PRIVACY NEEDS THE RIGHT SPACE: RETHINKING FEDERATED LORA BEYOND RANK

Anonymous authors
Paper under double-blind review

ABSTRACT

Low-Rank Adaptation (LoRA) can still expose many coordinates to private optimization, while reducing the number of coordinates can discard useful adaptation directions. We study how both private dimension and retained update directions shape a compact trainable space for federated LoRA under clipping and DP noise. We propose FedTIE, a differentially private federated LoRA framework built on Tied Isometric Embedding (TIE), which uses privatized warm-up profiles to determine which factor coordinates across layers and factors share latent variables, with statistical load penalties regularizing the resulting assignment. Its sparse embedding preserves the Euclidean energy of latent perturbations when lifted to the concatenated factor space. The embedding is then frozen, and main-stage optimization and communication operate in the latent space. We establish an end-to-end sample-level differential privacy guarantee covering space selection and main training, together with a finite-horizon convergence bound for the induced subspace objective. On GLUE, FedTIE improves average accuracy over the strongest evaluated baseline by 3.40% and 4.44% at $\varepsilon = 6$ and 3, respectively, with $\delta = 10^{-5}$, while also improving non-private average accuracy and achieving competitive generation performance. On RoBERTa-base, FedTIE reduces per-round communication by 51.9% relative to FFA-LoRA and 85.2% relative to FedASK. Code is available at `https://anonymous.4open.science/r/fedhsip/`.

1 INTRODUCTION

Federated fine-tuning of pre-trained language models offers a promising paradigm for adapting foundation models to distributed, privacy-sensitive data without centralizing raw client datasets (Yin et al., 2021; Li et al., 2020; Zhang et al., 2026). However, full-parameter federated fine-tuning is often prohibitive due to the substantial memory, computation, and communication costs imposed on local clients. Parameter-efficient fine-tuning methods, especially Low-Rank Adaptation (LoRA) (Hu et al., 2021), have emerged as a practical alternative. By freezing the backbone model and training only a small set of low-rank adapters, LoRA substantially reduces the number of trainable parameters and the communication burden, making federated adaptation of large pre-trained models considerably more practical (Zhang et al., 2024; Guo et al., 2024a; Sun et al., 2024).

Although federated learning prevents direct sharing of raw data (McMahan et al., 2017), the transmitted gradients or model updates may still reveal sensitive information pertaining to local training samples. Differential privacy (DP) (Dwork & Roth, 2014), commonly instantiated via sample-level DP-SGD (Song et al., 2013; Bassily et al., 2014; Abadi et al., 2016), provides a rigorous framework to mitigate such leakage. Integrating DP into federated LoRA presents non-trivial challenges. When DP noise is injected into both LoRA factors, the noise components interact through the matrix product, producing a quadratic term that can substantially amplify the effective noise and degrade model utility (Kang et al., 2024; Sun et al., 2024). FFA-LoRA (Sun et al., 2024) freezes one shared factor to avoid cross-factor noise interactions. FedSVD (Lee et al., 2025) improves conditioning through SVD reparameterization, while FedASK (Wen et al., 2025) reconstructs privatized updates via two-stage sketching. LA-LoRA (Liu et al., 2026) and AS-LoRA (Kim et al., 2026) alternate or adaptively select which factor to privately train, respectively.

Low matrix rank alone does not determine the utility of private adaptation. Even when only B is privately trained, clients clip and perturb $q_B = \sum_{\ell=1}^{L} m_\ell r_\ell$ factor coordinates across L adapted

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.

modules, where m_ℓ and r_ℓ denote output width and rank. At fixed per-coordinate variance, the expected Gaussian noise energy grows linearly with this private dimension. Reducing the number of coordinates optimized under DP addresses this noise burden, but can also remove useful adaptation directions. Under client heterogeneity, the retained directions further determine which gradient components are combined before local clipping and server aggregation. Private federated LoRA therefore involves two coupled design choices: how many coordinates to optimize under DP, and which update directions they should parameterize.

By the right space, we mean a compact trainable subspace suited to the task, client distributions, and privacy budget: it should retain useful adaptation directions while limiting the combined effects of parameter-sharing error, clipping, and DP noise. Building on Uni-LoRA (Li et al., 2025), we study the affine factor subspace $\theta_{D,\mathrm{ref}} + \mathrm{range}(P)$, parameterized by $\theta_D = \theta_{D,\mathrm{ref}} + P\theta_d$ with $\theta_d \in \mathbb{R}^d$. Here, d determines how many latent coordinates are trained, while the columns of P determine which factor-update directions remain available. Unlike reducing each module's rank, global parameter sharing allows factor coordinates from different layers and factors to share the same trainable latent variables. The central question is therefore which coordinates can share a variable without imposing incompatible updates. Our goal is to improve utility under a fixed end-to-end privacy guarantee by choosing this sharing structure before per-example clipping and perturbation.

We propose FedTIE, a differentially private federated LoRA framework that uses Tied Isometric Embedding (TIE) to select shared update directions from privatized warm-up information. TIE compares the client-indexed update profiles of LoRA factor coordinates and favors sign-aligned sharing among coordinates with similar patterns, including coordinates from different layers or factors. This compatibility criterion guides which directions remain trainable; sensitivity, energy, heterogeneity, and occupancy penalties regularize the allocation of coordinates to latent slots. A sparse, column-normalized embedding satisfies $P^\top P = I_d$, so lifting a latent perturbation does not amplify its Euclidean energy in the concatenated factor space. The dimension d controls the direct Gaussian noise term at fixed per-coordinate variance, whereas the assignment determines the retained directions. After construction, FedTIE freezes the embedding, restarts from a public reference, and performs main-stage optimization, clipping, perturbation, and aggregation in the latent space. Warm-up and main-stage releases enter the end-to-end privacy accounting, so the benefit of selecting the space must also justify its privacy cost. Our contributions are summarized as follows:

- We frame private federated LoRA in terms of two coupled design choices: the private dimension and the trainable factor subspace. A conditional single-step decomposition separates discarded signal, projected clipping bias, and dimension-dependent Gaussian noise, explaining why compactness alone is insufficient.
- We introduce TIE, which uses privacy-accounted client update profiles to select cross-layer and cross-factor sharing patterns before clipping and perturbation. FedTIE combines this sparse isometric parameterization with latent-space DP-SGD and $O(d)$ main-stage communication, with an end-to-end sample-level privacy guarantee and a finite-horizon convergence bound for the induced subspace objective.
- Experiments show improved average GLUE accuracy under private and non-private training, competitive generation performance, and benefits across several heterogeneity settings. A comparison with a random isometric embedding supports statistics-informed sharing in the evaluated setting. On RoBERTa-base at rank $r = 8$, FedTIE trains 46,080 main-stage coordinates versus 147,456 for single-factor baselines, reducing communication by 51.9% relative to FFA-LoRA and 85.2% relative to FedASK.

2 Related Work

LoRA in Differentially Private Federated Learning. DP-LoRA Guo et al. (2024b); Liu et al. (2025) provides a straightforward solution by separately computing, clipping, and perturbing the per-sample gradients of the two factors, A and B, before averaging the privatized updates at the server. However, independently injected DP noise in A and B produces quadratic cross-noise terms when constructing the LoRA update $\Delta W = BA$, thereby amplifying the effective perturbation. FFA-LoRA (Sun et al., 2024) freezes a shared random factor A_0 and privately trains only B, avoiding simultaneous perturbation of both factors while restricting each layer's effective updates to the

row space of A_0. FedSVD (Lee et al., 2025) uses SVD-based reparameterization to obtain an orthonormal factor A for subsequent single-factor training, thereby changing gradient conditioning. FedASK (Wen et al., 2025) uses two-stage sketching and QR/SVD reconstruction to update the global factors from privatized local updates. LA-LoRA (Liu et al., 2026) alternates the active factor across local steps and optionally smooths privatized gradients, whereas AS-LoRA (Kim et al., 2026) selects an active factor for each layer and round using curvature-aware scores. These choices can affect subsequent local gradients and clipping. Their focus is factor selection, conditioning, or aggregation; they do not explicitly learn a global coordinate-sharing map from client update profiles. FedTIE studies this additional choice: which factor coordinates across layers and factors can share a limited set of latent variables optimized under DP before per-example clipping and perturbation.

Parameter Sharing and Subspace Design for Private Learning. Uni-LoRA (Li et al., 2025) parameterizes the concatenated LoRA factors with one globally shared vector and a sparse random embedding whose column normalization yields exact isometry. This establishes compact global sharing, but the random assignment does not use federated client update profiles. FedTIE builds on this representation and uses privatized warm-up information to select which factor coordinates share each latent variable. Its additional design criterion is compatibility of the shared directions with heterogeneous private training, with both space construction and subsequent training included in the privacy accounting. Low-dimensional structure has also been studied directly in private optimization. Projected DP-SGD (Zhou et al., 2020) estimates a gradient subspace from public data and projects noisy gradients onto it, reducing noise outside the selected subspace. FedTIE constructs a sparse sharing map from privacy-accounted client information and fixes the corresponding affine factor subspace for main-stage training. Per-example gradients are formed in its latent coordinates before clipping and perturbation. The distinction lies in the information used to select the space, the imposed sharing constraints, and the gradients seen by clipping.

3 METHOD

3.1 PRELIMINARIES AND MOTIVATION

Low-Dimensional LoRA Parameterization. Suppose the pretrained backbone contains L LoRA-adapted modules, with $B^\ell \in \mathbb{R}^{m^\ell \times r_\ell}, A^\ell \in \mathbb{R}^{r_\ell \times n^\ell}$, where r_ℓ is the LoRA rank. Following Uni-LoRA (Li et al., 2025), we row-flatten and concatenate all LoRA parameters as

$$\theta_D = \mathrm{Concat}\Big(\mathrm{vecrow}(B^1), \mathrm{vecrow}(A^1), \ldots, \mathrm{vecrow}(B^L), \mathrm{vecrow}(A^L)\Big) \in \mathbb{R}^D, \qquad (1)$$

where D is the total number of LoRA parameters. Rather than optimizing θ_D directly, we optimize $\theta_d \in \mathbb{R}^d$ with $d \ll D$ and recover

$$\theta_D = \theta_{D,\mathrm{ref}} + P\theta_d, \qquad P \in \mathbb{R}^{D\times d}. \qquad (2)$$

The fixed reference satisfies $B^\ell_{\mathrm{ref}} = 0$ and $A^\ell_{\mathrm{ref},ab} \sim \mathcal{U}[-1/\sqrt{n^\ell}, 1/\sqrt{n^\ell}]$, using a public, data-independent seed. At the start of Stage II, we set $\theta_d^{R_w} = 0$, yielding $B^\ell A^\ell = 0$ and thus preserving the pretrained output. For $G^\ell = \partial\mathcal{L}/\partial(B^\ell A^\ell)$, the initial gradients are $\frac{\partial\mathcal{L}}{\partial B^\ell} = G^\ell(A^\ell_{\mathrm{ref}})^\top, \frac{\partial\mathcal{L}}{\partial A^\ell} = 0$, avoiding vanishing gradients from all-zero factor initialization. Since $\theta_{D,\mathrm{ref}}$ is independent of private data, it incurs no privacy cost. Warm-up only selects P, and its provisional parameters are discarded before main training.

Why private dimension matters. Even when only B is trained, the private dimension is $q_B = \sum_{\ell=1}^{L} m^\ell r_\ell$, which grows with module count and width despite low matrix rank. For isotropic Gaussian noise with per-coordinate variance v^2, the expected noise energy in a q-dimensional coordinate space is qv^2. Fig. 1 illustrates this dimensionality burden. Reducing the private dimension addresses this term, but does not determine which adaptation directions remain available.

What makes a subspace suitable for private federated adaptation? The dimension d determines how many coordinates are perturbed, while $\mathrm{range}(P)$ determines which factor-update directions remain trainable. Following Uni-LoRA (Li et al., 2025), we consider sparse global sharing

maps with exact isometry, but compactness and isometry alone do not determine whether the retained directions are useful.

Fix the warm-up output, a model state, a minibatch $\mathcal{B}$, and an embedding satisfying $P^\top P = I_d$. Let $g_D(z)$ be the per-example gradient in the concatenated factor space, and define $\bar{g}_D = b_0^{-1} \sum_{z \in \mathcal{B}} g_D(z)$ and $a_d = b_0^{-1} \sum_{z \in \mathcal{B}} \mathrm{clip}_{C_\mathrm{m}}(P^\top g_D(z))$, using the predeclared denominator $b_0 > 0$. For $\xi \sim \mathcal{N}(0, v^2 I_d)$, orthogonality gives

$$\mathbb{E}_\xi \|P(a_d + \xi) - \bar{g}_D\|_2^2 = \underbrace{\|(I_D - PP^\top)\bar{g}_D\|_2^2}_{\text{discarded signal}} + \underbrace{\|a_d - P^\top \bar{g}_D\|_2^2}_{\text{projected clipping bias}} + \underbrace{dv^2}_{\text{Gaussian noise energy}}. \tag{3}$$

Here $v = \nu_\mathrm{m} C_\mathrm{m} / b_0$ for the normalized main-stage Gaussian release. Reducing d lowers the last term at fixed per-coordinate variance, but the chosen subspace also determines the discarded signal and projected clipping bias. At the same dimension and noise calibration, isometric embeddings have identical total Gaussian noise energy yet can retain different learning signals.

For example, consider a single-example gradient $g_D = (a, a)^\top$ with $0 < \sqrt{2}a \le C_\mathrm{m}$ and $b_0 = 1$. The embeddings $P_+ = (1, 1)^\top / \sqrt{2}$ and $P_- = (1, -1)^\top / \sqrt{2}$ are both one-dimensional and isometric. The first retains the entire gradient and incurs error v^2, whereas the second discards it and incurs error $2a^2 + v^2$. A small isometric space can therefore still discard the useful direction.

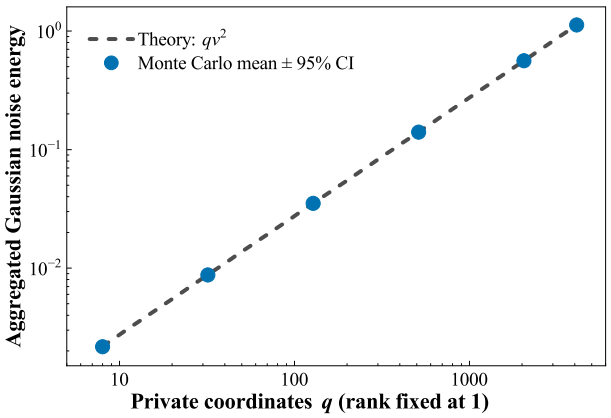

Figure 1: The noise cost of private dimension. Gaussian noise energy scales as qv^2 in synthetic rank-one LoRA at fixed v^2. Error bars: 95% confidence intervals for Monte Carlo means.

Under the sparse sharing map defined below, a latent gradient coordinate takes the form $g_{d,j} = n_j^{-1/2} \sum_{i \in G_j} \sigma_i g_{D,i}$. Gradient contributions from factor coordinates assigned to the same slot can reinforce or cancel one another before clipping. TIE uses privatized client update profiles as a proxy for deciding which coordinates can share a latent variable, while statistical load penalties regularize the assignment. This targets the choice of retained directions; normalization separately guarantees that lifting latent perturbations does not amplify their Euclidean energy.

3.2 Tied Isometric Embedding

We now introduce Tied Isometric Embedding (TIE), the core mapping of FedTIE. Appendix A distinguishes the properties of global sharing, load balancing, and isometry, and establishes the isometry of the actual statistics-driven signed construction. The end-to-end (ϵ, δ)-DP guarantee is established in Appendix B.

