CLOAKFL: MITIGATING GLOBAL MODEL EXPOSURE IN FEDERATED LEARNING VIA CONVERGENCE-AWARE PROXY DISTRIBUTION

Anonymous authors
Paper under double-blind review

ABSTRACT

In federated learning, the server repeatedly sends the global model to participating clients for local training, exposing the model to the risk of leakage or redistribution outside the federation. Proxy Model Distribution (PMD) mitigates this risk by distributing intentionally degraded proxy models instead of the clean global model. Existing PMD methods, however, face a protection–convergence trade-off: stronger degradation reduces the proxy's inference performance but impedes global convergence, whereas weaker degradation better preserves global convergence but provides weaker protection. We propose CloakFL, a PMD method that adds convergence-aware perturbations to the global model. CloakFL constructs perturbation directions derived from the previous round's local training, enabling effective degradation with relatively small perturbations. It then scales each perturbation according to the target client's latest update to better preserve global convergence. Through convergence analysis and experiments on CNN and Transformer architectures, we show that the performance of the proxy models remains substantially below that of the global model, while keeping the global model's performance close to that of standard federated learning.

1 INTRODUCTION

Federated Learning (FL) (McMahan et al., 2017) is a distributed learning paradigm in which multiple clients collaboratively train a shared global model without sharing their raw data with a central server. In each round, the server distributes the global model to participating clients, each client trains the received model on its local data, and the server updates the global model by averaging the locally trained models. As this process is repeated over multiple rounds, the global model, trained using private data and substantial computational and communication resources, is continuously exposed to client devices (Kairouz et al., 2021). Consequently, even a single malicious or compromised client can leak the model. Nevertheless, existing FL research has primarily focused on data privacy (Wei et al., 2020), while the protection of the global model itself has received relatively limited attention. Prior work has addressed this risk from several directions. Some methods focus on ownership verification or leak traceability after exposure, for example through watermarking and fingerprinting (Tekgul et al., 2020; Li et al., 2023; Shao et al., 2025). Other approaches prevent direct access to the clean model through trusted execution environments or cryptographic computation (Nguyen et al., 2021; Jin et al., 2023). However, the former can respond only after the model has already been leaked, whereas the latter may incur substantial system overhead due to specialized hardware or expensive computation. We defer a broader discussion of model-protection approaches to Appendix B.

Proxy Model Distribution (PMD) (Dong et al., 2023; Zhu et al., 2024; Xu et al., 2025; Kwon & Choi, 2026) protects the global model by distributing degraded proxy models instead of exposing the clean global model. Even if a proxy model is leaked, its low inference performance limits the value of the exposed model. PMD therefore provides protection at the model distribution stage without requiring specialized security hardware or expensive cryptographic computation. However, because clients begin local training from degraded proxy models, proxy degradation also affects global optimization. Strong degradation improves protection but can impede global convergence, whereas weak degradation preserves convergence but provides insufficient protection. The central challenge

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

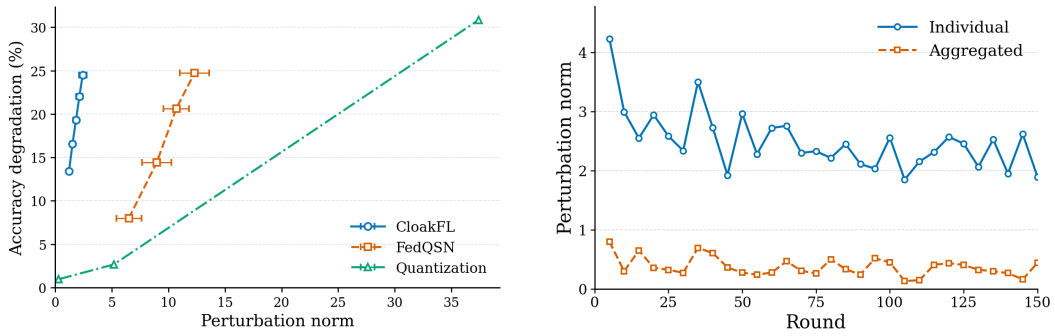


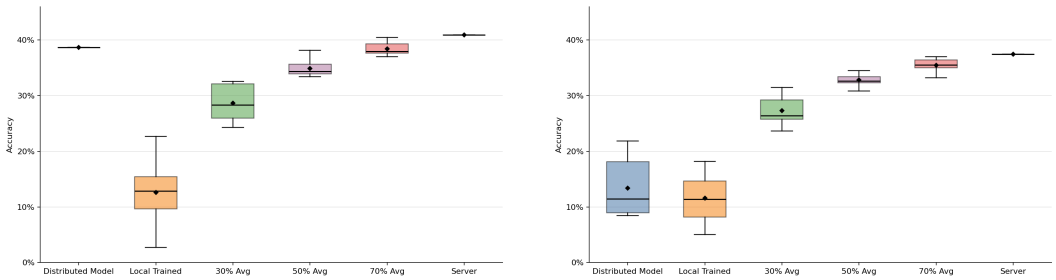
Figure 1: Key properties of CloakFL perturbations. (a) CloakFL achieves substantial accuracy degradation with small perturbations. (b) The perturbation norm is much smaller after aggregation.

of PMD is therefore to simultaneously achieve low proxy-model performance and high global-model performance. FedLPP (Zhu et al., 2024) and FedQSN (Xu et al., 2025) are existing PMD methods that exemplify this trade-off between protection and convergence. FedLPP constructs proxy models by quantizing the LoRA parameters (Hu et al., 2022) distributed to clients, whereas FedQSN constructs proxy models using quantization and random masking. Such degradation mechanisms can make the protection-convergence trade-off difficult to manage, as stronger proxy degradation increasingly interferes with local training and can consequently impede global convergence.

To address this problem, we propose CloakFL, a PMD method that adds convergence-aware perturbations to the global model. CloakFL constructs its perturbation directions from local training in the previous round. Specifically, it uses client drift (Karimireddy et al., 2020) from local training—the bias induced by overfitting to local data—as a perturbation direction. This enables strong proxy degradation even with small perturbations, while the perturbations tend to cancel across clients during aggregation, as shown in Figure 1. CloakFL also calibrates the magnitude of each perturbation to the scale of the target client's recent update, keeping the injected perturbation commensurate with the scale of local optimization. As a result, CloakFL can substantially reduce proxy inference performance while preserving stable local training and global convergence. Under a simplified full-participation setting with strongly convex and smooth local objectives, we establish a FedAvg-style $\mathcal{O}(1/T)$ convergence guarantee, with extra terms from the aggregated perturbation's second moment and dispersion. This result clarifies the conditions under which unbiased perturbation directions can reduce proxy inference performance without substantially hindering global convergence.

We evaluate CloakFL on both CNN and Transformer architectures and compare it with FedQSN (Xu et al., 2025), a representative PMD baseline. Existing PMD studies primarily assess model protection based on the raw performance of proxy models. To consider more realistic threats, we introduce practical recovery scenarios in which clients attempt to restore proxy performance using their local data, either individually or through collaboration with other clients. These scenarios enable a more rigorous evaluation of protection strength under realistic client capabilities. CloakFL substantially degrades exposed proxy models and improves robustness against recovery attacks, while maintaining global performance close to that of standard FedAvg. Our main contributions are summarized as follows:

- **Convergence-aware proxy model distribution.** CloakFL designs convergence-aware perturbations from local training to construct proxy models that remain effectively degraded even under recovery attacks, while preserving stable global convergence.
- **Convergence analysis for generalized proxy distribution in federated learning.** We derive a FedAvg-style convergence bound showing that the additional error depends on the magnitude and dispersion of the aggregated proxy perturbation.
- **Extensive evaluation on CNNs, Transformers, and recovery attacks.** Extensive experiments, including practical recovery scenarios, show that CloakFL substantially reduces the performance of proxy models while maintaining global performance close to that of clean FedAvg.



(a) Standard Federated Learning **(b) CloakFL: Proxy Model Distribution**

Figure 2: Model performance during local training and aggregation in standard FL and CloakFL. Panel (a) shows the performance drop of individual locally trained models and its recovery through aggregation in standard FL. Panel (b) shows CloakFL distributing degraded client-specific proxy models while recovering high server-model performance through aggregation.

2 PRELIMINARIES

2.1 FEDERATED LEARNING AND CLIENT DRIFT

In standard FL (McMahan et al., 2017), at round r, the server distributes the global model w_r to K participating clients $\mathcal{S}_r$, and each client $i \in \mathcal{S}_r$ trains w_r on its local data and returns a locally trained model $w_{r,i}^+ = w_r + \Delta_{r,i}$ that incorporates its local update $\Delta_{r,i}$. The server then constructs the next global model by averaging the returned local models: $w_{r+1} = \frac{1}{K} \sum_{i \in \mathcal{S}_r} w_{r,i}^+$.

When client data are non-IID (Li et al., 2020b; Karimireddy et al., 2020; Wang et al., 2020), local objectives differ across clients and from the global objective, causing local updates to follow different directions. We view each local update as the sum of a shared component that contributes to global aggregation and a client-specific deviation induced by local training. We refer to this deviation as the client drift $d_{r,i}$:

$$\Delta_r = \frac{1}{K} \sum_{i \in \mathcal{S}_r} \Delta_{r,i}, \qquad d_{r,i} = \Delta_{r,i} - \Delta_r. \tag{1}$$

As shown in Figure 2(a), local training from a global model can produce substantially degraded client models due to client drift. When these models are aggregated, however, their heterogeneous drifts tend to cancel out, allowing the global model to progressively recover its performance.

2.2 PROXY MODEL DISTRIBUTION

Proxy Model Distribution (PMD) protects the global model by distributing degraded, client-specific proxy models $\tilde{w}_{r,i}$ instead of directly exposing the global model w_r. At round r, each participating client i receives $\tilde{w}_{r,i} = w_r + \delta_{r,i}$, where $\delta_{r,i}$ is a perturbation designed to degrade proxy-model performance. The client then performs local training on the proxy and returns the updated model $\tilde{w}_{r,i}^+$, yielding the local update $\Delta_{r,i} = \tilde{w}_{r,i}^+ - \tilde{w}_{r,i}$. The server aggregates the returned models to obtain the next global model: $w_{r+1} = \frac{1}{|\mathcal{S}_r|} \sum_{i \in \mathcal{S}_r} \tilde{w}_{r,i}^+$. While PMD protects the global model by degrading client-specific proxies, local training must still proceed from these perturbed models. Therefore, the perturbation $\delta_{r,i}$ should sufficiently suppress proxy-model performance while minimizing its impact on local optimization and global convergence.

2.3 THREAT MODEL

We consider malicious or compromised clients that attempt to recover a high-performing model from the client-specific proxies distributed during federated learning. In the Single-Client Recovery setting, an adversarial client has access only to its own proxy model and local data, which it may use to further train or fine-tune the proxy. We assume that no additional task-relevant data are available to the adversary. We also consider Multi-Client Recovery, where clients participating in the same communication round collude by sharing and averaging their proxy or recovered models. Because

these models originate from the same global model, averaging can effectively cancel client-specific perturbations, making same-round collusion a strong recovery setting. We evaluate coalitions of up to 30% of participating clients as a bounded-collusion stress test, following prior robust distributed learning studies (Yin et al., 2018; Fung et al., 2020; Cao et al., 2021; Shayan et al., 2018). Our goal is not to guarantee protection against unrestricted collusion, but to prevent individual clients or limited coalitions from recovering a high-performing approximation of the clean global model.

3 Theoretical Analysis

3.1 Problem Formulation

We use r to denote a communication round and $e \in \{0, \dots, E-1\}$ to denote a local SGD step within the round. Federated learning aims to minimize the following global objective:

$$\min_{w \in \mathbb{R}^d} F(w) \coloneqq \frac{1}{N} \sum_{i=1}^{N} \underbrace{\mathbb{E}_{\xi \sim \mathcal{D}_i}[\ell(w; \xi)]}_{F_i(w)}, \tag{2}$$

where N denotes the total number of clients and $\mathcal{D}_i$ is the local data distribution of client i. Following the notation in Li et al. (2020b), FedAvg proceeds in rounds of local updates and periodic parameter averaging. Specifically, during each communication round r, each client i performs E steps of stochastic gradient descent (SGD), indexed by $e = 0, \dots, E-1$, before the server aggregates the resulting local models. Let $v_{r,i}^{(e+1)}$ denote the result of one step of stochastic gradient descent from $w_{r,i}^{(e)}$:

$$v_{r,i}^{(e+1)} = w_{r,i}^{(e)} - \eta_{r,e} \nabla F_i\left(w_{r,i}^{(e)}; \xi_{r,i}^{(e)}\right), \tag{3}$$

where $\eta_{r,e}$ denotes the learning rate and $\xi_{r,i}^{(e)}$ is a mini-batch sampled from client i's local dataset. Within the round, we set $w_{r,i}^{(e+1)} = v_{r,i}^{(e+1)}$. After the final local step of round r, the server first computes the clean global model by $w_{r+1} = \frac{1}{N} \sum_{i=1}^{N} v_{r,i}^{(E)}$. Instead of broadcasting w_{r+1} directly, the server distributes a client-specific perturbed proxy model at the beginning of the next round:

$$w_{r+1,i}^{(0)} \leftarrow w_{r+1} + \delta_{r+1,i}, \qquad i = 1, \dots, N, \tag{4}$$

where $\delta_{r+1,i} \in \mathbb{R}^d$ denotes the perturbation added for client i.

3.2 Convergence Result

We establish the convergence of CloakFL by adapting the convergence analysis of FedAvg (Li et al., 2020b) to the full-participation setting. In particular, we treat the aggregation perturbation δ analogously to the error term introduced by partial participation. As in that analysis, we assume that each local objective function F_i is μ-strongly convex and L-smooth. We also adopt the standard assumptions on stochastic gradients: bounded variance (Assumption 3.1) and uniformly bounded expected squared norm (Assumption 3.2).

