FEDGPA: WHAT TO ADAPT AND WHAT TO SHARE IN PERSONALIZED FEDERATED FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Personalized federated fine-tuning allows clients to adapt pre-trained models to local data while benefiting from knowledge shared across clients, but repeatedly synchronizing model updates can incur substantial communication cost. Existing methods typically make adaptation and sharing decisions at the granularity of complete layers, adapters, or other coarse components, overlooking the fact that different attention heads and feed-forward network (FFN) units within the same Transformer layer can contribute differently to downstream adaptation. We propose FEDGPA, a personalized federated fine-tuning framework that jointly determines adaptation placement and sharing roles at the granularity of intra-layer Transformer units. FEDGPA constructs a unified, cost-aligned candidate space of attention heads and FFN clusters, estimates the benefit of each candidate for global sharing and local adaptation, and assigns each unit to a *Global*, *Private*, or *Frozen* role. To limit the overhead of fine-grained selection, FEDGPA performs a one-time compressed role allocation; during subsequent federated training, only the always-shared task head and Global units are synchronized, while Private units are adapted and retained locally. Experiments with BERT-base and RoBERTa-base under non-IID federated settings show that FEDGPA maintains strong final test performance while substantially improving communication efficiency, reducing the cumulative communication required to reach a target accuracy by up to 76.5% compared with the personalized FedDPA baseline.

1 INTRODUCTION

Personalized federated fine-tuning enables multiple clients to adapt a shared pre-trained model to their local data while still benefiting from knowledge aggregated across clients. This setting is particularly important when client data are heterogeneous and cannot be freely centralized because of privacy or data-sharing constraints. Pre-trained foundation models have demonstrated strong representation and transfer capabilities across a wide range of downstream tasks (Devlin et al., 2019; Liu et al., 2019; Raffel et al., 2020), while federated learning provides a natural way to adapt them without exchanging raw data (McMahan et al., 2017; Kairouz et al., 2021). However, federated fine-tuning of large Transformer models is communication intensive because model updates must be repeatedly exchanged over many communication rounds, making communication a major bottleneck (McMahan et al., 2017; Woisetschläger et al., 2024).

Existing federated fine-tuning methods mainly reduce training and communication costs by limiting the parameters updated in each round. Parameter-efficient fine-tuning (PEFT) methods introduce lightweight modules such as Adapters (Houlsby et al., 2019) and LoRA (Hu et al., 2022), while federated variants synchronize only these additional parameters (Zhang et al., 2023; Wang et al., 2024; Sun et al., 2024; Bai et al., 2024; Yan et al., 2025; Wang et al., 2025a). Other methods directly update selected parameters, submodels, or Transformer layers in the pre-trained backbone (Peng et al., 2024; Wang et al., 2025b; Wu et al., 2026; Shen et al., 2025; Zeng et al., 2026). These approaches typically make adaptation decisions at the granularity of complete layers, external modules, or other coarse model components. This can be restrictive because attention heads and feed-forward network (FFN) substructures within the same Transformer layer can serve different functions (Voita et al., 2019; Michel et al., 2019; Geva et al., 2021). Under a limited adaptation budget, updating a whole layer or module therefore treats functionally different units uniformly. This may allocate capacity to less useful structures while overlooking more useful units elsewhere in the model.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

In federated applications, clients often differ in label distributions, sample populations, or local domains, resulting in non-independent and identically distributed (non-IID) data. Under such heterogeneity, globally aggregated updates enable cross-client knowledge transfer but may interfere with client-specific adaptation (Karimireddy et al., 2020; Li et al., 2020; Bian et al., 2026). Personalized federated learning separates shared and client-specific knowledge (Deng et al., 2020; Dinh et al., 2020; Li et al., 2021a). Recent personalized federated fine-tuning methods further introduce global/local adapters, separate LoRA branches, personalized prompts, or expert mixtures (Yang et al., 2024; Chen et al., 2024; Saha et al., 2025; Lu et al., 2024; Bian et al., 2026; Shen et al., 2026; Tran et al., 2025; Fan et al., 2025). However, their shared/private boundaries are also typically defined at the granularity of complete modules, layers, or other coarse components. As a result, functionally different units within the same Transformer layer are often forced to follow the same sharing role. More general methods such as FedSelect (Tamirisa et al., 2024) and FedLAG (Nguyen et al., 2026) further refine personalization through parameter- or layer-level selection, but do not explicitly determine sharing roles for functionally distinct intra-layer units.

Figure 1 illustrates this distinction. Existing approaches typically make adaptation and sharing decisions over complete layers or modules. In contrast, we make both decisions at the granularity of intra-layer functional units. Attention heads and FFN units can be selected independently for adaptation and assigned different sharing roles across clients.

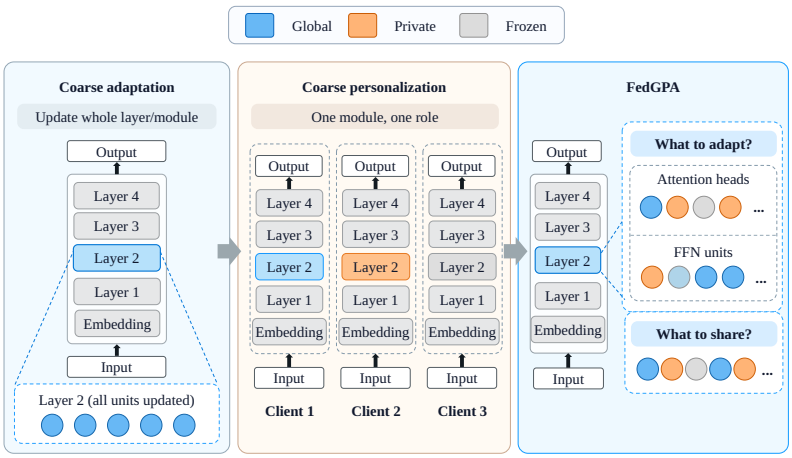


Figure 1: From coarse-grained to fine-grained adaptation and sharing decisions.

To examine the need for this finer granularity, we conduct controlled non-IID federated fine-tuning experiments and obtain three observations. First, under matched training and communication budgets, selecting fine-grained intra-layer units across the Transformer achieves greater adaptation gains than selecting a few complete layers. Second, different downstream tasks favor substantially different allocations between attention heads and FFN units, indicating that a fixed head/FFN composition can unnecessarily restrict adaptation. Third, under the same total adaptation budget, retaining a small amount of client-specific capacity improves personalized performance compared with sharing all selected units. Together, these observations motivate jointly deciding *which intra-layer units to adapt* and, among the adapted units, *which adapted units to share across clients or keep client-specific*.