3.2.1 Embedding Parameterization

Let $i = 1, \ldots, D$ index factor coordinates of θ_D and let $j = 1, \ldots, d$ index latent coordinates of θ_d. We define an assignment map

$$\pi : \{1, \ldots, D\} \to \{1, \ldots, d\}, \tag{4}$$

where $\pi(i) = j$ means that the i-th factor coordinate is assigned to the j-th low-dimensional slot. We further introduce a sign variable $\sigma_i \in \{-1, +1\}$, which is used for optional within-group sign

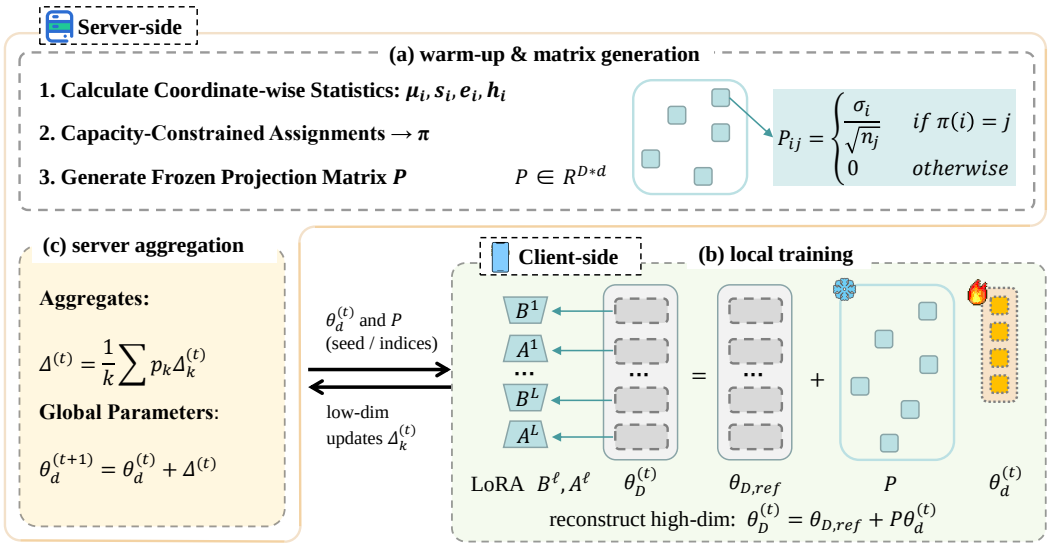

Figure 2: Overview of FedTIE. (a) The server uses privatized warm-up outputs to tie LoRA factor coordinates with compatible client update profiles to shared latent slots and constructs TIE. (b) Clients reconstruct LoRA factors through the frozen isometric embedding, perform sample-level DP-SGD in the latent space, and upload privatized low-dimensional updates.

alignment. Based on π and σ, we define the embedding matrix $P \in \mathbb{R}^{D\times d}$ as

$$P_{ij} = \begin{cases} \frac{\sigma_i}{\sqrt{n_j}}, & \pi(i) = j, \\ 0, & \text{otherwise}, \end{cases} \qquad n_j = |\{i : \pi(i) = j\}|\,. \tag{5}$$

Here n_j denotes the number of factor coordinates assigned to slot j. We require $1 \le d \le D$ and impose the hard constraint $n_j \ge 1$ for every j. Assignment is initialized with one distinct coordinate per slot, and subsequent moves cannot empty their source slot. If a candidate assignment contains empty slots, each is filled by moving the smallest-index coordinate from a largest occupied slot with at least two coordinates, breaking ties by slot index, before computing normalization. Thus $P^\top P = I_d$ holds exactly regardless of assignment randomness, signs, or load imbalance. The mapping and optional signs depend only on private warm-up outputs. In practice, the dense matrix P need not be materialized; assignment indices, signs, and normalization factors suffice. The coordinate relation $\theta_{D,i} - \theta_{D,\mathrm{ref},i} = \sigma_i \theta_{d,\pi(i)}/\sqrt{n_{\pi(i)}}$ makes the tying explicit: all coordinates in G_j share the same trainable variable $\theta_{d,j}$, up to fixed signs and normalization. Thus, D factor coordinates are controlled by only d trainable latent coordinates.

3.2.2 Warm-up Statistics Collection

To capture federated heterogeneity and DP-related sensitivity patterns, Stage I provisionally trains both LoRA factors in the full D-dimensional space from the public reference $\theta_{D,\mathrm{ref}}$, using only privatized optimization blocks and server aggregation between rounds. After R_w rounds, only the embedding is retained; Stage II restarts from the same reference with $\theta_d^{R_w} = 0$. Discarding warm-up parameters avoids reconstruction loss but forfeits their optimization progress.

Each factor coordinate i receives a profile (u_i, s_i, e_i, h_i). The client update profile $u_i \in \mathbb{R}^K$ summarizes standardized update patterns across clients, for example through the following normalized mean updates: $u_{i,k} = \frac{\mathbb{E}_t\left[\Delta_{k,i}^{(t)}\right]}{\sqrt{\mathbb{E}_t\left[\left(\Delta_{k,i}^{(t)}\right)^2\right]+\epsilon}}$. The nonnegative statistics s_i, e_i, and h_i summarize sample-level sensitivity, update energy, and cross-client variability, respectively, using proxies such as per-sample gradient magnitudes, mean squared updates, and variance of client mean updates. All moments and profiles are estimated from private outputs, yielding noisy, clipped proxies rather than exact population statistics or sensitivity bounds.

Specifically, each example contributes a vector $\Phi(z) \in \mathbb{R}^{m_{\mathrm{w}}}$ concatenating the provisional gradient and all statistic blocks with public normalization constants (Eq. 13). For each Poisson-sampled minibatch $\mathcal{B}^{\mathrm{w}}$, the client releases only

$$Y^{\mathrm{w}} = \sum_{z \in \mathcal{B}^{\mathrm{w}}} \frac{\Phi(z)}{\max(1, \|\Phi(z)\|_2 / C_{\mathrm{w}})} + \mathcal{N}(0, \nu_{\mathrm{w}}^2 C_{\mathrm{w}}^2 I_{m_{\mathrm{w}}}). \tag{6}$$

The optimizer extracts the optimization block, reverses its public scaling, and divides by the predeclared constant $b_{0,\mathrm{w}} > 0$. Profiles are computed solely from the client-indexed noisy transcript $\mathcal{Y}_{\mathrm{w}}$ (Appendix B.2); raw moments are never separately released. Each joint Gaussian release is accounted for in Stage I; profile construction, assignment, and provisional parameter aggregation are post-processing.

3.2.3 Cost-Aware Balanced Assignment

After warm-up, we assign factor coordinates to slots $G_j = \{i : \pi(i) = j\}$ using a heuristic that balances profile compatibility and proxy loads. Define $\mathbf{q}_j = (S_j, E_j, H_j, N_j) = \sum_{i \in G_j} (s_i, e_i, h_i, 1)$ and $\bar{\mathbf{q}} = d^{-1} \sum_j \mathbf{q}_j$. We seek a low-cost feasible assignment for

$$\min_{\substack{\pi,\sigma \\ N_j \geq 1,\ \forall j}} \sum_{j=1}^{d} \left[\alpha \mathcal{C}(G_j, \sigma) + \|\mathbf{q}_j - \bar{\mathbf{q}}\|_{\mathbf{W}}^2 \right], \tag{7}$$

where $\mathbf{W} = \mathrm{diag}(\beta, \gamma, \eta_h, \lambda)$, $\|x\|_{\mathbf{W}}^2 = x^\top \mathbf{W} x$, and

$$\mathcal{C}(G_j, \sigma) = \sum_{\substack{i,i' \in G_j \\ i < i'}} w_{ii'} \left[1 - \cos(\sigma_i u_i, \sigma_{i'} u_{i'}) \right]. \tag{8}$$

With optional weights $w_{ii'} \geq 0$, the conflict term favors sign-aligned client profiles to reduce parameter-sharing conflicts; the quadratic term balances sensitivity, energy, and heterogeneity proxies and occupancy. Exact isometry follows from normalization and nonempty slots, independently of this objective. The conflict term compares coordinates through their client-indexed profiles; it does not require agreement among clients. Because these profiles are standardized, their similarity does not guarantee compatible raw update magnitudes, whereas tying imposes equal-magnitude signed increments within a slot. The assignment is a proxy for useful sharing, not a direct minimizer of Eq. 3, and does not guarantee smaller clipping bias or client heterogeneity. Exact isometry preserves distances within the selected affine factor subspace, not arbitrary full-space directions or distances between effective weights BA.

3.3 Federated Optimization with TIE

After warm-up, we freeze the assignment, signs, and embedding P (Fig. 2) and optimize only latent increments from $\theta_d^{R_w} = 0$ (Appendix D). Clients cache the public reference seed and frozen embedding rule at setup. In round t, selected clients receive $\theta_d^{(t)}$, reconstruct $\theta_D^{(t)} = \theta_{D,\mathrm{ref}} + P\theta_d^{(t)}$ (Eq. 2), and reshape it into layer-wise LoRA factors. Each local step Poisson-samples $\mathcal{B}^{\mathrm{m}}$ at the declared record sampling probability, draws $Z \sim \mathcal{N}(0, \nu_{\mathrm{m}}^2 C_{\mathrm{m}}^2 I_d)$, and updates

$$\theta_d \leftarrow \theta_d - \frac{\eta}{b_{0,\mathrm{m}}} \left[\sum_{z \in \mathcal{B}^{\mathrm{m}}} \mathrm{clip}_{C_{\mathrm{m}}}(P^\top g_D(z)) + Z \right]. \tag{9}$$

Noise precedes division, including for empty batches, which use a zero sum and still execute the step. Both stages use positive denominators $b_{0,\mathrm{w}}, b_{0,\mathrm{m}}$ fixed independently of unprivatized data, never realized batch sizes. Expected batch sizes are allowed only from a public sampling rate and a public reference size fixed across neighboring datasets. Both noise multipliers follow the end-to-end accountant in Appendix B.5. After τ steps, client k uploads $\Delta_k^{(t)} = \theta_{d,k}^{(t,\mathrm{final})} - \theta_d^{(t)}$, and the server updates $\theta_d^{(t+1)} = \theta_d^{(t)} + \sum_{k=1}^{K} p_k \Delta_k^{(t)}$. Weights p_k are predeclared or derived from private outputs, normalized over participants, and zero otherwise; unprotected dataset sizes are excluded. The shared reference and frozen embedding make latent averaging equivalent to averaging reconstructed factor vectors. We also provide a finite-horizon convergence analysis of FedTIE's main-stage optimization in Appendix C.

3.4 Complexity Analysis

Since each row of P contains exactly one nonzero entry, reconstructing $\theta_D = \theta_{D,\text{ref}} + P\theta_d$ takes $O(D)$ time, including the reference addition. The implicit embedding requires $O(D + d)$ indices, signs, and normalization values instead of a dense $D \times d$ matrix. The public reference factors require up to $O(D)$ additional values if materialized, or can be regenerated from a public seed and the declared initialization rule. Their initial communication consists of this seed and rule, rather than a private learned reference vector. Warm-up and embedding construction incur separate setup costs; the frozen assignment requires $O(D + d)$ setup metadata. The $O(d)$ per-round parameter communication applies after clients have cached that metadata and the reference seed.

4 Experiments

4.1 Experimental Setups

In this section, we evaluate the proposed method through a series of experiments. We first compare FedTIE with state-of-the-art federated fine-tuning methods on the GLUE benchmark, including FedAvg, FFA-LoRA, FedSVD, FedASK, LA-LoRA and AS-LoRA. We then extend our analysis to natural language generation tasks. To ensure reproducibility, Appendix E provides detailed descriptions of the datasets, baseline methods, and hyperparameter settings for each experiment.

4.2 Main Results

Natural language understanding. Table 1 reports results on four GLUE tasks under non-private training and sample-level DP-SGD with $\epsilon \in \{3, 6\}$ and $\delta = 10^{-5}$. FedTIE achieves the highest average accuracy in all three settings: 87.76 without DP, 83.25 at $\epsilon = 6$, and 82.87 at $\epsilon = 3$, outperforming the strongest baseline by 1.09%, 3.40%, and 4.44%, respectively. Under DP, it also performs best on all five reported task scores, while tightening the privacy budget from $\epsilon = 6$ to $\epsilon = 3$ reduces its average accuracy by only 0.38%. These results show that FedTIE preserves strong non-private utility and yields larger gains under DP, consistent with the benefit of optimizing in a shared low-dimensional private space.

Heterogeneity of the data distribution (α). We evaluate robustness to client heterogeneity by partitioning MNLI with Dirichlet concentration parameters $\alpha \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$, where smaller α yields more skewed client distributions. Under DP-SGD with $(\epsilon, \delta) = (6, 10^{-5})$, FedTIE performs best for every $\alpha \geq 0.2$ and achieves the highest mean accuracy of 75.62%, exceeding FedASK and FedSVD by 2.96% and 3.64%, respectively (Table 2). At the most heterogeneous setting ($\alpha = 0.1$), FedTIE trails FedSVD and FedASK but remains stronger than the other baselines. The sharp recovery for $\alpha \geq 0.2$ suggests that statistics-aware grouping preserves compatible client directions once warm-up profiles adequately represent the participating distributions. Overall, these results demonstrate that FedTIE is robust across a broad range of non-IID distributions, while also exposing its limitation under extreme client heterogeneity.

Natural language generation. Table 3 reports non-private results on the E2E NLG Challenge using GPT-2 with LoRA rank $r = 4$. FedTIE achieves the best score on four of the five metrics: 41.80 BLEU, 6.33 NIST, 38.37 MET, and 1.93 CIDEr, improving over the strongest baseline by 1.11, 0.27, 0.42, and 0.10, respectively. Its ROUGE-L score of 56.69 is also competitive, trailing the best result by only 0.10. This cross-metric consistency suggests that the low-dimensional parameterization preserves both surface realization quality and broader agreement with human references. These results indicate that the shared low-dimensional parameterization remains effective for generation while retaining the θ_d-only communication interface used in the main training stage.

Communication cost. Table 4 reports communication averaged over all 150 rounds on RoBERTa-base with LoRA rank $r = 8$: 15 warm-up rounds exchanging standard LoRA matrices and 135 main-training rounds exchanging only the shared latent variable θ_d of dimension 46,080. FedTIE averages 70,963 parameters per direction per round, yielding the lowest upload-plus-broadcast cost among all evaluated methods. Including warm-up, this reduces total communication by 51.9% relative to FFA-LoRA and AS-LoRA, 67.9% relative to FedSVD, 75.9% relative to FedAvg and LA-LoRA,

Table 1: Results on five GLUE tasks under non-private and differentially private settings. For DP experiments, we use DP-SGD with $\epsilon \in \{3, 6\}$ and $\delta = 10^{-5}$. The best and second-best results are highlighted in **bold** and underline, respectively. Standard deviations marked with † are randomly simulated placeholders for formatting purposes.