Assumption 3.1 (Bounded variance of stochastic gradients)**.** *For each client i, there exists a constant $\sigma_i^2 > 0$ such that, for every communication round r and local step e,*

$$\mathbb{E}\left[\left\|\nabla F_i(w_{r,i}^{(e)}; \xi_{r,i}^{(e)}) - \nabla F_i(w_{r,i}^{(e)})\right\|^2\right] \le \sigma_i^2.$$

Assumption 3.2 (Uniformly bounded stochastic gradients)**.** *There exists a constant $G > 0$ such that, for every communication round r, local step e, and client $i = 1, \dots, N$,*

$$\mathbb{E}\left[\left\|\nabla F_i(w_{r,i}^{(e)}; \xi_{r,i}^{(e)})\right\|^2\right] \le G^2.$$

We next introduce the additional assumptions needed to guarantee convergence in the presence of aggregation perturbations. For each communication round r, define $\bar{\delta}_r \coloneqq \frac{1}{N} \sum_{i=1}^{N} \delta_{r,i}$. Let $\mathcal{F}_r$ denote the filtration containing all randomness up to the point immediately before the round-r perturbations are applied. In particular, w_r is $\mathcal{F}_r$-measurable.

Assumption 3.3 (Conditionally unbiased aggregated perturbation)**.** *For every round r,*

$$\mathbb{E}\big[\bar{\delta}_r \mid \mathcal{F}_r\big] = 0.$$

Assumption 3.4 (Bounded second moment of aggregated perturbation)**.** *There exists a constant $\varepsilon_{\mathrm{agg}} > 0$ such that, for every communication round r,*

$$\mathbb{E}\Big[\big\|\bar{\delta}_r\big\|^2\Big] \le \eta_{r,0}^2 \varepsilon_{\mathrm{agg}}^2.$$

Assumption 3.5 (Bounded dispersion of client-level perturbations)**.** *There exists a constant $D > 0$ such that, for every communication round r,*

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\big\|\delta_{r,i} - \bar{\delta}_r\big\|^2\right] \le \eta_{r,0}^2 D^2.$$

Theorem 3.1. *Suppose that Assumptions 3.1 to 3.5 hold, and define $\kappa := L/\mu$, $\gamma := \max\{8\kappa, E\}$, $\Gamma := F^\star - \frac{1}{N}\sum_{i=1}^{N} F_i^\star$, and $\eta_{r,e} := \frac{2}{\mu(\gamma + rE + e)}$. Then, under full client participation, after T communication rounds CloakFL satisfies*

$$\mathbb{E}\Big[F(\bar{w}_T^{(0)})\Big] - F^\star \le \frac{2\kappa}{\gamma + TE}\left(\frac{B + C}{\mu} + \frac{\mu\gamma}{4}\|w_0 - w^\star\|^2\right),$$

$$B = \frac{1}{N^2}\sum_{i=1}^{N}\sigma_i^2 + 6L\Gamma + 8(E-1)^2G^2, \quad C = 8(E-1)^2G^2 + 16D^2 + \varepsilon_{\mathrm{agg}}^2.$$

The complete proof is provided in Appendix F.

3.3 Comparison with Partial Participation

The convergence result under partial device participation in Li et al. (2020b) is almost identical to Theorem 3.1, except for the constant term C. Specifically, let $\mathcal{S}_r$ denote the set of clients participating at round r, where $\mathcal{S}_r$ contains a subset of K client indices sampled uniformly with replacement from the N clients. Under partial participation, the resulting constant is

$$C_{\mathcal{S}_r} = \frac{4}{K}E^2G^2. \tag{5}$$

In contrast, CloakFL introduces an additional constant

$$C_{\mathrm{CloakFL}} = 8(E-1)^2G^2 + 16D^2 + \varepsilon_{\mathrm{agg}}^2. \tag{6}$$

In this bound, the additional term for partial participation scales inversely with the number of sampled clients K, whereas the additional term for CloakFL depends on the second moment and dispersion of the injected perturbations. This highlights that controlling perturbation moments is central to preserving convergence.

4 CloakFL

We propose CloakFL, a convergence-aware PMD method that uses client drift to construct small perturbations that strongly degrade proxy performance and tend to cancel during aggregation. CloakFL also calibrates their magnitudes using each target client's recent model-update scale to align the perturbation scale with local optimization.

4.1 Perturbation Direction Construction

Following the decomposition introduced in Section 2.1, such a local update can be viewed as containing both a shared component, which reflects the global optimization direction, and a client-specific deviation induced by local training. By removing the shared component, CloakFL isolates the client-specific component while keeping the resulting directions zero-mean: $\frac{1}{K}\sum_{i\in\mathcal{S}_r} d_{r,i} = 0$.

In each round with K participating clients, CloakFL assigns the K client drifts obtained from $\mathcal{S}_{r-1}$ to clients in $\mathcal{S}_r$ without replacement, using each drift as a perturbation direction. To prevent clients from exploiting alignment between their own updates and the assigned perturbations, CloakFL avoids assigning a client a drift derived from its own update. While individual client drifts can substantially degrade proxy-model performance, their heterogeneous effects are naturally attenuated through server-side aggregation.

Algorithm 1 CloakFL Training Procedure

Require: Initial model w_0, number of rounds T, clients per round K, scaling factor ρ, stability constant ϵ
Ensure: Final global model w_T

Run a standard FL round at $r = 0$ to obtain w_1, $\mathcal{S}_0$, and $\{w_{0,i}^+ \mid i \in \mathcal{S}_0\}$
$d_{0,i} \leftarrow (w_{0,i}^+ - w_0) - \frac{1}{K}\sum_{k\in\mathcal{S}_0}(w_{0,k}^+ - w_0), \quad m_i^{\text{last}} \leftarrow w_{0,i}^+ - w_0, \quad \forall i \in \mathcal{S}_0$
for $r = 1, \ldots, T-1$ **do**
  Sample K clients $\mathcal{S}_r$
  Assign distinct $j \in \mathcal{S}_{r-1}$ to each $i \in \mathcal{S}_r$, with $j \neq i$
  for all $i \in \mathcal{S}_r$ **in parallel do**
    if m_i^{last} is unavailable, set $m_i^{\text{last}} \leftarrow m_k^{\text{last}}$ for a randomly selected client k
    for each layer ℓ **do**
      $\delta_{r,i}^{(\ell)} \leftarrow \rho \frac{\left\|m_i^{\text{last},(\ell)}\right\|_2}{\left\|d_{r-1,j}^{(\ell)}\right\|_2 + \epsilon} d_{r-1,j}^{(\ell)}$
    end for
    $\tilde{w}_{r,i} \leftarrow w_r + \delta_{r,i}, \quad \tilde{w}_{r,i}^+ \leftarrow \text{LocalTrain}(i, \tilde{w}_{r,i})$
  end for
  $w_{r+1} \leftarrow \frac{1}{K}\sum_{i\in\mathcal{S}_r} \tilde{w}_{r,i}^+$
  $d_{r,i} \leftarrow (\tilde{w}_{r,i}^+ - \tilde{w}_{r,i}) - \frac{1}{K}\sum_{k\in\mathcal{S}_r}(\tilde{w}_{r,k}^+ - \tilde{w}_{r,k}), \quad m_i^{\text{last}} \leftarrow \tilde{w}_{r,i}^+ - w_r, \quad \forall i \in \mathcal{S}_r$
end for
return w_T

4.2 Perturbation Magnitude Calibration

Once the perturbation direction is determined, CloakFL calibrates its magnitude according to the target client's recent optimization scale. Because model-update magnitudes can vary substantially across clients and rounds, CloakFL uses the target client's most recent model update as a reference. After client i participates in round r, we define its model-update scale as the difference between the locally trained model and the corresponding clean global model. CloakFL stores the most recent value of $m_{r,i}$, denoted by m_i^{last}, and uses it to scale future perturbations. When the client drift $d_{r-1,j}$ from source client j is assigned to target client i, the reference magnitude and layer-wise perturbation are defined as

$$m_{r,i} = \tilde{w}_{r,i}^+ - w_r \qquad \delta_{r,i}^{(\ell)} = \rho \frac{\|m_i^{\text{last},(\ell)}\|_2}{\|d_{r-1,j}^{(\ell)}\|_2 + \epsilon} d_{r-1,j}^{(\ell)}. \tag{7}$$

Here, $\rho > 0$ controls the overall perturbation strength, and $\epsilon > 0$ ensures numerical stability. The server then constructs the proxy model as $\tilde{w}_{r,i} = w_r + \delta_{r,i}$. Although the client drifts are zero-mean before magnitude calibration, client-specific layer-wise scaling does not preserve exact cancellation. Nevertheless, Figure 1(b) shows that the aggregated perturbation remains substantially attenuated through server-side aggregation.

Algorithm 1 summarizes the overall training procedure of CloakFL. The first communication round serves as an initialization phase, during which the server collects the client drifts and model-update scales needed in subsequent rounds. For clients without prior participation history, CloakFL initializes the model-update scale using the most recent scale available from another client.

5 Experimental Evaluation

5.1 Experimental Setup

We evaluate CloakFL on four tasks: MobileNetV2 (Sandler et al., 2018) on CIFAR-100 (Krizhevsky, 2009), ResNet-50 (He et al., 2016) on Tiny-ImageNet (Le & Yang, 2015), DistilBERT (Sanh et al., 2020) on 20 Newsgroups (Lang, 1995), and GPT-2 Small (Radford et al., 2019) on the E2E dataset (Novikova et al., 2017). We use accuracy for the classification tasks and corpus-level BLEU (Papineni et al., 2002) for E2E.

We consider both cross-silo and cross-device FL (Kairouz et al., 2021). In the cross-silo setting, we use 20 clients and sample 50% of them at each communication round. In the cross-device setting, we use 100 clients and sample 10% of them per round. We partition client data using a Dirichlet

Table 1: Server and proxy-model performance (%) across cross-device and cross-silo settings. Proxy results report the mean ± standard deviation across participating clients in the final round, and Gap denotes the difference between the server performance and the mean proxy performance.

Env.	β	Method	CIFAR-100			Tiny-ImageNet			E2E			20 Newsgroups		
			Server	Proxy	Gap	Server	Proxy	Gap	Server	Proxy	Gap	Server	Proxy	Gap
Cross-device	0.1	FedAvg	42.1	42.1 ± 0.0	0.0	36.0	36.0 ± 0.0	0.0	61.3	61.3 ± 0.0	0.0	67.3	67.3 ± 0.0	0.0
		FedQSN	34.2	26.3 ± 3.3	7.9	29.1	26.7 ± 0.4	2.4	61.0	54.8 ± 1.2	6.2	59.7	59.7 ± 0.1	0.0
		CloakFL	37.9	13.3 ± 3.5	**24.6**	29.6	7.1 ± 1.3	**22.5**	62.9	18.5 ± 8.2	**44.4**	62.0	34.3 ± 5.7	**27.7**
	0.3	FedAvg	54.4	54.4 ± 0.0	0.0	39.6	39.6 ± 0.0	0.0	60.2	60.2 ± 0.0	0.0	68.6	68.6 ± 0.0	0.0
		FedQSN	50.1	44.0 ± 2.9	6.1	35.6	26.6 ± 2.9	9.0	58.4	47.6 ± 1.0	10.7	64.4	64.4 ± 0.2	0.1
		CloakFL	51.2	21.5 ± 10.1	**29.8**	36.2	13.3 ± 3.9	**23.0**	60.3	1.0 ± 1.5	**59.3**	65.5	50.8 ± 2.0	**14.7**
	0.6	FedAvg	57.2	57.2 ± 0.0	0.0	46.0	46.0 ± 0.0	0.0	61.7	61.7 ± 0.0	0.0	69.6	69.6 ± 0.0	0.0
		FedQSN	50.3	32.5 ± 4.9	17.8	40.8	33.2 ± 1.5	7.6	56.5	55.1 ± 1.0	1.4	65.6	67.3 ± 0.1	-1.7
		CloakFL	54.3	24.7 ± 12.8	**29.6**	41.1	16.7 ± 2.6	**24.4**	59.6	5.4 ± 4.5	**54.2**	66.0	53.6 ± 1.9	**12.4**
Cross-silo	0.1	FedAvg	46.6	46.6 ± 0.0	0.0	40.8	40.8 ± 0.0	0.0	62.0	62.0 ± 0.0	0.0	65.1	65.1 ± 0.0	0.0
		FedQSN	42.8	38.6 ± 1.0	4.2	36.4	30.1 ± 0.6	6.3	59.8	30.4 ± 2.8	29.4	58.3	58.4 ± 0.1	-0.1
		CloakFL	43.1	20.3 ± 2.0	**22.8**	37.3	15.1 ± 1.1	**22.2**	60.9	0.0 ± 0.0	**60.9**	60.0	31.8 ± 0.6	**28.3**
	0.3	FedAvg	57.3	57.3 ± 0.0	0.0	48.9	48.9 ± 0.0	0.0	62.7	62.7 ± 0.0	0.0	69.2	69.2 ± 0.0	0.0
		FedQSN	53.1	36.2 ± 6.8	16.9	46.9	39.4 ± 1.9	7.5	63.2	32.8 ± 3.7	30.4	66.8	66.8 ± 0.2	0.0
		CloakFL	53.8	22.7 ± 2.4	**31.2**	47.9	21.8 ± 7.2	**26.1**	63.2	2.6 ± 2.2	**60.6**	67.1	46.9 ± 0.8	**20.2**
	0.6	FedAvg	62.1	62.1 ± 0.0	0.0	52.6	52.6 ± 0.0	0.0	62.3	62.3 ± 0.0	0.0	68.8	68.8 ± 0.0	0.0
		FedQSN	58.6	42.8 ± 7.3	15.8	50.3	42.8 ± 2.5	7.5	61.2	27.1 ± 4.6	34.1	67.3	67.5 ± 0.1	-0.2
		CloakFL	59.4	28.9 ± 4.4	**30.5**	51.2	27.8 ± 1.3	**23.4**	62.2	7.1 ± 10.0	**55.1**	67.8	51.5 ± 2.0	**16.3**

distribution (Hsu et al., 2019) with $\beta \in \{0.1, 0.3, 0.6\}$. For each task, all methods use the same training configuration. Detailed experimental settings are provided in Appendix D.