These fine-grained decisions are also closely coupled with communication efficiency. Selecting more useful units can accelerate optimization and reduce the number of communication rounds required to reach a desired performance. Keeping client-specific units local reduces the amount of data repeatedly synchronized. At the same time, fine-grained unit selection itself introduces additional communication overhead. We therefore focus on reducing the *cumulative communication required to reach a target personalized accuracy*, rather than only the communication volume of a single round. This measure jointly accounts for the number of rounds needed to reach the target, the one-time overhead of selecting adaptation and sharing roles, and the recurring cost of synchronizing shared parameters.

We propose **FedGPA**, a personalized federated fine-tuning framework that performs fine-grained Global–Private allocation over intra-layer Transformer units. FedGPA first constructs a unified, cost-aligned candidate space of attention heads and FFN clusters, allowing units from different layers and functional types to compete under a common adaptation budget. It then estimates the benefit of each candidate for global sharing and local adaptation. Each unit is assigned to a **Global**,

Private, or **Frozen** role. Global units are adapted and synchronized across clients, Private units are adapted but remain client-specific, and Frozen units are left unchanged. To limit the overhead of fine-grained selection, FedGPA performs this role allocation only once using compressed statistics after a brief task-head warm-up. During subsequent federated training, only the always-shared task head and Global units are synchronized, while Private units continue to adapt locally without recurring communication.

Our main contributions are summarized as follows:

- We identify three key properties of personalized federated fine-tuning: fine-grained intra-layer units can provide greater adaptation gains than complete Transformer layers under the same budgets, different tasks require different attention/FFN allocations, and retaining client-specific capacity improves personalized performance under non-IID data.
- We propose **FedGPA**, which performs fine-grained Global–Private allocation over a unified, cost-aligned space of attention heads and FFN clusters. FedGPA evaluates each candidate for both global sharing and local adaptation, assigns units to Global, Private, or Frozen roles, and combines one-time compressed allocation with sparse shared synchronization to reduce cumulative communication.
- We evaluate FedGPA under non-IID settings on four GLUE tasks with BERT-base and on AG News and QNLI with RoBERTa-base. FedGPA maintains competitive final performance while substantially reducing cumulative communication to target accuracy, achieving up to a 76.5% reduction compared with the personalized FedDPA baseline.

2 Related Work

2.1 Efficient Federated Fine-Tuning: What to Adapt

Efficient federated fine-tuning reduces computation, memory, and communication while preserving adaptation quality. One line adopts parameter-efficient modules such as adapters, prompts, and LoRA with federated PEFT strategies for aggregation or heterogeneous adaptation (Zhang et al., 2023; Wang et al., 2024; Sun et al., 2024; Bai et al., 2024). Another reduces backbone adaptation cost through proxy/submodel construction, selective layer updates, or low-memory optimization, including FedPFT, DropPEFT, DevFT, FedKSeed, FedMeZO, and Spry (Peng et al., 2024; Wang et al., 2025b; Wu et al., 2026; Qin et al., 2024; Ling et al., 2024; Panchal et al., 2024). These methods typically decide *what to adapt* using coarse units such as PEFT modules, complete layers, proxy models, or generic parameter coordinates. FedGPA instead allocates adaptation capacity across attention heads and FFN clusters under a unified budget.

2.2 Personalized Federated Fine-Tuning: What to Share

Personalized federated fine-tuning retains client-specific knowledge while exploiting transferable information across clients. Existing methods use global/local adapters or adapter integration (Yang et al., 2024; Chen et al., 2024; Saha et al., 2025), decoupled LoRA branches (Lu et al., 2024; Bian et al., 2026; Shen et al., 2026), personalized prompts or expert mixtures (Tran et al., 2025; Fan et al., 2025), and client- or group-specific aggregation (Yang et al., 2026). FedDPA further combines global/local adapters with instance-wise dynamic weighting (Yang et al., 2024). However, these methods usually decide *what to share* at the granularity of entire adapters, LoRA branches, prompts, experts, layers, or other coarse components. FedGPA instead assigns sharing roles to intra-layer attention heads and FFN clusters.

2.3 Fine-Grained Shared–Private Decomposition

Beyond foundation-model fine-tuning, personalized federated learning has explored progressively finer shared/private decompositions. Early approaches separate global and personalized models through interpolation or regularization (Deng et al., 2020; Dinh et al., 2020; Li et al., 2021a), while later methods personalize representations, components, layers, or parameter subsets (Arivazhagan et al., 2019; Collins et al., 2021; Li et al., 2021b; Pillutla et al., 2022; Ma et al., 2022; Nguyen et al., 2026; Chen et al., 2025; Tamirisa et al., 2024). These decompositions are still typically defined

by architectural blocks, complete layers, or generic masks rather than function-aware Transformer units. Since attention heads and FFN substructures exhibit distinct functional roles (Voita et al., 2019; Michel et al., 2019; Geva et al., 2021), FedGPA jointly determines *what to adapt* and *what to share* at the intra-layer unit level.

3 Preliminary and Motivation

3.1 Layer-wise vs. Intra-layer Fine-tuning

Transformer layers consist of attention and FFN modules (Vaswani et al., 2017). Layer-wise fine-tuning updates selected layers in full, whereas intra-layer fine-tuning adapts finer-grained units such as attention heads and FFN neurons. The latter allows a fixed adaptation budget to be distributed more flexibly across layers and functional structures.

3.2 Observations

We conduct controlled non-IID experiments on SST-2 and QNLI with matched adaptation and communication budgets across configurations.

Observation 1. Fine-grained intra-layer units provide greater adaptation gains than complete layers under matched budgets.

We compare Layer-Select with Head-Select and Neuron-Select using the same adaptation and communication budgets, full-parameter FedAvg as a reference, and the same $|w| \cdot |g|$ first-order Taylor importance metric (Molchanov et al., 2019). As shown in Figure 2, head/neuron selection reaches the same SST-2 targets with substantially less cumulative communication than Layer-Select, motivating finer-grained adaptation.

