Beyond Parameter Aggregation: Semantic Consensus for Federated Fine-Tuning of LLMs

Amr Abourayya
Lamarr Institute, TU Dortmund, and
Institute for AI in medicine (IKIM), UK Essen
amr.abourayya@tu-dortmund.de

Jens Kleesiek
Institute for AI in medicine (IKIM), UK Essen

Michael Kamp
Lamarr Institute, TU Dortmund, and
Institute for AI in medicine (IKIM), UK Essen

Abstract

Federated fine-tuning of large language models is commonly formulated as a parameter aggregation problem. However, even parameter-efficient methods require transmitting large collections of trainable weights, assume aligned architectures, and rely on white-box access to model parameters. As model sizes continue to grow and deployments become increasingly heterogeneous, these assumptions become progressively misaligned with practical constraints. We introduce Semantic Consensus, a behavior-space alternative to parameter aggregation in which federated collaboration is mediated through model outputs rather than model parameters. Clients fine-tune local models on private data and exchange generated outputs on a shared public prompt set. The server maps these outputs into a semantic representation space, forms a per-prompt semantic consensus, and returns pseudo-labels for further local fine-tuning. This formulation fundamentally changes the communication scaling of federated LLM fine-tuning. The amount of information exchanged depends only on the public prompt budget and the size of the communicated behaviors, independent of model size. As a consequence, the protocol naturally accommodates heterogeneous architectures and applies directly to open-ended text generation. We present a theoretical analysis and empirical results demonstrating that this approach can match strong federated fine-tuning baselines while substantially reducing communication by orders of magnitude (e.g., 0.006 GB for FedCoFiT versus 3.75 GB for FlexLoRA on LLaMA-7B, approximately a $600\times$ reduction), alongside reductions in runtime and energy consumption. These results suggest that, for generative foundation models, behavior-level consensus provides a promising alternative to parameter aggregation for federated adaptation.

1 Introduction

Large language models are increasingly being adapted to specific sensitive domains, including healthcare, finance, and enterprise workflows. In many practical deployments, the data required for effective fine-tuning is distributed across organizations and devices and cannot be centralized due to privacy, legal, or competitive constraints. Federated learning provides a natural framework for collaborative adaptation in such settings, enabling participants to improve local models without exposing sensitive raw data. Yet for modern LLMs, the dominant practical constraint is often not local compute but **what can be communicated.**

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

Most federated learning protocols are parameter-centric, which means that clients transmit and aggregate weights, gradients, or adapter updates. For LLMs, this formulation faces three fundamental difficulties. First, communication cost becomes a dominant bottleneck. Even parameter-efficient fine-tuning approaches require repeatedly transmitting large collections of trainable weights, which are increasingly constrained by realistic bandwidth limits. Second, clients may differ in LoRA rank, adapted layers, tokenizers, or even base architectures, making parameter aggregation difficult or ill-defined. Third, white-box access to model parameters is often unavailable in practice.

These difficulties reflect a more fundamental misalignment between parameter aggregation and the deployment and use of modern LLMs. In practice, what downstream systems consume from an LLM is not its internal parameters but its behaviour. For example, responses to prompts, preference judgments, tool calls, structured outputs, safety decisions, or intermediate reasoning artifacts emitted by the interface. This motivates a more general question: **Can federated collaboration be mediated through behaviour rather than parameters?**

At a high level, behaviour sharing treats each client model as a black-box client that can be queried on shared public stimuli and that can return compact, shareable summaries of its behaviour. The shared object need not be restricted to predictions in the classical supervised sense, depending on the application. It could be ranked preferences (for alignment), structured API calls (for tool-augmented workflows), extracted entities or plans (for enterprise pipelines), or free-form generation (for instruction following). If collaboration is expressed in a common *semantic* space over such behaviours, then communication can be decoupled from model size and remain meaningful even when clients are heterogeneous.

We call this general approach **Semantic Consensus**: a federated co-training principle in which clients exchange model behaviour on a shared public prompt set and a server aggregates these behaviours to construct consensus targets that guide further local fine-tuning. For LLMs, behaviour is naturally captured by responses generated to a set of shared public prompts. By embedding these responses into a semantic space, consensus can be formed via geometric operations (e.g., clustering) that identify semantically consistent groups and select representative behaviours as supervision. We provide a concrete instantiation of this principle in Sec. 3. This design directly addresses the main friction points of the federated LLM adaptation. Communication becomes independent of model size, since clients transmit only generated text responses and receive consensus targets. Because aggregation occurs in the behaviour space, the framework naturally accommodates heterogeneity in architectures, tokenizers, and LoRA configurations. Finally, Semantic Consensus requires only forward access to model behaviour, making it applicable to open-ended generation without task-specific heads.

In summary, our contributions are as follows.

- We introduce **Semantic Consensus**, a general federated co-training principle that replaces parameter aggregation with consensus over client behaviour on shared public inputs, yielding communication independent of model size.
- We propose **FedCoFiT**, an instantiation of Semantic Consensus for open-ended text generation that builds consensus using semantic embeddings, similarity metrics, and clustering to obtain high-quality pseudo-labels for further local fine-tuning.
- We provide a theoretical analysis explaining why behaviour-sharing yields substantially improved communication scaling relative to adapter aggregation, and we show that FedCoFiT is compatible with record-level differential privacy.
- We empirically evaluate FedCoFiT across multiple benchmarks, showing that it can match or approach strong federated fine-tuning baselines while significantly reducing communication and improving practical efficiency (runtime and energy consumption).

2 Related Work

Traditional federated learning approaches rely on iterative model averaging (e.g., FedAvg McMahan et al. [2017]), requiring clients to upload full model updates constantly. Although effective for smaller architectures, it becomes impractical for large language models due to the high cost of communication. A promising solution to this bottleneck is the adoption of parameter-efficient fine-tuning (PEFT) within the Federated Learning framework.

Parameter-efficient federated fine-tuning of LLMs such as LoRA Hu et al. [2022] is widely adopted to reduce the number of trainable parameters, and several works adapt LoRA to federated fine-tuning by aggregating LoRA updates at the server. A key challenge is that naively averaging LoRA adapters can be mathematically inconsistent, introducing aggregation noise and degrading performance when client configurations differ. For example, FedIT Zhang et al. [2024] integrates LoRA directly with FedAvg by having the server independently average the local LoRA adapters $\mathbf{A}$ and $\mathbf{B}$ received from clients. While this reduces communication costs, it cannot aggregate local LoRAs with heterogeneous ranks and introduces aggregation noise as the matrix multiplication required to reconstruct the update is non-linear. Specifically, the product of the averaged matrices is not equal to the sum of the individual products, leading to mathematical errors in the global model.

FLoRA addresses this issue via a stacking-based aggregation method that supports heterogeneous LoRA adapters and improves the stability of federated LoRA fine-tuning Wang et al. [2024]. FlexLoRA synthesizes and redistributes LoRA weights using SVD to exploit heterogeneous client budgets Bai et al. [2024], while HetLoRA (and related heterogeneous-LoRA methods) allows different ranks across devices and aggregates while accounting for differences in data and sparsity patterns Cho et al. [2024]. More recent studies improved initialization, aggregation, and robustness for federated LoRA tuning Bian et al. [2025], Koo et al. [2025]. These approaches are strong baselines when clients share a compatible parameter space and have white-box access to adapters. However, they fundamentally rely on transmitting trainable parameters (or structured adapter weights), which remains costly at scale and becomes inapplicable when clients use different architectures, tokenizers, or only have black-box access. More recent work continues to refine parameter-space aggregation for federated LoRA. Fed-PLoRA Zhang et al. [2026] introduces parallel one-rank LoRA modules and a Select-N-Fold strategy to reduce initialization and aggregation noise under heterogeneous client ranks, while SFed LoRA Huang et al. [2026] studies the interaction between LoRA rank, client count, and aggregation stability through a rank-dependent scaling factor. These methods further improve heterogeneous LoRA aggregation, but they remain parameter-centric as clients must still communicate adapter-side information and the server must reason about compatible LoRA structures.

An alternative to parameter aggregation is to avoid sharing weights altogether, e.g., via distillation or black-box control. Distillation-based FL can be viewed as a special case of behavior sharing under a fixed label space, where the shared behavior is a soft predictive distribution over classes. FedID Ma et al. [2023] is a direct application of this distillation paradigm, but it applies only to classification with fixed label spaces and does not extend to open-ended generation, where token-level supervision and an unconstrained output space make aggregation more complex and bandwidth-intensive. Additionally, it assumes server-held labels and iterative feedback. FedCT [Abourayya et al., 2025] is a particularly close methodological precursor. It constructs consensus pseudo-labels from client-generated hard-label predictions on public unlabeled data. Unlike FedCT, which constructs consensus over predictions in a fixed label space, FedCoFiT extends consensus-based collaboration to unconstrained, variable-length generative responses by aggregating their semantic representations. When parameters are inaccessible, black-box FL instead tunes prompts via gradient-free updates (e.g., FedBPT Sun et al. [2024]), but prompt tuning is often less expressive than weight adaptation and can be query-intensive.

3 Method

We consider K clients indexed by $i \in \{1, \ldots, K\}$. Client i has a private dataset $D_i = \{z_{i,k}\}_{k=1}^{n_i}$ drawn from a client-specific distribution $\mathcal{P}_i$ and a local language model $\mathcal{M}_i$. Clients may differ in architecture, tokenizer, adaptation module, LoRA rank, or access level. They collaborate without sharing raw data, gradients, optimizer states, model parameters, or adapter weights. All clients also have access to a public prompt set $D_{\text{pub}} = \{x_j\}_{j=1}^{M}$ drawn from a public prompt distribution $\mathcal{P}_{\text{pub}}$, containing inputs only. Let $\mathcal{B}_i(x) \in \mathcal{Y}$ denote the observable behavior of client i on input x, such as a generated response, preference ranking, tool call, or structured output. Our goal is to use D_{pub} to transfer behavior across clients while keeping communication independent of model size, avoiding parameter- or adapter-alignment assumptions, and preserving the federated constraint that private data remain local.

Semantic Consensus is a behavior-level aggregation principle for federated fine-tuning. Instead of averaging parameters, clients expose their behavior on public prompts, and the server aggregates behaviors in a shared semantic space. Let $g : \mathcal{Y} \to \mathbb{R}^{d_e}$ be a semantic encoder and $\text{dist}(\cdot, \cdot)$ a

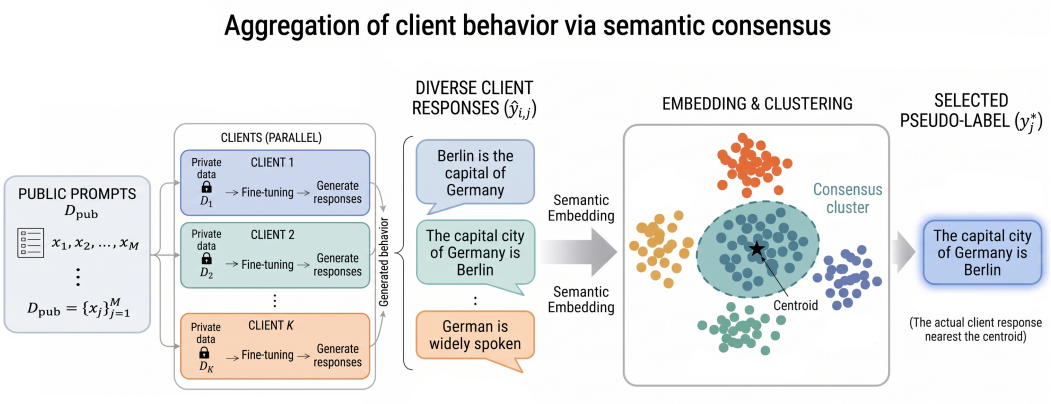

Figure 1: **Overview of FedCoFiT.** Clients generate free-form responses to shared public prompts. The server embeds these responses into a semantic space, identifies the largest non-outlier cluster, and selects the client-generated response nearest to its centroid as the consensus pseudo-label for further local fine-tuning.

distance function in embedding space. For each prompt x_j, client i produces a behavior $b_{i,j} = \mathcal{B}_i(x_j)$ and sends it to the server. The server computes $e_{i,j} = g(b_{i,j})$ and aggregates $\{e_{i,j}\}_{i=1}^{K}$ to form a consensus target t_j. This yields a pseudo-labeled public set $\tilde{D}_{\text{pub}} = \{(x_j, t_j)\}_{j=1}^{M}$, which is returned to clients for local fine-tuning. Because aggregation occurs over behaviors, Semantic Consensus does not require clients to share architectures, tokenizers, adapter shapes, or white-box model access.