Baselines. FedLPP (Zhu et al., 2024) is designed for LoRA-based (Hu et al., 2022) fine-tuning and is therefore difficult to apply directly to full-model FL settings. We thus use FedQSN (Xu et al., 2025), which combines the quantization-based proxy degradation of FedLPP with masking, as the PMD baseline, and use FedAvg (McMahan et al., 2017) without model protection as a reference for server performance. For each experimental setting, we select operating points where FedQSN and CloakFL achieve similar final server performance and compare the corresponding proxy and recovered model performance. Appendix E additionally reports quantization and quantization-with-masking baselines at multiple bit-widths and masking ratios.

5.2 Protection-Convergence Trade-off

Table 1 reports the final server performance and raw proxy performance of FedAvg, FedQSN, and CloakFL. Proxy reports the mean and standard deviation across participating clients in the final round, and Gap denotes the difference between the server performance and the mean proxy performance.

Across all 24 settings, CloakFL achieves a larger server–proxy performance gap than FedQSN, with a consistent trend across both CNN and Transformer architectures. This indicates that CloakFL more effectively suppresses raw proxy performance while maintaining server performance comparable to that of FedQSN.

5.3 Recovery Evaluation

Following the threat model in Section 2.3, we evaluate model protection under both Single-Client and Multi-Client Recovery. For recovery fine-tuning, we use one local epoch with a learning rate set to 10% of that used for standard local training.

Single-Client Recovery. In the Single-Client Recovery setting, we evaluate three variants: *Proxy*, *Proxy Train*, and *Proxy FT*. *Proxy* denotes the raw proxy model distributed to a client, whereas *Proxy Train* denotes the model obtained by applying the standard local training procedure to the

Table 2: Single-client recovery performance (%) across cross-device and cross-silo settings. Proxy, Proxy Train, and Proxy FT report the mean performance across participating clients in the final communication round, and Max denotes the maximum of the three means. Underlined values exceed the corresponding Server performance.

Env.	β	Method	CIFAR-100					Tiny-ImageNet					E2E					20 Newsgroups				
			Server	Proxy	Proxy Train	Proxy FT	Max	Server	Proxy	Proxy Train	Proxy FT	Max	Server	Proxy	Proxy Train	Proxy FT	Max	Server	Proxy	Proxy Train	Proxy FT	Max
Cross-device	0.1	FedQSN	34.2	26.3	11.8	36.0	36.0	29.1	26.7	7.0	28.8	28.8	61.0	54.8	43.3	59.5	59.5	59.7	59.7	24.3	56.9	59.7
		CloakFL	37.9	13.3	15.4	22.5	**22.5**	29.6	7.1	8.4	16.1	**16.1**	62.9	18.5	53.5	57.1	**57.1**	62.0	34.3	36.8	38.0	**38.0**
	0.3	FedQSN	50.1	44.0	20.6	51.3	51.3	35.6	26.6	13.1	36.6	36.6	58.4	47.6	43.7	56.7	56.7	64.4	64.4	41.6	63.4	64.4
		CloakFL	51.2	21.5	27.2	32.8	**32.8**	36.2	13.3	16.1	24.3	**24.3**	60.3	1.0	48.7	49.0	**49.0**	65.5	50.8	55.7	54.0	**55.7**
	0.6	FedQSN	50.3	32.5	19.9	47.6	47.6	40.8	33.2	17.8	41.4	41.4	56.5	55.1	44.5	56.6	56.6	65.6	67.3	52.7	65.9	67.3
		CloakFL	54.3	24.7	28.0	33.0	**33.0**	41.1	16.7	21.4	28.6	**28.6**	59.6	5.4	52.1	48.6	**52.1**	66.0	53.6	56.2	56.0	**56.2**
Cross-silo	0.1	FedQSN	42.8	38.6	15.0	40.6	40.6	36.4	30.1	7.4	31.5	31.5	59.8	30.4	45.7	59.4	59.4	58.3	58.4	17.8	32.3	58.4
		CloakFL	43.1	20.3	17.1	32.4	**32.4**	37.3	15.1	10.7	25.1	**25.1**	60.9	0.0	53.4	54.2	**54.2**	60.0	31.8	25.8	37.5	**37.5**
	0.3	FedQSN	53.1	36.2	20.0	53.7	53.7	46.9	39.4	19.8	45.3	45.3	63.2	32.8	51.6	58.6	58.6	66.8	66.8	44.0	60.3	66.8
		CloakFL	53.8	22.7	24.9	41.6	**41.6**	47.9	21.8	22.8	38.5	**38.5**	63.2	2.6	54.2	57.0	**57.0**	67.1	46.9	49.1	53.6	**53.6**
	0.6	FedQSN	58.6	42.8	27.4	59.3	59.3	50.3	42.8	16.4	49.5	49.5	61.2	27.1	53.4	60.2	**60.2**	67.3	67.5	52.6	64.3	67.5
		CloakFL	59.4	28.9	36.2	48.4	**48.4**	51.2	27.8	21.6	43.4	**43.4**	62.2	7.1	59.3	61.0	61.0	67.8	51.5	56.4	57.7	**57.7**

Table 3: Multi-client recovery performance (%) across cross-device and cross-silo settings. Proxy reports the mean performance across participating clients, while Coll. Proxy, Coll. Train, and Coll. FT report the mean across the evaluated colluding groups. Max denotes the maximum of the four means. Underlined values exceed the corresponding Server performance.

Env.	β	Method	CIFAR-100						Tiny-ImageNet						E2E						20 Newsgroups					
			Server	Proxy	Coll. Proxy	Coll. Train	Coll. FT	Max	Server	Proxy	Coll. Proxy	Coll. Train	Coll. FT	Max	Server	Proxy	Coll. Proxy	Coll. Train	Coll. FT	Max	Server	Proxy	Coll. Proxy	Coll. Train	Coll. FT	Max
Cross-device	0.1	FedQSN	34.2	26.3	28.0	29.4	38.7	38.7	29.1	26.7	26.3	20.1	31.9	31.9	61.0	54.8	57.1	57.4	62.0	62.0	59.7	59.7	59.6	38.1	61.1	61.1
		CloakFL	37.9	13.3	30.7	34.3	37.1	**37.1**	29.6	7.1	23.0	20.9	26.9	**26.9**	62.9	18.5	53.4	60.4	60.0	**60.4**	62.0	34.3	49.0	48.7	50.1	**50.1**
	0.3	FedQSN	50.1	44.0	46.7	41.2	52.9	52.9	35.6	26.6	27.4	29.1	38.4	38.4	58.4	47.6	50.7	52.0	59.9	59.9	64.4	64.4	64.4	54.0	64.9	64.9
		CloakFL	51.2	21.5	38.6	43.6	45.1	**45.1**	36.2	13.3	30.2	31.0	33.8	**33.8**	60.3	1.0	33.8	57.4	48.8	**57.4**	65.5	50.8	60.8	62.7	61.4	**62.7**
	0.6	FedQSN	50.3	32.5	37.4	39.3	49.9	**49.9**	40.8	33.2	35.1	32.9	42.6	42.6	56.5	55.1	56.2	54.6	58.8	58.8	65.6	67.3	67.3	59.8	67.4	67.4
		CloakFL	54.3	24.7	47.4	46.4	50.5	50.5	41.1	16.7	35.3	36.1	39.1	**39.1**	59.6	5.4	40.6	55.3	47.2	**55.3**	66.0	53.6	63.1	64.0	63.4	**64.0**
Cross-silo	0.1	FedQSN	42.8	38.6	37.1	33.7	49.0	49.0	36.4	30.1	31.0	21.2	40.4	40.4	59.8	30.4	35.8	57.9	60.1	60.1	58.3	58.4	58.4	34.4	53.8	58.4
		CloakFL	43.1	20.3	36.6	34.2	44.7	**44.7**	37.3	15.1	30.2	27.8	37.3	**37.3**	60.9	0.0	45.5	57.5	59.2	**59.2**	60.0	31.8	48.2	46.1	49.4	**49.4**
	0.3	FedQSN	53.1	36.2	37.1	42.0	58.1	58.1	46.9	39.4	40.7	36.6	50.3	50.3	63.2	32.8	34.7	58.9	60.6	60.6	66.8	66.8	66.9	56.7	65.7	66.9
		CloakFL	53.8	22.7	44.1	45.2	53.2	**53.2**	47.9	21.8	40.9	39.5	47.4	**47.4**	63.2	2.6	17.6	58.4	46.2	**58.4**	67.1	46.9	58.5	62.2	61.6	**62.2**
	0.6	FedQSN	58.6	42.8	44.7	51.1	62.8	62.8	50.3	42.8	44.5	39.8	52.6	52.6	61.2	27.1	29.1	60.7	62.4	**62.4**	67.3	67.5	67.7	58.6	65.6	67.7
		CloakFL	59.4	28.9	51.4	53.1	58.1	**58.1**	51.2	27.8	45.6	43.2	50.9	**50.9**	62.2	7.1	56.4	62.5	61.0	62.5	67.8	51.5	58.8	62.6	61.5	**62.6**

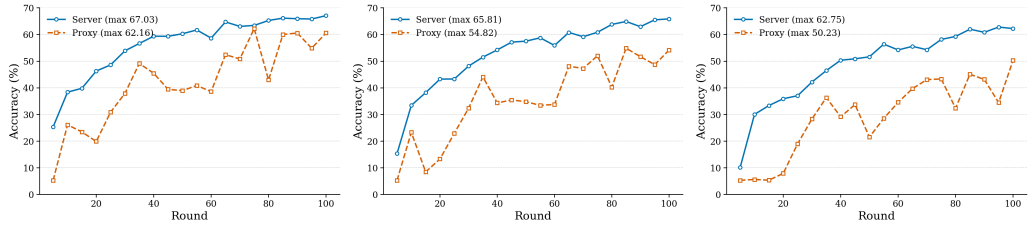
proxy. This comparison measures how much of the proxy model's performance can be recovered through the local training that is already performed as part of the standard FL process. *Proxy FT* also starts from the raw proxy, but applies one epoch of fine-tuning with a reduced learning rate. This setting evaluates whether an adversarial client can recover proxy performance by mitigating the effect of the injected perturbation while avoiding excessive client drift.

For CloakFL, Max remains below the corresponding server performance in all 24 settings, whereas for FedQSN, Max reaches or exceeds the server performance in 11 settings. Moreover, the performance difference between the server and Max is larger for CloakFL than for FedQSN in all 24 settings. These results show that CloakFL provides stronger model protection than FedQSN under Single-Client Recovery.

Multi-Client Recovery. For Multi-Client Recovery, we consider three variants: *Coll.Proxy*, *Coll.Train*, and *Coll.FT*. For each round with K participating clients, we set the coalition size to $0.3K$ and randomly sample 10 coalitions from the $\binom{K}{0.3K}$ possible coalitions. *Coll.Proxy* uniformly averages the raw proxy models of the colluding clients, measuring recovery through model averaging alone as client-specific perturbations may partially cancel during aggregation. *Coll.Train* instead averages the locally trained models after standard local training, approximating actual server-side aggregation. *Coll.FT* fine-tunes each raw proxy using the same configuration as *Proxy FT* and then uniformly averages the resulting models. This setting represents a stronger recovery scenario in which the injected perturbations can be further attenuated before aggregation.

Table 4: Effect of degradation strength on server and proxy performance (%) for MobileNetV2 on CIFAR-100 and DistilBERT on 20 Newsgroups in the cross-device setting with $\beta = 0.1$.

Strength	CIFAR-100												20 Newsgroups									
	Client Drift (CloakFL)			Gaussian			Quantization			Masking			Client Drift (CloakFL)			Gaussian			Quantization			Masking		
	ρ	Server	Proxy	ρ	Server	Proxy	ω	Server	Proxy	p_2	Server	Proxy	ρ	Server	Proxy	ρ	Server	Proxy	ω	Server	Proxy	p_2	Server	Proxy
Weak	0.5	41.62	26.43	0.9	42.41	40.82	Full	42.12	42.12	0.005	40.35	40.35	0.7	67.03	55.19	1.1	67.23	67.21	Full	67.49	67.49	0.004	64.23	64.13
Medium	0.6	38.64	19.21	1.0	40.77	32.98	8	43.70	43.65	0.010	34.17	26.25	0.8	65.81	50.06	1.2	67.17	66.10	8	52.77	61.91	0.008	59.68	59.69
Strong	0.7	37.90	13.29	1.1	3.23	2.44	4	9.02	7.64	0.020	26.42	16.07	0.9	62.19	37.72	1.3	23.46	14.60	4	50.82	60.25	0.016	53.07	52.48



(a) Client Drift ($\rho = 0.7$) **(b)** Client Drift ($\rho = 0.8$) **(c)** Client Drift ($\rho = 0.9$)

Figure 3: Ablation of client-drift perturbations across perturbation ratios ρ. Each panel shows the server and proxy accuracy over communication rounds.

Table 3 reports the mean performance across the 10 sampled coalitions for Coll. Proxy, Coll. Train, and Coll. FT. Proxy reports the mean raw proxy performance across individual clients in the final round, and Max denotes the highest mean performance among Proxy, Coll. Proxy, Coll. Train, and Coll. FT. For FedQSN, Max reaches or exceeds the corresponding server performance in 22 of 24 settings, whereas this occurs in only 3 settings for CloakFL. Notably, all three cases occur only in the cross-silo setting, where each client has a larger local dataset and thus more data available for model recovery. Moreover, the performance difference between the server and Max is larger for CloakFL than for FedQSN in all 24 settings. These results show that CloakFL provides stronger model protection than FedQSN even under the stronger Multi-Client Recovery setting.

5.4 Ablation Study

To examine the role of perturbation design, we compare CloakFL's client-drift-based perturbations with representative degradation mechanisms used by prior PMD methods: Gaussian noise from SentinelFL (Kwon & Choi, 2026), quantization from FedLPP and FedQSN (Zhu et al., 2024; Xu et al., 2025), and masking from FedQSN (Xu et al., 2025). As shown in Table 4, these optimization-agnostic mechanisms either provide limited proxy degradation or increasingly disrupt convergence as degradation becomes stronger, particularly for more complex models and tasks. In contrast, CloakFL consistently achieves a larger server–proxy gap while better preserving server performance.

