,B}^2 + K_{v,A}^2K_{v,B}^2\big)G^4. \end{aligned} \tag{45}$$

Defining $k_{\min} \triangleq \min\big\{k_{q,A}+k_{q,B},\, k_{v,A}+k_{v,B}\big\}$, and dividing both sides of (45) by $\eta k_{\min}$ yields:

$$\begin{aligned}\left\|\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2 + \left\|\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2 \leq \frac{\mathcal{L}_i(\mathbf{W}_i^{(t)}) - \mathcal{L}_i(\mathbf{W}_i^{(t+1)})}{\eta k_{\min}} \\ + \frac{1}{k_{\min}}\left(\frac{2E}{K}\big(K_{v,A}^4+K_{v,B}^4\big)+\frac{1}{2}\right)G^2 \\ + \frac{\eta}{k_{\min}}\Bigg(K_{q,A}K_{q,B}G + \frac{3L}{2}K_{q,A}^2 + \frac{3L}{2}K_{q,B}^2 \\ + \frac{2EL}{K}K_{v,A}^4 + \frac{2EL}{K}K_{v,B}^4 \\ + K_{v,A}K_{v,B}G + \frac{3L}{2}K_{v,A}^2 + \frac{3L}{2}K_{v,B}^2\Bigg)G^2 \\ + \frac{3L}{2k_{\min}}\eta^3\big(K_{q,A}^2K_{q,B}^2 + K_{v,A}^2K_{v,B}^2\big)G^4. \end{aligned} \tag{46}$$

As the joint gradient norm over all trainable parameters on client i is:

$$\left\|\nabla\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2 \triangleq \left\|\nabla_{W_q}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2 + \left\|\nabla_{W_v}\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2.$$

By averaging (46) over $t = 1, \ldots, T$ and using the telescoping property of the loss terms, we obtain:

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\left\|\nabla\mathcal{L}_i(\mathbf{W}_i^{(t)})\right\|_F^2 \leq\ & \frac{\mathcal{L}_i(\mathbf{W}_i^{(1)}) - \mathcal{L}_i(\mathbf{W}_i^{(T)})}{\eta T k_{\min}} \\ & + \frac{1}{k_{\min}}\left(\frac{2E}{K}\left(K_{v,A}^4 + K_{v,B}^4\right) + \frac{1}{2}\right)G^2 \\ & + \frac{\eta}{k_{\min}}\Bigg(K_{q,A}K_{q,B}G + \frac{3L}{2}K_{q,A}^2 + \frac{3L}{2}K_{q,B}^2 \\ & + \frac{2EL}{K}K_{v,A}^4 + \frac{2EL}{K}K_{v,B}^4 \\ & + K_{v,A}K_{v,B}G + \frac{3L}{2}K_{v,A}^2 + \frac{3L}{2}K_{v,B}^2\Bigg)G^2 \\ & + \frac{3L}{2k_{\min}}\eta^3\left(K_{q,A}^2K_{q,B}^2 + K_{v,A}^2K_{v,B}^2\right)G^4. \end{aligned} \tag{47}$$

□

A.5 Hyperparameter Settings

In Table 7, we specify the learning rates used for the NLU tasks.

Table 7: Selected learning rates for different federated fine-tuning methods in NLU experiments using RoBERTa-large.

	MNLI-m	MNLI-mm	SST2	QQP	RTE	MLS
FedLoRA	0.01	0.01	0.02	0.005	0.005	0.01
FFA-LoRA	0.05	0.05	0.05	0.02	0.02	0.05
FedSA-LoRA	0.005	0.005	0.01	0.005	0.005	0.005
LoRA-FAIR	0.005	0.005	0.02	0.005	0.005	0.01
FedLoQA	0.005	0.005	0.005	0.005	0.005	0.005
FedLoVA	0.02	0.02	0.02	0.01	0.005	0.01

A.6 Performance Comparison Under an Equal Communication Budget

Table 1 in the main text reports accuracy at a fixed 1000 rounds without accounting for communication cost, which understates FedLoVA's advantage (although we see a constant performance gain among all datasets in this scenario as well). FedLoRA and LoRA-FAIR transmit $2\times$ the parameters per round, so a round-matched comparison is not a communication-matched one. Following prior federated LoRA work, which reports accuracy as a function of transferred bits rather than round count alone [31, 19], in this section, we instead evaluate FedLoRA and LoRA-FAIR at round 500, matching FedLoVA's cumulative communication at round 1000. Other baselines, including FedSA-LoRA, have the same communication overhead as FedLoVA per round, so they are also evaluated at round 1000.