DP Budget	Method	MNLI Matched	MNLI Mismatched	SST-2	QQP	QNLI	Average
Non-DP	FedAvg	85.01 ± 2.61	85.50 ± 0.58	50.92 ± 1.41	86.89 ± 1.24	50.56 ± 2.93	71.78 ± 2.73
	FFA-LoRA	83.02 ± 3.44	83.42 ± 0.79	93.00 ± 1.89	84.54 ± 0.60	84.66 ± 1.22	85.73 ± 2.17
	FedSVD	83.50 ± 0.59	83.79 ± 1.16	92.55 ± 2.64	85.32 ± 2.30	88.14 ± 1.23	86.67 ± 2.44
	FedASK	**85.42** ± 3.17	**85.65** ± 0.52	50.92 ± 3.16	**87.13** ± 2.80	50.74 ± 1.62	71.97 ± 1.01
	LA-LoRA	80.06 ± 2.21	80.33 ± 1.52	91.28 ± 0.56	82.78 ± 2.54	86.91 ± 1.40	84.27 ± 1.66
	AS-LoRA	81.34 ± 3.76	82.00 ± 3.01	90.48 ± 0.99	84.44 ± 2.40	88.17 ± 2.75	85.29 ± 2.23
	FedTIE	83.94 ± 3.66	84.08 ± 1.61	**94.04** ± 0.81	85.65 ± 0.82	**91.09** ± 3.30	**87.76** ± 2.49
$\epsilon = 6$	FedAvg	75.16 ± 5.68	75.64 ± 5.23	82.57 ± 4.11	79.21 ± 6.64	53.49 ± 3.20	73.21 ± 4.20
	FFA-LoRA	66.61 ± 5.81	68.01 ± 4.59	91.17 ± 6.00	80.39 ± 4.35	80.43 ± 5.09	77.32 ± 1.27
	FedSVD	74.95 ± 2.32	76.00 ± 2.68	88.42 ± 1.46	79.38 ± 2.35	80.51 ± 1.59	79.85 ± 2.61
	FedASK	75.96 ± 4.69	76.45 ± 3.12	80.16 ± 3.15	79.25 ± 2.22	61.41 ± 2.55	74.65 ± 6.43
	LA-LoRA	59.93 ± 2.53	62.35 ± 3.49	88.30 ± 1.35	75.56 ± 2.02	75.32 ± 4.43	72.29 ± 1.18
	AS-LoRA	73.81 ± 6.77	74.96 ± 5.53	85.67 ± 1.93	78.90 ± 3.54	73.06 ± 5.41	77.28 ± 1.71
	FedTIE	**78.93** ± 2.76	**79.92** ± 2.53	**92.20** ± 1.99	**80.42** ± 2.23	**84.77** ± 1.95	**83.25** ± 3.20
$\epsilon = 3$	FedAvg	73.61 ± 6.74	74.37 ± 4.71	79.01 ± 4.23	77.84 ± 4.97	57.39 ± 5.89	72.44 ± 5.50
	FFA-LoRA	53.96 ± 2.33	56.38 ± 1.19	91.06 ± 2.83	79.85 ± 2.55	78.11 ± 2.22	71.87 ± 6.47
	FedSVD	72.79 ± 6.08	74.51 ± 2.83	88.76 ± 4.80	78.29 ± 3.29	77.80 ± 6.30	78.43 ± 3.66
	FedASK	73.09 ± 2.54	74.89 ± 2.43	82.91 ± 4.26	78.77 ± 2.52	62.84 ± 4.39	74.50 ± 6.21
	LA-LoRA	49.75 ± 2.01	51.54 ± 3.19	86.47 ± 1.02	74.15 ± 2.31	72.36 ± 3.25	66.85 ± 1.24
	AS-LoRA	70.51 ± 3.95	72.63 ± 5.67	68.92 ± 1.89	77.97 ± 2.91	74.10 ± 5.07	72.83 ± 1.96
	FedTIE	**78.37** ± 3.32	**79.54** ± 2.27	**92.09** ± 1.79	**79.94** ± 3.96	**84.42** ± 1.53	**82.87** ± 1.27

Table 2: Results on MNLI with RoBERTa-base under varying client data heterogeneity. The data are partitioned by Dirichlet distributions with concentration parameter $\alpha \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$, where smaller α denotes stronger non-IID heterogeneity.

Method	MNLI $\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.3$	$\alpha = 0.4$	$\alpha = 0.5$
FedAvg	63.92	55.79	60.19	72.78	75.16
FFA-LoRA	67.05	67.09	59.32	62.32	66.61
FedSVD	**70.71**	70.25	71.22	72.79	74.94
FedASK	70.18	70.52	72.39	74.25	75.95
LA-LoRA	47.02	46.09	43.36	51.70	59.93
AS-LoRA	49.96	48.88	66.15	72.06	73.81
FedTIE	68.24	**77.58**	**76.17**	**77.18**	**78.93**

and 85.2% relative to FedASK. The advantage is more pronounced when comparing only main training: with the TIE rule and public reference seed cached, FedTIE transmits 92,160 parameters per round—reducing communication by 68.8% relative to FFA-LoRA and AS-LoRA and 90.4% relative to FedASK. We additionally evaluate the computational efficiency of FedTIE, with detailed results provided in Appendix E.3.

4.3 Ablation Study

Does the choice of subspace matter at fixed dimension? To isolate the effect of the proposed heterogeneity- and sensitivity-aware assignment, we replace TIE with a random uniform embedding while keeping the rest of the training pipeline unchanged. Experiments are conducted on MNLI with RoBERTa-base under DP-SGD with $\epsilon = 6$ and $\delta = 10^{-5}$. As shown in Fig. 3, the random embedding exhibits unstable optimization and reaches 0.5505 accuracy, whereas FedTIE reaches 0.7893, yielding a gain of 23.88%. With the same column normalization and nonempty-slot constraint, the random map is exactly isometric by Theorem 1; its occupancy and statistical loads need not be equal. The observed gap therefore concerns the choice of shared directions. Random grouping may

Table 3: Results with GPT-2 on the E2E NLG Challenge, comparing various federated LoRA methods at rank $r = 4$. Each client performs 3 local epochs per round.

Method	E2E NLG Challenge				
	BLEU ↑	NIST ↑	MET ↑	ROUGE-L ↑	CIDEr ↑
FedAvg	38.67	4.91	36.23	52.77	1.67
FFA-LoRA	39.83	5.25	36.93	55.75	1.79
FedSVD	40.14	5.27	37.95	55.01	1.81
FedASK	40.44	5.31	37.21	56.43	1.83
LA-LoRA	39.43	6.06	36.67	54.96	1.72
AS-LoRA	40.69	5.96	37.34	**56.79**	1.76
FedTIE	**41.80**	**6.33**	**38.37**	56.69	**1.93**

Table 4: Communication cost per round when using RoBERTa-base and LoRA with rank $r = 8$.

Method	Upload	Broadcast
FedAvg	294,912	294,912
FFA-LoRA	147,456	147,456
FedSVD	147,456	294,912
FedASK	442,368	516,096
LA-LoRA	294,912	294,912
AS-LoRA	147,456	147,456
FedTIE	**70,963**	**70,963**

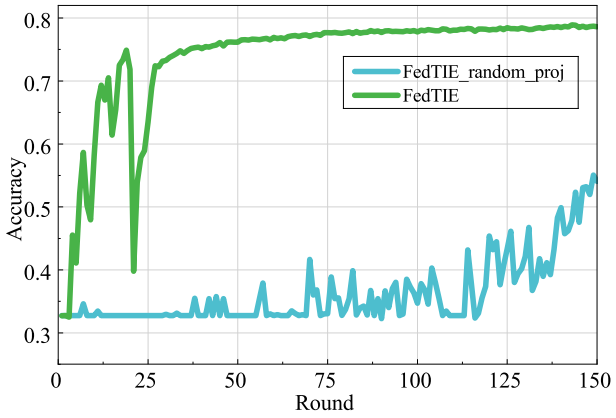

Figure 3: Ablation study comparing TIE with a random uniform embedding using RoBERTa-base on MNLI with $\epsilon = 6$ and $\delta = 10^{-5}$.

Table 5: Ablation of TIE assignment coefficients on MNLI with RoBERTa-base under DP-SGD ($\epsilon = 6$, $\delta = 10^{-5}$). Each row sets the indicated coefficient c to 0 or 10, keeping all others at 1. Default FedTIE sets all five coefficients to 1. Accuracy is reported in percent.

Coefficient c	Accuracy (%)	
	$c = 0$	$c = 10$
FedTIE-α	74.12	81.02
FedTIE-β	76.20	79.94
FedTIE-η	76.01	80.39
FedTIE-γ	75.93	79.81
FedTIE-λ	76.51	79.32
FedTIE (default)	78.93	

combine incompatible client profiles, whereas TIE uses private statistics to favor compatible sharing and balanced proxy loads. This comparison supports the assignment design in the evaluated setting.

Ablation on assignment coefficients. We set each TIE assignment coefficient to 0 or 10, keeping the others at 1 (Table 5). Removing any term reduces accuracy from 78.93% to 74.12–76.51%, supporting the contribution of all five terms in this setting. Increasing individual coefficients to 10 improves accuracy to 79.32–81.02%. The conflict coefficient α has the largest effect: removing its penalty costs 4.81 percentage points, while increasing it yields a 2.09-point gain. These results highlight the importance of compatible coordinate sharing and suggest that tuning the assignment weights can further improve utility.

5 Conclusion

We studied private federated LoRA through two coupled choices: private dimension and shared update directions. FedTIE uses privacy-accounted warm-up profiles to select cross-layer and cross-factor coordinate sharing, then trains privately in the fixed affine factor subspace. Its isometric embedding preserves lifted perturbation energy, while its assignment aims to retain useful adaptation directions. We establish end-to-end sample-level privacy and a finite-horizon convergence bound for the induced subspace objective. Experiments show stronger private GLUE utility, robustness across heterogeneous partitions, competitive non-private generation performance, and per-round communication reductions of 51.9% relative to FFA-LoRA and 85.2% relative to FedASK. These findings support jointly designing private dimension and trainable directions, with benefits depending on the task, client distribution, and privacy cost of space selection.

AI USE STATEMENT

Large language models (LLMs) were used to support manuscript preparation and language refinement, including grammar correction, sentence rewriting, improving clarity and readability, and verifying the consistency of mathematical notation and theoretical arguments. The research ideas, methodological design, experiments, and analyses were developed and conducted by the authors without relying on LLMs for their origination. All AI-assisted material was carefully examined and revised by the authors to ensure its correctness and consistency with the underlying research. The authors take full responsibility for the final manuscript.

REPRODUCIBILITY STATEMENT

To support reproducibility, we provide the FedTIE implementation in an anonymous repository at `https://anonymous.4open.science/r/fedhsip/`. Section 3 describes the Tied Isometric Embedding construction and the federated optimization procedure, while Appendix D provides pseudocode for private warm-up, embedding construction, and latent-space training. Appendix E documents the datasets and evaluation splits, baseline methods, model and LoRA configurations, client data partitioning, training hyperparameters, random seeds, generation settings, and computing hardware. The privacy model and end-to-end accounting are detailed in Appendix B, including the budget allocation and noise calibration protocol in Appendix B.5, which accounts for both warm-up and main-stage releases. For the theoretical results, Appendix A establishes the embedding properties, Appendix B proves the sample-level differential privacy guarantee, and Appendix C states the assumptions and provides proofs for the convergence results.

REFERENCES

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, 2016.

Borja Balle, Gilles Barthe, Marco Gaboardi, Justin Hsu, and Tetsuya Sato. Hypothesis testing interpretations and renyi differential privacy. In *International Conference on Artificial Intelligence and Statistics*, pp. 2496–2506. PMLR, 2020.

Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th annual symposium on foundations of computer science*, pp. 464–473. IEEE, 2014.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and trends® in theoretical computer science*, 9(3-4):211–487, 2014.

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

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

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, 2021. URL `https://arxiv.org/abs/2106.09685`.

Tianqu Kang, Zixin Wang, Hengtao He, Jun Zhang, Shenghui Song, and Khaled B Letaief. Federated low-rank adaptation with differential privacy over wireless networks. *arXiv preprint arXiv:2411.07806*, 2024.

Myoungjun Kim, Sangwoo Park, Yoseob Han, and Jin-Hyun Ahn. Adaptive selection of lora components in privacy-preserving federated learning. *arXiv preprint arXiv:2605.05769*, 2026.

Seanie Lee, Sangwoo Park, Dong Bok Lee, Dominik Wagner, Haebin Seong, Tobias Bocklet, Juho Lee, and Sung Ju Hwang. Fedsvd: Adaptive orthogonalization for private federated learning with lora. *arXiv preprint arXiv:2505.12805*, 2025.

Kaiyang Li, Shaobo Han, Qing Su, Wei Li, Zhipeng Cai, and Shihao Ji. Uni-lora: One vector is all you need. *arXiv preprint arXiv:2506.00799*, 2025.

Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE signal processing magazine*, 37(3):50–60, 2020.

Jin Liu, Yinbin Miao, Ning Xi, and Junkang Liu. Rethinking lora for privacy-preserving federated learning in large models. *arXiv preprint arXiv:2602.19926*, 2026.

Xiao-Yang Liu, Rongyi Zhu, Daochen Zha, Jiechao Gao, Shan Zhong, Matt White, and Meikang Qiu. Differentially private low-rank adaptation of large language model using federated learning. *ACM Transactions on Management Information Systems*, 16(2):1–24, 2025.

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

Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th computer security foundations symposium (CSF)*, pp. 263–275. IEEE, 2017.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. The e2e dataset: New challenges for end-to-end generation. In *Proceedings of the 18th annual SIGdial meeting on discourse and dialogue*, pp. 201–206, 2017.

Lakshay Sharma, Laura Graesser, Nikita Nangia, and Utku Evci. Natural language understanding with the quora question pairs dataset. *arXiv preprint arXiv:1907.01041*, 2019.

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

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Shuang Song, Kamalika Chaudhuri, Anand D Sarwate, et al. Stochastic gradient descent with differentially private updates. In *GlobalSIP*, pp. 245–248, 2013.

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

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel 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.

Yu-Xiang Wang, Borja Balle, and Shiva Prasad Kasiviswanathan. Subsampled rényi differential privacy and analytical moments accountant. In *The 22nd international conference on artificial intelligence and statistics*, pp. 1226–1235. PMLR, 2019.

Ming Wen, Jiaqi Zhu, Yuedong Xu, Yipeng Zhou, and Dingding Han. Differentially private federated low rank adaptation beyond fixed-matrix. *arXiv preprint arXiv:2507.09990*, 2025.

Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long papers)*, pp. 1112–1122, 2018.

Xuefei Yin, Yanming Zhu, and Jiankun Hu. A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions. *ACM Computing Surveys (CSUR)*, 54(6): 1–36, 2021.

Jianyi Zhang, Saeed Vahidian, Martin Kuo, Chunyuan Li, Ruiyi Zhang, Tong Yu, Guoyin Wang, and Yiran Chen. Towards building the federatedgpt: Federated instruction tuning. In *ICASSP 2024-2024 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 6915–6919. IEEE, 2024.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Guoyin Wang, et al. Instruction tuning for large language models: A survey. *ACM Computing Surveys*, 58(7):1–36, 2026.

Yingxue Zhou, Zhiwei Steven Wu, and Arindam Banerjee. Bypassing the ambient dimension: Private sgd with gradient subspace identification. *arXiv preprint arXiv:2007.03813*, 2020.

A Analysis of the Embedding Matrix

Following the design desiderata of Uni-LoRA (Li et al., 2025), we distinguish three properties of a sharing map. Their guarantees differ: the sparse normalized structure ensures exact isometry, whereas global sharing is allowed by the assignment domain and load balancing is encouraged by the objective.

- **Global sharing:** Coordinates from all adapted layers and both factor types are eligible for every slot. This permits cross-layer and cross-factor sharing, but does not require every slot to contain multiple layers or factor types.
- **Load balancing:** The assignment objective penalizes unequal occupancy and unequal statistical proxy loads. The hard feasibility condition is only $n_j \geq 1$; equal counts or statistical loads are not guaranteed by one-hot structure or isometry.
- **Isometry:** The map from latent increments to the selected affine factor subspace preserves Euclidean distances exactly. This says nothing about preservation of full-space directions excluded by that subspace or about distances after the bilinear map to BA.

Theorem 1 applies to the statistics-driven, signed TIE assignment, as well as to any random baseline satisfying the same nonempty-column and normalization conditions.

Theorem 1. *Let $1 \leq d \leq D$, let $\pi : \{1, \ldots, D\} \to \{1, \ldots, d\}$ be any surjective assignment, and let $\sigma_i \in \{-1, +1\}$. Write $n_j = |\{i : \pi(i) = j\}| \geq 1$ and define $P_{ij} = \sigma_i / \sqrt{n_j}$ when $\pi(i) = j$, and zero otherwise. The assignment and signs may depend on privatized warm-up statistics; no randomness or load balance is needed. Then $P^\top P = I_d$ and, for any fixed reference $\theta_{D,\mathrm{ref}}$, the affine map $F(x) = \theta_{D,\mathrm{ref}} + Px$ is exactly isometric:*

$$\|F(x) - F(y)\|_2 = \|P(x - y)\|_2 = \|x - y\|_2, \quad \forall x, y \in \mathbb{R}^d.$$

Proof. Given the construction of P in the theorem statement, we begin by showing that $P^\top P = I_d$. Consider the (j, k)-th entry of $P^\top P$:

$$[P^\top P]_{j,k} = \sum_{i=1}^{D} P_{i,j} P_{i,k}.$$

Case 1: $j \neq k$. Since each row contains only one nonzero entry, there exists no row i such that both $P_{i,j}$ and $P_{i,k}$ are nonzero. Hence, every term in the summation is zero, and we have

$$[P^\top P]_{j,k} = 0.$$

Case 2: $j = k$. Column j contains n_j nonzero entries, each with squared value $\sigma_i^2 / n_j = 1/n_j$. Thus,

$$[P^\top P]_{j,j} = \sum_{i=1}^{D} P_{i,j}^2 = n_j \cdot \left(\frac{1}{\sqrt{n_j}} \right)^2 = 1.$$

From the analysis above, we have $[P^\top P]_{j,k} = 1$ if $j = k$ and 0 otherwise, thus $P^\top P = I_d$. Therefore,

$$\|P(x - y)\|^2 = (x - y)^\top P^\top P (x - y) = (x - y)^\top (x - y) = \|x - y\|^2, \quad \forall x, y \in \mathbb{R}^d.$$

This completes the proof. □

Scope of the guarantee. The fixed offset does not change $g_d = P^\top g_D$. For $P^\top P = I_d$, write $\Pi = PP^\top$. A latent perturbation $\xi \sim \mathcal{N}(0, v^2 I_d)$ satisfies

$$\mathrm{Cov}(P\xi) = v^2 \Pi, \qquad \mathbb{E}\|P\xi\|_2^2 = v^2 d. \tag{10}$$

This is a statement in the concatenated factor space: the lifted noise is supported on $\mathrm{range}(P)$, not isotropic in $\mathbb{R}^D$. Eq. 3 separates discarded signal, projected clipping bias, and lifted Gaussian noise energy in this same space. To verify the identity, decompose $P(a_d + \xi) - \bar{g}_D$ into its orthogonal components in $\mathrm{range}(P)$ and its orthogonal complement, use $P^\top P = I_d$, and take expectation over the zero-mean Gaussian perturbation.