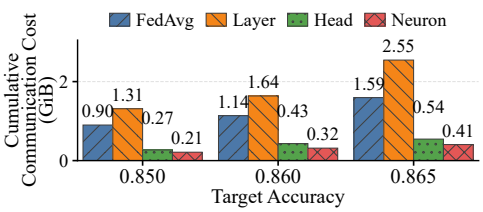


Figure 2: Cumulative Communication Cost.

Observation 2. Different tasks favor different attention-head/FFN-neuron allocations under the same total budget.

We vary the head/neuron composition while keeping the total adaptation and communication budget approximately fixed. Figure 3 shows different optima for SST-2 (0.75:0.25) and QNLI (0:1). Together with the distinct functions of attention heads and FFN structures (Voita et al., 2019; Geva et al., 2021), this suggests allocating adaptation capacity across unit types according to the task rather than using a fixed ratio.

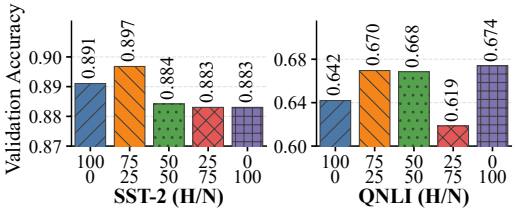


Figure 3: Task-dependent head/FFN allocation.

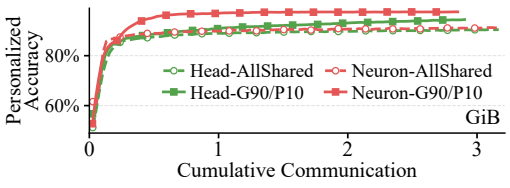


Figure 4: Effect of Private capacity.

Observation 3. How selected adaptation capacity is shared also matters: retaining a small Private subset improves personalized performance over sharing all selected units.

We compare an All-shared configuration with a Global–Private configuration using the same number of trainable units, where Global–Private retains a small client-specific subset selected for local benefit. Figure 4 shows higher weighted personalized accuracy for both Head-Select and Neuron-Select, demonstrating the value of Private capacity.

Together, Observations 1–2 address *what to adapt*: which intra-layer units and unit types should receive limited adaptation capacity. Observation 3 addresses *what to share*: which selected units should be globally synchronized or retained locally. These results motivate jointly making both decisions at the granularity of intra-layer functional units.

4 METHOD

4.1 PROBLEM SETTING AND COMMUNICATION COST

We consider personalized federated fine-tuning with N clients. Client i holds a local dataset $\mathcal{D}_i$ of n_i samples with loss $\mathcal{L}_i$, and the server maintains shared parameters $\mathbf{w}^t$ at communication round t. Let $\mathcal{U}$ denote the candidate adaptation units. The server maintains a Global set $\mathcal{G} \subseteq \mathcal{U}$ for cross-client aggregation, while client i maintains a Private set $\mathcal{P}_i \subseteq \mathcal{U} \setminus \mathcal{G}$ for local adaptation; all remaining units are Frozen. The task head forms an always-shared set $\mathcal{B}$, which participates in training and aggregation but is excluded from role allocation and the Global/Private budgets.

The candidate space consists of attention heads and clusters of FFN intermediate neurons. For a Transformer encoder with L layers, hidden dimension d, and head dimension d_h, an attention head and an FFN neuron contain $4dd_h + 3d_h$ and $2d + 1$ trainable parameters, respectively. To make the two unit types comparable under a common budget, we group FFN neurons by capacity-constrained k-means over their pre-trained parameter vectors (Bradley et al., 2000; Malinen & Fränti, 2014)

$$\phi_{\ell,j} = [W_\ell^{\text{int}}[j,:]; b_\ell^{\text{int}}[j]; W_\ell^{\text{out}}[:,j]],$$

using m neurons per cluster such that

$$c_H = 4dd_h + 3d_h, \qquad c_C = m(2d+1) \approx c_H. \tag{1}$$

This yields $\mathcal{U} = \bigcup_{\ell=1}^{L}(\mathcal{H}_\ell \cup \mathcal{C}_\ell)$, where $\mathcal{H}_\ell$ and $\mathcal{C}_\ell$ denote the attention-head and FFN-cluster sets in layer ℓ. With approximately aligned unit costs, we impose $|\mathcal{G}| \le K_G$ and $|\mathcal{P}_i| \le K_P$, with total adaptation budget $K_T = K_G + K_P$.

We measure communication efficiency by the cumulative communication required to reach a target personalized accuracy. For target accuracy τ, let T_τ be the first communication round at which the target is confirmed under the evaluation criterion, C_{alloc} the additional communication incurred by one-time role allocation beyond routine synchronization, and C_{sync}^t the routine synchronization cost at round t. We define

$$C_\tau = C_{\text{alloc}} + \sum_{t=1}^{T_\tau} C_{\text{sync}}^t. \tag{2}$$

FedGPA targets these three factors through role-conditioned one-step loss reduction for unit selection, one-time compressed allocation to control C_{alloc}, and a fixed sparse shared set to limit C_{sync}^t. Figure 5 illustrates the overall framework of FedGPA.

4.2 JOINT ADAPTATION AND SHARING ALLOCATION

FedGPA jointly determines *what to adapt* and *what to share* by allocating the limited budget to units with high expected benefit under either global sharing or local adaptation. Directly evaluating a candidate allocation through T_τ would require running the subsequent training trajectory, so we use a first-order approximation of its one-step loss reduction, following Taylor-based importance estimation (Molchanov et al., 2019).

Because the task head $\mathcal{B}$ is initially poorly adapted to the downstream label space, FedGPA first performs a brief head-only warm-up. For $t < t_{\text{sel}}$, clients update and aggregate only $\mathcal{B}$, while all candidate heads and FFN clusters remain frozen. Global importance is evaluated at t_{sel} from the current shared model, whereas Private importance is computed locally when each client initializes its Private set.