As shown in Table 8, FedLoVA reaches an average accuracy of 85.14%, compared to 83.74% for FedLoRA and 83.14% for LoRA-FAIR, corresponding to improvements of 1.40% and 2.00%, respectively. The margin is larger than in the round-matched comparison of Table 1, indicating that the round-matched setting understates the advantage of FedLoVA.

To test whether this improvement is reliable rather than noise, we ran a paired t-test comparing FedLoVA's per-task accuracy against each baseline, paired by task across the six NLU datasets. FedLoVA's improvement is statistically significant against all three baselines: p=0.0150 (FedLoRA), p=0.0037 (LoRA-FAIR), and p=0.0287 (FedSA-LoRA).

A.7 Learning Rate Sensitivity Analysis

As reported in Table 7, FedLoQA achieves its highest performance at a relatively small learning rate (0.005) across all datasets, whereas FedLoVA maintains strong performance even when larger

Table 8: Accuracy (%) under an equal communication budget. FedLoRA and LoRA-FAIR are evaluated at round 500, and FedLoVA at round 1000, so that all methods have the same cumulative communication cost.

Method	MNLI-m	MNLI-mm	SST-2	QQP	RTE	MLS	Avg
FedLoRA	86.72 ± 0.84	87.11 ± 0.39	93.54 ± 0.86	83.84 ± 0.48	83.76 ± 0.56	67.49 ± 0.25	83.74
LoRA-FAIR	86.05 ± 0.37	86.71 ± 0.23	93.60 ± 0.82	83.01 ± 0.30	82.28 ± 3.54	67.20 ± 0.77	83.14
FedLoVA	$\mathbf{88.04 \pm 1.46}$	$\mathbf{87.81 \pm 0.41}$	$\mathbf{94.45 \pm 0.47}$	$\mathbf{86.23 \pm 0.65}$	$\mathbf{84.13 \pm 1.31}$	$\mathbf{70.18 \pm 0.28}$	**85.14**

learning rates are used. To examine this behavior, we conduct an analysis of learning rate sensitivity for both methods. In particular, we evaluate their performance over a range of learning rates on the SST-2 and MNLI-mm datasets, with the results summarized in Table 9. The results indicate that FedLoQA is sensitive to the learning rate, with noticeable performance degradation as the learning rate increases. In contrast, FedLoVA exhibits more robust behavior across a broader range of learning rates. This difference in behavior could be explained through the roles of different components in the attention mechanism. In particular, the value projection contributes linearly to the output. In contrast, the query projection affects the attention distribution through the softmax nonlinearity, making it more sensitive to scale and direction. Consequently, aggregating Q across heterogeneous clients may produce inconsistent attention patterns, making the optimization more sensitive to the learning rate and less stable at larger learning rates.

Table 9: Performance comparison of FedLoVA and FedLoQA across different learning rates on SST2 and MNLI-mm.