3.1 FedCoFiT: Semantic Consensus for Generative LLM Fine-Tuning

We instantiate Semantic Consensus for open-ended generation as **FedCoFiT**. For each public prompt x_j, client i generates a free-form response $\hat{y}_{i,j} \leftarrow \mathcal{M}_i(x_j)$. The server embeds each response with a sentence encoder, normalizes the embedding, clusters the responses by semantic similarity, and selects a representative response as the pseudo-label. For each response $\hat{y}_{i,j}$, the server computes $v_{i,j} = g(\hat{y}_{i,j}) \in \mathbb{R}^{d_e}$ and normalizes it as $e_{i,j} = v_{i,j}/\|v_{i,j}\|_2$. All clustering and representative-selection steps are performed on the normalized embeddings. We use cosine distance, $\text{dist}(e, e') = 1 - e^\top e'$, so semantically similar responses are close even when they differ in surface form. The encoder choice, decoding parameters, and clustering hyperparameters are fixed within each run and reported in the experimental setup.

Algorithm 1: FedCoFiT via Semantic Consensus

Input: clients $\{1, \ldots, K\}$, private datasets $\{D_i\}$; public prompts $D_{\text{pub}} = \{x_j\}_{j=1}^{M}$; encoder g; distance dist; clustering rule Cluster
Output: updated models $\mathcal{M}_1, \ldots, \mathcal{M}_K$

initialize $\mathcal{M}_1, \ldots, \mathcal{M}_K$;
repeat
  foreach *client* $i \in \{1, \ldots, K\}$ *in parallel* **do**
    update $\mathcal{M}_i$ on D_i;
    generate $\hat{y}_{i,j} \leftarrow \mathcal{M}_i(x_j)$ for all $x_j \in D_{\text{pub}}$;
    send $\{\hat{y}_{i,j}\}_{j=1}^{M}$ to server;
  end
  foreach $x_j \in D_{\text{pub}}$ **do**
    compute $e_{i,j} = g(\hat{y}_{i,j})/\|g(\hat{y}_{i,j})\|_2$, $i \in \{1, \ldots, K\}$;
    cluster $\{e_{i,j}\}_{i=1}^{K}$ using Cluster and dist;
    select consensus cluster $C_j^\star$;
    compute centroid $\bar{e}_j$ of $C_j^\star$;
    choose $i^\star = \arg\min_{i \in C_j^\star} \text{dist}(e_{i,j}, \bar{e}_j)$;
    set $y_j^\star = \hat{y}_{i^\star,j}$;
  end
  broadcast $\tilde{D}_{\text{pub}} = \{(x_j, y_j^\star)\}_{j=1}^{M}$;
  foreach *client* $i \in \{1, \ldots, K\}$ *in parallel* **do**
    update $\mathcal{M}_i$ on $D_i \cup \tilde{D}_{\text{pub}}$;
  end
until *convergence*;

For each prompt x_j, the server clusters the embeddings $\{e_{i,j}\}_{i=1}^{K}$ using a density-based clustering rule. Density-based clustering is useful because the number of semantic response modes is unknown and isolated responses can be treated as outliers. The server selects the largest non-outlier cluster as the consensus cluster $C_j^\star$. If multiple clusters have the same size, the server selects the cluster with the smallest average pairwise cosine distance, with any remaining tie broken by the smallest client index. If all responses are marked as outliers, the server uses all responses as the candidate set by setting $C_j^\star = \{1, \ldots, K\}$. Given $C_j^\star$, the server computes

its normalized centroid $\bar{e}_j = \sum_{i \in C_j^\star} e_{i,j} / \| \sum_{i \in C_j^\star} e_{i,j} \|_2$. The pseudo-label is the original response closest to this centroid: $i^\star = \arg\min_{i \in C_j^\star} \mathrm{dist}(e_{i,j}, \bar{e}_j)$ and $y_j^\star = \hat{y}_{i^\star,j}$. If multiple responses are equally close to the centroid, ties are broken by the shortest response length and then by the smallest client index. This keeps the pseudo-label as a real client-generated response rather than requiring the server to synthesize a new target. The server constructs $\tilde{D}_{\mathrm{pub}} = \{(x_j, y_j^\star)\}_{j=1}^{M}$ and broadcasts it to all clients. Each client then continues fine-tuning on $D_i \cup \tilde{D}_{\mathrm{pub}}$ using the standard causal language-modeling loss. Private examples and consensus pseudo-labeled examples receive equal loss weight. The local objective is

$$\mathcal{L}_i(\theta_i) = \frac{1}{2}\mathbb{E}_{(x,y)\sim D_i}\big[\ell_{\mathrm{CE}}(\mathcal{M}_{\theta_i}(x), y)\big] + \frac{1}{2}\mathbb{E}_{(x,y^\star)\sim \tilde{D}_{\mathrm{pub}}}\big[\ell_{\mathrm{CE}}(\mathcal{M}_{\theta_i}(x), y^\star)\big],$$

where ℓ_{CE} is applied to response tokens. In the convergence analysis, this corresponds to the normalized objective weights $\alpha = \beta = 1/2$. FedCoFiT performs aggregation in semantic behavior space rather than parameter space. The majority-cluster rule favors responses supported by multiple clients, while centroid-based representative selection keeps the pseudo-label as a natural response generated by an actual client. Since clients communicate only public-prompt behaviors and receive pseudo-labels, the protocol remains communication-efficient, architecture-agnostic, and compatible with heterogeneous LLMs. FedCoFiT assumes the availability of a shared set of public prompts that is accessible to all clients. The prompts need not be sampled from exactly the same distribution as the private client data, but they should remain reasonably aligned with the downstream task. Public prompts may be collected from existing datasets or generated synthetically, and no ground-truth responses are required. We empirically investigate the effect of the public prompt budget and distribution shift in Appendix E.1 and E.2.

4 Theory

4.1 Communication Complexity

Semantic Consensus avoids parameter exchange and instead communicates model behavior on a shared public prompt set. Let $M = |D_{\mathrm{pub}}|$ be the number of public prompts per round, K the number of clients, and $\bar{\ell}$ the average response length in tokens. In each round, every client uploads one text response per public prompt, so the upload scales as $KM\bar{\ell}$. The server computes semantic embeddings and performs clustering and representative selection locally. It then broadcasts one pseudo-label per public prompt to all clients, so the download also scales as $KM\bar{\ell}$. Let c_t be the average number of bytes per token. The total per-round communication, upload plus download and aggregated over all clients, is $C_{\mathrm{SC}} = \Theta\big(KM\bar{\ell}c_t\big)$, which is independent of model size and does not require aligned architectures or adapter shapes.

Parameter sharing baselines. In contrast, parameter-centric FL methods communicate model updates. For parameter subsampling, client i transmits a fraction $\rho \in (0, 1]$ of an update with P_i parameters. If c_{param} denotes bytes per communicated scalar, the per-round total communication scales as $C_{\mathrm{subsample}} = \Theta\big(K\rho P_i c_{\mathrm{param}}\big)$, up to comparable cost for broadcasting aggregated updates.

LoRA aggregation. Suppose LoRA is applied to L_{layers} layers at client i, with rank r_i, and layer ℓ has dimensions $(d_\ell^{\mathrm{in}}, d_\ell^{\mathrm{out}})$. The LoRA parameter count is $\sum_{\ell=1}^{L_{\mathrm{layers}}} r_i(d_\ell^{\mathrm{in}} + d_\ell^{\mathrm{out}})$, so the per-round total communication scales as $C_{\mathrm{LoRA}} = \Theta\Big(Kc_{\mathrm{param}} \sum_{\ell=1}^{L_{\mathrm{layers}}} r_i(d_\ell^{\mathrm{in}} + d_\ell^{\mathrm{out}})\Big)$, up to method-specific overhead. This cost scales with model geometry and LoRA rank and assumes aligned adapter shapes. Semantic Consensus instead scales with $(K, M, \bar{\ell})$ and remains valid under LoRA, architecture, tokenizer, and access heterogeneity.

Example: Llama 3.1 405B scale. Consider $K = 10$ clients, $M = 1024$ prompts per round, average response length $\bar{\ell} = 128$, and $c_t \approx 2$ bytes/token. Semantic Consensus communicates only public-prompt responses and pseudo-labels, so its total upload-plus-download cost is

$$C_{\mathrm{SC}} \approx 2KM\bar{\ell}c_t = 2 \cdot 10 \cdot 1024 \cdot 128 \cdot 2 = 5.0 \text{ MiB},$$

where the leading factor 2 accounts for upload and download. Now consider a Llama 3.1–405B-scale backbone with $L_{\mathrm{layers}} = 126$, hidden size $d_{\mathrm{model}} = 16{,}384$, FFN size $d_{\mathrm{ff}} = 53{,}248$,

FP16 communication $c_{\text{param}} = 2$, and LoRA rank $r = 32$. If LoRA is applied only to W_q and W_v, the upload cost is $C^{\uparrow}_{\text{LoRA}(q,v)} = c_{\text{param}} L_{\text{layers}} 2r(d_{\text{model}} + d_{\text{model}}) = 528{,}482{,}304$ bytes ≈ 504.0 MiB, or 1008.0 MiB including download, which is $\approx 194\times$ larger than Semantic Consensus. If LoRA is applied to all attention projections and the MLP, the upload cost is $C^{\uparrow}_{\text{LoRA(attn+MLP)}} = c_{\text{param}} L_{\text{layers}} r\{4(d_{\text{model}} + d_{\text{model}}) + 3(d_{\text{model}} + d_{\text{ff}})\} \approx 2{,}741{,}501{,}952$ bytes ≈ 2614.5 MiB, or 5229.0 MiB including download, which is $\approx 1006\times$ larger. Thus, for a 405B-scale model, Semantic Consensus reduces total per-round communication by roughly $201\times$–$1046\times$ while remaining independent of model size. Additional communication examples are provided in Appendix. E.6. Our communication analysis characterizes theoretical scaling with the number of clients, whereas our end-to-end experiments evaluate federations of $K = 5$ and $K = 10$ clients.

4.2 Convergence

FedCoFiT alternates between local updates on private data and local updates on a pseudo-labeled public prompt set. Since pseudo-labels may be noisy and the public prompt distribution may differ from a reference prompt distribution, each client performs SGD on a time-varying surrogate objective whose gradient is a biased approximation of a fixed reference objective gradient. In the non-IID setting, client distributions differ, introducing additional variability through client heterogeneity. We provide a descent-to-stationarity guarantee where the bias is controlled by covariate shift, pseudo-label noise, and finite-sample concentration from using B_{pub} public examples per step.