For client i and candidate unit u, let $\Delta\boldsymbol{\theta}_{i,u}$ denote the parameter change produced by one virtual optimization step on client i's local objective from the shared model $\mathbf{w}$ used for the corresponding

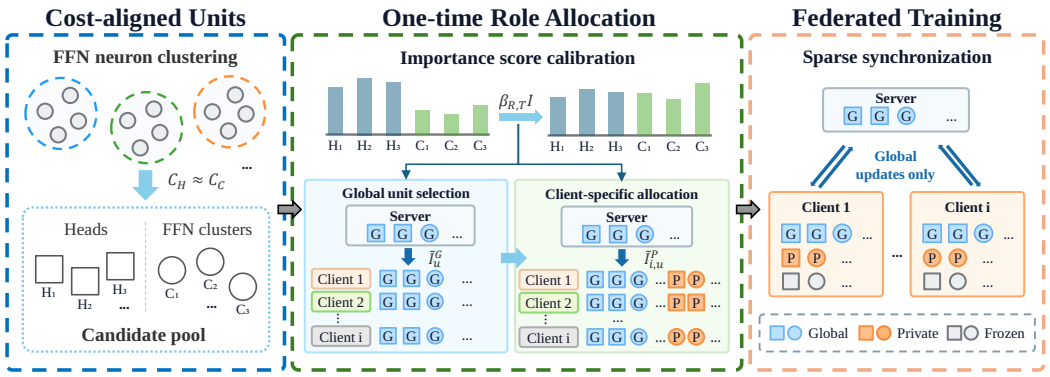


Figure 5: Framework of FedGPA. Cost-aligned heads and FFN clusters undergo one-time Global–Private allocation; only the task head and Global units are subsequently synchronized.

allocation, with only u updated. The one-step loss reduction is approximated by

$$I_{i,u}^{(1)} \approx -\langle \nabla_{\mathbf{w}_u} \mathcal{L}_i(\mathbf{w}), \Delta\boldsymbol{\theta}_{i,u} \rangle . \tag{3}$$

The expected benefit depends on the sharing role. A Private unit follows the client-local update, whereas a Global unit uses the aggregated update. Let $\mathcal{S}_{t_{\text{sel}}}$ be the participating clients and $\alpha_i = n_i / \sum_{j \in \mathcal{S}_{t_{\text{sel}}}} n_j$. The corresponding virtual updates are

$$\Delta\boldsymbol{\theta}_{i,u}^{P} = \Delta\boldsymbol{\theta}_{i,u}, \qquad \Delta\boldsymbol{\theta}_{u}^{G} = \sum_{j \in \mathcal{S}_{t_{\text{sel}}}} \alpha_j \Delta\boldsymbol{\theta}_{j,u}. \tag{4}$$

The role-conditioned importance scores are

$$I_u^G = -\sum_{i \in \mathcal{S}_{t_{\text{sel}}}} \alpha_i \left\langle \nabla_{\mathbf{w}_u} \mathcal{L}_i(\mathbf{w}), \Delta\boldsymbol{\theta}_u^G \right\rangle, \qquad I_{i,u}^P = -\left\langle \nabla_{\mathbf{w}_u} \mathcal{L}_i(\mathbf{w}), \Delta\boldsymbol{\theta}_{i,u}^P \right\rangle . \tag{5}$$

Although heads and FFN clusters have approximately aligned parameter costs, their surrogate scores may differ in scale. We therefore fit a calibration factor $\beta_{R,T}$ for each role $R \in \{G, P\}$ and unit type $T \in \{H, C\}$, using predicted and measured loss reductions on calibration set $\mathcal{V}_{R,T}$:

$$\beta_{R,T} = \frac{\sum_{\xi \in \mathcal{V}_{R,T}} I_\xi^R \Delta\mathcal{L}_\xi^R}{\sum_{\xi \in \mathcal{V}_{R,T}} \left(I_\xi^R\right)^2 + \epsilon_{\text{cal}}}, \qquad \widetilde{I}_\xi^R = \beta_{R,T} I_\xi^R. \tag{6}$$

FedGPA then performs Global-first allocation. The server first selects $\mathcal{G}^\star$, after which each client selects $\mathcal{P}_i^\star$ from $\mathcal{U} \setminus \mathcal{G}^\star$. Let $\text{TopK}(\{s_u\}_{u \in \mathcal{A}}, K)$ return up to K units with the largest positive scores. We obtain

$$\mathcal{G}^\star = \text{TopK}\left(\{\widetilde{I}_u^G\}_{u \in \mathcal{U}}, K_G\right), \qquad \mathcal{P}_i^\star = \text{TopK}\left(\{\widetilde{I}_{i,u}^P\}_{u \in \mathcal{U} \setminus \mathcal{G}^\star}, K_P\right). \tag{7}$$

4.3 Communication-Efficient Realization

One-Time Compressed Allocation. To control C_{alloc} in Eq. (2), role allocation is performed only once. Global allocation occurs at $t = t_{\text{sel}}$, while each client determines $\mathcal{P}_i$ once when its Private set is first initialized; the resulting roles remain fixed thereafter.

Global scoring requires aggregating candidate-level statistics and estimating the inner products in Eq. (5). Rather than uploading full candidate vectors, FedGPA applies CountSketch independently to each unit (Charikar et al., 2002). Its linearity supports weighted aggregation in sketch space, while consistent mappings enable approximate inner products (Rothchild et al., 2020; Larsen et al., 2021). The server thus estimates I_u^G, calibrates the scores, and selects $\mathcal{G}^\star$ without receiving full candidate-level vectors.

Private scores remain local; only low-dimensional statistics required for calibration are exchanged. After $\mathcal{G}^\star$ is determined, each client performs one-time Private allocation over $\mathcal{U} \setminus \mathcal{G}^\star$. Clients absent at the selection round initialize their Private sets upon their next participation.

Shared-Parameter Synchronization. After allocation, let $\mathcal{G} = \mathcal{G}^\star$ and $\mathcal{P}_i = \mathcal{P}_i^\star$. Let M_i denote the mask induced by $\mathcal{P}_i$, and $\mathbf{r}_i^t$ the client-specific residual retained on Private units. Client i's personalized model is

$$\boldsymbol{\theta}_i^t = \mathbf{w}^t + M_i \odot \mathbf{r}_i^t. \tag{8}$$

Each participating client updates only $\mathcal{B} \cup \mathcal{G} \cup \mathcal{P}_i$, but uploads changes only on $\mathcal{B} \cup \mathcal{G}$. Private residuals remain local for subsequent personalized training and evaluation. The server aggregates the shared updates using sample-size-weighted FedAvg (McMahan et al., 2017), while all other server parameters remain unchanged.

After allocation, for a fixed number of participating clients, routine communication depends only on the shared parameter set:

$$C_{\text{sync}}^t = \Theta\left(|\mathcal{B}| + \sum_{u \in \mathcal{G}} c_u\right), \tag{9}$$

where c_u is the parameter count of unit u. Under fixed $K_T = K_G + K_P$, increasing Private capacity does not increase routine synchronization volume, decoupling client-specific adaptation from recurring communication.