	MNLI-mm		SST2	
	FedLoQA	FedLoVA	FedLoQA	FedLoVA
$lr = 0.005$	$\mathbf{86.39 \pm 0.90}$	87.51 ± 0.33	$\mathbf{93.81 \pm 1.57}$	93.57 ± 0.42
$lr = 0.01$	37.58 ± 0.00	87.52 ± 0.50	93.60 ± 2.01	94.06 ± 0.54
$lr = 0.02$	33.09 ± 4.49	$\mathbf{87.81 \pm 0.41}$	56.04 ± 19.25	$\mathbf{94.45 \pm 0.47}$
$lr = 0.05$	35.7 ± 1.88	35.70 ± 1.88	57.02 ± 17.85	75.21 ± 23.51

A.8 Qualitative Analysis of Reasoning Behavior on GSM8K Dataset

To complement the quantitative results on GSM8K, we provide a qualitative comparison of the reasoning behavior exhibited by different federated fine-tuning methods. Table 10 presents several representative examples with varying question lengths and reasoning complexity. As shown in Table 10, FedLoVA consistently generates more coherent and logically structured reasoning steps, leading to correct final answers across both simple and complex mathematical problems. In contrast, other baselines often exhibit incorrect intermediate steps, or confusion even on relatively simple questions. These observations suggest that FedLoVA has more reliable generation behavior in federated settings.

Table 10: Examples of generation task on GSM8K for different federated fine-tuning methods.

	FedLoRA	**FFA-LoRA**	**FedSA-LoRA**	**FedLoQA**	**FedLoVA**
Q: John arm wrestles 20 people. He beats 80%. How many people did he lose to?	If he lost to 80% of people, then he lost to 80% of 20 people. 20 * 80% = 16 people. The answer is 16. ✗	John beat 80% of the people. 80% is the same as 80/100 = 0.8. So 0.8 of 20 is 20 x 0.8 = 16. The answer is 16. ✗	John beats 20 people. 80% of 20 is 20 x 0.8 = 16. The answer is 16. ✗	John beat 20 people. He beat 80% of the people. 20 * 80% = 16 people. The answer is 16. ✗	John arm wrestled 20 people. He beats 80% of them, which is 80 / 100 * 20 = 16. So he lost 20 - 16 = 4 people. The answer is 4. ✓