Let there be K clients. Client i draws i.i.d. data from $\mathcal{P}_i(x, y) = \mathcal{P}_{i,X}(x)\mathcal{P}_i(y \mid x)$. Define the global mixture $\mathcal{P}(x, y) = K^{-1} \sum_{i=1}^{K} \mathcal{P}_i(x, y)$, with marginal $\mathcal{P}_X$ and conditional $\mathcal{P}(y \mid x)$. The private objectives are $F_i(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{P}_i}[\ell(\theta; x, y)]$ and $F(\theta) = K^{-1} \sum_{i=1}^{K} F_i(\theta)$.

At communication round q, public prompts follow $Q_X^{(q)}$. Given prompt x, the pseudo-labeling mechanism induces $\widetilde{\mathcal{P}}^{(q)}(\cdot \mid x)$ over pseudo-labels $\tilde{Y}$. Since the public pool is unlabeled, define the reference public objective as $G(\theta) = \mathbb{E}_{x\sim\mathcal{P}_X} \mathbb{E}_{y\sim\mathcal{P}(\cdot|x)}[\ell(\theta; x, y)]$. For the equal-weight objective used in FedCoFiT, define the round-q surrogate objective as

$$\Phi^{(q)}(\theta) = \frac{1}{2} F(\theta) + \frac{1}{2} G(\theta; \widetilde{\mathcal{P}}^{(q)}, Q_X^{(q)}), \tag{1}$$

where $G(\theta; \widetilde{\mathcal{P}}^{(q)}, Q_X^{(q)}) = \mathbb{E}_{x\sim Q_X^{(q)}} \mathbb{E}_{\tilde{Y}\sim\widetilde{\mathcal{P}}^{(q)}(\cdot|x)}[\ell(\theta; x, \tilde{Y})]$. We measure stationarity with respect to $\bar{\Phi}(\theta) = \frac{1}{2} F(\theta) + \frac{1}{2} G(\theta)$.

Assumption 1 (Smoothness and stochastic gradients)**.** Assume: (i) $\bar{\Phi}$ is Λ-smooth along $\{\theta^t\}_{t=0}^{T}$; and (ii) for each SGD step t, $\mathbb{E}[g^t \mid \theta^t, q(t)] = \nabla\Phi^{(q(t))}(\theta^t)$ and $\mathbb{E}\big[\|g^t - \nabla\Phi^{(q(t))}(\theta^t)\|^2 \mid \theta^t, q(t)\big] \le \sigma^2 + \frac{1}{4}\zeta^2$. Here σ^2 captures minibatch noise, and ζ^2 captures client heterogeneity.

Assumption 2 (Client heterogeneity)**.** Along the iterates $\{\theta^t\}_{t=0}^{T}$, assume $K^{-1} \sum_{i=1}^{K} \|\nabla F_i(\theta^t) - \nabla F(\theta^t)\|^2 \le \zeta^2$ for all $t \in \{0, \dots, T\}$.

Assumption 3 (Public shift, pseudo-label noise, and finite public sampling)**.** Assume: (i) for each round q, $\mathrm{KL}(\mathcal{P}_X \| Q_X^{(q)}) \le \tau_q$ and $\tau = \max_q \tau_q$; (ii) for all (θ, x, y), $\|\nabla_\theta \ell(\theta; x, y)\| \le G_{\text{grad}}$; (iii) for each round q, $\eta_q = \mathbb{E}_{x\sim Q_X^{(q)}}[\mathrm{TV}(\mathcal{P}(\cdot \mid x), \widetilde{\mathcal{P}}^{(q)}(\cdot \mid x))]$ and $\eta_{\text{pl}} = \max_q \eta_q$; and (iv) whenever a client forms a public-gradient estimate at step t, it uses B_{pub} i.i.d. samples $\{(x_k, \tilde{Y}_k)\}_{k=1}^{B_{\text{pub}}}$ with $x_k \sim Q_X^{(q(t))}$ and $\tilde{Y}_k \sim \widetilde{\mathcal{P}}^{(q(t))}(\cdot \mid x_k)$, independent of θ^t conditional on the past.

Define the high-probability public-gradient gap

$$\Delta_T(\rho) = G_{\text{grad}} \sqrt{2\tau} + 2G_{\text{grad}} \eta_{\text{pl}} + 4G_{\text{grad}} \sqrt{\frac{2}{B_{\text{pub}}} \left(d_\theta \log 5 + \log \frac{2T}{\rho} \right)}. \tag{2}$$

Lemma 1 (Surrogate–reference public gradient gap)**.** *Under Assumption 3, with probability at least $1 - \rho$ over the sampling of public prompts and pseudo-labels across the T SGD steps, $\|\nabla_\theta G_t(\theta^t) - \nabla_\theta G(\theta^t)\| \le \Delta_T(\rho)$ for all $t \in \{0, \dots, T-1\}$, where $G_t(\theta)$ is the empirical public objective formed from the B_{pub} samples in Assumption 3(4).*

Theorem 1 (Stationarity under shift, pseudo-label noise, and non-IID heterogeneity)**.** *Let $\{\theta^t\}_{t=0}^T$ be produced by SGD, $\theta^{t+1} = \theta^t - \eta g^t$ for $t = 0, \dots, T-1$. Under Assumptions 1, 2, and 3, choose $\eta \le 1/(4\Lambda)$ and fix $\rho \in (0,1)$. Then, on the event of Lemma 1,*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\big[\|\nabla\bar{\Phi}(\theta^t)\|^2\big] \le \frac{4\big(\bar{\Phi}(\theta^0) - \bar{\Phi}^\star\big)}{\eta T} + 2\Lambda\eta\big(\sigma^2 + \frac{1}{4}\zeta^2\big) + \frac{3}{4}\Delta_T(\rho)^2, \tag{3}$$

where $\bar{\Phi}^\star = \inf_\theta \bar{\Phi}(\theta)$ and the expectation is over SGD minibatch randomness.

Remark. The full proof is available in the appendix. A. The term $\zeta^2/4$ captures client heterogeneity under the equal-weight global SGD abstraction. An analysis with multiple explicit local steps between communication rounds typically introduces additional drift terms depending on the number of local steps and the heterogeneity level.

4.3 Privacy Analysis

Federated LoRA-based fine-tuning methods such as FlexLoRA and FLoRA keep raw client data local while aggregating adapter-side information at the server. Prior work shows that PEFT can be combined with differential privacy through mechanisms such as DP-SGD [Liu et al., 2025]. Motivated by this, we introduce DP-FedCoFiT and show that FedCoFiT admits a record-level differential privacy guarantee when each client's local LoRA fine-tuning mechanism is differentially private.

Let ϕ_i denote the trainable LoRA parameters of client i, with the backbone frozen. At communication round q, client i applies a randomized local training mechanism $\phi_i^{(q)} \leftarrow \mathcal{A}_{i,q}(D_i, H^{(q-1)})$, where $H^{(q-1)}$ is the transcript up to round $q-1$. Assume that, for every fixed history $H^{(q-1)}$, $\mathcal{A}_{i,q}$ is $(\varepsilon_{i,q}, \delta_{i,q})$-differentially private with respect to D_i. After local training, client i generates responses on $D_{\text{pub}} = \{x_j\}_{j=1}^M$ and releases $\mathcal{R}_i^{(q)} = \{\hat{y}_{i,j}^{(q)}\}_{j=1}^M$, where $\hat{y}_{i,j}^{(q)} \leftarrow \mathcal{M}_i(x_j; \phi_i^{(q)})$.

Definition 1 (Neighboring datasets)**.** For a fixed client i, two datasets D_i and D_i' are neighboring if they differ in one training example.

Theorem 2. *Suppose that, for each communication round q, the client-side mechanism $\mathcal{A}_{i,q}(\cdot, H^{(q-1)})$ is $(\varepsilon_{i,q}, \delta_{i,q})$-differentially private with respect to D_i. Then the released public-prompt responses $\mathcal{R}_i^{(q)}$ are also $(\varepsilon_{i,q}, \delta_{i,q})$-differentially private with respect to D_i. Moreover, the overall multi-round mechanism induced by client i over Q communication rounds is $(\sum_{q=1}^Q \varepsilon_{i,q}, \sum_{q=1}^Q \delta_{i,q})$-differentially private with respect to D_i.*

Proof sketch. At communication round q, the local LoRA fine-tuning mechanism is $(\varepsilon_{i,q}, \delta_{i,q})$-DP by assumption. The released responses $\mathcal{R}_i^{(q)}$ depend only on the resulting private model state and the public prompts, so they are DP by post-processing. Server-side embedding, clustering, representative selection, and broadcast are also post-processing. Adaptive composition across communication rounds gives the stated bound. □

Details are given in Appendix B. In DP-FedCoFiT, sharing responses on the public prompt set does not break the privacy guarantee, as those responses are outputs of a differentially private client-side fine-tuning mechanism. This guarantee is conditional on differentially private local fine-tuning and does not imply that standard FedCoFiT is differentially private.

5 Empirical Evaluation

We evaluate FedCoFiT along three axes: performance, communication, and efficiency. In all experiments, clients fine-tune locally on private data and collaborate through semantic consensus rounds using a shared public prompt set D_{pub} that was sampled from the dataset and excluded from the training data (the effect of this shared public prompt set size and distribution is studied in Appendix E.1 and E.2). Consequently, the baselines and FedCoFiT receive identical private labeled data. FedCoFiT uses only the input prompts from the reserved examples and never accesses their ground-truth responses during semantic-consensus training. We report task performance on held-out evaluation benchmarks, total communication volume measured as gigabytes uploaded and downloaded between

clients and server across all rounds, and end-to-end wall-clock runtime and energy consumption. All methods are run with the same number of clients and rounds, and we report the mean score across clients. Full training hyperparameters, hardware details, and additional results are provided in Appendix C. In the empirical evaluation, we consider three adaptation regimes: instruction compliance, knowledge generalization, and chat quality. For instruction compliance, clients fine-tune on Dolly-15k Zhang et al. [2024], and Alpaca Taori et al. [2023]. We evaluate instruction-following behaviour using IFEval benchmark Zhou et al. [2023]. For knowledge generalization, clients fine-tune on Wizard Luo et al. [2025] and OpenOrca Lian et al. [2023], we evaluate on MMLU benchmark Hendrycks et al. [2021]. For chat quality, clients fine-tune on ShareGPT Chiang et al. [2023] and UltraChat Ding et al. [2023] and evaluate on MT-Bench Zheng et al. [2023]. For each setting, we compare FedCoFiT against parameter aggregation baselines (e.g., FLoRA, FedIT, FlexLoRA) under matched federation budgets. We report final scores over rounds, together with total communication bytes (upload and download). To connect communication to practical cost, we additionally report wall-clock runtime and energy usage measured with Lamarr Energy Tracker Fischer [2026]. Finally, we perform ablations to study the effect of design choices in public prompt size $|D_{\text{pub}}|$, public prompt distribution, and the semantic encoder [1].