Isometry preserves distances within the selected affine factor subspace, not arbitrary full-space directions or distances between effective weights BA and predictions. Even the coordinate-space noise comparison with a q_B-dimensional baseline requires $v_{\mathrm{TIE}}^2 d < v_{\mathrm{base}}^2 q_B$; $d < q_B$ alone is insufficient when warm-up consumes part of a fixed total privacy budget. Comparisons in effective-weight or prediction space additionally depend on each method's mapping and current factor scales. Neither lower task error nor balanced statistical loads follows from isometry.

B Proof of Privacy Guarantee of FedTIE

This section establishes an end-to-end sample-level privacy guarantee for FedTIE. The guarantee covers both data-dependent stages of the method: Stage I, which learns the TIE embedding from warm-up statistics, and Stage II, which trains the low-dimensional model under the frozen embedding. In particular, the embedding is not regarded as public side information: its privacy cost is charged to Stage I.

B.1 Privacy model and accounting notation

Let $\mathcal{D} = (\mathcal{D}_1, \ldots, \mathcal{D}_K)$ denote the collection of disjoint client datasets. We use add/remove sample-level adjacency: $\mathcal{D} \simeq \mathcal{D}'$ if the two collections differ by one example in one client dataset. The protected output is the complete observable transcript, including all warm-up messages, the learned assignment and signs (π, σ), the embedding P, all Stage II uploads, and the final model. Thus, privacy of the transcript implies privacy of any subset of these objects by post-processing. Client selection, batch sampling, clipping thresholds, and noise multipliers are required to be independent of the unprivatized data; they may instead be fixed in advance or chosen as functions of previous differentially private outputs. The public reference seed, positive normalization constants $b_{0,\mathrm{w}}, b_{0,\mathrm{m}}$, and aggregation weights follow the same restriction. In particular, normalization never uses the realized batch size or an unprotected private dataset size. If a denominator is described as an expected batch size, it is computed from a declared sampling rate and a public reference size that is identical on neighboring datasets; otherwise it is simply a fixed scaling constant. The two stages may use different constants, and client- or step-specific constants may be predeclared under this rule. For an empty sampled batch, the sum is zero and the mechanism still adds noise and performs its scheduled update; no raw batch count is released.

For $\alpha > 1$, let

$$\rho_{\mathrm{SGM}}(\alpha; q, \nu) \tag{11}$$

denote any valid order-α Rényi differential privacy (RDP) upper bound for a Poisson-sampled Gaussian mechanism with record sampling probability q and noise multiplier ν. We evaluate this quantity with the analytical sampled-Gaussian accountant rather than a small-q asymptotic approximation (Mironov, 2017; Wang et al., 2019). When $q = 1$, the Gaussian-mechanism value is $\rho_{\mathrm{SGM}}(\alpha; 1, \nu) = \alpha/(2\nu^2)$. If batches are sampled without replacement, the corresponding valid without-replacement accountant is used in place of Eq. 11. The sampling scheme used by the accountant must match the implementation.

We use two standard RDP facts. First, adaptive sequential composition adds RDP costs at every fixed order α. Second, if a mechanism is $(\alpha, \rho(\alpha))$-RDP for every order in the accountant's order set, then, for any $0 < \delta < 1$, it is

$$\left(\min_{\alpha > 1} \left\{ \rho(\alpha) + \frac{\log(1/\delta)}{\alpha - 1} \right\}, \delta \right) \text{-DP}. \tag{12}$$

The tighter RDP-to-DP conversion in Balle et al. (2020) can be used in the numerical accountant without changing any argument below.

B.2 Stage I: a private warm-up mechanism

The raw coordinate-wise quantities used by TIE are data-dependent and must not be released directly. We therefore implement every warm-up communication as a sampled Gaussian mechanism. The construction below makes explicit how the update-pattern, sensitivity, energy, and heterogeneity statistics are covered by one joint mechanism.

Fix warm-up round r, local step ℓ, and a participating client k. Conditional on the transcript preceding this step, let

$$\Phi_{k,r,\ell}(z) = \begin{bmatrix} a_g \psi_g(z) \\ a_u \psi_u(z) \\ a_s \psi_s(z) \\ a_e \psi_e(z) \\ a_h \psi_h(z) \end{bmatrix} \in \mathbb{R}^{m_{\mathrm{w}}} \tag{13}$$

be the per-example contribution to the complete warm-up message. Here, ψ_g supplies the provisional optimization update, while $\psi_u, \psi_s, \psi_e, \psi_h$ supply the first moments, sensitivity proxies, second moments, and client-variation moments from which $\{u_i, s_i, e_i, h_i\}_{i=1}^{D}$ are formed. For example, these blocks may contain signed gradients, squared gradients, and clipping indicators. The weights a_g, a_u, a_s, a_e, a_h are public positive normalization constants. The proof does not require a particular choice of these feature maps because their joint sensitivity is bounded by clipping the concatenated vector:

$$\overline{\Phi}_{k,r,\ell}(z) = \frac{\Phi_{k,r,\ell}(z)}{\max\{1, \|\Phi_{k,r,\ell}(z)\|_2 / C_{\mathrm{w}}\}}. \tag{14}$$

After Poisson sampling a minibatch $B^{\mathrm{w}}_{k,r,\ell}$ at rate $q^{\mathrm{w}}_{k,r,\ell}$, the only data-dependent value released at this step is

$$Y^{\mathrm{w}}_{k,r,\ell} = \sum_{z \in B^{\mathrm{w}}_{k,r,\ell}} \overline{\Phi}_{k,r,\ell}(z) + \mathcal{N}\big(0, (\nu^{\mathrm{w}}_{k,r,\ell} C_{\mathrm{w}})^2 I_{m_{\mathrm{w}}}\big)\,. \tag{15}$$

Division by the predeclared positive constant $b_{0,\mathrm{w}}$, extraction of the individual blocks, inversion of their public normalization constants, and the provisional optimizer update are post-processing of Eq. 15. If an implementation releases these blocks through separate Gaussian mechanisms instead, every such release is entered separately in the privacy ledger; concatenation as in Eq. 13 avoids an unaccounted within-step composition.

The server constructs the coordinate profiles only from the privatized collection

$$\mathcal{Y}_{\mathrm{w}} = \{Y^{\mathrm{w}}_{k,r,\ell} : r \le R_{\mathrm{w}},\ k \in \mathcal{S}_r,\ \ell \le \tau^{\mathrm{w}}_{k,r}\}. \tag{16}$$

In particular, the standardized client-update profiles u_i, sensitivity loads s_i, energies e_i, and cross-client heterogeneities h_i are deterministic functions of noisy first and second moments in $\mathcal{Y}_{\mathrm{w}}$. The balanced assignment, optional sign alignment, and embedding construction can consequently be written as

$$(\pi, \sigma, P) = \mathcal{B}(\mathcal{Y}_{\mathrm{w}}), \tag{17}$$

where $\mathcal{B}$ is post-processing and has no additional privacy cost. Warm-up starts from the public reference factors, and every provisional update uses the private optimization block. Provisional model aggregation is also post-processing. At the transition to Stage II, these learned parameters are discarded and the latent increments are initialized to zero around the public reference. Thus initialization discloses no additional data-dependent vector. Public, data-independent sketches may be applied before or after the Gaussian mechanism as long as clipping controls the sensitivity of the vector actually queried. Secure aggregation and quantization can reduce exposure and communication, respectively, but neither is used as a substitute for the clipping, noise, and accounting in Eq. 15.

Lemma 1 (Privacy of the warm-up transcript)**.** *For every $\alpha > 1$, the complete Stage I transcript and the resulting TIE embedding satisfy $(\alpha, \rho_{\mathrm{w}}(\alpha))$-RDP, where*

$$\rho_{\mathrm{w}}(\alpha) = \sum_{r=1}^{R_{\mathrm{w}}} \max_{k \in \mathcal{S}_r} \sum_{\ell=1}^{\tau^{\mathrm{w}}_{k,r}} \rho_{\mathrm{SGM}}\big(\alpha; q^{\mathrm{w}}_{k,r,\ell}, \nu^{\mathrm{w}}_{k,r,\ell}\big)\,. \tag{18}$$

For a homogeneous schedule with τ_{w} local steps per round, sampling rate q_{w}, and noise multiplier ν_{w}, this reduces to

$$\rho_{\mathrm{w}}(\alpha) = R_{\mathrm{w}} \tau_{\mathrm{w}} \rho_{\mathrm{SGM}}(\alpha; q_{\mathrm{w}}, \nu_{\mathrm{w}}). \tag{19}$$

Proof. Under add/remove adjacency, the sum in Eq. 15 has ℓ_2-sensitivity at most C_{w} because of Eq. 14. Hence one local step has RDP cost $\rho_{\mathrm{SGM}}(\alpha; q^{\mathrm{w}}_{k,r,\ell}, \nu^{\mathrm{w}}_{k,r,\ell})$. Within one round, client datasets are disjoint, so parallel composition takes the maximum over participating clients. The local steps

and warm-up rounds may be adaptive, but adaptive RDP composition adds their costs, which gives Eq. 18. Finally, $(u, s, e, h, \pi, \sigma, P)$ are functions of the privatized transcript and inherit the same guarantee by post-processing. □

For replace-one adjacency, the sensitivity of the sum is at most $2C_{\mathrm{w}}$; equivalently, one may double the Gaussian standard deviation in Eq. 15 or use an accountant parameterized for replace-one adjacency.

B.3 Stage II: private training under the learned embedding

We now condition on an arbitrary fixed realization of the Stage I transcript. This fixes P, even though P is data-dependent before conditioning. In main-training round t, local step ℓ, and client k, the projected per-example gradient is

$$g_{d,k,t,\ell}(z) = P^{\top} g_{D,k,t,\ell}(z), \qquad \overline{g}_{d,k,t,\ell}(z) = \frac{g_{d,k,t,\ell}(z)}{\max\{1, \|g_{d,k,t,\ell}(z)\|_2 / C_{\mathrm{m}}\}}. \tag{20}$$

For a Poisson-sampled minibatch $B^{\mathrm{m}}_{k,t,\ell}$, the client uses

$$Y^{\mathrm{m}}_{k,t,\ell} = \sum_{z \in B^{\mathrm{m}}_{k,t,\ell}} \overline{g}_{d,k,t,\ell}(z) + \mathcal{N}\big(0, (\nu^{\mathrm{m}}_{k,t,\ell} C_{\mathrm{m}})^2 I_d\big) \tag{21}$$

and divides by the predeclared positive constant $b_{0,\mathrm{m}}$ before applying the optimizer, exactly as in Eq. 9. The normalized noise covariance is therefore $(\nu^{\mathrm{m}}_{k,t,\ell} C_{\mathrm{m}} / b_{0,\mathrm{m}})^2 I_d$. Only the final local update need be uploaded; because it is a function of the sequence of private gradients in Eq. 21, accounting for that sequence is sufficient. Server aggregation, reconstruction $\theta_D = \theta_{D,\mathrm{ref}} + P\theta_d$, and reshaping into LoRA factors are post-processing, using the public reference and the declared aggregation weights.

Lemma 2 (Privacy of the main-training transcript)**.** *Uniformly for every fixed warm-up transcript and its induced embedding, the complete Stage II transcript satisfies $(\alpha, \rho_{\mathrm{m}}(\alpha))$-RDP, where*

$$\rho_{\mathrm{m}}(\alpha) = \sum_{t=1}^{R_{\mathrm{m}}} \max_{k \in \mathcal{S}_t} \sum_{\ell=1}^{\tau^{\mathrm{m}}_{k,t}} \rho_{\mathrm{SGM}}\big(\alpha; q^{\mathrm{m}}_{k,t,\ell}, \nu^{\mathrm{m}}_{k,t,\ell}\big)\,. \tag{22}$$

For a homogeneous schedule,

$$\rho_{\mathrm{m}}(\alpha) = R_{\mathrm{m}} \tau_{\mathrm{m}} \rho_{\mathrm{SGM}}(\alpha; q_{\mathrm{m}}, \nu_{\mathrm{m}}). \tag{23}$$

Proof. For every fixed P and previous private transcript, Eq. 20 bounds the add/remove sensitivity of the summed gradient by C_{m}, regardless of the dimension or operator norm of P. Each local step therefore incurs the sampled-Gaussian RDP cost shown in Eq. 22. Parallel composition over disjoint client datasets, adaptive composition over local steps and rounds, and post-processing of the private gradients yield the result. The bound is uniform in P, which is the condition needed to compose this stage adaptively after Stage I. □

As in Stage I, C_{m} must be replaced by $2C_{\mathrm{m}}$ under replace-one adjacency unless the chosen accountant already incorporates that convention.

B.4 End-to-end guarantee

Let $\mathcal{M}_{\mathrm{w}}$ denote the mechanism producing the warm-up transcript and let $\mathcal{B}$ construct the embedding. Let $\mathcal{M}_{\mathrm{m}}(\cdot; P)$ denote main training under that embedding. The released algorithm is the adaptive composition

$$\mathcal{M}(\mathcal{D}) = \big(\mathcal{M}_{\mathrm{w}}(\mathcal{D}), \mathcal{B}(\mathcal{M}_{\mathrm{w}}(\mathcal{D})), \mathcal{M}_{\mathrm{m}}\big(\mathcal{D}; \mathcal{B}(\mathcal{M}_{\mathrm{w}}(\mathcal{D}))\big)\big)\,. \tag{24}$$

Theorem 2 (End-to-end privacy of FedTIE)**.** *Suppose all Stage I releases follow Eq. 15, all Stage II releases follow Eq. 21, and no other data-dependent value is released. Then, for every $\alpha > 1$, the complete FedTIE transcript is*

$$\big(\alpha, \rho_{\mathrm{tot}}(\alpha)\big)\text{-RDP}, \qquad \rho_{\mathrm{tot}}(\alpha) = \rho_{\mathrm{w}}(\alpha) + \rho_{\mathrm{m}}(\alpha). \tag{25}$$

Consequently, for every $0 < \delta < 1$*, FedTIE is* $(\varepsilon_{\mathrm{tot}}, \delta)$*-DP with*

$$\varepsilon_{\mathrm{tot}} = \min_{\alpha>1} \left\{ \rho_{\mathrm{w}}(\alpha) + \rho_{\mathrm{m}}(\alpha) + \frac{\log(1/\delta)}{\alpha - 1} \right\}. \tag{26}$$

Proof. Lemma 1 gives the first-stage RDP cost. Conditional on every possible first-stage output, Lemma 2 gives a uniform RDP guarantee for the second stage. Adaptive RDP composition therefore adds the two costs. Eq. 26 follows from Eq. 12. Because the theorem protects the entire transcript, repeatedly using the frozen embedding during Stage II incurs no cost beyond $\rho_{\mathrm{w}} + \rho_{\mathrm{m}}$: the embedding and all of its later uses are post-processing of already-accounted releases. □

The theorem deliberately does not rely on client-subsampling amplification, because the identities of participating clients may be visible to the server. If an implementation hides the sampled client set and satisfies the assumptions of a valid hierarchical-subsampling theorem, the corresponding amplified RDP bound may replace the conservative per-round maxima in Eq. 18 and Eq. 22. Such amplification is optional and is not needed for Theorem 2.