	FedLoRA	FFA-LoRA	FedSA-LoRA	FedLoQA	FedLoVA
Q: Liam is 16 years old now. Two years ago, Liam's age was twice the age of Vince. How old is Vince now?	Liam is 16 years old. Liam is 2 years younger than he is today. So, two years ago, Liam was 16 - 2 = 14. Two years ago, Liam's age was twice the age of Vince. So, two years ago, Liam was 14, which is twice the age of Vince, 2 x Vince = 14. So Vince's age was 14 / 2 = 7. Vince is 7 years old now. The answer is 7. ✗	Liam is 16 years old now. Two years ago, his age was 16 - 2 = 14. Two years ago, Liam's age was twice the age of Vince. That means that 14 is twice the age of Vince. So 14 / 2 = 7. The answer is 7. ✗	Liam is 16 now. Two years ago, Liam's age was twice the age of Vince. 16 - 2 is 14. Two years ago, Liam was 14. Twice the age of Vince, 2 x (Vince's age) = 14. So Vince must be half that number. Vince is 7 years old now. The answer is 7. ✗	Liam is 16 now. Two years ago, Liam was 16 - 2 = 14. Liam's age is twice the age of Vince. So Liam's age was 2 x Vince's age. So if Liam's age is 14, Vince's age was 14 ÷ 2 = 7. The answer is 7. ✗	Two years ago, Liam was 16 - 2 = 14 years old. That was twice the age of Vince, so Vince was 14/2 = 7 years old. Now Liam is 16 years old and Vince is 16 - 7 = 9 years old. The answer is 9. ✓
Q: A tub of ice cream costing $13 is now sold at $11. A packet of milk was sold at a discount of $0.5. How much will you save if you buy 2 tubs of ice cream and 4 packets of milk?	2 tubs of ice cream will cost 2 * 11 = $22. 4 packets of milk will cost 4 * 0.5 = $2. 22 + 2 = $24. The answer is $24. ✗	The original price of the ice cream is $13. After it is sold at $11, then we will save $13 - $11 = $2. The original price of the packet of milk is $0.5. We can buy 4 packets of milk for $0.5 x 4 = $2. So we have saved $2. So in total, we will save $2 + $2 = $4. The answer is 4. ✗	The original price of a tub of ice cream is $13. If the price is now $11, then that is $13 - $11 = $2 less than it used to cost. So you save $2 on the ice cream. If the price of the milk is reduced by $0.5, then you save $0.5. So you save $2 on the ice cream and $0.5 on the milk. So if you buy 2 tubs of ice cream and 4 packets of milk, then you will save a total of 2 * $2 + 4 * $0.5 = $5. The answer is $5. ✗	The tub of ice cream was originally $13 and now sold at $11, a total of $13 - $11 = $2 saving. 2 tubs of ice cream will save $2 x 2 = 4. A packet of milk was originally $0.5 and now sold at $0, a total of $0.5 - $0 = $0.5 saving. 4 packets of milk will save $0.5 x 4 = 2. In total, you save $2 + 2 = 4. The answer is 4. ✗	The original price of the ice cream is 13 dollars. Then it was sold at 11 dollars. So the discount is 13 - 11 = 2 dollars. If you buy 2 tubs, you will get a 2 * 2 = 4 dollar discount. There are 4 packets of milk. Then the total discount on the milk is 4 * 0.5 = 2 dollars. So you will get a 2 + 4 = 6 dollar discount. The answer is 6 dollars. ✓
Q: The great dragon, Perg, sat high atop mount Farbo, breathing fire upon anything within a distance of 1000 feet. Polly could throw the gold javelin, the only known weapon that could sleigh the dragon, for a distance of 400 feet, well within the reach of the dragon's flames. But when Polly held the sapphire gemstone, she could throw the javelin three times farther than when not holding the gemstone. If holding the gemstone, how far outside of the reach of the dragon's flames could Polly stand and still hit the dragon with the gold javelin?	The gold javelin can be thrown 400 feet. The sapphire gemstone allows Polly to throw it 3 times farther, which is 3 * 400 = 1200 feet. The answer is 1200. ✗	Polly can throw the javelin 400 feet when not holding the gemstone. But when holding the gemstone, she can throw the javelin for 400 * 3 = 1200 feet. So now, Polly can hit the dragon from 1200 feet away from the dragon. But since the dragon can breathe fire 1000 feet away from the dragon, Polly can be 1000 + 1200 = 2200 feet away. The answer is 2200. ✗	The dragon's flames could reach 1000 feet. Polly could throw the gold javelin 400 feet. Holding the sapphire gemstone, she could throw it 400 * 3 = 1200 feet. The answer is 1200. ✗	The dragon can breathe fire 1000 feet away. Polly can throw the javelin 400 feet. When Polly holds the gemstone, she can throw the javelin 400 * 3 = 1200 feet. The answer is 1200. ✗	Perg was on mount Farbo, which was 1000 feet away from the place where the javelin could hit Perg. Polly could hold the gemstone and throw the javelin 400 feet * 3 = 1200 feet. So the javelin would hit the dragon 1200 feet - 1000 feet = 200 feet away from the place where Polly was standing. The answer is 200. ✓