Experiment results: We first evaluate the performance of FedCoFiT under a homogeneous LoRA setting, where all clients use the same LoRA rank $r = 32$. For TinyLlama, LLaMA, and LLaMA2, we use 10, 10, and 5 clients, respectively, with a micro-batch size of 16. We additionally use a public prompt budget $M = 500$ prompts sampled from the fine-tuning dataset. Table 1 compares FedCoFiT with parameter-aggregation baselines across the three regimes. We also report the total communication in GB. Two trends are consistent across all three foundation models. First, FedCoFiT matches or exceeds the strongest federated baseline in task quality. On TinyLlama, FedCoFiT attains the best IFEval score on Dolly (51.19) and competitive performance across MMLU and MT-Bench benchmarks. On LLaMA-7B, it achieves the top score on IFEval (54.36 Dolly), MMLU (35.78 OpenOrca), and on LLaMA2-13B, it achieved the highest score across most settings (e.g., MMLU Wizard 45.02). These results indicate that behavior-level consensus provides learning signals that are effective. Second, these quality levels are achieved with orders of magnitude lower communication. Across all experiments and baselines, FedCoFiT communication budget is in the range of $\approx 0.001 - 0.007GB$, where parameter aggregation-based methods require substantially larger transfers, e.g., FLoRA ranges from $\approx 1.98 - 21GB$ and FlexLoRA from $\approx 0.36 - 3.75GB$. Concretely, on LLaMA-7B instruction tuning (Dolly), FedCoFiT uses $0.006GB$ versus $3.13GB$ (FedIT), $3.75GB$ (FlexLoRA), and $20.63GB$ (FLoRA), while achieving the highest IFEval score. Similar gaps hold for MMLU and MT-Bench, demonstrating that FedCoFiT delivers comparable accuracy at a much smaller federation budget. Overall, FedCoFiT is consistently competitive in performance while minimizing communication, and this communication reduction translates into practical system benefits. For example, FedCoFiT reduces end-to-end runtime, energy consumption, and estimated CO_2 emissions relative to parameter aggregation (Figure 2). As model size grows and adapter updates become more expensive to transmit, these efficiency gains become increasingly pronounced.

Heterogeneous LoRA Ranks: In many federated deployments, clients cannot be assumed to use identical PEFT configurations due to different compute, memory, or personalization needs. To evaluate robustness under such heterogeneity, we consider a heterogeneous LoRA-rank setting in which clients fine-tune the same TinyLlama backbone but use different LoRA ranks. In this setting, we report task scores. Let r_i denote the LoRA rank used by client i. We assign ranks across clients by sampling from a small set of representative ranks $r_i \in \{8, 16, 32\}$ and distributing across the 10 clients. All other hyperparameters (target modules, learning rate, batch size, rounds, and public prompt budget) are kept identical to the homogeneous setting. Table 2 reports performance under heterogeneous ranks across instruction following (IFEval), knowledge generalization (MMLU), and chat quality (MT-Bench). FedIT is marked as N/A because it does not support heterogeneous LoRA ranks. Parameter-aggregation baselines such as FLoRA and FlexLoRA baselines degrade under rank heterogeneity because adapter shapes differ across clients. In contrast, FedCoFiT aggregates client behaviour in prediction space and is therefore agnostic to rank choices. Empirically, FedCoFiT remains stable under heterogeneous ranks and achieves the best or competitive scores across tasks, demonstrating that behaviour-level consensus naturally supports heterogeneous PEFT configurations.

[1] Code available at https://github.com/kamp-lab-mlai/FedCoFiT

Table 1: Comparison of Semantic Consensus FedCoFiT with baselines across instruction compliance (IFEval), knowledge generalization (MMLU), and chat quality (MT-Bench). GB denotes the cumulative communication volume (in gigabytes) summed across all clients and all rounds.

Foundation	Method	IFEval ↑				MMLU ↑				MT-Bench ↑			
		Dolly-15K		Alpaca		Wizard		OpenOrca		ShareGPT		UltraChat	
		Score	GB	Score	GB	Score	GB	Score	GB	Score	GB	Score	GB
TinyLlama	Centralized (LoRA)	49.32	–	52.03	–	21.32	–	21.36	–	2.74	–	2.61	–
	FedIT	43.82	0.9	49.21	0.9	22.19	0.54	21.38	0.54	2.53	0.18	2.65	0.18
	FLoRA	50.26	11.35	51.10	11.35	22.25	3.96	22.84	3.96	2.81	1.98	3.10	1.98
	FlexLoRA	50.73	1.2	50.45	1.2	21.08	0.72	21.80	0.72	2.73	0.36	2.84	0.36
	FedCoFiT (ours)	51.19	0.007	50.71	0.002	22.60	0.005	22.73	0.001	2.79	0.002	2.97	0.002
LLaMA	Centralized (LoRA)	54.18	–	55.17	–	34.28	–	35.19	–	3.34	–	3.25	–
	FedIT	50.37	3.13	50.92	3.13	31.46	1.88	31.74	1.88	3.29	0.625	3.17	0.625
	FLoRA	53.29	20.63	54.91	20.63	34.92	13.75	35.21	13.75	3.41	6.875	3.31	6.875
	FlexLoRA	52.61	3.75	53.79	3.75	34.85	2.5	34.62	2.5	3.36	1.25	3.27	1.25
	FedCoFiT (ours)	54.36	0.006	54.83	0.004	34.89	0.004	35.78	0.001	3.45	0.002	3.34	0.002
LLaMA2	Centralized (LoRA)	58.71	–	59.24	–	43.27	–	43.85	–	5.86	–	6.13	–
	FedIT	52.38	2.62	53.29	2.62	43.29	1.57	42.06	1.57	5.93	0.52	6.03	0.52
	FLoRA	58.49	17.30	60.25	17.30	44.57	11.53	44.52	11.53	6.03	5.77	6.21	5.77
	FlexLoRA	57.24	3.14	59.50	3.14	44.29	2.1	43.96	2.1	5.97	1.05	6.14	1.05
	FedCoFiT (ours)	57.96	0.006	60.31	0.004	45.02	0.004	44.28	0.001	6.12	0.002	6.19	0.003

Table 2: TinyLlama results under **heterogeneous LoRA ranks**. Clients use ranks $r_i \in \{8, 16, 32\}$ distributed across K=10 clients.

Method	IFEval ↑		MMLU ↑		MT-Bench ↑	
	Dolly-15K	Alpaca	Wizard	OpenOrca	ShareGPT	UltraChat
FedIT	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*
FLoRA	49.87	50.26	20.93	21.04	2.58	2.61
FlexLoRA	49.38	50.12	20.19	20.93	2.46	2.57
FedCoFiT (ours)	51.20	50.39	21.34	22.08	2.74	2.84

Statistical Heterogeneity FedCoFiT remains robust under strongly non-IID client data (Dirichlet $\alpha = 0.1$), outperforming FLoRA by 2.52 and local training by 6.55 IFEval points (Appendix E.3).

FedCoFiT Efficiency Evaluation Beyond accuracy and communication, we evaluate the efficiency of FedCoFiT in terms of end-to-end wall clock runtime, energy consumption, and estimated CO_2 emissions. We measure runtime, aggregate energy in kWh and CO_2 in kg using the Lamarr Energy tracker Fischer [2026]. We report end-to-end measurements, including local fine-tuning, public-prompt generation, server-side consensus, and communication overhead. To ensure a fair comparison, all methods are run with the same number of clients and communication rounds, and matched local training budgets. Hardware and hyperparameter details are provided in Appendix C. Figure 2 shows that FedCoFiT consistently achieves the lowest wall-clock time, energy consumption, and estimated CO_2 emissions on Dolly-15K using LLaMA-7B. Compared with FedCoFiT, FedIT, FLoRA, and FlexLoRA incur $1.43\times$, $2.47\times$, and $1.58\times$ higher runtime, respectively, with corresponding energy overheads of $1.41\times$, $2.16\times$, and $1.44\times$. Similar trends are observed for CO_2 emissions. Notably, the largest efficiency gains are observed against FLoRA, suggesting that the reduced communication and synchronization overhead of FedCoFiT translates into substantial end-to-end computational and environmental savings, rather than merely reducing communication volume.

Medical Domain Evaluation To evaluate FedCoFiT in a specialized application domain, we consider federated fine-tuning of Qwen3-0.6B [Yang et al., 2025] using the Medical Meadow Medical Flashcards [Han et al., 2023]. We distribute the training data across $K = 10$ clients and compare independent local training, FLoRA, and FedCoFiT under matched experimental settings (details in Appendix C). The models are fine-tuned using medical questions paired with free-text answers. Thus, FedCoFiT constructs semantic consensus over open-ended generated medical responses. We evaluate the resulting models on three held-out medical benchmarks: MedQA [Jin et al., 2021], MedMCQA [Pal et al., 2022], and PubMedQA [Jin et al., 2019]. As shown in Table 3, FedCoFiT consistently achieves the highest accuracy across all three benchmarks, improving the average accuracy by 1.71 and 3.68 percentage points over FLoRA and local training, respectively. Compared with FLoRA, the largest improvements are observed on PubMedQA (+2.14 points) and MedMCQA (+1.98 points), while MedQA also improves by 1.02 point.

Table 3: Medical-domain accuracy (%) using Qwen3-0.6B fine-tuned on Medical Meadow Medical Flashcards. Results are reported as mean ± standard deviation over three independent runs. The average column reports the mean accuracy across the three benchmarks.

Method	MedQA	MedMCQA	PubMedQA	Avg.
Local only	19.78 ± 0.05	20.32 ± 0.02	20.69 ± 0.01	20.26
FLoRA	22.14 ± 0.04	21.75 ± 0.01	22.79 ± 0.08	22.23
FedCoFiT (ours)	$\mathbf{23.16 \pm 0.07}$	$\mathbf{23.73 \pm 0.04}$	$\mathbf{24.93 \pm 0.02}$	**23.94**

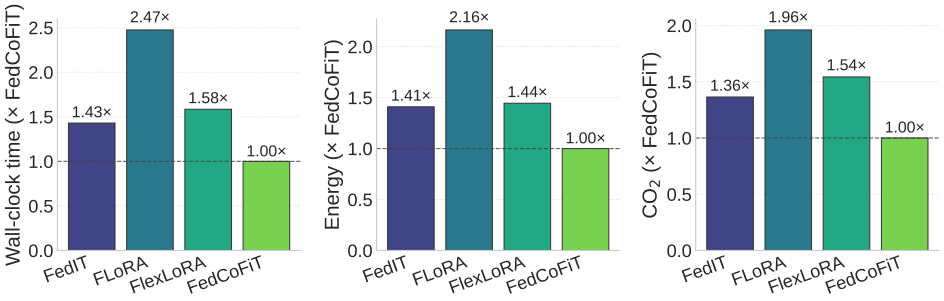

Figure 2: Relative efficiency comparison on Dolly-15K using LLaMA-7B

Ablations and Robustness: FedCoFiT is stable for moderate public-prompt budgets but degrades when prompts are too few (Appendix E.1), and remains robust to instruction-style distribution shifts while degrading under stronger chat-format shift (Appendix E.2). Further ablations study the semantic encoder, consensus-selection rule, evaluation on a recent model family, and conservative fairness control (Appendices E.4,E.5,E.7 and E.8).

6 Discussion and Conclusion

This work introduces Semantic Consensus as a fundamental principle for federated fine-tuning of large language models, in which collaboration is mediated through shared model behaviours rather than shared parameters. FedCoFiT shows that clients can collaborate by exchanging responses to public prompts, allowing the server to form consensus pseudo-labels without sharing weights, gradients, or adapters. This directly reduces communication, supports heterogeneous architectures, and remains applicable when clients have different LoRA configurations.

The theoretical analysis shows that FedCoFiT communication cost depends on the number of public prompts and response length, not model size, giving a fundamentally different scaling law from LoRA or parameter aggregation approaches. Furthermore, we theoretically establish high-probability convergence to a stationary neighborhood under standard assumptions and identify the main factors determining the residual optimization error. Our privacy analysis shows that if local client fine-tuning is differentially private, then the released public-prompt responses and server-side consensus steps preserve record-level differential privacy through post-processing and composition.

Empirically, FedCoFiT matches or exceeds strong federated LoRA baselines across instruction following, knowledge generalization, and chat-quality tasks while using orders of magnitude less communication. It also remains effective under heterogeneous LoRA ranks, where parameter aggregation is difficult or unsupported. As shown by our ablation studies, the main limitation of FedCoFiT is dependence on the public prompt set and semantic clustering quality, as performance can degrade when prompts are too few or poorly aligned with the target task. Overall, Semantic Consensus suggests that behavior-level sharing is a promising alternative to parameter aggregation for federated adaptation of large and heterogeneous LLMs. A limitation of semantic consensus is that semantic agreement does not necessarily imply factual correctness. When multiple clients generate consistent but incorrect responses, the majority cluster may reinforce shared hallucinations. Future work could mitigate this risk through abstention mechanisms, confidence calibration, or external verification of consensus responses.