B.5 Budget allocation and accounting protocol

For a target (ε, δ), our accounting ledger contains every sampled Gaussian event from both stages. The tightest reported guarantee is obtained by accumulating $\rho_{\mathrm{w}}(\alpha)$ and $\rho_{\mathrm{m}}(\alpha)$ first and applying Eq. 26 once. For transparent stage-wise calibration, one may equivalently pre-specify

$$\varepsilon_{\mathrm{w}} + \varepsilon_{\mathrm{m}} \le \varepsilon, \qquad \delta_{\mathrm{w}} + \delta_{\mathrm{m}} \le \delta, \tag{27}$$

and independently choose ν_{w} and ν_{m} so that

$$\min_{\alpha>1} \left\{ \rho_{\mathrm{w}}(\alpha) + \frac{\log(1/\delta_{\mathrm{w}})}{\alpha - 1} \right\} \le \varepsilon_{\mathrm{w}}, \tag{28}$$

$$\min_{\alpha>1} \left\{ \rho_{\mathrm{m}}(\alpha) + \frac{\log(1/\delta_{\mathrm{m}})}{\alpha - 1} \right\} \le \varepsilon_{\mathrm{m}}. \tag{29}$$

Basic adaptive composition then gives the target end-to-end guarantee, while the joint calculation in Eq. 26 can only improve the reported bound.

We use the following deterministic allocation rule so that changing the number of warm-up rounds cannot silently change the total privacy guarantee. Let

$$N_{\mathrm{w}} = \sum_{r=1}^{R_{\mathrm{w}}} \tau_r^{\mathrm{w}}, \qquad N_{\mathrm{m}} = \sum_{t=1}^{R_{\mathrm{m}}} \tau_t^{\mathrm{m}}, \qquad \lambda = \frac{N_{\mathrm{w}}}{N_{\mathrm{w}} + N_{\mathrm{m}}}, \tag{30}$$

where the maxima over client-specific local-step counts are used if clients have different schedules. Before training, we set

$$(\varepsilon_{\mathrm{w}}, \delta_{\mathrm{w}}) = (\lambda\varepsilon, \lambda\delta), \qquad (\varepsilon_{\mathrm{m}}, \delta_{\mathrm{m}}) = ((1-\lambda)\varepsilon, (1-\lambda)\delta), \tag{31}$$

and calibrate the two noise multipliers with Eq. 29. With the same number of local steps per federated round, the default $R_{\mathrm{w}} = 20$ and $R_{\mathrm{m}} = 130$ configuration uses

$$\lambda = \frac{20}{20 + 130} = \frac{2}{15}, \qquad (\varepsilon_{\mathrm{w}}, \varepsilon_{\mathrm{m}}) = \left(\frac{2}{15}\varepsilon, \frac{13}{15}\varepsilon \right), \tag{32}$$

with the same fractions applied to δ. The warm-up ablations use

R_{w}	λ	$(\varepsilon_{\mathrm{w}}, \varepsilon_{\mathrm{m}})$
15	3/29	$(3\varepsilon/29,\ 26\varepsilon/29)$
10	1/14	$(\varepsilon/14,\ 13\varepsilon/14)$

(33)

for $R_{\mathrm{m}} = 130$. Thus, each choice of $(R_{\mathrm{w}}, R_{\mathrm{m}})$ is recalibrated against the same total (ε, δ); neither stage is treated as privacy-free.

For example, under the default 20/130 schedule, a total budget $\varepsilon = 3$ gives $(\varepsilon_{\mathrm{w}}, \varepsilon_{\mathrm{m}}) = (0.4, 2.6)$, while $\varepsilon = 6$ gives $(0.8, 5.2)$. When $\delta = 10^{-5}$, the corresponding split is $(\delta_{\mathrm{w}}, \delta_{\mathrm{m}}) = (2 \times 10^{-5}/15, 13 \times 10^{-5}/15)$.

The proportional rule in Eq. 31 is a declared experimental convention, not a requirement of Theorem 2. Other pre-specified allocations are valid, and unequal sample rates or local schedules may motivate a different split, provided that all Stage I and Stage II events remain in the same ledger and Eq. 26 does not exceed the claimed target. For a fair comparison, every private method is evaluated under the same adjacency relation, total (ε, δ), sampling convention, and accountant; FedTIE's reported budget always includes the warm-up transcript and the data-dependent embedding.

C Convergence Analysis of FedTIE

We analyze the main-stage SGD updates of Algorithm 2, including per-example clipping, Poisson record sampling, Gaussian noise, multiple local steps, and partial client participation. The principal result is a finite-horizon bound on the expected squared gradient of the objective restricted to the learned affine factor subspace. We subsequently distinguish this guarantee from stationarity in the full factor space and give an additional result under a Polyak–Łojasiewicz (PL) condition.

The analysis uses a smooth nonconvex descent argument, with the perturbations of the actual latent update accounted for explicitly. This is related in spirit to the error decomposition in FedRot-LoRA (Zhang et al., 2026), but our state variable and error sources are different. The geometric convergence result for LA-LoRA (Liu et al., 2026) concerns a specialized noiseless problem with restricted-isometry and related structural conditions. Those conditions are not assumed for the neural-network loss studied here, and that result is not used as a convergence guarantee for FedTIE.

C.1 Objective, conditioning, and the exact update

Conditioning on private warm-up. Fix the client datasets and condition on the complete Stage I transcript, the public reference seed, and any randomness used to construct the assignment. The matrix P is then fixed and satisfies $P^\top P = I_d$ by Theorem 1. All expectations below are over fresh main-stage randomness, conditional on this information, unless stated otherwise. This conditioning is an optimization device: it neither makes warm-up privacy-free nor requires fresh datasets for Stage II. In particular, the same training records may be reused. The final unconditional result follows by averaging over warm-up realizations.

Reindex the main rounds by $t = 0, \dots, T-1$, where $T = R - R_w = R_{\mathrm{m}}$, and write

$$x_t = \theta_d^{R_w+t}, \qquad x_{k,t,0} = x_t, \qquad \Theta(x) = \theta_{D,\mathrm{ref}} + Px. \tag{34}$$

Thus $x_0 = 0$, and $\Theta(x_0)$ is the public reference, not the terminal warm-up model. The symbol Θ denotes the concatenated LoRA factors, not the effective weight matrices BA. All vector norms and inner products in this appendix are Euclidean; matrix norms are specified when needed.

The objective induced by the declared normalization. Let $N_k = |\mathcal{D}_k| > 0$ and let $\ell(\Theta; z)$ be the loss for record z with the backbone frozen. If dropout or another per-example randomization is used, ℓ denotes its expectation, with differentiation and expectation assumed interchangeable. For the main theorem, take client-specific record sampling probabilities $q_k \in (0, 1]$ and predeclared denominators $b_k > 0$ that are constant during Stage II. The manuscript's common-denominator setting is $b_k = b_{0,\mathrm{m}}$. Define

$$\begin{aligned}
\mathcal{L}_k(\Theta) &= \frac{1}{N_k} \sum_{z \in \mathcal{D}_k} \ell(\Theta; z), & a_k &= \frac{q_k N_k}{b_k}, \\
\mathcal{F}_k(\Theta) &= a_k \mathcal{L}_k(\Theta), & f_k(x) &= \mathcal{F}_k(\Theta(x)), \\
\mathcal{F}(\Theta) &= \sum_{k=1}^{K} \omega_k \mathcal{F}_k(\Theta), & F(x) &= \mathcal{F}(\Theta(x)) = \sum_{k=1}^{K} \omega_k f_k(x),
\end{aligned} \tag{35}$$

where $\omega_k \geq 0$ and $\sum_k \omega_k = 1$ are the fixed target client weights. We may omit clients with zero target weight. The factors a_k are necessary because the expectation of an unclipped, normalized Poisson sum is

$$\mathbb{E}\left[\frac{1}{b_k} \sum_{z \in \mathcal{B}_k} P^\top g_D(\Theta(x); z)\right] = \frac{q_k}{b_k} \sum_{z \in \mathcal{D}_k} P^\top \nabla_\Theta \ell(\Theta(x); z) = \nabla f_k(x). \tag{36}$$

When $a_k = 1$ for all relevant clients, the objective is $F = \sum_k \omega_k \mathcal{L}_k \circ \Theta$. If all a_k equal the same positive constant, the stationary points are unchanged but the objective and smoothness constants are rescaled. Unequal a_k change the relative client weights of the optimized loss. The appearance of N_k in this analysis does not authorize its release or its use as an unprotected algorithmic denominator. The actual b_k must still satisfy the public-reference-size convention of Appendix B.

Local oracle and server aggregation. For readability, put $C = C_{\mathrm{m}}$ and allow client-specific $\nu_k = \nu_k^{\mathrm{m}}$. At local step $s = 0, \ldots, \tau - 1$, set

$$\begin{aligned} g_k(x; z) &= P^\top g_D(\Theta(x); z), & c_k(x; z) &= \mathrm{clip}_C(g_k(x; z)), \\ \widehat{g}_{k,t,s} &= \frac{1}{b_k} \sum_{z \in \mathcal{B}_{k,t,s}} c_k(x_{k,t,s}; z) + \xi_{k,t,s}, & \xi_{k,t,s} &\sim \mathcal{N}(0, v_k^2 I_d), \\ v_k &= \frac{\nu_k C}{b_k}, & x_{k,t,s+1} &= x_{k,t,s} - \eta \widehat{g}_{k,t,s}. \end{aligned} \tag{37}$$

The notation c_k includes fresh per-example randomness when present. The Bernoulli record-inclusion variables, per-example random seeds, and Gaussian draws are fresh and independent across clients and local steps, conditional on the preceding history. Empty batches contribute a zero sum and still receive noise and execute an update.

Let $p_{k,t} \geq 0$ be the aggregation weights, zero for nonparticipants, with $\sum_k p_{k,t} = 1$. They are chosen before the current round's local randomness and remain fixed throughout the round. Define the effective round step size $\gamma = \eta\tau$. The exact server recursion is

$$x_{t+1} = \sum_k p_{k,t} x_{k,t,\tau} = x_t - \gamma H_t, \qquad H_t = \frac{1}{\tau} \sum_{s=0}^{\tau-1} \sum_k p_{k,t} \widehat{g}_{k,t,s}. \tag{38}$$

This identity uses the common initialization of the participating clients and equal local-step counts. It is exactly the SGD algorithm specified in the manuscript. Momentum, AdamW, additional trainable parameters outside x, unequal local-step counts, or changing P require a corresponding modification of the analysis.

C.2 Embedding geometry and local estimation error

Lemma 3 (Geometry of the restricted objective)**.** *For $\Pi = PP^\top$, the gradients satisfy*

$$\nabla f_k(x) = P^\top \nabla \mathcal{F}_k(\Theta(x)), \qquad \nabla F(x) = P^\top \nabla \mathcal{F}(\Theta(x)). \tag{39}$$

If $\mathcal{F}_k$ has L_D-Lipschitz gradient on the affine subspace, then f_k has L_D-Lipschitz gradient. If it is twice differentiable,

$$\nabla^2 f_k(x) = P^\top \nabla^2 \mathcal{F}_k(\Theta(x)) P. \tag{40}$$

Furthermore, at the same factor state $\Theta(x)$,

$$\sum_k \omega_k \|\nabla f_k(x) - \nabla F(x)\|^2 \leq \sum_k \omega_k \|\nabla \mathcal{F}_k(\Theta(x)) - \nabla \mathcal{F}(\Theta(x))\|^2, \tag{41}$$

For $\xi \sim \mathcal{N}(0, v^2 I_d)$,

$$\mathbb{E}\|P\xi\|^2 = dv^2, \qquad \mathrm{Cov}(P\xi) = v^2 \Pi. \tag{42}$$

Proof. The chain rule gives the gradient and Hessian identities. Since $\|P\|_2 = \|P^\top\|_2 = 1$,

$$\|\nabla f_k(x) - \nabla f_k(y)\| \leq L_D \|P(x - y)\| = L_D \|x - y\|.$$

Applying the same contraction to each client-gradient difference gives Eq. 41. Finally, $\|P\xi\|^2 = \|\xi\|^2$ and $\mathrm{tr}(\Pi) = d$, proving the noise statements. □

The smoothness assumption here concerns the loss as a function of the *concatenated factors*. Smoothness with respect to effective weights alone does not imply it with the same constant: the map from factors to BA is bilinear. Isometry does not make the neural-network loss linear or convex. In the theorem below, a uniform smoothness constant is an explicit analytical assumption, not

a consequence of clipping. In particular, Gaussian noise is unbounded, so a smoothness estimate on an arbitrary bounded neighborhood does not by itself justify a global expectation bound for the unmodified algorithm.

Define the conditional mean of the normalized clipped oracle and its bias by

$$h_k(x) = \frac{q_k}{b_k} \sum_{z \in \mathcal{D}_k} \mathbb{E}[c_k(x; z)], \qquad e_k(x) = h_k(x) - \nabla f_k(x). \tag{43}$$

The expectation in this definition is only over per-example randomization. Clipping a stochastic gradient generally produces a biased estimate of the gradient of the original objective.

Lemma 4 (Bias and variance of a private Poisson step)**.** *Conditional on the history immediately before local step (k, t, s), the oracle in Eq. 37 admits the decomposition*

$$\widehat{g}_{k,t,s} = \nabla f_k(x_{k,t,s}) + e_k(x_{k,t,s}) + u_{k,t,s}, \qquad \mathbb{E}[u_{k,t,s} \mid \textit{preceding history}] = 0. \tag{44}$$

For $\mu_{k,z}(x) = \mathbb{E}[c_k(x; z)]$ and $Q_{k,z}(x) = \mathbb{E}\|c_k(x; z)\|^2$, its conditional variance is

$$V_k(x) = \frac{1}{b_k^2} \sum_{z \in \mathcal{D}_k} \left(q_k Q_{k,z}(x) - q_k^2 \|\mu_{k,z}(x)\|^2 \right) + d \left(\frac{\nu_k C}{b_k} \right)^2. \tag{45}$$

In particular, one may take the uniform bound

$$V_k(x) \le V_k := S_k + d v_k^2, \qquad S_k = \frac{q_k N_k C^2}{b_k^2}. \tag{46}$$

If the per-example gradients are deterministic at a fixed model state, the sharper choice $S_k = q_k(1 - q_k)N_k C^2/b_k^2$ is valid. Also,

$$\|e_k(x)\| \le a_k \mathbb{E}_{z,k}\big[(\|g_k(x; z)\| - C)_+\big], \qquad (r)_+ = \max\{r, 0\}, \tag{47}$$

where $\mathbb{E}_{z,k}$ averages uniformly over client k's records and over their per-example randomization.

Proof. Write each Poisson inclusion indicator as $I_z \sim \text{Bernoulli}(q_k)$. At a fixed current state, independence gives $\mathbb{E}[I_z c_k] = q_k \mu_{k,z}$ and

$$\mathbb{E}\|I_z c_k - q_k \mu_{k,z}\|^2 = q_k Q_{k,z} - q_k^2 \|\mu_{k,z}\|^2.$$

Centered contributions from distinct records are orthogonal in expectation. Adding the independent Gaussian variance $d v_k^2$ proves Eq. 45. The bounds follow from $\|c_k\| \le C$. With deterministic gradients, $Q_{k,z} = \|\mu_{k,z}\|^2$, giving the factor $q_k(1 - q_k)$. Finally,

$$\|\operatorname{clip}_C(g) - g\| = (\|g\| - C)_+, \qquad e_k(x) = a_k \mathbb{E}_{z,k}[\operatorname{clip}_C(g_k(x; z)) - g_k(x; z)].$$

The triangle inequality proves Eq. 47. □

These statements also hold at noisy, history-dependent local iterates because they are conditioned on the state *before* the new draw. They do not assert that a complete multi-step client update equals a noise-free update plus an independent Gaussian: previous noise changes subsequent gradients and clipping decisions.

If there are per-example random effects shared across records, the independence argument for Eq. 45 must be replaced by a bound including their covariances. The subsequent theorem continues to hold with any valid uniform oracle-variance bounds V_k.