	FedLoRA	FFA-LoRA	FedSA-LoRA	FedLoQA	FedLoVA
Q: Jason works as a salesperson at a car dealership. He needs to sell 15 cars this month to earn a big bonus. He knows based on historical averages, that for every 25 telephone calls he makes to potential customers, he gets one person to come into the car dealership to look at new cars. And for every two customers that come into the car dealership, one will buy a car. Based on these average numbers, how many telephone calls would Jason need to make to sell 15 cars and earn his bonus?	If Jason sells 15 cars, then 15 ÷ 2 = 7.5 people will have bought cars. So to get 15 cars, Jason would need 15 ÷ 1 = 15 people to come into the car dealership. So he would need 15 ÷ 25 = 0.6 telephone calls to make in order to sell 15 cars. 0.6 is rounded up to 1. The answer is 1. ✗	There are originally 15 cars that he needs to sell. For every 25 calls, he gets one person to come into the dealership. So for 25 calls, he gets 1 person to come into the dealership. To sell 15 cars, he needs to get 15 customers into the dealership. 25 * 15 = 375 calls. The answer is 375. ✗	Jason needs to sell 15 cars. If for every 25 telephone calls he makes, he gets one person to come into the car dealership to look at new cars. He would need to make 25 x 15 = 375 telephone calls. And for every two customers that come into the car dealership, one will buy a car. If 15 cars are sold, then 2 x 15 = 30 customers came into the car dealership. This means he needs to make 375 / 30 = 12.5 telephone calls to sell each car and earn his bonus. The answer is 12.5. ✗	We know he needs to sell 15 cars to earn his bonus. So for every 25 telephone calls, he will get one customer to come into the dealership to look at cars. For every two customers that come in, one will buy a car. So for every 25 calls, he will get 1 customer to come in. 2 customers come in for every 50 calls. So 1 + 2 = 3 customers come in for every 25 + 50 = 75 calls. And for every 3 customers that come in, 1 buys a car. For every 75 calls, he will have 3 customers come in and 1 buys a car. So for every 75 calls, he will have 1 car sold. For every 75 calls he has to make, he can sell 1 car. So for 15 cars, he will have to make 75 / 1 = 75 x 15 = 1125 calls. The answer is 1125. ✗	Jason needs to sell 15 cars. He gets one car for every two people who come into the dealership to look at cars. So, to sell 15 cars, he needs 15 x 2 = 30 people who come into the car dealership. He knows based on historical averages, that for every 25 telephone calls he makes to potential customers, he gets one person to come into the car dealership to look at new cars. So, to get 30 people to come in and buy a car, Jason would need to make 25 x 30 = 750 telephone calls. The answer is 750. ✓

A.9 Communication Efficiency Analysis

To better capture the overall communication efficiency, we report other metrics, including communication rounds required to reach the target accuracy, local trainable parameters, and transmitted parameters per round. As shown in Table 11, FedLoVA is the fastest method among all datasets and demonstrates a significant improvement in convergence speed—up to $10\times$ faster compared to other baselines. In addition, although the number of trainable parameters in FedLoVA is similar to that of other methods (except for FFA-LoRA), it converges significantly faster. Furthermore, the number of transmitted parameters in FedLoVA is half that of FedLoRA, highlighting its superior communication efficiency.

Table 11: Comparison of different methods based on the number of communication rounds, local trainable parameters, and transmitted parameters per round.

Target Accuracy	MNLI-m 87%	MNLI-mm 86%	SST2 93%	QQP 82%	RTE 80%	MLS 67%	# Trainable Param.	# Communicated Param.
FedLoRA	276 (0.6×)	170 (0.6×)	200 (0.9×)	304 (0.6×)	295 (0.9×)	368 (0.6×)	1.83 M	0.78 M
FFA-LoRA	474 (0.4×)	414 (0.3×)	903 (0.2×)	311 (0.6×)	477 (0.6×)	286 (0.8×)	1.44 M	0.39 M
FedSA-LoRA	333 (0.5×)	330 (0.3×)	263 (0.7×)	287 (0.6×)	329 (0.8×)	360 (0.7×)	1.83 M	0.39 M
LoRA-FAIR	528 (0.3×)	381 (0.3×)	305 (0.6×)	342 (0.5×)	359 (0.8×)	358 (0.7×)	1.83 M	0.78 M
FedLoQA	475 (0.4×)	854 (0.1×)	290 (0.6×)	260 (0.7×)	295 (0.9×)	389 (0.6×)	1.83 M	0.39 M
FedLoVA	**175** (1×)	**110** (1×)	**179** (1×)	**182** (1×)	**292** (1×)	**239** (1×)	1.83 M	0.39 M

A.10 Ablation Study

In this section, we conduct ablation studies to investigate the impact of several key factors on the performance of FedLoVA, including different levels of data heterogeneity, the number of clients, and the LoRA rank. We also conduct some experiments to evaluate the robustness of FedLoVA under differential privacy (DP). All ablation studies are conducted on the MNLI-m and SST2 datasets. Each

ablation experiment is conducted with three independent random seeds, and we report the average performance across these runs.