Acknowledgements

This work received support from the Cancer Research Center Cologne Essen (CCCE). The authors declare no competing interests.

References

Martín 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*, pages 308–318, 2016. doi: 10.1145/2976749.2978318.

Amr Abourayya, Jens Kleesiek, Kanishka Rao, Erman Ayday, Bharat Rao, Geoffrey I Webb, and Michael Kamp. Little is enough: Boosting privacy by sharing only hard labels in federated semi-supervised learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pages 15293–15301, 2025.

Gholamali Aminian, Mahed Abroshan, Mohammad Mahdi Khalili, Laura Toni, and Miguel Rodrigues. An information-theoretical approach to semi-supervised learning under covariate-shift. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 7433–7449. PMLR, 2022. URL https://proceedings.mlr.press/v151/aminian22a.html.

Jiamu Bai, Daoyuan Chen, Bingchen Qian, Liuyi Yao, and Yaliang Li. Federated fine-tuning of large language models under heterogeneous tasks and client resources. *Advances in Neural Information Processing Systems*, 37:14457–14483, 2024.

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

Xiangyi Chen, Zhiwei Steven Wu, and Mingyi Hong. Understanding gradient clipping in private sgd: A geometric perspective. In *Advances in Neural Information Processing Systems*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/9ecff5455677b38d19f49ce658ef0608-Abstract.html.

Wei-Lin Chiang, Zhuohan Li, Ziqing Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3):6, 2023.

Yae Jee Cho, Luyang Liu, Zheng Xu, Aldi Fahrezi, and Gauri Joshi. Heterogeneous lora for federated fine-tuning of on-device foundation models. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pages 12903–12913, 2024.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3029–3051, 2023.

Raphael Fischer. Ground-truthing ai energy consumption: validating codecarbon against external measurements. *it-Information Technology*, 2026.

Saeed Ghadimi and Guanghui Lan. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013. doi: 10.1137/120880811.

Tianyu Han, Lisa C Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexei Figueroa, Alexander Löser, Daniel Truhn, and Keno K Bressem. Medalpaca–an open-source collection of medical conversational ai models and training data. *arXiv preprint arXiv:2304.08247*, 2023.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Jiayu Huang, Xiaohu Wu, Tiantian He, and Qicheng Lao. Stabilized fine-tuning with lora in federated learning: Mitigating the side effect of client size and rank via the scaling factor. *arXiv preprint arXiv:2603.08058*, 2026.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421, 2021.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. Pubmedqa: A dataset for biomedical research question answering. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 2567–2577, 2019.

Michael Kamp. *Black-Box Parallelization for Machine Learning*. PhD thesis, Rheinische Friedrich-Wilhelms-Universität Bonn, Universitäts-und Landesbibliothek Bonn, 2019.

Michael Kamp, Sebastian Bothe, Mario Boley, and Michael Mock. Communication-efficient distributed online learning with kernels. In *ECMLPKDD*, pages 805–819. Springer, 2016.

Michael Kamp, Linara Adilova, Joachim Sicking, Fabian Hüger, Peter Schlicht, Tim Wirtz, and Stefan Wrobel. Efficient decentralized deep learning by dynamic model averaging. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 393–409. Springer, 2018.

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

Jabin Koo, Minwoo Jang, and Jungseul Ok. Towards robust and efficient federated low-rank adaptation with heterogeneous clients. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 416–429, 2025.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020a.

Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*, 2020b. URL https://openreview.net/forum?id=HJxNAnVtDS.

Wing Lian, Bleys Goodson, Eugene Pentland, Austin Cook, Chanvichet Vong, and "Teknium". Openorca: An open dataset of gpt augmented flan reasoning traces. https://huggingface.co/datasets/Open-Orca/OpenOrca, 2023.

Hongbin Liu, Lun Wang, Om Thakkar, Abhradeep Thakurta, and Arun Narayanan. Differentially private parameter-efficient fine-tuning for large asr models. In *Interspeech 2025*, 2025.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jian-Guang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, Yansong Tang, et al. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. In *International Conference on Learning Representations*, volume 2025, pages 49573–49609, 2025.

Xinge Ma, Jiangming Liu, Jin Wang, and Xuejie Zhang. Fedid: Federated interactive distillation for large-scale pretraining language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8566–8577, 2023.

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

Nagarajan Natarajan, Inderjit S. Dhillon, Pradeep K. Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in Neural Information Processing Systems*, pages 1196–1204, 2013. URL https://papers.nips.cc/paper/5073-learning-with-noisy-labels.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on health, inference, and learning*, pages 248–260. PMLR, 2022.

Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(35):985–1005, 2007. URL https://www.jmlr.org/papers/v8/sugiyama07a.html.

Jingwei Sun, Ziyue Xu, Hongxu Yin, Dong Yang, Daguang Xu, Yudong Liu, Zhixu Du, Yiran Chen, and Holger Roth. FedBPT: Efficient federated black-box prompt tuning for large language models. In *International Conference on Machine Learning (ICML)*, July 2024.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford alpaca: An instruction-following llama model, 2023.

Ziyao Wang, Zheyu Shen, Yexiao He, Guoheng Sun, Hongyi Wang, Lingjuan Lyu, and Ang Li. Flora: Federated fine-tuning large language models with heterogeneous low-rank adaptations. *Advances in Neural Information Processing Systems*, 37:22513–22533, 2024.

Colin Wei, Kendrick Shen, Yining Chen, and Tengyu Ma. Theoretical analysis of self-training with deep networks on unlabeled data. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=rC8sJ4i6kaH.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

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

Zikai Zhang, Rui Hu, and Jiahao Xu. Heterogeneous federated fine-tuning with parallel one-rank adaptation. In *The International Conference on Learning Representations (ICLR)*, 2026.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

A Convergence Proofs and Supporting Lemmas

A.1 Proof of Lemma 1

Fix a parameter $\theta \in \mathbb{R}^{d_\theta}$ and a communication round q. Define the population risks using the global reference distribution $\mathcal{P}$:

$$R_{\mathcal{P}}(\theta) := \mathbb{E}_{x\sim\mathcal{P}_X}\mathbb{E}_{y\sim\mathcal{P}(\cdot|x)}[\ell(\theta;x,y)] \quad (= G(\theta)),$$

$$R_Q^{(q)}(\theta) := \mathbb{E}_{x\sim Q_X^{(q)}}\mathbb{E}_{y\sim\mathcal{P}(\cdot|x)}[\ell(\theta;x,y)], \qquad \widetilde{R}_Q^{(q)}(\theta) := \mathbb{E}_{x\sim Q_X^{(q)}}\mathbb{E}_{\tilde{Y}\sim\widetilde{\mathcal{P}}^{(q)}(\cdot|x)}[\ell(\theta;x,\tilde{Y})].$$

For any θ and x, define $v_\theta(x) := \mathbb{E}_{y\sim\mathcal{P}(\cdot|x)}[\nabla_\theta\ell(\theta;x,y)]$ and $\widetilde{v}_\theta^{(q)}(x) := \mathbb{E}_{\tilde{Y}\sim\widetilde{\mathcal{P}}^{(q)}(\cdot|x)}[\nabla_\theta\ell(\theta;x,\tilde{Y})]$. By Assumption 3(2), $\|v_\theta(x)\| \le G_{\mathrm{grad}}$ and $\|\widetilde{v}_\theta^{(q)}(x)\| \le G_{\mathrm{grad}}$.

A. Covariate shift bound.

Lemma 2 (Pinsker). *For distributions μ, ν on the same space,* $\mathrm{TV}(\mu,\nu) \le \sqrt{\frac{1}{2}\,\mathrm{KL}(\mu\|\nu)}$.

Lemma 3 (Expectation shift under total variation). *Let $h : \mathcal{X} \to \mathbb{R}^{d_\theta}$ be measurable and satisfy $\|h(x)\| \le B$ for all x. Then $\|\mathbb{E}_{x\sim Q}[h(x)] - \mathbb{E}_{x\sim P}[h(x)]\| \le 2B\,\mathrm{TV}(P,Q)$.*

Lemma 4 (Covariate shift gradient bound). *If $\mathrm{KL}(\mathcal{P}_X\|Q_X^{(q)}) \le \tau_q$ and $\|v_\theta(x)\| \le G_{\mathrm{grad}}$, then*

$$\|\nabla R_Q^{(q)}(\theta) - \nabla R_{\mathcal{P}}(\theta)\| \le G_{\mathrm{grad}}\sqrt{2\tau_q} \le G_{\mathrm{grad}}\sqrt{2\tau}.$$

Proof. Since $\nabla R_Q^{(q)}(\theta) = \mathbb{E}_{Q_X^{(q)}}[v_\theta(x)]$ and $\nabla R_{\mathcal{P}}(\theta) = \mathbb{E}_{\mathcal{P}_X}[v_\theta(x)]$, Lemma 3 with $h = v_\theta$ and $B = G_{\mathrm{grad}}$ gives $\|\nabla R_Q^{(q)}(\theta) - \nabla R_{\mathcal{P}}(\theta)\| \le 2G_{\mathrm{grad}}\,\mathrm{TV}(\mathcal{P}_X, Q_X^{(q)})$. Pinsker and $\mathrm{KL}(\mathcal{P}_X\|Q_X^{(q)}) \le \tau_q$ imply $\mathrm{TV}(\mathcal{P}_X, Q_X^{(q)}) \le \sqrt{\tau_q/2}$, yielding the claim. □

B. Pseudo-label noise bound.

Lemma 5 (Pseudo-label noise gradient bound). *Under Assumption 3, for all θ and each communication round q,*

$$\|\nabla\widetilde{R}_Q^{(q)}(\theta) - \nabla R_Q^{(q)}(\theta)\| \le 2G_{\mathrm{grad}}\eta_q \le 2G_{\mathrm{grad}}\eta_{\mathrm{pl}}.$$

Proof. For fixed θ and x, $\widetilde{v}_\theta^{(q)}(x) - v_\theta(x) = \mathbb{E}_{\tilde{Y}\sim\widetilde{\mathcal{P}}^{(q)}(\cdot|x)}[\nabla_\theta\ell(\theta;x,\tilde{Y})] - \mathbb{E}_{y\sim\mathcal{P}(\cdot|x)}[\nabla_\theta\ell(\theta;x,y)]$. Since $\|\nabla_\theta\ell(\theta;x,y)\| \le G_{\mathrm{grad}}$, the total-variation expectation bound gives $\|\widetilde{v}_\theta^{(q)}(x) - v_\theta(x)\| \le 2G_{\mathrm{grad}}\mathrm{TV}(\mathcal{P}(\cdot\mid x), \widetilde{\mathcal{P}}^{(q)}(\cdot\mid x))$. Taking $\mathbb{E}_{x\sim Q_X^{(q)}}$ yields the result. □

C. Finite-sample concentration for the empirical public gradient. At a public-gradient evaluation using communication round q, let $\{(x_k, \tilde{Y}_k)\}_{k=1}^{B_{\mathrm{pub}}}$ be i.i.d. with $x_k \sim Q_X^{(q)}$ and $\tilde{Y}_k \sim \widetilde{\mathcal{P}}^{(q)}(\cdot\mid x_k)$. Define $\widehat{g}^{(q)}(\theta) := B_{\mathrm{pub}}^{-1}\sum_{k=1}^{B_{\mathrm{pub}}}\nabla_\theta\ell(\theta;x_k,\tilde{Y}_k)$, so $\mathbb{E}[\widehat{g}^{(q)}(\theta)] = \nabla\widetilde{R}_Q^{(q)}(\theta)$.