C.3 Assumptions and participation constants

Let $\mathcal{T}_t$ contain the fixed warm-up information and all main-stage history before selecting participants for round t, so x_t is $\mathcal{T}_t$-measurable. We use the following conditions.

1. **Smoothness and lower boundedness.** Each f_k has L_P-Lipschitz gradient on $\mathbb{R}^d$, for a common finite $L_P > 0$, and $F_* := \inf_x F(x) > -\infty$. A sufficient condition for smoothness is the one in Lemma 3. No convexity is required.

2. **Bounded clipping bias and projected heterogeneity.** There are finite $B_k \geq 0$ and $\zeta_P \geq 0$ such that, for every x,

$$\|e_k(x)\| \leq B_k, \qquad \sum_k \omega_k \|\nabla f_k(x) - \nabla F(x)\|^2 \leq \zeta_P^2. \tag{48}$$

Put $B_P^2 = \sum_k \omega_k B_k^2$. These are assumptions about the actual gradients, not about the warm-up proxies s_i, e_i, h_i. The bias condition can be verified from Eq. 47 when a uniform tail-moment bound is available. It is not implied by the boundedness of clipped gradients.

3. **Unbiased participation for a fixed objective.** The weights are selected before local training and satisfy

$$\mathbb{E}[p_{k,t} \mid \mathcal{T}_t] = \omega_k, \qquad \mathbb{E}\left[\left\|\sum_k (p_{k,t} - \omega_k)\nabla f_k(x_t)\right\|^2 \,\middle|\, \mathcal{T}_t\right] \leq \chi_P^2. \tag{49}$$

The sampling law may depend on past information as long as these conditional conditions hold. Full participation with $p_{k,t} = \omega_k$ is allowed and gives $\chi_P = 0$.

4. **Fresh oracle randomness.** The conditional centering and independence in Lemma 4 hold, with finite uniform bounds V_k. Define

$$V_{\mathrm{loc}} = \sum_k \omega_k V_k, \qquad \mathbb{E}\left[\sum_k p_{k,t}^2 V_k \,\middle|\, \mathcal{T}_t\right] \leq V_{\mathrm{agg}}. \tag{50}$$

V_{loc} controls noise accumulated before communication; V_{agg} controls fresh noise averaged across clients. Since $p_{k,t}^2 \leq p_{k,t}$, $V_{\mathrm{agg}} = V_{\mathrm{loc}}$ is always a valid, possibly loose, choice.

These assumptions do not require an almost-sure bound on Gaussian noise or on the unclipped stochastic gradients. They do require uniform smoothness, bias, and heterogeneity bounds; verifying or estimating those constants for a particular neural network is separate from the proof.

Uniform sampling without replacement. If $M = K'$ clients are selected uniformly without replacement and $p_{k,t} = \mathbf{1}\{k \in \mathcal{S}_t\}/M$, then $\omega_k = 1/K$ and

$$\chi_P^2 = \frac{K - M}{M(K-1)}\zeta_P^2, \qquad V_{\mathrm{agg}} = \frac{V_{\mathrm{loc}}}{M} \quad (K > 1). \tag{51}$$

For $K = M = 1$, take $\chi_P = 0$. To verify the first expression, put $d_k = \nabla f_k(x_t) - \nabla F(x_t)$, so $\sum_k d_k = 0$. With $J_k = \mathbf{1}\{k \in \mathcal{S}_t\}$, $\mathbb{E}J_k = M/K$ and $\mathbb{E}J_k J_j = M(M-1)/(K(K-1))$ for $k \neq j$. Therefore

$$\begin{aligned}\mathbb{E}\left\|\frac{1}{M}\sum_k J_k d_k\right\|^2 &= \frac{1}{M^2}\left(\frac{M}{K}\sum_k \|d_k\|^2 + \frac{M(M-1)}{K(K-1)}\sum_{k\neq j}\langle d_k, d_j\rangle\right) \\ &= \frac{K-M}{M(K-1)}\frac{1}{K}\sum_k \|d_k\|^2.\end{aligned}$$

The second expression follows directly from $\mathbb{E}\sum_k p_{k,t}^2 V_k = (MK)^{-1}\sum_k V_k$. Normalizing arbitrary unequal weights over a sampled subset need not satisfy Eq. 49 for a prescribed ω. Privacy-compatible weights and unbiased optimization weights are distinct requirements; a more general residual bound is given below.

C.4 Finite-horizon nonconvex convergence

Theorem 3 (Main-stage stationarity of FedTIE)**.** *Under the conditions in Appendix C.3, let $T \geq 1$, $\tau \geq 1$, and use a constant learning rate*

$$0 < \eta \leq \frac{1}{16 L_P \tau}. \tag{52}$$

Set $m = \tau - 1$ and $\Delta_P = F(0) - F_*$. Then

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(x_t)\|^2 \le \frac{4\Delta_P}{\eta\tau T} + 10B_P^2 + 120L_P^2\eta^2m^2(\zeta_P^2 + B_P^2) + 40L_P^2\eta^2 m V_{\mathrm{loc}} + 4L_P\eta\tau\chi_P^2 + 4L_P\eta V_{\mathrm{agg}}. \tag{53}$$

The same right-hand side bounds $\min_{0\le t<T}\mathbb{E}\|\nabla F(x_t)\|^2$ and $\mathbb{E}\|\nabla F(x_{\widehat{t}})\|^2$, where $\widehat{t}$ is selected uniformly from $\{0,\dots,T-1\}$ independently of training. The latter selection requires no evaluation of private training gradients.

The first term is the initial restricted-objective gap. The next two terms capture clipping and deterministic local drift. The remaining terms describe stochastic local drift, client sampling, and averaged oracle noise. The m factors make both local-drift terms vanish exactly when $\tau = 1$. The theorem establishes expected first-order stationarity in the selected subspace; it is not a last-iterate, global-optimality, or generalization guarantee for an arbitrary nonconvex loss.

Lemma 5 (Local drift)**.** *Under Eq. 52, define the conditional mean drift*

$$D_t = \frac{1}{\tau}\sum_{s=0}^{\tau-1}\mathbb{E}\left[\sum_k p_{k,t}\|x_{k,t,s} - x_t\|^2 \,\middle|\, \mathcal{T}_t\right]. \tag{54}$$

Writing $g_t = \nabla F(x_t)$, we have

$$D_t \le 12\eta^2m^2(\|g_t\|^2 + \zeta_P^2 + B_P^2) + 4\eta^2 m V_{\mathrm{loc}}. \tag{55}$$

Proof. Condition further on the selected set and its weights, denoting this conditional expectation by $\mathbb{E}_{t,\mathcal{S}}$. For a selected client k, write $\delta_{k,s} = x_{k,t,s} - x_t$ and $D_{k,s} = \mathbb{E}_{t,\mathcal{S}}\|\delta_{k,s}\|^2$. Expanding the local recursion gives

$$\delta_{k,s} = -\eta\sum_{j=0}^{s-1}\big(\nabla f_k(x_{k,t,j}) + e_k(x_{k,t,j})\big) - \eta\sum_{j=0}^{s-1}u_{k,t,j}.$$

The second sum has second moment at most sV_k, by conditional martingale orthogonality. For the first sum, use $\|\sum_{j<s}v_j\|^2 \le s\sum_{j<s}\|v_j\|^2$ and

$$\begin{aligned}\|\nabla f_k(x_{k,t,j}) + e_k(x_{k,t,j})\|^2 &\le 3\|\nabla f_k(x_t)\|^2 + 3\|\nabla f_k(x_{k,t,j}) - \nabla f_k(x_t)\|^2 + 3B_k^2\\ &\le 3\|\nabla f_k(x_t)\|^2 + 3L_P^2\|\delta_{k,j}\|^2 + 3B_k^2.\end{aligned}$$

Using $\|a+b\|^2 \le 2\|a\|^2 + 2\|b\|^2$ yields

$$D_{k,s} \le 6\eta^2s^2(\|\nabla f_k(x_t)\|^2 + B_k^2) + 6\eta^2sL_P^2\sum_{j=0}^{s-1}D_{k,j} + 2\eta^2sV_k. \tag{56}$$

If $m = 0$, all evaluated local states equal x_t, so $D_t = 0$. Otherwise take the maximum over $s \le m$ and put $D_{k,\max} = \max_{0\le s\le m}D_{k,s}$. The coefficient of this maximum on the right is at most $6/256 < 1/2$. Absorbing it into the left-hand side gives

$$D_{k,\max} \le 12\eta^2m^2(\|\nabla f_k(x_t)\|^2 + B_k^2) + 4\eta^2mV_k. \tag{57}$$

Multiply by $p_{k,t}$, sum over selected clients, and average over selection. The identity

$$\sum_k \omega_k\|\nabla f_k(x_t)\|^2 = \|g_t\|^2 + \sum_k\omega_k\|\nabla f_k(x_t) - g_t\|^2$$

and the unbiased-weight condition prove the result. Only states $s \le \tau - 1$ are needed because those are the states at which the round's gradients are evaluated. □

Proof of Theorem 3. Let $\mathbb{E}_t[\cdot] = \mathbb{E}[\cdot \mid \mathcal{T}_t]$. Decompose the exact round direction as

$$\begin{aligned} H_t &= g_t + A_t + R_t + U_t, \\ A_t &= \sum_k (p_{k,t} - \omega_k)\nabla f_k(x_t), \\ R_t &= \frac{1}{\tau}\sum_{s=0}^{\tau-1}\sum_k p_{k,t}\big(\nabla f_k(x_{k,t,s}) - \nabla f_k(x_t) + e_k(x_{k,t,s})\big), \\ U_t &= \frac{1}{\tau}\sum_{s=0}^{\tau-1}\sum_k p_{k,t} u_{k,t,s}. \end{aligned} \tag{58}$$

Here A_t is the client-sampling error at the common starting state, R_t contains local drift and clipping bias, and U_t is the centered oracle error accumulated during the round.

Step 1: moments of the centered errors. The participation condition gives $\mathbb{E}_t A_t = 0$ and $\mathbb{E}_t\|A_t\|^2 \le \chi_P^2$. Conditional on the selection, the oracle errors are martingale differences across local steps, with independent fresh randomness across clients. Thus

$$\mathbb{E}_t U_t = 0, \qquad \mathbb{E}_t\|U_t\|^2 \le \frac{1}{\tau^2}\sum_{s=0}^{\tau-1}\mathbb{E}_t\sum_k p_{k,t}^2 V_k \le \frac{V_{\mathrm{agg}}}{\tau}. \tag{59}$$

Moreover, $\mathbb{E}_t\langle A_t, U_t\rangle = 0$, since A_t is fixed after selection and $\mathbb{E}_{t,\mathcal{S}} U_t = 0$. Consequently,

$$\mathbb{E}_t\|g_t + A_t + U_t\|^2 \le \|g_t\|^2 + \chi_P^2 + V_{\mathrm{agg}}/\tau. \tag{60}$$

We do not assume R_t and U_t are independent: earlier noise affects the states occurring in R_t.

Step 2: the noncentered residual. The coefficients $p_{k,t}/\tau$ are nonnegative and sum to one. Jensen's inequality and smoothness imply

$$\mathbb{E}_t\|R_t\|^2 \le \frac{2}{\tau}\sum_{s=0}^{\tau-1}\mathbb{E}_t\sum_k p_{k,t}\big(L_P^2\|x_{k,t,s} - x_t\|^2 + B_k^2\big) = 2L_P^2 D_t + 2B_P^2. \tag{61}$$

Therefore, without discarding any dependence between the terms,

$$\mathbb{E}_t\|H_t\|^2 \le 2\|g_t\|^2 + 2\chi_P^2 + 2V_{\mathrm{agg}}/\tau + 2\mathbb{E}_t\|R_t\|^2. \tag{62}$$

Step 3: one-round descent. Since F is L_P-smooth and $x_{t+1} = x_t - \gamma H_t$,

$$\begin{aligned} \mathbb{E}_t F(x_{t+1}) &\le F(x_t) - \gamma\langle g_t, \mathbb{E}_t H_t\rangle + \frac{L_P\gamma^2}{2}\mathbb{E}_t\|H_t\|^2 \\ &\le F(x_t) - \gamma\|g_t\|^2 - \gamma\langle g_t, \mathbb{E}_t R_t\rangle + L_P\gamma^2\big(\|g_t\|^2 + \chi_P^2 + V_{\mathrm{agg}}/\tau + \mathbb{E}_t\|R_t\|^2\big). \end{aligned} \tag{63}$$

Use $-\langle g_t, \mathbb{E}_t R_t\rangle \le \|g_t\|^2/4 + \mathbb{E}_t\|R_t\|^2$. Since $L_P\gamma \le 1/16 \le 1/4$, this gives

$$\begin{aligned} \mathbb{E}_t F(x_{t+1}) &\le F(x_t) - \frac{\gamma}{2}\|g_t\|^2 + \frac{5\gamma}{4}\mathbb{E}_t\|R_t\|^2 + L_P\gamma^2(\chi_P^2 + V_{\mathrm{agg}}/\tau) \\ &\le F(x_t) - \frac{\gamma}{2}\|g_t\|^2 + \frac{5\gamma}{2}L_P^2 D_t + \frac{5\gamma}{2}B_P^2 + L_P\gamma^2(\chi_P^2 + V_{\mathrm{agg}}/\tau). \end{aligned} \tag{64}$$

Substitute Lemma 5. The additional coefficient of $\gamma\|g_t\|^2$ is $30L_P^2\eta^2 m^2 \le 30/256 < 1/4$. It can be absorbed into the descent term, yielding

$$\mathbb{E}_t F(x_{t+1}) \le F(x_t) - \frac{\gamma}{4}\|g_t\|^2 + \gamma\mathcal{E}_P(\eta,\tau), \tag{65}$$

where the nonnegative error level is

$$\begin{aligned} \mathcal{E}_P(\eta,\tau) = \frac{5}{2}B_P^2 + 30L_P^2\eta^2 m^2(\zeta_P^2 + B_P^2) + 10L_P^2\eta^2 m V_{\mathrm{loc}} \\ + L_P\eta\tau\chi_P^2 + L_P\eta V_{\mathrm{agg}}. \end{aligned} \tag{66}$$

Step 4: telescoping. Take expectations, sum Eq. 65 over t, and use $F(x_T) \geq F_*$. Then

$$\frac{\gamma}{4}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(x_t)\|^2 \leq F(0) - \mathbb{E}F(x_T) + \gamma T\mathcal{E}_P \leq \Delta_P + \gamma T\mathcal{E}_P.$$

Dividing by $\gamma T/4$ proves Eq. 53. The minimum is bounded by the average, and independent uniform selection of $\widehat{t}$ gives exactly that average in expectation. □

C.5 Rates, privacy noise, and stronger objective conditions

Explicit finite-horizon rate. For example, choose $\eta = 1/(16L_P\tau\sqrt{T})$. Substitution gives

$$\frac{1}{T}\sum_{t<T}\mathbb{E}\|\nabla F(x_t)\|^2 \leq 10B_P^2 + \frac{64L_P\Delta_P + \chi_P^2/4 + V_{\mathrm{agg}}/(4\tau)}{\sqrt{T}} + \frac{15m^2}{32\tau^2 T}(\zeta_P^2 + B_P^2) + \frac{5m}{32\tau^2 T}V_{\mathrm{loc}}. \tag{67}$$

For fixed τ and constants independent of T, this is $10B_P^2 + O(T^{-1/2})$. If clipping is inactive, so $B_P = 0$, it yields an $O(T^{-1/2})$ expected squared-stationarity guarantee. This rate uses one constant learning rate within each prescribed horizon; it is not a claim that a single infinite run uses a horizon-independent step size. In the nonprivate, unclipped case, use $\nu_k = 0$ and a finite bound on the actual unclipped sampling variance in place of Eq. 46. Substituting $C = \infty$ into that clipping-based upper bound would be meaningless.

Persistent clipping bias can prevent stationarity of the original loss, even with vanishing learning rates. For example, take one client with three scalar losses $\ell_1(x) = (x+3)^2/2$ and $\ell_2(x) = \ell_3(x) = (x-1/2)^2/2$, with full-batch averaging and $C = 1$. At $x = 0$ the gradients are $3, -1/2, -1/2$, so the clipped average is zero while the true average gradient is $2/3$. No deterministic clipped step moves from this point. Thus the bias term cannot generally be removed without an additional assumption.