Figure 3 further shows that ρ controls the protection–convergence trade-off, with larger values strengthening proxy degradation at some cost to server performance. Additional ablations are provided in Appendix C.

6 Conclusion

This paper addresses global-model exposure in federated learning through CloakFL, which distributes client-specific proxy models perturbed using convergence-aware signals derived from local training. By leveraging client drift, centering shared components, and calibrating perturbation magnitudes to recent update scales, CloakFL degrades exposed proxy models while preserving the convergence of the aggregated global model. Experiments across CNN and Transformer workloads show that CloakFL achieves larger server–proxy performance gaps than existing proxy-distribution baselines, while ablation and recovery studies highlight the importance of both perturbation design and robustness against fine-tuning and collusion. Overall, the results demonstrate the potential of optimization-aware proxy construction as a practical direction for protecting federated models without substantially sacrificing global performance.

Reproducibility Statement

Our code is available in an anonymous repository. Detailed experimental settings are provided in Appendix D, and the proof of our convergence result is provided in Appendix F.

AI Use Statement

We used generative AI tools to assist with manuscript review and refinement, as well as the review of mathematical derivations and proofs. The authors reviewed the resulting suggestions for correctness and consistency and incorporated revisions where appropriate. We take full responsibility for the final manuscript, including its technical claims, mathematical arguments, and experimental conclusions.

Ethics Statement

This work aims to protect federated-learning models from unauthorized exposure and reuse. Experiments use public benchmark datasets in a simulated federated-learning environment, without human-subject studies or new personal-data collection. Recovery attacks are evaluated only on models trained in our controlled experimental setup, not on third-party deployed systems. We acknowledge that these techniques could be misused beyond the defensive setting studied here.

References

Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=B7v4QMR6Z9w.

Samiul Alam, Luyang Liu, Ming Yan, and Mi Zhang. FedRolex: Model-heterogeneous federated learning with rolling sub-model extraction. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/bf5311df07f3efce97471921e6d2f159-Abstract.html.

Xiaoyu Cao, Minghong Fang, Jia Liu, and Neil Zhenqiang Gong. FLTrust: Byzantine-robust federated learning via trust bootstrapping. In *28th Annual Network and Distributed System Security Symposium (NDSS)*, 2021.

Jinyin Chen, Mingjun Li, Yao Cheng, and Haibin Zheng. Fedright: An effective model copyright protection for federated learning. *Computers & Security*, 2024. doi: 10.1016/j.cose.2023.103653. URL https://doi.org/10.1016/j.cose.2023.103653.

Enmao Diao, Jie Ding, and Vahid Tarokh. HeteroFL: Computation and communication efficient federated learning for heterogeneous clients. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=TNkPBBYFkXg.

Chenhe Dong, Yuexiang Xie, Bolin Ding, Ying Shen, and Yaliang Li. Tunable soft prompts are messengers in federated learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 14665–14675. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.976. URL https://aclanthology.org/2023.findings-emnlp.976/.

Clement Fung, Chris J. M. Yoon, and Ivan Beschastnikh. The limitations of federated learning in sybil settings. In *23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020)*, pp. 301–316, San Sebastian, October 2020. USENIX Association. URL https://www.usenix.org/conference/raid2020/presentation/fung.

Kuangpu Guo, Qingyan Zheng, Jian Liang, Yongcan Yu, Zilei Wang, Ran He, and Tieniu Tan. On the vulnerability of parameter-level defenses to model merging. In *Computer Vision – ECCV 2026*, pp. 455–472. Springer Nature Switzerland, 2026. doi: 10.1007/978-3-032-37335-9_25. URL https://doi.org/10.1007/978-3-032-37335-9_25.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification, 2019. URL https://arxiv.org/abs/1909.06335.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Weizhao Jin, Yuhang Yao, Shanshan Han, Jiajun Gu, Carlee Joe-Wong, Srivatsan Ravi, Salman Avestimehr, and Chaoyang He. Fedml-he: An efficient homomorphic-encryption-based privacy-preserving federated learning system. *arXiv preprint arXiv:2303.10837*, 2023. URL https://arxiv.org/abs/2303.10837.

Wei Junhao, Yu Zhe, and Jun Sakuma. Disrupting model merging: A parameter-level defense without sacrificing accuracy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17698–17707, 2025.

Peter Kairouz, H. Brendan McMahan, et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021. doi: 10.1561/2200000083. URL https://arxiv.org/abs/1912.04977.

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

Minjae Kim, Sangyoon Yu, Suhyun Kim, and Soo-Mook Moon. DepthFL: Depthwise federated learning for heterogeneous clients. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=pf8RIZTMU58.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations (ICLR)*, 2015. URL https://arxiv.org/abs/1412.6980.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Ajit Kumar and Bong Jun Choi. Global model privacy protection mechanism in federated learning. In *2024 International Conference on Information Networking (ICOIN)*, pp. 398–402. IEEE, 2024. doi: 10.1109/ICOIN59985.2024.10572139.

Daeho Kwon and Bong Jun Choi. SentinelFL: Noise-driven global model protection against model theft in federated learning. In *2026 40th International Conference on Information Networking (ICOIN)*. IEEE, 2026. URL https://d2j16w31g89z0j.cloudfront.net/site/icoin2026/abs/P3-17.pdf.

Ken Lang. Newsweeder: Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning*, pp. 331–339, 1995.

Ya Le and Xuan Yang. Tiny ImageNet visual recognition challenge. Cs231n course project report, Stanford University, 2015. URL https://cs231n.stanford.edu/reports/2015/pdfs/yle_project.pdf.

Bowen Li, Lixin Fan, Hanlin Gu, Jie Li, and Qiang Yang. Fedipr: Ownership verification for federated deep neural network models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4521–4536, 2023. doi: 10.1109/TPAMI.2022.3195956. URL https://doi.org/10.1109/TPAMI.2022.3195956.

Qinfeng Li, Miao Pan, Jintao Chen, Fu Teng, Zhiqiang Shen, Ge Su, Hao Peng, and Xuhong Zhang. Do not merge my model! safeguarding open-source LLMs against unauthorized model merging. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(37):31662–31670, 2026. doi: 10.1609/aaai.v40i37.40433. URL https://doi.org/10.1609/aaai.v40i37.40433.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems*, volume 2, pp. 429–450, 2020a. URL https://proceedings.mlsys.org/paper_files/paper/2020/file/1f5fe83998a09396ebe6477d9475ba0c-Paper.pdf.

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.

Zhong Li, Xiaoyan Chen, Jian Liu, and Zhe Wang. Copyright protection framework for federated learning models against collusion attacks. *Information Sciences*, 2024. URL https://www.sciencedirect.com/science/article/pii/S0020025524010752.

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

Anh Nguyen, Nhan Tran, Mohammad Mofrad, Khaled Benkrid, Shuguang Cui, Kaoru Ota, Mianxiong Dong, and Victor C. M. Leung. Securefl: Privacy preserving federated learning with sgx and trustzone. In *Proceedings of the 6th ACM/IEEE Symposium on Edge Computing (SEC)*, pp. 55–67, 2021. doi: 10.1145/3453142.3491287. URL https://doi.org/10.1145/3453142.3491287.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. The e2e dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pp. 201–206, 2017. doi: 10.18653/v1/W17-5525. URL https://aclanthology.org/W17-5525/.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, 2002. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040/.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9, 2019.

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2020. URL https://arxiv.org/abs/1910.01108.

Shuo Shao, Wenyuan Yang, Hanlin Gu, Zhan Qin, Lixin Fan, Qiang Yang, and Kui Ren. Fedtracker: Furnishing ownership verification and traceability for federated learning model. *IEEE Transactions on Dependable and Secure Computing*, 22(1):114–131, 2025. doi: 10.1109/TDSC.2024.3390761. URL https://doi.org/10.1109/TDSC.2024.3390761.

Muhammad Shayan, Clement Fung, Chris J. M. Yoon, and Ivan Beschastnikh. Biscotti: A ledger for private and secure peer-to-peer machine learning. *arXiv preprint arXiv:1811.09904*, 2018.

Buse G. A. Tekgul, Yuxi Xia, Samuel Marchal, and N. Asokan. Waffle: Watermarking in federated learning. *arXiv preprint arXiv:2008.07298*, 2020. URL https://arxiv.org/abs/2008.07298.

Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H. Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Zihao Wang, Enneng Yang, Lu Yin, Shiwei Liu, and Li Shen. Model unmerging: Making your models unmergeable for secure model sharing, 2025. URL https://arxiv.org/abs/2509.01548.

Kang Wei, Jun Li, Ming Ding, Chuan Ma, Howard H. Yang, Farhad Farokhi, Shi Jin, Tony Q. S. Quek, and H. Vincent Poor. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, 15:3454–3469, 2020. doi: 10.1109/TIFS.2020.2988575.

Zhibo Xu, JianHao Zhu, Jingwen Xu, Changze Lv, Zhenghua Wang, Zisu Huang, Xiaohua Wang, Muling Wu, Qi Qian, Xiaoqing Zheng, and Xuanjing Huang. Enhancing model privacy in federated learning with random masking and quantization. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 11804–11816. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.632. URL https://aclanthology.org/2025.findings-emnlp.632/.

Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5650–5659. PMLR, 2018. URL https://proceedings.mlr.press/v80/yin18a.html.

JianHao Zhu, Changze Lv, Xiaohua Wang, Muling Wu, Wenhao Liu, Tianlong Li, Zixuan Ling, Cenyuan Zhang, Xiaoqing Zheng, and Xuanjing Huang. Promoting data and model privacy in federated learning through quantized LoRA. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 10501–10512. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.615. URL https://aclanthology.org/2024.findings-emnlp.615/.

A Discussions & Limitations

Toward improved proxy model distribution. Our results suggest that there is still substantial room to improve both perturbation design and aggregation strategies for proxy model distribution. To the best of our knowledge, CloakFL is the first PMD method to construct perturbations from optimization information obtained during federated training. We therefore view this work not as the final design point, but as a first step toward a broader class of convergence-aware protection mechanisms. More effective and more robust variants may be possible through better perturbation directions, more principled client-specific scaling rules, or aggregation methods explicitly co-designed with the protection objective.

Combining PMD with complementary protections. More broadly, PMD is a promising direction for model protection in federated learning. It is unlikely to address every model-exposure risk on its own, but it can serve as a practical component of a layered defense strategy. In particular, PMD could be combined with complementary techniques such as watermarking, hardware-assisted protections, and cryptographic mechanisms (Tekgul et al., 2020; Li et al., 2023; Shao et al., 2025; Chen et al., 2024; Li et al., 2024; Nguyen et al., 2021; Jin et al., 2023) to provide stronger end-to-end protection for deployed federated learning systems.

Calibration to a desired protection level. In practice, the server may want to enforce a specific operating point, such as keeping raw-proxy inference performance below a target threshold while limiting the drop in final server performance. Compared with prior PMD baselines, CloakFL reduces tuning complexity because the main protection-convergence trade-off is governed by a single hyperparameter, the perturbation ratio ρ, searched over a relatively narrow range. Nevertheless, ρ is not a directly interpretable protection parameter. The same value of ρ can lead to different proxy degradation depending on the model architecture, task, client data heterogeneity, optimizer, and training stage. As a result, even though CloakFL is easier to tune than existing methods, the server

may still find it difficult to identify a configuration that achieves a desired protection-performance level. Designing adaptive controllers that automatically tune perturbation strength to meet a target protection-performance profile is an important direction for future work.

Cold-start calibration. CloakFL calibrates perturbation magnitudes from recent client-update norms. For newly participating clients, the current design bootstraps this scale using a stored update norm from another client. This is a practical initialization, but it may be imperfect when client optimization dynamics are highly heterogeneous, since the borrowed scale may not accurately reflect the target client's local training scale. Developing more principled cold-start calibration rules for newly participating clients is an important direction for future work.

Theory-practice gap. Our theoretical analysis is intended to provide analytical intuition for the proposed mechanism and therefore adopts a simplified setting relative to the experiments. In particular, the convergence result considers strongly convex and smooth objectives, full client participation, and moment bounds on the perturbations, whereas the empirical evaluation involves non-convex CNN and Transformer models, partial participation, and layer-wise perturbation scaling. Such abstraction is common in federated learning (Li et al., 2020b;a; Karimireddy et al., 2020; Acar et al., 2021; Wang et al., 2020), where tractable analyses are often used to clarify the role of a new design choice. We therefore interpret the theory as suggesting that keeping the perturbation norm small helps preserve optimization stability, while a tighter treatment of non-convex objectives, partial participation, and more practical perturbation schemes remains an important direction for future work.

Robustness under multi-client recovery. Although CloakFL is more robust than the PMD baseline under Multi-Client Recovery, model averaging can still recover performance close to or above that of the server model in some settings. Thus, CloakFL improves robustness to bounded collusion but does not prevent recovery entirely. Related studies, including PaRaMS, MergeLock, MergeBarrier, and ARF, attempt to prevent model merging by altering parameter geometry while preserving standalone model utility (Junhao et al., 2025; Wang et al., 2025; Li et al., 2026; Guo et al., 2026). Adapting these ideas to FL while preserving local training and legitimate server aggregation is an important direction for future work.

B Related Work

Federated learning has focused mainly on protecting client data (Wei et al., 2020), while protecting the model itself has received less attention. Because model parameters are repeatedly distributed to many participants, the global model becomes a broad surface for intellectual-property leakage and misuse. Accordingly, prior work on model protection in federated learning has explored several directions, including ownership verification, leak traceability, system-level confidentiality, and, more recently, proxy model distribution. This section reviews these lines of work to provide an overall view of model protection in federated learning.

B.1 Model IP Protection in Federated Learning

Prior research on model IP protection in federated learning can be broadly grouped into ownership-verification and traceability-oriented methods and system-level confidentiality mechanisms.