Lemma 6 (Vector Hoeffding via a 1/2-net). *Fix θ and $\rho \in (0,1)$. If $\|\nabla_\theta\ell(\theta;x,y)\| \le G_{\mathrm{grad}}$, then with probability at least $1-\rho$,*

$$\|\widehat{g}^{(q)}(\theta) - \nabla\widetilde{R}_Q^{(q)}(\theta)\| \le 4G_{\mathrm{grad}}\sqrt{\frac{2}{B_{\mathrm{pub}}}\left(d_\theta\log 5 + \log\frac{2}{\rho}\right)}.$$

Proof. Construct a 1/2-net $\mathcal{N}$ of $S^{d_\theta-1}$ with $|\mathcal{N}| \le 5^{d_\theta}$. For each $u \in \mathcal{N}$, apply scalar Hoeffding to $\langle u, \widehat{g}^{(q)}(\theta) - \nabla\widetilde{R}_Q^{(q)}(\theta)\rangle$ and union bound over $\mathcal{N}$. Lift from net control to full norm control by the standard approximation argument. □

D. Combine and union bound over T steps. Fix step t and write $q = q(t)$. By the triangle inequality,

$$\|\widehat{g}^{(q)}(\theta^t) - \nabla R_{\mathcal{P}}(\theta^t)\| \le \|\widehat{g}^{(q)}(\theta^t) - \nabla \widetilde{R}_Q^{(q)}(\theta^t)\| + \|\nabla \widetilde{R}_Q^{(q)}(\theta^t) - \nabla R_Q^{(q)}(\theta^t)\| + \|\nabla R_Q^{(q)}(\theta^t) - \nabla R_{\mathcal{P}}(\theta^t)\|.$$

Lemma 6 with ρ_t, Lemma 5, and Lemma 4 imply that with probability at least $1 - \rho_t$,

$$\|\widehat{g}^{(q)}(\theta^t) - \nabla R_{\mathcal{P}}(\theta^t)\| \le G_{\mathrm{grad}}\sqrt{2\tau} + 2G_{\mathrm{grad}}\eta_{\mathrm{pl}} + 4G_{\mathrm{grad}}\sqrt{\frac{2}{B_{\mathrm{pub}}}\left(d_\theta \log 5 + \log\frac{2}{\rho_t}\right)}.$$

Set $\rho_t = \rho/T$ and union bound over $t = 0, \dots, T-1$ to obtain, with probability at least $1-\rho$, $\|\widehat{g}^{(q(t))}(\theta^t) - \nabla R_{\mathcal{P}}(\theta^t)\| \le \Delta_T(\rho)$ for all t. Identifying $\nabla_\theta G_t(\theta^t)$ with $\widehat{g}^{(q(t))}(\theta^t)$ and $\nabla_\theta G(\theta^t)$ with $\nabla R_{\mathcal{P}}(\theta^t)$ proves Lemma 1. □

A.2 A Supporting Lemma: Heterogeneity Contributes Extra Variance

Lemma 7 (Client sampling variance bound)**.** *Fix $\theta \in \mathbb{R}^{d_\theta}$ and suppose a client index I is sampled uniformly from $\{1, \dots, K\}$. Let $\widehat{\nabla} F_I(\theta)$ be a possibly minibatched unbiased estimator of $\nabla F_I(\theta)$ satisfying $\mathbb{E}[\widehat{\nabla} F_i(\theta)] = \nabla F_i(\theta)$ and $\mathbb{E}[\|\widehat{\nabla} F_i(\theta) - \nabla F_i(\theta)\|^2] \le \sigma_{\mathrm{priv}}^2$ for all i. Then $\widehat{\nabla} F_I(\theta)$ is unbiased for $\nabla F(\theta) = K^{-1}\sum_{i=1}^K \nabla F_i(\theta)$ and*

$$\mathbb{E}\big[\|\widehat{\nabla} F_I(\theta) - \nabla F(\theta)\|^2\big] \le \sigma_{\mathrm{priv}}^2 + \frac{1}{K}\sum_{i=1}^K \|\nabla F_i(\theta) - \nabla F(\theta)\|^2.$$

In particular, under Assumption 2, $\mathbb{E}[\|\widehat{\nabla} F_I(\theta) - \nabla F(\theta)\|^2] \le \sigma_{\mathrm{priv}}^2 + \zeta^2$.

Proof. Unbiasedness follows from $\mathbb{E}[\widehat{\nabla} F_I(\theta)] = K^{-1}\sum_{i=1}^K \mathbb{E}[\widehat{\nabla} F_i(\theta)] = \nabla F(\theta)$. Also,

$$\mathbb{E}\|\widehat{\nabla} F_I - \nabla F\|^2 = \frac{1}{K}\sum_{i=1}^K \mathbb{E}\|\widehat{\nabla} F_i - \nabla F\|^2.$$

Adding and subtracting ∇F_i gives $\widehat{\nabla} F_i - \nabla F = (\widehat{\nabla} F_i - \nabla F_i) + (\nabla F_i - \nabla F)$. The cross term vanishes because $\mathbb{E}[\widehat{\nabla} F_i - \nabla F_i] = 0$, so $\mathbb{E}\|\widehat{\nabla} F_i - \nabla F\|^2 \le \sigma_{\mathrm{priv}}^2 + \|\nabla F_i - \nabla F\|^2$. Averaging over i proves the claim. □

A.3 Proof of Theorem 1

Let $t = 0, \dots, T-1$ and let $q(t)$ be the communication round corresponding to step t. The SGD update is $\theta^{t+1} = \theta^t - \eta g^t$.

Bias–noise decomposition. Define the surrogate gradient bias

$$b^t := \nabla \Phi^{(q(t))}(\theta^t) - \nabla \bar{\Phi}(\theta^t) = \frac{1}{2}\big(\nabla_\theta G_t(\theta^t) - \nabla_\theta G(\theta^t)\big).$$

On the event of Lemma 1, $\|b^t\| \le \Delta_T(\rho)/2$ for all t. Define $\xi^t := g^t - \nabla \Phi^{(q(t))}(\theta^t)$. By Assumption 1(2), $\mathbb{E}[\xi^t \mid \theta^t, q(t)] = 0$ and $\mathbb{E}[\|\xi^t\|^2 \mid \theta^t, q(t)] \le \sigma^2 + \zeta^2/4$. Hence

$$g^t = \nabla \bar{\Phi}(\theta^t) + b^t + \xi^t.$$

One-step descent on $\bar{\Phi}$. By Λ-smoothness of $\bar{\Phi}$ along iterates,

$$\bar{\Phi}(\theta^{t+1}) \le \bar{\Phi}(\theta^t) - \eta\langle \nabla \bar{\Phi}(\theta^t), g^t\rangle + \frac{\Lambda\eta^2}{2}\|g^t\|^2.$$

Taking conditional expectation given $\theta^t, q(t)$, using Young's inequality, and bounding the squared norm gives

$$\mathbb{E}[\bar{\Phi}(\theta^{t+1}) \mid \theta^t, q(t)] \le \bar{\Phi}(\theta^t) - \eta\left(\frac{1}{2} - \Lambda\eta\right)\|\nabla \bar{\Phi}(\theta^t)\|^2 + \left(\frac{\eta}{2} + \Lambda\eta^2\right)\|b^t\|^2 + \frac{\Lambda\eta^2}{2}\left(\sigma^2 + \frac{1}{4}\zeta^2\right).$$

Choose $\eta \le 1/(4\Lambda)$ so that $(1/2 - \Lambda\eta) \ge 1/4$ and $(\eta/2 + \Lambda\eta^2) \le 3\eta/4$. Since $\|b^t\| \le \Delta_T(\rho)/2$,

$$\mathbb{E}[\bar{\Phi}(\theta^{t+1})] \le \mathbb{E}[\bar{\Phi}(\theta^t)] - \frac{\eta}{4}\mathbb{E}[\|\nabla \bar{\Phi}(\theta^t)\|^2] + \frac{3\eta}{16}\Delta_T(\rho)^2 + \frac{\Lambda\eta^2}{2}\left(\sigma^2 + \frac{1}{4}\zeta^2\right).$$

Telescope. Summing over $t = 0, \dots, T-1$ and using $\mathbb{E}[\bar{\Phi}(\theta^T)] \geq \bar{\Phi}^\star := \inf_\theta \bar{\Phi}(\theta)$ gives

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}[\|\nabla\bar{\Phi}(\theta^t)\|^2] \leq \frac{4(\bar{\Phi}(\theta^0) - \bar{\Phi}^\star)}{\eta T} + \frac{3}{4}\Delta_T(\rho)^2 + 2\Lambda\eta\left(\sigma^2 + \frac{1}{4}\zeta^2\right),$$

which proves Theorem 1. □

Proof overview and standard assumptions. Our convergence analysis follows a standard nonconvex SGD template. Lemma 1 bounds the gap between the empirical public gradient and the reference public gradient uniformly over T steps by decomposing it into prompt covariate shift, pseudo-label conditional mismatch, and finite-sample concentration. Theorem 1 then plugs this uniform gap into a biased-SGD descent argument using Λ-smoothness, a bias–noise decomposition, Young's inequality, and telescoping. The assumptions are standard in nonconvex SGD and federated convergence analyses Ghadimi and Lan [2013], Karimireddy et al. [2020], Li et al. [2020b]; bounded-gradient assumptions are common and can be enforced via gradient clipping Li et al. [2020b], Abadi et al. [2016], Chen et al. [2020]; gradient-dissimilarity assumptions appear in FedProx/SCAFFOLD-style analyses Li et al. [2020a], Karimireddy et al. [2020]; KL/TV controls are standard in covariate-shift and domain-adaptation theory Sugiyama et al. [2007], Aminian et al. [2022]; and modeling pseudo-labels as noisy labels or conditional mismatch aligns with noisy-label and self-training analyses Natarajan et al. [2013], Wei et al. [2021].

B Detailed Privacy Analysis of FedCoFiT

This appendix provides a detailed privacy analysis of DP-FedCoFiT. We show that if each client's local LoRA fine-tuning procedure is differentially private, then the responses released on the public prompt set are also differentially private, and the full multi-round mechanism remains differentially private under adaptive composition.

Setup. Let K be the number of clients, with client index set $[K] = \{1, \dots, K\}$. Training proceeds for Q communication rounds, with round index set $[Q] = \{1, \dots, Q\}$. For each client $i \in [K]$, let $D_i = \{z_{i,1}, \dots, z_{i,n_i}\}$ denote the private local dataset of client i, where $n_i = |D_i|$. Let $D_{\text{pub}} = \{x_1, \dots, x_M\}$ denote the public prompt set shared by all clients, where $M = |D_{\text{pub}}|$.

Each client maintains a local language model. For client i, ϕ_i denotes the trainable LoRA parameters, while the backbone parameters are frozen. At communication round $q \in [Q]$, the client updates its LoRA parameters using its private dataset D_i and the information available from previous rounds.

Round history. For each round $q \in [Q]$, let $H^{(q-1)}$ denote the history available before round q. This history may include all previous client releases, server-side computations, and pseudo-labels or broadcasts sent back to clients up to round $q-1$. The exact contents of $H^{(q-1)}$ are not important for the privacy argument; the only requirement is that round q may depend adaptively on earlier rounds through $H^{(q-1)}$.

Client-side private fine-tuning mechanism. At round q, client i applies a randomized local training mechanism $\phi_i^{(q)} \leftarrow \mathcal{A}_{i,q}(D_i, H^{(q-1)})$. We assume that, for every fixed history $H^{(q-1)}$, the mechanism $\mathcal{A}_{i,q}(\cdot, H^{(q-1)})$ is $(\varepsilon_{i,q}, \delta_{i,q})$-differentially private with respect to D_i.