5 Experiments

We evaluate FedGPA with BERT-base on four GLUE tasks and with RoBERTa-base on AG News and QNLI. We report final test accuracy, cumulative communication to target accuracy, mechanism ablations, and budget sensitivity.

5.1 Experimental Setup

Datasets and Federated Setting. We evaluate BERT-base-uncased (Devlin et al., 2019) on SST-2, QNLI, MNLI, and QQP from GLUE (Wang et al., 2018), and RoBERTa-base (Liu et al., 2019) on AG News (Zhang et al., 2015) and QNLI. Each task is partitioned across 40 clients using balanced label-Dirichlet sampling (Hsu et al., 2019) with $\alpha = 0.1$; 20 clients participate per round for one local epoch. BERT-base and RoBERTa-base run for 150 and 120 rounds, respectively. Main results use three seeds with matched data partitions and client-participation schedules.

FedGPA and Baselines. Unless otherwise specified, FedGPA uses $K_G = 1$ and $K_P = 4$, excluding the always-shared task head $\mathcal{B}$, with three warm-up rounds before one-time Global allocation. For BERT-base, we compare against FedAvg (McMahan et al., 2017), FedLoRA (McMahan et al., 2017; Hu et al., 2022), LEFF (Shen et al., 2025), and FedDPA (Yang et al., 2024); RoBERTa-base experiments compare FedLoRA and FedDPA. Implementation details and method-specific hyperparameters are provided in Appendix A.

Metrics. We report final local held-out test accuracy and cumulative communication C_τ required to reach dataset-specific validation targets, using the same targets across methods and seeds. Validation is performed on all 40 clients every round, and validation/test accuracies are weighted by local evaluation-set size. FedGPA uses each client's personalized model, non-personalized baselines use their shared models, and FedDPA follows its personalized prediction rule.

A target τ is confirmed when the mean validation accuracy over the most recent five rounds remains at least τ for 10 consecutive evaluation windows. Communication is accumulated through the confirmation round and includes both uplink and downlink traffic for role allocation, associated statistics, and subsequent synchronization; routine synchronization uses FP16. Unreached targets are marked NR.

Table 1: BERT-base final test accuracy (%) over three seeds (mean±s.d.).

Method	SST-2	QNLI	MNLI	QQP
FedAvg	90.93±1.02	74.04±11.93	76.49±2.06	84.23±1.42
FedLoRA	87.29±2.17	66.55±14.33	70.92±4.61	76.21±3.69
LEFF	83.90±3.35	74.02±6.16	33.75±0.41	53.18±9.48
FedDPA	95.09±2.26	93.11±2.07	91.23±0.48	94.66±1.79
FedGPA (Ours)	**97.38±1.43**	**95.79±0.93**	**92.11±2.25**	**96.16±1.12**

5.2 Main Results

Final Performance. Table 1 shows that FedGPA achieves the highest mean final accuracy on all four BERT-base tasks, exceeding FedDPA by 0.88–2.68 percentage points. This indicates that fine-grained intra-layer adaptation can retain strong personalized performance despite updating only a sparse subset of the backbone.

Communication to Target Accuracy. Table 2 reports C_τ, while Figure 6 shows the accuracy–communication trajectories and FedGPA's communication breakdown. FedDPA is cheaper at some low targets, including 70% on SST-2 and QNLI, whereas FedGPA becomes more advantageous at higher targets. At 93% QNLI accuracy, FedGPA reduces mean C_τ from 2.248 to 0.529 GiB (76.5%); at 89% MNLI accuracy, it reduces C_τ from 1.335 to 0.753 GiB (43.6%), with similar high-target trends on SST-2 and QQP. The breakdown further shows that one-time allocation contributes only a small fraction of the cumulative cost, consistent with synchronizing only $\mathcal{B}$ and Global units thereafter.

Table 2: BERT-base cumulative communication C_τ (GiB) over three seeds (mean±s.d.).

Method	SST-2		QNLI	
	70%	93%	70%	93%
FedAvg	184.896±53.074	NR	372.511±242.850	NR
FedLoRA	1.812±1.219	NR	2.071±0.340	NR
LEFF	153.953±11.619	NR	222.699±37.911	NR
FedDPA	**0.311±0.000**	0.984±1.070	**0.311±0.000**	2.248±0.227
FedGPA (Ours)	0.421±0.031	**0.529±0.044**	0.361±0.017	**0.529±0.039**
Method	**MNLI**		**QQP**	
	50%	89%	70%	92%
FedAvg	122.359±0.000	NR	138.672±14.129	NR
FedLoRA	0.474±0.100	NR	0.651±0.164	NR
LEFF	NR	NR	NR	NR
FedDPA	0.311±0.000	1.335±0.289	0.311±0.000	0.851±0.514
FedGPA (Ours)	**0.303±0.009**	**0.753±0.490**	**0.283±0.009**	**0.352±0.026**

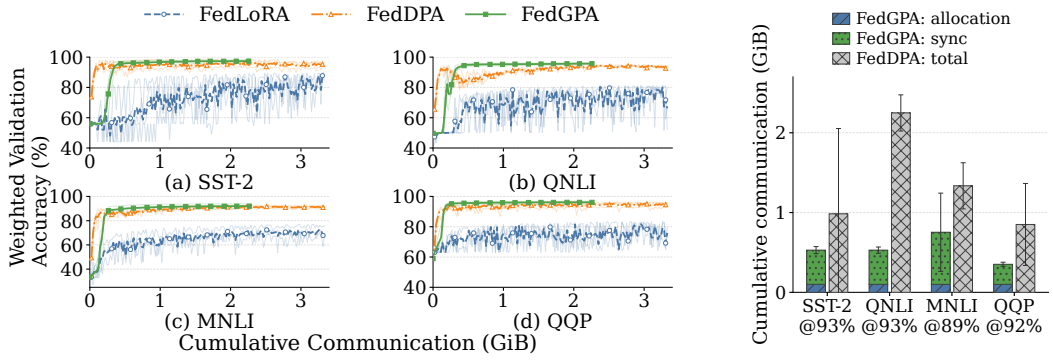


Figure 6: BERT-base communication efficiency on four GLUE tasks. Left: accuracy versus cumulative communication. Right: FedGPA communication breakdown at high-accuracy targets.

Table 3: RoBERTa-base results over three seeds (mean±s.d.).