How the private dimension enters. Under homogeneous $q_k = q$, $b_k = b$, $\nu_k = \nu_{\mathrm{m}}$, equal oracle-variance bounds $V_k = V$, and uniform sampling of M clients,

$$V = S + d\left(\frac{\nu_{\mathrm{m}}C_{\mathrm{m}}}{b}\right)^2, \qquad V_{\mathrm{loc}} = V, \qquad V_{\mathrm{agg}} = V/M. \tag{68}$$

The two explicitly Gaussian contributions to Eq. 53 are therefore

$$d\left(\frac{\nu_{\mathrm{m}}C_{\mathrm{m}}}{b}\right)^2\left(\frac{4L_P\eta}{M} + 40L_P^2\eta^2(\tau-1)\right). \tag{69}$$

The first contribution is reduced by client averaging. The second arises because noise perturbs local trajectories before aggregation; this bound does not assign it a $1/M$ reduction. This distinction is lost if the single-step noise identity is applied directly to complete local updates. For heterogeneous clients, replace the two Gaussian variances by $d\sum_k \omega_k v_k^2$ and a bound on $d\mathbb{E}_t\sum_k p_{k,t}^2 v_k^2$, respectively.

Fixed noise scales versus a fixed total privacy budget. Eq. 67 is an optimization rate with specified oracle constants. At a fixed end-to-end (ε, δ), ν_{m} is calibrated jointly with warm-up and the horizon, as in Appendix B.5. If T or τ changes, so can ν_{m} and hence both variance constants. The finite-horizon theorem remains valid with the recalibrated values, but the $O(T^{-1/2})$ simplification requires them to remain bounded independently of T. It does not establish arbitrarily accurate optimization at a fixed privacy budget merely by increasing the number of private updates.

Similarly, comparing a d-dimensional interface with a q_B-dimensional single-factor interface requires comparing the actual noise energies

$$d\left(\frac{\nu_{\mathrm{TIE}}C_{\mathrm{TIE}}}{b_{\mathrm{TIE}}}\right)^2 \quad \text{and} \quad q_B\left(\frac{\nu_{\mathrm{base}}C_{\mathrm{base}}}{b_{\mathrm{base}}}\right)^2,$$

as well as their bias, geometry, and local-drift constants. Smaller private dimension alone does not prove a smaller task loss or a uniformly faster rate than another method.

Clipping tails and a vanishing-bias regime. One explicit sufficient condition for the bias assumption is $\sup_x \mathbb{E}_{z,k}\|g_k(x;z)\|^{1+\rho} \le M_{k,1+\rho}$ for some $\rho > 0$. Since $(r-C)_+ \le r^{1+\rho}/C^\rho$,

$$B_k \le \frac{a_k M_{k,1+\rho}}{C^\rho}, \qquad B_P^2 \le \sum_k \omega_k \frac{a_k^2 M_{k,1+\rho}^2}{C^{2\rho}}. \tag{70}$$

This condition is stronger than clipping itself. Increasing C reduces this tail bound but increases Gaussian variance at a fixed multiplier. For a sequence of finite-horizon runs, a sufficient condition for the right-hand side of Eq. 53 to vanish is that $\Delta_P/(\eta\tau T)$, B_P^2, $L_P^2\eta^2 m^2(\zeta_P^2 + B_P^2)$, $L_P^2\eta^2 m V_{\mathrm{loc}}$, $L_P\eta\tau\chi_P^2$, and $L_P\eta V_{\mathrm{agg}}$ all vanish, with the step-size condition holding in every run. Such a regime must also be checked against the intended privacy accounting.

Theorem 4 (Linear convergence to a neighborhood under a restricted PL condition)**.** *In addition to the conditions of Theorem 3, suppose the restricted objective satisfies, for some $\mu_P > 0$,*

$$\|\nabla F(x)\|^2 \ge 2\mu_P(F(x) - F_*), \qquad x \in \mathbb{R}^d. \tag{71}$$

Choose η as in Eq. 52 and with $\mu_P\eta\tau \le 1$. Then

$$\mathbb{E}[F(x_T) - F_*] \le \left(1 - \frac{\mu_P\eta\tau}{2}\right)^T \Delta_P + \frac{2\mathcal{E}_P(\eta,\tau)}{\mu_P}\left[1 - \left(1 - \frac{\mu_P\eta\tau}{2}\right)^T\right]. \tag{72}$$

There is also a trajectory-moment version: retain smoothness, the PL condition, fresh centered oracle errors, and weights selected before local training, but replace the global bias and heterogeneity bounds and unbiased participation by

$$\sup_{t<T}\left\{\frac{9}{2}\mathbb{E}[L_P^2 D_t + B_t^{\mathrm{clip}} + A_t^{\mathrm{sel}}] + 2L_P\eta\mathbb{E}V_{\mathrm{agg},t}\right\} \le \overline{\mathcal{E}} < \infty. \tag{73}$$

The quantities here are defined in Eq. 82 and Eq. 54. Under the same step-size conditions, Eq. 72 then holds with $\mathcal{E}_P(\eta,\tau)$ replaced by $\overline{\mathcal{E}}$.

Proof. Substituting Eq. 71 into Eq. 65, and taking expectations, gives

$$\mathbb{E}[F(x_{t+1}) - F_*] \le \left(1 - \frac{\mu_P\gamma}{2}\right)\mathbb{E}[F(x_t) - F_*] + \gamma\mathcal{E}_P.$$

Summing the geometric series proves Eq. 72. For the alternative version, the one-round bound preceding Eq. 84 yields the same recursion with error $\overline{\mathcal{E}}$. Here we use the smaller descent coefficient $\gamma/4$ in place of $\gamma/2$. □

The trajectory-moment version is useful with finite clipping thresholds and unbounded Gaussian perturbations, where a global uniform bound on clipping bias may fail even for a quadratic loss. Its error level may depend on the prescribed horizon and must be bounded for the actual trajectory; it is not an automatic asymptotic guarantee. The PL condition is an additional property of the restricted loss, not a consequence of $P^\top P = I_d$ or the TIE assignment objective. With nonzero clipping, stochasticity, or heterogeneous local drift, the result guarantees approach to the stated neighborhood. Exact linear convergence follows in the zero-error case. For example, full participation, one local step, inactive clipping, and deterministic exact gradients give $\mathcal{E}_P = 0$. The theorem does not assume or prove that transformer fine-tuning satisfies a global PL condition.

C.6 Full-space interpretation and the role of TIE

Discarded directions and approximation error. Let $G_t = \nabla\mathcal{F}(\Theta(x_t))$. Orthogonal decomposition gives the exact identity

$$\|G_t\|^2 = \|P^\top G_t\|^2 + \|(I_D - \Pi)G_t\|^2 = \|\nabla F(x_t)\|^2 + \|(I_D - \Pi)G_t\|^2. \tag{74}$$

If $\mathcal{B}_T$ denotes the right-hand side of Eq. 53, then

$$\frac{1}{T}\sum_{t<T}\mathbb{E}\|G_t\|^2 \le \mathcal{B}_T + \mathcal{R}_{P,T}, \qquad \mathcal{R}_{P,T} = \frac{1}{T}\sum_{t<T}\mathbb{E}\|(I_D - \Pi)G_t\|^2. \tag{75}$$

Thus a small latent gradient does not imply a small full-factor gradient unless the discarded component is also controlled. For example, for $\mathcal{F}(\Theta) = \|\Theta - \Theta_*\|^2/2$, the restricted minimizer is $x_* = P^\top(\Theta_* - \theta_{D,\mathrm{ref}})$, yet

$$\nabla\mathcal{F}(\Theta(x_*)) = -(I_D - \Pi)(\Theta_* - \theta_{D,\mathrm{ref}}),$$

which may be nonzero. Full-factor stationarity itself would still be different from unconstrained stationarity with respect to all backbone weights.

When $\mathcal{F}_* := \inf_\Theta \mathcal{F}(\Theta) > -\infty$, define the parameter-sharing approximation gap

$$\mathcal{A}_P = F_* - \mathcal{F}_* \geq 0. \tag{76}$$

It is distinct from the stationarity residual $\mathcal{R}_{P,T}$. Under the additional PL condition, the objective error decomposes as

$$\mathbb{E}[\mathcal{F}(\Theta(x_T)) - \mathcal{F}_*] = \mathbb{E}[F(x_T) - F_*] + \mathcal{A}_P, \tag{77}$$

so Theorem 4 controls the optimization term while $\mathcal{A}_P$ describes the cost of the fixed sharing constraint.

What the assignment can affect. At a fixed factor state, for any vector g_D,

$$\|P^\top g_D\|^2 = \sum_{j=1}^{d} \frac{1}{n_j} \left(\sum_{i \in G_j} \sigma_i g_{D,i} \right)^2. \tag{78}$$

This equation makes the effect of compatible versus conflicting grouping explicit. Together with Eq. 74, it relates grouping to discarded signal. The analogous projected differences in Eq. 41 determine ζ_P, while the tails of $\|P^\top g_D\|$ determine clipping bias through Eq. 47. Because $\|P^\top g_D\| \leq \|g_D\|$, projected per-example clipping residuals obey the pointwise tail inequality

$$(\|P^\top g_D\| - C)_+ \leq (\|g_D\| - C)_+.$$

This compares tail upper bounds at the same factor state and threshold; it does not order the norms of averaged clipping biases after cancellation, or compare different algorithms' trajectories. Also, retaining more aligned signal can increase the projected norm relative to another embedding and hence increase its clipping frequency.

The TIE profiles and load penalties are proxies for choosing favorable values of these quantities. They do not establish inequalities such as $B_{\mathrm{TIE}} \leq B_{\mathrm{random}}$, $\zeta_{\mathrm{TIE}} \leq \zeta_{\mathrm{random}}$, or $\mathcal{A}_{\mathrm{TIE}} \leq \mathcal{A}_{\mathrm{random}}$ without additional assumptions connecting warm-up statistics to main-stage gradients. Every feasible isometric assignment at the same d, clipping threshold, multiplier, and denominator has the same direct Gaussian energy dv_k^2. The theorem therefore explains which optimization quantities the embedding design can influence, while the advantage of the particular heuristic remains an empirical question.

Factor averaging and effective-weight averaging. The actual algorithm has no extra reconstruction residual in its latent recursion: for each realization of the participant weights,

$$\sum_k p_{k,t}\Theta(x_{k,t,\tau}) = \theta_{D,\mathrm{ref}} + P\sum_k p_{k,t}x_{k,t,\tau} = \Theta(x_{t+1}). \tag{79}$$

This does not imply that the factor product commutes with aggregation. For one layer, put $\bar{B} = \sum_k p_k B_k$ and $\bar{A} = \sum_k p_k A_k$. Direct expansion gives

$$\bar{B}\bar{A} - \sum_k p_k B_k A_k = -\sum_k p_k (B_k - \bar{B})(A_k - \bar{A}), \tag{80}$$

which is generally nonzero, including for a shared affine parameterization. Moreover,

$$\left\| \bar{B}\bar{A} - \sum_k p_k B_k A_k \right\|_F \leq \left(\sum_k p_k \|B_k - \bar{B}\|_F^2 \right)^{1/2} \left(\sum_k p_k \|A_k - \bar{A}\|_F^2 \right)^{1/2}.$$

Similarly, splitting a lifted latent perturbation into the factor blocks $(\delta B, \delta A)$ gives

$$(B + \delta B)(A + \delta A) - BA = B\delta A + \delta B A + \delta B \delta A. \tag{81}$$

The last term need not vanish, and the two blocks need not be independent because they arise from the same latent noise. These nonlinear effects are part of the objective F and its smoothness assumptions. Our theorem does not set the effective-weight aggregation error to zero or claim that the map from latent noise to predictions is linear. The exact statements are the additive-noise latent recursion and the isometric factor-space lift in Lemma 3.

C.7 General weights and averaging over warm-up

A residual bound without unbiased participation. To make the scope of the weight condition precise, keep a fixed target F but permit arbitrary nonnegative, normalized participant weights chosen before local training. Retain smoothness and conditionally centered oracle errors, and define

$$A_t^{\mathrm{sel}} = \mathbb{E}_t \left\| \sum_k p_{k,t} \nabla f_k(x_t) - \nabla F(x_t) \right\|^2,$$
$$B_t^{\mathrm{clip}} = \frac{1}{\tau} \sum_{s<\tau} \mathbb{E}_t \sum_k p_{k,t} \|e_k(x_{k,t,s})\|^2, \qquad V_{\mathrm{agg},t} = \mathbb{E}_t \sum_k p_{k,t}^2 V_k. \tag{82}$$

Use D_t as in Eq. 54, without assuming its specialized upper bound. If $L_P \gamma \le 1/4$, the following finite-horizon inequality holds:

$$\frac{1}{T} \sum_{t<T} \mathbb{E} \|\nabla F(x_t)\|^2 \le \frac{2\Delta_P}{\gamma T} + \frac{9}{T} \sum_{t<T} \mathbb{E}[L_P^2 D_t + B_t^{\mathrm{clip}} + A_t^{\mathrm{sel}}] + \frac{4 L_P \eta}{T} \sum_{t<T} \mathbb{E} V_{\mathrm{agg},t}. \tag{83}$$

For completeness, write $H_t = g_t + \widetilde{R}_t + U_t$ by placing selection error together with drift and clipping in $\widetilde{R}_t$. Jensen's inequality gives

$$\mathbb{E}_t \|\widetilde{R}_t\|^2 \le 3(L_P^2 D_t + B_t^{\mathrm{clip}} + A_t^{\mathrm{sel}}).$$

The inequalities

$$\mathbb{E}_t \|H_t\|^2 \le 2\|g_t\|^2 + 4\mathbb{E}_t \|\widetilde{R}_t\|^2 + 4\mathbb{E}_t \|U_t\|^2, \qquad -\langle g_t, \mathbb{E}_t \widetilde{R}_t \rangle \le \|g_t\|^2/4 + \mathbb{E}_t \|\widetilde{R}_t\|^2$$

then imply

$$\mathbb{E}_t F(x_{t+1}) \le F(x_t) - \frac{\gamma}{2} \|g_t\|^2 + \frac{9\gamma}{2} (L_P^2 D_t + B_t^{\mathrm{clip}} + A_t^{\mathrm{sel}}) + \frac{2 L_P \gamma^2}{\tau} V_{\mathrm{agg},t}.$$

Telescoping proves Eq. 83. In particular,

$$A_t^{\mathrm{sel}} \ge \left\| \sum_k (\mathbb{E}_t p_{k,t} - \omega_k) \nabla f_k(x_t) \right\|^2. \tag{84}$$

Persistent selection bias can therefore remain an obstruction to stationarity of a prescribed target loss. Under unbiased participation, Theorem 3 is sharper because it treats the centered sampling error as variance with a learning-rate factor. Weights computed from the current round's final noisy updates are outside both results, since they can correlate with current oracle noise.

If q_k/b_k changes during main training, a similar issue arises even with full participation: the expected unclipped oracle may no longer be the gradient of one fixed f_k. To analyze a fixed target, add the normalization discrepancy to e_k in Eq. 44 and bound it together with clipping. For example, if the target is $\mathcal{L}_k \circ \Theta$, this discrepancy is $(a_{k,t,s} - 1) \nabla (\mathcal{L}_k \circ \Theta)(x_{k,t,s})$ in addition to the scaled clipping bias. A privacy-valid adaptive schedule alone does not remove this optimization discrepancy.