Effect of data heterogeneity. To investigate the impact of data heterogeneity on FedLoVA, we follow [10, 28] and consider three data distribution settings: 1) IID, 2) mild non-IID modeled using a Dirichlet distribution with $\beta = 1.0$, and 3) severe non-IID modeled using a Dirichlet distribution with $\beta = 0.5$. The results for severe non-IID are reported in Section 5.1. For the IID and mild non-IID settings, we report performance after 700 communication rounds due to observed overfitting beyond this point. This behavior is expected, as models converge faster under lower heterogeneity. The results are summarized in Table 12, showing that FedLoVA consistently outperforms other baselines across different levels of data heterogeneity. Based on Table 12, FedLoVA improves accuracy on the MNLI-m dataset by 0.42% and 1.05% under the IID and mild non-IID settings, respectively, compared to LoRA. For the SST2 dataset, FedLoVA outperforms FedLoRA by 0.31% and 0.49% under the IID and mild non-IID scenarios, respectively. FedLoVA achieves these improvements while reducing the number of transmitted parameters by half compared to FedLoRA, demonstrating its effectiveness under varying levels of data heterogeneity

Table 12: Performance comparison under different levels of data heterogeneity.

	MNLI-m		SST2	
	IID	Non-IID ($\beta = 1.0$)	IID	Non-IID ($\beta = 1.0$)
FedLoRA	88.15 ± 0.03	87.46 ± 0.33	95.37 ± 0.10	94.57 ± 0.52
FFA-LoRA	88.09 ± 0.09	87.43 ± 0.14	95.64 ± 0.40	94.56 ± 0.34
FedSA-LoRA	88.12 ± 0.20	87.63 ± 0.28	95.49 ± 0.10	95.02 ± 0.89
FedLoQA	88.12 ± 0.09	87.26 ± 0.37	95.34 ± 0.14	94.52 ± 0.65
FedLoVA	**88.57** ± 0.10	**88.51** ± 0.04	**95.68** ± 0.23	**95.06** ± 0.09

Effect of number of clients. To demonstrate the scalability of FedLoVA to a larger number of clients, we extend our evaluation to settings with 10 and 50 clients under a severe non-IID data distribution. The results are reported in Table 13. As shown, FedLoVA outperforms other methods as the number of clients increases, demonstrating its effectiveness even in large-scale federated learning scenarios.

Table 13: Performance comparison under varying numbers of clients (10 and 50) in a severe non-IID setting.

	MNLI-m		SST2	
	10 clients	50 clients	10 clients	50 clients
FedLoRA	86.57 ± 0.54	88.53 ± 0.11	96.19 ± 0.31	96.17 ± 0.30
FFA-LoRA	86.59 ± 0.43	88.39 ± 0.11	95.51 ± 1.02	95.93 ± 0.15
FedSA-LoRA	85.93 ± 1.07	88.60 ± 0.50	96.13 ± 0.21	96.11 ± 0.40
FedLoQA	85.94 ± 0.47	88.49 ± 0.17	94.92 ± 0.50	95.48 ± 0.76
FedLoVA	**86.90** ± 0.73	**89.18** ± 0.10	**96.53** ± 0.46	**96.18** ± 0.21

Effect of LoRA rank. As the LoRA rank plays a crucial role in LoRA method, we investigate the impact of the LoRA rank on the performance of FedLoVA. A smaller LoRA rank r results in fewer trainable parameters and lower communication costs between clients and the server, while a larger r increases both the number of trainable parameters and the communication overhead. In Section 5.1, we set the LoRA rank to $r = 8$. Here, we further evaluate the performance with $r = 4$ and $r = 16$. For all settings, the LoRA scaling factor is set to $2r$ to ensure a fair comparison. The results are summarized in Table 14. As shown, FedLoVA achieves better performance than other methods across different LoRA ranks, demonstrating its robustness with respect to the number of trainable parameters.