Method	AG News		QNLI	
	Acc. (%)	$C_{96\%}$ (GiB)	Acc. (%)	$C_{93\%}$ (GiB)
FedLoRA	87.46±5.63	NR	75.27±11.26	NR
FedDPA	96.44±0.84	4.620±1.254	**95.25±1.29**	2.957±1.801
FedGPA (Ours)	**97.42±0.34**	**2.785±0.207**	95.08±1.34	**1.349±0.156**

Results with RoBERTa-base. Table 3 shows that FedGPA also generalizes to RoBERTa-base. Relative to FedDPA, it reduces mean C_τ by 39.7% on AG News and 54.4% on QNLI, while final accuracy is 0.98 points higher and 0.17 points lower, respectively.

5.3 Ablation Study

Table 4: Ablation results.

Variant	Acc. (%)	Comm. (GiB)		
		85%	88%	89%
w/o Importance	89.66	1.218	2.218	3.453
w/o Adaptive H/C	90.14	1.218	1.983	2.747
w/o Private	70.24	NR	NR	NR
FedGPA	**90.21**	**1.159**	**1.748**	**2.395**

Table 5: Sensitivity to adaptation budget.

K_T	Acc. (%)	Comm. (GiB)		
		80%	88%	90%
5	89.16	**0.382**	**0.976**	NR
10	89.79	0.631	1.310	NR
20	**90.32**	1.041	1.747	NR
40	90.30	1.862	2.685	**5.504**

We ablate three design choices on MNLI with $K_T = 20$ and 150 rounds. FedGPA, w/o Importance, and w/o Adaptive H/C use $K_G = 4$ and $K_P = 16$. w/o Importance randomly replaces selected units while preserving layer/type counts; w/o Adaptive H/C fixes the Global quota to 1 head and 3 clusters and each Private quota to 5 heads and 11 clusters; w/o Private uses $K_G = 20$ and $K_P = 0$.

Table 4 shows that random selection increases C_τ, confirming the importance of choosing *what to adapt*. Fixing the head/cluster composition retains similar final accuracy but increases communication at higher targets, showing the benefit of adaptive cross-type allocation. Removing Private adaptation reduces test accuracy to 70.24% and prevents all reported targets from being reached, confirming the importance of deciding *what to share* under non-IID data.

5.4 Budget Sensitivity

We study the performance–communication trade-off on MNLI by fixing the Global-to-Private ratio at 1:4 and varying $K_T \in \{5, 10, 20, 40\}$. All configurations run for 100 rounds, and we report final test accuracy together with cumulative communication to validation targets.

Increasing K_T from 5 to 20 raises test accuracy from 89.16% to 90.32%, while $K_T = 40$ provides similar final performance at 90.30%. Smaller budgets are more efficient at moderate targets, whereas only $K_T = 40$ reaches the 90% validation target, showing the expected trade-off between adaptation capacity, achievable performance, and communication cost.

6 Conclusion

We presented FedGPA, a personalized federated fine-tuning framework that jointly determines *what to adapt* and *what to share* at the granularity of intra-layer Transformer units. By combining fine-grained Global–Private allocation with one-time compressed selection and sparse synchronization, FedGPA reduces recurring communication while preserving client-specific adaptation. Experiments with BERT-base and RoBERTa-base show competitive final performance and up to a 76.5% reduction in cumulative communication to target accuracy relative to FedDPA.

AI USE STATEMENT

In this work, we used generative AI tools to provide feedback on the research methodology and experimental design, assist with mathematical formulation, method implementation, and translation, and support the interpretation of experimental results. We additionally used generative AI tools for literature search and summarization, manuscript drafting and editing for readability, and figure and presentation planning. All AI-assisted outputs were reviewed and verified by the authors against the cited literature, experimental implementation, and original experimental results. The authors made the final decisions regarding the method, experimental protocol, analysis, and manuscript content, and take responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We provide the methodological details of FedGPA in Section 4. Section 5.1 specifies the main experimental protocol and evaluation criteria, while implementation details and baseline configurations are provided in Appendix A.

REFERENCES

Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818*, 2019. URL https://arxiv.org/abs/1912.00818.

Jiamu Bai, Daoyuan Chen, Bingchen Qian, Liuyi Yao, and Yaliang Li. Federated fine-tuning of large language models under heterogeneous tasks and client resources. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-0461. URL https://papers.neurips.cc/paper_files/paper/2024/hash/1a134b50202088aa8c595cc99b310e5a-Abstract-Conference.html.

Jieming Bian, Lei Wang, Letian Zhang, and Jie Xu. FedALT: Federated fine-tuning through adaptive local training with rest-of-world LoRA. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 19728–19736, 2026. doi: 10.1609/aaai.v40i24.39054.

P. S. Bradley, K. P. Bennett, and A. Demiriz. Constrained K-means clustering. Technical Report MSR-TR-2000-65, Microsoft Research, 2000. URL https://www.microsoft.com/en-us/research/publication/constrained-k-means-clustering/.

Moses Charikar, Kevin Chen, and Martin Farach-Colton. Finding frequent items in data streams. In *Automata, Languages and Programming*, volume 2380 of *Lecture Notes in Computer Science*, pp. 693–703. Springer, 2002. doi: 10.1007/3-540-45465-9_59. URL https://doi.org/10.1007/3-540-45465-9_59.

Haokun Chen, Yao Zhang, Denis Krompass, Jindong Gu, and Volker Tresp. FedDAT: An approach for foundation model finetuning in multi-modal heterogeneous federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 11285–11293, 2024. doi: 10.1609/aaai.v38i10.29007. URL https://ojs.aaai.org/index.php/AAAI/article/view/29007.

Weihang Chen, Cheng Yang, Jie Ren, Zhiqiang Li, and Zheng Wang. Optimizing personalized federated learning through adaptive layer-wise learning. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 4860–4868, 2025. doi: 10.24963/ijcai.2025/541. URL https://www.ijcai.org/proceedings/2025/541.

Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*, 2020. URL https://arxiv.org/abs/2003.13461.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423/.

Canh T. Dinh, Nguyen H. Tran, and Tuan Dung Nguyen. Personalized federated learning with moreau envelopes. In *Advances in Neural Information Processing Systems*, 2020.

Dongyang Fan, Bettina Messmer, Nikita Doikov, and Martin Jaggi. On-device collaborative language modeling via a mixture of generalists and specialists. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 15833–15861, 2025. URL https://proceedings.mlr.press/v267/fan25h.html.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.emnlp-main.446.