Ownership verification and traceability methods focus on establishing ownership of a trained model or identifying the source of leakage after exposure. WAFFLE (Tekgul et al., 2020) introduces watermarking for federated learning by embedding a resilient server-side watermark through retraining after aggregation. FedIPR (Li et al., 2023) allows clients to embed and verify private watermarks without revealing local data or watermark secrets. FedTracker (Shao et al., 2025) further combines global watermarking with fingerprints for leak tracing, while related approaches such as FedRight (Chen et al., 2024) and FedFP (Li et al., 2024) strengthen black-box verification and collusion-aware attribution. These methods can help recover value after leakage through ownership claims and attribution, but they do not prevent the leaked model itself from retaining substantial value and continuing to be exploited once exposed.

System-level confidentiality methods aim to reduce model exposure during training through secure execution or encrypted computation. TEE-based confidential FL systems such as SecureFL (Nguyen

et al., 2021) protect the training process by executing local and global computation inside trusted execution environments. Homomorphic-encryption-based systems such as FedML-HE (Jin et al., 2023) protect model updates during aggregation and improve practicality through selective encryption of sensitive parameters. These approaches can offer stronger confidentiality guarantees, but they typically require additional trust assumptions, hardware support, or cryptographic overhead.

B.2 Proxy Model Distribution

Proxy model distribution (PMD) methods (Dong et al., 2023; Zhu et al., 2024; Xu et al., 2025) aim to reduce the exposure risk of the clean global model by altering what is sent to clients during training. Rather than exposing the clean model directly, these methods redesign the models distributed to clients so that leaked proxies are less useful to an attacker.

Model-heterogeneous FL methods commonly distribute client-specific subnetworks to accommodate heterogeneous computational capabilities (Diao et al., 2021; Alam et al., 2022; Kim et al., 2023). However, prior work has shown that subnetwork distribution alone may provide limited model protection because individual subnetworks can retain accuracy comparable to that of the global model (Kumar & Choi, 2024).

A central challenge in PMD is to degrade the exposed model while preserving the usefulness of client updates for global training. Existing methods differ mainly in how they construct the proxy and how aggressively they degrade the model before distribution.

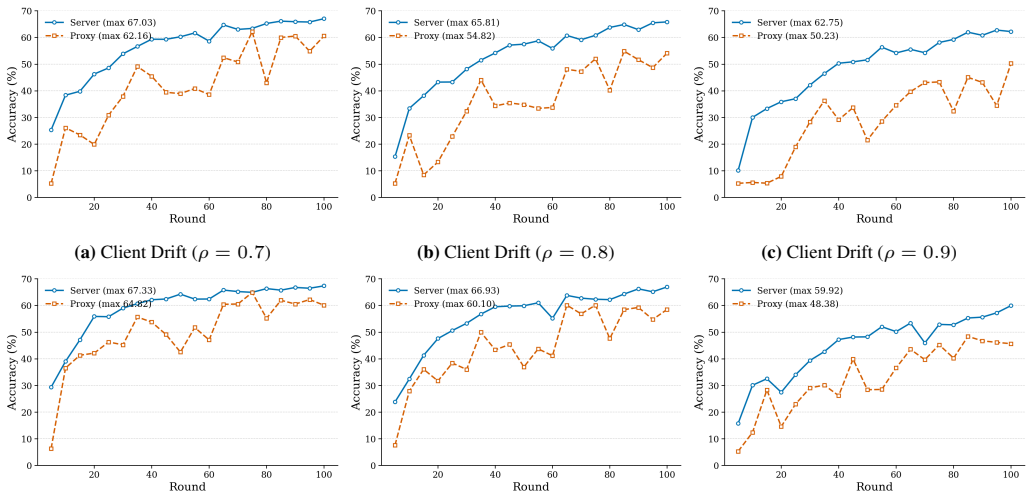
FedSP (Dong et al., 2023) presents an early PMD-style approach for federated foundation models by using tunable soft prompts as messengers between the server and clients, thereby avoiding direct distribution of the full model. However, FedSP relies on auxiliary proxy construction and server-side alignment procedures that require additional labeled data on the server, which limits practicality in privacy-sensitive federated settings.

FedLPP (Zhu et al., 2024) removes the dependence on an auxiliary server-side dataset by distributing block-wise quantized LoRA (Hu et al., 2022) parameters. More specifically, the server keeps the public backbone frozen and broadcasts a low-bit proxy of the trainable LoRA matrices, so clients update a degraded proxy model while still producing informative gradient estimates and while reducing communication cost. The paper evaluates $w \in \{1, 2, 3\}$ quantization bits and reports 2-bit quantization as the best trade-off between protection and convergence: 1-bit quantization is often too coarse for accuracy, whereas 3-bit quantization can leave the proxy too close to the server-side model to reliably preserve model privacy.

FedQSN (Xu et al., 2025) further strengthens proxy degradation by combining hierarchical random masking with quantization. In particular, it first applies a server-side mask to hide a fixed subset of parameters from all clients, which helps prevent any single client or colluding group from reconstructing the full model. It then applies an additional client-specific mask before each transmission, so different clients train different masked subnetworks in each round. The resulting doubly masked model is finally quantized before distribution, producing a more aggressively degraded proxy than FedLPP while still allowing the server to aggregate informative updates over time.

SentinelFL (Kwon & Choi, 2026) injects Gaussian noise into distributed models and controls its magnitude to reduce their utility while maintaining global-model performance.

Although FedSP, FedLPP, SentinelFL and FedQSN use different mechanisms, they all primarily aim to reduce the inference performance of the distributed proxy model itself. These methods focus on degrading the deployed proxy, with less attention to how such degradation affects local training and the overall federated learning process. As a result, stronger degradation can more severely distort local optimization, which in turn impairs global convergence, making the trade-off between protection and convergence harder to manage.



(a) Client Drift ($\rho = 0.7$) **(b)** Client Drift ($\rho = 0.8$) **(c)** Client Drift ($\rho = 0.9$)

(d) Local Update ($\rho = 0.7$) **(e)** Local Update ($\rho = 0.8$) **(f)** Local Update ($\rho = 0.9$)

Figure 4: Ablation of perturbation directions across perturbation ratios ρ. The top row uses centered client drift, whereas the bottom row uses raw local updates. Each panel shows the server and proxy accuracy over communication rounds.

C Additional Ablation Study

C.1 Centering

Using raw local updates as perturbation directions is less effective because they retain the shared component associated with global optimization. Centering the local updates removes this shared component, isolating client drift and thereby producing a more effective perturbation direction.

C.2 Perturbation Cancellation During Aggregation

Small individual perturbation norms alone are insufficient to preserve global convergence; their aggregate effect must also remain small. We therefore compare CloakFL's client-specific centered directions with a baseline that applies the same centered direction to every client.

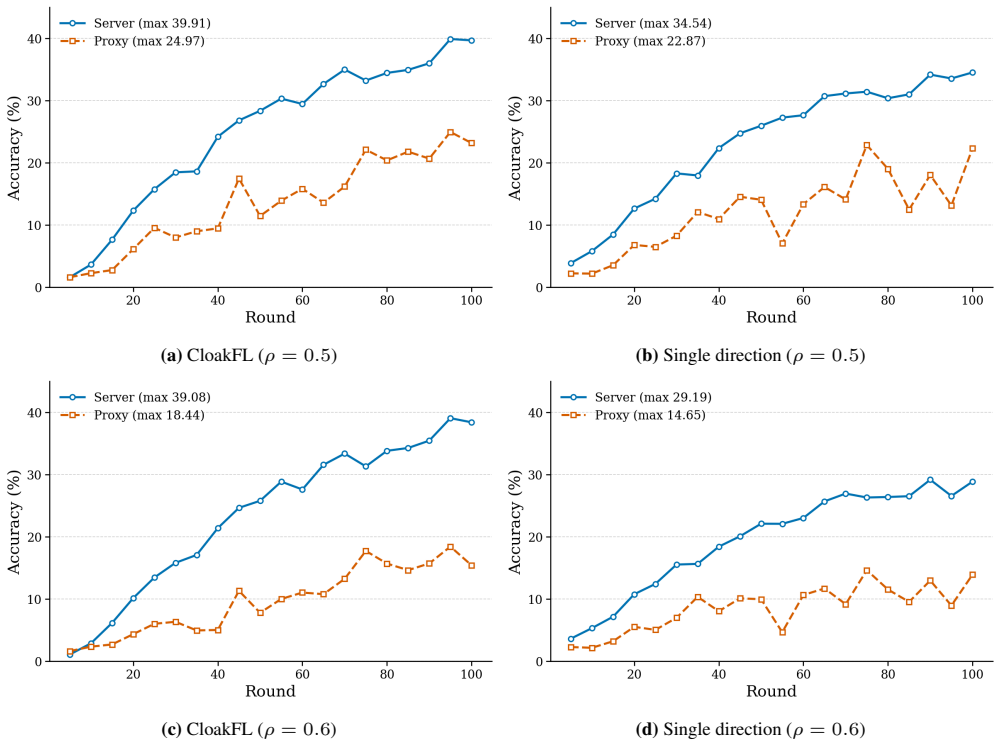
As shown in Figure 5, using a shared direction consistently impairs global-model convergence. Thus, beyond perturbation magnitude and centering, client-specific direction diversity is important because it allows the perturbation effects to cancel during aggregation.

D Detailed Experimental Settings

We evaluate four model-dataset pairs: MobileNetV2 on CIFAR-100, ResNet-50 on Tiny-ImageNet, DistilBERT on 20 Newsgroups, and GPT-2 Small on E2E. We consider both cross-device and cross-silo FL. The cross-device setting uses 100 clients with 10% participation per round, whereas the cross-silo setting uses 20 clients with 50% participation per round. Client datasets have equal sizes.

For non-IID partitioning, we use a Dirichlet distribution with $\beta \in \{0.1, 0.3, 0.6\}$. For CIFAR-100, Tiny-ImageNet, and 20 Newsgroups, the Dirichlet split is applied over class labels. For E2E, it is applied over source inputs, yielding source-concentrated client partitions. All compared methods use the same client partitions and client-sampling schedules for each experimental setting.

For CIFAR-100 and Tiny-ImageNet, clients use SGD with a learning rate of 0.1, batch size 64, and 5 local epochs per communication round. CIFAR-100 is trained for 30 and 100 communication rounds in the cross-silo and cross-device settings, respectively, while Tiny-ImageNet is trained for 50 and 150 rounds.



(a) CloakFL ($\rho = 0.5$)

(b) Single direction ($\rho = 0.5$)

(c) CloakFL ($\rho = 0.6$)

(d) Single direction ($\rho = 0.6$)

Figure 5: Ablation of perturbation-direction diversity on CIFAR-100. The left column uses CloakFL's client-specific centered directions, whereas the right column reuses a single centered direction for all clients. Client-specific directions preserve global convergence more effectively because their effects cancel during aggregation.

For 20 Newsgroups and E2E, clients use Adam (Kingma & Ba, 2015) with one local epoch per communication round. DistilBERT uses a learning rate of 10^{-4}, maximum sequence length 256, and batch size 16, and is trained for 30 and 100 rounds in the cross-silo and cross-device settings, respectively. GPT-2 Small uses a learning rate of 10^{-3}, maximum sequence length 512, and batch size 4, and is trained for 10 and 30 rounds, respectively. We apply exponential learning-rate decay with a factor of 0.999 across communication rounds for all tasks.

Within each task, all methods share the same optimizer, training configuration, client partitions, and sampling schedule. Unless otherwise stated, proxy and recovery performance are evaluated using models from the final communication round.

E Detailed Experimental Results

This appendix reports the full set of operating points used to study the protection–convergence trade-off. Each paired entry is written as maximum/minimum performance across the evaluated individual clients or colluding groups. CloakFL rows are shaded for readability.

Table 5: Detailed final-round accuracy (%) for 20 Newsgroups in the cross-device setting.

β	Method	Server	Proxy	Single-client recovery		Multi-client recovery		
				Proxy FT	Proxy Train	Coll. Proxy	Coll. Train	Coll. FT
0.1	FedAvg	67.34	67.34 / 67.34	67.63 / 65.57	59.11 / 37.39	67.34 / 67.34	65.23 / 53.04	68.23 / 67.35
	Quant. (8-bit)	52.77	61.91 / 61.91	59.73 / 56.29	35.93 / 21.32	61.91 / 61.91	46.26 / 31.65	62.72 / 59.28
	Quant. (full precision)	67.49	67.49 / 67.49	67.51 / 65.68	51.61 / 37.40	67.49 / 67.49	61.42 / 52.97	67.95 / 67.27
	Quant.&Mask (full, 0.004)	64.23	64.17 / 64.11	64.14 / 61.36	40.77 / 30.28	64.15 / 64.14	50.32 / 45.35	65.12 / 64.25
	Quant.&Mask (full, 0.008)	59.68	59.78 / 59.52	59.44 / 54.65	36.43 / 17.30	59.68 / 59.52	42.49 / 31.86	61.74 / 59.93
	Quant.&Mask (full, 0.016)	53.07	52.62 / 52.32	53.16 / 49.18	31.61 / 16.25	52.52 / 52.47	41.10 / 29.21	57.13 / 55.28
	CloakFL ($\rho = 0.8$)	65.80	46.12 / 33.05	49.92 / 35.10	52.12 / 39.46	63.45 / 49.06	60.30 / 56.97	63.95 / 51.02
	CloakFL ($\rho = 0.9$)	61.98	38.65 / 26.22	43.02 / 29.02	46.55 / 28.69	56.74 / 36.50	51.18 / 46.03	57.85 / 38.38
	CloakFL ($\rho = 1.0$)	37.89	32.70 / 13.30	32.16 / 22.40	29.50 / 15.44	35.22 / 21.15	34.63 / 25.16	35.59 / 26.05
0.3	FedAvg	68.61	68.61 / 68.61	68.91 / 68.28	61.67 / 57.00	68.61 / 68.61	67.92 / 64.91	69.07 / 68.84
	Quant.&Mask (full, 0.004)	67.45	67.55 / 67.38	68.03 / 66.45	59.67 / 45.78	67.46 / 67.39	64.71 / 61.43	68.06 / 67.57
	Quant.&Mask (full, 0.008)	64.41	64.52 / 64.15	65.37 / 61.99	51.71 / 26.61	64.43 / 64.35	59.16 / 51.33	65.41 / 64.11
	Quant.&Mask (full, 0.016)	61.70	61.83 / 61.42	62.44 / 58.13	48.11 / 22.21	61.87 / 61.64	56.77 / 43.16	62.44 / 61.25
	CloakFL ($\rho = 0.8$)	67.34	61.44 / 58.92	62.29 / 60.52	63.46 / 55.70	65.91 / 61.79	67.15 / 65.80	66.37 / 63.04
	CloakFL ($\rho = 0.9$)	65.49	53.43 / 48.55	57.34 / 52.03	60.06 / 49.50	63.99 / 55.34	64.68 / 61.13	64.34 / 56.28
	CloakFL ($\rho = 1.0$)	57.90	37.08 / 27.38	39.42 / 35.44	44.53 / 34.24	48.57 / 44.03	56.55 / 49.61	49.24 / 46.46
0.6	FedAvg	69.64	69.64 / 69.64	69.57 / 69.46	64.11 / 58.36	69.64 / 69.64	68.39 / 66.38	69.98 / 69.70
	Quant.&Mask (8-bit, 0.001)	65.64	67.38 / 67.22	66.61 / 64.75	55.79 / 46.76	67.34 / 67.26	64.13 / 54.33	67.57 / 67.03
	Quant.&Mask (8-bit, 0.002)	66.10	67.26 / 67.15	66.34 / 64.76	53.72 / 49.06	67.39 / 67.23	63.18 / 54.50	67.70 / 67.19
	CloakFL ($\rho = 0.8$)	67.80	60.93 / 55.10	60.90 / 58.31	63.57 / 56.03	67.18 / 63.91	66.56 / 65.03	67.26 / 64.35
	CloakFL ($\rho = 0.9$)	65.99	55.87 / 51.18	59.41 / 53.58	60.53 / 53.40	64.39 / 60.86	64.55 / 63.53	64.34 / 61.52
	CloakFL ($\rho = 1.0$)	59.44	43.47 / 29.37	44.44 / 32.22	46.32 / 39.86	46.24 / 41.42	49.04 / 44.26	46.65 / 42.25