Robustness under differential privacy. We evaluate how each method behaves when a standard privacy mechanism is applied, since any practical privacy-preserving deployment incurs a utility cost from such mechanisms. We evaluate all methods when differential privacy (DP) noise is added to local model parameters before transmission to the server, and we measure the resulting task accuracy. We use the Gaussian mechanism with a clipping norm of $C = 0.005$ and a privacy budget of $(\epsilon, \delta) = (50, 10^{-5})$.

Table 14: Performance comparison with different LoRA ranks ($r = 4$ and $r = 8$).

	MNLI-m		SST2	
	$r = 4$	$r = 16$	$r = 4$	$r = 16$
FedLoRA	87.80 ± 0.59	87.06 ± 0.23	94.18 ± 0.28	94.10 ± 0.64
FFA-LoRA	87.08 ± 0.57	87.40 ± 0.31	93.96 ± 0.64	94.23 ± 0.47
FedSA-LoRA	86.39 ± 1.57	85.53 ± 0.54	94.12 ± 0.76	93.73 ± 0.17
FedLoQA	82.89 ± 0.70	85.22 ± 1.04	93.51 ± 1.84	93.75 ± 0.99
FedLoVA	$\mathbf{88.32} \pm 0.77$	$\mathbf{87.73} \pm 0.79$	$\mathbf{94.19} \pm 0.47$	$\mathbf{94.45} \pm 0.25$

Table 15: Accuracy (%) when DP noise is added to the transmitted local parameters.

	FedLoRA	FFA-LoRA	FedSA-LoRA	FedLoQA	FedLoVA
SST-2	88.06 ± 5.84	78.95 ± 10.29	92.21 ± 3.17	91.54 ± 3.56	$\mathbf{92.83} \pm 2.15$
MNLI-m	80.41 ± 1.81	67.72 ± 6.43	$\mathbf{85.73} \pm 1.79$	79.73 ± 5.32	$\mathbf{85.73} \pm 1.75$

FedLoVA retains the highest accuracy with the lowest variance under DP noise, indicating that it is more robust to the utility cost typically incurred when adding a privacy-preserving mechanism. We note that this experiment demonstrates a favorable utility-privacy trade-off when a standard privacy mechanism, i.e., DP noise is applied.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly describe our main contributions, including the proposed FedLoVA and its communication efficiency for Federated LoRA Fine-Tuning of Large Language Models. These claims are consistent with the results shown in Section 5, within the scope of the evaluated datasets, models, and baselines.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations and future directions of FedLoVA are discussed in Section 6. Additional experimental analyses are provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The theoretical results are supported with the relevant assumptions stated in Section 4.2. Complete proofs and details are provided in Appendix A.4.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The main experimental setup and evaluation protocol are described in Section 5. Further implementation details are summarized in Appendix A.5.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code is uploaded in the supplementary material, and the code will be made public upon acceptance. The datasets and models used in this paper are publicly available, properly cited, and used in accordance with their applicable licenses and terms.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The main training and evaluation details are described in Section 5. Further implementation details and hyperparameter settings are provided in Appendix A.5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Although we do not include error bars, each main experiment is repeated five times and reported with the mean and standard deviation. The experimental setup is described in Section 5 and Appendix A.5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Section 5 reports the compute resources used in our experiments, and Appendix A.5 provides additional implementation details needed for reproduction.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research adheres to the NeurIPS Code of Ethics. It uses standard public benchmark datasets and does not involve human-subject studies or personally identifiable information.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The broader implications of this work are discussed in Section 6 and further addressed in Appendix A.1.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This paper does not release any pretrained models, datasets, or deployed systems that carry a high risk of misuse. Therefore, specific safeguards for high-risk release are not applicable.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The code will be made public upon acceptance. The datasets and models used in this paper are publicly available, properly cited, and used in accordance with their applicable licenses and terms.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: This paper does not introduce new datasets, pretrained models, or other assets requiring separate documentation.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper does not involve any human subjects and therefore does not require IRB approval.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: This paper does not use LLMs in any important, original, or non-standard way as part of the core methodology.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.