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 *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2790–2799, 2019. URL https://proceedings.mlr.press/v97/houlsby19a.html.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. In *NeurIPS Workshop on Federated Learning*, 2019.

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. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14 (1–2):1–210, 2021. doi: 10.1561/2200000083.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5132–5143, 2020. URL https://proceedings.mlr.press/v119/karimireddy20a.html.

Kasper Green Larsen, Rasmus Pagh, and Jakub Tětek. Countsketches, feature hashing and the median of three. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6011–6020. PMLR, 2021. URL https://proceedings.mlr.press/v139/larsen21a.html.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems*, volume 2, 2020. URL https://proceedings.mlsys.org/paper/2020/hash/1f5fe83998a09396ebe6477d9475ba0c-Abstract.html.

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *Proceedings of the 38th International Conference on Machine Learning*, 2021a.

Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. FedBN: Federated learning on non-IID features via local batch normalization. In *International Conference on Learning Representations*, 2021b.

Zhenqing Ling, Daoyuan Chen, Liuyi Yao, Yaliang Li, and Ying Shen. On the convergence of zeroth-order federated tuning for large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1827–1838, 2024. doi: 10.1145/3637528.3671865.

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. URL https://arxiv.org/abs/1907.11692.

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. URL https://arxiv.org/abs/2406.07925.

Xiaosong Ma, Jie Zhang, Song Guo, and Wenchao Xu. Layer-wised model aggregation for personalized federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10082–10091, 2022. doi: 10.1109/CVPR52688.2022.00985.

Mikko I. Malinen and Pasi Fränti. Balanced k-means for clustering. In *Structural, Syntactic, and Statistical Pattern Recognition*, volume 8621 of *Lecture Notes in Computer Science*, pp. 32–41. Springer, 2014. doi: 10.1007/978-3-662-44415-3_4.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282, 2017. URL https://proceedings.mlr.press/v54/mcmahan17a.html.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems*, 2019.

Pavlo Molchanov, Arun Mallya, Stephen Tyree, Iuri Frosio, and Jan Kautz. Importance estimation for neural network pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11264–11272, 2019. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Molchanov_Importance_Estimation_for_Neural_Network_Pruning_CVPR_2019_paper.html.

Minh-Duong Nguyen, Hoang Khoi Do, Nam-Khanh Le, Nguyen Hoang Tran, Zhaohui Yang, Van-Duc Nguyen, and Trinh Van Chien. Toward layer-wise personalized federated learning: Adaptive layer disentanglement via conflicting gradients. *IEEE Transactions on Networking*, 34:3210–3225, 2026. doi: 10.1109/TON.2026.3653812.

Kunjal Panchal, Nisarg Parikh, Sunav Choudhary, Lijun Zhang, Yuriy Brun, and Hui Guan. Thinking forward: Memory-efficient federated finetuning of language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/7fc914993440219b64254e0c27964e11-Abstract-Conference.html.

Zhaopeng Peng, Xiaoliang Fan, Yufan Chen, Zheng Wang, Shirui Pan, Chenglu Wen, Ruisheng Zhang, and Cheng Wang. FedPFT: Federated proxy fine-tuning of foundation models. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 4806–4814, 2024. doi: 10.24963/ijcai.2024/531. URL https://www.ijcai.org/proceedings/2024/531.

Krishna Pillutla, Kshitiz Malik, Abdel-Rahman Mohamed, Mike Rabbat, Maziar Sanjabi, and Lin Xiao. Federated learning with partial model personalization. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 17716–17758. PMLR, 2022. URL https://proceedings.mlr.press/v162/pillutla22a.html.

Zhen Qin, Daoyuan Chen, Bingchen Qian, Bolin Ding, Yaliang Li, and Shuiguang Deng. Federated full-parameter tuning of billion-sized language models with communication cost under 18 kilobytes. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 41473–41497. PMLR, 2024. URL https://proceedings.mlr.press/v235/qin24a.html.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL https://jmlr.org/papers/v21/20-074.html.

Daniel Rothchild, Ashwinee Panda, Enayat Ullah, Nikita Ivkin, Ion Stoica, Vladimir Braverman, Joseph Gonzalez, and Raman Arora. FetchSGD: Communication-efficient federated learning with sketching. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8253–8265. PMLR, 2020. URL https://proceedings.mlr.press/v119/rothchild20a.html.

Pramit Saha, Divyanshu Mishra, Felix Wagner, Konstantinos Kamnitsas, and J. Alison Noble. FedPIA – permuting and integrating adapters leveraging wasserstein barycenters for finetuning foundation models in multi-modal federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 20228–20236, 2025. doi: 10.1609/aaai.v39i19.34228. URL https://ojs.aaai.org/index.php/AAAI/article/view/34228.

Jinglong Shen, Nan Cheng, Wenchao Xu, Haozhao Wang, Yifan Guo, and Jiajie Xu. LLM at network edge: A layer-wise efficient federated fine-tuning approach. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://papers.nips.cc/paper_files/paper/2025/hash/972ce3dca0f2d4a6f4bffbec859dae78-Abstract-Conference.html.

Zhikang Shen, Jianrong Lu, Haiyuan Wan, and Jianhai Chen. SDFLoRA: Selective decoupled federated LoRA for privacy-preserving fine-tuning with heterogeneous clients. In *Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence*, pp. 4786–4794, 2026. doi: 10.24963/ijcai.2026/533. URL https://doi.org/10.24963/ijcai.2026/533.

Youbang Sun, Zitao Li, Yaliang Li, and Bolin Ding. Improving LoRA in privacy-preserving federated learning. In *International Conference on Learning Representations*, 2024.

Rishub Tamirisa, Chulin Xie, Wenxuan Bao, Andy Zhou, Ron Arel, and Aviv Shamsian. FedSelect: Personalized federated learning with customized selection of parameters for fine-tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23985–23994, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Tamirisa_FedSelect_Personalized_Federated_Learning_with_Customized_Selection_of_Parameters_for_CVPR_2024_paper.html.

Linh Tran, Wei Sun, Stacy Patterson, and Ana Milanova. Privacy-preserving personalized federated prompt learning for multimodal large language models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/4431224d3762aa655f0aee4eaf04ff16-Abstract-Conference.html.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://papers.nips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 5797–5808. Association for Computational Linguistics, 2019. doi: 10.18653/v1/P19-1580.

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. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, 2018. doi: 10.18653/v1/W18-5446. URL https://aclanthology.org/W18-5446/.