Table 6: Detailed final-round accuracy (%) for 20 Newsgroups in the cross-silo setting.

β	Method	Server	Proxy	Single-client recovery		Multi-client recovery		
				Proxy FT	Proxy Train	Coll. Proxy	Coll. Train	Coll. FT
0.1	FedAvg	65.06	65.06 / 65.06	66.02 / 55.82	50.03 / 13.65	65.06 / 65.06	54.38 / 49.02	66.32 / 64.42
	Quant. (4-bit)	52.35	50.72 / 50.72	38.85 / 28.16	21.15 / 12.43	50.72 / 50.72	40.88 / 32.99	51.39 / 43.60
	Quant. (8-bit)	60.46	63.98 / 63.98	55.11 / 43.80	28.36 / 21.31	63.98 / 63.98	47.23 / 42.33	64.88 / 62.45
	Quant.&Mask (8-bit, 0.016)	61.60	62.98 / 62.43	50.73 / 31.72	24.92 / 15.88	62.87 / 62.55	48.67 / 42.14	61.91 / 59.41
	Quant.&Mask (8-bit, 0.032)	58.26	58.46 / 58.32	38.64 / 21.55	23.18 / 11.67	58.56 / 58.18	39.58 / 31.04	57.24 / 51.94
	Quant.&Mask (8-bit, 0.064)	47.42	48.15 / 47.73	29.25 / 16.58	22.97 / 12.03	48.01 / 47.73	28.61 / 22.85	48.09 / 39.92
	CloakFL ($\rho = 0.7$)	62.04	50.11 / 40.37	49.58 / 39.87	35.90 / 23.38	57.17 / 49.79	53.07 / 49.28	57.65 / 52.64
	CloakFL ($\rho = 0.8$)	60.04	32.57 / 31.40	41.00 / 33.98	33.64 / 17.84	55.60 / 41.48	47.72 / 42.94	55.66 / 43.39
0.3	FedAvg	69.21	69.21 / 69.21	69.28 / 65.68	61.39 / 46.93	69.21 / 69.21	66.86 / 61.76	69.09 / 68.84
	Quant.&Mask (8-bit, 0.008)	67.51	68.03 / 67.46	66.81 / 59.09	55.78 / 45.22	67.96 / 67.74	63.81 / 59.84	67.64 / 66.81
	Quant.&Mask (8-bit, 0.016)	66.82	67.05 / 66.54	64.72 / 55.70	54.58 / 38.73	66.97 / 66.69	59.23 / 53.01	66.64 / 65.02
	Quant.&Mask (8-bit, 0.032)	64.71	64.75 / 64.41	60.32 / 50.03	50.78 / 33.11	64.82 / 64.66	56.47 / 43.30	63.90 / 62.29
	CloakFL ($\rho = 0.7$)	68.04	54.46 / 53.52	60.37 / 55.70	61.43 / 48.29	65.60 / 60.13	65.59 / 60.67	66.58 / 63.62
	CloakFL ($\rho = 0.8$)	67.06	47.60 / 45.75	58.23 / 50.37	58.84 / 39.19	64.43 / 52.15	64.01 / 59.35	65.40 / 58.94
0.6	FedAvg	68.84	68.84 / 68.84	69.17 / 66.16	63.78 / 56.78	68.84 / 68.84	64.94 / 64.34	68.80 / 68.53
	Quant.&Mask (8-bit, 0.008)	68.20	68.87 / 68.52	67.90 / 64.47	59.40 / 48.71	68.85 / 68.79	61.55 / 60.01	67.90 / 67.22
	Quant.&Mask (8-bit, 0.016)	67.30	67.60 / 67.37	66.56 / 62.81	54.93 / 49.26	67.74 / 67.63	59.25 / 57.87	66.38 / 65.02
	Quant.&Mask (8-bit, 0.032)	65.44	65.06 / 64.96	62.44 / 56.85	52.80 / 41.83	65.15 / 65.04	56.32 / 53.90	63.24 / 61.02
	CloakFL ($\rho = 0.7$)	68.63	62.21 / 57.21	64.33 / 60.71	59.01 / 53.43	64.52 / 60.97	65.19 / 62.68	65.41 / 61.86
	CloakFL ($\rho = 0.8$)	67.75	54.24 / 49.71	60.09 / 55.40	61.29 / 53.49	62.07 / 56.59	64.19 / 60.90	63.20 / 58.88

Table 7: Detailed final-round BLEU for E2E in the cross-device setting.

β	Method	Server	Proxy	Single-client recovery		Multi-client recovery		
				Proxy FT	Proxy Train	Coll. Proxy	Coll. Train	Coll. FT
0.1	FedAvg	61.26	61.26 / 61.26	60.48 / 59.94	57.76 / 54.86	61.26 / 61.26	63.11 / 60.10	62.15 / 61.76
	Quant. (4-bit)	58.06	36.85 / 36.85	59.77 / 49.87	54.90 / 43.48	36.85 / 36.85	61.30 / 58.78	57.24 / 53.60
	Quant. (8-bit)	61.67	61.78 / 61.78	60.97 / 59.64	57.57 / 52.55	61.78 / 61.78	60.35 / 59.83	62.28 / 61.92
	Quant.&Mask (8-bit, 0.1)	63.09	62.69 / 60.06	60.53 / 57.75	51.64 / 47.82	62.80 / 61.56	62.87 / 62.23	63.54 / 61.39
	Quant.&Mask (8-bit, 0.2)	61.04	56.43 / 53.84	60.68 / 58.38	48.24 / 36.56	58.55 / 56.39	58.43 / 56.35	62.34 / 61.73
	Quant.&Mask (8-bit, 0.3)	52.58	39.81 / 29.96	56.07 / 53.70	42.58 / 29.98	42.91 / 41.01	55.29 / 50.16	57.86 / 53.99
	CloakFL ($\rho = 0.8$)	63.33	50.36 / 39.33	60.26 / 56.29	55.42 / 53.21	60.20 / 57.11	62.19 / 60.59	61.92 / 61.14
	CloakFL ($\rho = 0.9$)	62.90	28.74 / 8.77	61.26 / 54.35	56.76 / 51.79	55.05 / 50.31	62.73 / 58.32	62.57 / 58.38
	CloakFL ($\rho = 1.0$)	58.41	4.64 / 0.59	58.87 / 37.93	52.70 / 51.42	58.59 / 27.26	62.85 / 60.30	59.75 / 51.94
0.3	FedAvg	60.16	60.16 / 60.16	61.38 / 57.46	56.91 / 51.20	60.16 / 60.16	61.01 / 60.33	61.66 / 61.35
	Quant.&Mask (4-bit, 0.1)	56.84	53.24 / 52.42	59.58 / 56.31	51.01 / 42.89	54.16 / 53.60	57.06 / 47.80	61.28 / 60.17
	Quant.&Mask (4-bit, 0.2)	58.37	48.94 / 46.65	57.88 / 55.18	46.74 / 40.01	52.48 / 49.13	55.27 / 49.91	60.55 / 58.95
	Quant.&Mask (4-bit, 0.3)	53.30	30.51 / 29.22	50.77 / 45.21	36.41 / 22.64	37.72 / 35.03	44.37 / 38.73	52.53 / 52.10
	CloakFL ($\rho = 0.9$)	63.31	28.21 / 21.40	63.48 / 51.78	55.00 / 46.38	56.87 / 50.47	61.49 / 55.98	61.69 / 55.94
	CloakFL ($\rho = 1.0$)	60.34	3.12 / 0.00	57.93 / 44.35	50.57 / 44.97	43.14 / 16.50	59.44 / 54.04	51.91 / 44.11
	CloakFL ($\rho = 1.1$)	48.28	0.00 / 0.00	47.06 / 14.84	51.84 / 22.89	10.95 / 0.00	48.05 / 40.30	18.41 / 0.00
0.6	FedAvg	61.66	61.66 / 61.66	63.27 / 61.12	60.81 / 52.24	61.66 / 61.66	60.85 / 58.99	62.85 / 61.94
	Quant.&Mask (4-bit, 0.1)	56.48	56.38 / 53.95	59.20 / 54.67	49.87 / 35.54	56.29 / 56.09	55.95 / 53.82	59.04 / 58.60
	Quant.&Mask (4-bit, 0.2)	61.28	51.86 / 48.66	57.98 / 51.08	49.31 / 9.43	57.30 / 54.01	50.94 / 44.77	59.38 / 57.69
	CloakFL ($\rho = 0.9$)	58.23	28.52 / 18.78	60.38 / 52.43	57.04 / 43.85	58.19 / 51.31	60.76 / 57.09	61.40 / 58.04
	CloakFL ($\rho = 1.0$)	59.60	10.93 / 0.00	54.15 / 42.04	55.92 / 44.91	45.07 / 31.64	57.41 / 53.84	50.49 / 41.28
	CloakFL ($\rho = 1.1$)	47.81	0.51 / 0.00	48.66 / 11.40	56.38 / 23.29	34.43 / 17.07	48.50 / 42.77	38.41 / 37.80

Table 8: Detailed final-round BLEU for E2E in the cross-silo setting.

β	Method	Server	Proxy	Single-client recovery		Multi-client recovery		
				Proxy FT	Proxy Train	Coll. Proxy	Coll. Train	Coll. FT
0.1	Quant. (2-bit)	55.23	30.61 / 30.61	57.09 / 53.93	45.91 / 39.72	30.61 / 30.61	55.43 / 53.68	59.25 / 57.21
	Quant. (4-bit)	60.46	63.83 / 63.83	61.73 / 59.92	55.14 / 51.26	63.83 / 63.83	61.82 / 60.32	62.62 / 61.47
	Quant.&Mask (4-bit, 0.3)	63.81	55.34 / 52.71	62.76 / 59.44	49.17 / 48.51	56.00 / 54.18	63.06 / 61.62	65.07 / 64.37
	Quant.&Mask (4-bit, 0.4)	59.78	34.27 / 28.09	61.28 / 57.99	49.29 / 41.92	36.75 / 34.17	58.65 / 56.74	60.29 / 60.02
	CloakFL ($\rho = 0.9$)	60.93	0.00 / 0.00	58.72 / 48.85	59.33 / 48.51	50.76 / 40.72	57.82 / 57.22	60.83 / 57.67
	CloakFL ($\rho = 1.0$)	45.96	0.00 / 0.00	51.87 / 45.16	51.97 / 43.06	49.00 / 1.01	35.06 / 2.44	44.17 / 7.51
0.3	FedAvg	62.73	62.73 / 62.73	62.10 / 60.89	61.72 / 55.89	62.73 / 62.73	59.14 / 57.66	62.01 / 61.73
	Quant.&Mask (4-bit, 0.3)	65.42	54.95 / 50.10	62.38 / 61.12	53.35 / 48.63	55.93 / 53.56	61.91 / 54.98	65.41 / 64.30
	Quant.&Mask (4-bit, 0.4)	63.15	37.01 / 27.96	59.23 / 58.10	55.19 / 49.11	38.54 / 32.00	59.83 / 57.49	61.09 / 59.85
	CloakFL ($\rho = 0.8$)	61.70	29.29 / 0.00	62.71 / 60.24	60.19 / 55.66	60.55 / 45.64	62.04 / 58.65	62.80 / 58.99
	CloakFL ($\rho = 0.9$)	63.17	5.36 / 0.00	58.35 / 54.95	59.39 / 49.39	52.56 / 0.00	61.95 / 56.41	57.24 / 38.28
	CloakFL ($\rho = 1.0$)	54.15	0.61 / 0.00	48.90 / 40.04	56.79 / 48.37	34.75 / 0.00	49.30 / 0.56	33.14 / 0.00
0.6	FedAvg	62.28	62.28 / 62.28	63.04 / 62.61	59.94 / 59.16	62.28 / 62.28	64.14 / 63.72	63.75 / 62.63
	Quant.&Mask (8-bit, 0.3)	64.64	57.80 / 50.73	61.49 / 59.98	58.64 / 52.81	55.78 / 55.26	62.80 / 57.21	64.21 / 63.56
	Quant.&Mask (8-bit, 0.4)	61.17	33.59 / 23.17	61.36 / 59.58	55.72 / 51.82	29.74 / 27.99	62.01 / 58.32	62.78 / 62.10
	CloakFL ($\rho = 0.8$)	62.19	21.30 / 0.00	61.29 / 60.73	59.75 / 58.89	58.22 / 53.71	63.60 / 61.61	62.36 / 59.51
	CloakFL ($\rho = 0.9$)	59.66	0.62 / 0.00	60.54 / 55.17	54.07 / 53.42	58.13 / 51.78	59.55 / 56.43	60.68 / 53.34

Table 9: Detailed final-round accuracy (%) for CIFAR-100 in the cross-device setting.