Released responses on the public prompt set. After the local update, client i evaluates its local model on each public prompt $x_j \in D_{\text{pub}}$. We write $\hat{y}_{i,j}^{(q)} \leftarrow \mathcal{M}_i(x_j; \phi_i^{(q)})$ for $j \in [M]$. The round-q message released by client i is $\mathcal{R}_i^{(q)} = \{\hat{y}_{i,j}^{(q)}\}_{j=1}^M$.

Server-side processing. After receiving $\{\mathcal{R}_i^{(q)}\}_{i=1}^K$, the server performs the standard FedCoFiT consensus step, including embedding the generated responses, clustering them in semantic space, selecting representative pseudo-labels, and broadcasting the pseudo-labeled public set back to clients. These operations are functions only of the released client messages and public information.

Definition 2 (Neighboring datasets)**.** Fix a client $i \in [K]$. Two datasets D_i and D_i' are neighboring if they differ in exactly one training example.

Definition 3 (Differential privacy)**.** A randomized mechanism $\mathcal{A}$ is (ε, δ)-differentially private with respect to client dataset D_i if, for every pair of neighboring datasets D_i, D_i' and every measurable set of outputs S,

$$\Pr[\mathcal{A}(D_i) \in S] \leq e^{\varepsilon} \Pr[\mathcal{A}(D_i') \in S] + \delta.$$

Round-level privacy. We first show that the public-prompt responses released by a client in a single round inherit the privacy guarantee of the local training mechanism.

Lemma 8 (Privacy of released round-q responses)**.** *Fix a client $i \in [K]$ and a communication round $q \in [Q]$. Suppose that, for the fixed history $H^{(q-1)}$, the local mechanism $\mathcal{A}_{i,q}(\cdot, H^{(q-1)})$ is $(\varepsilon_{i,q}, \delta_{i,q})$-differentially private with respect to D_i. Then the released message $\mathcal{R}_i^{(q)} = \{\hat{y}_{i,j}^{(q)}\}_{j=1}^{M}$ is also $(\varepsilon_{i,q}, \delta_{i,q})$-differentially private with respect to D_i.*

Proof. For fixed history $H^{(q-1)}$, the output $\phi_i^{(q)}$ is produced by $\mathcal{A}_{i,q}(D_i, H^{(q-1)})$, which is $(\varepsilon_{i,q}, \delta_{i,q})$-differentially private by assumption. The released message $\mathcal{R}_i^{(q)}$ is obtained by applying $\mathcal{M}_i(\cdot; \phi_i^{(q)})$ to the public prompt set D_{pub}. Since D_{pub} is public and fixed independently of D_i, the map $\phi_i^{(q)} \mapsto \mathcal{R}_i^{(q)}$ is a function of the private mechanism output and public information only. Therefore, by post-processing, $\mathcal{R}_i^{(q)}$ is also $(\varepsilon_{i,q}, \delta_{i,q})$-differentially private with respect to D_i. □

Multi-round privacy. Let $\mathcal{G}_i(D_i) = (\mathcal{R}_i^{(1)}, \mathcal{R}_i^{(2)}, \ldots, \mathcal{R}_i^{(Q)})$ denote the overall multi-round mechanism induced by client i. Since each $\mathcal{R}_i^{(q)}$ may depend on the previous history $H^{(q-1)}$, $\mathcal{G}_i$ is an adaptive composition of the per-round release mechanisms.

Theorem 3 (Record-level privacy of DP-FedCoFiT)**.** *Suppose that, for every communication round $q \in [Q]$ and every fixed history $H^{(q-1)}$, the client-side mechanism $\mathcal{A}_{i,q}(\cdot, H^{(q-1)})$ is $(\varepsilon_{i,q}, \delta_{i,q})$-differentially private with respect to D_i. Then:*

1. *the released public-prompt responses $\mathcal{R}_i^{(q)}$ are $(\varepsilon_{i,q}, \delta_{i,q})$-differentially private with respect to D_i for every round $q \in [Q]$;*
2. *the overall multi-round mechanism $\mathcal{G}_i(D_i) = (\mathcal{R}_i^{(1)}, \ldots, \mathcal{R}_i^{(Q)})$ is*

$$\left(\sum_{q=1}^{Q} \varepsilon_{i,q},\ \sum_{q=1}^{Q} \delta_{i,q}\right) \text{-differentially private}$$

with respect to D_i.

Proof. Part (1) follows directly from Lemma 8. For Part (2), the sequence $\mathcal{R}_i^{(1)}, \ldots, \mathcal{R}_i^{(Q)}$ is adaptive because the mechanism used at round q may depend on $H^{(q-1)}$. Differential privacy is closed under adaptive composition, so composing the per-round mechanisms over Q rounds gives the stated guarantee. Finally, all server-side operations in FedCoFiT, including response embedding, clustering, representative selection, and pseudo-label broadcast, are functions of released messages and public information only; hence, they incur no additional privacy loss by post-processing. □

Interpretation. The theorem establishes a record-level privacy guarantee for each client's private dataset. Sharing responses on the public prompt set does not invalidate privacy in DP-FedCoFiT because those responses are outputs of a differentially private client-side LoRA fine-tuning mechanism.

C Details on Experiments

C.1 Environments, Datasets and Metrics

Compute Resources : All experiments are run on a server with an Intel(R) Platinum 8480 and 5 x NVIDIA H100 80GB.

Public Prompt Set: Across all experiments, clients collaborate via a shared public prompt set $\mathcal{D}_{pub}$, which is sampled from the corresponding dataset and excluded from the private fine-tuning data used by clients in that setting.

Datasets:

Dolly-15K. An instruction-tuning dataset of $15K$ examples spanning diverse tasks (e.g., QA, summarization, extraction). In our instruction-compliance regime, we partition Dolly-15K across clients as private data and sample in-distribution public prompts from the same instruction pool when studying prompt-budget effects.

Alpaca. A widely used instruction-following corpus (tens of thousands of instruction–response pairs). We use Alpaca as a private fine-tuning dataset in the instruction-compliance regime and also as an out-of-distribution source of public prompts for robustness tests.

Wizard. A complex instruction-tuning dataset (Evol-Instruct style) emphasizing harder instructions and reasoning-heavy responses. We use Wizard for the knowledge/generalization regime as clients' private fine-tuning data.

OpenOrca. A GPT-augmented instruction dataset with reasoning traces (derived from FLAN-style tasks). We use OpenOrca as a private fine-tuning corpus in the knowledge/generalization regime and as an OOD public-prompt source in robustness experiments.

ShareGPT. A chat-style conversation dataset collected from ShareGPT, containing multi-turn user–assistant interactions. We use ShareGPT as a private fine-tuning corpus for chat-quality experiments and also as a format-shifted public-prompt source; compared to instruction prompts, these chat-oriented prompts can induce more diverse generations and reduce consensus strength.

UltraChat. A large-scale, high-quality conversational/instructional dialogue dataset. We use UltraChat as a private fine-tuning corpus for chat-quality experiments.

Medical Meadow Medical Flashcards. Medical Meadow Medical Flashcards [Han et al., 2023] is a medical question-answering dataset containing $33,955$ question answer pairs derived from medical educational flashcards. The dataset covers diverse medical subjects, including anatomy, physiology, pathology, and pharmacology. The original flashcards were reformulated into natural-language questions paired with free-text answers, providing supervision for medical instruction fine-tuning.

Benchmarks and Metrics:

IFEval (Instruction-Following Evaluation). We measure instruction compliance by evaluating models fine-tuned on instruction data (Dolly-15K/Alpaca) using the IFEval benchmark, which tests whether a model follows explicit instruction constraints.

MMLU (Massive Multitask Language Understanding). We evaluate knowledge and generalization after fine-tuning on Wizard/OpenOrca using MMLU, a broad multi-subject benchmark designed to probe factual knowledge and reasoning across many domains.

MT-Bench. We evaluate chat quality after fine-tuning on ShareGPT/UltraChat using MT-Bench, a multi-turn open-ended benchmark for chat assistants (commonly judged by a strong LLM-as-a-judge protocol).

MedQA. MedQA [Jin et al., 2021] is a medical question-answering benchmark based on professional medical licensing examinations, including the United States Medical Licensing Examination (USMLE). It consists of multiple-choice questions that assess clinical knowledge, diagnosis, and medical reasoning. We use MedQA to evaluate the ability of the fine-tuned models to generalize to medical examination questions.

MedMCQA. MedMCQA [Pal et al., 2022] is a large-scale medical multiple-choice question-answering benchmark containing approximately 194,000 questions collected from medical entrance examinations. It covers 21 medical subjects and approximately 2,400 healthcare topics, including basic medical sciences and clinical medicine. We use MedMCQA to assess medical knowledge generalization across diverse subjects.

PubMedQA. PubMedQA [Jin et al., 2019] is a biomedical question-answering benchmark constructed from PubMed research articles. It evaluates the ability to answer biomedical research questions using three possible labels: *yes*, *no*, and *maybe*.

D Experiment setup

We evaluate FedCoFiT across three adaptation regimes: instruction compliance, knowledge, and generalization and chat quality. For instruction compliance, clients fine-tune on Dolly-15K Zhang et al. [2024] and Alpaca Taori et al. [2023], and we evaluate instruction-following behaviour on IFEval Zhou et al. [2023]. For knowledge and generalization, clients fine-tune on Wizard Luo et al. [2025] and OpenOrca Lian et al. [2023], and we evaluate on MMLU Hendrycks et al. [2021]. For chat quality, clients fine-tune on ShareGPT Chiang et al. [2023] and UltraChat Ding et al. [2023], and we evaluate on MT-Bench Zheng et al. [2023]. We consider three foundation models: TinyLlama, LLaMA-7B, and LLaMA2-13B. For TinyLlama and LLaMA-7B we use $K = 10$ clients, and for LLaMA2-13B, we use $K = 5$ clients. All methods are evaluated under the same federation budget (same number of clients and communication rounds). We run IFEval experiments for 3 communication rounds, MMLU experiments for 2 rounds, and MT-Bench experiment for 1 communication round. In FedCoFiT experiments, clients share a public prompt set D_{pub} constructed by sampling prompts from the corresponding fine-tuning dataset and excluding them from private training splits. We use a fixed public prompt budget of $M = 500$ prompts. We apply LoRA with rank $r = 32$. For all models, we adapt the attention projection modules:`q_proj`, `k_proj`, `v_proj`, `o_proj`. We use a micro-batch size of 16 and an effective batch size of 128. The learning rate is set to 3×10^{-4} across experiments.

E Additional Experiments

E.1 Effect of Public Prompt Budget:

We first vary the number of public prompts by subsampling $M \in \{1000, 800, 500, 100, 50\}$ prompts from the Dolly-15K instruction pool, yielding public sets that are in-distribution with respect to the private fine-tuning data. Fig. 3 shows that FedCoFiT is stable for moderate public prompt size as performance is nearly unchanged at $M = 1000, 800, 500$ (IFEval$\approx 0.514 - 0.510$). When the size is reduced to $M \leq 100$, performance drops to ≈ 0.481. Importantly, even at large budgets, FedCoFiT matches or exceeds strong baselines (e.g., FLoRA at 0.5026) while at very small budgets it remains substantially above FedIT (0.4382).

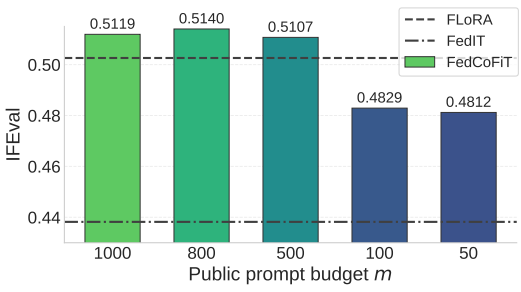

Figure 3: IFEval vs public prompt budget M.