Removing the warm-up conditioning. Let $\mathcal{H}_{\mathrm{w}}$ collect the warm-up transcript and the setup randomness on which we conditioned. The quantities F, F_*, Δ_P, L_P, B_P, and ζ_P may depend on it. If the stated conditions hold for almost every realization and the chosen learning rate satisfies Eq. 52 for each such realization, the law of total expectation gives

$$\mathbb{E}_{\mathcal{H}_{\mathrm{w}},\mathrm{m}} \left[\frac{1}{T} \sum_{t<T} \|\nabla F(x_t)\|^2 \right] \le \mathbb{E}_{\mathcal{H}_{\mathrm{w}}} [\mathcal{B}_T], \tag{85}$$

Algorithm 1 FedTIE: private warm-up and TIE construction

Input: Pretrained model, client datasets $\{\mathcal{D}_k\}_{k=1}^{K}$, rounds R, R_w, clients per round K', local steps τ, learning rate η, dimensions $1 \le d \le D$, public reference seed, assignment weights, and declared sampling rates and client weights.
Predeclare $C_{\mathrm{w}}, C_{\mathrm{m}}, b_{0,\mathrm{w}}, b_{0,\mathrm{m}} > 0$ and calibrate $\nu_{\mathrm{w}}, \nu_{\mathrm{m}}$ to the total (ε, δ) using Appendix B.5.
Output: Frozen TIE rule $(\pi, \sigma, \{n_j\})$ and public reference seed for Algorithm 2.
Generate $B_{\mathrm{ref}}^{\ell} = 0$ and public random A_{ref}^{ℓ} as in Eq. 2; initialize provisional full-space factors to $\theta_{D,\mathrm{ref}}$.
Stage I: Warm-up and TIE construction.
for $t = 0, \ldots, R_w - 1$ **do**
  Sample clients $\mathcal{S}_t \subset \{1, \ldots, K\}$ with $|\mathcal{S}_t| = K'$; broadcast current parameters.
  for each client $k \in \mathcal{S}_t$ **in parallel do**
    For each of τ steps, Poisson-sample a batch and form the publicly normalized joint per-example vector $\Phi(z)$ in Eq. 13.
    Jointly clip, add Gaussian noise to the sum, and release Y^{w} via Eq. 6; record each event in the Stage I accountant.
    Update provisional factors using only the private optimization block, its public inverse scaling, and denominator $b_{0,\mathrm{w}}$; return the resulting private provisional parameters.
  end for
  Aggregate provisional parameters with the declared client weights for the next warm-up round.
end for
Construct $\{u_i, s_i, e_i, h_i\}_{i=1}^{D}$ solely from the client-indexed private transcript $\mathcal{Y}_{\mathrm{w}}$.
Initialize assignment with one distinct coordinate per slot; assign remaining coordinates and forbid moves that empty a slot during $\mathrm{BalancedAssign}$.
Obtain assignment map and signs under $|G_j| \ge 1$:

$$(\pi, \sigma) \leftarrow \mathrm{BalancedAssign}\big(\{u_i, s_i, e_i, h_i\}_{i=1}^{D}; \alpha, \beta, \gamma, \eta_h, \lambda\big).$$

Repair any empty candidate slots by the deterministic rule in the embedding parameterization before normalization.
Define $G_j = \{i : \pi(i) = j\}$, $n_j = |G_j| \ge 1$, and the frozen embedding $P \in \mathbb{R}^{D \times d}$ by

$$P_{ij} = \begin{cases} \sigma_i / \sqrt{n_j}, & \pi(i) = j, \\ 0, & \text{otherwise.} \end{cases}$$

provided the right-hand side is finite. Uniform constants over warm-up realizations give the same deterministic upper bound directly. Conditioning only on favorable embeddings would instead yield a conditional guarantee and would require a separate treatment of the remaining event.

There is no additive warm-up optimization-error term because the provisional parameters are discarded: $F(0)$ is always evaluated at the public reference. Warm-up affects the distribution of the selected subspace and its approximation, bias, and heterogeneity constants, as well as the main-stage noise calibration. Thus Eq. 85 is a convergence guarantee for the two-stage procedure's main-stage output, conditional on the stated regularity assumptions, without assuming that the combinatorial assignment heuristic finds an optimal embedding. Its privacy guarantee remains the separate end-to-end guarantee already proved in Appendix B.

D ALGORITHM

Algorithms 1 and 2 summarize the two-stage FedTIE procedure. The first stage uses a privacy-accounted warm-up transcript to construct the TIE rule, whereas the second stage freezes this rule and performs federated optimization entirely in the shared latent space. The separation is deliberate: the provisional warm-up factors are discarded rather than used to initialize main training, and only the embedding constructed from privatized warm-up outputs is retained. Consequently, all data dependence introduced during embedding construction is covered by the Stage I privacy accountant, while the subsequent construction of the assignment and embedding is post-processing.

Algorithm 1 details the private warm-up and embedding construction. At each local step, a selected client jointly encodes its provisional optimization signal and the statistics required by TIE, applies a

Algorithm 2 FedTIE: main training under the frozen TIE rule

Input: Frozen rule and public reference seed from Algorithm 1, and the declared main-stage hyperparameters and client weights.
Output: Global latent increments θ_d^R, frozen rule, and public reference seed.
Discard provisional factors and set $\theta_d^{R_w} = 0$ around $\theta_{D,\mathrm{ref}}$; send the frozen rule and reference seed once to each client's cache.
for $t = R_w, \ldots, R-1$ **do**
  Sample clients $\mathcal{S}_t \subset \{1, \ldots, K\}$ with $|\mathcal{S}_t| = K'$.
  Broadcast θ_d^t to clients in $\mathcal{S}_t$; supply setup metadata if their cache is missing.
  for each client $k \in \mathcal{S}_t$ **in parallel do**
    Reconstruct $\theta_D^t = \theta_{D,\mathrm{ref}} + P\theta_d^t$ and reshape it into LoRA matrices $\{A^\ell, B^\ell\}_{\ell=1}^L$.
    Initialize local parameter $\theta_{d,k}^{t,0} \leftarrow \theta_d^t$.
    For $\ell = 0, \ldots, \tau - 1$, Poisson-sample $\mathcal{B}_{k,t,\ell}^{\mathrm{m}}$ and compute gradients at $\theta_{D,\mathrm{ref}} + P\theta_{d,k}^{t,\ell}$:

$$Z \sim \mathcal{N}(0, \nu_{\mathrm{m}}^2 C_{\mathrm{m}}^2 I_d),$$

$$\theta_{d,k}^{t,\ell+1} = \theta_{d,k}^{t,\ell} - \frac{\eta}{b_{0,\mathrm{m}}} \left[\sum_{z \in \mathcal{B}_{k,t,\ell}^{\mathrm{m}}} \mathrm{clip}_{C_{\mathrm{m}}}(P^\top g_D(z)) + Z \right].$$

    Use a zero sum for empty batches, retain Gaussian noise, and account for every scheduled step.
    Send low-dimensional update $\Delta_k^t = \theta_{d,k}^{t,\tau} - \theta_d^t$ to the server.
  end for
  Aggregate updates:

$$\theta_d^{t+1} \leftarrow \theta_d^t + \sum_{k \in \mathcal{S}_t} p_{k,t} \Delta_k^t.$$

  Here $p_{k,t}$ are the declared weights, normalized over $\mathcal{S}_t$; unprotected dataset counts are not released or used to set weights.
end for

single per-example clipping operation, and releases only the corresponding Gaussian-perturbed sum. The provisional factors are updated from the privatized optimization block, and the server aggregates these private parameters to continue warm-up. After R_w rounds, the resulting client-indexed transcript is used to estimate the update-profile, sensitivity, energy, and heterogeneity proxies. The balanced assignment then groups factor coordinates into nonempty latent slots and determines their optional signs. Normalizing every slot by $1/\sqrt{n_j}$ yields the frozen rule $(\pi, \sigma, \{n_j\})$ and guarantees $P^\top P = I_d$; hence, clients need only cache the rule and public reference seed rather than materialize a dense embedding matrix.

Algorithm 2 describes the subsequent private federated training stage. Main training restarts from $\theta_d^{R_w} = 0$ around the public reference, so the provisional warm-up parameters do not carry optimization progress into Stage II. In each round, selected clients reconstruct the LoRA factors from the common affine parameterization, project each per-example gradient into $\mathbb{R}^d$ before clipping and perturbation, and return only the resulting latent update. The server therefore aggregates updates in a shared coordinate system and broadcasts only the next d-dimensional iterate. Empty sampled batches still incur the scheduled Gaussian mechanism, and every local release is included in the Stage II accountant; together with Stage I, these releases are calibrated to the declared end-to-end privacy budget.

E Experimental Details

E.1 Datasets and Baselines

Natural Language Understanding. Following the experimental protocols of FFA-LoRA and FedSVD, we evaluate on four tasks from the GLUE benchmark (Wang et al., 2018). Multi-Genre Natural Language Inference (MNLI) (Williams et al., 2018) is a three-way natural language inference task with entailment, neutral, and contradiction labels, for which we report both matched (in-domain) and mismatched (cross-domain) results. Stanford Sentiment Treebank v2 (SST-2) (Socher

et al., 2013) performs binary sentiment classification, Quora Question Pairs (QQP) (Sharma et al., 2019) identifies whether two questions are semantically equivalent, and Question Natural Language Inference (QNLI) determines whether a context sentence entails the answer to a given question. Because the labels of the official GLUE test sets are not publicly available, we report performance on the corresponding validation sets, including both the matched and mismatched validation sets for MNLI.

Natural Language Generation. Following the experimental protocol of FedEX-LoRA (Singhal et al., 2024), we evaluate data-to-text generation on the E2E NLG Challenge dataset (Novikova et al., 2017). The dataset contains approximately 42,000 training examples and 4,600 examples in each of the validation and test sets, all drawn from the restaurant domain. Each example pairs a meaning representation—a sequence of slot–value attributes—with one or more natural-language references, thereby assessing whether a model can generate fluent text that faithfully realizes the structured input. The dataset is distributed under the Creative Commons BY-NC-SA 4.0 license.

Baselines. We compare FedTIE with the following six federated LoRA baselines:

1. **FedAvg** (Zhang et al., 2024; Liu et al., 2025): Each client locally fine-tunes both LoRA factors A and B, which the server aggregates independently by element-wise averaging.
2. **FFA-LoRA** (Sun et al., 2024): A shared randomly initialized factor A is frozen throughout training, while clients privately optimize and communicate only B for server aggregation.
3. **FedSVD** (Lee et al., 2025): Clients update and upload only B, after which the server forms the effective update using the previous-round A and applies an SVD-based reparameterization to update the LoRA factors.
4. **FedASK** (Wen et al., 2025): A two-stage sketching procedure reconstructs a communication-efficient high-rank approximation to the aggregated effective update $\sum_k B_k A_k$ before updating the global LoRA factors.
5. **LA-LoRA** (Liu et al., 2026): Clients alternate private optimization of A and B across successive local steps and can smooth the perturbed gradients with a low-pass filter to mitigate gradient coupling and DP noise.
6. **AS-LoRA** (Kim et al., 2026): Curvature-aware scores adaptively select a shared active factor for each layer and communication round, so that clients privately optimize and communicate only the selected factor.

E.2 Hyperparameter Details

Natural Language Understanding. For all natural language understanding experiments on MNLI-matched, MNLI-mismatched, SST-2, QQP, and QNLI, we use RoBERTa-base as the backbone and insert LoRA modules into the query and value projections of each Transformer layer. We set the LoRA rank to $r = 8$, the scaling factor to $\alpha = 8$, and the dropout rate to 0, with rank-stabilized LoRA disabled. Input sequences are truncated to 128 tokens, and both the training and evaluation batch sizes are set to 32. We tune the main learning rate over $\{1 \times 10^{-3}, 5 \times 10^{-3}, 1 \times 10^{-2}\}$; the classifier learning rate is tuned over $\{1 \times 10^{-4}, 1 \times 10^{-3}, 1 \times 10^{-2}\}$ for every task. We use standard SGD with a linear-decay schedule, a warmup ratio of 0.06, and weight decay of 0.01. Federated training uses three clients, all of which participate in every communication round. We train for $R = 150$ rounds with $\tau = 10$ local steps per round and partition each task among clients using a non-IID Dirichlet distribution with concentration parameter $\alpha_{\mathrm{Dir}} = 0.5$. For FedTIE, the first $R_w = 20$ rounds are used to construct the embedding, with three local warm-up steps per round, and the remaining 130 rounds optimize a latent parameter vector of dimension $d = 46{,}080$. Unless varied in an ablation, the TIE coefficients are set to $\alpha_{\mathrm{TIE}} = \beta_{\mathrm{TIE}} = \gamma_{\mathrm{TIE}} = \eta_{\mathrm{TIE}} = \lambda_{\mathrm{TIE}} = 1$. For private experiments, we consider total privacy budgets $\epsilon \in \{3, 6\}$ with $\delta = 10^{-5}$ and use 10 gradient-accumulation steps. We select the clipping threshold from $C \in \{1, 2, 5, 10\}$; the noise multipliers are calibrated rather than manually specified. For FedTIE, this calibration accounts for the warm-up and main-stage release schedules under the same end-to-end privacy budget. We report results over random seeds 42, 43, and 44.

Table 6: Runtime and GPU memory consumption of different federated fine-tuning methods. Lower values indicate better computational efficiency.

Method	Mean Round Time (s) ↓	Total Runtime (s) ↓	Peak Alloc. GPU (MiB) ↓	Peak Res. GPU (MiB) ↓
FedAvg	40.88	6131.67	2304.94	2974.0
FFA-LoRA	39.49	5922.76	2159.16	2850.0
FedSVD	39.91	5986.93	2157.53	2850.0
FedASK	39.53	5929.43	2159.16	3440.0
LA-LoRA	40.84	6125.66	2303.26	2972.0
AS-LoRA	40.53	6080.10	2297.41	2966.0
FedTIE (ours)	41.93	6290.19	2157.53	2850.0

Natural Language Generation. For the E2E NLG Challenge, we use GPT-2 as the backbone and configure LoRA with rank $r = 4$, scaling factor $\alpha = 32$, and dropout rate 0, while disabling rank-stabilized LoRA and gradient checkpointing. We set the maximum input length to 160 tokens and both the maximum target length and the maximum number of newly generated tokens to 96. The training and evaluation batch sizes are 8 and 32, respectively, with one gradient-accumulation step. We use a learning rate of 1×10^{-3}, weight decay of 0.01, and a linear-decay schedule with 500 warmup steps. Federated training proceeds for 10 communication rounds over three clients; all clients participate in each round and perform three local epochs before aggregation. The client data are partitioned IID. Differential privacy is enabled with $\epsilon = 6$ and $\delta = 10^{-5}$; per-sample gradients are computed using hooks, and the noise multiplier is calibrated rather than manually specified. We select the clipping threshold from $C \in \{1, 2, 5, 10\}$. At inference time, we use beam search with 10 beams, a length penalty of 0.9, and a no-repeat 4-gram constraint; label smoothing is set to 0.1. For FedTIE, the first $R_w = 3$ rounds are used to construct the embedding, and the remaining 7 rounds optimize a latent parameter vector of dimension $d = 46{,}080$. Unless varied in an ablation, the TIE coefficients are set to $\alpha_{\text{TIE}} = \beta_{\text{TIE}} = \gamma_{\text{TIE}} = \eta_{\text{TIE}} = \lambda_{\text{TIE}} = 1$. We report results over random seeds 42, 43, and 44. All experiments are conducted using one NVIDIA RTX 4090 GPU and one NVIDIA RTX 5090 GPU.

E.3 Computational Efficiency

We evaluate runtime and peak GPU memory consumption on MNLI with RoBERTa-base at LoRA rank $r = 8$, under $(\epsilon, \delta) = (6, 10^{-5})$ and a non-IID Dirichlet partition with $\alpha_{\text{Dir}} = 0.5$. Table 6 reports a mean round time of 41.26 s and a total runtime of 6,189.51 s for FedTIE over 150 communication rounds. Relative to FFA-LoRA, the fastest baseline, FedTIE increases total runtime by 4.5%; relative to FedAvg, the increase is 0.9%. Thus, the utility and communication benefits reported above come with a modest runtime overhead in this setting. Although main-stage optimization operates in $d = 46{,}080$ latent coordinates, clients still reconstruct the LoRA factors and execute the backbone forward and backward passes. The smaller trainable space therefore does not imply a proportional reduction in training time. FedTIE's peak allocated GPU memory is 2,157.53 MiB, tying FedSVD for the lowest value and reducing the peak by 6.4% relative to FedAvg. Its peak reserved memory is 2,850 MiB, also tying the lowest value among the evaluated methods. These measurements are consistent with a compact implementation of TIE: the embedding is stored through assignment indices, signs, and normalization factors, without materializing a dense $D \times d$ matrix. The resulting memory footprint remains comparable to the most memory-efficient baselines, while the runtime measurements establish the practical cost of the evaluated training configuration rather than isolate the costs of warm-up, embedding construction, or factor reconstruction.