β	Method	Server	Proxy	Single-client recovery		Multi-client recovery		
				Proxy FT	Proxy Train	Coll. Proxy	Coll. Train	Coll. FT
0.1	FedAvg	42.13	42.13 / 42.13	45.59 / 43.02	24.72 / 18.23	42.13 / 42.13	43.41 / 37.39	46.85 / 46.00
	Quant. (4-bit)	9.02	7.64 / 7.64	9.68 / 8.41	6.11 / 4.03	7.64 / 7.64	7.34 / 6.65	10.36 / 9.70
	Quant. (8-bit)	43.70	43.65 / 43.65	46.53 / 43.94	24.85 / 9.36	43.65 / 43.65	39.72 / 36.14	47.63 / 46.46
	Quant.&Mask (8-bit, 0.005)	40.35	40.50 / 40.25	43.97 / 41.81	24.19 / 9.04	40.38 / 40.30	38.16 / 32.75	45.00 / 43.91
	Quant.&Mask (8-bit, 0.01)	34.17	30.62 / 21.75	36.48 / 34.68	18.52 / 4.27	29.62 / 25.39	32.22 / 26.08	39.82 / 37.93
	Quant.&Mask (8-bit, 0.02)	26.42	19.79 / 13.07	30.22 / 26.14	12.90 / 2.15	17.63 / 15.03	25.23 / 21.25	31.79 / 28.55
	CloakFL ($\rho = 0.6$)	38.64	26.23 / 10.66	32.47 / 23.40	21.94 / 9.61	35.84 / 33.75	37.48 / 34.02	41.10 / 38.81
	CloakFL ($\rho = 0.7$)	37.90	19.15 / 8.53	26.79 / 17.09	20.07 / 9.84	33.86 / 28.47	35.95 / 31.15	38.41 / 35.61
0.3	FedAvg	54.36	54.36 / 54.36	55.68 / 53.72	40.13 / 6.25	54.36 / 54.36	52.05 / 37.39	56.17 / 55.55
	Quant.&Mask (8-bit, 0.01)	50.09	47.21 / 39.13	52.36 / 50.37	34.73 / 4.00	48.07 / 45.70	45.74 / 35.81	53.18 / 52.46
	Quant.&Mask (8-bit, 0.02)	44.51	36.27 / 22.17	46.64 / 43.59	26.86 / 1.85	34.93 / 31.76	41.39 / 28.02	47.28 / 45.99
	CloakFL ($\rho = 0.6$)	52.27	41.86 / 25.43	46.12 / 34.46	38.81 / 7.74	49.08 / 40.97	49.71 / 42.49	52.40 / 47.65
	CloakFL ($\rho = 0.7$)	51.23	34.81 / 8.54	40.11 / 23.06	36.87 / 10.55	46.53 / 32.28	47.53 / 40.60	50.10 / 41.39
0.6	FedAvg	57.19	57.19 / 57.19	58.09 / 56.33	45.36 / 2.00	57.19 / 57.19	53.22 / 31.43	58.40 / 57.74
	Quant.&Mask (8-bit, 0.01)	55.70	52.19 / 40.99	55.91 / 52.33	37.32 / 2.68	52.77 / 51.04	51.30 / 37.78	57.01 / 55.82
	Quant.&Mask (8-bit, 0.02)	50.27	41.81 / 28.44	50.56 / 46.33	31.44 / 1.50	39.33 / 36.14	45.94 / 32.93	51.11 / 49.03
	Quant.&Mask (8-bit, 0.03)	46.15	27.15 / 16.36	44.28 / 39.64	27.99 / 1.52	27.30 / 24.44	41.88 / 24.70	45.41 / 43.59
	CloakFL ($\rho = 0.6$)	55.68	46.64 / 13.49	49.68 / 25.38	42.05 / 8.78	53.96 / 48.09	52.75 / 44.34	54.90 / 51.66
	CloakFL ($\rho = 0.7$)	54.26	37.78 / 6.22	44.00 / 17.84	40.83 / 3.02	52.40 / 41.96	51.58 / 40.23	53.46 / 47.45

Table 10: Detailed final-round accuracy (%) for CIFAR-100 in the cross-silo setting.

β	Method	Server	Proxy	Single-client recovery		Multi-client recovery		
				Proxy FT	Proxy Train	Coll. Proxy	Coll. Train	Coll. FT
0.1	FedAvg	46.62	46.62 / 46.62	44.66 / 41.46	22.37 / 10.16	46.62 / 46.62	37.29 / 33.56	51.83 / 49.49
	Quant. (4-bit)	25.43	27.25 / 27.25	22.51 / 19.16	13.99 / 7.20	27.25 / 27.25	20.34 / 16.32	29.51 / 27.62
	Quant. (8-bit)	46.53	46.36 / 46.36	44.49 / 41.96	20.37 / 13.49	46.36 / 46.36	38.03 / 33.27	52.23 / 50.59
	Quant.&Mask (8-bit, 0.01)	42.83	40.31 / 37.45	41.97 / 39.27	19.39 / 4.27	38.35 / 36.00	35.97 / 31.36	50.03 / 47.86
	Quant.&Mask (8-bit, 0.02)	39.86	34.62 / 20.44	38.65 / 36.12	16.46 / 4.53	29.66 / 22.61	32.29 / 28.32	47.38 / 45.22
	CloakFL ($\rho = 0.6$)	43.06	23.83 / 18.02	36.23 / 28.73	20.32 / 11.05	41.43 / 31.85	35.53 / 32.56	45.84 / 42.41
	CloakFL ($\rho = 0.7$)	41.36	19.46 / 11.78	30.96 / 24.95	19.58 / 15.37	38.10 / 26.96	33.89 / 29.99	43.02 / 37.08
0.3	FedAvg	57.26	57.26 / 57.26	59.27 / 56.62	34.94 / 6.37	57.26 / 57.26	51.33 / 42.39	61.56 / 59.60
	Quant.&Mask (8-bit, 0.01)	54.74	52.34 / 45.62	56.76 / 54.87	29.54 / 15.54	49.45 / 45.28	47.67 / 41.01	60.35 / 59.27
	Quant.&Mask (8-bit, 0.02)	53.10	45.55 / 28.53	54.93 / 52.18	26.93 / 9.89	41.40 / 33.52	46.00 / 37.44	58.82 / 57.49
	Quant.&Mask (8-bit, 0.03)	49.11	33.60 / 9.91	53.30 / 50.39	24.47 / 18.29	27.54 / 16.39	42.86 / 34.48	57.03 / 56.01
	CloakFL ($\rho = 0.6$)	55.82	36.03 / 33.38	50.02 / 43.86	32.82 / 22.08	54.70 / 47.33	48.10 / 43.53	58.46 / 55.62
	CloakFL ($\rho = 0.7$)	53.81	25.26 / 18.38	43.86 / 38.30	31.45 / 11.41	50.88 / 38.95	47.02 / 42.34	56.39 / 50.94
0.6	FedAvg	62.09	62.09 / 62.09	62.83 / 61.30	42.24 / 30.24	62.09 / 62.09	57.45 / 53.53	64.50 / 63.87
	Quant.&Mask (8-bit, 0.01)	60.47	58.64 / 52.04	61.13 / 60.67	36.91 / 25.82	56.16 / 53.11	53.73 / 52.49	63.70 / 63.20
	Quant.&Mask (8-bit, 0.02)	58.60	52.26 / 35.68	60.00 / 58.66	34.45 / 20.24	48.73 / 41.80	52.34 / 49.06	63.08 / 62.25
	Quant.&Mask (8-bit, 0.03)	56.59	34.82 / 16.27	57.92 / 56.57	31.38 / 22.14	30.70 / 19.11	50.69 / 48.55	61.32 / 60.25
	CloakFL ($\rho = 0.6$)	60.30	44.36 / 32.96	54.27 / 52.49	40.12 / 27.45	58.97 / 49.68	54.32 / 52.77	61.63 / 59.33
	CloakFL ($\rho = 0.7$)	59.42	34.55 / 22.33	49.25 / 47.76	38.16 / 32.06	55.94 / 48.12	55.04 / 51.92	59.68 / 56.83

Table 11: Detailed final-round accuracy (%) for Tiny-ImageNet in the cross-device setting.

β	Method	Server	Proxy	Single-client recovery		Multi-client recovery		
				Proxy FT	Proxy Train	Coll. Proxy	Coll. Train	Coll. FT
0.1	FedAvg	36.01	36.01 / 36.01	36.82 / 32.52	15.49 / 2.19	36.01 / 36.01	30.14 / 23.56	37.93 / 36.83
	Quant. (4-bit)	4.06	2.52 / 2.52	3.84 / 2.46	3.00 / 1.30	2.52 / 2.52	3.30 / 2.41	4.32 / 3.92
	Quant. (8-bit)	36.06	36.07 / 36.07	36.70 / 32.86	14.78 / 4.79	36.07 / 36.07	30.79 / 26.38	37.85 / 36.69
	Quant.&Mask (8-bit, 0.01)	33.83	32.62 / 30.56	34.67 / 30.32	12.15 / 2.74	32.82 / 31.18	27.04 / 22.35	36.43 / 35.35
	Quant.&Mask (8-bit, 0.02)	29.07	27.41 / 26.23	31.21 / 26.59	9.26 / 2.50	27.34 / 25.58	21.80 / 18.09	32.46 / 31.25
	Quant.&Mask (8-bit, 0.03)	26.75	22.61 / 21.56	28.00 / 23.22	7.14 / 2.60	23.64 / 22.60	18.72 / 16.19	28.74 / 28.25
	CloakFL ($\rho = 0.6$)	32.92	13.96 / 9.67	24.18 / 19.70	12.56 / 3.98	29.61 / 26.56	25.70 / 22.33	32.33 / 30.67
	CloakFL ($\rho = 0.7$)	29.58	8.87 / 5.35	17.97 / 14.05	10.28 / 5.54	24.60 / 20.98	22.29 / 19.55	28.00 / 25.47
	CloakFL ($\rho = 0.8$)	24.55	4.53 / 2.66	11.62 / 8.02	7.79 / 4.75	19.83 / 13.13	17.69 / 16.15	22.24 / 18.48
0.3	FedAvg	39.56	39.56 / 39.56	42.19 / 41.34	25.26 / 8.14	39.56 / 39.56	38.54 / 35.89	43.06 / 42.46
	Quant.&Mask (8-bit, 0.02)	38.90	35.59 / 31.34	40.64 / 38.86	19.21 / 3.79	35.65 / 34.75	33.34 / 31.51	41.58 / 40.70
	Quant.&Mask (8-bit, 0.03)	35.60	29.55 / 23.03	37.60 / 35.46	16.80 / 4.73	28.69 / 25.96	30.21 / 27.80	38.51 / 38.29
	Quant.&Mask (8-bit, 0.05)	31.06	19.36 / 10.19	32.88 / 31.52	14.06 / 3.59	17.70 / 13.94	24.57 / 20.65	34.41 / 33.62
	CloakFL ($\rho = 0.6$)	38.24	24.40 / 9.90	31.33 / 27.36	21.38 / 12.18	36.06 / 33.13	34.32 / 32.86	38.23 / 37.13
	CloakFL ($\rho = 0.7$)	36.24	17.77 / 6.43	26.30 / 21.61	19.60 / 10.40	33.61 / 26.98	32.20 / 29.93	35.22 / 32.69
0.6	FedAvg	45.97	45.97 / 45.97	46.30 / 45.05	29.75 / 11.12	45.97 / 45.97	41.91 / 37.58	47.08 / 46.54
	Quant.&Mask (8-bit, 0.02)	43.09	40.53 / 36.48	43.39 / 42.86	23.19 / 8.39	40.33 / 38.96	34.67 / 33.03	44.68 / 44.32
	Quant.&Mask (8-bit, 0.03)	40.76	35.21 / 30.83	41.70 / 40.62	20.91 / 9.87	35.37 / 34.74	33.62 / 31.38	42.82 / 42.44
	Quant.&Mask (8-bit, 0.05)	36.36	25.52 / 19.44	37.91 / 36.66	16.71 / 6.41	25.36 / 22.31	28.75 / 26.09	39.07 / 38.27
	CloakFL ($\rho = 0.6$)	43.75	29.84 / 24.18	37.37 / 32.85	27.45 / 15.53	41.33 / 38.90	40.05 / 35.96	43.36 / 42.35
	CloakFL ($\rho = 0.7$)	41.14	21.22 / 13.65	30.68 / 25.64	25.00 / 12.13	37.11 / 32.41	37.16 / 34.05	39.99 / 38.39

Table 12: Detailed final-round accuracy (%) for Tiny-ImageNet in the cross-silo setting.