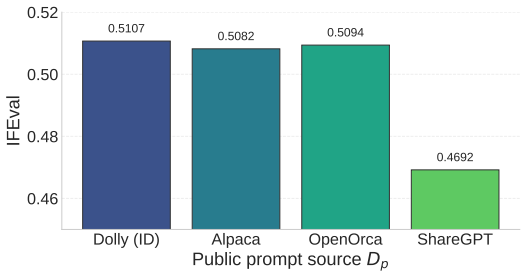

Figure 4: FedCoFiT robustness to public-prompt distribution shift

E.2 Effect of public prompt distribution shift:

Next, we test whether FedCoFiT requires the public prompts to match the client data distribution. We fix the prompt budget to a constant 500 and keep the private fine-tuning data unchanged (Dolly-15k), but replace the public prompt source. Specifically, we compare (1) D_{pub} sampled from Dolly-15k prompts (in-distribution) against out-of-distribution public prompts sampled from (2) Alpaca, (3) OpenOrca, and (4) ShareGPT, using prompts only in all cases. We report IFEval score for each setting. As shown in Fig. 4, using instruction-style public prompts such as Alpaca and OpenOrca yields only minor changes compared to Dolly-15k. This suggests that FedCoFiT is robust to moderate shifts in the public prompt distribution as long as the prompts remain instruction-following and aligned in format. In contrast, using ShareGPT prompts leads to a drop to 0.4692. We attribute this to a stronger distribution shift. ShareGPT prompts are chat-oriented and often reflect multi-turn conversational context, informal style, and different intent distributions than Dolly-style single-run instructions. As a result, client generations on these prompts become more diverse in surface form and content. Overall, these results indicate that FedCoFiT does not require a perfectly matched public prompt set, but benefits from public prompts that are at least format-consistent with the downstream adaptation task.

E.3 Evaluation under Heterogeneous Data Distribution

We evaluate FedCoFiT under category-skewed non-IID client distributions using TinyLlama and Dolly-15K. Specifically, we partition the training data across $K = 10$ clients using a Dirichlet distribution with concentration parameter $\alpha = 0.1$ to induce heterogeneous client data distributions. We retain $M = 500$ public prompts. All methods share matched client partitions, optimization budgets, and communication rounds. Under non-IID data, FedCoFiT achieves an IFEval score of 49.81 ± 0.012, compared with 47.29 ± 0.017 for FLoRA and 43.26 ± 0.015 for local training, averaged over three runs. FedCoFiT therefore improves over FLoRA by 2.52 points and local training by 6.55 points. Moving from IID to category-skewed non-IID data reduces FedCoFiT performance by 1.38 points, compared with 2.97 points for FLoRA.

E.4 Choice of Semantic Encoder

FedCoFiT builds server-side consensus in a semantic embedding space, where each client-generated response is mapped by a sentence encoder into a fixed-dimensional vector. The encoder choice affects both (1) the quality of semantic similarity and thus the quality of pseudo-labels and (2) algorithm efficiency, since embeddings are computed for every public prompt and client output each round. We therefore ablate a small set of widely used SBERT-style encoders that span different speed/size regimes. We fix the federated setup (TinyLlama clients, Dolly-15K private fine-tuning, fixed public prompt budget, identical decoding and clustering hyperparameters) and vary only the semantic encoder. We then evaluate the resulting fine-tuning model on IFEval benchmark. Table 4 summarizes encoder size, embedding dimension, encoding throughput (sentence/sec), and the downstream IFEval score obtained when fine-tuning on Dolly-15K and evaluating on IFEval. Across the tested encoders, all-MiniLM-L6-v2 provides the best overall trade-off. While larger encoders such as all-mpnet-base-v2 can be competitive, they are substantially slower and heavier. Multilingual encoders (e.g., distiluse-base-multilingual-cased-v1) also introduce additional overhead without improving Dolly/IFEval performance in our instruction-following setting. We therefore select all-MiniLM-L6-v2 as the default semantic encoder in all experiments due to its strong IFEval performance coupled with low model size and high encoding speed. While the tested encoders achieve relatively similar downstream IFEval scores, these results do not establish that they produce identical pairwise semantic rankings or clustering assignments. A detailed analysis of consensus stability across embedding spaces is left for future work.

Table 4: Semantic encoder ablation for FedCoFiT. We report encoder size, embedding dimension, encoding throughput (sentences/sec), and downstream IFEval when training on Dolly-15K and evaluating on IFEval.

Encoder	Size (MB)	Dim	Speed (sent/s)	IFEval ↑
all-mpnet-base-v2	420	768	2800	52.47
all-distilroberta-v1	290	768	4000	51.29
all-MiniLM-L12-v2	120	384	7500	51.12
all-MiniLM-L6-v2	**80**	**384**	**14200**	**51.19**
distiluse-base-multilingual-cased-v1	480	512	4000	52.17

E.5 Ablation on Consensus Representative Selection

FedCoFiT selects one client-generated response as the pseudo-label for each public prompt. To isolate the effect of this selection rule, we run FedCoFiT on TinyLlama with Dolly-15K private fine-tuning data and evaluate the resulting models on IFEval. We keep the semantic encoder, DBSCAN clustering hyperparameters, decoding configuration, public prompt budget $M = 500$, and local training setup fixed, and vary only how the final pseudo-label $y_j^\star$ is chosen. We compare three alternatives: (i) our default *majority-cluster centroid* rule, which first selects the largest non-outlier cluster and then chooses the response closest to its centroid; (ii) *majority-cluster random*, which selects a uniformly random response from the largest non-outlier cluster; and (iii) *global medoid*, which ignores clustering and chooses the response with the smallest average distance to all other client responses. This ablation tests whether both parts of the consensus rule are useful: majority-cluster selection filters minority modes and outliers, while centroid-based representative selection avoids arbitrary pseudo-label choice within the consensus cluster. Table 5 shows that our default representative-selection rule performs best. Selecting the response closest to the centroid of the majority cluster improves over choosing a random response from the same cluster, indicating that centrality within the consensus cluster matters for pseudo-label quality. The larger drop for the global-medoid baseline shows that clustering and outlier filtering are also important: choosing the most central response over all client outputs can be affected by minority modes or noisy generations. These results support the two-stage design of FedCoFiT, where the server first identifies the dominant semantic mode and then selects the most representative response within that mode.

Table 5: Ablation of pseudo-label representative selection for FedCoFiT on TinyLlama with Dolly-15K private data and IFEval evaluation. We keep DBSCAN, the semantic encoder, decoding, public prompt budget, and training setup fixed, and vary only how the final pseudo-label $y_j^\star$ is selected.

Selection rule	Uses majority cluster?	Outlier filtering?	IFEval ↑
Majority-cluster centroid representative	Yes	Yes	**51.10**
Random representative from majority cluster	Yes	Yes	50.06
Global medoid over all responses	No	No	48.17

E.6 Communication Example: LLaMA-2-13B scale.

Consider $K = 10$ clients, $M = 1024$ prompts per round, average response length $\bar{\ell} = 128$, and $c_t \approx 2$ bytes/token. Then $C_{\mathrm{SC}} \approx 2KM\bar{\ell}c_t = 2 \cdot 10 \cdot 1024 \cdot 128 \cdot 2 \approx 5.0$ MiB, where the leading factor 2 accounts for upload and download. For a LLaMA-2-13B-scale model with $L_{\mathrm{layers}} = 40$, hidden size $d_{\mathrm{model}} = 5120$, FP16 communication $c_{\mathrm{param}} = 2$, and rank $r = 32$, LoRA on W_q and W_v gives upload $C^{\uparrow}_{\mathrm{LoRA}(q,v)} = c_{\mathrm{param}} L_{\mathrm{layers}} 2r(d_{\mathrm{model}} + d_{\mathrm{model}}) = 52{,}428{,}800$ bytes ≈ 50.0 MiB, or about 100.0 MiB upload plus download. If LoRA is applied to all attention projections and the MLP with $d_{\mathrm{ff}} = 13824$, the upload is $C^{\uparrow}_{\mathrm{LoRA(attn+MLP)}} = c_{\mathrm{param}} L_{\mathrm{layers}} r\{4(d_{\mathrm{model}} + d_{\mathrm{model}}) + 3(d_{\mathrm{model}} + d_{\mathrm{ff}})\} \approx 238.7$ MiB, or about 477.4 MiB total. Orthogonal communication-saving mechanisms, such as dynamic or event-triggered communication protocols, could in principle be combined with Semantic Consensus by skipping consensus rounds when local model behavior or its semantic embedding changes only marginally across rounds [Kamp, 2019, Kamp et al., 2018, 2016].

E.7 Evaluation on a Recent Model Family.

To investigate whether semantic consensus generalizes beyond the original LLaMA-family backbones, we additionally evaluate Qwen3-0.6B on Dolly-15K using the same experiment setup in D. FedCoFiT achieves an IFEval score of 58.83, compared with 57.42 for FLoRA, corresponding to an improvement of 1.41 points. This targeted experiment provides additional evidence that the benefits of semantic consensus are not restricted to the original evaluated model families.

E.8 Conservative Fairness Control.

We additionally evaluate a conservative FLoRA control in which the 500 reserved examples are returned to the private training pool, including their ground-truth responses. Each reserved example is allocated once across clients, whereas FedCoFiT continues to use only the corresponding unlabeled public prompts. Under this setting, FLoRA improves from 50.26 to 50.29 IFEval, while FedCoFiT achieves 51.19. Thus, FedCoFiT remains 0.90 points above FLoRA even when the baseline receives these additional labeled examples.

F Impact Statement

FedCoFiT has the potential to make federated fine-tuning of large language models substantially more practical by replacing costly parameter aggregation with semantic consensus over model outputs. By exchanging only generated responses to shared public prompts, the method reduces communication independently of model size, supports heterogeneous clients with different architectures, tokenizers, or LoRA ranks, and can operate even when only black-box model access is available. Empirically, the paper shows that FedCoFiT matches or exceeds strong federated LoRA baselines across instruction following, knowledge generalization, and chat-quality tasks while using orders of magnitude less communication, with additional reductions in runtime, energy use, and estimated emissions. Its compatibility with differential privacy further strengthens its relevance for sensitive domains such as healthcare, finance, and enterprise settings, where data cannot be centralized but collaborative adaptation is valuable.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction clearly and accurately reflect the paper's scope and main contributions. They effectively summarize the objectives, methods, and key findings, providing a coherent overview that aligns well with the paper's content.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: The work limitations has been discussed in sec 6

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [Yes]

Justification: All full assumptions and the full, complete, correct proofs are available in the appendix

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes]

Justification: we provided the experiment setup in sec.5. Additional experimental details are provided in Appendix.C, and we provide the code to run all experiments in a GitHub repository.

Guidelines:

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

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

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

   Answer: [Yes]

   Justification: The code is available at https://github.com/kamp-lab-mlai/FedCoFiT

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Experimental details are provided in App.C

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: We report mean $\pm$ standard deviation over three independent runs for the corresponding multi-run experiments; experimental details are provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Details are in the Appendix.C

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification:

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Potential broader impacts are discussed in Appendix.F

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: We do not release pretrained models or scraped datasets.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [N/A]

    Justification: The paper does not release new assets. We do provide the code to reproduce our experiments, which is documented.

    Guidelines:

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

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

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

    Answer: [N/A]

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

    Guidelines:

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

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

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

    Answer: [N/A]

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

    Guidelines:

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

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [Yes]

    Justification: LLMs are the models being federatively fine-tuned and generate the public-prompt responses used by FedCoFiT; this is described in Sections 3 and 5.

    Guidelines:

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