Lei Wang, Jieming Bian, Letian Zhang, and Jie Xu. Adaptive LoRA experts allocation and selection for federated fine-tuning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 84623–84650, 2025a. doi: 10.52202/085713-2553. URL https://papers.neurips.cc/paper_files/paper/2025/hash/6df1b2b45e64d402588746f79b68b82c-Abstract-Conference.html.

Shilong Wang, Jianchun Liu, Hongli Xu, Jiaming Yan, and Xianjun Gao. Efficient federated fine-tuning of large language models with layer dropout. *arXiv preprint arXiv:2503.10217*, 2025b. URL https://arxiv.org/abs/2503.10217.

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. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-0708. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/28312c9491d60ed0c77f7fff4ad86dd1-Abstract-Conference.html.

Herbert Woisetschläger, Alexander Erben, Shiqiang Wang, Ruben Mayer, and Hans-Arno Jacobsen. A survey on efficient federated learning methods for foundation model training. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 8317–8325, 2024. doi: 10.24963/ijcai.2024/919. URL https://www.ijcai.org/proceedings/2024/919.

Yebo Wu, Jingguang Li, Zhijiang Guo, and Li Li. Learning like humans: Resource-efficient federated fine-tuning through cognitive developmental stages. In *International Conference on Learning Representations*, 2026.

Yunlu Yan, Chun-Mei Feng, Wangmeng Zuo, Rick Siow Mong Goh, Yong Liu, and Lei Zhu. Federated residual low-rank adaption of large language models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/906c860f1b7515a8ffec02dcdac74048-Abstract-Conference.html.

Yiyuan Yang, Guodong Long, Tao Shen, Jing Jiang, and Michael Blumenstein. Dual-personalizing adapter for federated foundation models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/45a30141c6719e9cfedfb51f1c665a37-Abstract-Conference.html.

Yiyuan Yang, Guodong Long, Qinghua Lu, Liming Zhu, and Jing Jiang. Bi-level personalization for federated foundation models: A task-vector aggregation approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 27702–27710, 2026. doi: 10.1609/aaai.v40i33.39991. URL https://ojs.aaai.org/index.php/AAAI/article/view/39991.

Yue Zeng, Jie Zhang, Song Guo, Baoliu Ye, Zhaihao Qu, Zicong Hong, Bin Tang, Jinyu Chen, Junlong Zhou, and Jiayin Yu. AFedLF: Adaptive layer freezing of foundation models in heterogeneous federated learning. *IEEE Transactions on Mobile Computing*, 25(4):5666–5682, 2026. doi: 10.1109/TMC.2025.3632655.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28, pp. 649–657, 2015. URL https://papers.nips.cc/paper/2015/hash/250cf8b51c773f3f8dc8b4be867a9a02-Abstract.html.

Zhuo Zhang, Yuanhang Yang, Yong Dai, Qifan Wang, Yue Yu, Lizhen Qu, and Zenglin Xu. FedPETuning: When federated learning meets the parameter-efficient tuning methods of pre-trained language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 9963–9977, 2023. doi: 10.18653/v1/2023.findings-acl.632. URL https://aclanthology.org/2023.findings-acl.632/.

A Implementation Details

Data Partitioning and Evaluation Splits. For each task, the official training data are distributed across 40 clients using balanced label-Dirichlet partitioning (Hsu et al., 2019) with concentration parameter $\alpha = 0.1$. Each client's local data are split approximately $80\%/10\%/10\%$ into training, validation, and held-out test sets. At each communication round, 20 clients participate and perform one local epoch. Training uses a batch size of 32 and a maximum sequence length of 128.

We evaluate BERT-base-uncased (Devlin et al., 2019) on SST-2, QNLI, MNLI, and QQP from GLUE (Wang et al., 2018), and RoBERTa-base (Liu et al., 2019) on AG News (Zhang et al., 2015) and QNLI. BERT-base experiments run for 150 communication rounds, whereas RoBERTa-base experiments run for 120 rounds. Main results use seeds 42, 43, and 44. For each seed, all compared methods use the same data partition and client-participation schedule. Ablation and budget sensitivity experiments use a single random seed.

FedGPA Configuration. Unless otherwise specified, FedGPA uses $K_G = 1$ and $K_P = 4$, giving a total adaptation budget of $K_T = 5$ excluding the always-shared task head $\mathcal{B}$. CountSketch uses sketch dimension $D = 1024$ with three repetitions. FedGPA first performs three task-head warm-up rounds and conducts one-time Global allocation in round 4; the warm-up rounds are included in the corresponding training horizon. Local optimization uses AdamW with a learning rate decayed from 6×10^{-6} to 3×10^{-6}.

Baseline Configurations. For the BERT-base experiments, we compare FedGPA with FedAvg, FedLoRA, LEFF, and FedDPA. FedAvg performs full-parameter federated fine-tuning using sample-size-weighted aggregation. FedLoRA applies rank-8 LoRA modules to the attention query and value projections and synchronizes the task head together with the LoRA parameters. LEFF (Shen et al., 2025) fine-tunes one Transformer layer per round with a context compression ratio of 0.8 and server-side distillation. FedDPA (Yang et al., 2024) uses its Global–Private adapter design with LoRA rank 8. The RoBERTa-base experiments compare FedGPA with FedLoRA and FedDPA. All methods use matched data partitions and client-participation schedules, with method-specific optimization hyperparameters.

Evaluation Protocol. Validation is performed on all 40 clients at every communication round. Validation and held-out test accuracies are weighted by the corresponding local evaluation-set sizes. FedGPA is evaluated using each client's current personalized model; non-personalized baselines use their shared models, while FedDPA follows its personalized prediction rule.

For a target validation accuracy τ, a target is confirmed when the mean validation accuracy over the most recent five rounds remains at least τ for 10 consecutive evaluation windows. We report cumulative communication C_τ through the confirmation round. Targets that are not confirmed within the corresponding training horizon are reported as NR.

Communication Accounting. Communication cost includes both uplink and downlink traffic. For FedGPA, the reported cost includes communication incurred by one-time role allocation, the associated calibration and allocation statistics, and subsequent shared-parameter synchronization. After role allocation, only parameters in the always-shared task head $\mathcal{B}$ and Global set $\mathcal{G}$ are repeatedly synchronized; Private residuals remain local. Routine parameter synchronization uses FP16 transmission. For all baselines, communication is computed from the actual transmitted parameter volume under their respective training protocols.