β	Method	Server	Proxy	Single-client recovery		Multi-client recovery		
				Proxy FT	Proxy Train	Coll. Proxy	Coll. Train	Coll. FT
0.1	FedAvg	40.83	40.83 / 40.83	37.84 / 33.75	15.41 / 2.24	40.83 / 40.83	32.42 / 22.82	45.45 / 43.30
	Quant. (4-bit)	25.81	26.75 / 26.75	21.46 / 19.54	10.68 / 2.08	26.75 / 26.75	18.60 / 16.51	30.73 / 29.45
	Quant.&Mask (8-bit, 0.02)	37.64	34.38 / 33.02	34.67 / 31.76	13.76 / 1.79	34.46 / 34.14	26.49 / 19.51	42.62 / 41.54
	Quant.&Mask (8-bit, 0.03)	36.43	30.96 / 29.37	33.23 / 29.91	13.92 / 2.40	31.23 / 30.67	23.85 / 18.05	41.13 / 39.74
	Quant.&Mask (8-bit, 0.05)	32.88	23.04 / 15.84	29.91 / 26.29	12.06 / 2.44	23.42 / 20.68	20.23 / 16.30	38.58 / 37.05
	CloakFL ($\rho = 0.6$)	37.26	16.97 / 14.15	28.24 / 23.15	14.73 / 2.98	34.65 / 25.24	28.40 / 27.26	38.11 / 36.50
	CloakFL ($\rho = 0.7$)	34.21	12.49 / 6.59	23.40 / 19.52	13.76 / 6.37	30.35 / 20.27	27.07 / 25.47	34.02 / 31.23
0.3	FedAvg	48.94	48.94 / 48.94	48.81 / 47.23	26.96 / 13.19	48.94 / 48.94	42.12 / 38.60	52.18 / 51.43
	Quant.&Mask (8-bit, 0.02)	46.80	43.02 / 40.40	46.95 / 45.84	23.61 / 18.70	44.32 / 43.63	41.28 / 37.76	51.14 / 50.40
	Quant.&Mask (8-bit, 0.03)	46.87	41.63 / 36.47	45.92 / 44.72	21.70 / 18.37	41.95 / 39.53	40.07 / 33.90	51.16 / 49.50
	Quant.&Mask (8-bit, 0.05)	44.75	29.74 / 24.80	43.93 / 42.62	20.31 / 15.41	29.59 / 28.57	36.68 / 33.49	49.51 / 48.03
	CloakFL ($\rho = 0.6$)	47.88	29.50 / 8.07	40.01 / 37.42	24.62 / 21.23	45.18 / 37.26	40.35 / 38.93	48.35 / 46.41
	CloakFL ($\rho = 0.7$)	44.70	18.43 / 1.43	34.77 / 31.57	23.51 / 19.79	40.87 / 30.44	38.20 / 36.79	44.83 / 42.27
0.6	FedAvg	52.57	52.57 / 52.57	51.86 / 50.17	30.83 / 6.15	52.57 / 52.57	45.82 / 41.27	53.89 / 53.23
	Quant.&Mask (8-bit, 0.02)	51.57	47.67 / 44.32	50.62 / 49.33	26.27 / 4.65	48.88 / 46.95	40.63 / 34.39	53.37 / 52.62
	Quant.&Mask (8-bit, 0.03)	50.33	45.38 / 38.29	50.41 / 48.86	25.94 / 8.34	45.62 / 43.53	41.16 / 38.00	52.86 / 52.15
	Quant.&Mask (8-bit, 0.05)	49.04	35.09 / 26.97	48.27 / 46.99	23.44 / 9.71	36.07 / 33.16	39.49 / 36.29	51.35 / 50.83
	CloakFL ($\rho = 0.6$)	51.20	29.57 / 25.99	44.28 / 42.04	28.48 / 7.08	49.13 / 42.76	44.55 / 41.76	51.94 / 50.08
	CloakFL ($\rho = 0.7$)	49.66	24.02 / 15.60	40.39 / 38.26	27.19 / 8.81	46.65 / 38.23	43.01 / 41.11	49.71 / 47.03

F Proof of Theorem 3.1

F.1 Additional Notation

We follow the proof structure of the FedAvg convergence analysis in Li et al. (2020b) and develop the convergence proof for CloakFL in an analogous manner. Let N denote the total number of clients. At local step e of communication round r, each client i performs one step of stochastic gradient descent:

$$v_{r,i}^{(e+1)} = w_{r,i}^{(e)} - \eta_{r,e} \nabla F_i\left(w_{r,i}^{(e)}; \xi_{r,i}^{(e)}\right), \tag{8}$$

where $\eta_{r,e}$ is the learning rate and $\xi_{r,i}^{(e)}$ denotes a stochastic sample drawn from the local data distribution of client i. Within each communication round, the local model update rule is

$$w_{r,i}^{(e+1)} = v_{r,i}^{(e+1)}, \qquad e = 0, \ldots, E-1. \tag{9}$$

At the end of the round, the clean global model and the proxy models for the next round are given by

$$w_{r+1} = \frac{1}{N} \sum_{j=1}^{N} v_{r,j}^{(E)}, \qquad w_{r+1,i}^{(0)} = w_{r+1} + \delta_{r+1,i}. \tag{10}$$

At an aggregation step, the clean global model is the average of the final local updates. Then $\delta_{r+1,i}$ denotes the client-specific perturbation injected at the aggregation step.

We define the uniform averages of the intermediate and local models as

$$\bar{v}_r^{(e)} \coloneqq \frac{1}{N} \sum_{i=1}^{N} v_{r,i}^{(e)} \qquad \text{and} \qquad \bar{w}_r^{(e)} \coloneqq \frac{1}{N} \sum_{i=1}^{N} w_{r,i}^{(e)}.$$

We also define the aggregated stochastic gradient and the aggregated full gradient as

$$g_r^{(e)} \coloneqq \frac{1}{N} \sum_{i=1}^{N} \nabla F_i\left(w_{r,i}^{(e)}; \xi_{r,i}^{(e)}\right),$$

and

$$\bar{g}_r^{(e)} \coloneqq \frac{1}{N} \sum_{i=1}^{N} \nabla F_i\left(w_{r,i}^{(e)}\right).$$

Furthermore, for every communication round r, we define the aggregated perturbation as

$$\bar{\delta}_r \coloneqq \frac{1}{N} \sum_{i=1}^{N} \delta_{r,i}.$$

Then

$$\bar{w}_{r+1}^{(0)} = \bar{v}_r^{(E)} + \bar{\delta}_{r+1}.$$

F.2 Key Lemmas

We directly adopt Lemmas 1 and 2 from Li et al. (2020b). These lemmas characterize the progress of one-step local SGD and the variance of aggregated stochastic gradients, respectively. Since both results depend only on the local update dynamics and are independent of the server-side aggregation procedure, they remain valid for CloakFL despite the use of client-specific proxy models at aggregation steps.

Lemma F.1 (Results of one-step SGD)**.** *Assume that each local objective function F_i is μ-strongly convex and L-smooth. If $\eta_{r,e} \le 1/(4L)$, then*

$$\begin{aligned} \mathbb{E}\|\bar{v}_r^{(e+1)} - w^\star\|^2 \le (1 - \eta_{r,e}\mu)\, \mathbb{E}\|\bar{w}_r^{(e)} - w^\star\|^2 + \eta_{r,e}^2\, \mathbb{E}\|g_r^{(e)} - \bar{g}_r^{(e)}\|^2 + 6L\eta_{r,e}^2\Gamma \\ + 2\,\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} \|\bar{w}_r^{(e)} - w_{r,i}^{(e)}\|^2\right], \end{aligned} \tag{11}$$

where $\Gamma \coloneqq F^\star - \frac{1}{N}\sum_{i=1}^{N} F_i^\star \ge 0$.

Lemma F.2 (Bounding the variance)**.** *Assume Assumption 3.1. It follows that*

$$\mathbb{E}\|g_r^{(e)} - \bar{g}_r^{(e)}\|^2 \leq \frac{1}{N^2}\sum_{i=1}^{N}\sigma_i^2. \tag{12}$$

The only place where CloakFL differs from FedAvg is that, at an aggregation step, the server broadcasts client-specific proxy models by adding perturbations. In FedAvg, all local models are reinitialized to the same global model at each periodic aggregation step. In contrast, CloakFL introduces additional model dispersion, as each client receives a differently perturbed proxy model due to client-specific perturbations. We control this effect in the following lemma.

Lemma F.3. *Suppose that Assumptions 3.2 and 3.5 hold and that the learning-rate sequence is non-increasing and satisfies*

$$\eta_{r,0} \leq 2\eta_{r,e}, \qquad e = 0, \ldots, E-1.$$

It follows that

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\|\bar{w}_r^{(e)} - w_{r,i}^{(e)}\|^2\right] \leq 8\eta_{r,e}^2\Big((E-1)^2G^2 + D^2\Big). \tag{13}$$

Proof. Fix a communication round r and consider any $e \in \{0, \ldots, E-1\}$. We bound the client-level dispersion as follows:

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\|\bar{w}_r^{(e)} - w_{r,i}^{(e)}\|^2\right] \\
&\quad= \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\big\|(w_{r,i}^{(e)} - \bar{w}_r^{(0)}) - (\bar{w}_r^{(e)} - \bar{w}_r^{(0)})\big\|^2\right] \\
&\quad\leq \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\|w_{r,i}^{(e)} - \bar{w}_r^{(0)}\|^2\right] &(14)\\
&\quad= \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\big\|(w_{r,i}^{(e)} - w_{r,i}^{(0)}) + (w_{r,i}^{(0)} - \bar{w}_r^{(0)})\big\|^2\right] \\
&\quad\leq 2\,\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\|w_{r,i}^{(e)} - w_{r,i}^{(0)}\|^2\right] + 2\,\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\|w_{r,i}^{(0)} - \bar{w}_r^{(0)}\|^2\right] \\
&\quad\leq 2\eta_{r,0}^2(E-1)^2G^2 + 2\,\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\|\delta_{r,i} - \bar{\delta}_r\|^2\right] &(15)\\
&\quad\leq 2\eta_{r,0}^2(E-1)^2G^2 + 2\eta_{r,0}^2D^2 = 2\eta_{r,0}^2\Big((E-1)^2G^2 + D^2\Big) &(16)\\
&\quad\leq 8\eta_{r,e}^2\Big((E-1)^2G^2 + D^2\Big). &(17)
\end{aligned}$$

Here, equation 14 uses the fact that dropping the mean decreases the uniform variance. For equation 15, we use Assumption 3.2 and note that

$$w_{r,i}^{(0)} - \bar{w}_r^{(0)} = \delta_{r,i} - \bar{\delta}_r.$$

Inequality equation 16 uses Assumption 3.5. Finally, equation 17 uses the step-size relation $\eta_{r,0} \leq 2\eta_{r,e}$. □

F.3 One-step Recursion with Perturbed Aggregation

Define

$$\Delta_{r,e} \coloneqq \mathbb{E}\|\bar{w}_r^{(e)} - w^\star\|^2.$$

At the boundary between rounds r and $r+1$, we have

$$\bar{w}_{r+1}^{(0)} = \bar{v}_r^{(E)} + \bar{\delta}_{r+1}$$

and thus

$$\begin{aligned}\|\bar{w}_{r+1}^{(0)} - w^\star\|^2 &= \|\bar{v}_r^{(E)} - w^\star + \bar{\delta}_{r+1}\|^2 \\ &= \|\bar{v}_r^{(E)} - w^\star\|^2 + \|\bar{\delta}_{r+1}\|^2 + 2\langle \bar{v}_r^{(E)} - w^\star, \bar{\delta}_{r+1}\rangle. \end{aligned} \tag{18}$$

Since $\bar{v}_r^{(E)} - w^\star$ is $\mathcal{F}_{r+1}$-measurable, Assumption 3.3 implies that the cross term vanishes in expectation:

$$\begin{aligned}\mathbb{E}\Big[\Big\langle \bar{v}_r^{(E)} - w^\star, \bar{\delta}_{r+1}\Big\rangle\Big] &= \mathbb{E}\Big[\Big\langle \bar{v}_r^{(E)} - w^\star, \mathbb{E}\big[\bar{\delta}_{r+1} \mid \mathcal{F}_{r+1}\big]\Big\rangle\Big] \\ &= 0.\end{aligned}$$

Therefore, taking expectation in equation 18, applying Assumption 3.4, and using the non-increasing step size give

$$\Delta_{r+1,0} \le \mathbb{E}\|\bar{v}_r^{(E)} - w^\star\|^2 + \eta_{r+1,0}^2 \varepsilon_{\mathrm{agg}}^2 \le \mathbb{E}\|\bar{v}_r^{(E)} - w^\star\|^2 + \eta_{r,E-1}^2 \varepsilon_{\mathrm{agg}}^2. \tag{19}$$

Combining Lemma F.1, Lemma F.2, and Lemma F.3, we obtain, for $e = 0, \dots, E-2$,

$$\Delta_{r,e+1} \le (1 - \eta_{r,e}\mu)\Delta_{r,e} + \eta_{r,e}^2 (B + C), \tag{20}$$

and, at the round boundary,

$$\Delta_{r+1,0} \le (1 - \eta_{r,E-1}\mu)\Delta_{r,E-1} + \eta_{r,E-1}^2 (B + C), \tag{21}$$

where

$$B = \frac{1}{N^2}\sum_{i=1}^{N} \sigma_i^2 + 6L\Gamma + 8(E-1)^2 G^2, \qquad C = 8(E-1)^2 G^2 + 16D^2 + \varepsilon_{\mathrm{agg}}^2.$$

Finally, we complete the proof by solving recursions equation 20 and equation 21 with the step size

$$\eta_{r,e} = \frac{2}{\mu(\gamma + rE + e)},$$

where $\kappa = L/\mu$ and $\gamma = \max\{8\kappa, E\}$. This is identical to the final steps in Li et al. (2020b). In particular, sequentially unrolling the recursion over T communication rounds, corresponding to TE local SGD steps, yields

$$\Delta_{T,0} \le \frac{1}{\gamma + TE}\left(\frac{4(B+C)}{\mu^2} + \gamma \Delta_{0,0}\right).$$

Using L-smoothness and the initialization convention $\Delta_{0,0} = \|w_0 - w^\star\|^2$, this gives

$$\mathbb{E}\Big[F(\bar{w}_T^{(0)})\Big] - F^\star \le \frac{2\kappa}{\gamma + TE}\left(\frac{B+C}{\mu} + \frac{\mu\gamma}{4}\|w_0 - w^\star\|^2\right),$$

which completes the proof of Theorem 3.1.